

Perbandingan Fitur Ekstraksi Untuk Klasifikasi Emosi Pada Sosial Media

Riska Dwi Handayani¹⁾, Kusri²⁾, Hanif Al Fatta³⁾

^{1, 2, 3)} Program Studi S2 Teknik Informatika, Universitas Amikom Yogyakarta

¹⁾ riska.1116@students.amikom.ac.id, ²⁾ kusri@amikom.ac.id, ³⁾ hanif.a@amikom.ac.id

ABSTRACT

Emotions are complex conscious experiences characterizing mental states, such as excitement, anger, love, fear, and so on, as part of important human nature. Nowadays, many people express themselves as a reflection of their personality using social media. Social Media grows and becomes a method for social interaction and information sharing. Based on that, researchers tried to use social media data to classify someone's emotions. Emotional detection of text from social media is a field of research that is gaining a high interest, especially for the sake of emotional analysis. To be able to classify such emotions, researchers use comparative feature comparison and algorithms classification. The comparison of features in this research is the extraction features TF-IDF and N-gram which are then classified using Naïve bayes algorithm. However, before the extraction feature is applied, there is a pre-processing text technique using several methods: Case Folding, Stopword Removal, and Stemming. Based on this research, techniques of extraction features in this research generating the highest accuracy value after the classification method is the TF-IDF with an accuracy value of 80%, 98% for the highest value of precision in measurement of "pleasure" emotion, 99% for the highest recall value is in "happy" emotion, and 95% for the highest F1-score value in "pleasure" emotion.

Keywords: Emotion, TF-IDF, N-Gram, Naïve Bayes.

I. PENDAHULUAN

Emosi adalah pengalaman sadar yang kompleks yang mencirikan suatu keadaan mental seperti kegembiraan, kemarahan, cinta, ketakutan dan sebagainya, yang menjadi penting sebagai bagian dari sifat manusia. Saat ini banyak orang yang mengekspresikan diri mereka sebagai cerminan atas kepribadianya menggunakan sosial media. Media sosial tumbuh dan menjadi metode untuk interaksi sosial dan distribusi informasi (Dandannavar et al., 2018). Facebook adalah sosial media yang mempunyai fitur update status yang memungkinkan pengguna melakukan posting gambar, video, dan terutama melalui teks yang akan tampil pada timeline pengguna terkait (Deshpande & Rao, 2018). Data tekstual menjadi bentuk komunikasi yang paling banyak digunakan karena menawarkan banyak karakteristik yang menjadikannya pilihan terbaik untuk melakukan data analisis, untuk emosi AI termasuk deteksi depresi.

Deteksi emosi pada teks dapat dianggap sebagai masalah klasifikasi pada konsep pemrosesan bahasa alami/NLP dan *Machine learning*. Deteksi emosi pada teks dari media sosial adalah bidang penelitian yang memperoleh minat tinggi terutama untuk kepentingan analisis emosi. Menurut hasil survey yang telah dilakukan pada tahun 2014-2018 terdapat 6 dari 10 paper yang melakukan penelitian tentang deteksi emosi pada teks (Al-Saaqa et al., 2018). Terdapat beberapa metode yang dapat digunakan untuk melakukan klasifikasi pada teks, diantaranya *Support Vector Machine*, *Naïve Bayes*, *K-Nearest Neighbour*, *Multinomial Naïve Bayes*, *Multi Task Regression*, *Algoritma Incremental Regression*. Sedangkan untuk metode fitur ekstraksi juga terdapat beberapa yang sering digunakan, diantaranya TF-IDF, N-Gram, *Word Embedding*, NLP. Berdasarkan latar belakang tersebut

peneliti mencoba melakukan perbandingan fitur ekstraksi antara TF-IDF dan N-Gram yang kemudian diklasifikasikan dengan metode Naïve Bayes. Peneliti menggunakan dataset dari media sosial facebook yang telah dikumpulkan oleh peneliti sebelumnya (Rohman et al., 2019).

II. TINJAUAN PUSTAKA

2.1. Literatur Review

Penelitian yang dilakukan oleh (Indrayuni & Wahyudi, 2015) tentang penerapan character n-gram untuk sentiment analysis review hotel menggunakan metode algoritma naïve bayes. Dalam penelitian ini metode fitur ekstraksi yang digunakan adalah metode n-gram. Dan untuk metode klasifikasi yang digunakan adalah metode algoritma naïve bayes. Hasil dari penelitian ini adalah Algoritma Naive Bayes yang merupakan algoritma paling sederhana yang terbukti menghasilkan nilai akurasi hingga 82.67%. Penerapan character N-gram pada tahap preprocessing algoritma Naive Bayes membuat nilai akurasi meningkat hingga 2%, yaitu menjadi 84.67%. Dari uraian diatas, dapat disimpulkan bahwa penerapan character N-gram pada tahap preprocessing algoritma Naive Bayes untuk klasifikasi sentimen review hotel dapat meningkatkan nilai rata-rata akurasi sehingga secara keseluruhan diperoleh kesimpulan bahwa penerapan character. N-gram pada algoritma Naive Bayes merupakan model pengujian algoritma yang memiliki unjuk kerja lebih baik jika dibandingkan penggunaan algoritma *Naïve bayes* saja untuk permasalahan klasifikasi sentimen review hotel.

Selanjutnya adalah penelitian yang dilakukan oleh (Olivita, 2016) tentang perbandingan klasifikasi tugas akhir mahasiswa jurusan teknik informatika menggunakan metode *naïve bayes* classifier dan k-nearest neighbor. Fitur ekstraksi yang digunakan pada penelitian ini adalah unigram dan *Augmented Term Frequency*. Dalam penelitian ini juga dilakukan perbandingan metode klasifikasi yang digunakan. Hasil dari penelitian ini adalah metode *Naïve bayes* Classifier dapat melakukan klasifikasi terhadap tugas akhir lebih baik daripada K-Nearest Neighbor dengan akurasi sebesar 87%. Akurasi data antara metode *Naïve bayes* Classifier dan K-Nearest Neighbor pada seratus data tugas akhir memiliki selisih data yang kecil. Hal ini disebabkan karena jumlah data dari masing-masing kelasnya tidak seimbang, sehingga mengakibatkan kurangnya keefektifan data latih pada salah satu kelas.

Selanjutnya penelitian yang dilakukan oleh (Rachmat C & Lukito, 2016) tentang klasifikasi sentiment komentar politik dari facebook page menggunakan naïve bayes. Dalam penelitian ini fitur ekstraksi yang digunakan adalah pembobotan kata TF-IDF. Dan metode klasifikasi yang digunakan adalah metode algoritma naïve bayes. Hasil dari penelitian ini adalah metode Naive Bayes memiliki tingkat akurasi klasifikasi sentimen mencapai lebih dari 83%.

2.2. N-Gram

Menurut (Gencosman, Ozmutlu, & Ozmutlu, 2014) N-gram didefinisikan sebagai sub-urutan n karakter dari kata diberikan. Misalnya, "*mountain*" dapat diwakili dengan *character* n-gram yang ditunjukkan pada Tabel 1.

Tabel 1. Contoh Penerapan N-Gram

n	Character n-gram samples
2-Grams (n=2)	mo-ou-un-nt-ta-ai-in
3-Grams (n=3)	mou-oun-unt-nta-tai-ain
4-Grams (n=4)	moun-ount-unta-ntai-tain

Dapat disimpulkan bahwa untuk string berukuran n akan dimiliki n unigram dan $n+1$ bigram, $n+1$ trigram, $n+1$ quadgram dan seterusnya. Penggunaan N-gram untuk matching kata memiliki keuntungan sehingga dapat diterapkan pada recovery pada input karakter ASCII yang terkena noise, interpretasi kode pos, information retrieval dan berbagai aplikasi dalam pemrosesan bahasa alami. Keuntungan N-gram dalam matching string adalah berdasarkan karakteristik N-gram sebagai bagian dari suatu string, sehingga kesalahan pada sebagian string hanya akan berakibat perbedaan pada sebagian N-gram. Sebagai contoh jika N-gram dari dua string dibandingkan, kemudian kita menghitung cacah N-gram yang sama dari dua string tersebut maka akan didapatkan nilai similaritas atau kemiripan dua string tersebut yang bersifat resistan terhadap kesalahan tekstual. Kemiripan antara kata JOKO dengan JOKI (ada perbedaan 1 huruf), dapat diukur derajat kesamaan dengan cara menghitung berapa buah N-gram yang diambil dari dua kata tersebut yang bernilai sama, yaitu : JOKO: _J, JO, OK,KO,O_ , JOKI : _J, JO, OK,KI, I_ kesamaan :3 Sementara antara kata JOKO dengan JONI (ada perbedaan 2 huruf), nilai kesamaan adalah : JOKO: _J, JO, OK,KO,O_ , JONI : _J, JO, ON,NLI_ kesamaan : 2 Sehingga dapat disimpulkan bahwa kemiripan atau kesamaan antara JOKO-JOKI dari pada antara JOKO-JONI.

2.3. TF-IDF

Metode TF-IDF merupakan fitur ekstraksi teks dengan cara mencari bobot suatu kata (term) pada sebuah dokumen (Robertson, 2004). Dalam TF-IDF terdapat dua rumus yang digabungkan yaitu menghitung frekuensi kemunculan kata di sebuah dokumen tertentu (TF) dan perhitungan invers document frequency (IDF) menghitung kuantitas term yang berfungsi sebagai ukuran tingkat signifikansi suatu term dalam sebuah dokumen. Menurut (Fauzi et al., 2015) persamaan untuk mencari nilai TF dan IDF dapat dilihat pada persamaan 1 dan 2 berikut ini.

$$TF(d,t) = f(d,t) \quad (1)$$

$$IDF(t) = 1 + \log(Nd/df(t)) \quad (2)$$

dimana:

$f(d,t)$: frekuensi kemunculan term t pada dokumen d

Nd : jumlah seluruh dokumen

$df(t)$: jumlah dokumen yang terdapat term t

Sehingga untuk menemukan nilai TF-IDF dapat digunakan rumus pada persamaan 3.

$$TF - IDF = TF(d,t).IDF(t) \quad (3)$$

2.4. Naive Bayes

Naive bayes adalah salah satu algoritma Machine Learning yang memanfaatkan konsep probabilitas. Cara menghitung klasifikasinya menghitung nilai probabilitas $p(h|x)$, yaitu probabilitas kelas h jika diketahui suatu b , berdasarkan algoritma NaiveBayes. Proses klasifikasi dapat dilakukan dengan menentukan nilai suatu kelas $h \in H$ dari suatu dokumen $x \in X$ dengan $H = \{h_1, h_2, h_3, \dots, h_p\}$ dan $X = \{x_1, x_2, x_3, \dots, x_q\}$. Penentuan kelas dalam klasifikasi dokumen tersebut dilakukan dengan cara memilih nilai maximum dari $p(h|x)$, berdasarkan distribusi probabilitas $P = \{p(h|x) \mid h \in H \text{ dan } x \in X\}$.

Suatu dokumen x ke i dapat dipresentasikan sebagai vector dan nilai-nilai fitur yang ada pada dokumen tersebut sehingga $x = \{f_1, f_2, f_3, \dots, f_n\}$. Nilai dari elemen tiap vector $F = \{f_1, f_2, f_3, \dots, f_n\}$ merupakan nilai untuk fitur f_j pada himpunan fitur dengan f_j merupakan g nilai dari fitur ke j pada dokumen x ke i . Berdasarkan algoritma NaiveBayes berikut ini merupakan persamaan perhitungan nilai dari probabilitas $p(h|x)$:

$$\text{Posterior} = \frac{\text{Likelihood} \cdot \text{Prior}}{\text{Evidence}} \quad (4)$$

$$p(h|x) = \frac{p(x|h) \cdot p(h)}{p(x)} \quad (5)$$

Keterangan:

$p(h|x)$ = Nilai posterior atau probabilitas kata h dari kelas x

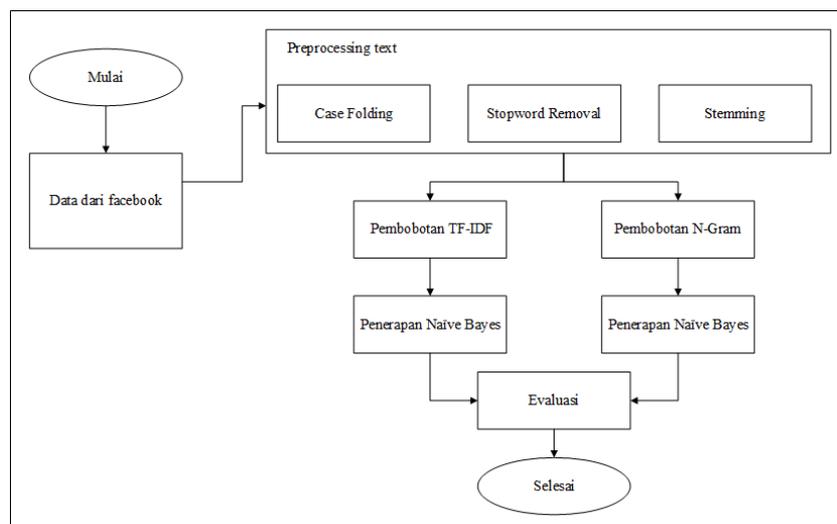
$p(x|h)$ = Nilai likelihood atau probabilitas kemunculan kelas x untuk kata h

$p(h)$ = Nilai prior atau probabilitas kata h

$p(x)$ = Nilai evidence atau probabilitas kelas x

III. METODE PENELITIAN

Metode penelitian yang dilakukan oleh peneliti untuk mendapatkan hasil terbaik dapat dilihat pada Gambar 1.



Gambar 1. Metode Penelitian

Peneliti mendapatkan dataset update status *facebook* yang sudah terlabeli. Selanjutnya dataset tersebut dibagi menjadi 2 kelompok data yaitu data latih dan data uji. Sebelum dilakukannya pembobotan dengan fitur maka data harus melewati teknik *preprocessing* teks terlebih dahulu. Teknik *preprocessing* teks yang dilakukan oleh peneliti diantaranya *case foling*, *Stopword removal* dan *Stemming*. Setelah selesai pada teknik *preprocessing* selanjutnya adalah teknik fitur ekstraksi pembobotan kata. Dalam fitur ini semua data baik data latih maupun data uji diterapkan teknik ekstraksi untuk mendapatkan bobot dokumen update status. Kemudian setelah selesai, maka dilanjutkan penerapan metode algoritma naïve bayes. Untuk tahap terakhir adalah melakukan evaluasi dari hasil klasifikasi yang dilakukan oleh *naïve bayes* tersebut.

IV. HASIL DAN PEMBAHASAN

4.1. Dataset

Dataset yang digunakan oleh peneliti adalah dataset yang berasal dari update status facebook yang telah dikumpulkan oleh peneliti sebelumnya. Data yang didapat berjumlah 550 dan hanya teks Bahasa Indonesia saja dan selanjutnya dibagi menjadi data

latih dan data uji dengan perbandingan 60:40. Sehingga untuk data latih mendapat 330 data dan data uji berjumlah 220 data. Data yang didapatkan sudah dilakukan pelabelan oleh author. Contoh dataset update status facebook dapat dilihat pada Tabel 1.

Tabel 1. Contoh dataset update status

Update status	Keyword	Label
Mentari selalu menampakkan sinarnya	Mentari	sedih
Hari itu aku juga inget, aku pernah berjanji pada diriku sendiri.	Hari itu	sedih
Aku harus membuat mereka bangga	Bangga	Takut
Jangan menilai seseorang dari sampul	menilai	marah
semoga engkau jadi pribadi yang mandiri, menabung menjadi pribadi yang tangguh...	Tangguh	Bahagia

4.2. Preprocessing Teks

Selanjutnya masuk ke dalam tahap preprocessing teks. Dalam tahap ini terdapat 3 teknik yang perlu dilakukan.

a. Case folding

Pada tahap ini, semua huruf akan diubah menjadi huruf kecil. Contoh hasil penerapan *case folding* dapat dilihat pada Tabel 2.

Tabel 2. Contoh penerapan *case folding*

Update status	Case folding
Mentari selalu menampakkan sinarnya	mentari selalu menampakkan sinarnya
Hari itu aku juga inget, aku pernah berjanji pada diriku sendiri.	hari itu aku juga inget, aku pernah berjanji pada diriku sendiri.
Aku harus membuat mereka bangga	aku harus membuat mereka bangga
Jangan menilai seseorang dari sampul	jangnan menilai seseorang dari sampul
semoga engkau jadi pribadi yang mandiri, menabung menjadi pribadi yang tangguh...	semoga engkau jadi pribadi yang mandiri, menabung menjadi pribadi yang tangguh...

b. Stopword Removal

Stopword didefinisikan sebagai term yang tidak berhubungan dengan subyek utama dari database meskipun kata tersebut sering kali hadir di dalam dokumen dan kata yang dianggap tidak dapat memberikan pengaruh dalam menentukan suatu kategori emosi. Contoh penerapan *Stopword removal* dapat dilihat di Tabel 3.

Tabel 3. Contoh penerapan *Stopword removal*

Update status	Stopword removal
Mentari selalu menampakkan sinarnya	mentari selalu menampakkan sinarnya
Hari itu aku juga inget, aku pernah berjanji pada diriku sendiri.	hari inget, pernah berjanji sendiri.
Aku harus membuat mereka bangga	harus membuat mereka bangga
Jangan menilai seseorang dari sampul	menilai seseorang sampul
semoga engkau jadi pribadi yang mandiri, menabung menjadi pribadi yang tangguh...	semoga engkau pribadi mandiri, menabung menjadi pribadi tangguh...

c. *Stemming*

Stemming merupakan tahap untuk mentransformasi kata-kata yang terdapat dalam suatu dokumen ke kata-kata akarnya (*root word*) dengan menggunakan aturan-aturan tertentu. Dengan menggunakan *Stemming* dapat mengurangi variasi kata yang sebenarnya memiliki kata dasar yang sama. Salah satu algoritma *Stemming* yaitu Algoritma Nazief dan Adriani. Contoh penerapan *Stemming* dapat dilihat pada table 4.

Tabel 4. Contoh penerapan *Stemming*

Update status	<i>Stemming</i>
Mentari selalu menampakkan sinarnya	mentari selalu tampak sinar
Hari itu aku juga inget, aku pernah berjanji pada diriku sendiri.	Inget pernah janji sendiri.
Aku harus membuat mereka bangga	harus buat mereka bangga
Jangan menilai seseorang dari sampul	nilai orang sampul
semoga engkau jadi pribadi yang mandiri, menabung menjadi pribadi yang tangguh...	semoga pribadi mandiri, nabung jadi pribadi tangguh

4.3. N-Gram

N-gram didefinisikan sebagai sub-urutan n karakter dari kata diberikan. Berikut adalah contoh hasil perhitungan N-Gram.

Tabel 5. Hasil n-gram dokumen

Dokumen	<i>Stemming</i>	Bigram
Latih	mentari	_m; me; en; nt; ta;ar;ri;i_
	selalu	_s; se; el; la; al;lu;u;i_
	tampak	_t; ta; am; mp; pa;ak;k;i_
	sinar	_s; si; in; na; ar;i_
Uji	harus	_h;ha;ar;ru;us;s_
	buat	_b;bu;ua;at;t_
	mereka	_m;me;er;re;ek;ka;a_
	bangga	_b;ba;an;ng;gg;ga;a_

Tabel 6. Hasil Frekuensi N-gram

Dokumen Latih			Dokumen Uji		
No	Bigram	f	No	Bigram	f
1	_m	1	14	u_	1
2	me	1	15	_t	1
3	en	1	16	am	1
4	nt	1	17	mp	1
5	ta	2	18	pa	1
6	ar	2	19	ak	1
7	ri	1	20	k_	1
8	i_	1	21	_s	1
9	_s	1	22	si	1
10	se	1	23	in	1
11	el	1	24	na	1
12	la	1	25	r_	1
13	lu	1			
			1	_h	1
			2	ha	1
			3	ar	1
			4	ru	1
			5	us	1
			6	s_	1
			7	_b	2
			8	bu	1
			9	ua	1
			10	at	1
			11	t_	1
			12	_m	1
			13	me	1
			14	er	1
			15	re	1
			16	ek	1
			17	ka	1
			18	a_	2
			19	ba	1
			20	an	1
			21	ng	1
			22	gg	1
			23	ga	1

Setelah model n-gram dokumen dari masing-masing dokumen yang telah diketahui bahasanya (disebut juga sampel bahasa) didapatkan dari dokumen pelatihan bahasa dan model n-gram dokumen untuk pengujian didapatkan dari dokumen inputan pengujian (dokumen uji atau data uji), maka dapat dilakukan pengklasifikasian dengan membandingkan kedua model n-gram dokumen tersebut.

4.4. TF-IDF

TF-IDF merupakan fitur ekstraksi teks dengan cara mencari bobot suatu kata (term) pada sebuah dokumen. Berikut adalah contoh hasil perhitungan dari TF-IDF.

Tabel 7. Hasil perhitungan TF-IDF

Stemming	tf		df	D/df	IDF	W (bobot)	
	D1	D2				D1	D2
mentari	1		1	2	0,30103	0,30103	
selalu	1		1	2	0,30103	0,30103	
tampak	1		1	2	0,30103	0,30103	
sinar	1		1	2	0,30103	0,30103	
harus		1	1	2	0,30103		0,30103
buat		1	1	2	0,30103		0,30103
mereka		1	1	2	0,30103		0,30103
bangga		1	1	2	0,30103		0,30103
Jumlah tiap dokumen						1,20412	1,20412

Dari Tabel tersebut telah didapatkan jumlah nilai tiap dokumen. Selanjutnya dari nilai tersebut dibandingkan antara dokumen latih dengan dokumen yang diuji mana yang lebih mendekati dengan nilai dari dokumen latih.

4.5. Hasil Pengujian

Pengujian dalam penelitian ini dilakukan dengan menguji sebanyak 550 data update status bahasa Indonesia dengan dibagi menjadi 330 data latih dan 220 data uji dan kategori klasifikasi yang ditentukan adalah emosi senang, sedih, takut, marah, jijik, bahagia, kaget. Setelah dilakukannya preprocessing teks maka selanjutnya adalah teknik ekstraksi teks dimana dalam penelitian ini terdapat 2 ekstraksi teks yang akan digunakan yang nantinya akan dibandingkan hasil terbaiknya.

Berdasarkan pengujian yang telah dilakukan dengan menggunakan algoritma *naïve bayes* yang dijalankan pada bahasa pemrograman python, maka akurasi yang didapatkan dengan metode *naïve bayes* dan TF-IDF adalah sebesar 80%. Sedangkan metode *naïve bayes* dan N-Gram menghasilkan nilai akurasi sebesar 76,4%. Dalam penelitian ini juga menampilkan nilai lain selain nilai akurasi yaitu nilai presisi, recall dan f1-score . Recall dan precision adalah dua perhitungan yang banyak digunakan untuk mengukur kinerja dari sistem/metode yang digunakan. *Recall* adalah tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi. *Precision* adalah seberapa besar tingkat ketepatan antara informasi yang diminta oleh pengguna dengan jawaban yang diberikan oleh sistem. Sedangkan F1-Score adalah perbandingan anatara recall dan presisi yang dibobotkan. Untuk nilai presisi dari masing-masing emosi dapat dilihat pada Tabel 8.

Tabel 8. Hasil presisi masing-masing emosi

Emosi	<i>Naïve bayes</i> dan TF-IDF	<i>Naïve bayes</i> dan N-Gram
senang	0.98	1.00
takut	0	0
kaget	0	0
jijik	0	0
bahagia	0.68	0.62
sedih	0.89	0.94
marah	0	0

Tabel 9. Hasil *recall* masing-masing emosi

Emosi	<i>Naïve bayes</i> dan TF-IDF	<i>Naïve bayes</i> dan N-Gram
senang	0.91	0.83
takut	0	0
kaget	0	0
jijik	0	0
bahagia	0.99	1
sedih	0.98	0.89
marah	0	0

Tabel 10. Hasil F1-score masing-masing emosi

Emosi	<i>Naïve bayes</i> dan TF-IDF	<i>Naïve bayes</i> dan N-Gram
senang	0.95	0.91
takut	0	0
kaget	0	0
jijik	0	0
bahagia	0.81	0.76
sedih	0.93	0.92
marah	0	0

Selain hasil yang telah ditampilkan diatas, dalam penelitian ini juga menghasilkan waktu yang dibutuhkan untuk memproses data latih dan data uji dari masing-masing metode fitur ekstraksi. Waktu yang dibutuhkan untuk memproses data tersebut dapat dilihat pada Tabel 11.

Tabel 11. Waktu yang dibutuhkan fitur ekstraksi untuk memproses data

Jenis Data	TF-IDF	N-Gram
Data latih	0.005990 s	0.007597 s
Data Uji	0.001000 s	0.006582 s

V. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan mengenai perbandingan fitur ekstraksi untuk klasifikasi emosi pada sosial media menggunakan *naïve bayes* dapat disimpulkan bahwa diantara metode TF-IDF dan *Naïve bayes* dengan N-Gram dan *Naïve bayes* yang menghasilkan nilai akurasi tertinggi adalah metode TF-IDF dan *Naïve Bayes*. Nilai akurasi yang dihasilkan sebesar 80% , nilai presisi tertinggi pada emosi senang yaitu 0.98, nilai recall tertinggi pada emosi bahagia 0.99, nilai f1-score tertinggi pada emosi senang 0.95. Sedangkan untuk metode N-Gram dan *Naïve bayes* menghasilkan nilai akurasi sebesar 76,4%, nilai presisi tertinggi pada emosi senang yaitu 1.00, nilai recall tertinggi pada emosi bahagia yaitu 1 dan nilai f1-score tertinggi pada emosi sedih yaitu 0.92.

5.2 Saran

Berdasarkan penelitian yang telah dilakukan ada kekurangan yang didapat yaitu jumlah data yang didapatkan kurang banyak dan kurang bervariasi, sehingga penulis memberikan saran untuk penelitian selanjutnya agar menambah dataset yang lebih banyak lagi sehingga nilai akurasi bisa menjadi lebih baik dan nilai presisi, recall dan f1-score bisa lebih merata.

DAFTAR PUSTAKA

- Al-Saaqa, S., Abdel-Nabi, H., & Awajan, A. (2018). A Survey of Textual Emotion Detection. *2018 8th International Conference on Computer Science and Information Technology, CSIT 2018*, 136–142. <https://doi.org/10.1109/CSIT.2018.8486405>
- C, A. R., & Lukito, Y. (2017). Deteksi Komentar Spam Bahasa Indonesia Pada Instagram Menggunakan Naive Bayes. *Jurnal ULTIMATICS*, 9(1), 50–58. <https://doi.org/10.31937/ti.v9i1.564>
- Dandannavar, P. S., Mangalwede, S. R., & Kulkarni, P. M. (2018). Social Media Text - A Source for Personality Prediction. *Proceedings of the International Conference on Computational Techniques, Electronics and Mechanical Systems, CTEMS 2018*, 62–65. <https://doi.org/10.1109/CTEMS.2018.8769304>
- Deshpande, M., & Rao, V. (2018). Depression detection using emotion artificial intelligence. *Proceedings of the International Conference on Intelligent Sustainable Systems, ICISS 2017*, (Iciss), 858–862. <https://doi.org/10.1109/ISS1.2017.8389299>
- Devita, R. N., Herwanto, H. W., & Wibawa, A. P. (2018). Perbandingan Kinerja Metode Naive Bayes dan K-Nearest Neighbor untuk Klasifikasi Artikel Berbahasa Indonesia. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 5(4), 427. <https://doi.org/10.25126/jtiik.201854773>
- Fauzi, M. A., Arifin, A. Z., Kom, S., Kom, M., Yuniarti, A., Kom, S., & Sc, M. C. (2015). Term Weighting Berbasis Indeks Buku Dan Kelas Untuk Perangkingan Dokumen Berbahasa Arab. *Lontar Komputer : Jurnal Ilmiah Teknologi Informasi*, 5(2), 110–117. <https://doi.org/ISSN:2088-1541>
- Gencosman, B. C., Ozmutlu, H. C., & Ozmutlu, S. (2014). Character n-gram application for automatic new topic identification. In *Information Processing and Management* (Vol. 50). <https://doi.org/10.1016/j.ipm.2014.06.005>
- Indrayuni, E., & Wahyudi, M. (2015). Penerapan Character N-Gram Untuk Sentiment Review Hotel Menggunakan Algoritma Naive Bayes. *Konfrensi Nasional Ilmu Pengetahuan Dan Teknologi (KNIT)*, 88–93.

- Olivita, D. (2016). Perbandingan Klasifikasi Tugas Akhir Mahasiswa Jurusan Teknik Informatika Menggunakan Metode *Naïve bayes* Classifier dan K-Nearest Neighbor. *Sains*, 14(1), 79–85. <https://doi.org/10.1002/mame.201200226>
- Rachmat C, A., & Lukito, Y. (2016). Klasifikasi Sentimen Komentar Politik dari Facebook Page Menggunakan Naive Bayes. *Jurnal Informatika Dan Sistem Informasi Universitas Ciputra*, 02(02), 26–34. <https://doi.org/10.1080/10408398.2013.809690>
- Robertson, S. (2004). Understanding inverse document frequency: On theoretical arguments for IDF. *Journal of Documentation*, 60(5), 503–520. <https://doi.org/10.1108/00220410410560582>
- Rohman, A. N., Utami, E., & Raharjo, S. (2019). Deteksi Kondisi Emosi pada Media Sosial Menggunakan Pendekatan Leksikon dan Natural Language Processing. *Eksplora Informatika*, 9(1), 70–76. <https://doi.org/10.30864/eksplora.v9i1.277>
- Santosa, B. (2007). *Data Mining Teknik Pemanfaatan Data Untuk Keperluan Bisnis*. Yogyakarta: Graha Ilmu.
- Widyawati, & Sutanto. (2019). *Perbandingan Algoritma Naïve bayes Dan Support Vector Machine (Svm)*. 3(2), 178–194.