



Deteksi Kesamaan Teks Jawaban pada Sistem Test *Essay* Online dengan Pendekatan Neural Network

I Made Suwija Putra¹, Putu Jhonarendra², Ni Kadek Dwi Rusjyanthi³

^{1,3} Program Studi Teknologi Informasi, Fakultas Teknik, Universitas Udayana

² Programmer PT Laksita Emi Saguna

¹ putrasuwija@unud.ac.id, ² jhonarendra@gmail.com, ³ dwi.rusjyanthi@unud.ac.id

Abstract

E-learning is an online learning system that applies information technology in the teaching process. E-learning used to facilitate information delivery, learning materials and online test or assignments. The online test in evaluating students' abilities can be multiple choice or essay. Online test with essay answers is considered the most appropriate method for assessing the results of complex learning activities. However, there are some challenges in evaluating students' essay answers. One of the challenges is how to make sure the answers given by students are not the same as other students answers or 'copy-paste'. This study makes a similarity detection system (Similarity Checking) for students' essay answers that are automatically embedded in the e-learning system to prevent plagiarism between students. In this paper, we use Artificial Neural Network (ANN), Latent Semantic Index (LSI), and Jaccard methods to calculate the percentage of similarity between students' essays. The essay text is converted into array that represents the frequency of words that have been preprocessed data. In this study, we evaluate the result with mean absolute percentage error (MAPE) approach, where the Jaccard method is the actual value. The experimental results show that the ANN method in detecting text similarity has closer performance to the Jaccard method than the LSI method and this shows that the ANN method has the potential to be developed in further research.

Keywords: Similarity Checker, Neural Network, LSI, Jaccard

Abstrak

E-learning adalah sistem pembelajaran daring yang menerapkan teknologi informasi dalam proses belajar mengajar. Fungsi e-learning adalah untuk mempermudah penyampaian informasi, memberi materi pembelajaran dan mengerjakan soal atau tugas. Penerapan kuis ujian dalam evaluasi kemampuan siswa dapat berupa pilihan ganda atau jawaban essay. Tugas dengan kuis essay dianggap sebagai metode yang paling tepat untuk menilai hasil kegiatan belajar yang kompleks. Namun, ada beberapa tantangan dalam mengevaluasi jawaban essay siswa. Salah satu tantangannya adalah bagaimana memastikan jawaban yang diberikan siswa tidak sama dengan jawaban siswa lain atau 'copy-paste' jawaban siswa lain. Penelitian ini membuat sebuah sistem pendeteksi kesamaan (Similarity Checking) untuk jawaban essay siswa secara otomatis tertanam dalam sistem e-learning untuk membantu mencegah plagiarisme antar sesama siswa dalam tugas yang dikerjakan. Dalam paper ini, kami menggunakan metode Artificial Neural Network (ANN), Latent Semantic Index (LSI), dan Jaccard untuk menghitung persentase kesamaan antar essay siswa. Teks essay diubah menjadi array yang mewakili frekuensi kata yang sebelumnya sudah dilakukan preprocessing data. Dalam penelitian ini, kami menggunakan hasil evaluasi dengan pendekatan mean absolute percentage error (MAPE), dimana metode Jaccard sebagai nilai aktualnya. Hasil percobaan menunjukkan bahwa metode ANN dalam pendeteksian kesamaan teks memiliki kinerja yang lebih mendekati metode Jaccard dibandingkan dengan metode LSI dan hal ini menunjukkan kedepannya metode ANN berpotensi untuk dikembangkan pada penelitian lebih lanjut.

Kata kunci: Deteksi Similarity, Neural Network, LSI, Jaccard

1 Pendahuluan

Perkembangan pesat teknologi mempengaruhi semua bidang kehidupan manusia. Teknologi membantu kehidupan manusia dalam menyelesaikan pekerjaan dengan cepat, efisien dan menekan human error.

Teknologi dalam perkembangan juga dapat dirasakan pada lingkup pendidikan yakni salah satunya teknologi informasi. Teknologi informasi pada dunia pendidikan dapat *mengubah* bentuk aktivitas yang selama ini secara tatap muka menjadi bisa secara jarak jauh [1]. Masa

pandemi Covid 19 secara tidak langsung membuat pemanfaatan teknologi informasi menjadi peran yang sangat penting guna mewujudkan aktivitas pembelajaran jarak jauh yang baik [2]. Salah satu teknologi informasi pembelajaran jarak jauh adalah dengan menggunakan sistem *e-learning*. Saat ini setiap elemen di bidang pendidikan telah menerapkan *e-learning* dalam proses pembelajarannya dan ini diproyeksikan akan terus digunakan pasca pandemic dengan konsep *hybrid* yang memadukan dengan pembelajaran tatap muka.

E-learning ke depannya memiliki tantangan untuk melengkapi fitur-fitur yang efektif dalam menyajikan dan menawarkan informasi yang akurat serta spesifik kepada pengguna, seperti batasan sistem, profil pengguna, fakta dan opini pengetahuan web pengguna, komunikatif dan kemampuan multitasking dalam satu arsitektur agar manfaat dasar dengan adanya teknologi informasi itu bisa benar-benar dirasakan [3][4]. Salah satu contoh fitur *e-learning* adalah manajemen materi pembelajaran yang dungguh oleh pengajar secara terstruktur sehingga bisa diakses oleh siswa dan terdokumentasi dengan baik. Fitur lainnya yang tidak kalah penting adalah proses evaluasi pembelajaran siswa, di sini pengajar mengatur waktu ujian, jenis ujian, membuat soal-soal ujian dan memberikan penilaian terhadap hasil ujian siswa.

Khusus untuk jenis ujian dapat berupa pilihan ganda ataupun *essay*. Umumnya, penggunaan pilihan ganda lebih banyak diterapkan pada ujian *online* karena lebih sederhana praktis dalam evaluasi jawaban siswa, karena sistem secara otomatis mencocokkan dengan kunci jawaban yang ada dan nilai bisa didapatkan secara cepat. Sedangkan pertanyaan *essay* dalam proses evaluasi jawaban harus mempertimbangkan beberapa faktor, salah satunya dalam evaluasi jawaban *essay* mengharuskan untuk membaca secara saksama jawaban *essay* berupa kalimat-kalimat karangan untuk dapat mengerti informasi yang tersampaikan pada jawaban siswa tersebut, sehingga bisa ditentukan apakah jawaban tersebut bernilai benar atau salah [5]. Selain itu proses pemberian nilai jawaban *essay* juga perlu mempertimbangkan kesamaan isi jawaban *essay* antar siswa. Hal ini untuk melihat tingkat kejujuran siswa sehingga bisa sebagai bahan pertimbangan dalam memberikan nilai.

Tugas deteksi tingkat kesamaan jawaban antar siswa secara teknologi bisa dimasukkan sebagai fitur dalam *e-learning* dengan menggunakan teknologi Natural Language Processing (NLP) pada sub bidang ilmu Information Retrieval (IR) yaitu tugas pengambilan informasi terkait dengan representasi, penyimpanan, akses peran ke informasi [6], sehingga pengajar dimudahkan dalam melihat tingkat kesamaan jawaban *essay* antar siswa

Deteksi kesamaan adalah teknik untuk mencari, menghitung, dan menemukan kesamaan dari dua atau lebih objek dengan *output* berupa persentase (0-100%). Deteksi kesamaan pada *IR* secara teknis memiliki beberapa langkah sebelum masuk ke metode utama. Tahap pertama adalah pengumpulan data atau akuisisi data. Tahap kedua adalah tahap *preprocessing* dan tahap yang terakhir adalah proses perhitungan tingkat kesamaan text dalam bentuk persentase.

Beberapa peneliti telah meneliti teknik pendeteksian kesamaan teks. Salah satu penelitian tentang deteksi kesamaan teks adalah artikel berjudul *Context similarity measure using fuzzy formal concept analysis* [7]. Penelitian ini mengusulkan metode empiris untuk memperkirakan kesamaan semantik menggunakan Analisis Konsep Formal Fuzzy. Pengelompokan pola leksikal yang berbeda memungkinkan untuk mewakili hubungan semantik antara dua kata secara akurat.

Studi lain yang berjudul *Learning Text Similarity with Siamese Recurrent Networks* [8] memperkenalkan konsep pembelajaran mendalam untuk mempelajari metrik kesamaan dalam variabel panjang urutan karakter. Model ini menggabungkan metode LSTM dengan arsitektur *Siamese*.

Penelitian lain tentang metode kesamaan teks adalah Rancang Bangun Aplikasi Deteksi Kesamaan Dokumen Teks Menggunakan Algoritma Ratcliff/Obershelp [9]. Penelitian ini menggunakan algoritma Ratcliff/Obershelp untuk menilai seberapa mirip pola dua string yang berbeda. Pada pengujian algoritma, pengujian dilakukan pada 12 dokumen dengan tingkat kesamaan dan posisi yang bervariasi. Nilai yang diperoleh mendekati nilai prediksi, dan nilai pada dokumen yang memiliki kesamaan yang sama tetapi bagian yang berbeda mendapatkan nilai yang berbeda.

Deteksi Plagiarisme Tingkat Kesamaan Judul Skripsi dengan Algoritma Winnowing [10] mengimplementasikan suatu algoritma untuk menentukan kesamaan teks, salah satu algoritma untuk menentukan kesamaan teks adalah algoritma Winnowing. Algoritma winnowing adalah metode yang digunakan untuk mendeteksi common subsequence dalam membandingkan dua teks atau lebih.

State of the art penelitian sebelumnya memberikan gambaran bahwa teknik pendeteksian kesamaan teks telah banyak dilakukan dengan berbagai macam metode mulai dari metode algoritma matematis seperti Ratcliff, Jaccard, Cosine Similarity, Winnowing dan LSI, sampai dengan saat ini menggunakan metode pendekatan *learning-based (machine learning dan neural network)*. Deteksi kesamaan teks menggunakan *neural network* memang mulai telah banyak dikerjakan namun khusus untuk *neural network* dengan arsitektur AND Operator belum pernah diteliti secara langsung untuk kasus text

jawaban ujian *essay* siswa pada sistem e-learning. Oleh karena itu pada penelitian ini dibuat sebuah sistem pendeteksi kesamaan jawaban siswa menggunakan metode utama yaitu *neural network* (NN) dengan pengembangan pengujian pada fungsi aktivasi dan penggunaan *multi hidden layer*. Sedangkan untuk melakukan evaluasi metode ANN, dalam penelitian ini menggunakan metode pembandingan lainnya, yaitu LSI dan Jaccard Similarity. Metode LSI ini digunakan dalam penelitian ini dengan alasan masih sedikit penelitian menggunakan metode LSI dalam mendeteksi kesamaan teks, sedangkan untuk penggunaan metode Jaccard Similarity lebih sebagai metode acuan hasil nilai tingkat kesamaan sebuah jawaban sehingga bisa mengukur kinerja antara metode ANN dengan LSI menggunakan pendekatan evaluasi MAPE. Hal ini dengan pertimbangan metode Jaccard Similarity terbilang metode yang sederhana tetapi cukup akurat untuk mendeteksi kesamaan teks secara lexical [11].

Adapun secara struktur yang dibahas pada paper ini adalah sebagai berikut: Bagian 2 metode yang digunakan dalam penelitian; Bagian 3 menyajikan hasil eksperimen yang didapat dan menganalisisnya; Bagian 4 menguraikan kesimpulan yang didapat.

2 Metode Penelitian

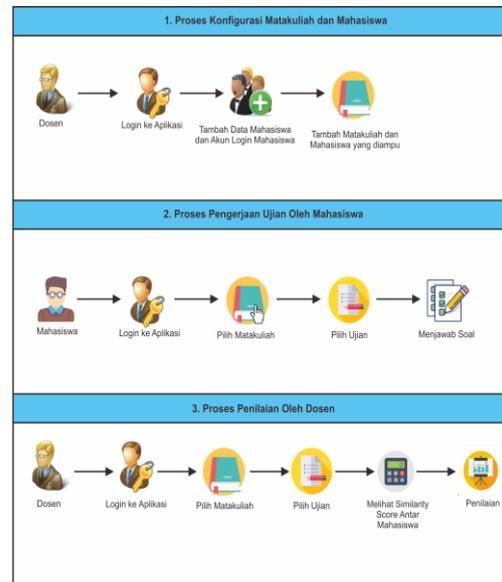
Bagian ini menjelaskan mengenai langkah-langkah proses penerapan teknologi kesamaan teks dengan metode *artificial neural network* (ANN) yang ditanamkan pada *prototype* sistem e-learning pada penelitian ini. Tujuan penelitian ini adalah menguji fungsionalitas dan kinerja dari metode ANN dalam menghasilkan persentase kesamaan dengan membandingkannya dengan metode lain yaitu Latent Semantic Indexing (LSI) yang dimana nilai persentase tingkat *similarity* metode Jaccard sebagai nilai acuan.

Gambar 1 adalah gambaran alur *prototype* sistem e-learning dalam proses menguji metode ANN dalam mengeluarkan hasil presentase kesamaan jawaban *essay* antar siswa. Terdapat 3 proses berbeda. Proses pertama adalah mengkonfigurasi mata pelajaran dan siswa, atau proses inisialisasi penggunaan aplikasi dimulai dengan pengajar membuat mata pelajaran. Pengajar mendaftarkan siswa ke mata pelajaran tersebut. Dalam setiap mata pelajaran, pengajar dapat membuat sesi tes online dan pertanyaan yang akan dijawab oleh siswa.

Proses kedua adalah pengerjaan ujian oleh siswa. Siswa yang mengikuti ujian memulai dengan login ke sistem, dimana aplikasi akan menampilkan halaman dashboard siswa. Seorang siswa memilih mata pelajaran dan sesi tes online. Kemudian, siswa menjawab setiap pertanyaan yang diberikan secara langsung dalam sistem.

Tahap terakhir adalah proses penilaian oleh pengajar.

Proses ini dilakukan oleh pengajar dengan memberi nilai jawaban siswa berdasarkan tingkat kesetaraan jawaban siswa yang dihitung dengan menggunakan perbandingan tiga metode yaitu ANN, LSI, dan Jaccard.



Gambar 1 Alur sistem deteksi kesamaan jawaban pada aplikasi e-learning

Dalam penelitian ini, menggunakan metode ANN yang dikerjakan melalui empat tahap yaitu akuisisi data, *preprocessing*, dan pengukuran kesamaan teks. Akuisisi data adalah tahap pengambilan data, *preprocessing* adalah tahap pembersihan data, *processing* atau kalkulasi adalah proses inti dalam memberikan keputusan tentang tingkat persentase kesamaan antar teks.

2.1 Pengolahan Data

Pada penelitian ini pengolahan data dilakukan untuk menyiapkan data *input* yaitu data teks agar bisa dibaca oleh metode deteksi kesamaan text (ANN, LSI, dan Jaccard) untuk diterjemahkan hasilnya berupa presentase tingkat kesamaan. Proses dilakukan secara bertahap hingga berhasil *mengubah* data text menjadi deretan nilai numerik berupa *array*. Adapun tahapan pengolahan data adalah sebagai berikut

2.1.1 Akuisisi Data

Penelitian ini menyiapkan data *input* yang bersumber dari jawaban *essay* siswa pada sebuah sistem e-learning yang sudah berjalan di suatu universitas. Proses akuisisi data dikerjakan dengan menyalin dan *menginput* data secara manual ke dalam database sistem tes online yang dibangun pada penelitian ini. Metode *input* manual dipilih dengan pertimbangan agar pada saat proses pengujian metode pada penelitian ini tidak mengganggu jalannya sistem e-learning universitas yang sedang beroperasi.

Data jawaban essay siswa disimpan dalam database MySQL dalam format raw text atau teks mentah sehingga tidak memerlukan konversi dokumen saat mengekstrak teks. Contoh data *raw text* jawaban siswa dapat dilihat pada Tabel 1.

Tabel 1. Contoh jawaban *ssay siswa* pada soal yang sama

Nama	Text corpus
Septiana Eka Arwanda Riski	Peran Pihak Internal yaitu untuk menjalankan perusahaan dengan memanfaatkan sumber daya yang ada secara efektif. Peran Pihak Eksternal yaitu untuk mendukung keberlangsungan usaha.
I Gusti Agung Surya Pramana w	ligkungan internal bekerja dengan baik dalam lingkungan bisnis dalam lingkungan tersebut external bekerja di lingkungan tidak resmi yang bertugas dalam memximalkan bisnis
Agung Adhika Mas Pratama	Lingkungan Internal -> Tenaga kerja,modal dan peralatan. Peran dari lingkungan Internal ini untuk menentukan kekuatan perusahaan dan juga mengetahui kelemahan perusahaan. Lingkungan Eksternal -> Konsumen,Pesaing, Ekonomi. Peran dari lingkungan Eksternal ini untuk menentukan peluang pasar dan juga ancaman dari luar/pesaing
Made Satya Wibawa Esa Narendra	Lingkungan Internal (Dalam) contohnya karyawan, manager, CEO dan lain-lain yang memiliki peran langsung (direct) untuk mempengaruhi nasib perusahaan Lingkungan Eksternal (Luar) contohnya Konsumen, pesaing, pemasok dan lain-lain yang memiliki peran bahwa kekuatan lingkungan luar tersebut memiliki potensi mempengaruhi kinerja atau keberlangsungan perusahaan.

Tabel 1 merupakan contoh *raw text* dari 5 siswa pada satu soal. Secara keseluruhan, data *essay* diambil dari 2 sesi ujian dari 2 matakuliah yang berbeda. Dengan total terdapat 290 *essay* dengan ukuran 398 KB di dalam database.

Raw text berguna untuk tahap pemrosesan teks karena jenis data ini tidak memiliki format yang tidak diperlukan dalam analisis teks, seperti format teks cetak tebal, ukuran font, dan warna teks.

2.1.2 Preprocessing

Tahap *preprocessing* dilakukan sebelum sistem mengeksekusi proses utama. Tahap *preprocessing* dalam penelitian ini menggunakan data teks corpus jawaban siswa yang yang sebelumnya tersimpan di database sistem tes online. Alasan utama dari tahap *preprocessing* adalah sebagai proses membersihkan data agar data lebih sempurna untuk diolah pada proses utama dengan sedikit *noise*. Alur tahapan *preprocessing* seperti terlihat pada **Error! Reference source not found.**

Tahap akhir dari *preprocessing* akan membentuk TF (*Term Frequency*) dengan tipe data *array* dari kata-kata dasar yang ada di dalam *inputan* corpus jawaban *essay* siswa agar bisa digunakan sebagai nilai *input* untuk algoritma *similarity* yang digunakan pada penelitian ini.

Preprocessing dapat mempengaruhi kinerja metode ANN, LSI, dan Jaccard saat mendeteksi kesamaan teks. Proses *preprocessing* juga sama dilakukan pada text corpus jawaban siswa yang akan diperbandingkan. Berikut ini detail tahapan dari *preprocessing*.



Gambar 2. Flow chart preprocessing

2.1.2.1 Tokenisasi

Langkah pertama dalam tahap *preprocessing* adalah tokenisasi, yaitu mengambil fitur kata, frasa, atau simbol terdapat dalam teks [12] yang dinamakan sebagai token. Metode deteksi kata atau *word tokenization* mengekstraksi kata dalam teks dengan cara sederhana yaitu menandai kelompok huruf yang dikelilingi dengan spasi untuk dipisahkan sebagai sebuah token [13].

Dalam pemrograman, token-token tersebut disajikan dalam bentuk *array index*. Pada penelitian ini karakter dan simbol khusus dihilangkan karena dalam mendeteksi kesamaan teks, hanya perlu menghitung kata-kata yang sama antara dua dokumen atau lebih.

2.1.2.2 Stop word

Hasil dari proses tokenisasi yang berupa kumpulan kata dalam *array*, diproses lebih lanjut dengan menghilangkan kata umum yang sering muncul (*stop word*), namun tidak perlu diperhitungkan untuk mendapatkan konteks utama dalam sebuah teks. Contoh yang *stop word* pada Bahasa Indonesia adalah 'yang', 'di', 'untuk', dan 'dari' [14].

Pada penelitian ini *stop word* diperlukan, karena jika kata-kata umum atau konjungsi yang sering muncul

tidak dihilangkan, maka persentase kesamaan text akan cenderung tinggi sehingga mengganggu keakuratan metode *similarity*. Kata-kata umum yang sering muncul bisa disebut *noise* dalam *text processing*.

2.1.2.3 Stemming

Stemming adalah teknik pencarian kata dasar dari kata yang sudah berimbuhan. Dalam Bahasa Indonesia imbuhan ada berupa awalan, akhiran, konfiks, dan sisipan dalam kata [15].

Penelitian ini menggunakan proses *stemming* untuk bisa mendapatkan kata dasar dari token yang terkumpul pada setiap text jawaban siswa agar lebih tepat dicocokkan secara sintaksis. Misalnya, text jawaban siswa 'A' memiliki kata 'memukul', sedangkan text jawaban siswa 'B' memiliki kata 'dipukul'. Setelah proses *stemming*, Corpus A dan B akan memiliki 'pukul' karena 'pukul' adalah kata dasar. Dengan demikian, sistem deteksi kesamaan teks akan mengatakan kedua dokumen sangat mirip, walaupun berbeda bentuk imbuhan kata. Pada *stemming* Bahasa Indonesia, teknik terkenal digunakan adalah Sastrawi. Sastrawi mampu mengurangi *over stemming* dan *under stemming* serta waktu pengerjaan lebih cepat dibandingkan *library* lain [16][17].

2.1.2.4 Term Frequency

Hasil dari kumpulan kata-kata dasar yang didapat dari proses *stemming* kemudian dihitung frekuensi kemunculannya dalam sebuah dokumen. Teknik ini disebut dengan istilah *term frequency* (TF). TF digunakan untuk mendapatkan data terkuantisasi untuk diproses lebih lanjut pada tahap perhitungan kesamaan teks.

TF juga merupakan metode untuk mengubah kata-kata yang telah diproses sebelumnya menjadi *array*. konversi kata menjadi nilai *array* dilakukan dengan menghitung jumlah kemunculan kata (t) ke i dalam suatu corpus (d) ke j seperti pada persamaan 1 dan mengubahnya menjadi angka sesuai dengan jumlah kemunculannya.

$$W_{TF}(t_i, d_j) = f(t_i, d_j) \quad (1)$$

Pada penelitian ini tahap TF menghasilkan dua nilai *array* yaitu nilai perbandingan jawaban *essay* siswa (*text query*) dan nilai frekuensi jawaban *essay* siswa perbandingan (*indexed object*) yang digunakan sebagai *input* untuk metode Jaccard dan ANN.

Sedangkan untuk metode LSI, diperlukan proses lanjutan data *input* berupa frekuensi kata yang sudah terhitung bobot setiap kata untuk satu corpus atau yang lebih dikenal dengan Term Frequency - Inverse Document Frequency (TF-IDF). IDF langkah untuk kurangi bobot kata yang sering muncul di corpus dan tambah bobot kata yang jarang terjadi [18] dengan menggunakan persamaan 2.

$$W_{IDF}(t_i, d_j) = 1 + \log \left(\frac{D}{d(t_i)} \right) \quad (2)$$

$W_{IDF}(t_i, d_j)$ adalah nilai IDF dari term ke i pada corpus ke j , D adalah jumlah keseluruhan corpus dan $d(t_i)$ adalah jumlah corpus yang mengandung term ke i . dari hasil IDF tersebut kemudian dilakukan perkalian dengan TF seperti pada persamaan 3.

$$W_{TF-IDF}(t_i, d_j) = W_{TF}(t_i, d_j) * W_{IDF}(t_i, d_j) \quad (3)$$

$W_{TF-IDF}(t_i, d_j)$ adalah nilai TF-IDF dari term ke i pada corpus ke j .

2.1.3 Perhitungan Kesamaan

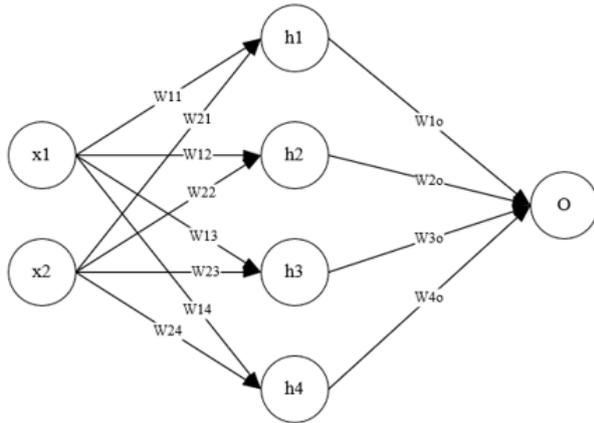
Proses utama untuk menghitung kesamaan teks pada penelitian ini mengambil data masukan dari proses sebelumnya yaitu TF yang menghasilkan deretan 2 nilai frekuensi kata dalam format *array*. Kedua nilai *array* ini dimasukkan ke dalam ANN sebagai *input* data untuk mencari nilai prediktif kesamaan teks untuk kedua *input* tersebut.

Proses pendeteksian kesamaan teks ini juga membandingkan hasil dari metode LSI dan Jaccard, yang merupakan metode terkenal untuk teknik kesamaan teks. Perbandingan ini membandingkan persentase nilai kesamaan yang dihasilkan oleh metode ANN dan LSI dengan Jaccard sebagai nilai referensi untuk tingkat kesamaan yang baik.

2.1.3.1 Artificial Neural Network (ANN)

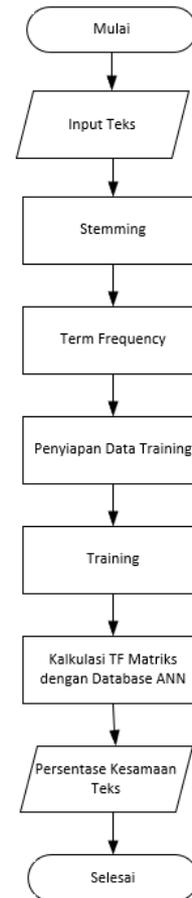
Metode *Artificial Neural Network* (ANN) merupakan metode utama yang diterapkan dalam penelitian untuk mengatasi permasalahan bagaimana mendeteksi kesamaan jawaban siswa pada sistem *e-learning*. Neuron pada ANN mencoba mereplikasi struktur dan perilaku *neuron* biologis manusia. Sebuah *neuron* terdiri dari *input* (*dendrit*) dan *output* (*sinaps* melalui *akson*). Setiap *neuron* memiliki fungsi yang menentukan aktivasi *neuron* [19]. Alur proses dari metode ANN dapat dilihat pada Gambar 3.

Alur proses metode ANN untuk menghitung kesamaan teks terbagi menjadi dua tahap. Dua tahap tersebut adalah pelatihan (*training*) dan pengujian (*testing*). Pada tahap pelatihan metode ANN dimulai dengan meng-*import library* dan menginisialisasi arsitektur *Neural Network* (NN). Penelitian ini menggunakan dua node pada *input layer*, satu *hidden layer* dengan empat node dan satu node pada *output layer* seperti terlihat pada

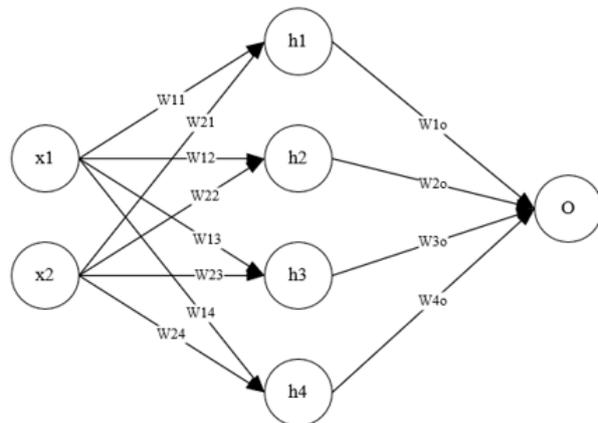


Gambar 4.

TF dua corpus akan mengisi lapisan *input* yaitu node x_1 dan x_2 . *Input* x_1 merupakan *query* atau objek data teks siswa yang diuji, sedangkan x_2 merupakan *input* pembanding atau data teks siswa pembanding. Setelah itu, vektor kalimat ini diproses pada lapisan tersembunyi dengan empat neuron (h_1, h_2, h_3 , dan h_4) menggunakan operasi TANH (Tangent Hyperbolic) sebagai fungsi aktivasi yang menghitung data *input* dengan bobot yang diinisialisasi terlebih dahulu.



Gambar 3. Proses Perhitungan Kesamaan Teks dengan Artificial Neural Network



Gambar 4. Arsitektur *neural network* dalam penelitian ini

Proses ini kemudian akan mengulangi proses ini untuk mendapatkan bobot terkoreksi (W). Hasil perhitungan hidden *layer* dengan bobot (W_{10}, W_{20}, W_{30} , dan W_{40}) yang dihasilkan pada proses kedua kemudian menjadi nilai *output* (o). Proses ini dilakukan untuk membandingkan nilai prediksi dalam jaringan dengan setiap sampel data (x) sehingga dapat memodifikasi hubungan bobot (w) dalam jaringan untuk bisa

mendapatkan nilai prediksi yang lebih dekat dengan nilai sebenarnya [20][21].

ANN dalam penelitian ini menggunakan konsep AND Operator. Konsep AND Operator adalah memberikan keputusan antara dua nilai yang bernilai sama dan berbeda. Neural Network dengan AND Operator mampu memberikan keputusan tidak mutlak atau persentase sehingga jawaban dari proses kesamaan berupa persentase bukan ‘ya’ atau ‘tidak’.

2.1.3.2 Latent Semantic Indexing (LSI)

LSI adalah metode berbasis ruang vektor yang dapat membuat asosiasi antara dokumen yang terkait secara konseptual [22]. Metode ini mencakup teknik pengindeksan dan pencarian yang menggunakan fungsi matematika untuk mengidentifikasi hubungan antara istilah dan konsep dalam kumpulan teks yang tidak terstruktur. Teknik aljabar matriks yang disebut Singular Value Decomposition menghasilkan matriks dengan dimensi yang jauh lebih sedikit [23].

LSI adalah prinsip algoritmik yang digunakan oleh mesin pencari untuk memproses sinonim untuk memahami hubungan topik antara kata-kata [24]. Proses *Singular Value Decomposition* ditunjukkan pada

Gambar 5.

$$A \approx USV^T$$

A: m x n matrix
 U: m x m orthogonal matrix
 S: m x n diagonal matrix
 V: n x n orthogonal matrix

Gambar 5. Singular Value Decomposition

LSI bertujuan untuk mendapatkan pemodelan yang efektif untuk merepresentasikan hubungan antara kata kunci dan dokumen yang dicari. Dari kumpulan kata kunci yang sebelumnya tidak lengkap dan tidak kompatibel ke sekelompok objek terkait.

2.1.3.3 Jaccard Similarity

Dalam tujuan pengelompokan dokumen, fungsi Koefisien Jaccard adalah salah satu metode yang terkenal. Jaccard juga dapat menghitung tingkat kesamaan antara dua kumpulan data yang disebut The Jaccard Index [25]–[27]. Persamaan untuk menghitung kesamaan dengan Jaccard Index dapat dilihat pada Persamaan 4.

$$\text{Similarity}(X, Y) = \frac{|X \cap Y|}{|X| + |Y| - |X \cap Y|} \quad (4)$$

Seperti terlihat dari Persamaan 4, untuk menghitung kesamaan antara 2 data (x dan y), yaitu gabungan x dan

y dibagi banyaknya x dan y dikurangi gabungan x dan y. Keluaran dari Indeks Jaccard memiliki tingkat kesamaan berkisar dari 0% hingga 100%.

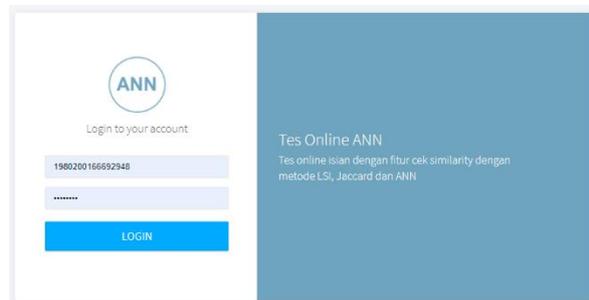
3 Hasil dan Pembahasan

3.1 Fungsional Sistem

3.1.1 Hak Akses Sistem

Sistem Tes Online dengan Deteksi Kesamaan Jawaban *Essay* Siswa yang dikembangkan dalam penelitian ini memiliki dua jenis akses yaitu Pengajar dan Siswa. Hak akses Pengajar memiliki kewenangan dalam sistem yang lebih luas daripada hak akses sebagai Siswa.

Sistem ini berbasis web yang bisa diakses melalui *web browser* dengan halaman *login* sebagai tampilan utama sistem ini, seperti terlihat pada Gambar 6.



Gambar 6. Halaman login

Setelah berhasil *login*, pengguna akan masuk ke *dashboard* dan dapat memilih mata pelajaran dan memilih menu lainnya yang sesuai dengan akses yang digunakan untuk masuk ke sistem.

3.1.2 Registrasi Master Data Siswa, Mata Pelajaran dan Ujian

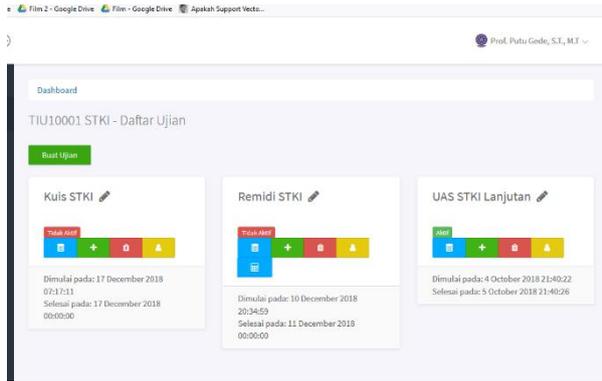
Master data siswa berfungsi untuk mendaftarkan data identitas siswa yang akan menggunakan e-learning ini oleh Pengajar. Pendaftaran siswa dilakukan dengan mengisi form Nama, NIM, dan Password.

Siswa yang telah terdaftar dapat melakukan *login* dan pendaftaran mata pelajaran yang akan diujikan, tentunya sebelumnya sudah dibuatkan master mata pelajaran dan data ujian oleh user Pengajar.

3.1.3 Membuat sesi ujian

Setelah mendaftarkan siswa ke dalam mata pelajaran yang diambil, Pengajar harus membuat sesi ujian. Pembuatan sesi ujian tes berfungsi untuk membuat sesi ujian dalam suatu mata pelajaran agar ujian secara *online* dapat dilakukan sesuai dengan waktu yang telah ditentukan.

Jika berhasil memasuki sesi ujian, kumpulan sesi ujian setiap siswa muncul seperti ditampilkan pada Gambar 7.



Gambar 7. Tampilan Sesi Ujian

Di sesi ujian, ada pilihan menu seperti yang ditampilkan pada Gambar 7. Fungsi pada tiap menu dapat dilihat pada Tabel 2.

Tabel 2. Fungsi Menu Sesi Ujian

No.	Menu	Fungsi
1.		Melihat detail ujian seperti data soal dan waktu mulai ujian.
2.		Menambah soal pada ujian yang dipilih
3.		Menghapus data ujian yang dipilih
4.		Melihat jawaban dan persentase kesamaan jawaban antar siswa
5.		Menu untuk melakukan proses kalkulasi persentase kesamaan jawaban antar siswa

Tampilan menu dengan bentuk *icon* dibuat untuk memudahkan pengguna memahami maksud dari menu.

3.1.4 Menghitung Kesamaan Jawaban Siswa

Pengukuran kesamaan jawaban siswa dihitung dengan menggunakan tiga metode yaitu ANN, LSI, dan Jaccard. Metode tersebut akan menghitung kesamaan jawaban setelah admin atau Pengajar menekan tombol hitung kesamaan pada sesi ujian. Khusus metode ANN diperlukan proses pelatihan untuk membentuk model dengan bobot ideal dan dipakai untuk melakukan proses pencocokan.

Setelah proses pelatihan selesai, sistem akan secara otomatis menguji kesamaan teks jawaban antara satu siswa dengan siswa lainnya dalam satu sesi ujian. Hasil kesamaan akan muncul berdasarkan metode yang digunakan untuk menghitung kesamaan, seperti terlihat pada Tabel 3.

Tabel 3 menunjukkan pengajar yang menggunakan sistem ini dapat langsung melihat kesamaan jawaban antara satu siswa dengan siswa lainnya. Para Pengajar dapat menggunakannya sebagai salah satu pertimbangan

penting dalam menilai *essay* siswa dengan melihat kejujuran siswa dalam menjawab.

Tabel 3. Hasil Persentase Kesamaan Teks Pada Tiap Metode

ID	Nama Siswa	LSI	Jaccard	Neural Network
A	Agung Adhika Mas Pratama	24.6%	20.5%	24.2%
B	Made Satya Wibawa Esa Narendra	33.1%	25.0%	31.2%
C	I Putu Gede Adi Khusuma	30.4%	23.9%	28.4%
D	Kadek Effa Krisma Buriartha	33.6%	19.6%	26.8%
E	Kadek Lanang Mahardika	26.4%	26.7%	32.0%
F	I Gede Widya Antar Nusa	18.8%	24.6%	27.7%
G	Diego Tristan Nababan	23.2%	17.9%	22.2%
H	Angga Destia Faturrahman	35.2%	23.9%	28.4%
I	I Gede Nyoman Ambara Yasa	27.2%	18.3%	25.0%
J	Jibril Ali Akbar	29.9%	26.1%	30.8%
K	Michael Shummaecer Purba	21.5%	18.9%	23.0%
L	Kadek Eka Yuda Trisna S	36.3%	35.8%	40.2%
M	Adib humaid wajjih	30.6%	24.7%	27.6%
N	I Dewa Gede Suryadiantha Wedagama	33.1%	27.3%	31.4%
O	Fahri Choirul Anam	29.8%	26.9%	32.4%
P	Gusti Arya Agung Adi Rahaditya	24.1%	17.5%	22.3%
Q	Ni Made Tara Okta Adriana	39.7%	27.0%	33.0%
R	Ketut Mediana Ayu Candrayani	42.3%	32.7%	38.3%
S	I Gusti Agung Surya Pramana w	19.5%	28.8%	29.8%
MAPE (%)		28.0%	0.0%	20.0%

3.2 Experimental Setup

Proses pengujian metode *similarity* dengan ANN, LSI, dan Jaccard menggunakan data *input* berupa corpus yang diambil dari ujian *essay* siswa dari salah satu program studi di universitas X. Dalam sesi ujian *essay* dibuat lima pertanyaan yang sudah dijawab oleh siswa.

Jumlah data yang digunakan adalah 100 corpus jawaban *essay*. Khusus untuk metode ANN yang memerlukan data latih diambil dari corpus jawaban siswa lainnya yang diperbandingkan dalam satu sesi ujian.

Dalam proses pelatihan, metode ANN menggunakan beberapa parameter. Konsep operator AND digunakan dalam pembuatan fitur *knowledge* data latih. Nilai 0 ketika bertemu 1 memiliki *output* 0, nilai 1 dan 1 memiliki 1. Nilai TF dikonversi menjadi -1; ...; 1 karena *library neural network* yang digunakan menggunakan nilai *input* dengan range -1 sampai 1. Parameter yang digunakan dalam pelatihan dapat dilihat pada

Tabel 4.

Tabel 4. Nilai yang digunakan pada Parameter ANN

Parameter	Value
Learning Rate	0.1
Momentum	0.2
Round	1000

Error Rate	0.01
Epoch	100

Learning rate adalah nilai yang mempengaruhi akurasi dan *output* yang diharapkan dari setiap *epoch*. *Learning rate* juga menentukan berapa banyak pembaruan yang dilakukan pada bobot selama pelatihan. Momentum adalah parameter yang menentukan banyak nilai yang dapat ditoleransi dalam memenuhi suatu kondisi. *Round* adalah jumlah pengulangan pelatihan (algoritma *Backpropagation*). *Epoch* adalah jumlah iterasi dari *Round*.

Penentuan value dari tiap parameter ditentukan berdasarkan percobaan yang sebelumnya dilakukan dengan ketentuan dasar nilai parameter disesuaikan pada data yang hendak dilatih. Hasil pelatihan yang memberikan nilai terbaik ditetapkan sebagai parameter dalam metode ANN. Setelah proses pelatihan, nilai NN dicari berdasarkan data pelatihan dengan bobot yang telah diperoleh dan memenuhi kondisi di bawah *Error Rate* yang ditentukan yaitu 0.01. Kemudian, menyimpan nilai NN ke *database*. Nilai TF yang diperoleh di-*inputkan* ke dalam NN untuk mendapatkan nilai kesamaan jawaban. Hasil *output* metode ANN adalah persentase kesamaan antar dua buah text.

Dalam penelitian ini proses evaluasi hasil menggunakan konsep evaluasi metode MAPE (*Mean Absolute Percentage Error*), yaitu dengan membandingkan nilai yang dihasilkan dari masing-masing metode (*forecast*) dengan nilai referensi yang ditentukan (*actual*), seperti yang ditunjukkan pada Persamaan 5.

$$MAPE = \left(\frac{1}{n} \sum \frac{|Actual - Forecast|}{|Actual|} \right) * 100 \quad (5)$$

Kami memilih metode MAPE sebagai metode evaluasi kinerja metode ANN dalam tugas mengukur *similarity text*, karena tugas pedeteksian *similarity text* dalam penelitian lebih kepada proses untuk memprediksi hasil kesamaan antar teks jawaban mahasiswa. Besar hasil persentase tingkat kesamaan antar jawaban mahasiswa tentunya tidak diketahui sebelumnya besaran hasilnya dan hasil persentase tingkat kesamaan text dalam penelitian ini tergantung dari jawaban mahasiswa yang diperbandingkan, data latih jawaban mahasiswa dan pendekatan metode yang digunakan. Sehingga secara proses hal ini tidak sama seperti tugas klasifikasi objek, yang dimana sudah ditentukan sebelumnya label untuk setiap data latih maupun ujinya. sehingga jika menggunakan metode evaluasi *confusion matrix* menjadi tidak relevan lagi dalam penelitian ini. Oleh karena itu, menurut kami MAPE bisa sebagai opsi dalam mengukur kinerja metode dengan menghasilkan nilai *error* dari metode ANN dan LSI terhadap nilai *actual* yang dijadikan acuan yaitu metode Jaccard Similarity.

Metode Jaccard pada penelitian ini digunakan sebagai acuan atau nilai aktual untuk mengukur metode lainnya,

hal ini berdasarkan pertimbangan pada studi-studi yang ada, metode Jaccard Similarity adalah metode yang populer dan masih memiliki kinerja yang efektif dan baik untuk menentukan kesamaan teks dengan fitur sintaksis kata [29] [30]. Sehingga berdasarkan Persamaan 5, variabel *actual* adalah nilai semua data tingkat kesamaan yang dihasilkan oleh metode Jaccard. Variabel *forecast* adalah nilai dari setiap tingkat kesamaan data dengan metode LSI dan ANN. Kemudian, nilai tersebut dirata-ratakan dan digunakan sebagai persentase MAPE.

Ada beberapa percobaan yang akan dilakukan untuk menguji metode ANN dalam melakukan tugas mendeteksi kesamaan teks, yaitu percobaan utama dengan menggunakan data latih konversi 4 nilai TF dan data latih konversi 8 nilai TF yang sama-sama menggunakan fungsi aktivasi Tanh pada *hidden layer* (hasil dapat dilihat pada bagian 3.3). Selain itu ada percobaan tambahan yang mengujikan pengaruh fungsi aktivasi pada ANN terhadap tugas mendeteksi kesamaan teks, percobaan ini membandingkan hasil dari fungsi Tanh, ReLU, dan Sigmoid (hasil dapat dilihat pada bagian 3.4).

3.3 Hasil Percobaan Utama

Hasil utama penelitian ini berfokus seberapa dekat hasil pencocokan kata dalam corpus teks jawaban siswa yang satu dengan jawaban siswa yang lain dalam satu sesi ujian dengan menggunakan metode ANN dibandingkan dengan metode LSI. Objek siswa yang akan diukur tingkat kesamaannya dinamakan dengan istilah “Query”. Sedangkan istilah “Input” digunakan untuk objek siswa yang digunakan untuk perbandingan dengan “Query”.

Misalnya, Dalam pengujian penelitian ini akan mencari tingkat kesamaan jawaban dari siswa bernama Septiana Eka Arwanda Riski, maka jawaban siswa tersebut dilabeli sebagai “Query” (x1), kemudian dilakukan proses *preprocessing*. Hasil *preprocessing* dapat dilihat pada

Gambar 8.

Query
 rencana, planning, adalah, giat, yang, laku, untuk, tentu, arah, tuju, dan, target, usaha, dengan, susun, strategi, untuk, tentu, apa, saja, sumber, daya, yang, butuh, dalam, capai, tuju, sama, organisasi, organizing, adalah, giat, untuk, tentu, bagaimana, susun, organisasi, usaha, dalam, tentu, job, description, dan, wenang, dalam, anggota, usah, dalam, jalan, aktivitas, yang, sudah, rencak, arah, directing, adalah, giat, untuk, tuntun, motivasi, dan, beri, pengarah, kepada, anggota, usaha, dalam, jalan, giat, dan, aktivitas, apa, saja, dalam, capai, tuju, sama, jadi, dalam, arah, ini, butuh, orang, leader, controlling, awas, adalah, giat, untuk, memonitoring, dari, tiap, giat, yang, laku, untuk, kendali, jika, jadi, simpang, dalam, jalan, giat, dan, baik, dari, simpang, yang, sudah, jadi,

Gambar 8. Hasil Preprocessing Query

Sedangkan yang menjadi label “*Input*” (x2) adalah masing-masing teman dari “Septiana Eka Arwanda Riski”, misalnya jawaban siswa lain yang bernama “Agung Adhika Mas Pratama”. Proses *preprocessing* juga dilakukan pada label “*Input*”. Contoh hasil *preprocessing* jawaban “Agung Adhika Mas Pratama” terlihat pada Gambar 9.

Input
 1, rencana, -, fungsi, yaitu, proses, tentu, tuju, dan, buat, strategi,
 untuk, capai, tuju, ingin, 2, organisasi, -, fungsi, yaitu, tentu, struktur,
 organisasi, yang, tepat, untuk, capai, tuju, yang, ingin, 3, arah, -,
 fungsi, yaitu, bagai, tanggung, jawab, jalan, tuju, organisasi, 4,
 kendali, -, fungsi, yaitu, pasti, bahwa, giat, yang, rencana, sesuai,
 dengan, target, tuju,

Gambar 9. Hasil Preprocessing *Input*

Dari dua *input* yang ada tersebut, kemudian sistem menghitung TF untuk setiap suku kata di masing-masing corpus seperti pada Persamaan 1. Setelah itu, sistem menandai untuk setiap kata yang beririsan dengan melihat kata yang sama muncul di kedua *input*. Hasilnya ditampilkan pada Tabel 5.

Tabel 5. Irisan Kata

Kata	Query	<i>Input</i>	Irisan
rencana	1	2	1
giat	7	1	1
yang	5	3	1
untuk	6	2	1
tentu	4	2	1
arah	3	1	1
tuju	3	5	1
dan	5	1	1
target	1	1	1
dengan	1	1	1
strategi	1	1	1
capai	2	2	1
organisasi	2	3	1
jalan	3	1	1
kendali	1	1	1

Proses selanjutnya menghitung tingkat kesamaan sesuai dengan masing-masing metode yaitu ANN, LSI, dan Jaccard. Proses pendeteksian kesamaan teks menggunakan metode ANN memiliki pendekatan yang berbeda dengan LSI dan Jaccard. Proses pelatihan diperlukan untuk dapat menghasilkan nilai pencocokan kata dengan fungsi AND. Hasil pelatihan menghasilkan nilai bobot paling optimal untuk operasi AND.

Terdapat 2 percobaan dalam mengukur kinerja dari metode ANN. Seperti yang telah disebutkan sebelumnya, fitur pembeda antara teks yang satu dengan yang lainnya adalah jumlah kemunculan kata. Sehingga nilai TF menjadi *input* pada ANN untuk dilakukan pencocokan.

3.3.1 Konversi 4 Nilai TF

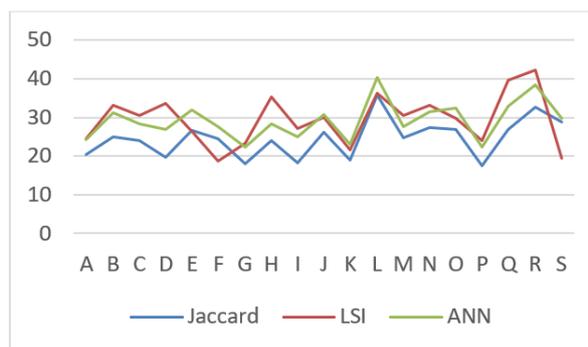
Percobaan pertama yang dilakukan adalah menggunakan data latihan konversi 4 nilai TF. Nilai TF dalam proses *preprocessing* yang dihasilkan tentunya

tidak hanya bernilai 0 dan 1, bisa saja nilai yang muncul 2, 3, dan 4, dst. Sehingga hal ini tentunya perlu dikonversi menjadi rentang nilai yang bisa dikerjakan oleh operator AND pada *hidden layer* ANN pada penelitian ini. Oleh karena itu, percobaan pertama ini, menggunakan data latihan yang mengambil nilai TF yang bernilai mulai dari 0 sampai 4. Konversi 4 Nilai TF ditampilkan seperti pada Tabel 6.

Tabel 6 Konversi 4 Nilai TF terhadap *Input* Neural Network

TF	Nilai <i>Input</i> Neural Network
0	-1
1	-0.5
2	0
3	0.5
4	1

Setelah proses pelatihan selesai dan menghasilkan model pelatihan, proses selanjutnya adalah pengujian dengan dua vektor kalimat “*query*” dan “*input*”. Kedua vektor ini mengandung nilai frekuensi. Proses pencocokan dilakukan dengan fungsi aktivasi Tanh yang dihitung dengan bobot yang disesuaikan pada saat proses *backpropagation*. Kemudian, nilai ini diteruskan ke lapisan *output* untuk menghasilkan persentase kesamaan antara dua teks yang diuji.



Gambar 10. Grafik perbandingan hasil persentase tingkat *similarity* jawaban *essay* siswa bernama “Septiana Eka Arwanda” dengan siswa lainnya dalam percobaan menggunakan data latihan konversi 4 nilai TF

Gambar 10 merupakan tabulasi tingkat persentase kesamaan jawaban dari seorang siswa bernama “Septiana Eka Arwanda Riski” dengan para siswa lainnya dalam satu mata kuliah yang diukur dari tiga metode yaitu metode ANN, LSI, dan Jaccard.

Hasil persentase tingkat kesamaan dari metode ANN terlihat mendekati hasil tingkat kesamaan metode Jaccard jika dibandingkan dengan metode LSI. Hal ini juga dibuktikan dengan nilai MAPE (Persamaan 5), dihasilkan Nilai MAPE untuk metode ANN lebih kecil dibandingkan dengan nilai MAPE yang dihasilkan metode LSI terhadap Metode Jaccard. Nilai MAPE untuk percobaan ini dapat dilihat pada Tabel 3.

3.3.2 Konversi 8 Nilai TF

Percobaan lainnya dilakukan kembali untuk mengukur efektifitas data latihan terhadap kemampuan ANN dalam

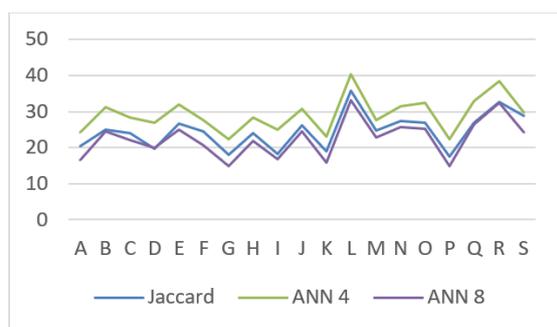
melakukan tugas perhitungan persentase kesamaan text. Percobaan ini menggunakan data latih tambahan yaitu konversi 8 nilai TF.

Basis pengetahuan ANN menggunakan 4 nilai TF masih terbilang kurang bisa mencakup keseluruhan kemungkinan nilai TF yang ada, karena banyak data yang memiliki nilai TF lebih dari 4, sehingga ANN kurang akurat dalam memberi keputusan jika ditemukan nilai TF lebih dari 4. Oleh karena itu pada percobaan ini ditentukan data latih dengan dengan konversi 8 Nilai TF seperti yang ditampilkan pada Tabel 7.

Tabel 7 Konversi 8 Nilai TF terhadap *Input Neural Network*

TF	Nilai <i>Input Neural Network</i>
0	-1
1	-0.75
2	-0.5
3	-0.25
4	0
5	0.25
6	0.5
7	0.75
8	1

Percobaan ini dilakukan dengan jawaban siswa yang sama seperti pada percobaan sebelumnya yang dibandingkan dengan siswa lainnya di dalam satu mata kuliah. Hasil percobaan menunjukkan nilai MAPE untuk metode ANN meningkat 11.1% menjadi 8.9% setelah menggunakan data latih dengan konversi 8 Nilai TF. Hasil perbandingan persentase kesamaan jawaban antar siswa untuk ANN yang menggunakan data latih konversi 4 Nilai TF (ANN4) dengan ANN yang menggunakan konversi 8 Nilai TF (ANN8) untuk percobaan ini ditampilkan pada grafik Gambar 11.



Gambar 11. Grafik perbandingan hasil persentase tingkat *similarity* jawaban *essay* siswa bernama “Septiana Eka Arwanda” dengan siswa lainnya dalam percobaan menggunakan data latih konversi 8 nilai TF.

Secara keseluruhan dari hasil-hasil dua percobaan utama yang dilakukan dapat dianalisa bahwa dengan menambahkan data latih yang basis pengetahuan nilai konversi TF nya lebih lebar untuk metode ANN menghasilkan persentase MAPE yang lebih kecil, yang menandakan kemampuan ANN semakin mendekati hasil persentase dari metode Jaccard dalam melakukan tugas pendeteksian *similarity* text. Hasil ini juga

menunjukkan bahwa metode ANN dapat diandalkan pada tugas pendeteksian kesamaan dua teks yang kinerjanya mendekati metode Jaccard dibandingkan dengan metode LSI.

Sebagai catatan, hasil tingkat kesamaan teks pada metode ANN, tidak akan menghasilkan nilai tingkat kesamaan teks yang tetap untuk objek teks yang sama untuk setiap sesi perhitungan yang berbeda. Hal ini dipengaruhi oleh nilai inialisasi parameter yang diberikan dalam Tabel 4, *activation function*, dan proses perbaikan untuk bobot yang terbentuk pada proses *backpropagation*. Sedangkan untuk metode LSI dan Jaccard memberikan nilai tetap untuk pengukuran berulang dalam satu objek karena proses pehitungannya menggunakan persamaan matematika yang sudah ditetapkan.

3.4 Hasil Percobaan Tambahan

3.4.1 Analisis Pengujian Fungsi Aktivasi

Percobaan tambahan dilakukan dengan tujuan menguji efektifitas *activation function* pada ANN dalam memberikan keputusan akhir. Selain aktivasi Tanh yang digunakan pada percobaan utama sebelumnya, pada percobaan ini juga menggunakan fungsi aktivasi lainnya yaitu ReLU dan Sigmoid. Masing-masing fungsi aktivasi diujikan agar bisa melihat dan memperbandingkan hasil tingkat persentase kesamaan text yang dihasilkan, sehingga bisa dianalisa fungsi aktivasi yang mana lebih baik kinerjanya untuk metode ANN pada penelitian ini.

Pengujian aktivasi ini dilakukan dengan data latih konversi 8 nilai TF dengan pengaturan inialisasi nilai parameter sama seperti pada Tabel 4.

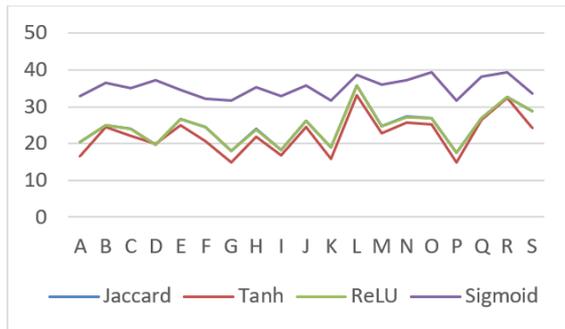
Tabel 8 Nilai MAPE Pengujian Aktivasi

Aktivasi	MAPE
Tanh	8.9%
ReLU	0.2%
Sigmoid	47.8%

Pengujian fungsi aktivasi ini juga menggunakan objek siswa dan sesi ujian yang sama seperti pada percobaan utama, dengan tugas utama yaitu mengukur tingkat kesamaan jawaban *essay* siswa bernama “Septiana Eka Arwanda Riski” terhadap siswa lainnya. Tabel 8 menampilkan nilai MAPE pada pengujian fungsi aktivasi ANN. Aktivasi Tanh memperoleh nilai MAPE sebesar 8.9%, sedangkan ReLU 0.1% dan Sigmoid 47.8%.

Gambar 12 menampilkan perbandingan detail persentase kesamaan jawaban antar siswa pada metode ANN dengan fungsi aktivasi yang berbeda. Terlihat aktivasi ReLU lebih berhimpitan dengan metode Jaccard dibandingkan aktivasi Tanh yang masih terlihat menyimpangnya. Sedangkan aktivasi Sigmoid memiliki penyimpangan terjauh dengan metode Jaccard

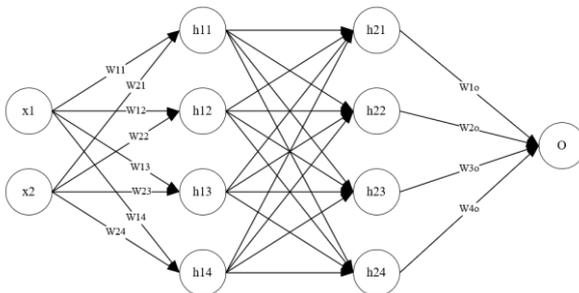
yang sebagai nilai faktual. Hal ini menggambarkan bahwa fungsi ReLU lebih baik dari fungsi aktivasi lainnya sesuai dengan nilai MAPE yang dihasilkan paling kecil.



Gambar 12. Grafik hasil perbandingan fungsi aktivasi ANN (Tanh, ReLU, dan Sigmoid terhadap nilai acuan persentase tingkat kesamaan text pada metode Jaccard

3.4.2 Analisis Pengujian Model Neural Network

Percobaan tambahan dilakukan dengan menguji model ANN lainnya dengan menambahkan 1 hidden layer dari model sebelumnya. Penambahan *hidden layer* sehingga menjadi lebih dari satu *hidden layer* seperti pada Gambar 13 sering disebut dengan arsitektur MLP (*Multi-Layer Perceptron*) [31] [32].



Gambar 13. Arsitektur neural network dengan tambahan 1 hidden layer

Gambar 13 menampilkan model MLP dengan 2 *hidden layer* dimana tiap *hidden layer* berisi 4 *node*. Pengujian dilakukan dengan data dan pengaturan yang sama seperti yang dijelaskan pada sub bagian 3.2. Dari pengujian yang dilakukan didapatkan hasil pengujian seperti terlihat pada Tabel 9.

Tabel 9 Nilai MAPE Pengujian Model

Model	Waktu Pelatihan	MAPE
1 Hidden Layer	2.32 detik	0.26%
2 Hidden Layer	26.97 detik	0.07%

Tabel 9 merupakan hasil pengujian kinerja ANN dengan dengan model MLP. Terlihat nilai MAPE yang didapat dari model MLP lebih kecil dari metode ANN dengan 1 *hidden layer*. Hal ini menandakan metode ANN dengan menggunakan model MLP memiliki dampak positif

terhadap efektivitas deteksi tingkat *similarity* karena Hasil tingkat *similarity*nya mendekati hasil Jaccard *similarity* dengan perolehan nilai MAPE 0,07%. Namun, masa waktu pelatihan dengan menggunakan model MLP jauh lebih lama dibandingkan ANN yang menggunakan 1 *hidden layer*. Hal ini disebabkan semakin rumit node jaringan *Neural Network*, semakin banyak komputasi penyesuaian bobot yang dilakukan sehingga kebutuhan waktu pemrosesan semakin meningkat.

Secara keseluruhan dari percobaan tambahan yang dilakukan ini, menandakan bahwa ketepatan pemilihan fungsi aktivasi, serta penambahan *hidden layer* pada arsitektur metode ANN memiliki kontribusi penting dalam peningkatan efektivitas kinerja metode ANN untuk menghasilkan persentase tingkat kesamaan sebuah *text*. Hal ini dibuktikan dengan persentase tingkat *similarity* lebih mendekati hasil dari metode acuan yaitu Jaccard *Similarity*.

4 Kesimpulan

Sistem Pendeteksi Kesamaan Jawaban *Essay* Siswa yang dibangun dalam penelitian ini berhasil mengadopsi konsep fungsional *e-learning*, seperti pengelolaan data siswa, pengelolaan data Pengajar, pengelolaan data mata pelajaran, dan pengelolaan tugas siswa.

Penelitian ini berhasil mengimplementasikan fitur mendeteksi kesamaan text menggunakan metode ANN dengan arsitektur 3-layer: *input*, *hidden*, dan *output layer*. Operator yang digunakan adalah AND dengan fungsi aktivasi pada *hidden layer*. Dalam mengevaluasi kinerja metode ANN, digunakan metode MAPE, dengan menetapkan metode Jaccard sebagai nilai aktualnya. Hasil persentase MAPE pada metode ANN yang paling baik adalah 0.07% yang didapat dengan menggunakan fungsi aktivasi ReLU dan dengan model arsitektur MLP. Sedangkan nilai MAPE metode LSI adalah 28,0%. Hasil ini menunjukkan bahwa kinerja metode ANN lebih efektif dibandingkan dengan LSI dalam tugas pendeteksian kesamaan teks karena memiliki kinerja yang mendekati metode Jaccard yang dikenal sebagai metode *similarity* yang andal dalam fitur sintaksis kata. Pekerjaan selanjutnya yang bisa dilakukan adalah mengembangkan metode ANN untuk bisa mendeteksi kesamaan text baik itu secara *lexical* maupun *semantic*.

Ucapan Terimakasih

Terima kasih kepada Universitas Udayana yang telah memberikan dukungan berupa hibah untuk penelitian ini dan pihak-pihak yang telah membantu menyelesaikan penelitian ini..

Daftar Rujukan

- [1] G. Strimel and M. E. Grubbs, "Positioning technology and engineering education as a key force in STEM education," *J.*

- Technol. Educ.*, 2016, doi: 10.21061/jte.v27i2.a.2.
- [2] W. He, Z. (Justin) Zhang, and W. Li, "Information technology solutions, challenges, and suggestions for tackling the COVID-19 pandemic," *Int. J. Inf. Manage.*, vol. 57, 2021, doi: 10.1016/j.ijinfomgt.2020.102287.
- [3] G. Sakarkar, "Intelligent Online e-Learning Systems: A Comparative Study," vol. 56, no. 4, pp. 21–25, 2012.
- [4] A. Shah and S. Jain, "An Agent based Personalized Intelligent elearning," *Int. J. Comput. Appl.*, 2011, doi: 10.5120/2411-3215.
- [5] E. L. Amalia, A. J. Jumadi, I. A. Mashudi, and D. W. Wibowo, "Analisis Metode Cosine Similarity Pada Aplikasi Ujian Online Otomatis (Studi Kasus JTI POLINEMA)," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 8, no. 2, p. 343, 2021, doi: 10.25126/jtiik.2021824356.
- [6] N. M. A. Lestari and M. Sudarma, "Perencanaan Search Engine E-commerce dengan Metode Latent Semantic Indexing Berbasis Multiplatform," *Lontar Komput. J. Ilm. Teknol. Inf.*, vol. 8, no. 1, p. 31, 2017, doi: 10.24843/lkjiti.2017.v08.i01.p04.
- [7] K. Selvi and R. M. Suresh, "Context similarity measure using fuzzy formal concept analysis," in *ACM International Conference Proceeding Series*, 2012, pp. 420–426, doi: 10.1145/2393216.2393287.
- [8] P. Neculoiu, M. Versteegh, and M. Rotaru, "Learning Text Similarity with Siamese Recurrent Networks," Aug. 2016, pp. 148–157, doi: 10.18653/v1/w16-1617.
- [9] Y. L. Joane, A. Sinsuw, and A. Jacobus, "Rancang Bangun Aplikasi Deteksi Kemiripan Dokumen Teks Menggunakan Algoritma Ratcliff/Obershelp," *J. Tek. Inform.*, vol. 11, no. 1, 2017, doi: 10.35793/jti.11.1.2017.17654.
- [10] N. Alamsyah, "Deteksi Plagiarisme Tingkat Kemiripan Judul Skripsi Dengan Algoritma Winnowing," *Technol. J. Ilm.*, vol. 8, no. 4, p. 205, 2017, doi: 10.31602/tji.v8i4.1119.
- [11] N. E. Diana and I. Hanana Ulfa, "Measuring performance of n-gram and jaccard-similarity metrics in document plagiarism application," in *Journal of Physics: Conference Series*, 2019, vol. 1196, no. 1, doi: 10.1088/1742-6596/1196/1/012069.
- [12] D. Thorleucher, D. Van Den Poel, and A. Prinzie, "Mining ideas from textual information," *Expert Syst. Appl.*, vol. 37, no. 10, 2010, doi: 10.1016/j.eswa.2010.04.013.
- [13] E. Iftikhar, A. Iftikhar, and M. K. Mehmood, "Identification of textual entailments in business rules," *2016 6th Int. Conf. Innov. Comput. Technol. INTECH 2016*, pp. 706–711, 2017, doi: 10.1109/INTECH.2016.7845104.
- [14] A. W. Pradana and M. Hayaty, "The Effect of Stemming and Removal of Stopwords on the Accuracy of Sentiment Analysis on Indonesian-language Texts," *Kinet. Game Technol. Inf. Syst. Comput. Network, Comput. Electron. Control*, 2019, doi: 10.22219/kinetik.v4i4.912.
- [15] J. Wibowo, "Aplikasi Penentuan Kata Dasar Dari Kata Berimbuhan Pada Kalimat Bahasa Indonesia Dengan Algoritma Stemming," *J. Ris. Komput.*, vol. 3, no. 5, 2016.
- [16] M. A. Rosid, A. S. Fitriani, I. R. I. Astutik, N. I. Mulloh, and H. A. Gozali, "Improving Text Preprocessing for Student Complaint Document Classification Using Sastrawi," in *IOP Conference Series: Materials Science and Engineering*, 2020, vol. 874, no. 1, doi: 10.1088/1757-899X/874/1/012017.
- [17] S. Fahmi, L. Purnamawati, G. F. Shidik, M. Muljono, and A. Z. Fanani, "Sentiment analysis of student review in learning management system based on sastrawi stemmer and SVM-PSO," 2020, doi: 10.1109/iSemantic50169.2020.9234291.
- [18] I. M. Suwija Putra, Y. Adiwinata, D. P. Singgih Putri, and N. P. Sutramiani, "Extractive Text Summarization of Student Essay Assignment Using Sentence Weight Features and Fuzzy C-Means," *Int. J. Artif. Intell. Res.*, vol. 5, no. 1, pp. 13–24, 2021, doi: 10.29099/ijair.v5i1.187.
- [19] S. Harvey and R. Harvey, "An introduction to artificial intelligence," *Appita J.*, vol. 51, no. 1, 1998, doi: 10.2514/6.1994-294.
- [20] M. A. Hanafiah, N. L. W. S. R. Ginantra, and A. D. GS, "Analysis of ANN Backpropagation Ability to Predict Expenditure Group Inflation," *IJISTECH (International J. Inf. Syst. Technol.)*, vol. 4, no. 2, 2020, doi: 10.30645/ijistech.v4i2.103.
- [21] D. Marlina and F. Arifin, "Predicting The Number of Tourists Based on Backpropagation Algorithm," *RESTI J. (System Eng. Inf. Technol.)*, vol. 5, no. 3, pp. 439 – 445, 2021, doi: <https://doi.org/10.29207/resti.v5i3.3061>.
- [22] T. Hofmann, "Probabilistic Latent Semantic Indexing," *ACM SIGIR Forum*, vol. 51, no. 2, 2017, doi: 10.1145/3130348.3130370.
- [23] R. Anita, C. N. Subalalitha, A. Dorle, and K. Venkatesh, "Semantic search using Latent Semantic Indexing and Word Net," *ARPJ J. Eng. Appl. Sci.*, vol. 12, no. 2, 2017.
- [24] M. S. Hasibuan, L. E. Nugroho, and P. I. Santosa, "Model detecting learning styles with artificial neural network," *J. Technol. Sci. Educ.*, vol. 9, no. 1, 2019, doi: 10.3926/jotse.540.
- [25] M. Halkidi, Y. Batistakis, and M. Vazirgiannis, "Cluster validity methods: Part I," *SIGMOD Record*, vol. 31, no. 2, 2002, doi: 10.1145/565117.565124.
- [26] S. Niwattanakul, J. Singthongchai, E. Naenudorn, and S. Wanapu, "Using of jaccard coefficient for keywords similarity," in *Lecture Notes in Engineering and Computer Science*, 2013, vol. 2202.
- [27] H. T. Huynh, N. Duong-Trung, D. Q. Truong, and H. X. Huynh, "Vietnamese text classification with textrank and jaccard similarity coefficient," *Adv. Sci. Technol. Eng. Syst.*, vol. 5, no. 6, 2020, doi: 10.25046/aj050644.
- [28] A. Gielczyk, M. Choras, and R. Kozik, "Lightweight verification schema for image-based palmprint biometric systems," *Mob. Inf. Syst.*, vol. 2019, 2019, doi: 10.1155/2019/2325891.
- [29] S. Bag, S. K. Kumar, and M. K. Tiwari, "An efficient recommendation generation using relevant Jaccard similarity," *Inf. Sci. (Ny)*, vol. 483, 2019, doi: 10.1016/j.ins.2019.01.023.
- [30] T. T. N. Le and T. V. X. Phuong, "Privacy Preserving Jaccard Similarity by Cloud-Assisted for Classification," *Wirel. Pers. Commun.*, vol. 112, no. 3, 2020, doi: 10.1007/s11277-020-07131-6.
- [31] A. Hashemi Fath, F. Madanifar, and M. Abbasi, "Implementation of multilayer perceptron (MLP) and radial basis function (RBF) neural networks to predict solution gas-oil ratio of crude oil systems," *Petroleum*, vol. 6, no. 1, 2020, doi: 10.1016/j.petlm.2018.12.002.
- [32] C. K. Dewa, A. L. Fadhilah, and A. Afiahayati, "Convolutional Neural Networks for Handwritten Javanese Character Recognition," *IJCCS (Indonesian J. Comput. Cybern. Syst.)*, 2018, doi: 10.22146/ijccs.31144.