Jurnal TICOM: *Technology of Information and Communication*
Volume 10, Nomor 3, Mei 2022 Hal: 218-223 E-ISSN: 2962-7982

Implementasi Algoritma Pembobotan TF-IDF dan Cosine Similarity untuk Penetapan Kategori Artikel pada Website Universitas Budi Luhur

Nico[1], Utomo Budiyanto[2*], Titin Fatimah[3]

[1,2,3]Fakultas Teknologi Informasi, Teknik Informatika, Universitas Budi Luhur, Jakarta, Indonesia
Jl. Raya Ciledug Petukangan Utara, Kebayoran Lama, Jakarta Selatan 12260
E-mail: [1]1511501858@student.budiluhur.ac.id, [2*]utomo.budiyanto@budiluhur.ac.id, [3]titin.fatimah@budiluhur.ac.id
(*: corresponding author)

Abstrak— **Universitas Budi Luhur merupakan universitas komputer swasta pertama di Indonesia yang sejak awal berdiri memiliki tujuan menghasilkan tenaga – tenaga trampil atau profesional di bidang komputer yang cerdas dan berbudi luhur. Sebagai institusi pendidikan Universitas Budi Luhur memiliki *website* yang saat ini menggunakan *platform Content Management System* (CMS) Wordpress untuk mempublikasikan kegiatannya. Salah satu masalah dalam penyebaran informasi di *website* adalah kategorisasi artikel yang dipublikasi oleh admin yang dapat mengakibatkan ambiguitas artikel, karena setiap admin memiliki persepsi yang berbeda pada tiap artikel, sehingga menimbulkan kebingungan dari sisi pembaca, juga ketika ingin menampilkan atau mencari artikel/berita yang sesuai dengan kategori. Pengkategorian artikel yang ambigu kerap kali terjadi disebabkan oleh kategori yang dipilih berdasarkan opini masing - masing admin. Penelitian ini bertujuan untuk membuat alat bantu yang dapat mengkategorikan artikel secara otomatis menggunakan teknik *Information Retrieval*. Metode yang digunakan untuk mengelola serta mengkategorikan informasi yaitu ruang vektor (*Vector Space Model*) menggunakan algoritma pembobotan TF-IDF dan *Cosine Similarity*. Hasil uji dari penelitian ini adalah akurasi ketepatan pada klasifikasi kategori terhadap *dataset* sebesar 61.11%.**

Kata Kunci— ***Information Retrieval*, Klasifikasi Artikel, Pembobotan, TF-IDF, *Cosine Similarity*.**

I. PENDAHULUAN

Kegiatan klasifikasi artikel di website Universitas Budi Luhur saat ini dilakukan oleh *user admin* yang bekerja memasukkan artikel melalui *Content Management System*. Pada prosesnya masing-masing admin dapat memiliki persepsi yang berbeda terhadap kategori sebuah artikel. Hal ini bisa menimbulkan ambiguitas bahkan kebingungan di sisi pembaca terhadap kategori dari artikel tersebut. Kategorisasi artikel yang ada di website Universitas Budi Luhur adalah Info Akademik, Informasi dan Acara Kampus.

Untuk dapat mengklasifikasikan artikel secara otomatis sesuai kategori dapat menggunakan teknik *information retrieval*. Salah satu teknik yang digunakan adalah *cosine similarity* serta pembobotan TF-IDF seperti pada penelitian sebelumnya mengenai sinopsis buku [1], abstrak jurnal ilmiah [2], ekstraksi fitur berita di *web* [3], berita *online* [4], otomasi jawaban essay [5], berita di *website* [6], dokumen [7] serta menggabungkan teknik lain seperti K-NN dan *cosine similarity* untuk klasifikasi abstrak jurnal internasional [8].

Penelitian [4] menyatakan bahwa proses *preprocessing* diperlukan untuk menerapkan pengelompokan berita *online* dengan menggunakan algoritma *Single Pass Clustering*. Untuk mempercepat proses perhitungan bobot suatu *term* dapat menggunakan *preprocessing*, sehingga perhitungan persamaan *cosine similarity* dapat lebih cepat. Pengujian dilakukan sebanyak 4 kali yang dapat disimpulkan bahwa tingkat akurasi akan tinggi bila jumlah data uji semakin banyak. Hasil rata-rata pengujian 91.25% dan mempunyai tingkat akurasi paling tinggi pada pengujian ke tiga dengan akurasi 100%. Hal tersebut dapat terjadi karena hasil pada kategori data hasil sistem sesuai dengan data asli.

Berdasarkan studi literatur yang dilakukan, pembobotan TF-IDF dan *cosine similarity* dianggap cocok untuk klasifikasi artikel pada *website*. Karena banyaknya artikel yang dimasukkan ke dalam *website*, sehingga penentuan kategori artikel tersebut terpublikasi kadang tidak sesuai dengan kategorinya. Hal ini disebabkan karena penentuan kategori pada artikel dinilai secara subjektif, dimana setiap admin *website* memiliki penilaian tersendiri atas artikel yang akan dipublikasi.

Penelitian ini akan membuat sebuah alat bantu berupa perangkat lunak yang ditempel ke dalam *Content Management System* berbasis Wordpress untuk mengklasifikasikan kategori pada artikel yang dipublikasi di *website* Universitas Budi Luhur menggunakan *cosine similarity* dan pembobotan TF-IDF

Pengelompokan artikel secara otomatis merupakan proses mencari kategori artikel berdasarkan *dataset* yang telah dipelajari oleh program, dimana artikel dibagi ke dalam kategori yang telah ditentukan dan dapat langsung diketahui kategori artikel tersebut, sehingga penetapan kategori pada suatu artikel dapat dinilai secara obyektif. Proses pengkategorian artikel ini meliputi proses *Data Collections – Text Preprocessing – Text Representation – Term Weighting – Text Similarity Measurement.*

III. METODOLOGI PENELITIAN

A. *Text Mining*

Text mining atau *text analytics* adalah istilah yang mendeskripsikan sebuah teknologi yang mampu menganalisis data teks semi-terstruktur, maupun tidak terstruktur, hal inilah yang membedakannya dengan data *mining* dimana data *mining* mengolah data yang sifatnya terstruktur. Pada dasarnya, text mining merupakan bidang interdisiplin yang mengacu pada perolehan informasi (*information retrieval*), *data mining*, pembelajaran mesin (*machine learning*), statistik dan komputasi *linguistic* [9]. Secara umum konsep pekerjaan *text mining* mirip dengan *data mining*, yaitu prediktir dan penggalian deskriptif. *Text mining* mengekstrak indeks numerik yang bermakna dari teks dan kemudian informasi yang terkandung dalam teks akan diakses dengan menggunakan berbagai algortima *data mining* (statistik dan *machine learning*) [10]. *Text mining* adalah penelitian dunia komputer yang mampu menyelesaikan permasalahan informasi yang berlebih dan merupakan gabungan dari *data mining*, *machine learning*, sistem temu kembali, *knowledge management* dan *natural language processing* [11].

B. Konsep Perolehan Informasi

Definisi umum untuk informasi dalam sistem informasi adalah data yang telah di transformasi menjadi bentuk yang lebih berguna bagi pemakai [12]. Informasi merupakan data yang telah diformat dan atau terorganisir dengan berbagai cara sehingga menjadi berguna bagi orang yang menggunakan. Jadi informasi adalah data yang telah diolah menjadi suatu bentuk dengan berbagai cara sehingga berguna bagi orang yang menggunakannya [13].

Information Retrieval terbagi dari beberapa bagian yang dijabarkan sebagai berikut [14]:

1) *Text Operations*, meliputi pemilihan kata-kata dalam *query* maupun dokumen (*term selection*) dalam proses transformasi dokumen atau *query* menjadi *term index* (indeks kata-kata).
2) *Query formulation*, memberi bobot pada indeks kata-kata *query*.
3) *Ranking*, mencari dokumen-dokumen yang relevan terhadap *query* dan mengurungkan dokumen tersebut berdasarkan kesesuaiannya dengan *query*.
4) *Indexing*, membangun basis data indeks dari koleksi dokumen dilakukan terlebih dahulu sebelum pencarian dokumen dilakukan.

C. Pembobotan TF-IDF

Metode TF-IDF (*Term Frequency – Inverse Document Frequency*) adalah penggabungan dua konsep untuk perhitungan bobot suatu kata (*term*) yaitu frekuensi kemunculan sebuah kata di dalam dokumen tertentu dan *inverse* frekuensi dokumen yang mengandung kata tersebut. Frekuensi kemunculan kata tersebut menunjukan seberapa penting dan seberapa umum kata tersebut di dalam dokumen [15].

D. *Term Frequency (TF)*

Term Frequency (TF) adalah bobot dari suatu kata, t, dalam suatu dokumen, d dan dilambangkan dengan tf_{td}. Pendekatan paling sederhana dari konsep ini adalah dengan menyatakan bobot suatu kata t sebagai jumlah kemunculannya pada dokumen d. Sebagai contoh, jika dalam suatu dokumen, kata plagiat muncul sebanyak 10 kali, maka nilai TF adalah 10.

Konsep *term frequency* memandang suatu dokumen sebagai *bag of words* (kantong data) dimana urutan dari kemunculan suatu kata diabaikan dan hanya jumlah kemunculan dari kata itu saja yang penting.

Konsep *term frequency* memiliki kelemahan yaitu semua kata dianggap setara. Hal ini mengakibatkan relevansi suatu kata menjadi sangat tinggi jika kata itu sering muncul dalam suatu kumpulan dokumen. Padahal tingginya frekuensi kemunculan suatu kata tidak selalu menyatakan bahwa kata tersebut penting.

E. *Document Frequency (DF)*

Document Frequency merupakan jumlah dokumen yang mengandung suatu *term*. Setiap *term* akan dihitung nilai *Document Frequency*-nya (DF). *Term* yang jarang ditemukan dapat mengurangi dimensi fitur yang besar pada *text mining*. Perbaikan dalam pengelompokan dokumen ini juga dapat terjadi jika *term* yang dibuang tersebut juga merupakan *noise term*. *Document Frequency* merupakan metode *feature selection* yang paling sederhana dengan waktu komputasi yang rendah.

F. *Inverse Document Frequency (IDF)*

Konsep *inverse document frequency* (IDF) dibuat untuk mengurangi efek dari kata yang frekuensinya terlalu tinggi dalam kumpulan dokumen. Ide dsasarnya adalah untuk menurunkan bobot dari kata dengan frekuensi kolektif (frekuensi total kemunculan kata di semua dokumen) yang tinggi. Dengan kata lain, semakin banyak dokumen kata tersebut pada suatu kumpulan dokumen, maka semakin rendah bobotnya.

Berapapun besarnya nilai tf_{ij}, apabila N = n, maka akan didapatkan hasil 0 (nol), hal itu dikarenakan hasil dari log 1. Maka untuk perhitungan IDF, dapat ditambahkan nilai 1 pada sisi IDF, sehingga perhitungan bobotnya seperti pada Rumus 1.

$$W_{ij} = tf\ x\ idf$$
$$W_{ij} = tf_{ij}\ x\ \left(\log\frac{N}{n} + 1\right) \qquad (1)$$

Keterangan :

W_{ij} = bobot kata/*term* $t_j$terhadap dokumen d_i.
tf_{ij} = jumlah kemunculan kata/*term* t di dalam dokumen d
N = Jumlah keseluruhan dokumen.
n = jumlah dokumen yang mengandung kata/*term* t_j (minimal ada satu kata yaitu *term* t_j.

G. *Cosine Similarity*

Cosine similarity atau kemiripan kosinus adalah ukuran jarak yang digunakan untuk data yang berupa vektor dokumen. Pada dasarnya sebuah dokumen bisa dipandang sebagai data

yang berisi ratusan atau bahkan ribuan atribut, dimana setiap atribut menyatakan sebuah *term* atau istilah (kata) yang nilainya berupa frekuensi kemunculan istilah dalam dokumen tertentu. Dengan dimikian vektor dokumen adalah sebuah vektor yang menyatakan frekuensi kemunculan kata dalam suatu dokumen. Karena yang hanya diperhitungkan nilai *term* dari masing-masing dokumen adalah ukuran kesamaan antara dua buah vektor dalam sebuah ruang dimensi yang didapat dari nilai *cosinus* sudut dari perkalian dua buah vektor yang dibandingkan karena *cosinus* dari 0 adalah 1 dan kurang dari 1 untuk nilai sudut yang lain, maka nilai *similarity* dari dua buah vektor dikatakan mirip ketika nilai dari *cosine similarity* adalah 1. Perhitungan *cosine similarity* ditunjukan pada Rumus 2:

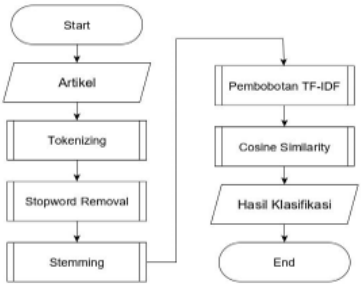
$$Sim\ (\alpha) = \frac{A.B}{|A|\ |B|} = \frac{\sum_{i=1}^{n} A1.B1}{\sqrt{\sum_{i=1}^{n}(A1)^2\ x\ \sum_{i=1}^{n}(B1)^2}} \quad (2)$$

Keterangan :

A = vektor Dokumen
B = vektor Query
A⋅B = dot product antara vektor A dan vektor B
|A| = panjang vektor A
|B| = panjang vektor B
|A||B| = cross product antara |A| dan |B|
α = sudut yang terbentuk antara vektor A & B

H. Analisa dan Perancangan

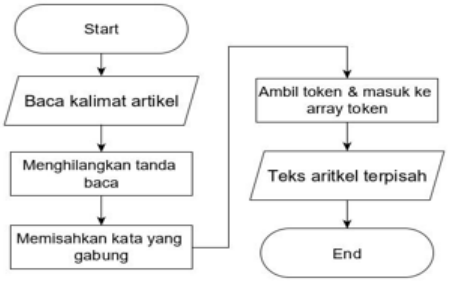
Tahapan analisan dan perancangan penetapan kategori pada artikel website secara umum dapat menggunakan pembobotan *tf-idf* dan *cosine similarity*. Gambar 1 merupakan alur proses penetapan kategori.



Gambar 1. *Flowchart* Proses Penetapan Kategori

I. *Tokenizing*

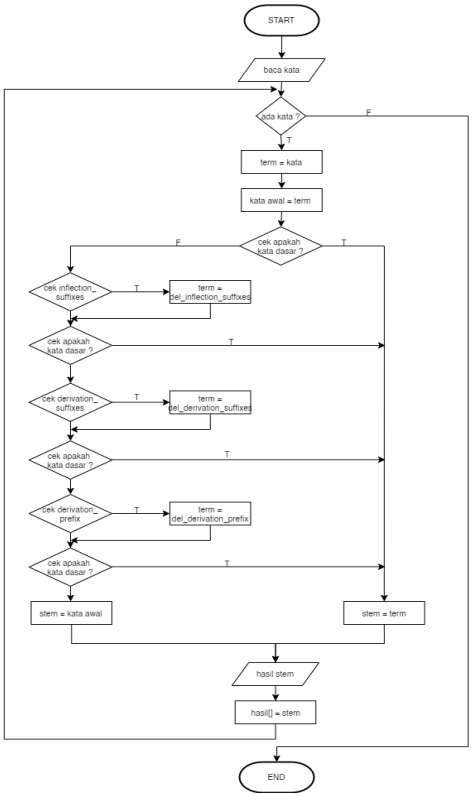
Tokenizing menjelaskan bagaimana kalimat artikel yang di-*input* melakukan proses *tokenizing*. Kalimat yang dimasukan akan dihilangkan tanda baca dan simbol yang kemudian setiap kalimatnya dipisahkan menjadi kata per kata. Lalu kata tersebut diambil dan disimpan ke dalam *array*. Gambar *flowchart* proses *tokenizing* dapat dilihat pada Gambar 2.



Gambar 2. *Flowchart* Proses Tokenizing

J. *Stemming*

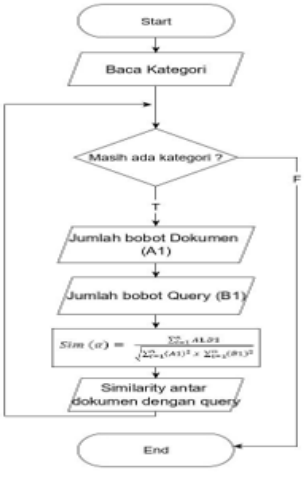
Tahap ini menjelaskan bagaimana proses *stemming* atau proses mengubah suatu kata berimbuhan menjadi kata dasar. Seperti "berlari" menjadi "lari", "menyapu" menjadi "sapu", "menukar" menjadi "tukar", "memukul" menjadi "pukul", "berpikir" menjadi "pikir", dan lain lain. Gambar *flowchart* proses *stemming* dapat dilihat pada Gambar 3.



Gambar 3. *Flowchart* Proses Stemming

K. *Stopword*

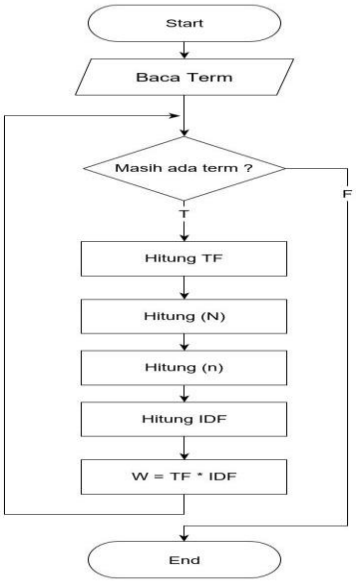
Stopword menjelaskan bagaimana proses menghilangkan kata yang dianggap tidak memiliki arti signifikan pada suatu kalimat. Seperti "yang", "dan", "di" dan sebagainya. *Flowchart* proses *stopword* dapat dilihat pada Gambar 4.



Gambar 4. *Flowchart* Proses Stopword

L. Pembobotan TF-IDF

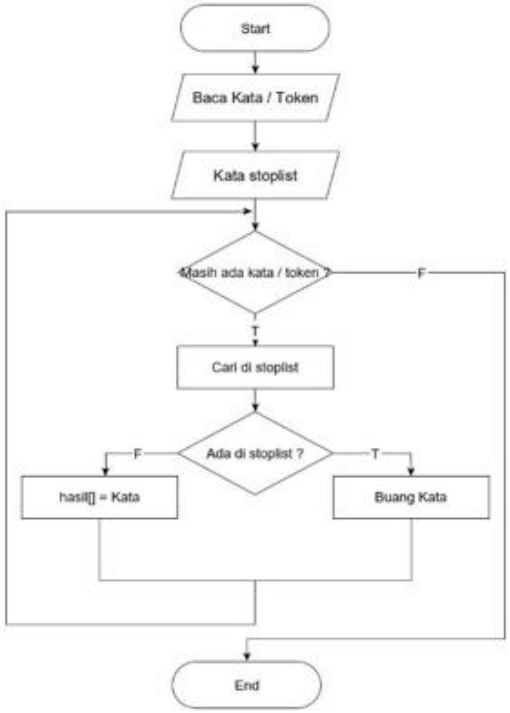
Pembobotan TF-IDF menjelaskan bagaimana proses pembobotan TF-IDF berjalan pada pengklasifikasian artikel. Dimana TF (*Term Frequency*) adalah jumlah kemunculan kata/*term* dalam dokumen, N adalah jumlah keseluruhan dokumen, dan n adalah jumlah dokumen yang mengandung kata/*term* tertentu. *Flowchart* proses pembobotan TF-IDF dapat dilihat pada Gambar 5.



Gambar 5. *Flowchart* Proses Pembobotan TF-IDF

M. *Cosine Similarity*

Flowchart pada Gambar 6 menjelaskan bagaimana proses perhitungan kesamaan dengan menggunakan metode *Cosine Similarity*.

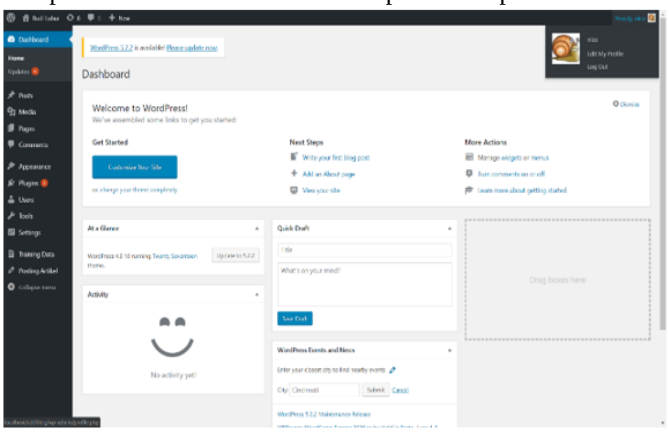


Gambar 6. *Flowchart* Proses Perhitungan *Cosine Similarity*

IV. Hasil Dan Pembahasan

A. *Tampilan Layar Utama*

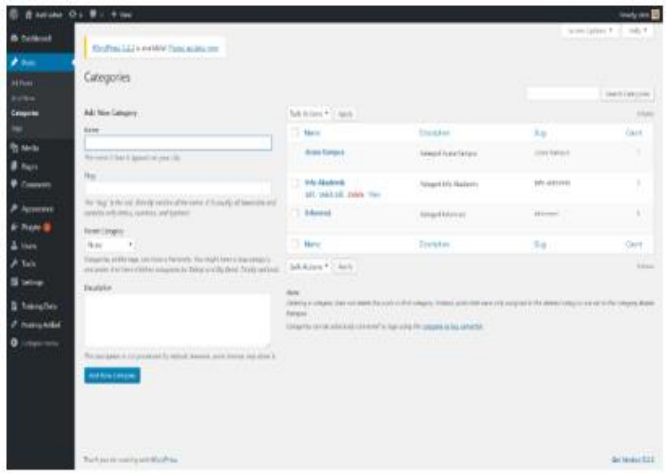
Pada halaman utama ada beberapa tugas yang dilakukan, seperti membuat data *categories*, membuat data *tags*, melakukan *training* data, dan melakukan *posting* artikel. Tampilan halaman *administrator* dapat dilihat pada Gambar 7.



Gambar 7. Tampilan Layar Menu Utama

B. *Tampilan Layar Categories*

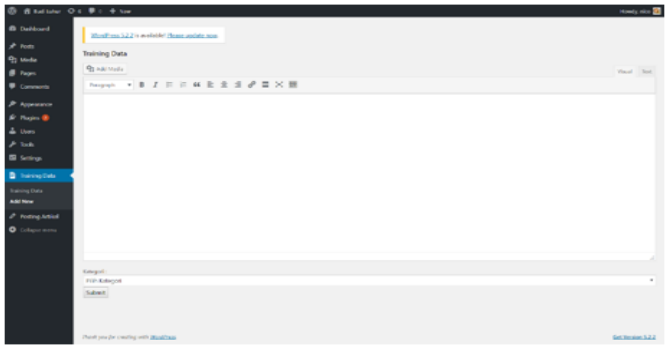
Pada tampilan layar Categories terdapat *form* untuk mengisi data kategori. *Field name, slug, description* merupakan data yang diperlukan oleh sistem CMS Wordpress. Data *gridview* akan menampilkan semua data kategori yang tersimpan oleh sistem CMS Wordpress. Tampilan layar *categories* dapat dilihat pada Gambar 8.



Gambar 8. Tampilan Layar Data Kategori

C. *Tampilan Layar Input Training*

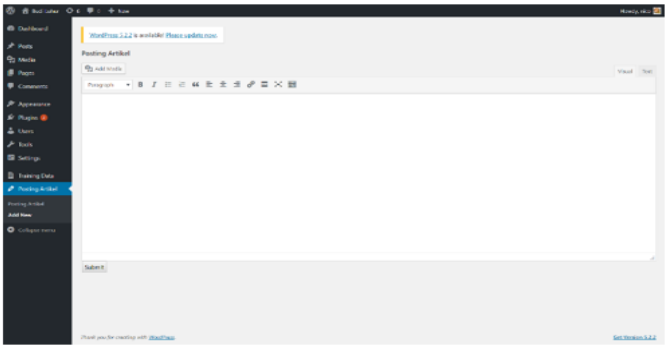
Pada tampilan layar *training* artikel terdapat kolom untuk mengisi artikel, pilihan kategori, dan tombol *training*. Kategori yang dapat dipilih atau ditampilkan hanya data kategori yang telah terdaftar oleh sistem CMS Wordpress. Tombol *training* akan menjalankan fungsi pelatihan mesin dengan data data yang telah di-*input*. Tampilan layar input data *training* artikel dapat dilihat pada Gambar 9.



Gambar 9. Tampilan Layar Input Data Training

D. Tampilan Layar Klasifikasi Artikel

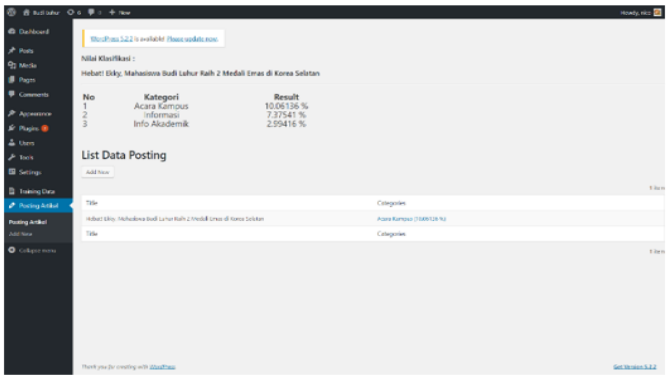
Pada tampilan layar klasifikasi artikel terdapat kolom untuk mengisi artikel dan tombol *publish*. *User* mengisi konten artikel pada kolom "isi artikel". Setelah mengisi konten artikel, selanjutnya user menekan tombol *publish* untuk menampilkan artikel pada halaman *website*. Pada halaman ini, *user* tidak diminta untuk memilih kategori yang tepat untuk artikel yang *user posting*, karena secara otomatis program akan mengklasifikasi artikel tersebut. Tampilan layar klasifikasi artikel dapat dilihat pada Gambar 10.



Gambar 10. Tampilan Layar Input Data Klasifikasi

E. Tampilan Layar Hasil Klasifikasi

Pada tampilan layar detail data klasifikasi artikel, terdapat layout artikel yang menampilkan nilai bobot artikel yang dipilih terhadap kategori-kategori yang tersedia. *Datagridview* menampilkan data klasifikasi artikel yang telah tersimpan dalam sistem CMS Wordpress. Tampilan layar detail data klasifikasi dapat dilihat pada Gambar 11.



Gambar 11. Tampilan Layar Hasil Klasifikasi

F. Pengujian

Pada tahap pengujian data, nilai akurasi yang akan menjadi tolak ukur dalam pengujian ini. *Dataset* yang digunakan sebanyak 90 data artikel yang terdiri dari 3 kategori. Masing masing kategori terdiri dari 30 artikel. Data yang digunakan untuk melakukan pengujian sebanyak 20% dari total jumlah *dataset* yang ada. Sehingga jumlah data pengujian sebanyak 18 data, dimana setiap kategorinya terdapat 6 data artikel. Hasil pengujian dapat dilihat pada Tabel 1.

Tabel 1 Tabel Pengujian Algoritma

Data Latih	Data Uji	Akurasi
80%	20%	61.11%

Untuk mendapatkan hasil akurasi pada pengujian data diatas, digunakan uji kepakaran. Hasil pengujian dapat dilihat pada Tabel 2.

Tabel 2 Tabel Hasil Pengujian Pakar

Terklasifikasi Benar	Terklasifikasi Salah	Total
11	7	18

V. PENUTUP DAN SARAN

Penelitian ini berhasil membuat alat bantu berupa perangkat lunak yang bertujuan untuk mengklasifikasi kategori artikel di website Universitas Budi Luhur dengan menggunakan metode *Vector Space Model*, persamaan *Cosine Similarity* dan pembobotan TF-IDF dengan beberapa poin sebagai berikut: (1) *Tools* dapat diterapkan pada sistem CMS Wordpress, (2) Pada saat publikasi artikel, admin website tidak lagi harus memahami isi konten *website* untuk menentukan kategori yang tepat dari artikel tersebut, dan (3) Akurasi ketepatan pada klasifikasi kategori terhadap *dataset* yang telah diuji menghasilkan nilai sebesar 61.11%.

Saran untuk penelitian berikutnya dapat menggunakan metode *information retrieval* maupun *machine learning* lainnya agar akurasi yang dihasilkan dapat meningkat.

REFERENSI

[1] M. M. Sya'bani and R. Umilasari, "Penerapan Metode Cosine Similarity dan Pembobotan TF/IDF pada Sistem Klasifikasi Sinopsis Buku di Perpustakaan Kejaksaan Negeri Jember," *JUSTINDO (Jurnal Sistem dan Teknologi Informasi Indonesia)*, vol. 3, no. 1, pp. 31-42, 2018.

[2] P. M. Hasugian, J. Manurung, L. Logaraz, U. Ram, "Implementation of TF-IDF and Cosine Similarity Algorithms for Classification of Documents Based on Abstract Scientific Journals," *INFOKUM*, vol. 9, no. 2, pp. 518-526, 2021.

[3] M. Xu, L. He and X. Lin, "A Refined TF-IDF Algorithm Based on Channel Distribution Information for Web News Feature Extraction," *2010 Second International Workshop on Education Technology and Computer Science*, 2010, pp. 15-19.

[4] B. Herwijayanti, D.E. Ratnawati and L. Muflikhah. "Klasifikasi Berita Online dengan menggunakan Pembobotan TF-IDF dan Cosine Similarity," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 2, no. 1 pp. 306-312, 2018.

[5] N. D. Arianti, M. Irfan, U. Syaripudin, D. Mariana, N. Rosmawarni and D. S. Maylawati, "Porter Stemmer and Cosine Similarity for Automated

Essay Assessment," *2019 5th International Conference on Computing Engineering and Design (ICCED)*, 2019, pp. 1-6.

[6] A. N. Khusna and I. Agustina, "Implementation of Information Retrieval Using Tf-Idf Weighting Method On Detik.Com's Website," *2018 12th International Conference on Telecommunication Systems, Services*, and Applications (TSSA), 2018, pp. 1-4.

[7] A. Mishra and S. Vishwakarma, "Analysis of TF-IDF Model and its Variant for Document Retrieval," *2015 International Conference on Computational Intelligence and Communication Networks (CICN)*, 2015, pp. 772-776.

[8] M. Nursalman, J. Kusnendar and U. F. Fadhila, "Implementation of K-Nearest Neighbor with Cosine Similarity for Classification Abstract International Journal of Computer Science," *2018 International Conference on Information Technology Systems and Innovation (ICITSI)*, 2018, pp. 43-48.

[9] H. Jiawei, M. Kamber, and J. Pei, *Data Mining. Concepts and Techniques*, 3rd Edition. 2012.

[10] G. Miner, J. Elder, A. Fast, T. Hill, R. Nisbet, and D. Delen, "*Practical Text Mining and Statistical Analysis for Non-structured Text Data Applications*," Academic Press, 2012.

[11] R. Feldman and J. Sanger, "*The Text Mining Handbook," Cambridge*: Cambridge University Press, 2006.

[12] G. B. Shelly, T. J. Cashman, and H. J. Rosenblatt, "*Systems analysis and design*," 9th ed. Boston: Course Technology, 2012.

[13] J. Valacich and C. Schneider, "*Information systems today: managing in the digital world*," 5th ed. Boston: Prentice Hall, 2012.

[14] H. Bunyamin, C. P. Negara, and Informasi, "Aplikasi Information Retrieval (IR) CATA Dengan Metode Generalized Vector Space Model," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 4, no. 1, pp. 29–38, 2017

[15] Suprianto, Sunardi and A. Fadlil, "Aplikasi Sistem Temu Kembali Angket Mahasiswa Menggunakan Application of Information Retrieval for Opinion Student," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 6, no. 1, pp. 33–40, 2019.