BAB 2
LANDASAN TEORI

2.1 Penelitian Terdahulu

Berikut merupakan beberapa penelitian sebelumnya terkait dengan penggunaan algoritma *logistic regression* dalam melakukan klasifikasi teks yang akan dijadikan acuan pada penelitian yang akan dilakukan. Tabel penelitian terdahulu dapat dilihat pada tabel 2.1 penelitian terdahulu.

Tabel 2.1. Penelitian Terdahulu

No.	Penulis	Judul	Metode	Sumber Data	Jumlah Data	Akurasi
1	Ravikant Kholwal	*Text-Classify: A Comprehensive Comparative Study of Logistic Regression, Random Forest, and Knn Models for Enhanced Text Classification Performance* [5]	*Logistic Regression, Random Forest, and K-nearest neighbour*	Dataset mengenai berita dari BBC (British Broadcasting Corporation)	+1300	*Logistic Regression:* 97%, *Random Forest:* 93%, *K-nearest neighbour: 92%*

Tabel 2.1. Penelitian Terdahulu (lanjutan)

No.	Penulis	Judul	Metode	Sumber Data	Jumlah Data	Akurasi
2	Satya Abdul Halim Bahtiar, Chandra Kusuma Dewa, Ahmad Luthfi	*Comparison of Naïve Bayes and Logistic Regression in Sentiment Analysis on Marketplace Reviews Using Rating-Based Labeling* [6]	*Naïve Bayes and Logistic Regression*	Ulasan pengguna mengenai aplikasi *marketplace* dari Google Play Store	36000 (9000 jumlah data per empat label)	*2-label dataset: Naïve Bayes:* 84.33% textitLogistic Regression: 84.58% *3-label dataset: Naïve Bayes:* 70.75%, *Logistic Regression*: 73.05%
3	Arash Hajikhani, Arho Suominen	*Mapping the sustainable development goals (SDGs) in science, technology and innovation: application of machine learning in SDG-oriented artefact detection* [7]	*Naïve Bayes, Linear Support Vector Machine, Logistic Regression*	Judul, abstrak, dan kata kunci dari ppublikasi dari Scopus SciVal	32000 (2000 jumlah data per 16 kategori)	Akurasi (*F1-Score*) tertinggi dari setiap 5 skenario: *Naïve Bayes*: 0.85, *Linear SVM*: 0.79, *Logistic Regression*: 0.88, *Logistic Regression with Word2Vec*: 0.83, *Logistic Regression with Doc2Vec*: 0.85

Berdasarkan pada penelitian terdahulu pada tabel 2.1 penelitian terdahulu, dapat disimpulkan bahwa pada jurnal ketiga merupakan acuan utama dalam melakukan penelitian ini karena topik yang diteliti pada penelitian tersebut adalah mengenai pengelompokkan publikasi atau penelitian ke kategori-kategori yang ada dari *Sustainable Development Goals* (SDG) dari Perserikatan Bangsa-Bangsa (PBB) yang terdapat penggunaan algoritma *logistic regression* itu sendiri dan beberapa metode yang akan digunakan dalam pembangunan model *logistic regression* seperti penggunaan *Word2Vec* dan pembagian *dataset* yaitu 70% *dataset* digunakan untuk melatih model dan 30% *dataset* digunakan untuk menguji model. Jurnal pertama dan kedua mengacu pada metode-metode alternatif yang digunakan seperti proses pada *data preprocessing*, penggunaan *Term Frequency - Inverse Document Frequency* (TF-IDF), dan pembagian *dataset* yaitu pada jurnal pertama menggunakan pembagian dataset 75% untuk melatih model dan 25% untuk menguji model serta pada jurnal kedua menggunakan pembagian dataset 80% untuk melatih model dan 25% untuk menguji model. Pada jurnal ketiga dijelaskan juga alasan dalam penggunaan algoritma *machine learning* dibandingkan dengan algoritma *deep learning* dalam melakukan klasifikasi atau pengelompokan teks ke tujuan-tujuan SDG. Hal mendasar yang menjadi alasan dari penggunaan algoritma dari *machine learning* sendiri adalah implementasi yang mudah untuk dilakukan dibanding penggunaan *deep learning* yang terbilang kompleks. Selain itu, penggunaan metode untuk melakukan *data preprocessing* seperti TF-IDF, vektorisasi, dan metode lainnya adalah pilihan yang popular untuk digunakan untuk algoritma *machine learning* yang dimana dapat membantu mengkonversi suatu teks menjadi bilangan vektor fitur.

2.2 Klasifikasi Teks

Klasifikasi teks adalah tugas yang sangat penting dan telah menjadi lebih populer berkat kemajuan baru dalam bidang penambangan teks dan *Natural Language Processing* (NLP). klasifikasi teks memiliki tiga metode, yaitu klasifikasi biner, multi-klasifikasi, dan klasifikasi *multi-label* [8]. Semua metode tersebut memiliki tujuan yang sama, yaitu memberikan label atau kategori yang telah ditetapkan untuk teks yang diberikan atau dimasukkan oleh pengguna ataupun sistem. Contoh-contoh dari klasifikasi teks yang sering digunakan berupa pengambilan informasi, pelabelan pada topik, analisis sentimen, dan klasifikasi berita [9]. Penggunaan klasifikasi teks sendiri disarankan untuk menggunkan

sistem yang otomatis seperti menggunakan *machine learning* (ML) dibandingkan melakukan klasifikasi secara manual yang berdasar pada pemahaman manusia itu sendiri. Hal ini dikarenakan akurasi yang dihasil oleh manusia sendiri bisa berbeda-beda dari faktor-faktor manusia, seperti kelelahan dan perbedaan keahlian. Selain itu, klasifikasi secara manual akan memakan banyak waktu dan akan sangat menantang dibanding dilakukan secara otomatis jika melakukan proses klasifikasi pada data dalam jumlah besar [10].

2.3 Logistic Regression

Algoritma *logistic regression* adalah metode analisis statistik yang membangun model statistik untuk menggambarkan hubungan antara hasil biner yang berupa data ya atau tidak (variabel dependen atau respons) dan satu set variabel prediktor atau penjelas independen [11]. *Logistic regression* juga merupakan metode yang mudah dan efisien dalam masalah klasifikasi secara biner dan linear. Walaupun *logistic regression* merupakan metode statistik untuk klasifikasi secara biner, tetapi dapat digeneralisasikan ke klasifikasi multikelas [12].*Logistic regression* dapat dianggap sebagai perluasan dari *linear regression* yang memiliki cara kerja yang mirip dengan *linear regression* dan juga termasuk dalam generalisasi model linear [13]. Yang membedakan kedua algoritma ini adalah *logistic regression* digunakan untuk dapat menghasilkan data yang bersifat kategori, sedangkan *linear regression* digunakan untuk dapat menghasilkan data yang bersifat kontinu [14]. Algoritma ini mengestimasi hubungan antara satu variabel dependen dan variabel independen, di mana *logistic regression* menggunakan logaritma peluang sebagai variabel dependen yang juga merupakan kombinasi linear dari satu atau lebih dari variabel independen yaitu prediktor.

Terdapat sebuah fungsi yang dapat mengubah logaritma peluang menjadi probabilitas yaitu fungsi logistik yang menggunakan logit sebagai unit pengukurannya dari unit logistik [15]. *Logistic regression* juga merupakan salah satu algoritma *machine learning* dalam klasifikasi yang digunakan untuk melakukan prediksi probabilitas variabel yang terikat pada suatu kategori. Berikut persamaan dari *logistik regression* [7].

$$p(y = k|xi) = \frac{exp(w_k^T xi)}{\sum_{j=1}^{K} exp(W_k^T xi)} \tag{2.1}$$

Keterangan:

1. $p(y = k|xi)$ adalah probabilitas titik berdimensi-D $xi \in \{x1, x2, ..., xN\}$ yang berkaitan dengan kelas $k \in \{1, 2, ..., K\}$.
2. xi adalah representasi berdimensi-D dari vektor fitur.
3. K adalah jumlah label kelas.
4. $W = \{w1, w2, ..., wK\}$ adalah vektor parameter untuk semua kelas k.

Kelebihan pada *Logistic regression* adalah tidak memerlukan daya komputasi yang tinggi, sehingga tidak memerlukan perangkat dengan spesifikasi sistem yang tinggi untuk menggunakannya. Selain itu, *logistic regression* menyediakan nilai probabilitas untuk dapat melakukan observasi lebih lanjut [15]. Pada penggunaan model algoritma *logistic regression* dari scikit-learn, terdapat beberapa parameter yang dapat digunakan, beberapanya adalah *penalty* dan *solver*. *Penalty* adalah parameter yang digunakan untuk menentukan jenis regularisasi yang akan digunakan pada model *logistic regression*. Regularisasi sendiri merupakan teknik yang digunakan untuk mencegah *overfitting* atau perilaku yang tidak diinginkan dalam melakukan pembelajaran mesin. Regularisasi yang tersedia pada model ini terdapat L1, L2, dan *Elastic-net*. Pada penelitian ini, regularisasi L1 akan digunakan. Regularisasi L1 digunakan untuk menambahkan penalti terhadap besarnya koefisien dalam model [16]. Sementara untuk *solver* sendiri, merupakan algoritma yang digunakan untuk mengoptimasi masalah. Pada penelitian ini, *solver* SAGA akan digunakan. *Solver* SAGA sendiri cocok digunakan untuk penggunaan *dataset* dalam jumlah besar.

2.4 Data Cleaning

Data Cleaning merupakan salah satu proses dalam pembangunan model sebelum digunakan untuk melakukan klasifikasi teks. Proses ini dilakukan untuk meningkatkan kualitas dari *dataset* yang akan digunakan dengan melewati beberapa langkah [7]. Langkah-langkah yang terdapat pada proses ini, diantaranya

1. *Case Folding* atau *Lowercase*
   Proses *case folding* atau *lowercase* adalah proses di mana kata dari setiap teks diubah menjadi huruf kecil. Hal ini ditujukan agar mengurangi perbedaan makna dari teks yang disebabkan oleh kata yang memiliki huruf kapital dan

kata yang tidak memiliki huruf kapital pada proses pembangunan model selanjutnya. Kata yang memiliki huruf kapital dan kata yang tidak memiliki huruf kapital dianggap dua kata yang berbeda oleh mesin [17]. Contoh dari *case folding* dengan menggunakan judul penelitan milik dosen-dosen UMN yang terpublikasi pada Google Scholar terdapat pada tabel 2.2 contoh data judul penelitian sebelum dan setelah *case folding*.

Tabel 2.2. Contoh Data Judul Penelitian Sebelum dan Setelah *Case Folding*

No	Sebelum *Case Folding*	Setelah *Case Folding*
1	Praktik Implementasi Sensitivitas Anak Dalam Kebijakan Redaksional Laptop Si Unyil Di Trans7	praktik implementasi sensitivitas anak dalam kebijakan redaksional laptop si unyil di trans7
2	Penerapan Pendekatan *Machine Learning* Pada Pengembangan Basis Data Herbal Sebagai Sumber Informasi Kandidat Obat Kanker	penerapan pendekatan *machine learning* pada pengembangan basis data herbal sebagai sumber informasi kandidat obat kanker
3	Rancang Bangun Website *E-Library* pada Tunas Mulia Montessori School	rancang bangun website *e-library* pada tunas mulia montessori school
4	*Java programming language learning application based on octalysis gamification framework*	*java programming language learn application based on octalysis gamification framework*
5	Telaah sistematis terhadap basis data bahan alam untuk pengembangan produk suplemen herbal	telaah sistematis terhadap basis data bahan alam untuk pengembangan produk suplemen herbal

2. *Remove non-alphabet character*

   Proses ini menghapus setiap kata atau bagian dari setiap teks yang tidak mengandung alfabet seperti tanda baca, angka, dan simbol lainnya yang tidak mengandung alfabet. Proses ini dilakukan karena mesin tidak mengerti tanda baca atau simbol-simbol yang terdapat pada *dataset.* [17]. Contoh proses

remove non-alphabet character dapat dilihat pada tabel 2.3 contoh data judul penelitian sebelum dan setelah dihapus *non-alphabet character*

Tabel 2.3. Contoh Data Judul Penelitian Sebelum dan Setelah Dihapus *Non-alphabet Character*

No	**Sebelum *Remove Non-alphabet Character***	**Setelah *Remove Non-alphabet Character***
1	DNA methylation-regulated microRNA pathways in ovarian serous cystadenocarcinoma: A meta-analysis	DNA methylationregulated microRNA pathways in ovarian serous cystadenocarcinoma A metaanalysis
2	*DISDAIN: An Auto Content Generation VR Game*	DISDAIN An Auto Content Generation VR Game
3	Gubernur, Media Sosial, dan Fenomena ”Like and Dislike”	Gubernur Media Sosial dan Fenomena Like and Dislike
4	Characterization Of In0. 3ga0. 7as (N) Quantum Wells In (001) Gaas Using Tem	Characterization Of In ga as N Quantum Wells In Gaas Using Tem
5	E-marketing: Principles, Dynamics & Optimization	Emarketing Principles Dynamics Optimization

3. *Remove stopwords*

   Proses menghapus *stopwords* ini menghapus kata-kata yang tidak memiliki makna penting dalam sebuah kalimat, sehingga hanya berfokus pada kata-kata yang bermakna penting. Kata-kata yang tidak bermakna penting dalam kasus klasifikasi teks tidak memberikan bobot terhadap kata-kata tersebut [17]. Contoh proses *remove stopwords* dapat dilihat pada tabel 2.4 contoh data judul penelitian sebelum dan setelah dihapus *stopword*.

Tabel 2.4. Contoh Data Judul Penelitian Sebelum dan Setelah Dihapus *stopwords*

No	Sebelum *Remove Stopwords*	Setelah *Remove Stopwords*
1	Praktik Implementasi Sensitivitas Anak Dalam Kebijakan Redaksional Laptop Si Unyil Di Trans7	Praktik Implementasi Sensitivitas Anak Kebijakan Redaksional Laptop Si Unyil Trans7
2	Penerapan Pendekatan *Machine Learning* Pada Pengembangan Basis Data Herbal Sebagai Sumber Informasi Kandidat Obat Kanker	Penerapan Pendekatan *Machine Learning* Pengembangan Basis Data Herbal Sumber Informasi Kandidat Obat Kanker
3	Rancang Bangun Website *E-Library* pada Tunas Mulia Montessori School	Rancang Bangun Website *E-Library* Tunas Mulia Montessori School
4	*Java programming language learning application based on octalysis gamification framework*	*Java programming language learning application based octalysis gamification framework*
5	Telaah sistematis terhadap basis data bahan alam untuk pengembangan produk suplemen herbal	Telaah sistematis basis data bahan alam pengembangan produk suplemen herbal

4. *Tokenization*

   Proses *tokenization* ini mengubah atau memecah suatu kalimat menjadi satuan kata. *Tokenization* bertujuan agar setiap kata dari *dataset* dapat berdiri sendiri tanpa berkaitan dengan kata-kata lainnya dan juga untuk dapat melakukan perhitungan bobot setiap kata yang dapat dilakukan dengan menggunakan TF-IDF (*Term Frequency - Inverse Document Frequency*), *Count Vectorizer*, *Word2Vec*, dan metode *preprocessing* lainnya. Proses ini juga membantu dalam menyaring kata-kata yang tidak diinginkan dalam proses pembangunan model selanjutnya [17]. Contoh proses *tokenization* dapat dilihat pada tabel 2.5 contoh data judul penelitian sebelum dan setelah *tokenization*.

Tabel 2.5. Contoh Data Judul Penelitian Sebelum dan Setelah *Tokenization*

No	Sebelum *Tokenization*	Setelah *Tokenization*
1	Praktik Implementasi Sensitivitas Anak Dalam Kebijakan Redaksional Laptop Si Unyil Di Trans7	['Praktik', 'Implementasi', 'Sensitivitas', 'Anak', 'Dalam', 'Kebijakan', 'Redaksional', 'Laptop', 'Si', 'Unyil', 'Di', 'Trans7']
2	Penerapan Pendekatan *Machine Learning* Pada Pengembangan Basis Data Herbal Sebagai Sumber Informasi Kandidat Obat Kanker	['Penerapan', 'Pendekatan', 'Machine', 'Learning', 'Pada', 'Pengembangan', 'Basis', 'Data', 'Herbal', 'Sebagai', 'Sumber', 'Informasi', 'Kandidat', 'Obat', 'Kanker', '&', 'penerapan', 'pendekatan']
3	Rancang Bangun Website *E-Library* pada Tunas Mulia Montessori School	['rancang', 'bangun', 'website', 'e-library', 'pada', 'tunas', 'mulia', 'montessori', 'school']
4	*Java programming language learning application based on octalysis gamification framework*	['Java', 'programming', 'language', 'learning', 'application', 'based', 'on', 'octalysis', 'gamification', 'framework']
5	Telaah sistematis terhadap basis data bahan alam untuk pengembangan produk suplemen herbal	['Telaah', 'sistematis', 'terhadap', 'basis', 'data', 'bahan', 'alam', 'untuk', 'pengembangan', 'produk', 'suplemen', 'herbal']

5. *Stemming*

   Proses ini melakukan mengurangi kata yang memiliki imbuhan pada awal atau akhir kata menjadi kata dasarnya yang tidak memedulikan konteks dari sebuah kalimat [6]. Imbuhan awal atau akhir kata dihapus sementara makna semantik dari semua bentuk yang berbeda akan bermakna sama [17]. Contoh

proses *stemming* dapat dilihat pada tabel 2.6 contoh data judul penelitian sebelum dan setelah *stemming*.

Tabel 2.6. Contoh Data Judul Penelitian Sebelum dan Setelah *Stemming*

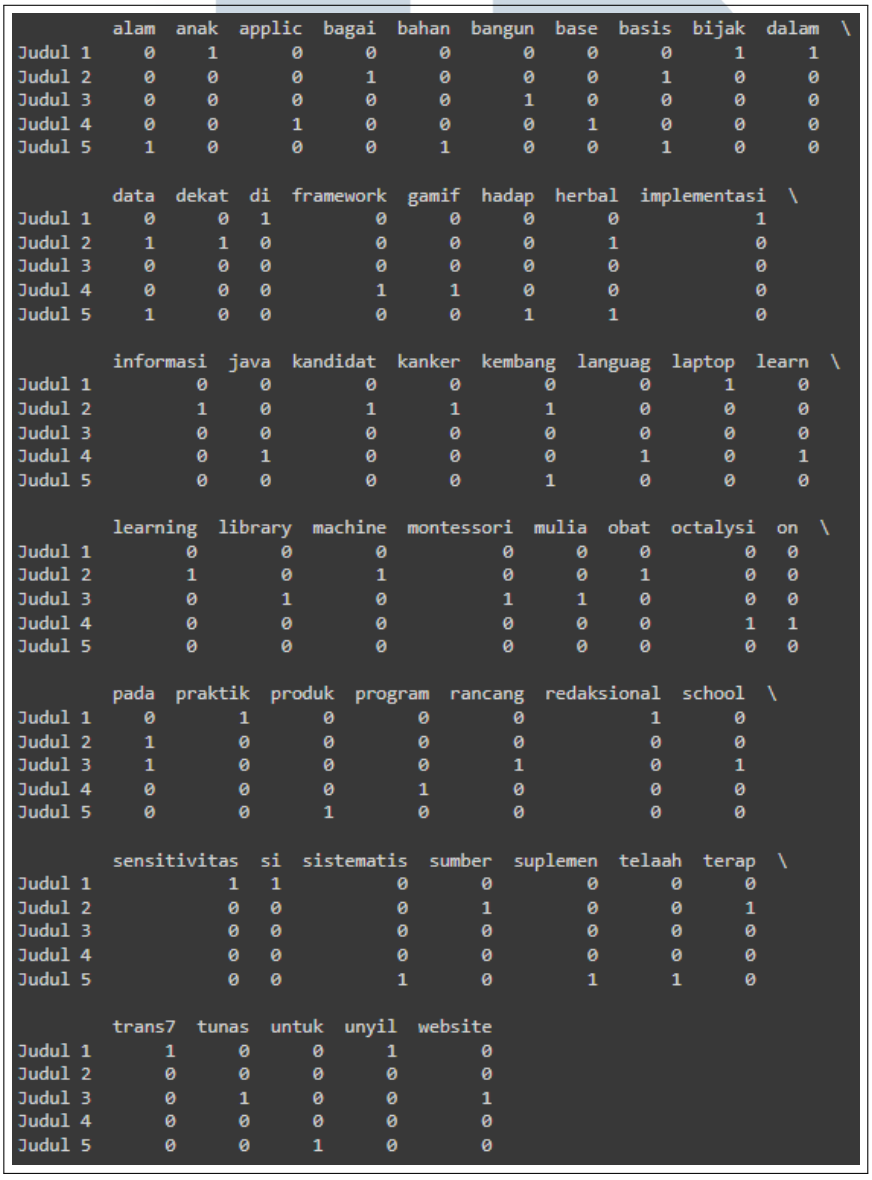
No	Sebelum *Stemming*	Setelah *Stemming*
1	Praktik Implementasi Sensitivitas Anak Dalam Kebijakan Redaksional Laptop Si Unyil Di Trans7	praktik implementasi sensitivitas anak dalam bijak redaksional laptop si unyil di trans7
2	Penerapan Pendekatan *Machine Learning* Pada Pengembangan Basis Data Herbal Sebagai Sumber Informasi Kandidat Obat Kanker	terap dekat machine learning pada kembang basis data herbal bagai sumber informasi kandidat obat kanker
3	Rancang Bangun Website *E-Library* pada Tunas Mulia Montessori School	rancang bangun website e-library pada tunas mulia montessori school
4	*Java programming language learning application based on octalysis gamification framework*	*java program languag learn applic base on octalysi gamif framework*
5	Telaah sistematis terhadap basis data bahan alam untuk pengembangan produk suplemen herbal	telaah sistematis hadap basis data bahan alam untuk kembang produk suplemen herbal

Setelah *dataset* melewati proses ini, maka *dataset* tersebut dapat digunakan pada proses *data preprocessing*.

2.5 Count Vectorizer

Count Vectorizer adalah salah satu metode atau *library* dari scikit-learn dalam melakukan data *preprocessing* yang mengubah teks atau dokumen menjadi sebuah representasi matriks atau vektor. *Count Vectorizer* menghitung representasi atau jumlah setiap kata yang muncul pada suatu dokumen yang digunakan [18].

Angka munculnya setiap kata akan terus bertambah setiap kali suatu kata disebutkan pada suatu *dataset*. Tujuan utama dari *Count Vectorizer* sendiri ialah untuk melakukan penyesuaian dan mempelajari setiap kata yang diberikan dalam kosakata ataupun *dataset* [19].



	alam	anak	applic	bagai	bahan	bangun	base	basis	bijak	dalam
Judul 1	0	1	0	0	0	0	0	0	1	1
Judul 2	0	0	0	1	0	0	0	1	0	0
Judul 3	0	0	0	0	0	1	0	0	0	0
Judul 4	0	0	1	0	0	0	1	0	0	0
Judul 5	1	0	0	0	1	0	0	1	0	0

	data	dekat	di	framework	gamif	hadap	herbal	implementasi
Judul 1	0	0	1	0	0	0	0	1
Judul 2	1	1	0	0	0	0	1	0
Judul 3	0	0	0	0	0	0	0	0
Judul 4	0	0	0	1	1	0	0	0
Judul 5	1	0	0	0	0	1	1	0

	informasi	java	kandidat	kanker	kembang	languag	laptop	learn
Judul 1	0	0	0	0	0	0	1	0
Judul 2	1	0	1	1	1	0	0	0
Judul 3	0	0	0	0	0	0	0	0
Judul 4	0	1	0	0	0	1	0	1
Judul 5	0	0	0	0	1	0	0	0

	learning	library	machine	montessori	mulia	obat	octalysi	on
Judul 1	0	0	0	0	0	0	0	0
Judul 2	1	0	1	0	0	1	0	0
Judul 3	0	1	0	1	1	0	0	0
Judul 4	0	0	0	0	0	0	1	1
Judul 5	0	0	0	0	0	0	0	0

	pada	praktik	produk	program	rancang	redaksional	school
Judul 1	0	1	0	0	0	1	0
Judul 2	1	0	0	0	0	0	0
Judul 3	1	0	0	0	1	0	1
Judul 4	0	0	0	1	0	0	0
Judul 5	0	0	1	0	0	0	0

	sensitivitas	si	sistematis	sumber	suplemen	telaah	terap
Judul 1	1	1	0	0	0	0	0
Judul 2	0	0	0	1	0	0	1
Judul 3	0	0	0	0	0	0	0
Judul 4	0	0	0	0	0	0	0
Judul 5	0	0	1	0	1	1	0

	trans7	tunas	untuk	unyil	website
Judul 1	1	0	0	1	0
Judul 2	0	0	0	0	0
Judul 3	0	1	0	0	1
Judul 4	0	0	0	0	0
Judul 5	0	0	1	0	0

Gambar 2.1. Contoh hasil *Count Vectorizer*

Pada gambar 2.1 contoh hasil *count vectorizer*, menunjukkan hasil dari merubah teks atau dokumen menjadi representasi matriks yang menampilkan jumlah setiap kata yang muncul pada suatu dokumen yang digunakan. Pada gambar ini, contoh dokumen atau kalimat-kalimat yang digunakan adalah menggunakan contoh judul penelitian yang telah melewati proses *stemming* sebelumnya. Berikut

data judul penelitian yang digunakan.

1. praktik implementasi sensitivitas anak dalam bijak redaksional laptop si unyil di trans7
2. terap dekat machine learning pada kembang basis data herbal bagai sumber informasi kandidat obat kanker
3. rancang bangun website e-library pada tunas mulia montessori school
4. java program languag learn applic base on octalysi gamif framework
5. telaah sistematis hadap basis data bahan alam untuk kembang produk suplemen herbal

Pada gambar tersebut, setiap judul akan dihitung kedapatannya setiap kata yang muncul pada kosakata yang ada pada kolom teratas matriks. Angka '0' dan '1' yang terdapat setiap baris pada setiap judul menunjukkan jumlah kata yang muncul dari setiap judul dengan setiap kosakata yang telah dibangun. Angka '0' menunjukkan tidak ada kata yang muncul dari judul dengan setiap kosakata, sedangkan angka '1' atau lebih dari '1' menunjukkan bahwa kata tersebut muncul satu atau lebih kali dari judul tersebut. Salah satu contoh interpretasi dari matriks pada gambar tersebut adalah pada judul satu yaitu "praktik implementasi sensitivitas anak dalam bijak redaksional laptop si unyil di trans7" terdapat satu kata "bijak" pada kolom "bijak", maka dari itu kolom "bijak" pada judul tersebut dinilai dengan angka '1'. Sebaliknya, pada judul satu tidak memilik kata "bagai", maka dari itu kolom "bagai" dinilai dengan angka '0'.

2.6 TF-IDF

Term Frequency-Inverse Document Frequency atau TF-IDF adalah juga salah satu metode atau *library* dalam scikit-learn yang memiliki fungsi yang serupa dengan *Count Vectorizer*, hanya saja TF-IDF menghitung frekuensi pentingnya setiap kata yang muncul pada dataset atau data teks. Metode pada *Term Frequency* bekerja untuk mengetahui frekuensi banyaknya setiap kata yang muncul pada suatu *dataset*, sementara sebaliknya *Inverse Document Frequency* menghapus kata-kata yang menambah sedikit makna atau arti pada suatu kalimat. Pada intinya, TF-IDF mengisolasi kata-kata yang jarang muncul selagi melakukan ektraksi pada fitur yang relevan dari *dataset* [5]. Berikut persamaan dari TF-IDF.

$$w_{jk} = tf_{jk} * idf_j \quad (2.2)$$

Keterangan:

1. w_{jk} adalah bobot kata dari kata j dalam suatu dokumen atau *dataset* k .
2. tf_{jk} adalah banyaknya kata j yang muncul pada dokumen atau *dataset* k. Rumus tf_{jk} didapatkan dari $tf_{jk} = \frac{j}{k}$
3. idf_j adalah banyaknya dokumen yang mengandung kata j dari suatu dokumen maupuan *dataset*. Rumus idf_j didapatkan dari $idf_j = log_2(\frac{jumlah\ dokumen\ atau\ n}{jumlah\ dokumen\ yang\ terdapat\ kata\ j\ atau\ df_j})$ [7].

Contoh sederhana dari TF-IDF adalah jika ingin melakukan vektorisasi menggunakan TF-IDF dari dokumen atau kalimat-kalimat yang digunakan adalah menggunakan contoh judul penelitian yang telah melewati proses *stemming* sebelumnya. Berikut data judul penelitian yang digunakan.

1. praktik implementasi sensitivitas anak dalam bijak redaksional laptop si unyil di trans7
2. terap dekat machine learning pada kembang basis data herbal bagai sumber informasi kandidat obat kanker
3. rancang bangun website e-library pada tunas mulia montessori school
4. java program languag learn applic base on octalysi gamif framework
5. telaah sistematis hadap basis data bahan alam untuk kembang produk suplemen herbal

Pada contoh ini, jika ingin mencari sebuah bobot dari salah satu kata yaitu ”kembang” dalam data pada judul penelitian kedua yang telah ditentukan sebelumnya. Berikut langkah-langkah yang akan dilakukan

1. Pertama, menghitung terlebih dahulu frekuensi atau banyaknya kata ”kembang” dari judul kedua tersebut. Kata ”kembang” muncul satu kali dalam judul tersebut. Setelah itu, dihitung jumlah kata dari judul tersebut dan dikurangi sebanyak kemunculan kata yang ingin dicari. Maka jumlah kata dari judul tersebut adalah 11. Setelah frekuensi kata dan jumlah kata

pada judul kedua telah diketahui, variabel tf dapat dihitung yaitu dengan cara

$$tf_{jk} = \frac{j}{k}$$
$$tf_{jk} = \frac{1}{11}$$
$$tf_{jk} = 0.090909$$

2. Kedua, setelah tf diketahui maka dicarilah banyaknya judul penelitian yang mengandung kata ”kembang” dari judul data judul-judul penelitian yang telah ditentukan sebelumnya. Data judul penelitian yang ditentukan sebelumnya memiliki 5 judul secara keseluruhan dan jumlah judul yang memiliki kata ”kembang” terdapat 2 judul. Maka untuk menghitung variabel idf untuk mengetahui banyaknya judul penelitian yang mengandung kata ”kembang” dapat dilakukan sebagai berikut.

$$idf_j = log(\frac{n}{df(j)})$$
$$idf_j = log(\frac{5}{2})$$
$$idf_j = 0.39794$$

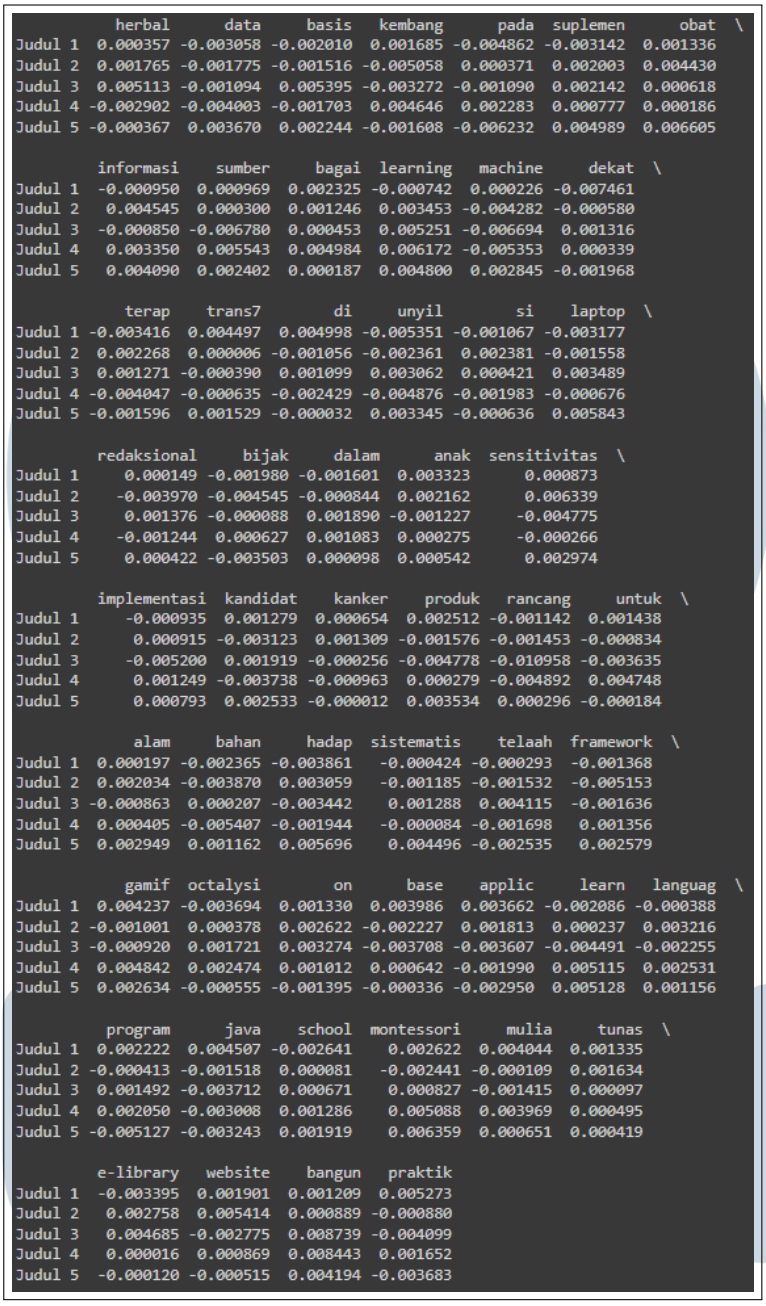
3. Setelah tf_{jk} dan idf_j diketahui, maka dapat dilakukan pencarian bobot kata dari kata ”kembang” dari kelima data judul penelitian dengan cara seperti berikut.

$$w_{jk} = tf_{jk} * idf_j$$
$$w_{jk} = 0.090909 * 0.39794$$
$$w_{jk} = 0.03617632746$$

Dari hasil perhitungan tersebut, diketahuilah maka bobot kata ”kembang” dari kelima judul penelitian yang telah ditentukan sebelumnya adalah sebesar 0.03617632746. Perhitungan yang dilakukan sebelumnya akan diulangi lagi untuk setiap kata pada masing-masing data judul penelitan.

2.7 Word2Vec

Word2Vec adalah suatu metode atau *library* dalam scikit-learn ini yang menempatkan kata-kata dalam makna serupa berdekatan dengan satu sama lain dengan ruang vektor menggunakan konteks pada setiap istilah dalam *dataset* atau dokumen untuk menghasilkan representasi vektor. *Word2Vec* bertujuan untuk memaksimalkan probabilitas kemunculan sebuah kata berdasarkan konteksnya. *Word2Vec* menggunakan *dataset* atau korpus teks yang besar untuk dapat menghasilkan model ruang vektor. *Word2Vec* memiliki dua macam penggunaan, yaitu *continuous bag-of-words* (CBOW) atau *continuous skip-gram*. *Continuous bag-of-words* (CBOW) melakukan prediksi dari kata yang sedang digunakan berdasarkan konteks pada sekitar kata, Sedangkan *continuous skip-gram*, menggunakan kata yang sedang digunakan untuk memprediksi konteks pada sekitaran kata-kata tersebut [7].



	herbal	data	basis	kembang	pada	suplemen	obat
Judul 1	0.000357	-0.003058	-0.002010	0.001685	-0.004862	-0.003142	0.001336
Judul 2	0.001765	-0.001775	-0.001516	-0.005058	0.000371	0.002003	0.004430
Judul 3	0.005113	-0.001094	0.005395	-0.003272	-0.001090	0.002142	0.000618
Judul 4	-0.002902	-0.004003	-0.001703	0.004646	0.002283	0.000777	0.000186
Judul 5	-0.000367	0.003670	0.002244	-0.001608	-0.006232	0.004989	0.006605

	informasi	sumber	bagai	learning	machine	dekat
Judul 1	-0.000950	0.000969	0.002325	-0.000742	0.000226	-0.007461
Judul 2	0.004545	0.000300	0.001246	0.003453	-0.004282	-0.000580
Judul 3	-0.000850	-0.006780	0.000453	0.005251	-0.006694	0.001316
Judul 4	0.003350	0.005543	0.004984	0.006172	-0.005353	0.000339
Judul 5	0.004090	0.002402	0.000187	0.004800	0.002845	-0.001968

	terap	trans7	di	unyil	si	laptop
Judul 1	-0.003416	0.004497	0.004998	-0.005351	-0.001067	-0.003177
Judul 2	0.002268	0.000006	-0.001056	-0.002361	0.002381	-0.001558
Judul 3	0.001271	-0.000390	0.001099	0.003062	0.000421	0.003489
Judul 4	-0.004047	-0.000635	-0.002429	-0.004876	-0.001983	-0.000676
Judul 5	-0.001596	0.001529	-0.000032	0.003345	-0.000636	0.005843

	redaksional	bijak	dalam	anak	sensitivitas
Judul 1	0.000149	-0.001980	-0.001601	0.003323	0.000873
Judul 2	-0.003970	-0.004545	-0.000844	0.002162	0.006339
Judul 3	0.001376	-0.000088	0.001890	-0.001227	-0.004775
Judul 4	-0.001244	0.000627	0.001083	0.000275	-0.000266
Judul 5	0.000422	-0.003503	0.000098	0.000542	0.002974

	implementasi	kandidat	kanker	produk	rancang	untuk
Judul 1	-0.000935	0.001279	0.000654	0.002512	-0.001142	0.001438
Judul 2	0.000915	-0.003123	0.001309	-0.001576	-0.001453	-0.000834
Judul 3	-0.005200	0.001919	-0.000256	-0.004778	-0.010958	-0.003635
Judul 4	0.001249	-0.003738	-0.000963	0.000279	-0.004892	0.004748
Judul 5	0.000793	0.002533	-0.000012	0.003534	0.000296	-0.000184

	alam	bahan	hadap	sistematis	telaah	framework
Judul 1	0.000197	-0.002365	-0.003861	-0.000424	-0.000293	-0.001368
Judul 2	0.002034	-0.003870	0.003059	-0.001185	-0.001532	-0.005153
Judul 3	-0.000863	0.000207	-0.003442	0.001288	0.004115	-0.001636
Judul 4	0.000405	-0.005407	-0.001944	-0.000084	-0.001698	0.001356
Judul 5	0.002949	0.001162	0.005696	0.004496	-0.002535	0.002579

	gamif	octalysi	on	base	applic	learn	languag
Judul 1	0.004237	-0.003694	0.001330	0.003986	0.003662	-0.002086	-0.000388
Judul 2	-0.001001	0.000378	0.002622	-0.002227	0.001813	0.000237	0.003216
Judul 3	-0.000920	0.001721	0.003274	-0.003708	-0.003607	-0.004491	-0.002255
Judul 4	0.004842	0.002474	0.001012	0.000642	-0.001990	0.005115	0.002531
Judul 5	0.002634	-0.000555	-0.001395	-0.000336	-0.002950	0.005128	0.001156

	program	java	school	montessori	mulia	tunas
Judul 1	0.002222	0.004507	-0.002641	0.002622	0.004044	0.001335
Judul 2	-0.000413	-0.001518	0.000081	-0.002441	-0.000109	0.001634
Judul 3	0.001492	-0.003712	0.000671	0.000827	-0.001415	0.000097
Judul 4	0.002050	-0.003008	0.001286	0.005088	0.003969	0.000495
Judul 5	-0.005127	-0.003243	0.001919	0.006359	0.000651	0.000419

	e-library	website	bangun	praktik
Judul 1	-0.003395	0.001901	0.001209	0.005273
Judul 2	0.002758	0.005414	0.000889	-0.000880
Judul 3	0.004685	-0.002775	0.008739	-0.004099
Judul 4	0.000016	0.000869	0.008443	0.001652
Judul 5	-0.000120	-0.000515	0.004194	-0.003683

Gambar 2.2. Contoh hasil *Word2Vec*

Pada gambar 2.2 contoh hasil *word2vec*, menunjukkan hasil dari penggunaan dari word2vec itu sendiri pada suatu dokumen yang digunakan. Pada gambar ini, contoh dokumen atau kalimat-kalimat yang digunakan adalah menggunakan contoh judul penelitian yang telah melewati proses *stemming* sebelumnya. Berikut data judul penelitian yang digunakan.

1. praktik implementasi sensitivitas anak dalam bijak redaksional laptop si unyil

di trans7

2. terap dekat machine learning pada kembang basis data herbal bagai sumber informasi kandidat obat kanker
3. rancang bangun website e-library pada tunas mulia montessori school
4. java program languag learn applic base on octalysi gamif framework
5. telaah sistematis hadap basis data bahan alam untuk kembang produk suplemen herbal

Pada gambar tersebut, setiap judul akan dihitung vektor kata yang mewakili hubungan dari satu kata ke kata lainnya. Nilai positif dan negatif dalam setiap baris judul menunjukkan bobot terhadap kata-kata lain dalam konteks yang dianalisis.

2.8 Sustainable Development Goals (SDG)

Sustainable Development Goals (SDG) yang juga merupakan tujuan global, diadaptasi oleh Perserikatan Bangsa-Bangsa (PBB) pada tahun 2015 dengan tujuan sebagai panggilan kepada seluruh umat manusia untuk bertindak dalam mengakhiri kemisikinan, melindungi bumi, dan memastikan bahwa pada tahun 2030 mendatang semua orang menikmati perdamaian dan kesejateraan [2]. SDG dari PBB juga bertujuan untuk dapat menginspirasi secara operasionalisasi dan intregasi SDG dari PBB ke dalam organisasi di seluruh dunia, dapat memenuhi kebutuhan pemangku kepentingan atau *stakeholder* pada saat ini dan di waktu yang akan mendatang, dan berkontribusi pada pencapaian dalam pembangunan keberlanjutan bagi masyarakat luas [4]. SDG dari PBB sendiri memiliki 17 tujuan, yaitu sebagai berikut [3].

1. Tanpa Kemiskinan
   Mengakhiri kemiskinan dalam segala bentuk di mana pun.
2. Tanpa Kelaparan
   Mengakhiri kelaparan, mencapai ketahanan pangan dan gizi yang baik, serta meningkatkan pertanian keberlanjutan.
3. Kesehatan yang Baik dan Kesejahteraan
   Menjamin kehidupan sehat dan meningkatkan kesejahteraan untuk seluruh penduduk tanpa memandang usia.

4. Pendidikan Berkualitas
   Menjamin pendidikan yang inklusif, berkualitas dan meningkatkan kesempatan belajar untuk semua sepanjang hayat.

5. Kesetaraan Gender
   Mencapai kesetaraan gender dan memberdayakan perempuan.

6. Air Bersih dan Sanitasi
   Menjamin akses air dan sanitasi yang berkelanjutan untuk semua.

7. Energi Bersih dan Terjangkau
   Menjamin akses energi yang terjangkau, andal, dan berkelanjutan untuk semua.

8. Pekerjaan Layak dan Pertumbuhan Ekonomi
   Meningkatkan pertumbuhan ekonomi yang inklusif dan berkelanjutan serta kesempatan kerja yang produkti dan menyeluruh untuk semua.

9. Industri, Inovasi, dan Infrastruktur
   Membangun infrastruktur yang tangguh dan berkelanjutan serta mendorong inovasi.

10. Mengurangi Ketimpangan
    Mengurangi ketimpangan di dalam dan antar negara.

11. Kota dan Komunitas Berkelanjutan
    Mewujudkan kota dan pemukiman inklusif, aman, dan berkelanjutan.

12. Konsumsi dan Produksi yang Bertanggung Jawab
    Menjamin pola konsumsi dan produksi yang berkelanjutan.

13. Penanganan Perubahan Iklim
    Mengambil tindakan segera dalam mengatasi masalah perubahan iklim dan dampaknya.

14. Ekosistem Lautan
    Melestarikan dan menggunakan sumber daya dari lautan dan samudera untuk pembangunan secara berkelanjutan.

15. Ekosistem Daratan
    Melindungi, memulihkan, dan mendukung penggunaan berkelanjutan ekosistem darat.

16. Perdamaian, Keadilan, dan Kelembagaan yang Tangguh
    Mendorong masyarakat yang damai dan inklusif serta menyediakan akses keadilan untuk semua tingkatan.

17. Kemitraan untuk Mencapai Tujuan
    Memperkuat cara-cara pelaksanaan dan merevitalisasi kemitraan global untuk pembangunan keberlanjutan.

PBB bekerja sama dengan pemerintah sebagai penanggung jawab dalam implementasi SDG dari PBB, memastikannya ada tindak lanjut dan meninjau selama 15 tahun kedepan pada tingkat nasional, regional, dan global. PBB juga membuka peluang untuk pemangku kepentingan, seperti pemerintah, bisnis, akademisi, masyarakat sipil, dan komunitas lokal untuk dapat ikut serta dalam implementasi SDG dari PBB ini seperti pada tujuan ke 17 pada SDG yaitu kemitraan untuk mencapai tujuan [20].