KRESNA: Jurnal Riset dan Pengabdian Masyarakat
Volume 3, Nomor 2, November, 2023, Hal: 156-165 E-ISSN: 2809-6509

Penerapan Metode *Doc-P* untuk Deteksi Trending Topik Pemilihan Presiden Pada *Twitter*

Implementation of The Doc-P Method for Detecting Trending Topics of Presidential Elections on Twitter

Rizky Fajarudin[1], Indra[2*]

[12]Teknik Informatika, Fakultas Teknologi Informasi
Universitas Budi Luhur
Email: [1]rizky18.rf@gmai.com, [2*]indra@budiluhur.ac.id
(* corresponding author)

Abstract

In the rapid era of digitization and the advancement of information technology, social media has become a primary source of information, including regarding the presidential candidates' selection in Indonesia. Platforms like Twitter, especially, serve as avenues for people to share views, opinions, and real-time information about presidential candidates, political parties, and related issues. However, the abundance of information on social media, particularly in the form of tweets on Twitter, poses a challenge for the public in recognizing the trending presidential candidates. Identifying popular topics becomes a crucial challenge in social media data analysis. To address this, this study introduces the Doc-p method that utilizes LSH (Locality Sensitive Hashing) clustering, cosine similarity, and TF-IDF (Term Frequency-Inverse Document Frequency) to identify trending topics on Twitter. The main issue tackled is the difficulty in manually detecting popular tweet topics related to presidential candidates. The aim of this research is to facilitate the public in recognizing the most talked-about presidential candidates on Twitter, aiding them in discovering viral tweets about the candidates. Text mining techniques are employed in this study, with the Doc-p method being one of its components. Based on the research, the highest clustering result is achieved with a score of 98.0 for cluster 0. Overall, this study successfully addresses the challenges of identifying trending topics related to the presidential election on Twitter. By applying the Doc-p method and text mining techniques, the goal of this research is to support the public in easily identifying and comprehending the most-discussed presidential candidates through popular tweets on social media.

Keywords*: Text Mining, Metode Doc-p, LSH, Cosine similarity, TF-IDF, Pre-processing, Twitter*

Abstrak

Dalam era digitalisasi yang pesat dan perkembangan teknologi informasi, media sosial telah menjadi sumber utama informasi, termasuk mengenai pemilihan calon presiden di Indonesia. Platform seperti Twitter khususnya menjadi wadah bagi orang-orang untuk berbagi pandangan, opini, dan informasi secara real-time mengenai calon presiden, partai politik, dan isu terkait. Namun, melimpahnya informasi di media sosial, terutama dalam bentuk tweet di Twitter, menciptakan kesulitan bagi masyarakat dalam mengenali calon presiden yang tengah menjadi tren. Menemukan topik yang sedang populer menjadi tantangan penting dalam analisis data media sosial. Untuk mengatasi hal ini, penelitian ini memperkenalkan metode Doc-p yang menggunakan klasterisasi LSH (*Locality Sensitive Hashing*), cosine similarity, dan TF-IDF (*Term Frequency-Inverse Document Frequency*) untuk mengidentifikasi topik yang sedang trend di Twitter. Isu utama yang diatasi adalah kesulitan dalam mendeteksi secara manual topik tweet yang sedang populer terkait calon presiden. Tujuan dari penelitian ini adalah memudahkan masyarakat dalam mengenali calon presiden yang paling banyak dibicarakan di Twitter, sehingga membantu mereka menemukan tweet yang sedang viral tentang calon presiden. Teknik text mining digunakan dalam penelitian ini, dengan metode Doc-P sebagai salah satu komponennya. Berdasarkan penelitian, hasil klasterisasi tertinggi diperoleh dengan skor 98.0 untuk klaster 0. Secara keseluruhan, penelitian ini berhasil mengatasi tantangan dalam mengenali topik yang sedang tren terkait pemilihan presiden di Twitter. Dengan menerapkan metode Doc-p dan teknik text mining, tujuan

penelitian ini adalah mendukung masyarakat dalam dengan mudah mengidentifikasi dan memahami calon presiden yang paling banyak dibicarakan melalui tweet yang sedang populer di media sosial.

Kata Kunci: *Text Mining, Metode Doc-p, LSH, Cosine similarity, TF-IDF, Pre-processing, Twitter*

1. PENDAHULUAN

Dalam era digital dan perkembangan teknologi informasi yang pesat, media sosial telah menjadi salah satu sumber utama untuk mendapatkan informasi tentang berbagai topik, termasuk mengenai pemilihan calon presiden di Indonesia [1]. Media sosial, khususnya platform seperti *Twitter,* telah menjadi tempat di mana masyarakat dapat dengan mudah berbagi pandangan, opini, dan informasi terkini mengenai calon presiden, partai politik, dan isu-isu terkait [2].

Namun, semakin banyaknya informasi yang dihasilkan di media sosial, terutama dalam bentuk *tweet* di *Twitter*, membuat masyarakat kesulitan untuk menentukan calon presiden mana yang sedang menjadi *trending topic* atau topik yang paling banyak dibicarakan oleh pengguna media sosial. Kondisi ini dapat menyebabkan kesulitan bagi masyarakat dalam memperoleh informasi yang relevan dan akurat mengenai pemilihan calon presiden, serta dapat menimbulkan kebingungan dalam menyusun pandangan atau opini mereka mengenai isu-isu politik [3].

Salah satu solusi yang dapat membantu masyarakat dalam mencari *tweet-tweet* yang sedang trending mengenai calon presiden adalah dengan menggunakan teknik *text mining. text mining* adalah proses ekstraksi pola berupa informasi dan pengetahuan yang berguna dari sejumlah besar sumber data teks, seperti dokumen Word, PDF, kutipan teks, dan tentu saja, *tweet* di media sosial. Dengan menggunakan metode *text mining*, kita dapat mengolah dan menganalisis data teks dalam jumlah besar dengan lebih efisien dan akurat [4].

Dalam konteks penelitian ini, salah satu metode *text mining* yang akan digunakan adalah metode *Doc-p. Doc-p* adalah salah satu pendekatan dalam *text mining* yang menggunakan klasterisasi LSH *(Locality Sensitive Hashing), cosine similarity,* dan *TF-IDF (Term Frequency-Inverse Document Frequency)* berdasarkan kemiripan isinya. Metode ini berfokus pada kemampuan untuk mengidentifikasi topik atau kelompok-kelompok *tweet* yang serupa secara tematik, sehingga memungkinkan kita untuk mengenali tren atau topik yang sedang populer di kalangan pengguna *Twitter* terkait pemilihan calon presiden [5].

Adapun beberapa penelitian yang sudah dilakukan dalam deteksi trending topik. Pada Penelitian Futuhul Hadi et al., 2017 yaitu *Text Mining* pada Media Sosial Twitter studi kasus: masa tenang PILKADA DKI 2017 putaran 2, metode unsupervised learning kami menemukan bahwa metode k-means tidak dapat memberikan hasil yang merata pada setiap kelompoknya. Sebaliknya, hasil luaran dari pemodelan topik (topic modeling – Latent Dirichlet Allocation) lebih merata [6]. Selain itu pada penelitian yang dilakukan Indra et al., 2019 Trending topics detection of Indonesian tweets using BN-grams and Doc-p, Deteksi trending topik di tweet berbahasa Indonesia adalah dipengaruhi oleh preprocessing dan jumlah total yang dikumpulkan tweet. Eksperimen menunjukkan bahwa deteksi topik yang sedang tren di Tweet bahasa Indonesia lebih akurat jika menggunakan BN-gram daripada Doc-p. BN-gram menghasilkan akurasi yang lebih tinggi dalam mendeteksi tren topik daripada Doc-p di ketiga set data. Namun, untuk presisi kata kunci, Doc-p lebih baik daripada BN-gram [5]. Selain itu pada penelitian Aiello et al.,2013 *Sensing Trending Topics in Twitter, we generated and evaluated topics at different stages of the event to capture the evolving stories related to it. All algorithms leverage the content dimension with different approaches, ranging from the analysis of co-occurrence of unigrams to co-occurrence of unigrams, up to co-occurrence of -grams. The -gram-based method outperforms the others, indicating that more complex keyword aggregation is better at capturing factual topics* [7].

2. METODE PENELITIAN

2.1 Text Mining

Text mining merupakan salah satu teknologi yang mengelola data teks untuk memperoleh informasi secara otomatis [8]. Dengan *text mining*, informasi baru bisa didapatkan melalui hasil analisis pada data teks semi terstruktur maupun tidak terstruktur (biasanya dalam jumlah besar) [9]. Hal ini tentu sangat membantu pekerjaan manusia seiring dengan semakin banyaknya data teks ataupun dokumen yang ada

pada aplikasi *web*, aplikasi *digital*, maupun media sosial[10]. Tentunya data-data tersebut memiliki jumlah yang besar dan kurang terstruktur sehingga perlu waktu lama untuk menganalisis informasi di dalamnya. *Text mining* dikenal juga dengan istilah lain seperti *Intelligent Text Analysis (ITA), Text Data Mining (TDM) atau Knowledge-Discovery in Text (KDT)* [11]. Menurut Hearst, *text mining* dapat diartikan sebagai penemuan informasi yang baru dan tidak diketahui sebelumnya oleh komputer dengan secara otomatis mengekstrak informasi dari sumber-sumber yang berbeda. Kunci dari proses ini adalah menggabungkan relasi dan fakta di dalam teks dari berbagai sumber hingga berhasil mengekstrak Informasi [12].

2.2 Crawling Data

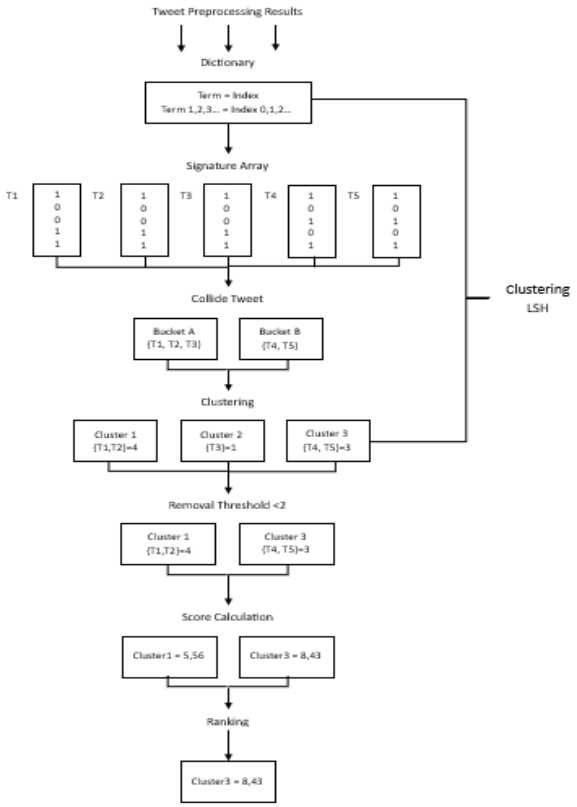
Data yang digunakan pada penelitian ini adalah data *tweet* maupun *retweet* yang diambil langsung dari jejaring sosial *twitter* dengan memanfaatkan *API key* yang sebelumnya sudah didaftarkan melalui *twitter developer*. Menggunakan Bahasa Pemograman Python, komponen *key* dan *secret* diisikan. Selanjutnya dilakukan memasukkan kata kunci dari topik yang dibuat yaitu yang berhubungan dengan Calon Presiden 2024 - 2029, Prabowo Subianto, Anis Baswedan, Ganjar Pranowo Gerindra, Nasdem, Pdip. *Data tweet* yang sudah di *crawling* disimpan ke dalam file excel [13].

2.3 Pre-Processing

Preprocessing bertujuan untuk membersihkan data-data yang tidak diperlukan serta menyeragamkan kata-kata yang memiliki arti sama agar proses mining lebih akurat. Tahapan ini terdiri dari lima 6 proses utama antara lain: *case folding, cleansing tokenizing, stopword, normalized, stemming* [14].

2.4 Metode *Document pivot*

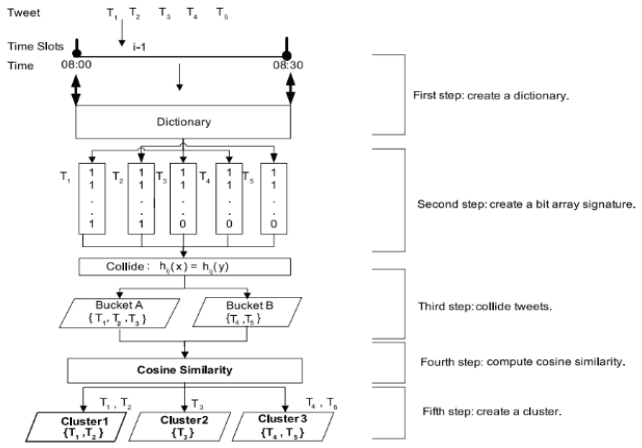
Metode *Doc-p* terdiri dari empat tahapan. Tahap pertama, pembentukan *clustering* dokumen dengan *Locality sensitivity hashing (LSH).* Tahap kedua, penghapusan klaster jika jumlah anggota dari klaster tersebut di bawah *threshold*. Tahap ketiga, setiap klaster dilakukan perhitungan skor. Tahap keempat, penentuan trending topik berdasarkan perangkingan topik [5].



Gambar 1. Alur Keseluruhan Metode Doc-p [5]

1) Pengelompokan tweet menggunakan LSH

Pengelompokan *tweet* menggunakan *LSH* memiliki lima langkah, seperti pada Gambar 2 Alur Clustering LSH.



Gambar 2. Alur *Clustering LSH* [5]

a. Pertama, kamus, yang terdiri dari glosarium unik kumpulan *tweet*, dibuat. Setiap entri dalam kamus memiliki istilah indeks, yaitu satu kata dalam sebuah kalimat.
b. Kedua, berdasarkan istilah indeks dalam kamus, setiap *tweet* yang terkumpul diubah menjadi *bit array signature* dan dimasukkan ke dalam kumpulan tabel hash S. Metode *LSH* menggunakan *k bit* dan tabel *hash L* dan dua dokumen dianggap bertabrakan jika dan hanya jika kedua dokumen tersebut memiliki *bit array signature* yang sama. Dokumen adalah beberapa *tweet* yang diposting dalam jangka waktu tertentu. Pada penelitian ini *bit* tanda tangan *array* adalah 17 *bit*.
c. Ketiga, *tweet* bertabrakan, yaitu *tweet* yang memiliki *bit array signature* yang sama dengan *tweet* lainnya, dimasukkan ke dalam keranjang yang sama di koleksi tabel *hash S*.
d. Keempat, cosine similarity dihitung pada *tweet* di *S*.
e. Pada langkah kelima, jika skor kesamaan kosinus melebihi ambang batas tertentu, *tweet* akan dimasukkan ke dalam *cluster* yang sama jika cosine similarity di bawah threshold, *cluster* baru akan terbentuk.

Rumus untuk pembobotan *tf-idf*

$$\text{Rumus } W_{i.j} = tf_{i.j}\ x\ idf_i = tf_{i.j} x \log\left(\frac{N}{df_j} + 1\right) \quad (1)$$

Keterangan :

- $W_{i.j}$: bobot *term* j pada dokumen i.
- idf_i : invers dokumen frekuensi.
- $tf_{i.j}$: frekuensi *term* j pada dokumen i.
- N : jumlah total dokumen yang diproses.
- df_j : dokumen yang dimiliki *term* j di dalamnya.

Rumus Cosine Similarity

$$\text{Rumus } CosSim(d_j, q_k) = \frac{\Sigma_{i=1}^{n}(td_{ij} x tq_{ik})}{\sqrt{\Sigma_{i=1}^{n} td_{ij}{}^2 x \Sigma_{i=1}^{n} tq_{ik}{}^2}} \quad (2)$$

Keterangan:

- CosSim(dj,qk) : tingkat kesamaan dokumen dengan *query* tertentu
- td_{ij} : *term* ke-i vektor untuk dokumen ke-j
- tq_{ik} : *term* ke-i dalam *vektor* untuk *query* ke-k
- n : jumlah *term* yang unik dalam *dataset*

2) Penghapusan cluster yang anggotanya berada di bawah threshold

Ambang batas yang digunakan dalam penelitian ini adalah 2; karenanya dihasilkan *cluster* yang anggotanya kurang dari 2 akan dihilangkan [5].

3) Perhitungan Skor Masing-Masing Cluster

Score cluster ditentukan oleh Persamaan

$$Score_c = \sum_{i=1}^{|Docs_c|} \sum_{j=1}^{|Words_i|} exp\left(-p(W_{ij})\right) \quad (3)$$

Dimana $P(W_{ij})$ adalah probabilitas frekuensi kejadian istilah *j* dalam dokumen *i* dalam *cluster* diberikan korpus yang digunakan

$$p(W|corpus) = \frac{N_w + \delta}{(\sum_u^n N_u) + \delta n} \quad (4)$$

Dimana N_w adalah jumlah kemunculan istilah *w* dalam korpus, N_u adalah total kejadian istilah *u*, dan δ adalah perataan konstan. Di dalam penelitian, δ diatur ke 0,5. Korpus secara bersamaan adalah kumpulan kata dan kumpulan dokumen [5].

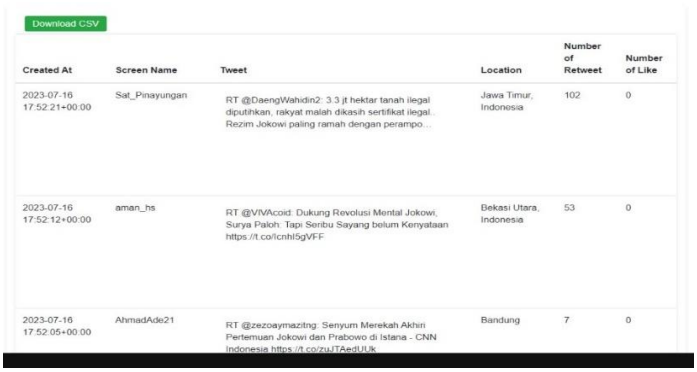
4) Peringkat Topik

Topik yang sedang tren direpresentasikan dalam bentuk sekumpulan kata kunci di setiap klaster. *Cluster* diurutkan berdasarkan skor masing-masing klaster dan klaster dengan skor tertinggi akan menjadi trending topik[5].

3. HASIL DAN PEMBAHASAN

3.1 Pengujian Crawling Data

Pengujian *crawling data* ini menggunakan *keyword* Calon Presiden 2024 – 2029, Prabowo Subianto, Anis Baswedan, Ganjar Pranowo Gerindra, Nasdem, Pdip.


Download CSV

Created At	Screen Name	Tweet	Location	Number of Retweet	Number of Like
2023-07-16 17:52:21+00:00	Sat_Pinayungan	RT @DaengWahidin2: 3.3 jt hektar tanah ilegal diputihkan, rakyat malah dikasih sertifikat ilegal. Rezim Jokowi paling ramah dengan perampo...	Jawa Timur, Indonesia	102	0
2023-07-16 17:52:12+00:00	aman_hs	RT @VIVAcoid: Dukung Revolusi Mental Jokowi, Surya Paloh: Tapi Serbu Sayang belum Kenyataan https://t.co/IcnhI5gVFF	Bekasi Utara, Indonesia	53	0
2023-07-16 17:52:05+00:00	AhmadAde21	RT @zezoaymazitng: Senyum Merekah Akhiri Pertemuan Jokowi dan Prabowo di Istana - CNN Indonesia https://t.co/zuJTAedUJk	Bandung	7	0

Gambar 3. Hasil *Crawling Data*

Pada pengujian ini pada saat menggunakan *api key twitter* terdapat limit dalam mengambil jumlah *data tweet.*

3.2 Pengujian *Preprocessing*

Pengujian *preprocessing* ini menggunakan *case folding, cleansing, tokenizing, stopword, normalized, and stemming.*


Preprocessing Results

Original Text	Case Folding	Cleansing	Tokenizing	Stopword	Normalization	Stemming	Processed Text
@FahmiAgustian Walaupun bukan pro atau kontra sama jokowi. Cuman ya serba salah jd anaknya. Ikut di pemerintahan sa... https://t.co/DyRNOZDiRr	@fahmiagustian walaupun bukan pro atau kontra sama jokowi. cuman ya serba salah jd anaknya. ikut di pemerintahan sa... https://t.co/dymozdirr	pro kontra jokowi cuman serba salah anak perintah sa	['pro', 'kontra', 'jokowi', 'cuman', 'serba', 'salah', 'anak', 'perintah', 'sa']	['pro', 'kontra', 'jokowi', 'cuman', 'serba', 'salah', 'anak', 'perintah', 'sa']	['pro', 'kontra', 'jokowi', 'cuman', 'serba', 'salah', 'anak', 'perintah', 'sa']	['pro', 'kontra', 'jokowi', 'cuman', 'serba', 'salah', 'anak', 'perintah', 'sa']	pro kontra jokowi cuman serba salah anak perintah sa
RT @Kanseulir: Hasil Survey, Capres yang Ikuti Arahan Jokowi dapat angka 19%? Berarti Capres Yang Diendorse Jokowi Akan Kalah? Begitulah!...	rt @kanseulir: hasil survey, capres yang ikuti arahan jokowi dapat angka 19%? berarti capres yang diendorse jokowi akan kalah? begitulah!...	hasil survey capres ikut arah jokowi angka capres diendorse jokowi kalah begitu	['hasil', 'survey', 'capres', 'ikut', 'arah', 'jokowi', 'angka', 'capres', 'diendorse', 'jokowi', 'kalah', 'begitu']	['hasil', 'survey', 'capres', 'arah', 'jokowi', 'angka', 'capres', 'diendorse', 'jokowi', 'kalah']	['hasil', 'survey', 'capres', 'arah', 'jokowi', 'capres', 'diendorse', 'jokowi', 'kalah']	['hasil', 'survey', 'capres', 'arah', 'jokowi', 'angka', 'capres', 'diendorse', 'jokowi', 'kalah']	hasil survey capres ikut arah jokowi angka capres diendorse jokowi kalah begitu
RT @Budi_12345678: Dukung Pak @mohmahfudmd 👍🙏	rt @budi_12345678: dukung pak @mohmahfudmd 👍	dukung cocok capre survei	['dukung', 'cocok', 'capre',	['dukung', 'cocok', 'capre',	['dukung', 'cocok', 'capre',	['dukung', 'cocok', 'capre',	dukung cocok capre survei

Gambar 4. Hasil *Pre-Processing*

Dapat dilihat pada Gambar 4. bagaimana proses preprocessing berlangsung dari tahap pertama sampai tahap berikutnya.

3.3 Metode Documen Pivot (*Doc-p*)

Metode *Doc-p* terdiri dari empat tahapan. Tahap pertama, pembentukan *clustering* dokumen dengan *Locality sensitivity hashing (LSH)*. Tahap kedua, penghapusan klaster jika jumlah anggota dari klaster tersebut di bawah *threshold*. Tahap ketiga, setiap klaster dilakukan perhitungan skor. Tahap keempat, penentuan trending topik berdasarkan perangkingan topik.

3.3.1. Klasterisasi dengan *LSH*

Klasterisasi *tweet* dengan *LSH* Klasterisasi dengan *LSH* menggunakan tiga tahapan. Penjelasan dari setiap tahapan *LSH* dijelaskan dengan tiga langkah yaitu:

a. Pembuatan *dictionary* dari sekumpulan *tweet*

Pada tahap ini dilakukan pembuatan *dictionary*. *Dictionary* adalah sekumpulan *term* dan indeks dari *tweet* yang berasal dari Tabel 1. Ilustrasi dari pembuatan *dictionary* dapat dilihat pada Tabel 2. Pembuatan *dictionary* diawali dengan prapemrosesan salah satunya untuk menghilangkan *stopword* dan tanda baca. Hasil prapemrosesan dapat dilihat pada Tabel 1. Tabel 2. berisi kamus kata (*dictionary*) yang berisi dua kolom yaitu indeks dan *term*. Indeks adalah *id* dari *term* unik yang berasal dari sekumpulan *tweet.* Sedangkan *term* adalah kata yang berupa *unigrams* dari pesan yang diposting pada *Twitter*.

Tabel 1. *Sample Data Tweet Preprocessing*

Tweet (T_i)	*Text Tweet*	*User Name*
T_1	['calon', 'presiden', 'pilih', 'prabowo', 'subianto', 'ganjar', 'pranowo', 'anies', 'baswedan']	e25857284
T_2	['calon', 'presiden', 'pilih', 'prabowo', 'subianto', 'ganjar', 'pranowo', 'anies', 'baswedan']	EedhySaputra
T_3	['jakarta', 'prabowo', 'subianto', 'unggul', 'ganjar', 'pranowo', 'anies', 'baswedan', 'gagal', 'tiket', 'maju', 'calon']	lehon_mangolu
T_4	['jakarta', 'prabowo', 'subianto', 'unggul', 'ganjar', 'pranowo', 'anies', 'baswedan', 'gagal', 'tiket', 'maju', 'calon']	CindyCarrolina
T_5	['pooling', 'presiden', 'ganjar', 'pranowo', 'prabowo', 'anies', 'baswedan', 'lihat', 'hasil']	YusakMachrus

Tabel 2. Pembuatan *Dictionary*

Indeks (x)	*Term*	Indeks (x)	*Term*	Indeks (x)	*Term*
0	calon	7	anies	14	pooling
1	presiden	8	baswedan	15	lihat
2	pilih	9	jakarta	16	hasil
3	prabowo	10	unggul		
4	subianto	11	gagal		
5	ganjar	12	tiket		
6	pranowo	13	maju		

b. Merubah *tweet* menjadi *signature bit array 5 bit* untuk dimasukkan ke dalam *LSH*.

Tabel 3. berisi kolom a dan b di mana a dan b merupakan bilangan *integer* yang dibuat secara *random*. Sedangkan kolom p adalah bilangan *prima*. Nilai x adalah indeks dari *term* pada *dictionary* seperti pada Tabel 2. Pada penelitian ini, nilai a dan b berturut-turut adalah 5, 2, 1, 2, 3 dan 59, 3, 4, 5, 2. Sedangkan nilai p sama untuk lima baris yakni 79. Nilai x adalah indeks *term* "calon" dan "presiden" yang bernilai 0 dan 1.

Selanjutnya, dilakukan pembuatan nilai *Distribusi Gaussian* yang berasal dari *m-dimensional random vector* dengan *range* 0 sampai 78 dengan *parameter mean* 0 dan *variance* 1 serta *m*=79 seperti pada Gambar 5. Selanjutnya, berdasarkan daftar nilai *Distribusi Gaussian* untuk indeks ke 59 dan 64 memiliki nilai *Distribusi Gaussian* berturut turut adalah 0,26920276957040257 (dibulatkan menjadi 0,27) dan 0,2231175194768114 (dibulatkan menjadi 0,22). Kedua nilai *Distribusi Gaussian* tersebut dilakukan penjumlahan dengan tiga *term* yang lain pada *tweet* ke-1 sehingga menghasilkan nilai 1,26.

Hasil penjumlahan *Distribusi Gaussian* tersebut jika nilainya lebih dari 0 maka *bit* bernilai 1. Sebaliknya jika hasil penjumlahan *Distribusi Gaussian* di bawah 0 maka *bit* bernilai 0. Hasil

penjumlahan *Distribusi Gaussian* pada baris pertama pada Tabel 3. adalah 1,26. Oleh karena itu, *bit* yang dihasilkan pada baris pertama adalah *bit* 1. Proses pada baris pertama Tabel 3. memiliki kesamaan untuk baris ke-2 sampai ke-5. Berdasarkan perhitungan seperti pada Tabel 3. *tweet T*1 menghasilkan *bit* 10000. Berdasarkan cara yang sama dengan *T*1, *tweet T*2 menghasilkan *bit* yang sama dengan *T*1 sehingga *tweet T*1 dan *T*2 dinyatakan *collide*. Berdasarkan cara yang sama dengan *T*1, *tweet T*2 menghasilkan *bit* yang sama dengan *T*1 sehingga *tweet T*1 dan *T*2 dinyatakan *collide*.

Selanjutnya, *tweet T*1 dan *T*2 dimasukkan ke dalam himpunan S dan dituliskan dengan S = {*T*1, *T*2}. Kemudian, berdasarkan cara yang sama dengan *tweet T*1 dan *T*2 dilakukan *konversi tweet T*3 , *T*4 dan *T*5 menjadi *signature bit array* seperti pada Tabel 3. Hasil perhitungan *signature bit array* seperti pada Tabel 3., *tweet T*3 dan *T*4 dikonversi menjadi *bit* 10011, sedangkan *tweet T*5 mengalami perubahan menjadi *bit* 11011. *Tweet T*3 dan *T*4 memiliki kesamaan *bit* sehingga dapat diidentifikasikan sebagai *tweet* yang *collide*.

Oleh karena itu, *tweet T*3 dan *T*4 dimasukkan ke dalam himpunan S dan dituliskan dengan S = {*T*3, *T*4}. Disisi lain, *tweet T*5 memiliki *bit* yang berbeda di antara empat *tweet* lain. Oleh karena itu, *tweet T*5 menjadi anggota himpunan baru yang beranggotakan hanya *tweet T*5 dan dituliskan dengan S={*T*5}.

Tabel 3. Perubahan tweet menjadi signature bit array untuk Tweet *T*1dan *T*2

a	b	p	(ax+b) mod p								
			calon (x=0)	presiden (x=1)	pilih (x=2)	prabowo (x=3)	subianto (x=4)	ganjar (x=5)	pranowo (x=6)	anies (x=7)	baswedan (x=8)
5	59	79	59	64	69	74	0	5	10	15	20
2	3	79	3	5	7	9	11	13	15	17	19
1	4	79	4	5	6	7	8	9	10	11	12
2	5	79	5	7	9	11	13	15	17	19	21
3	2	79	2	5	8	11	14	17	20	23	26

Tabel 4. Lanjutan Tabel Diatas

∑ *Distribusi Gaussian*	*Bit*
0,27 + 0,22 + 0,01 + 0,19 + 0,29 + 0,04 + 0,04 + 0,08 + 0,1 = 1,25 (hasil > 0)	1
0,03 + 0,04 + (-0,02) + (-0,09) + (-0,07) + (-0,05) + 0,08 + 0,17 + 0,05= 0,14 (hasil > 0)	1
(-0,01) + 0,04 + (-0,04) + (-0,02) + (-0,04) + (-0,09) + 0,04 + (-0,07) + 0,09= -0,1 (hasil < 0)	0
0,04 + (-0,02) + (-0,09) + (-0,07) + (-0,05) + 0,08 + 0,17 + 0,05 + 0,01= 0,12 (hasil > 0)	1
0,06 + 0,04 + (-0,04) + (-0,07) + (-0,01) + 0,17 + 0,1 + (-0,03) + (-0,11)= 0,11 (hasil < 0)	1

{0: 0.2880236091883922, 1: -0.024169848777495763, 2: 0.06279525312432852,
3: 0.032659590828312046, 4: -0.006691224328133619, 5: 0.04233210769558055,
6: -0.039400575417235295, 7: -0.017233564028582254, 8: -0.042220130349593606,
9: -0.08936045966315218, 10: 0.03914398108603037, 11: -0.07170234223525442,
12: 0.09052813845817041, 13: -0.05163204658465051, 14: -0.012173734906141308,
15: 0.07716699520790953, 16: -0.176148028187156, 17: 0.16965152250075582,
18: 0.14062873024197217, 19: 0.04802977871813637, 20: 0.09673641305445048,
21: 0.00937918747699141, 22: 0.133631235960881 47, 23: -0.029300111055126414,
24: -0.06330658698981516, 25: 0.11891207737106015, 26: -0.11329259753050239,
27: -0.020671579110751685, 28: -0.11206602186878074, 29: 0.06760472660148267,
30: 0.08210657243962496, 31: 0.14351065177709188, 32: -0.09698354786870844,
33: -0.16721212842905664, 34: 0.06387871905214342, 35: 0.012413428905722482,
36: -0.013938422603104137, 37: -0.27270768355981306, 38: 0.06547092066390159,
39: -0.04384727536710721, 40: 0.020250286761466025, 41: 0.1392514890251581,
42: -0.045389597113516394, 43: 0.048822140304740615, 44: -0.04988234916434659,
45: 0.0559406180831005, 46: 0.01710501404458232, 47: -0.05918321771964802,
48: -0.11328770353979561, 49: 0.045794651282216346, 50: 0.01759761463660221,
51: -0.0063063362131217 55, 52: -0.16880331441932298, 53: 0.11196002830143635,
54: -0.10863179982413235, 55: -0.12478786157562766, 56: -0.02052468253224801,
57: -0.07887330829565463, 58: -0.25578677852373516, 59: 0.26920276957040257,
60: -0.09621689790726412, 61: 0.018468532673647203, 62: 0.1597463387800809,
63: 0.05315651816714169, 64: 0.2231175194768114, 65: -0.09317827157524601,
66: 0.011942088550919664, 67: -0.08721818066925108, 68: -0.07379730703896488,
69: 0.00727145566805136, 70: -0.15740847486894377, 71: -0.07446819704174572,
72: -0.1031555914359891, 73: -0.1580989543702044, 74: 0.1900883673188416,
75: -0.10981057345479606, 76: 0.204655041479839, 77: -0.07135366814819553,
78: -0.18273103559191536}

Gambar 5 *Distribusi Gaussian* [5]

c. Langkah ketiga, perhitungan jarak antara *tweet* di dalam himpunan S

Kemiripan antara *tweet* satu dengan *tweet* lain pada himpunan S diukur menggunakan *cosine similarity. Cosine similarity* dapat dihitung dengan persamaan (1) & (2). Perhitungan *cosine similarity*

menggunakan dua tahap. Tahap pertama, *term* unik dari keseluruhan *tweet* dilakukan perhitungan bobot dengan 2.4.1 persamaan (1). Implementasi dari penggunaan persamaan (1) dapat dilihat pada perhitungan jarak antara *tweet* $T1$ dan $T2$ yang berada dalam himpunan S yang sama seperti dijelaskan di Tabel 5

Tabel 5. Pembobotan *tf-idf*

term	*Term frequency* $(tf_{i,j})$		idf_i	$W_{i.j} = tf_{i.j}\ x\ idf_i$	
	T_1	T_2		$w_i t_1$	$w_i t_2$
Calon	1	1	0,30	1.0,30=0,30	0,30
Presiden	1	1	0,30	0,30	0,30
pilih	1	1	0,30	0,30	0,30
Prabowo	1	1	0,30	0,30	0,30
Subianto	1	1	0,30	0,30	0,30
Ganjar	1	1	0,30	0,30	0,30
Pranowo	1	1	0,30	0,30	0,30
Anies	1	1	0,30	0,30	0,30
baswedan	1	1	0,30	0,30	0,30

Tabel 5. menggambarkan perhitungan *TF-IDF* untuk *tweet* T1 dan T2. Bobot *term* wi pada *tweet* pertama (T1) dijelaskan sebagai w1t1. Selanjutnya, dilakukan perhitungan kemiripan antara satu *tweet* dengan *tweet* lainnya. Kemiripan antara *tweet* pertama dengan *tweet* lainnya dihitung menggunakan *cosine similarity* sesuai dengan persamaan 2.4.1. Berdasarkan persamaan 2 tersebut, kemiripan antara *tweet* T1 dan T2 dapat diilustrasikan sebagai berikut:

$$\cos(T_1, T_2) = \frac{(0{,}30\ x\ 0{,}30 + 0{,}30\ x\ 0{,}30 + 0{,}30\ x\ 0{,}30 + 0{,}30\ x\ 0{,}30 + \ 0{,}3x}{\sqrt{0{,}30^2 + \ 0{,}30^2 + 0{,}30^2 + 0{,}30^2 + \ 0{,}30^2 + \ 0{,}30^2 + \ 0{,}30^2 + \ 0{,}30^2 + \ 0{,}30^2}\ x}$$

$$(Lanjutan)\ \frac{0{,}30 + 0{,}30\ x\ 0{,}30 + 0{,}30\ x\ 0{,}30 + 0{,}30\ x\ 0{,}30 + 0{,}30\ x\ 0{,}30)}{\sqrt{0{,}30^2 + \ 0{,}30^2 + 0{,}30^2 + 0{,}30^2 + \ 0{,}30^2 + \ 0{,}30^2 + \ 0{,}30^2 + \ 0{,}30^2 + \ 0{,}30^2}}$$

$$\cos(T_1, T_{12}) = \frac{(0{,}81)}{0{,}9\ x\ 0{,}9}$$

$$\cos(T_1, T_2) = 1$$

Hasil perhitungan di atas menunjukkan bahwa nilai kemiripan kosinus (*cosine similarity*) antara *tweet* $T1$ dan $T2$ adalah 1, melebihi nilai *threshold* yang ditetapkan sebesar 0,5. Oleh karena itu, *tweet* $T1$ dan $T2$ dimasukkan ke dalam klaster yang sama, yaitu klaster 1 (K1 = $\{T1, T2\}$). Selanjutnya, dengan menggunakan metode yang sama, nilai *cosine similarity* antara *tweet* $T3$ dan $T4$ juga bernilai 1, melebihi nilai *threshold*. Sebagai hasilnya, *tweet* $T3$ dan $T4$ ditempatkan dalam klaster yang sama, yaitu klaster 2 (K2 = $\{T3, T4\}$). Sedangkan klaster 3 (K3 = $\{T5\}$) hanya terdiri dari satu elemen, sehingga tidak dilakukan perhitungan *cosine similarity* untuk klaster tersebut.

Dengan menggunakan metode *Document pivot*, ditemukan beberapa klaster pada *time slot* yang sama. Setiap klaster terdiri dari beberapa *tweet* yang memiliki nilai kemiripan melebihi *threshold* yang ditentukan. Metode ini memungkinkan pengelompokan yang efektif dan efisien berdasarkan kemiripan kosinus antara *tweet-tweet* yang dianalisis.

3.3.2. Penghapusan klaster jika jumlah anggota klaster di bawah *threshold*

Dalam penelitian ini, klaster dengan jumlah *tweet* di bawah nilai ambang batas akan dihapus. Dalam konteks ini, ambang batas yang digunakan adalah dua. Artinya, klaster yang memiliki jumlah *tweet* kurang dari atau sama dengan dua tidak akan digunakan sebagai klaster untuk kandidat trending topik, atau dengan kata lain, klaster tersebut akan di *eliminasi*.

3.3.3. Perhitungan skor setiap klaster

Setiap klaster yang terbentuk dari metode *LSH* diatas dilakukan perhitungan skor berdasarkan persamaan 2.4.3 pers (3) dan pers(4) berikut.

Rumus 4

$$\text{P(calon|corpus)} = \frac{N_w + \delta}{(\Sigma_u^n N_u) + \delta n}$$

$$\text{P(claon|corpus)} = \frac{4+0{,}5}{51+0{,}5x17}$$

$$\text{P(claon|corpus)} = \frac{4+0{,}5}{51+8{,}5} = 0{,}075$$

Tabel 6. Perhitungan Corpus

I	*Term*	*Tweet*1	*Tweet*2	*Tweet*3	*Tweet*4	*Tweet*5	P(wi|corpus)	*Freq term*
1	Calon	1	1	1	1		0,075	4
2	Presiden	1	1			1	0,058	3
3	Pilih	1	1				0,042	2
4	Prabowo	1	1	1	1	1	0,092	5
5	Subianto	1	1	1	1		0,075	4
6	Ganjar	1	1	1	1	1	0,092	5
7	Pranowo	1	1	1	1	1	0,092	5
8	Anies	1	1	1	1	1	0,092	5
9	Baswedan	1	1	1	1	1	0,092	5
10	Jakarta			1	1		0,042	2
11	Unggul			1	1		0,042	2
12	Gagal			1	1		0,042	2
13	Tiker			1	1		0,042	2
14	Maju			1	1		0,042	2
15	Poling					1	0,025	1
16	Lihat					1	0,025	1
17	Hasil					1	0,025	1
	Total *term types appearing*							51

Rumus 3

$$Score_c = \sum_{i=1}^{|Docs_c|} \sum_{j=1}^{|Words_i|} exp\left(-p(W_{ij})\right) Score_0$$

$$= \sum_{i=1}^{|2|} \sum_{j=1}^{|9|} exp\left(-p(W_{ij})\right)$$

$$= \left[exp(-p(w_{11})) + exp(-p(w_{12})) + exp(-p(w_{13})) + exp(-p(w_{14})) + exp(-p(w_{15})) + exp(-p(w_{16})) + exp(-p(w_{17})) + exp(-p(w_{18})) + exp(-p(w_{19}))\right] + \left[exp(-p(w_{21})) + exp(-p(w_{22})) + exp(-p(w_{23})) + exp(-p(w_{24})) + exp(-p(w_{25})) + exp(-p(w_{26})) + exp(-p(w_{27})) + exp(-p(w_{28})) + exp(-p(w_{29}))\right]$$

$$= [exp(-0{,}075 * 9) + exp(-0{,}058 * 9) + exp(-0{,}042 * 9) + exp(-0{,}092 * 9) + exp(-0{,}075 * 9) + exp(-0{,}092 * 9) + exp(-0{,}092 * 9) + exp(-0{,}092 * 9) + exp(-0{,}092 * 9)] + [exp(-0{,}075 * 9) + exp(-0{,}058 * 9) + exp(-0{,}042 * 9) + exp(-0{,}092 * 9) + exp(-0{,}075 * 9) + exp(-0{,}092 * 9) + exp(-0{,}092 * 9) + exp(-0{,}092 * 9) + exp(-0{,}092 * 9)] = 8.2767.$$

3.3.4. Perangkingan Topik

Berdasarkan perhitungan skor diatas, ditemukan bahwa klaster ke-0 memiliki skor tertinggi. Klaster tersebut merepresentasikan trending topik yang sedang populer. Dengan menggunakan ekstraksi *term* pada klaster ke-0, berhasil diidentifikasi topik yang menjadi trending topik, yaitu "calon presiden pilih prabowo subianto ganjar pranowo anies baswedan".

3.3.5. Hasil Akhir

Pengujian ini menampilkan hasil akhir dari program yang dibuat. Dapat dilihat pada gambar 6. berikut ini.

Gambar 6. Hasil Metode *Document-Pivot*

Pada pengujian ini mendapatkan hasil klaster 0 yang tertinggi dengan skor 98.0 dengan isi *tweet* "masalah tidak suka jokowi sampe jokowi salahin".

4. KESIMPULAN

Berdasarkan hasil evaluasi dari sistem aplikasi deteksi trending topik pada data tweet dengan topik pemilihan presiden 2024-2025, Kesimpulan ini menunjukkan bahwa aplikasi deteksi trending topik dengan teknologi AI dapat menjadi alat yang bermanfaat dalam analisis data tweet terkait pemilihan presiden. Namun, perlu memperhatikan kebersihan dan kualitas data serta mempertimbangkan keterbatasan teknis yang ada dalam mengakses data dari API *Twitter*.

DAFTAR PUSTAKA

[1] P. Ariwibowo, "Deteksi Trending Topik Terkait Covid-19 Pada Tweet Bahasa Indonesia Menggunakan Metode Maximum Capturing," *Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA) Jakarta-Indonesia*, Oct. 2021.

[2] A. N. Assidyk, E. B. Setiawan, S. Si, I. Kurniawan, S. Pd, and M. Si, "Analisis Perbandingan Pembobotan TF-IDF dan TF-RF pada Trending Topic di Twitter dengan Menggunakan Klasifikasi K-Nearest Neighbor," Aug. 2020.

[3] R. Rafif, E. B. Setiawan, and I. Kurniawan, "Analisis dan implemenasi algoritma C4.5 dan pembobotan TF-IDF untuk menentukan trending topik pada media sosial twitter," Aug. 2020.

[4] A. Karami, M. Lundy, F. Webb, and Y. K. Dwivedi, "Twitter and Research: A Systematic Literature Review through Text Mining," *IEEE Access*, vol. 8, pp. 67698–67717, 2020, doi: 10.1109/ACCESS.2020.2983656.

[5] Indra, E. Winarko, and R. Pulungan, "Trending topics detection of Indonesian tweets using BN-grams and Doc-p," *Journal of King Saud University - Computer and Information Sciences*, vol. 31, no. 2, pp. 266–274, Apr. 2019, doi: 10.1016/j.jksuci.2018.01.005.

[6] A. Futuhul Hadi, D. C. Bagus W, M. Hasan, J. Matematika, F. Matematika dan Ilmu Pengetahuan Alam, and U. Jember Jln Kalimantan, "Seminar Nasional Matematika dan Aplikasinya, 21 Oktober 2017 Surabaya," 2017.

[7] L. M. Aiello *et al.*, "Sensing trending topics in twitter," *IEEE Trans Multimedia*, vol. 15, no. 6, pp. 1268–1282, 2013, doi: 10.1109/TMM.2013.2265080.

[8] M. Pramadani, R. Putra, K. Rizky, and N. Wardani, "Penerapan Text Mining Dalam Menganalisis Kepribadian Pengguna Media Sosial," *JUTIM (Jurnal Teknik Informatika Musirawas)*, vol. 05, Jun. 2020.

[9] F. Fathonah and A. Herliana, "Penerapan Text Mining Analisis Sentimen Mengenai Vaksin Covid - 19 Menggunakan Metode Naïve Bayes," *Jurnal Sains dan Informatika*, vol. 7, no. 2, pp. 155–164, Dec. 2021, doi: 10.34128/jsi.v7i2.331.

[10] A. Pramana and T. #1, "Text Mining Literature Review on Indonesian Social Media," *JEPIN (Jurnal Edukasi dan Penelitian Informatika)* , vol. 7, Aug. 2021.

[11] S. S. Ritonga[1], E. B. Setiawan, and I. Kurniawan, "Analisis Trending Topik Pada Twitter menggunakan Metode Naive Bayes dengan Pembobotan TF-IDF," Apr. 2020.

[12] L. Cahyani, *Aplikasi Text Mining Di Bidang Pendidikan*. 2023. Accessed: Jun. 09, 2023. [Online]. Available: Aplikasi Text Mining Di Bidang Pendidikan. (2023). (n.p.): CV Literasi Nusantara Abadi.

[13] S. Suhartini and B. Prasetya Adhi, "Pemetaan Riset Tentang Deteksi Topik Pada Twitter Dengan Teknik Systematic Literature Review," Jun. 2021, doi: doi.org.10.21009.

[14] F. N. Hikmah, S. Basuki, and Y. Azhar, "Deteksi Topik Tentang Tokoh Publik Politik Menggunakan Latent Dirichlet Allocation," *REPOSITOR*, vol. 2, no. 4, pp. 415–426, 2020, [Online]. Available: https://t.co/4lHGsbrnIK