Methotika : Jurnal Ilmiah Teknik Informatika
E-ISSN : 2776 – 5792
Vol. 6, No. 1, April 2026, pp. 1-11
https://ejurnal.methodist.ac.id/index.php/methotika/

Analisa Metode Cosine Similarity Dalam Mengenali Pengirim Pesan Singkat

Widia Margaretha Sidabutar[1], Harlen Gilbert Simanullang[2], Arina Prima Silalahi[3]
[1,2,3]Fakultas Ilmu Komputer, Universitas Methodist Indonesia

Info Artikel

Histori Artikel:

Received, July 23, 2025
Revised, Sept 20, 2025
Accepted, Nov 11, 2025

Keywords:

Cosine Similarity, Analisis Kemiripan, *Text Mining*, Pengenalan Pengirim Pesan, Pesan singkat.

ABSTRAK

Perkembangan teknologi komunikasi digital yang pesat telah menjadikan aplikasi pesan singkat, seperti *WhatsApp*, sebagai sarana utama dalam berkomunikasi. Namun, penggunaan pesan singkat juga menghadirkan tantangan, terutama dalam mengenali pengirim pesan ketika informasi pengirim tidak tercantum secara jelas. Hal ini dapat membuka peluang bagi penyalahgunaan, seperti penyamaran atau penipuan melalui pesan yang tidak terdeteksi. Oleh karena itu, diperlukan metode yang efektif untuk menganalisis pola penulisan dalam pesan guna membantu identifikasi pengirim. Penelitian ini menggunakan metode *Cosine Similarity*, yang mampu mengukur kemiripan antara teks berdasarkan representasi vektornya. Proses analisis dilakukan dengan menerapkan teknik *Text Mining*, meliputi *case folding*, *cleaning*, *tokenizing*, *stopword removal* dan *stemming* untuk memastikan data yang digunakan lebih bersih dan akurat. Dengan metode ini, kemiripan antara pesan uji dan pesan pembanding dihitung guna menentukan pesan yang paling mirip dengan data uji. Hasil penelitian menunjukkan bahwa D101 memiliki kemiripan tertinggi dengan D9 sebesar 71%, D102 dengan D4 sebesar 95%, dan D103 dengan D15 sebesar 77%. Nilai-nilai tersebut menunjukkan bahwa metode *Cosine Similarity* dapat digunakan untuk mengenali pola penulisan dalam pesan singkat dengan tingkat akurasi yang baik. Penelitian ini diharapkan dapat menjadi dasar bagi pengembangan dalam bentuk web atau aplikasi untuk mengidentifikasi pengirim pesan singkat, sehingga meningkatkan keamanan dan kenyamanan dalam komunikasi digital.

Penulis Koresponden:

Widia Margaretha Sidabutar,
Fakultas Ilmu Komputer,
Universitas Methodist Indonesia, Medan,
Jl. Hang Tuah No.8, Medan - Sumatera Utara.
Email: widiasidabutar670@gmail.com

1. PENDAHULUAN

Teknologi komunikasi dan informasi masa kini yang semakin canggih sangat memudahkan berkomunikasi jarak jauh dan memiliki berbagai macam manfaat pada kehidupan sehari-hari[1]. Salah satu inovasi utama dalam bidang ini adalah aplikasi pesan singkat, seperti *WhatsApp*, yang kini

telah menjadi salah satu sarana utama bagi individu untuk berkomunikasi, baik untuk percakapan pribadi, profesional, maupun dalam kelompok. Aplikasi ini menawarkan kemudahan dan kecepatan dalam berinteraksi, yang membuatnya menjadi alat komunikasi yang sangat populer di seluruh dunia. Setiap hari, jutaan pesan dikirimkan melalui *WhatsApp*, memberikan dampak signifikan dalam cara orang berinteraksi satu sama lain.

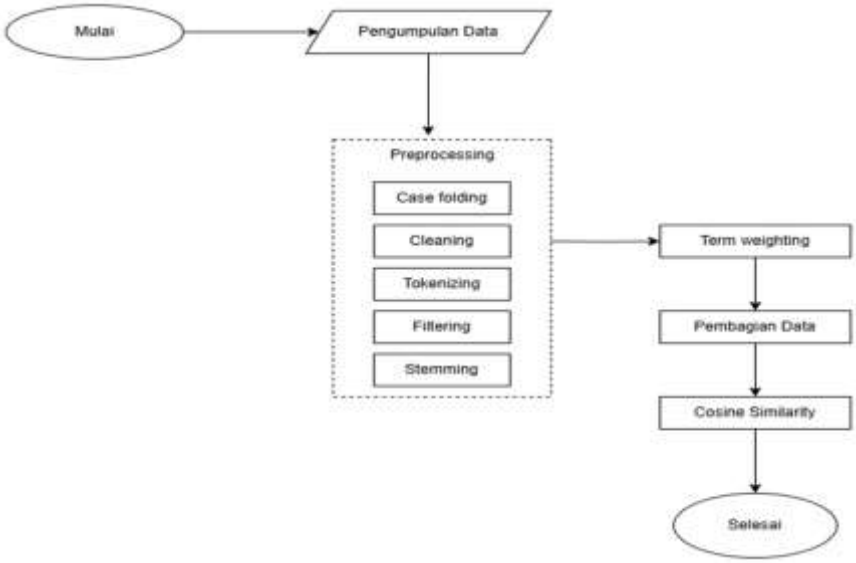
Namun, meskipun penggunaan aplikasi pesan singkat terus meningkat, muncul sejumlah permasalahan yang berkaitan dengan keamanan dan kenyamanan penggunanya. Salah satu masalah utama yang sering dihadapi oleh pengguna adalah kesulitan dalam mengidentifikasi pengirim pesan, terutama ketika pesan datang dari nomor yang tidak dikenal atau tidak mencantumkan nama pengirim dengan jelas. Situasi ini sering kali menimbulkan kebingungan dan bahkan dapat membuka peluang bagi tindakan penyalahgunaan, seperti penyamaran atau penipuan melalui pesan yang tidak terdeteksi. Hal ini menjadi sangat relevan mengingat banyaknya kasus penipuan yang terjadi melalui pesan singkat, terutama dengan memanfaatkan nomor yang tidak dikenal oleh penerima pesan.

Untuk mengatasi permasalahan tersebut, diperlukan upaya dalam mengenali pengirim pesan dengan lebih akurat. Salah satunya adalah dengan menganalisis pola penulisan dalam pesan singkat. Setiap individu memiliki gaya bahasa atau pola penulisan yang cenderung konsisten, yang menjadi ciri khas mereka dalam berkomunikasi. Salah satu metode yang telah terbukti efektif dalam mendeteksi kemiripan teks adalah metode cosine similarity[2].Metode *Cosine Similarity* dapat menjadi alat yang efektif untuk menganalisis kemiripan pesan dan mengenali pengirimnya. Metode ini mengukur sejauh mana dua vektor teks saling mendekati dan sering diterapkan dalam analisis teks dalam konteks bahasa alami. Implementasinya melibatkan teknik *Text Mining*, seperti *tokenization*, *case folding*, *stopword removal*, *cleaning*, dan *stemming*. Penelitian ini bertujuan mengidentifikasi pola penulisan dalam pesan *WhatsApp* untuk mengenali pengirim pesan tanpa pengenal yang jelas, sehingga membantu pengguna menghindari potensi penipuan atau kejahatan yang mungkin terjadi melalui pesan tidak sah.

2. METODE PENELITIAN

2.1. *Framework* Penelitian

Framework Penelitian merupakan panduan sistematis yang menggambarkan langkah-langkah yang akan ditempuh oleh peneliti dalam melaksanakan studinya. Kerangka ini juga mencakup serangkaian prosedur yang digunakan untuk merumuskan solusi atas permasalahan yang diangkat dalam penelitian. Pada studi ini, alur kegiatan yang dilakukan oleh peneliti disajikan pada Gambar 1.



Gambar 1. Framework Penelitian

Penelitian ini dimulai dengan mengumpulkan data pesan singkat dari *WhatsApp*, berasal dari percakapan mahasiswa Prodi Teknik Informatika dan Sistem Informasi Universitas Methodist

Indonesia angkatan 2021. Data diproses melalui tahap preprocessing: *case folding, cleaning, tokenizing, filtering*, dan *stemming,* untuk membersihkan elemen tidak relevan dan menyiapkan data untuk analisis. Setelah itu, dilakukan pembobotan kata dengan metode *Term Frequency (TF)* untuk menghitung frekuensi kemunculan kata. Data yang telah diproses dianalisis menggunakan metode *Cosine Similarity* untuk mengukur tingkat kemiripan antar pesan, menghasilkan nilai yang menunjukkan kesamaan berdasarkan pola penulisan dan isi teks. Hasil ini diharapkan membantu identifikasi pengirim pesan melalui kesamaan pola penulisan.

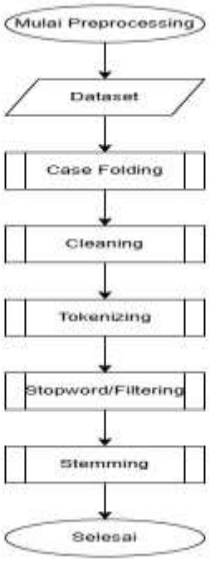
2.2. Teknik Pengumpulan Data

Data yang diolah pada penelitian ini merupakan pesan singkat dari mahasiswa/i Prodi Teknik Informatika dan Sistem Informasi Fakultas Ilmu Komputer Universitas Methodist Indonesia. Data diperoleh dari percakapan pada aplikasi *WhatsApp* yang dilakukan oleh mahasiswa angkatan 2021.

2.3. *Preprocessing Text*

Dataset yang telah terkumpul sebelumnya merupakan data yang belum terstruktur dengan baik, sehingga diperlukan tahapan penting pada bidang *text mining* yang disebut dengan istilah *preproccesing text* dimana bermaksud mengolah data awal yang masih belum terstruktur dengan baik untuk dijadikan sebuah data yang terstruktur agar dapat dikenali dan diterapkan kedalam beberapa metode *text mining* yang ada[3], [4]. Tujuan dari preprocessing teks adalah untuk meningkatkan kualitas data teks, menghilangkan noise, dan memastikan bahwa teks siap digunakan dalam berbagai aplikasi analisis teks, termasuk analisis sentimen, klasifikasi teks, ekstraksi informasi, dan lain-lain[5]. Adapun tahapan *preprocessing* sebagai berikut:[6]

1. *Case folding*, Pada tahap *case folding*, setiap huruf diubah menjadi huruf kecil, karena model yang dibangun bersifat *case sensitif*, sedangkan penggunaan huruf kapital bisa jadi tidak konsisten pada semua kata.
2. Data *Cleaning*, Pada tahap data *cleaning,* setiap kalimat yang mengandung noise yang berupa tautan situs, simbol !,@,#,$,%,^,&,*,(,),/,”,[,. dan lainnya atau angka akan dihapus.
3. *Filtering* dilakukan untuk menghapus kata sambung seperti “dan”, “atau”, “yang” atau kata-kata yang kurang berpengaruh pada penelitian.
4. *Tokenizing,* Tahapan *tokenizing* dilakukan untuk memecah kalimat berdasarkan spasi. menjadi *Tokenizing* kata akan digunakan untuk tahap selanjutnya yaitu *filtering* dan *stemming*.
5. *Stemming* digunakan untuk mengubah seluruh kata menjadi kata dasarnya, salah satunya adalah menghapus imbuhan. *Flowchart* dari tahapan *preprocessing* ini dapat dilihat pada Gambar 2.



Gambar 2. *Flowchart Preprocessing*

2.4. *Term Frequency*

Term frequency (TF) adalah Mengukur seberapa sering suatu kata muncul dalam sebuah dokumen[7]–[9]. Konsep ini digunakan untuk menentukan seberapa sering sebuah kata muncul dalam teks, yang kemudian dijadikan dasar dalam perhitungan bobot kata dalam dokumen. perhitungan bobot term sebagai berikut:[10]

q =(tf) * (idf)

Keterangan:

q : Nilai bobot *term*
tf : Nilai *term frequency*
idf : Nilai *inverse document frequency*

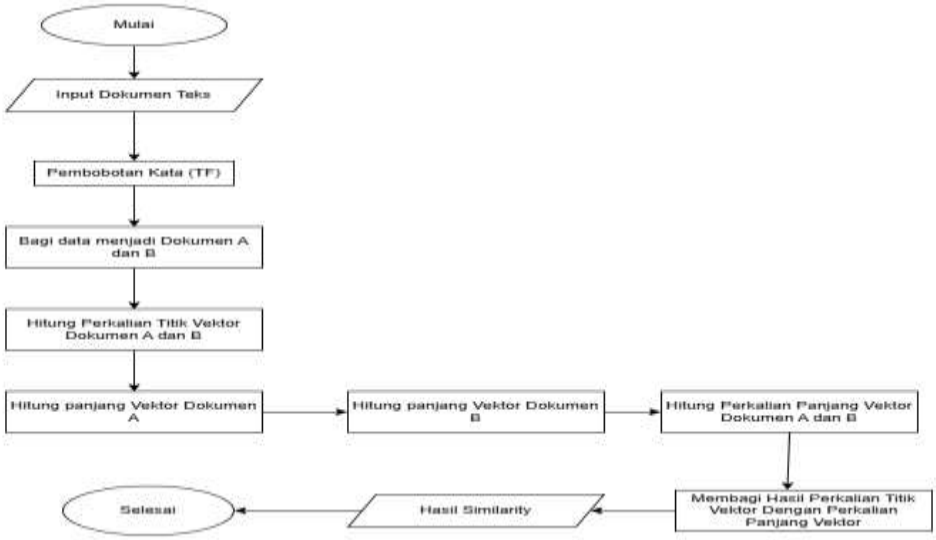
2.4. Tahapan *Cosine Similarity*

Cosine similarity adalah sebuah algoritma yang dapat digunakan sebagai metode dalam menghitung tingkat kesamaan (*similarity*) antar dokumen[11]. Pada dasarnya proses penghitungannya menggunakan ukuran kesamaan ruang vektor (*vector space similarity measure*). Algoritma *cosine similarity* ini menggunakan kata kunci pada sebuah dokumen sebagai ukuran untuk menghitung tingkat kesamaan antar dokumen yang dinyatakan dalam bentuk vektor[12]. Rumus *cosine similarity* adalah sebagai berikut:

$$\textit{Similarity}\ (A, B) = \cos(\theta) = \frac{A.B}{||A||\,||B||} \quad \frac{\sum_{i=1}^{n} A_i\, B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\cdot\sqrt{\sum_{i=1}^{n} B_i^2}}.$$

Keterangan:

Similarity (A, B) : Mengukur kemiripan antara vektor A dan B.
Cos (θ) : Nilai cosine dari sudut θ antara dua vektor.
A · B : Perkalian titik antara vektor A dan B.
|A| : Panjang (magnitudo) vektor A.
|B| : Panjang (magnitudo) vektor B.
|A||B| : Perkalian panjang vektor A dan B untuk normalisasi.
i : Indeks elemen vektor.
n : Jumlah elemen vektor.



Gambar 3. Alur Metode Cosine Similarity

Perhitungan *cosine similarity* dilakukan setelah tahap *preprocessing* dan *pembobotan Term Frequency (TF)* selesai pada setiap teks pesan singkat. Metode cosine similarity digunakan dalam penelitian ini untuk menentukan tingkat kemiripan antara dua pesan singkat. Untuk proses alur menggunakan metode *cosine similarity* dapat dilihat seperti Gambar 3.

3. HASIL DAN PEMBAHASAN

3.1. *Preprocessing* Data

Dalam data pesan singkat, teks telah dinormalisasi, dibersihkan dari karakter *non-alfabetik* dan simbol yang tidak diperlukan, serta dilakukan penghapusan *stopwords*. Proses ini bertujuan untuk menyederhanakan teks sehingga lebih seragam dan bersih, sehingga siap untuk tahap pembobotan kata serta analisis lebih lanjut menggunakan metode *Cosine Similarity*.

3.1.1 *Case Folding*

Pada tahap *case folding*, seluruh teks dalam pesan singkat diubah menjadi huruf kecil untuk menghilangkan perbedaan akibat huruf kapital. Proses ini membantu standarisasi data agar lebih seragam dan siap untuk pemrosesan selanjutnya. Rincian hasil *case folding* terdapat pada Tabel 1.

Tabel 1. Hasil *Case Folding*

Data Pesan	*Case Folding*
Template buat penilaian Gak ada kan di kasih kampus? Maksudku kan Wid, ini template yang kita kasih ke orang ini kan untuk kasih nilai ke kita. Biar Gak capek kali aku, karna udah pasti kotor kali rumahku. Kemarin ku bilang iya, tapi udah capek kali aku Wid, Gak sanggup lagi ini. Kalian kemarin bimbingan itu laporannya udah Kalian print langsung? Itu Kalian foto dulu berartibaru Kalian masukkan ke Wordnya? Tunggu di tanda tangan Bu Margaretha dulu baru Kalian masukkan ke lampiran? Laporan Kalian itu klo dah di ACC, Kalian kumpul kemana Wid? Eh Kaliana kemarin jilid di percetakan apa, Wid? Itu Wid, tanda tangan dekan itu kan ada?	template buat penilaian gak ada kan di kasih kampus? maksudku kan wid, ini template yang kita kasih ke orang ini kan untuk kasih nilai ke kita. biar gak capek kali aku, karna udah pasti kotor kali rumahku. kemarin ku bilang iya, tapi udah capek kali aku wid, gak sanggup lagi ini. kalian kemarin bimbingan itu laporannya udah kalian print langsung? itu kalian foto dulu baru kalian masukkan ke wordnya? tunggu di tanda tangan bu margaretha dulu baru kalian masukkan ke lampiran? laporan kalian itu klo dah di acc, kalian kumpul kemana wid? eh kalian kemarin jilid di percetakan apa, wid? itu wid, tanda tangan dekan itu kan ada?

3.1.2 *Cleaning*

Pada tahap *cleaning,* teks dibersihkan dari karakter yang tidak diperlukan seperti tanda baca, angka, simbol khusus, serta emotikon. Proses ini bertujuan untuk memastikan hanya kata-kata yang relevan yang digunakan dalam analisis lebih lanjut. Rincian hasil *case folding* dapat dilihat pada Tabel 2.

Tabel 2. Hasil *Cleaning*

Case Folding	*Cleaning*
template buat penilaian gak ada kan di kasih kampus? maksudku kan wid, ini template yang kita kasih ke orang ini kan untuk kasih nilai ke kita. biar gak capek kali aku, karna udah pasti kotor kali rumahku. kemarin ku bilang iya, tapi udah capek kali aku wid, gak sanggup lagi ini. kalian kemarin bimbingan itu laporannya udah kalian print langsung? itu kalian foto dulu baru kalian masukkan ke wordnya? tunggu di tanda tangan bu margaretha dulu baru kalian masukkan ke lampiran? laporan kalian itu klo dah di acc, kalian kumpul kemana wid? eh	template buat penilaian gak ada kan di kasih kampus maksudku kan wid ini template yang kita kasih ke orang ini kan untuk kasih nilai ke kita biar gak capek kali aku karna udah pasti kotor kali rumahku kemarin ku bilang iya tapi udah capek kali aku wid gak sanggup lagi ini kalian kemarin bimbingan itu laporannya udah kalian print langsung itu kalian foto dulu baru kalian masukkan ke wordnya tunggu di tanda tangan bu margaretha dulu baru kalian masukkan ke lampiran laporan kalian itu klo dah di acc kalian kumpul kemana wid eh kalian

kalian kemarin jilid di percetakan apa, wid? itu wid, tanda tangan dekan itu kan ada?	kemarin jilid di percetakan apa wid itu wid tanda tangan dekan itu kan ada

3.1.3 *Tokenizing*

Pada tahapan *tokenizing*, dilakukan pemecahan teks ke dalam unit-unit kecil berupa kata untuk memudahkan analisis di tahap berikutnya. Rincian hasil *tokenizing* dapat dilihat pada Tabel 3.

Tabel 3. Hasil *Tokenizing*

Cleaning	*Tokenizing*
template buat penilaian gak ada kan di kasih kampus maksudku kan wid ini template yang kita kasih ke orang ini kan untuk kasih nilai ke kita biar gak capek kali aku karna udah pasti kotor kali rumahku kemarin ku bilang iya tapi udah capek kali aku wid gak sanggup lagi ini kalian kemarin bimbingan itu laporannya udah kalian print langsung itu kalian foto dulu baru kalian masukkan ke wordnya tunggu di tanda tangan bu margaretha dulu baru kalian masukkan ke lampiran laporan kalian itu klo dah di acc kalian kumpul kemana wid eh kalian kemarin jilid di percetakan apa wid itu wid tanda tangan dekan itu kan ada	'template', 'buat', 'penilaian', 'gak', 'ada', 'kan', 'di', 'kasih', 'kampus', 'maksudku', 'kan', 'wid', 'ini', 'template', 'yang', 'kita', 'kasih', 'ke', 'orang', 'ini', 'kan', 'untuk', 'kasih', 'nilai', 'ke', 'kita', 'biar', 'gak', 'capek', 'kali', 'aku', 'karna', 'udah', 'pasti', 'kotor', 'kali', 'rumahku', 'kemarin', 'ku', 'bilang', 'iya', 'tapi', 'udah', 'capek', 'kali', 'aku', 'wid', 'gak', 'sanggup', 'lagi', 'ini', 'kalian', 'kemarin', 'bimbingan', 'itu', 'laporannya', 'udah', 'kalian', 'print', 'langsung', 'itu', 'kalian', 'foto', 'dulu', 'baru', 'kalian', 'masukkan', 'ke', 'wordnya', 'tunggu', 'di', 'tanda', 'tangan', 'bu', 'margaretha', 'dulu', 'baru', 'kalian', 'masukkan', 'ke', 'lampiran', 'laporan', 'kalian', 'itu', 'klo', 'dah', 'di', 'acc', 'kalian', 'kumpul', 'kemana', 'wid', 'eh', 'kalian', 'kemarin', 'jilid', 'di', 'percetakan', 'apa', 'wid', 'itu', 'wid', 'tanda', 'tangan', 'dekan', 'itu', 'kan', 'ada'

3.1.4 *Filtering*

Pada tahapan *filtering*, dilakukan penyaringan dengan menghapus kata-kata yang tidak memiliki makna analisis, termasuk kata depan dan kata penghubung dari dalam teks. Rincian hasil *filtering* dapat dilihat pada Tabel 4.

Tabel 4. Hasil *Filtering*

Tokenizing	*Filtering*
'template', 'buat', 'penilaian', 'gak', 'ada', 'kan', 'di', 'kasih', 'kampus', 'maksudku', 'kan', 'wid', 'ini', 'template', 'yang', 'kita', 'kasih', 'ke', 'orang', 'ini', 'kan', 'untuk', 'kasih', 'nilai', 'ke', 'kita', 'biar', 'gak', 'capek', 'kali', 'aku', 'karna', 'udah', 'pasti', 'kotor', 'kali', 'rumahku', 'kemarin', 'ku', 'bilang', 'iya', 'tapi', 'udah', 'capek', 'kali', 'aku', 'wid', 'gak', 'sanggup', 'lagi', 'ini', 'kalian', 'kemarin', 'bimbingan', 'itu', 'laporannya', 'udah', 'kalian', 'print', 'langsung', 'itu', 'kalian', 'foto', 'dulu', 'baru', 'kalian', 'masukkan', 'ke', 'wordnya', 'tunggu', 'di', 'tanda', 'tangan', 'bu', 'margaretha', 'dulu', 'baru', 'kalian', 'masukkan', 'ke', 'lampiran', 'laporan', 'kalian', 'itu', 'klo', 'dah', 'di', 'acc', 'kalian', 'kumpul', 'kemana', 'wid', 'eh', 'kalian', 'kemarin', 'jilid', 'di', 'percetakan', 'apa',	['template', 'penilaian', 'gak', 'kasih', 'kampus', 'maksudku', 'wid', 'template', 'kasih', 'orang', 'kasih', 'nilai', 'biar', 'gak', 'capek', 'kali', 'karna', 'udah', 'kotor', 'kali', 'rumahku', 'kemarin', 'ku', 'bilang', 'iya', 'udah', 'capek', 'kali', 'wid', 'gak', 'sanggup', 'kemarin', 'bimbingan', 'laporannya', 'udah', 'print', 'langsung', 'foto', 'masukkan', 'wordnya', 'tunggu', 'tanda', 'tangan', 'bu', 'margaretha', 'masukkan', 'lampiran', 'laporan', 'klo', 'dah', 'acc', 'kumpul', 'kemana', 'wid', 'eh', 'kemarin', 'jilid', 'percetakan', 'wid', 'wid', 'tanda', 'tangan', 'dekan']

'wid', 'itu', 'wid', 'tanda', 'tangan', 'dekan', 'itu', 'kan', 'ada'

3.1.5 *Stemming*

Pada tahapan *stemming*, setiap kata diubah menjadi kata dasar untuk meminimalisir keberagaman kata yang memiliki arti serupa. Rincian hasil stemming dapat dilihat pada Tabel 5.

Tabel 5. Hasil *Stemming*

Filtering	*Stemming*
['template', 'penilaian', 'gak', 'kasih', 'kampus', 'maksudku', 'wid', 'template', 'kasih', 'orang', 'kasih', 'nilai', 'biar', 'gak', 'capek', 'kali', 'karna', 'udah', 'kotor', 'kali', 'rumahku', 'kemarin', 'ku', 'bilang', 'iya', 'udah', 'capek', 'kali', 'wid', 'gak', 'sanggup', 'kemarin', 'bimbingan', 'laporannya', 'udah', 'print', 'langsung', 'foto', 'masukkan', 'wordnya', 'tunggu', 'tanda', 'tangan', 'bu', 'margaretha', 'masukkan', 'lampiran', 'laporan', 'klo', 'dah', 'acc', 'kumpul', 'kemana', 'wid', 'eh', 'kemarin', 'jilid', 'percetakan', 'wid', 'wid', 'tanda', 'tangan', 'dekan']	['template', 'nilai', 'gak', 'kasih', 'kampus', 'maksud', 'wid', 'template', 'kasih', 'orang', 'kasih', 'nilai', 'biar', 'gak', 'capek', 'kali', 'karna', 'udah', 'kotor', 'kali', 'rumah', 'kemarin', 'ku', 'bilang', 'iya', 'udah', 'capek', 'kali', 'wid', 'gak', 'sanggup', 'kemarin', 'bimbing', 'lapor', 'udah', 'print', 'langsung', 'foto', 'masuk', 'wordnya', 'tunggu', 'tanda', 'tangan', 'bu', 'margaretha', 'masuk', 'lampir', 'lapor', 'klo', 'dah', 'acc', 'kumpul', 'mana', 'wid', 'eh', 'kemarin', 'jilid', 'cetak', 'wid', 'wid', 'tanda', 'tangan', 'dekan']

3.2. Implementasi Perhitungan Manual

3.2.1 Pembobotan Data

proses *preprocessing* diselesaikan, frekuensi kemunculan kata dalam setiap dokumen dihitung dengan menggunakan metode *term frequency (TF). Term frequency* dapat dilihat pada Tabel 6.

Tabel 6. Hasil Pembobotan data

Term	**D1**	**D2**	**D3**	**D4**	**D5**	**D6**	**….**	**D101**	**D102**	**D103**
template	2	2	0	0	0	0	….	0	0	0
nilai	2	0	0	0	0	0	….	0	0	1
gak	3	1	1	2	0	0	….	1	2	0
kasih	3	0	0	0	0	0	….	0	0	1
kampus	1	0	0	0	0	0	….	0	0	1
maksud	1	0	0	0	0	0	….	0	0	0
…….	….	….	….	….	….	….	….	….	….	….
minta	0	0	0	0	0	0	….	0	0	0
contoh	0	0	0	0	0	0	….	0	0	0
pkl	0	0	0	0	0	0	….	0	0	0
makasih	0	0	0	0	0	0	….	0	0	0

3.2.2 Cosine Similarity

Untuk menghitung kemiripan antar data menggunakan metode Cosine Similarity yang dirumuskan sebagai berikut :

$$\textit{Similarity}\ (A, B) = \cos(\theta) = \frac{A.B}{||A||\,||B||} \quad \frac{\sum_{i=1}^{n} A_i\, B_i}{\sqrt{\sum_{i=1}^{n} A_i^2 \;\; \sum_{i=1}^{n} B_i^2}}$$

1. Perhitungan Data Uji 101

Perhitungan manual D101 menggunakan metode *cosine similarity* dapat dilihat pada tabel Tabel 7.

Tabel 7. Data pesan singkat hasil *cosine similarity* D101

Data Uji	Data Pembanding	Pengirim	*Similarity*
D101	D1	Wulan Siallagan	16%
D101	D2	Lister Barus	11%
D101	D3	Harun Marpaung	15%
D101	D4	Oky Alexander	12%
D101	D5	Iren Pricisillia	22%
D101	D6	Putri Pane	26%
D101	D7	Gresko Silalahi	28%
D101	D8	Gilbert Pangaribuan	41%
D101	D9	Nadyarni Duha	71%
D101	D10	Novita Nainggolan	10%
D101	D11	Melchiman	11%
D101	D12	Irwan Samosir	32%
D101	D13	Keren Gabriella	12%
D101	D14	Frans Gurusinga	18%
D101	D15	Irfan Martin	23%

Tingkat kemiripan antar dokumen beragam, dengan nilai tertinggi 71% antara D101 dan D9, serta terendah 10% pada D101 dan D10. Ini menunjukkan bahwa D101 memiliki pola paling mirip dengan D9 dibandingkan dokumen lainnya. Berdasarkan hasil ini, D9 yang memiliki kemiripan tertinggi diketahui sebagai pesan dari Nadyarni Duha.

2. Perhitungan Data Uji 102
   Perhitungan manual D102 menggunakan metode *cosine similarity* dapat dilihat pada tabel Tabel 8.

Tabel 8. Data pesan singkat hasil *cosine similarity* D102

Data Uji	Data Pembanding	Pengirim	*Similarity*
D102	D1	Wulan Siallagan	31%
D102	D2	Lister Barus	14%
D102	D3	Harun Marpaung	37%
D102	D4	Oky Alexander	95%
D102	D5	Iren Pricisillia	13%
D102	D6	Putri Pane	15%
D102	D7	Gresko Silalahi	23%
D102	D8	Gilbert Pangaribuan	17%
D102	D9	Nadyarni Duha	16%
D102	D10	Novita Nainggolan	26%
D102	D11	Melchiman	19%
D102	D12	Irwan Samosir	11%
D102	D13	Keren Gabriella	23%
D102	D14	Frans Gurusinga	28%
D102	D15	Irfan Martin	21%

Tingkat kemiripan antar dokumen beragam, dengan nilai tertinggi 95% antara D102 dan D4, serta terendah 11% pada D102 dan D12. Ini menunjukkan bahwa D102 memiliki pola paling mirip dengan D4 dibandingkan dokumen lainnya. Berdasarkan hasil ini, D4 yang memiliki kemiripan tertinggi diketahui sebagai pesan dari Oky Sihotang.

3. Perhitungan Data Uji 103

Perhitungan manual D103 menggunakan metode *cosine similarity* dapat dilihat pada tabel Tabel 9.

Tabel 9. Data pesan singkat hasil *cosine similarity* D103

Data Uji	Data Pembanding	Pengirim	*Similarity*
D103	D1	Wulan Siallagan	25%
D103	D2	Lister Barus	10%
D103	D3	Harun Marpaung	19%
D103	D4	Oky Alexander	14%
D103	D5	Iren Pricisillia	6%
D103	D6	Putri Pane	10%
D103	D7	Gresko Silalahi	5%
D103	D8	Gilbert Pangaribuan	14%
D103	D9	Nadyarni Duha	4%
D103	D10	Novita Nainggolan	14%
D103	D11	Melchiman	11%
D103	D12	Irwan Samosir	5%
D103	D13	Keren Gabriella	19%
D103	D14	Frans Gurusinga	7%
D103	D15	Irfan Martin	77%

Tingkat kemiripan antar dokumen bervariasi, dengan nilai tertinggi 77% antara D103 dan D15, serta terendah 4% pada D103 dan D9. Hal ini menunjukkan bahwa D103 memiliki pola paling mirip dengan D15 dibandingkan dokumen lainnya. Berdasarkan hasil ini, D15 yang memiliki kemiripan tertinggi diketahui sebagai pesan dari Irfan Tambunan.

3.3. Implementasai Berbasis *Python*

Penelitian ini diimplementasikan menggunakan bahasa pemrograman *Python* dan dijalankan melalui *Jupyter Notebook*, yang merupakan sebuah *executable document* atau dengan kata lain berbentuk *notebook* yang dapat digunakan untuk menyimpan, menulis, dan membagikan program yang telah ditulis. Hasil perhitungan dengan menggunakan python dapat di lihat pada gambar 4.

```
Cosine Similarity untuk D101:
Cosine Similarity D101 dengan D1: 0.1626 (16%)
Cosine Similarity D101 dengan D2: 0.1058 (11%)
Cosine Similarity D101 dengan D3: 0.1484 (15%)
Cosine Similarity D101 dengan D4: 0.1179 (12%)
Cosine Similarity D101 dengan D99: 0.2099 (21%)
Cosine Similarity D101 dengan D100: 0.0393 (4%)

Dokumen paling mirip dengan D101 adalah D9 dengan Cosine Similarity: 0.7101 (71%)
Pengirim: Nadyarni Duha


Cosine Similarity untuk D102:
Cosine Similarity D102 dengan D1: 0.3145 (31%)
Cosine Similarity D102 dengan D2: 0.1405 (14%)
Cosine Similarity D102 dengan D3: 0.3684 (37%)
Cosine Similarity D102 dengan D4: 0.9510 (95%)
Cosine Similarity D102 dengan D99: 0.0407 (4%)
Cosine Similarity D102 dengan D100: 0.0112 (1%)

Dokumen paling mirip dengan D102 adalah D4 dengan Cosine Similarity: 0.9510 (95%)
Pengirim: Oky Sihotang
Cosine Similarity untuk D103:
Cosine Similarity D103 dengan D1: 0.2472 (25%)
Cosine Similarity D103 dengan D2: 0.1005 (10%)
Cosine Similarity D103 dengan D3: 0.1922 (19%)
Cosine Similarity D103 dengan D4: 0.1374 (14%)
Cosine Similarity D103 dengan D99: 0.0544 (5%)
Cosine Similarity D103 dengan D100: 0.0560 (6%)

Dokumen paling mirip dengan D103 adalah D15 dengan Cosine Similarity: 0.7652 (77%)
Pengirim: Irfan Tambunan
```

Gambar 3. Hasil Perhitungan *python*

Dari hasil perhitungan *Cosine Similarity*, tingkat kemiripan setiap data uji dengan dokumen pembanding bervariasi. D101 memiliki kemiripan tertinggi sebesar 71% dengan D9, menunjukkan bahwa pola D101 paling mirip dengan dokumen D9, yang diketahui sebagai pesan yang dikirim oleh Nadyarni Duha. D102 memiliki kemiripan tertinggi sebesar 95% dengan D4, menunjukkan bahwa D4 memiliki pola yang paling mirip dengan data uji D102, yang diketahui sebagai pesan yang dikirim oleh Oky Sihotang. D103 memiliki kemiripan tertinggi sebesar 77% dengan D15, yang berarti D15 adalah dokumen paling mirip dengan D103, yang diketahui sebagai pesan yang dikirim oleh Irfan Tambunan.

4. KESIMPULAN

Berdasarkan hasil penelitian analisis yang dilakukan pada kasus di atas, maka dapat diambil kesimpulan bahwa metode Cosine Similarity dalam menganalisis kemiripan pesan teks berhasil diterapkan dalam penelitian ini untuk mengenali pengirim pesan singkat. Hasil analisa pengenalan pengirim pesan singkat dengan metode Cosine Similarity menunjukkan bahwa D101 memiliki kemiripan tertinggi dengan D9 sebesar 71%, D102 dengan D4 sebesar 95%, dan D103 dengan D15 sebesar 77%. Hal ini menunjukkan kemampuannya dalam mengenali pengirim pesan singkat. Pengujian menunjukkan bahwa sistem yang dikembangkan dapat mengelompokkan SMS ke dalam kategori seperti normal, penipuan, dan penipuan lainnya dengan tingkat akurasi yang baik[13].

REFERENSI

[1] L. Meilina, I. N. S. Kumara, and I. N. Setiawan, "Literature Review Klasifikasi Data Menggunakan Metode Cosine Similarity dan Artificial Neural Network," *Maj. Ilm. Teknol. Elektro*, vol. 20, no. 2, p. 307, 2021, doi: 10.24843/mite.2021.v20i02.p15.

[2] T. A. Prismadana, "Aplikasi Ruang Tugas Dengan Deteksi Kemiripan Teks Pada Dokumen Tugas Menggunakan Cosine Similarity," *J. Inform. dan Multimed.*, vol. 15, no. 1, pp. 31–37, 2023, doi: 10.33795/jtim.v15i1.4405.

[3] D. Robison Manalu, M. Christofell, L. Tobing, and M. Yohanna, "METHOMIKA: Jurnal Manajemen Informatika & Komputerisasi Akuntansi ANALISIS SENTIMEN TWITTER TERHADAP WACANA PENUNDAAN PEMILU DENGAN METODE SUPPORT VECTOR MACHINE," *Methomika*, vol. 6, no. 2, pp. 149–156, 2022.

[4] H. G. Simanullang, S. Wijono, B. Kristianto, W. Sulistyo, and A. P. Silalahi, "Detecting Cyberbullying Using Machine Learning Approaches," in *2024 Ninth International Conference on Informatics and Computing (ICIC)*, 2024, pp. 1–8. doi: 10.1109/ICIC64337.2024.10956842.

[5] S. M. Sipayung, M. Yohanna, H. G. Manullang, and R. Mandala, "ANALISIS SENTIMEN MENGGUNAKAN K-NEAREST NEIGHBOR TERHADAP FILM NGERI-NGERI SEDAP," vol. 4, no. 2, pp. 42–52, 2024.

[6] N. Arifin, U. Enri, and 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, p. 129, 2021, doi: 10.30998/string.v6i2.10133.

[7] D. Septiani and I. Isabela, "Analisis Term Frequency Inverse Document Frequency (TF-IDF) Dalam Temu Kembali Informasi Pada Dokumen Teks," *SINTESIA J. Sist. dan Teknol. Inf. Indones.*, vol. 1, no. 2, pp. 81–88, 2023.

[8] L. M. Barus, H. G. Simanullang, and A. P. Silalahi, "Classification of English-Language Racial Hate Speech Using Random Forest "," *J. Intell. Comput. Inf. Syst.*, vol. 1, no. 1, pp. 1–17, 2025.

[9] H. G. Simanullang, A. P. Silalahi, and G. P. Pangaribuan, "PENGELOMPOKAN KOMENTAR PENGGUNA JKN MOBILE MENGGUNAKAN METODE K-MEDOIDS," 2025.

[10] D. N. Lindang, A. Y. Muniar, A. Halid, M. Muhajirin, and A. Amiruddin, "Sistem Penentuan Kemiripan Antar Skripsi Menggunakan Metode Cosine Similarity Pada Perpustakaan," *Sntei*, pp. 321–324, 2022.

[11] N. Duha, H. G. Simanullang, and A. P. Silalahi, "ANALISIS PENGARUH VARIASI NILAI P PADA METODE MINKOWSKI DISTANCE DALAM MENENTUKAN KEMIRIPAN ABSTRAK SKRIPSI Harlen Gilbert Simanullang , Arina Prima Silalahi , Nadyarni Natalis Caesarin Duha," *METHOMIKA J. Manaj. Inform. dan Komputerisasi Akunt.*, vol. 9, no. 2, pp. 255–263, 2025.

[12] F. A. Nugroho, F. Septian, D. A. Pungkastyo, and J. Riyanto, "Penerapan Algoritma Cosine Similarity untuk Deteksi Kesamaan Konten pada Sistem Informasi Penelitian dan Pengabdian Kepada

Masyarakat," *J. Inform. Univ. Pamulang*, vol. 5, no. 4, p. 529, 2021, doi: 10.32493/informatika.v5i4.7126.

[13] A. Wara Putera and Y. Dewi Lestari, "Klasifikasi SMS Spam Menggunakan Algoritma K-Nearest Neighbor," *Jikstra*, vol. 5, no. 01, 2023.