SKANIKA: Sistem Komputer dan Teknik Informatika
Volume 6, Nomor 1, Januari 2023, Halaman 42-51 E-ISSN: 2721-4788

Penerapan Algoritma *Multiclass Support Vector Machine* dan TF-IDF Untuk Klasifikasi Topik Tugas Akhir

Alif Dewan Daru Wibiyanto[1], Arief Wibowo[2*]
[1,2]Fakultas Teknologi Informasi, Teknik Informatika, Universitas Budi Luhur, Jakarta, Indonesia
Email: [1]alifdewandaru262@gmail.com, [2*]arief.wibowo@budiluhur.ac.id
(* : corresponding author)

Abstrak

Dalam penyelesaian jenjang strata atau diploma, mahasiswa harus melewati tahap akhir yaitu penyelesaian tugas akhir. Dengan banyaknya mahasiswa yang mengajukan tugas akhir, proses pengarsipan dokumen tugas akhir berdasarkan topik membutuhkan klasifikasi yang dilakukan oleh Perpustakaan Universitas. Klasifikasi topik tugas akhir secara manual dapat menghambat proses lain yang harus diselesaikan oleh staff perpustakaan. Oleh karena itu, diperlukan adanya klasifikasi dokumen tugas akhir secara otomatis, cepat dan akurat untuk klasifikasi topik tugas akhir. Penelitian ini memanfaatkan pembobotan TF-IDF serta pendekatan algoritma *Multiclass Support Vector Machine*. Pembobotan TF-IDF ini merupakan formula yang merepresentasikan signifikansi suatu kata (*term*) dalam sebuah dokumen dan korpus. Pemobotan yang sudah dihitung dengan TF-IDF, kemudian menghitung hasil prediksi bobot menggunakan *Support Vector Machine*. Dikarenakan *Support Vector Machine* hanya bisa menentukan 2 kelas, diperlukannya metode *one against rest* dalam algoritma *support vector machine* untuk menentukan lebih dari 2 kelas. Penelitian ini bertujuan untuk membuat sistem yang dapat mengelompokkan topik tugas akhir secara otomatis, cepat dan akurat untuk pengarsipan tugas akhir di Perpustakaan. Hasil dari perhitungan pengujian dengan data latih dan data uji menggunakan pembobotan TF-IDF dan algoritma *multiclass support vector machine* menghasilkan persentase sebesar 90.71%.

Kata Kunci: *multiclass support vector machine,* TF-IDF, topik tugas akhir

Abstract

Students must pass the final stage, which is the completion of the final project, in order to complete the strata or diploma level. Because there are so many students submitting final projects, the process of archiving final project documents by topic necessitates classification by the University Library. Manually classifying final project topics can impede other processes that library staff must complete. As a result, a final assignment document classification that is automatic, quick, and accurate is required for the classification of the final project topic. This study employs TF-IDF weighting and the Multiclass Support Vector Machine algorithm. The TF-IDF weighting formula represents the importance of a word (term) in a document and a corpus.The weights calculated with TF-IDF are then used to predict weight results with the Support Vector Machine. Because the Support Vector Machine algorithm can only define two classes, a one against rest method is required to determine more than two classes. The goal of this research is to develop a system that can automatically, quickly, and accurately categorise final project topics for archiving final assignments in the library. The percentage produced by the test calculations with training and test data using TF-IDF weighting and multiclass support vector machine algorithms is 90.71%.

Keywords*: multiclass support vector machine, TF-IDF, final project topic*

1. P ENDAHULUAN

Perkembangan teknologi informasi dan komputer di Indonesia semakin meningkat seiring dengan perkembangan revolusi industri di Indonesia. Perkembangan teknologi informasi dan komputer di Indonesia telah memberikan banyak manfaat pada banyak bidang dalam kehidupan. Dijelaskan bahwa teknologi informasi dan komputer memiliki banyak manfaat dan manfaat bagi semua lapisan masyarakat [1].

Manfaat kemajuan teknologi dan komputer dalam penyusunan makalah penelitian. Kegiatan yang dilakukan secara sistematis untuk mengumpulkan, mengolah, dan menyimpulkan data dengan menggunakan metode/teknik tertentu dalam rangka mencari solusi dari permasalahan

yang dihadapi[2]. Teknik ini bertujuan untuk mengungkap berbagai teori yang relevan dengan problematika yang sedang dipertimbangkan sebagai bahan referensi dalam pembahasan temuan penelitian yang sedang dipertimbangkan.

Menggunakan *text mining* dalam kategorisasi otomatis adalah salah satu solusi yang bisa digunakan. *Text mining* menjadi salah satu teknik yang digunakan untuk mengklasifikasikan dokumen atau teks yaitu *Support Vector Machine* (SVM). Algoritma *Support Vector Machine* (SVM) adalah metode regresi atau pengklasifikasian data berlandaskan data sebelumnya, dimana model *supervised* terlebih dahulu [3].

Sistem yang mendeteksi nilai dokumen diperlukan untuk membuatnya berdasarkan topik tertentu. Dokumen judul tugas akhir program penelitian sistem informasi Universitas Budi Luhur tahun 2021 yang dipublikasikan di perpustakaan dan dijadikan sebagai data primer penelitian ini menjadi objek penelitian dalam penelitian ini.

Optimalisasi *Support Vector Machine* (SVM) terus ditingkatkan agar SVM dapat mengklasifikasikan data menjadi dua atau lebih kelas: model *one-against-one* dan *one-against-rest*. Kedua model tersebut dapat mendukung proses klasifikasi multikelas dalam penelitian ini dengan lebih dari dua kelas klasifikasi [4]. Pada tahap pembobotan kata (*term*) menggunakan metode *Term Frequency - Inverse Document Frequency* (TF-IDF) untuk menilai seberapa penting sebuah kata pada suatu dokumen.

Pada penelitian sebelumnya [5] melakukan klasifikasi topik tugas akhir dengan hasil akurasi sebesar 86,66% menggunakan metode *cosine similarity* dan pembobotan TF-IDF. Penelitian yang lain [6] melakukan klasifikasi topik tugas akhir menggunakan metode *naïve bayes* dengan pendekatan *design thinking* menununjukan akurasi sebesar 87,5% pada data. Penelitian selanjutnya [7] yang menggunakan pendekatan TF-IDF menghasilkan akurasi sebesar 80,83%.

Tujuan dari penelitian ini diharapkan sistem klasifikasi yang dikembangkan dengan menggunakan pendekatan *multiclass support vector machine* dan metode TF-IDF yang akan di implementasikan berbasis *web.* Penelitian ini diharapkan dapat membantu Perpustakaan Universitas Budi Luhur dalam upaya pengarsipan dokumen tugas akhir dengan mengkategorikan secara otomatis, tepat dan akurat.

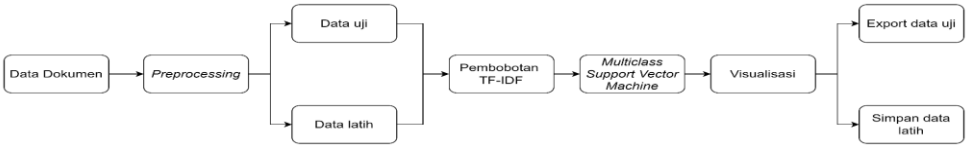
2. METODE PENELITIAN

2.1 Data Penelitian

Data latih yang digunakan dalam penelitian ini merupakan judul tugas akhir mahasiswa Universitas Budi Luhur, judul ini seperti yang terdapat pada dokumen skripsi mahasiswa program studi sistem informasi Fakultas Teknologi Informasi. Data yang didapat berjumlah 917 judul skripsi dan informasi ini didapat dari Perpustakaan Universitas Budi Luhur. Data tersebut kemudian diolah melalui tahap *preprocessing* sebelum digunakan sebagai data latih.

2.2 Penerapan Metode

Ada beberapa tahapan yang menjadi desain utama saat membangun sistem klasifikasi untuk kategorisasi topik ini menggunakan pendekatan pembobotan TF-IDF untuk metode *support vector machine one against rest*; perancangan ini merupakan gambaran proses dari awal hingga akhir sistem berjalan, seperti yang ada pada gambar berikut ini:



Gambar 1. Tahapan Metode

Pada Gambar 1 proses pengolahan dokumen judul skripsi mahasiswa sistem informasi Universitas Budi Luhur, pada tahapan ini dokumen judul skripsi digunakan sebagai data uji dan

data latih. Pada proses data dokumen, data yang di *import* akan diproses melalui tahapan *preprocessing*. Dokumen yang digunakan sebagai data latih dan uji kemudian disimpan ke sistem sementara, artinya ketika data baru diimpor, data sebelumnya secara otomatis digantikan oleh data baru. Dokumen yang digunakan adalah jenis file .xls dan .csv, dan isinya berupa teks. Pada tahap selanjutnya dokumen melawati tahapan *preprocessing*, akan dilakukan perhitungan dengan pembobotan TD-IDF, setelah hasil perhitungan pembobotan sudah didapat, proses selanjutnya adalah mencari garis *hyperplane* dari hasil pembobotan judul menggunakan *support vector machine*. *support vector machine* melakukan proses perhitungan untuk mencari topik menggunakan perhitungan *one against rest*. Semua proses perhitungan dapat dilihat oleh pengguna dari *preprocessing*, pembobotan TF-IDF, nilai *support vector machine* dan *one against rest,* sampai klasifikasi dokumen judul skripsi mahasiswa sistem informasi Universitas Budi Luhur masuk kedalam kategori topik yang dituju.

2.3 Pengumpulan Data

Pada tahap ini sedang berlangsung proses pengumpulan data dokumen judul tugas akhir mahasiswa sistem informasi Universitas Budi Luhur yang akan digunakan baik sebagai data latih maupun data uji. Data pelatihan berasal dari dokumen judul skripsi mahasiswa yang diterbitkan di Perpustakaan Universitas Budi Luhur. Data uji berasal dari dokumen judul tugas akhir mahasiswa program studi sistem informasi Universitas Budi Luhur yang akan menjalankan skripsi maupun sudah sehingga pengguna bisa melakukan pengecekan mengenai kategori topik tugas akhir tersebut.

2.4 *Preprocessing*

Tahapan pertama dalam *text mining* adalah *preprocessing* data, yang merupakan tahapan menyiapkan data menjadi lebih terstruktur dan ideal untuk diproses. Dokumen biasanya mempunyai format yang tidak beraturan atau tidak terstruktur [8]. Diperlukan proses yang dapat melakukan penstrukturan ulang data yang sebelumnya tidak beraturan menjadi data terstruktur. Tahapan-tahapan *preprocessing* antara lain, *case folding, stemming, stop word removal.*

2.5 Pembagian Data

Dataset akan dibagi pada tahap proses berbagi data untuk mendapatkan data uji serta data latih. Dalam penelitian ini, akurasi pelatihan dibandingkan dengan akurasi pengujian menggunakan persentase data latih 80% dan data uji 20%. Dalam penelitian ini, cara pembagian *dataset* menggunakan metode *hold out*. Metode ini membagi seluruh dataset (data) menjadi beberapa bagian [5].

2.6 Pembobotan TF-IDF

Metode TF-IDF, ketika digunakan dalam proses pembobotan, dapat menghasilkan vektor dengan banyak istilah, memungkinkan setiap kata dihitung sebagai fitur untuk dikenali. [9]. Pada proses pembobotan TF-IDF adalah kalkulasi yang menggambarkan seberapa penting kata (*term*) dalam suatu *corpus* serta dokumen. Proses mengkalkulasi bobot dari setiap dokumen terhadap *term* dengan formula sebagai berikut:

$$Wdt = tfdt \times Idft \quad (1)$$

Wdt : bobot dokumen ke-d terhadap kata ke-t
$tfdt$: banyak kata yang dicari pada sebuah dokumen
$idft$: *inversed document frequency* $(log\ log\ (\frac{n}{df}))$
N : total dokumen
df : banyak dokumen yang mengandung kata yang dicari

2.7 *Multiclass Support Vector Machine*

Support vector machine merupakan algoritme dengan tingkat akurasi yang lebih tinggi, yang umumnya digunakan untuk klasifikasi data adalah dalam bentuk teks[10]. Algoritma pemodelan *Support Vector Machine* adalah algoritma klasifikasi yang bekerja dengan mencari titik (*hyperplanes*) untuk mengklasifikasikan dua set data. *Hyperplane* klasifikasi linear *Support Vector Machine* dinotasikan sebagai berikut:

$$f(x) = w, x + b = 0 \tag{2}$$

Dari persamaan di atas di dapatkan pertidaksamaan kelas +1 dan pertidaksamaan kelas -1:

$$\begin{gathered} w.x + b \leq +1 \\ w.x + b \geq -1 \end{gathered} \tag{3}$$

w : bobot
b : bias
x : jumlah data

Setelah perhitungan linear telah didapat, *support vector machine* menggunakan pendekatan *gradient decent* untuk meminimalkan fungsi *cost* yang diaplikasikan sebagai berikut:

$$\begin{gathered} w_2 = w_1 - L * \frac{dJ}{dw} \\ b_2 = b_1 - L * \frac{dJ}{db} \end{gathered} \tag{5}$$

w : bobot
b : bias
L : kecepatan *learning* (*hyperparameter*)
$\frac{dJ}{dw}$: turunan parsial dari fungsi terhadap bobot
$\frac{dJ}{db}$: turunan parsial dari fungsi terhadap bias

Pengaplikasian *multiclass* pada penelitian ini menggunakan pendekatan *one against rest* dengan perhitungan sebagai berikut:

$$\begin{gathered} (w^m)^T \phi(x_m) + b^m \geq 1 - \xi_i^m \rightarrow y_i = m \\ (w^m)^T \phi(x_m) + b^m \leq 1 - \xi_i^m \rightarrow y_i \neq m \\ \xi_i^m \geq 0, \qquad i = 1, \dots, l \end{gathered} \tag{6}$$

2.8 Perhitungan Akurasi

Nilai presisi program tidak konstan karena berbagi data dengan *Hold Out* menyebabkan data diuji secara acak setiap kali dijalankan. Maka pengujian akurasi dihitung sebagai berikut[5]:

$$akurasi = \frac{jumlah\ terklasifikasi\ benar}{total\ data} \times 100\% \tag{7}$$

3. HASIL DAN PEMBAHASAN

Berdasarkan hasil pembagian *dataset* yang telah dilakukan proses *preprocessing* sebanyak 917 data. Kemudian akan dilakukan proses pembagian data terhadap *dataset* menjadi data *train* dan data *test* dengan perbandingan 80% data *train* dan 20% data *test*, maka menghasilkan 734 data *train* dan 183 data *test*.

3.1 Pengujian Metode

Pengujian merupakan salah satu proses penting dalam pembuatan sistem, guna untuk mengevaluasi and menganalisis seberapa besar tingkat akurasi sistem yang telah dibuat. Pada penelitian ini menggunakan pengujian pada akurasi. Data yang digunakan untuk pengujian menyumbang 20% dari keseluruhan dataset (917 data). Pada tabel 1 merupakan hasil pengujian data dari sistem yang telah dibuat:

Tabel 1. Data Hasil Pengujian

No.	Topik TA	Klasifikasi Sebenarnya	Klasifikasi Model
1	Aplikasi Sistem Penunjang Keputusan Pemilihan Guru Terbaik Metode Weighted Product Wp Smk Bina Insan Mandiri Berbasis Web	Sistem Penunjang Keputusan	Sistem Penunjang Keputusan
2	Perancangan Model E Commerce Penjualan Barang Optik Menggunakan Content Management System CMS	*E - Commerce*	*E - Commerce*
3	Analisa Desain Sistem Informasi Penjualan Menggunakan Metodologi Berorientasi Obyek PT Zona Inspirasi Indonesia	Perancangan Sistem	Perancangan Sistem
4	Analisa Desain Sistem Penjualan Biji Kopi Kopi Manu Berbasis Web Menggunakan Fitur *Content Management System CMS*	Perancangan Sistem	Perancangan Sistem
5	Analisa Implementasi E Commerce Menggunakan Content Management System CMS Meningkatkan Penjualan Memperluas Kawasan Pemasaran Toko Indah Jaya Sport	*E - Commerce*	*E - Commerce*
6	Analisa Implementasi E Commerce Menggunakan Content Management System CMS Wordpress Toko Cozette Candle	*E - Commerce*	*E - Commerce*
…	…	…	…
183	Startup marketplace rajin tangan	*Startup / Marketplace*	*Startup / Marketplace*

Dari hasil pengujian diatas dapat dihitung dengan rumus berikut:

$$Akurasi = \frac{Jumlah\ Terklasifikasi\ Benar}{Total\ Data} \times 100\%$$
$$Akurasi = \frac{166}{183} \times 100\% \quad (8)$$
$$Akurasi = 90{,}71\%$$

Pengujian data menggunakan pembobotan TF-IDF dan algoritma *multiclass support vector machine* pada data judul skripsi mahasiswa sistem informasi Universitas Budi Luhur menunjukan bahwa data *test* sebesar 183 data menghasilkan akurasi sebesar 90,71% pada penelitian ini.

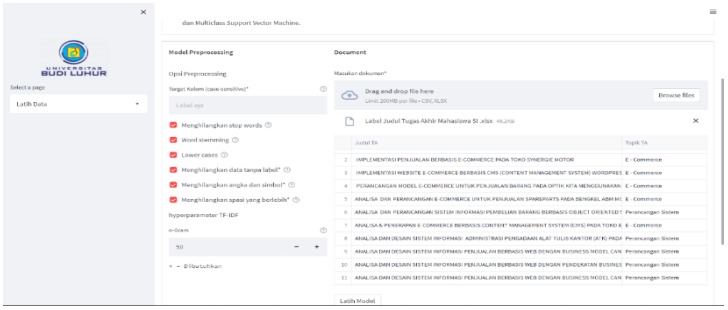
3.2 *Dashboard* Data Latih

Tampilan *dashboard* pada Gambar 2 data latih adalah tampilan pertama kali yang dijalankan oleh sistem. Di menu utama pengguna harus memasukkan data latih berupa *.xlsx* atau *.csv* untuk memulai proses melatih model.

Gambar 2. *Dashboard* Data Latih

3.2.1 *Upload* data latih

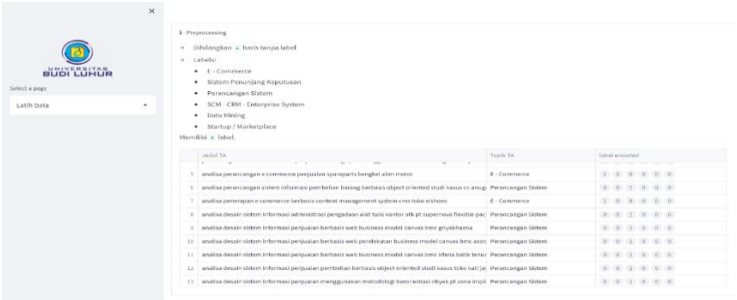
Upload data latih berupa *.xlsx* dan *.csv* di dalam menu latih data diperlukan untuk memasukan data untuk dilatih dan diproses oleh model. Merujuk pada Gambar 3, pada saat *button browse file* di klik, pengguna akan memasukan data yang ingin dijadikan data latih.



Gambar 3. *Upload* Data Latih

3.2.2 *Preprocessing* data latih

Proses fitur *Preprocessing* dapat dilihat dari tabel hasil akhir dari keluaran yang telah diproses. Tabel ini berisi judul skripsi yang sudah diproses, dan labeling berupa angka biner agar dengan mudah dapat diproses oleh model seperti pada Gambar 4.

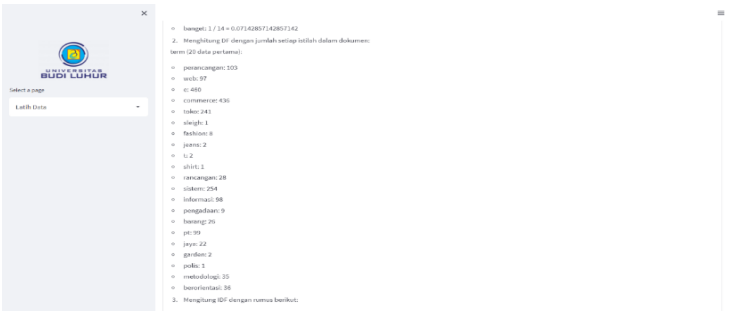


Gambar 4. Proses *Preprocessing*

3.2.3 Pembobotan TF-IDF data latih

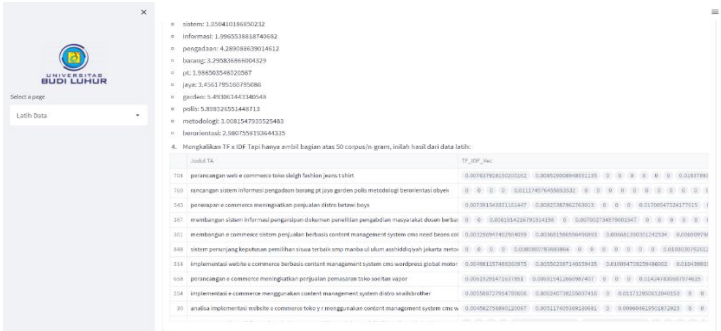
Fitur ini memperlihatkan bagaimana perhitungan pembobotan TF-IDF berlangsung. Pertama akan menggunakan rumus perhitungan TF pada Gambar 5, lalu *term* akan dihitung menggunakan perhitungan DF yang merujuk pada Gambar 6. *Term* yang sudah dihitung dengan DF, akan dilakukannya kalkulasi IDF pada Gambar 7. Kalkulasi terakhir yang dilakukan adalah mengkalikan nilai TF dan IDF, dan divisualisasikan pada Gambar 8.

Gambar 5. Proses Perhitungan TF



Gambar 6. Hasil Perhitungan *DF*

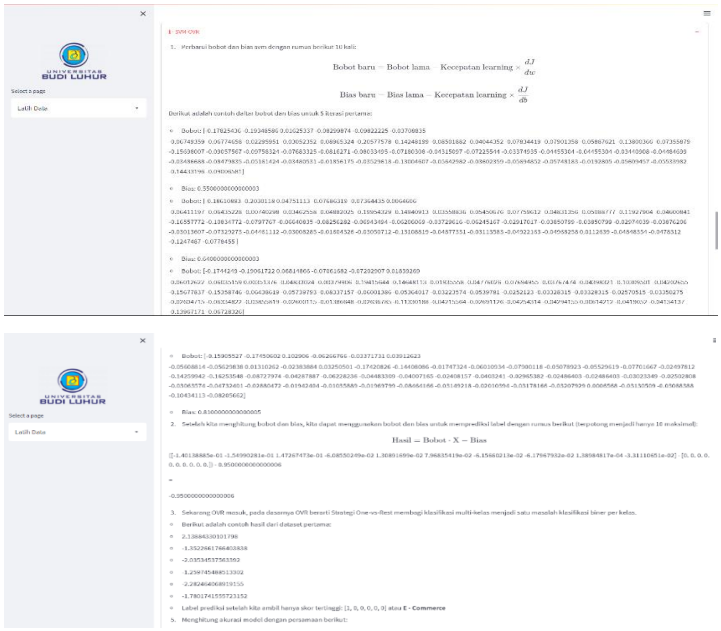
Gambar 7. Proses Perhitungan IDF



Gambar 8. Proses TF-IDF

3.2.4 *Multiclass Support Vector Machine* data latih

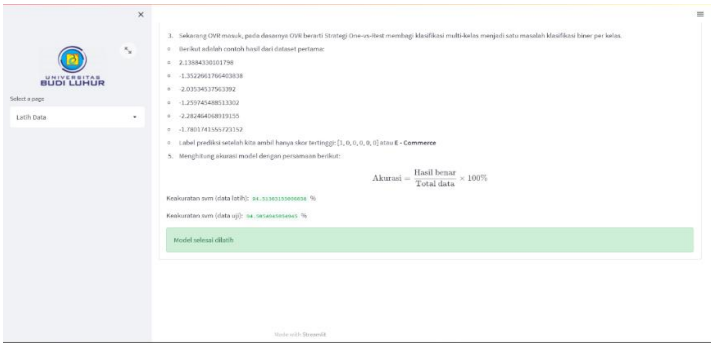
Pembobotan TF-IDF selesai dan dilanjutkan dengan perhitungan bobot dan bias untuk *support vector machine*. Digunakan bobot dan bias yang baru dengan cara menghitung menggunakan rumus yang tersedia pada Gambar 9. bobot dan bias tersebut untuk memprediksi label. Kemudian, dilanjutkan dengan prediksi label menggunakan *one against rest*.



Gambar 9. Proses *Multiclass Support Vector Machine*

3.2.5 Perhitungan Akurasi

Menggunakan rumus perhitungan akurasi, seperti pada Gambar 10. Pada saat ini juga model selesai dilatih dan akan digunakan sebagai model prediksi.



Gambar 10. Proses kalkulasi akurasi

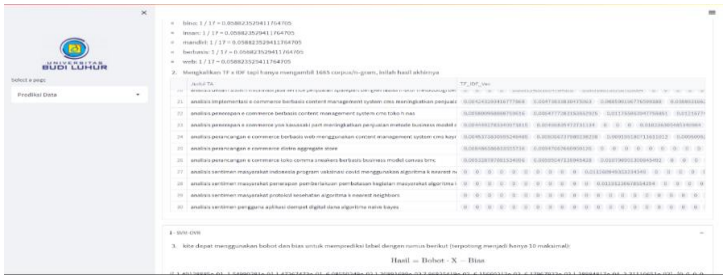
3.3 *Dashboard* Prediksi Data

Menu ini akan menampilkan proses prediksi terhadap data uji. Gambar 11 adalah gambaran tampilan menu yang akan tampil pada saat menu Prediksi Data pertama kali dibuka.

Gambar 11. *Dashboard* Prediksi Data

3.3.1 Pembobotan TF-IDF data uji

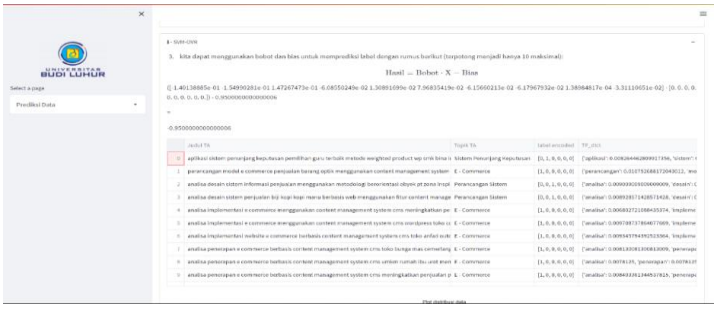
selanjutnya dilakukan proses perhitungan perkalian TF x IDF untuk mendapatkan hasil akhir pembobotan seperti pada Gambar 12:



Gambar 12. Pembobotan TF-IDF

3.3.2 *Multiclass Support Vector Machine* data uji

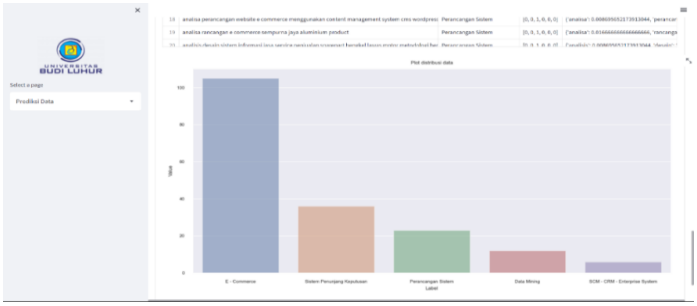
Perhitungan ini hanya melakukan perhitungan terakhir karena tidak diperlukan lagi menghitung bobot dan bias. Tampilan pada proses ini terdapat pada Gambar 13:



Gambar 13. Proses *Multiclass Support Vector Machine*

3.3.3 Visualisasi data uji

Berikut Gambar 14 berupa grafik yang menunjukan label data uji yang terdeteksi oleh Prediksi Data:



Gambar 14. Visualisasi data uji

4. KESIMPULAN

Berdasarkan hasil evaluasi dari klasifikasi topik tugas akhir menggunakan algoritma *multiclass support vector machine* dan metode TF-IDF berbasis web', maka dapat disimpulkan bahwa, berdasarkan judul skripsi, sistem ini dapat membaca topik suatu dokumen dengan sumber data dokumen skripsi mahasiswa sistem informasi Universitas Budi Luhur. Dalam proses pengujian terhadap 20% dari data keseluruhan dihasilkan nilai akurasi dari aplikasi ini sebesar

90.71%. Adapun saran yang dapat peneliti berikan sebagai pengembangan lebih lanjut untuk sistem ini agar dapat berjalan dengan sempurna dengan fungsi yang lebih baik dengan dipadukan dengan metode lain seperti metode *grid search* untuk *hyperparameter* metode *support vector machine*, agar hasil yang diperoleh lebih baik.

DAFTAR PUSTAKA

[1] Y. Surya, S. Al Faraby, dan M. Dwifebri, "Analisis Sentimen Terhadap Ulasan Film Menggunakan *Word2Vec* dan SVM," *eProceedings Eng.*, vol. 8, no. 4, pp. 4136–4144, 2021.

[2] M. Sari, "Penelitian Kepustakaan (*Library Research*) dalam Penelitian Pendidikan IPA," *NATURAL SCIENCE: Jurnal Penelitian Bidang IPA dan Pendidikan IPA*, vol. 6, no. 1, pp. 41–53, 2020.

[3] N. Arifin, U. Enri, dan N. Sulistiyowati, "Penerapan Algoritma *Support Vector Machine* (SVM) dengan TF-IDF N-Gram untuk *Text Classification*," *STRING (Satuan Tulisan Ris. dan Inov. Teknol.*, vol. 6, no. 2, pp. 129-136, 2021.

[4] A. Rahman Isnain, A. Indra Sakti, D. Alita, dan N. Satya Marga, "Sentimen Analisis Publik Terhadap Kebijakan *Lockdown* Pemerintah Jakarta Menggunakan Algoritma Svm," *Jdmsi*, vol. 2, no. 1, pp. 31–37, 2021.

[5] N. Andriani dan A. Wibowo, "Implementasi *Text Mining* Klasifikasi Topik Tugas Akhir Mahasiswa Teknik Informatika Menggunakan Pembobotan TF-IDF dan Metode *Cosine Similarity* Berbasis *Web*," Prosiding *Seminar Nasional. Mahasiswa Ilmu Komputer dan Aplikasinya*, vol. 2, no. 2, September 2021, pp. 130–137.

[6] M. B. Priyantono, M. Ahnan, M. A. Widhianto, dan D. D. Prasetyo, "Optimasi Sistem Pelabelan Topik Skripsi menggunakan Algoritma *Naive Bayes* dengan Pendekatan *Design Thinking*," *J. Edukasi Penelit. Inform.*, vol. 8, no. 1, pp. 168–174, 2022.

[7] H. Sari, G. L. Ginting, dan T. Zebua, "Penerapan Algoritma *Text Mining* dan TF-IDF Untuk Pengelompokan Topik Skripsi Pada Aplikasi Repository STMIK Budi Darma," *Terap. Inform. Nusant.*, vol. 2, no. 7, pp. 414–432, 2021.

[8] F. S. Jumeilah, "Penerapan *Support Vector Machine* (SVM) untuk Pengkategorian Penelitian," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 1, no. 1, pp. 19–25, 2017.

[9] D. Darwis, E. S. Pratiwi, dan A. F. O. Pasaribu, "Penerapan Algoritma Svm Untuk Analisis Sentimen Pada Data Twitter Komisi Pemberantasan Korupsi Republik Indonesia," *Edutic - Sci. J. Informatics Educ.*, vol. 7, no. 1, pp. 1–11, 2020

[10] O. Somantri, S. Wiyono, dan D. Dairoh, "Metode K-Means untuk Optimasi Klasifikasi Tema Tugas Akhir Mahasiswa Menggunakan *Support Vector Machine* (SVM)," *Sci. J. Informatics*, vol. 3, no. 1, pp. 34–45, 2016.