SIGMA
JURNAL TEKNOLOGI PELITA BANGSA

Vol. 12 No. 4 Desember 2021 p-ISSN: 2407-3903 e-ISSN: 2407-3903

Diterima, 16 November 2021 | Direvisi, 14 Desember 2021 | Dipublikasikan, 28 Desember 2021

ANALISA EXTRASI INFORMASI PADA ABSTRAKSI JURNAL SKRIPSI BERBAHASA INDONESIA MENGGUNAKAN ALGORITMA *K NEAREST NEIGHBOR*

Donny Maulana[1], Asep Saepudin[2]

Program Studi Teknik Informatika, Fakultas Teknik, Universitas Pelita Bangsa
[1]donny.maulana@pelitabangsa.ac.id, [2]asep051294@gmail.com

Abstraksi

Ekstraksi Informasi merupakan pengambilan fakta dan informasi terstruktur dari isi koleksi teks yang besar. Pengertian fakta disini adalah beragam entitas yang diperhitungkan atau keterhubungan dalam bentuk informasi terstruktur sebagai masukan untuk basis data. Mengolah data ekstraksi informasi pada abstrak jurnal skripsi menggunakan algoritma KNN dimulai dari tahap seleksi data (atribut yang digunakan dan penentuan data training serta data testing), tahap pengujian algoritma (KNN), dan tahap uji akurasi (menggunakan *split validation*) . Proses klasifikasi pada abstrak jurnal skripsi menggunakan algoritma KNN adalah salah satu cara dalam mengklasifikasikan ekstraksi informasi pada abstrak jurnal skripsi. Proses klasifikasi pada abstrak jurnal skripsi menggunakan algoritma KNN digunakan untuk menghindari kesalahan ekstraksi informasi pada abstrak jurnal skripsi. Mengolah data dimulai dari tahap praprocessing data dan perhitungan text mining yang terdiri dari pembobotan term frequency dan pembobotan concept frequency dan *Cosine Similarity* D7 0.0332, D15 0, D10 0,1296, D14 0,1296.

Kata kunci: Text Mining, Extrasi Informasi, K-NN.

Abstract

Information Extraction is the extraction of structured facts and information from the contents of a large collection of texts. The definition of facts here is a variety of entities that are calculated or connected in the form of structured information as input to the database. Processing information extraction data in thesis journal abstracts using the KNN algorithm starts from the data selection stage (attributes used and determination of training data and data testing), the algorithm testing stage (KNN), and the accuracy test stage (using split validation). The classification process in thesis journal abstracts using the KNN algorithm is one way to classify information extraction in thesis journal abstracts. The classification process in the thesis journal abstract using the KNN algorithm is used to avoid information extraction errors in the thesis journal abstract. processing data starting from the data preprocessing stage and text mining calculations consisting of weighting term frequency and weighting concept frequency and Cosine Similarity D7 0.0332, D15 0, D10 0.1296, D14 0.1296

Keywords*: Text Mining, Information Extraction, K-NN*

1. Pendahuluan

Ekstraksi Informasi merupakan pengambilan fakta dan informasi terstruktur dari isi koleksi teks yang besar. Pengertian fakta disini adalah beragam entitas yang diperhitungkan atau keterhubungan dalam bentuk informasi terstruktur sebagai masukan untuk basis data. Jadi ekstraksi informasi adalah sebuah proses mendapatkan fakta-fakta terstruktur dari data yang tersedia. Penelitian skripsi atau karya tulis ilmiah telah dilakukan untuk abstrak jurnal skripsi berbahasa Indonesia dan Penelitian tersebut melakukan ekstraksi informasi menggunakan sistem berbasis aturan untuk mendapatkan identitas pada jurnal skripsi.

Penelitian jurnal skripsi akan memuat intisari laporan peneliti yang disajikan secara padat dan jelas. Ketika menulis jurnal mahasiswa perlu memperhatikan penggunaan bahasanya agar gagasan dan hasil penelitiannya dapat tersampaikan dengan baik. Untuk dapat memaparkan suatu hasil penelitian seorang mahasiswa sekaligus peneliti, dilatih untuk terampil menerapkan aspek kebahasaan, seperti

kosakata, tata bahasa, ejaan, dan tata bunyi. Dalam kaitannya dengan aspek kebahasaan extrasi informasi.

Dalam analisa extrasi informasi abstrak jurnal skripsi menggunakan *Algoritma K-Nearest Neighb*or (K-NN) merupakan sebuah metode klasifikasi terhadap sekumpulan data berdasarkan pembelajaran data yang sudah terklasifikasikan sebelumya. Termasuk dalam supervised learning, dimana hasil query instance yang baru diklasifikasikan berdasarkan bahasa dan tujuan jurnal. Berdasarkan uraian tersebut, maka peneliti melakukan penelitian untuk Tugas Akhir yang diberi judul "Analisa Extrasi Informasi Jurnal Skripsi Bahasa Indonesia Dengan Menggunakan Algoritma *K Nearest Neighbor*"

2. Metode Penelitian

Dalam penelitian ini peneliti menggunakan pendekatan Text Mining untuk *melakukan* analisis data. Data yang akan dijadikan dataset dalam penelitian ini adalah data abstrak jurnal dan didapatkan dalam bentuk *file spreadsheet* berformat *excel*.

2.1. *Preprocessing* Data

Tahap awal sebelum melakukan proses pengelompokkan dokumen adalah mempersiapkan teks yang ada di dalam dokumen. Pada tahapan preprocessing ini dilakukan beberapa subproses diantaranya yaitu:

2.1.1. Tokenizer

Proses yang bertujuan untuk memisahkan teks menjadi beberapa token berdasarkan pembatas berupa spasi atau tanda baca. Proses *tokenizer* akan ditunjukkan pada Tabel 1.

Tabel 1. Proses *Tokenizer*

Teks Input	Teks Output
proyek	proyek
akhir	akhir
waterfall	waterfall

2.1.2. Stopword

Proses selanjutnya adalah menghilangkan teks yang tidak sesuai dengan teks yang terdapat pada daftar *stopword*, karena teks tersebut dianggap tidak dapat mewakili konten dokumen. Proses tersebut akan ditunjukkan pada Tabel 2.

Tabel 2. S*topword*

Teks Input		Teks Output
proyek	menggunakan	proyek
akhir	Metode	metode
dirancang	waterfall	waterfall

2.1.3. Stemming

Tabel 2.Tabel *Software* dan *Hardware* Pendukung

Teks Input	Teks Output
proyek	proyek
akhir	akhir
dirancang	rancang

2.2. Algoritma Term Frequency-Inverse Document Frequency

Algoritma Term Frequency-Inverse Document Frequency adalah salah satu algoritma yang dapat digunakan untuk menganalisa hubungan antara sebuah frase atau kalimat dengan sekumpulan dokumen. Inti utama dari algortima ini adalah melakukan perhitungan nilai TF dan nilai IDF dari setiap kata kunci terhadap masing-masing dokumen. Algoritma ini akan menghitung bobot setiap token t di dokumen dengan rumus 1.

$$W_{dt} = tf_{dt} * IDF_t \quad (1)$$

Dimana: d: dokumen ke-d t: kata ke-t dari kata kunci W: bobot dokumen ke-d terhadap kata ke-t tf: banyaknya kata yang dicari pada sebuah dokumen IDF: *Inversed Document Frequency* Nilai IDF didapatkan dari:

IDF t =log(D|df t) (2)
D : total dokumen, df : banyak dokumen yang mengandung kata yang dicari.

Concept Frequency-Inverse Frequency Document (CF-IDF) merupakan metode pengembangan dari metode *Term Frequency-Inverse Document Frequency* (TF-IDF) yang lebih dahulu popular. *Algoritma* CF-IDF adalah algoritma perhitungan bobot kesesuaian dokumen, pembobotan merupakan proses pemberian bobot terhadap kata yang telah dihasilkan dari tahap sebelumnya. Pada perhitungan metode ini tidak melakukan perhitungan terhadap term (seperti pada TF-IDF) tetapi dengan menghitung *key concept* yang ditemukan didalam pesan. Setiap kata yang muncul pada dokumen akan dipetakan ke dalam wordnet ke dalam sebuah konsep yang memiliki makna yang sama. Setelah itu kemudian dihitung bobot tiap dokumen permasalahan yang memiliki kemiripan (similaritas) dengan teks pencarian yang dimasukan oleh pengguna. Perhitungan bobot tersebut dengan menghitung terlebih dahulu frekuensi konsep dalam dokuen (CF) dan frekuensi dokumen yang terdapat kemunculan dokumen konsep (DF). Pada tahap pertama dalam metode CF-IDF yaitu melakukan pembobotan dengan menghitung CF (*Concept Frequency*):

$$cf_{ij} = \frac{ni,j}{\sum_k n_{k,j}} \quad (3)$$

Dimana $cf\ ij$ = *rasio frekuensi concept* pada dokumen ni,j = jumlah kemunculan *concept* dalam dokumen $\sum k\ nk,j$ = total kemunculan seluruh *concept* dalam dokumen Setelah itu, dilakukan perhitungan nilai IDF dengan membagi jumlah total dokumen dengan jumlah dokumen yang terdapat kemunculan konsep (Ci).

$$idf_i = log \frac{[D]}{[\{d: c_i \in d\}]} \quad (4)$$

Dimana idf = *rasio frekuensi dokumen* [D] = jumlah total dokumen [{d: $c_i \in d$}] = jumlah dokumen yang terdapat kemunculan concept Pada tahap terakhir nilai CF dikalikan dengan IDF. $W = cf\ ij * idf\ i$ Dimana, W = bobot CF-IDF $cf\ ij$ = *rasio frekuensi concept* pada dokumen $idf\ i$ = *rasio frekuensi* dokumen

2.3. Tabel

Setelah tahap praprocessing selesai dilakukan, tahap selanjutnya yaitu menghitung frekuensi kemunculan kata (*term*) pada dokumen uji dan dokumen latih, serta menghitung frekuensi jumlah dokumen yang mengandung kemunculan kata (*Document Frequency*) hasil dari perhitungan tersebut ditulis secara singkat pada Tabel 2.1

Tabel 2.1 Hasil Perhitungan Term Frequence

Konsep	TF					DF
	Q	D7	D10	D14	D15	
Proyek	0	1.	0	0	0	1
Akhir	0	1	0	0	0	1
Rancang	0	1	0	0	1	2
Metode	1	1	1	1	0	4
Waterfall	0	1	0	0	0	1

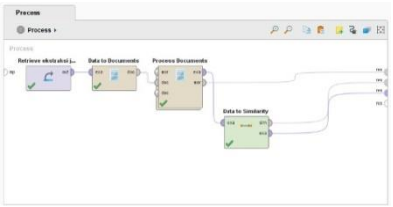
Analisa perhitungan dengan metode K-Nearest Neighbor akan menggunakan rumus cosine similarity. Nilai yang akan dijadikan dasar perhitungan dari rumus cosine similarity adalah nilai akar jumlah bobot per dokumen yang dijabarkan pada Tabel 4.3 dan nilai jumlah bobot per dokumen yang dijabarkan pada Tabel 4.4 Tahap selanjutnya yaitu mencari nilai kedekatan antar dokumen uji dengan dokumen latih. Nilai persentase kedekatan akan ditunjukkan pada Tabel 2.2

Tabel 2.2 Persentase Hasil Perhitungan *Cosine Similarity*

Dokumen	*Cosine Similarity*	Presesntase
D7	0.1296 / 2.34 * 0.6 = 0.0332	0.0332 %
D10	0.1296 / 0.6 * 0.6 = 0.1296	0.1296 %
D14	0.1296 / 0.6 * 0.6 = 0.1296	0.1296 %
D15	0 / 1.01 * 0.6 = 0	0 %

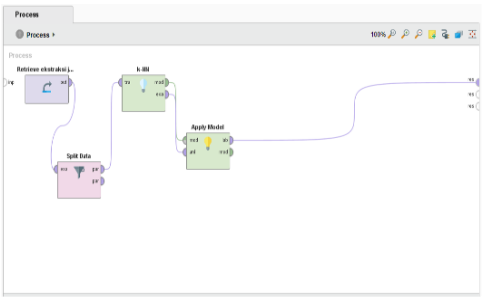
2.4. Praprocessing

Pemodelan *praprocessing* data terdiri dari beberapa proses diantaranya yaitu *tokenize* yang berfungsi untuk memecah kalimat menjadi beberapa kata, filter *stopwords* yang berfungsi untuk menghilangkan kata sambung, *filter tokens* dan *stem* (*Dictionary*) yang berfungsi untuk mengubah kata menjadi kata dasar.



Gambar 1. Pemodelan *Cosine Similarity*

Melakukan *select attributes* yaitu untuk mengetahui hasil prediksi dari RapidMiner, hasil perhitungan manual dan hasil uji di *RapidMiner*.



Gambar 2. Proses Evaluasi Model Rapidminer

3. Pembahasan

Analisis hasil dari pengujian ini bertujuan untuk mengklasifikasikan proses ekstraksi informasi pada abstrak jurnal skripsi menggunakan text mining dengan metode *K-Nearest Neighbor* (*Cosine Similarity*). Proses pengujian dilakukan untuk mengetahui nilai akurasi dalam proses klasifikasi ekstraksi informasi pada abstrak jurnal skripsi. Dibawah ini adalah table perbandingan antara hasil perhitungan manual dengan implementasi di *RapidMiner*.

Tabel 2.3 Hasil Perhitungan *RapidMiner*

Dokumen	*Keyword*	*Cosine Similarity* (*RapidMiner*)
D7	Metode	0.519
D15	Kesimpulan	0.349
D10	Kedua	0.653
D14	Keputusan	0.419

3. Penutup

Text mining adalah satu langkah dari analisis teks yang dilakukan secara otomatis oleh komputer untuk menggali informasi yang berkualitas dari suatu rangkaian teks yang terangkum dalam sebuah dokumen. Dalam mengekstraksi abstrak untuk mendapatkan identitas pada abstrak jurnal skripsi pada penelitian ini menggunakan *Text mining*.

Prosedur utama dalam metode ini terkait dengan menemukan kata-kata yang dapat mewakili isi dari dokumen untuk selanjutnya dilakukan analisis keterhubungan antar dokumen dengan menggunakan metode statistik tertentu seperti analisis kelompok,

klasifikasi dan asosiasi. tahapan dalam text mining secara umum adalah *tokenizing, filtering, stemming, tagging*, dan *analyzing*. *Text Mining* mengolah data dimulai dari tahap *praprocessing* data dan perhitungan text mining yang terdiri dari pembobotan *term frequency* dan pembobotan *concept frequency*. Melakukan pengujian menggunakan *algoritma* lain yang belum pernah dilakukan sebelumnya pada penelitian ini

Referensi

[1] A. Sulaiman, N. Indriani, J. Dipati, and U. Bandung, "Ekstraksi Informasi Pada Dokumen Surat Masuk Menggunakan Algoritma Fuzzy K-Nearest Neighbour (Fuzzy K-NN) Teknik Informatika - Universitas Komputer Indonesia," pp. 1–8.

[2] S. Basuki, G. I. Marthasari, and D. Arifandi, "Penelitian Berbahasa Indonesia Berbasis Fitur," Semin. Nas. Teknol. dan Rekayasa, pp. 15–20, 2018.

[3] J. Ilmiah et al., "EKSTRAKSI INFORMASI UNTUK NOVEL Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)," vol. 8, no. 1, 2019.

[4] B. Junrio, K. K. Purnamasari, J. Dipati, U. No, K. Bandung, and J. Barat, "PADA KARYA TULIS ILMIAH DENGAN GENERALIZED HIDDEN MARKOV MODEL."

[5] B. Amil et al., "No 主観的健康感を中心とした在宅高齢者における 健 康関連指標に関する共分散構造分析Title," J. Chem. Inf. Model., vol. 21, no. 1, pp. 1–9, 2020.

[6] F. Sasmita and K. K. Purnamasari, "Ekstraksi Informasi Dokumen Karya Tulis Ilmiah Menggunakan Algoritma Learning Vector Quantization," Skripsi, Univ. Komput. Indones., p. 8, 2017.

[7] F. Sukmana and F. Rozi, "Pengembangan Aplikasi Ektraksi Informasi Abstrak Dokumen Skripsi Menggunakan Javafx," JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform., vol. 3, no. 1, pp. 38–44, 2018, doi: 10.29100/jipi.v3i1.653.

[8] P. N. Lhokseumawe, K. Pengantar, rahayu deny danar dan alvi furwanti Alwie, A. B. Prasetio, and R. Andespa, Tugas Akhir Tugas Akhir, vol. 2, no. 1. 2010.

[9] J. Ilmiah, I. Komputa, D. Mustaqwa, N. I. Widiastuti, T. Informatika, and U. Komputer, "BERBASIS ATURAN Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)," vol. 7, no. 2, 2018.

[10] Suyanto, Data Mining. Yogyakarta: Informatika, 2017.

[11] G. widi N. Dicky Nofriansyah, Algoritma Data Mining Dan pengujian. Yogyakarta: Cv Budi Utama, 2015.

[12] Retno Tri vulandari, Data Mining. Yogyakarta: Gava Media, 2017.

[13] O. Villacampa, "(Weka - Thesis) Feature Selection and Classification Methods for Decision Making: A Comparative Analysis," ProQuest Diss. Theses, no. 63, p. 188, 2015.