



# TEXT MINING IN TWITTER: AN ANALYSIS AND MONITORING POLITICAL ISSUES

Sukirno Kasau<sup>\*1</sup>, Yuyun<sup>2</sup>, Syafruddin Syarif<sup>3</sup>

<sup>1,2</sup>STMIK Handayani Makassar

<sup>3</sup>Universitas Hasanuddin

e-mail: <sup>\*1</sup>[createino@gmail.com](mailto:createino@gmail.com), <sup>2</sup>[yuyunwabula@handayani.ac.id](mailto:yuyunwabula@handayani.ac.id), <sup>3</sup>[ssyariftuh376@gmail.com](mailto:ssyariftuh376@gmail.com)

## Abstrak

Media sosial merupakan media yang mempertemukan manusia di dunia maya, salah satu sosial media yang menjadi favorit adalah twitter. Berbagai hal bisa dibahas di twitter, salah satunya adalah tentang politik. Guna mengelola informasi perlu dilakukan analisis teks salah satunya dengan menggunakan *text mining*. Tujuan penelitian ini adalah untuk menggali informasi dari media sosial twitter terkait isu politik. Metode penelitian yang digunakan adalah studi pustaka (*library research*), metode pengumpulan data (*field research*) dan perancangan sistem serta analisis. Pada studi literatur, dilakukan pencarian penelitian-penelitian terkait menggunakan algoritma *Naïve Bayes* dan *K-Nearest Neighbor*. Temuan penelitian ini adalah 2 topik dengan nilai *coherence* terbaik dari 20 topik. Pengujian dari dua topik tersebut menunjukkan tingkat akurasi klasifikasi berdasarkan topik menggunakan metode *K-Nearest Neighbor* lebih tinggi dari *Naïve Bayes*.

**Kata kunci;** Sosial media, *Text Mining*, *Naïve Bayes*, *K-Nearest Neighbor*

## Abstract

*Social media is a medium that brings people together in cyberspace, one of the favorite social media is twitter. Various things can be discussed on twitter, one of which is politics. The purpose of this research is to collect information on twitter on social networks regarding political issues. In order to manage the information, it is necessary to do some text analysis, one of which is to use Text Mining. The research method used is library research, data collection methods (field research) and system design and analysis. In the literature review, a related research search was performed using the Naïve Bayes algorithm and the neighbor K-Nearest. The results of this study were 2 subjects with the best consistency value out of 20 subjects. The issue's two subject test shows that the subject-based classification accuracy level using the K-Nearest Neighbor method is higher than Naïve Bayes.*

**Keywords;** Social Media, *Text Mining*, *Naïve Bayes*, *K-Nearest Neighbor*

## 1. PENDAHULUAN

Media sosial merupakan aplikasi yang memungkinkan pembentukan dan pertukaran konten yang dibuat pengguna[1]. Media memiliki kekuatan yang luar biasa dalam mengatur *mood* dan perilaku orang awam dalam aktivitas sehari-hari [2]. Media sosial sudah jadi hal penting dalam dunia politik, banyaknya isu-isu yang bermunculan di media sosial selalu dikaitkan dengan politik. Dengan pertumbuhan konten *online*, ilmuwan

politik sekarang memiliki lebih banyak informasi yang mereka miliki daripada yang dapat mereka proses dan pahami secara manusiawi. Pemrosesan manual atas informasi semacam itu memakan waktu, mahal dan tidak berskala baik [3]. Dunia politik sendiri telah lama menggunakan teknik data mining untuk menganalisis preferensi politik masyarakat dalam pemilu partai politik seperti pemilu legislatif hingga pemilihan presiden [4].

*Text mining* sendiri banyak digunakan



dalam menggali informasi yang tersembunyi pada data yang sangat banyak berupa opini masyarakat terhadap suatu peristiwa yang di ambil dari sosial media. Menurut data, pengguna sosial media aktif di Indonesia sudah mencapai 3,484 milyar, sementara pengguna internet aktif 4,388 milyar [5].

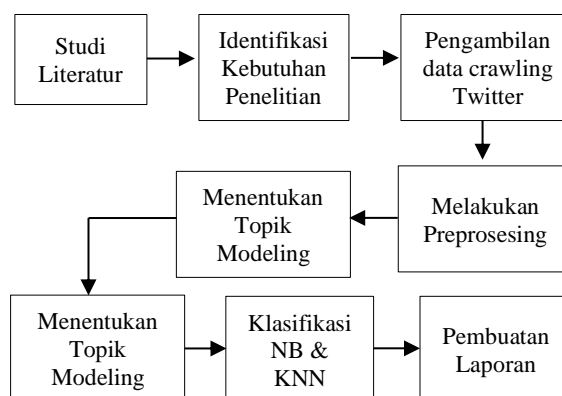
Pada penelitian pilkada DKI tahun 2017, dengan menggunakan metode *Naïve Bayes* dalam menganalisis data sentiment terhadap kandidat calon gubernur di peroleh pada hari yang sama sebesar 80% dan terdapat dugaan bahwa tren yang terjadi di media sosial bisa mempengaruhi hasil pilkada [6]. Sementara pada penelitian yang memprediksi trending di media sosial dengan *text mining* menggunakan metode *K-Nearest Neighbor Algorithm* diperoleh hasil 81,13% dari 60% data [7]. Dari kedua penelitian ini hanya memprediksi trending dan sentimen terhadap kandidat, tetapi tidak diketahui hubungan antara iterasi kata yang satu dengan kata yang lain

Tujuan penelitian ini ialah untuk menggali informasi dari media sosial twitter terkait isu politik lalu dilakukan pemodelan topik untuk mengetahui relasi keterkaitan perbincangan yang sering dibicarakan kemudian dilakukan anotasi terkait pemodelan topik tersebut dan dilakukan pengujian akurasi berdasarkan klasifikasi topik dengan membandingkan dua metode yaitu *Naïve Bayes* dan *K-Nearest Neighbor*. Penelitian ini menggunakan dua metode algoritma klasifikasi ini untuk uji akurasi klasifikasi berdasarkan topik yang dibentuk dan membandingkan hasil akurasi dari dua metode tersebut. Klasifikasi data memiliki tujuan untuk memperkirakan kelas dari suatu objek yang labelnya tidak diketahui pada pembuatan model data [8]. Sehingga dengan mengetahui sehingga kita dapat mengetahui kata-kata tersebut sebagai "data" yang menyandikan informasi tentang posisi politik penulis [9].

## 2. METODE PENELITIAN

Penelitian ini bersifat analisis sehingga dari ruang lingkup masalah dapat dilakukan dengan metode studi pustaka (*library research*), metode pengumpulan data (*field research*) dan perancangan sistem serta analisis.

Tahapan penelitian yang dilakukan adalah pengumpulan data, pembuatan prototipe, evaluasi *prototyping*, pengsosialan sistem, validasi sistem, pengujian sistem, penggunaan sistem. ada tahap ini, penulis mencari dan mempelajari referensi berupa *textbook*, artikel ilmiah maupun jurnal yang berkaitan dengan penelitian. Topik yang akan dibahas antara lain: analisis sentimen, *part-of-speech tagging* pada kalimat Bahasa Indonesia dan metode klasifikasi *Naïve Bayes* and *K-Nearest Neighbor*.



Gambar 1. Tahapan Penelitian

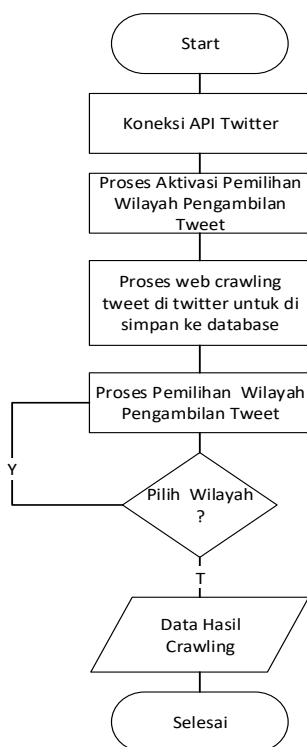
## 3. HASIL DAN PEMBAHASAN

Hasil dari perancangan sistem yang dimulai dengan *Crawling*, *filtering*, kemudian tahapan *preprocessing* Hasil *tweets* yang dihimpun berdasarkan kata kunci dan tanggal pencarian telah dapat direpresentasikan kembali kedalam bentuk *web* berupa data-data hasil proses *preprocessing* [1]. Melalui proses *tokenizing*, *stopword*, dan *stemming* dilanjutkan tahap *feature selection* dengan pemberian bobot dengan TF-IDF. Selanjutnya ditentukan hubungan antara teks yang satu dengan *text* yang lain dengan cara mengelompokan atau meng *cluster* data teks berdasarkan topik tertentu. Setelah dilakukan pengelompokan atau pengklusteran data, kemudian diberi label berdasarkan topik dan dilakukan pengklasifikasian menggunakan metode *Naïve Bayes* dan *K-Nearest Neighbor* sehingga pada proses klasifikasi dihasilkan nilai probabilitas label/kelas yang digunakan untuk pengkategorian berdasarkan topik.

### 1. Data *Crawling* twitter

*Crawling* data dapat diimplementasikan menggunakan bahasa pemrograman PHP dengan modifikasi *Application Programming*

*Interface* (API) [10]. Sehingga memungkinkan kita menambang data pada websitenya. Sebelum melakukan implementasi *crawling* pada twitter kita memerlukan API Key yang teregistrasi. Di twitter sendiri terdapat dua fungsi API yaitu API *search* dan API *stream*, karena penulis mengambil data pada waktu sekarang secara real time penulis menggunakan fungsi API *stream*. *Flowchart* Proses *Crawling* dapat dilihat pada Gambar 2.

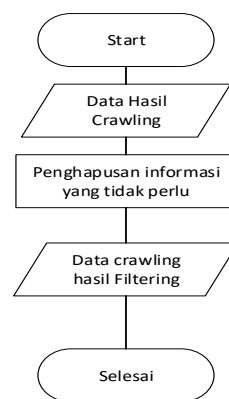


Gambar 2. *Flowchart* Proses *Crawling*

Pada saat sistem dijalankan secara otomatis akan dikoneksikan ke fungsi API twitter, setelah itu akan di pilih wilayah mana yang akan diaktifkan proses web *crawling*. Jika sudah aktif maka proses pengambilan web *crawling* tweet dari *user* akan otomatis dibaca dan disimpan ke *database*. Kemudian dilakukan *filtering* sesuai dengan isu yang akan di tampilkan ke grid hasil dari web *crawling*.

## 2. Filtering

*Filtering* bertujuan untuk mengambil data hasil dari *Crawling* kemudian melakukan proses pembersihan data yang tidak perlu. Proses *filtering* ini mempunyai alur yang ditampilkan pada Gambar 3.



Gambar 3. *Flowchart* tahapan *filtering*

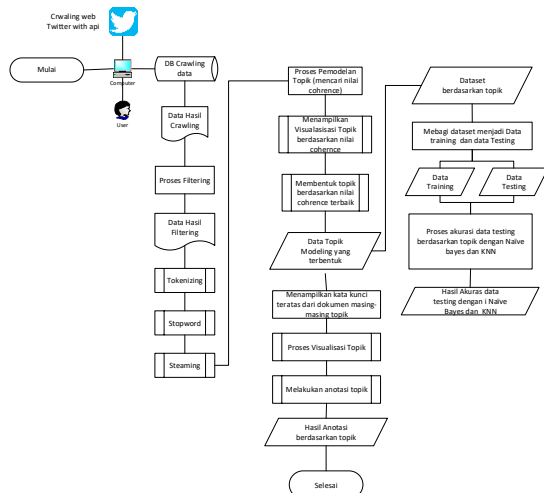
Proses *filtering* ini menghilangkan isi yang tidak dibutuhkan secara otomatis menggunakan metode *Text Cleaning* yaitu menghilangkan *link url*, huruf-huruf aneh dan *emoticon*, kode program. Modul yang dibutuhkan untuk proses *filtering*, lalu dilakukan *coding* untuk membaca data dalam bentuk *csv* untuk dilakukan *filtering*, kemudian dilakukan *coding* sebagai proses *filtering* nya yang berfungsi membersihkan data dari simbol atau *link* dalam dokumen. Contoh Hasil *Filtering* dapat dilihat pada Tabel 1.

Tabel 1. Contoh Hasil *Filtering*

Data <i>Crawling</i>	Hasil <i>Filtering</i>
RT @Candraasmara85: Ramalan Jokowi akan meninggal setelah 3 thn memerintah, ditolak.	ramalan jokowi akan meninggal setelah 3 thn memerintah, ditolak.
Ngancam Tuhan akan ada yg nyembah klo Jkw menang, dit...	ngancam tuhan akan ada yg nyembah klo Jkw menang

## 3. Proses Penentuan Jumlah Topik Model

Dalam menentukan jumlah topik alat ukur yang dipergunakan adalah *coherence value* [11], *coherence* adalah ukuran yang digunakan untuk mengevaluasi topik *modeling* nya, topik yang baik adalah topik dengan nilai *coherence* yang paling tinggi [12]. Semakin tinggi skor untuk jumlah *k* topik tertentu, artinya untuk setiap topik, akan ada lebih banyak kata yang terkait bersama dan topik tersebut akan lebih masuk akal [13]. Untuk menentukan model topik bisa dilihat di Gambar 4.



Gambar 4. Flow Proses Penentuan Topik dan Analisis Hasil Klasifikasi

Berdasarkan Gambar 4 pengguna melakukan *setting* untuk mengambil data *crawling* dari twitter yang dikoneksikan dengan fungsi API yang terdapat di twitter, data hasil *crawling* kemudian dilakukan *preprocessing* yaitu *tokenizing*, *stopword* dan *steaming*. Setelah dilakukan *preprocessing* dilanjutkan dengan pencarian nilai *coherence*, nilai dari *coherence* dari topik tersebut dibuat model visualisasi dalam bentuk grafik. Dari grafik dilihat model topik dengan nilai *coherence* yang paling tinggi untuk dijadikan model dalam membentuk jumlah jumlah topik. Setelah topik terbentuk dilihat kata-kata yang memegang kunci dari setiap kalimat yang saling berelasi untuk dilakukan anotasi. Langkah selanjutnya dihitung akurasi topik dengan cara membagi dataset yang sudah diberi label berdasarkan topik menjadi data *training* dan data *testing* untuk dihitung akurasi klasifikasi teks berdasarkan topik, proses ini bertujuan melihat tingkat akurasi dokumen yang sudah diberi label berdasarkan topik apakah sesuai atau tidak. langkah terakhir dibuat visualisasi topik *modeling* nya kemudian dilakukan anotasi berdasarkan topik yang dibentuk.

#### 4. Menentukan Topik

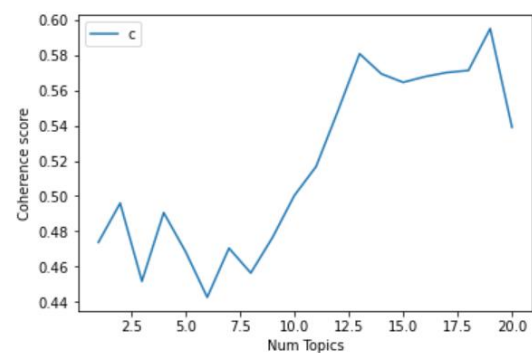
Model yang baik akan menghasilkan topik dengan skor nilai *coherence* yang tinggi [12] dalam percobaan untuk mencari nilai *coherence* bisa dibuat secara acak 5,10,20 pemodelan topik sampai ditemukan nilai *coherence* yang baik, pada penelitian ini melakukan pemodelan topik sebanyak 20 topik.

dari 20 topik tersebut didapatkan dari topik dengan nilai *coherence* terbaik, ini bisa dilihat Tabel 2.

Tabel 2. Nilai *Coherence*

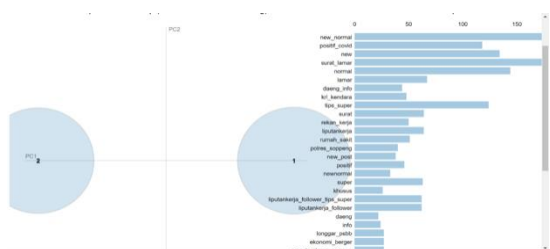
No	Topik	Nilai <i>Coherence</i>
1	Topik 1	0.473726
2	Topik 2	0.495973
3	Topik 3	0.45166
4	Topik 4	0.490594
5	Topik 5	0.468588
6	Topik 6	0.442557
7	Topik 7	0.47044
8	Topik 8	0.45634
9	Topik 9	0.476505
10	Topik 10	0.500256
11	Topik 11	0.516727
12	Topik 12	0.548159
13	Topik 13	0.580747
14	Topik 14	0.569336
15	Topik 15	0.564542
16	Topik 16	0.567751
17	Topik 17	0.57011
18	Topik 18	0.571273
19	Topik 19	0.595036
20	Topik 20	0.539061

Berdasarkan hasil terbaik pada nilai *coherence score* tersebut pada Tabel 2 dan Gambar 5 terdapat dua nilai terbaik, model topik dengan nilai *coherence* terbaik yang dihasilkan tersebut nantinya dijadikan acuan dalam membuat model, sehingga pada penelitian ini terdapat dua topik yang terbaik.



Gambar 5. Visualisasi Nilai *Coherence*

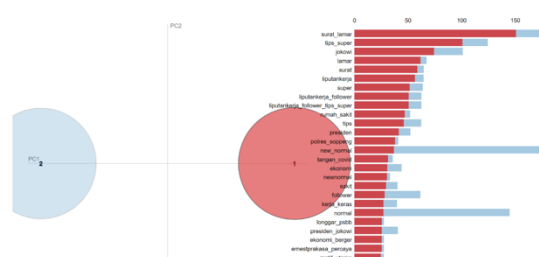
Visualisasi *Topic modelling* pada lingkaran nomor 1 dan 2 yang menunjukkan hal yang paling sering dibicarakan visualisasi dari *topic modeling*, terdapat 2 topik, apabila dipilih topik nomor 1, maka warna lingkaran akan berubah menjadi merah dan pada panel bagian kanan akan langsung menyesuaikan dengan memperlihatkan sejumlah 30 terminologi yang paling relevan terhadap topik tersebut.



Gambar 6. Visualisasi dari *Topic Modeling*

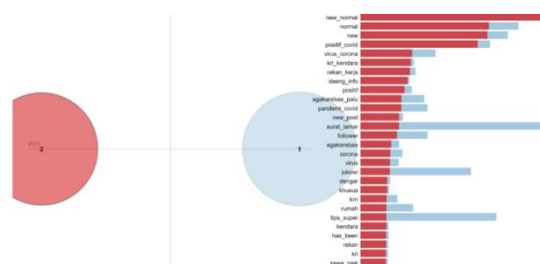
*Bar chart* berwarna biru pada panel bagian kanan mengindikasikan *term frequency* secara keseluruhan. Sedangkan *bar chart* berwarna merah menunjukkan estimasi *term frequency* pada topik yang dimaksud. Berikut disajikan visualisasi *Topic modelling* pada lingkaran nomor 1 dan 2 yang menunjukkan hal yang paling sering di bicarakan.

Topik 1 memiliki model dengan pembahasan pekerjaan, rumah sakit jokowi. Tidak hanya itu, topik ini juga membahas tentang ekonomi, covid, PSBB. Maka kelompok topik 1 isunya tentang bagaimana pemerintah dalam hal ini Presiden Jokowi menyediakan lapangan kerja di tengah pandemi covid dan perekonomian yang menurun.



Gambar 7. Relasi Topik 1

Topik 2 memiliki model dengan pembahasan *new\_normal*, *virus\_corona*, kendaraan tidak hanya itu, topik ini juga membahas tentang jokowi, krl, sewa\_ojek. Maka kelompok topik 2 isunya tentang bagaimana pemerintah membuat peraturan terkait berkendara di era *new normal*.



Gambar 8. Relasi Topik 2

### 5. Proses Klasifikasi

Dari 5137 dataset hasil *preprocessing* kemudian dilakukan pengelompokan/kluster menjadi 2 topik utama, yang mana topik 1 terkluster sebanyak 3098 data dan topik 2 terkluster sebanyak 1229 data, Langkah selanjutnya adalah menguji akurasi klasifikasi berdasarkan topik atau kluster tersebut menggunakan *Naïve Bayes* dan *K-Nearest Neighbor*.

### 6. Naïve Bayes

*Naïve Bayes* merupakan metode *classifier* berdasarkan probabilitas dan Teorema Bayes dengan asumsi bahwa setiap variabel X bersifat bebas (*independence*) [6].

Tabel 3. Jumlah Data Berdasarkan Topik

Topik	Data Testing	Hasil Klasifikasi	Ketidak sesuaian
1	1171	1124	47
2	371	418	0

Dalam hal ini modul-modul yang dibutuhkan untuk menghitung akurasi, program untuk memisahkan data *training* dan data *testing*, kemudian dilakukan coding untuk memprediksi akurasi dari data *testing* dan *training*. Hasil dari *kode* program tersebut dapat didapatkan nilai yang sama dengan perhitungan manual yaitu sebesar 0,97 atau 97 %, ini bisa dilihat di Gambar 7.

	precision	recall	f1-score	support
1.0	0.96	0.99	0.98	1149
2.0	0.98	0.89	0.93	393
accuracy			0.97	1542
macro avg	0.97	0.94	0.95	1542
weighted avg	0.97	0.97	0.97	1542

Gambar 9. *Naïve Bayes* Akurasi Score

## 7. K-Nearest Neighbor

Tabel 4. Hasil Klasifikasi KNN

Topik	Data Testing	Hasil Klasifikasi	Ketidak sesuaian
1	1171	1175	0
2	371	367	4

Berikut ini merupakan perhitungan nilai akurasi klasifikasi dengan persamaan sebagai berikut:  
 Akurasi =  $((TP+TN)/(TP+TN+FP+FN))$   
 \*100%

TP= Jumlah Topik 1 dari Data Testing

TN= Jumlah Topik 2 dari Data Testing

FP= Jumlah Kesalahan Klasifikasi Topik 1

FN= Jumlah Kesalahan Klasifikasi Topik 2

Akurasi =  $((1171+371)/ (1171+371+0+4))$   
 \*100 %

=  $(1542/1546) *100\%$

=  $0,99*100\%$

= 99 %

Dilakukan *coding* yang dibutuhkan untuk menghitung akurasi, baris kemudian *kode* program untuk memanggil data hasil pemodelan topik, serta dilakukan *kode* program untuk memisahkan data *training* dan data *testing*, serta *kode* program untuk memprediksi akurasi dari data *testing* dan *training*. Hasil dari kode program ini didapatkan hasil klasifikasi topik menggunakan K-Nearest Neighbor Sebesar 0,99 atau 99%, ini bisa dilihat di Gambar 8.

	precision	recall	f1-score	support
1.0	1.00	0.99	1.00	1175
2.0	0.98	1.00	0.99	367
accuracy			0.99	1542
macro avg	0.99	0.99	0.99	1542
weighted avg	0.99	0.99	0.99	1542

Gambar 10. KNN Akurasi Score

Berdasarkan pengujian kedua algoritma tersebut didapatkan hasil akurasi sangat baik, namun terjadi perbedaan antara algoritma *Naïve Bayes* dan KNN, akan tetapi perbedaan tidak terlalu signifikan hanya 0,02 % yang mana hasil KNN lebih besar di angka 0,99 atau 99% sementara *Naïve Bayes* diangka 0,97 atau 97 %. Algoritma *Naïve Bayes* dan KNN sangat baik dalam menganalisis akurasi dalam pemodelan topik modeling, ini dibuktikan dari hasil topik *modeling* dengan melihat tingkat koherence tertinggi hasil sama baik dengan akurasi dari *Naïve Bayes* dan KNN.

Postingan *text user* di media sosial twitter dapat digunakan untuk menganalisis isu politik. hasil analisis topik modeling didapatkan dua isu yang paling sering dibicarakan yaitu menurunnya perekonomian ditengah pandemi *covid* menyebabkan lapangan kerja berkurang dan perlunya pemerintah membuat aturan berkendara di era New Normal. Hasil pengujian dari dua topik isu tersebut menunjukkan tingkat akurasi klasifikasi berdasarkan topik menggunakan metode *K-Nearest Neighbor* lebih tinggi dari *naïve bayes*, hal ini dapat dilihat dari hasil *K-Nearest Neighbor* sebesar 0,99 atau 99 % sedangkan menggunakan *Naïve Bayes* sebesar 0,97 atau 97 %. Berdasarkan hasil pengujian menunjukkan *text* yang dibentuk berdasarkan topik menghasilkan akurasi yang sangat baik.

## 4. KESIMPULAN

Berdasarkan hasil analisis yang telah dilakukan dalam klasifikasi isu topik dapat disimpulkan bahwa postingan *text user* di media sosial twitter dapat digunakan untuk menganalisis isu politik. hasil analisis topik modeling didapatkan dua isu yang paling sering dibicarakan yaitu menurunnya perekonomian ditengah pandemi *covid* menyebabkan lapangan kerja berkurang dan perlunya pemerintah membuat aturan berkendara di era New Normal. Hasil pengujian dari dua topik isu tersebut menunjukkan tingkat akurasi klasifikasi berdasarkan topik menggunakan metode *K-Nearest Neighbor* lebih tinggi dari *naïve bayes*, hal ini dapat dilihat dari hasil *K-Nearest Neighbor* sebesar 0,99 atau 99 % sedangkan menggunakan *Naïve Bayes* sebesar 0,97 atau 97 %. Berdasarkan hasil pengujian menunjukkan *text* yang dibentuk berdasarkan topik menghasilkan akurasi yang sangat baik.

## 5. SARAN

Perlu pengembangan sistem dengan dengan mengambil data dari berbagai sumber media sosial dan memodifikasi pada tahapan preprocessing dengan menambah beberapa tahapan

## UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Yayasan Pendidikan tamalatea yang telah memberi dukungan financial terhadap penelitian ini.

## DAFTAR PUSTAKA

- [1] A. M. Kaplan and M. Haenlein, "Users of the world, unite! The challenges and opportunities of Social Media," *Bus. Horiz.*, vol. 53, no. 1, pp. 59–68, 2010, doi: 10.1016/j.bushor.2009.09.003.
- [2] S. V. Shri Bharathi and A. Geetha, "Determination of news biasedness using content sentiment analysis algorithm," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 16, no. 2, pp. 882–889, 2019, doi: 10.11591/ijeecs.v16.i2.pp882-889.
- [3] L. Bustikova, D. S. Siroky, S. Alashri, and S. Alzahrani, "Predicting Partisan Responsiveness: A Probabilistic Text Mining Time-Series Approach," *Polit. Anal.*, vol. 28, no. 1, pp. 47–64, 2020, doi: 10.1017/pan.2019.18.
- [4] D. Suyanto, *Data Mining Untuk Klasifikasi dan Klusterisasi Data*. Bandung: Informatika, 2019.
- [5] Y. Pratomo, "Ini Cara Cambridge Gunakan Data Facebook untuk Menangkan Trump," *kompas.com*, 2018. <https://tekno.kompas.com/read/2018/03/27/19110007/ini-cara-cambridge-gunakan-data-facebook-untuk-menangkan-trump> (accessed Mar. 30, 2020).
- [6] A. F. Hadi, D. Bagus, and M. Hasan, "Text Mining Pada Media Sosial Twitter Studi Kasus: Masa Tenang Pilkada Dki 2017 Putaran 2," *Semin. Nas. Mat. dan Apl. 21 Oktober 2017 Surabaya, Univ. Airlangga*, 2017, [Online]. Available: [http://math.fst.unair.ac.id/wp-content/uploads/2017/10/50-Dimas-Bagus\\_Sistem-Informasi\\_.pdf](http://math.fst.unair.ac.id/wp-content/uploads/2017/10/50-Dimas-Bagus_Sistem-Informasi_.pdf).
- [7] S. Syarif, Anwar, and Dewiani, "Trending topic prediction by optimizing K-nearest neighbor algorithm," *Proc. 2017 4th Int. Conf. Comput. Appl. Inf. Process. Technol. CAIPT 2017*, vol. 2018-Janua, pp. 1–4, 2018, doi: 10.1109/CAIPT.2017.8320711.
- [8] H. Februariyanti, M. Firmansyah, J. S. Wibowo, and M. S. Utomo, "ANALISIS SENTIMEN TANGGAPAN TERHADAP APLIKASI LAYANAN INFORMASI PENGINAPAN MENGGUNAKAN METODE KLASIFIKASI NAIVE BAYES," *semanTIK*, vol. 6, no. 2, pp. 115–124.
- [9] F. Hjorth, R. Klemmensen, S. Hobolt, M. E. Hansen, and P. Kurrild-Klitgaard, "Computers, coders, and voters: Comparing automated methods for estimating party positions," *Res. Polit.*, vol. 2, no. 2, p. 2053168015580476, 2015.
- [10] B. R. Aditya, "Penggunaan Web Crawler Untuk Menghimpun Tweets dengan Metode Pre-Processing Text Mining," *J. INFOTEL - Inform. Telekomun. Elektron.*, vol. 7, no. 2, p. 93, 2015, doi: 10.20895/infotel.v7i2.35.
- [11] J. Eka Sembodo, E. Budi Setiawan, and Z. Abdurahman Baizal, "Data Crawling Otomatis pada Twitter," no. August, pp. 11–16, 2016, doi: 10.21108/indosc.2016.111.
- [12] A. Nurlayli and M. A. Nasichuddin, "Topik Modeling Penelitian Dosen Jptei Uny Pada Google Scholar Menggunakan Latent Dirichlet Allocation," *Elinvo (Electronics, Informatics, Vocat. Educ.)*, vol. 4, no. 2, pp. 154–161, 2019, doi: 10.21831/elinvo.v4i2.28254.
- [13] K. Stevens, P. Kegelmeyer, D. Andrzejewski, and D. Buttler, "Exploring topic coherence over many models and many topics," *EMNLP-CoNLL 2012 - 2012 Jt. Conf. Empir. Methods Nat. Lang. Process. Comput. Nat. Lang. Learn. Proc. Conf.*, no. July, pp. 952–961, 2012.

