Deteksi Berita Rumor pada Sosial Media Twitter Menggunakan Metode Naïve Bayes Multinomial dengan Pembobotan TF-IDF

Tugas Akhir

diajukan untuk memenuhi salah satu syarat

memperoleh gelar sarjana

dari Program Studi Informatika

Fakultas Informatika

Universitas Telkom

NIM 1301162748

Refka Muhammad Furqon

Program Studi Sarjana Informatika

Fakultas Informatika Universitas

Telkom

Bandung

2020

LEMBAR PENGESAHAN

Deteksi Berita Rumor pada Sosial Media Twitter Menggunakan Metode Naïve Bayes Multinomial dengan Pembobotan TF-IDF

Detection of Rumor News on Twitter Social Media using Naïve Bayes Multinomial Method with TF-IDF Weighted

NIM : 1301162748
Refka Muhammad Furqon

Tugas akhir ini telah diterima dan disahkan untuk memenuhi sebagian syarat memperoleh gelar pada Program Studi Sarjana Informatika
Fakultas Informatika
Universitas Telkom

Bandung, 23 Juni 2020
Menyetujui

Pembimbing I,

Erwin Budi Setiawan, S.Si., M.T
NIP: 00760045

Ketua Program Studi
Sarjana Informatika

Niken Dwi Wahyu Cahyani, S.T., M.Kom , Ph.d.
NIP: 00750052

LEMBAR PERNYATAAN

Dengan ini saya Refka Muhammad Furqon menyatakan sesungguhnya bahwa Tugas Akhir saya dengan judul Deteksi Berita *Rumor* pada Sosial Media Twitter Menggunakan Metode *Naïve Bayes Multinomial* dengan Pembobotan TF-IDF beserta dengan seluruh isinya adalah merupakan hasil karya sendiri, dan saya tidak melakukan penjiplakan yang tidak sesuai dengan etika keilmuan yang belaku dalam masyarakat keilmuan. Saya siap menanggung resiko/sanksi yang diberikan jika di kemudian hari ditemukan pelanggaran terhadap etika keilmuan dalam buku TA atau jika ada klaim dari pihak lain terhadap keaslian karya,

Bandung, 23 Juni 2020

Yang Menyatakan

Refka Muhammad Furqon

Deteksi Berita Rumor pada Sosial Media Twitter Menggunakan Metode Naïve Bayes Multinomial dengan Pembobotan TF-IDF

Refka Muhammad Furqon[1], Erwin Budi Setiawan, S.Si., M.T.[2]

[1,2,3]Fakultas Informatika, Universitas Telkom, Bandung
[4]Divisi Digital Service PT Telekomunikasi Indonesia
[1]refkamf@students.telkomuniversity.ac.id, [2]erwinbudisetiawan@telkomuniversity.ac.id,

Abstrak

Media sosial adalah sarana tempat untuk berkomunikasi dan bertukar informasi sesama manusia, dan salah satu media sosial yang digunakan adalah Twitter. Ada beberapa informasi yang kita dapatkan belum tentu benar adanya, ada berita yang kurang tepat ataupun tidak tepat kebenarannya bisa disebut berita *rumor*. Berita *rumor* sudah sangat sering kita lihat dimedia sosial, banyak sekali pihak yang dirugikan dengan adanya berita *rumor*. Pada penelitian tugas akhir ini, penulis membangun sistem untuk mendeteksi berita *rumor* pada twitter. Diperlukan nilai bobot pada *tweet* yang diambil dari berita *rumor* yang disebarkan oleh beberapa orang di Twitter, *Term Frequency Inverse Document Frequency* (TF-IDF) pembobotan yang digunakan penulis dalam penelitian ini dengan fitur ekstraksi *TF-IDF N-gram*. Klasifikasi data menggunakan metode *Naïve bayes Multinomial* untuk memprediksi hasil akurasi dari penelitian ini. Hasil performansi yang diperoleh menggunakan TF-IDFdalam pengujian sebesar 78,53% dengan data uji sebesar 10%. Sedangkan untuk klasifikasi tanpa menggunakan *TF-IDF* sebesar 77,98% dengan data uji sebesar 10%.

Keyword: Rumor, TF-IDF, Naïve Bayes Multinomial, Non Rumor, Sosial Media

Abstract

Social media is a place to communicate and exchange information about fellow human beings, and one of the social media used is Twitter. There is some information that we get is not necessarily true, there is a lack of precise or improper news of the truth can be called rumors news. Rumor news has been very often seen in social media, so many parties were harmed by rumors of news. In this final research assignment, the author builds a system to detect rumors of news on Twitter. It takes a weighted value to a tweet taken from rumor news spread by some people on Twitter, the weighting Term Frequency Inverse Document Frequency (TF-IDF) that the author used in this study. The data classification uses the Naïve Bayes Classifier method to predict the accuracy results of this study. Performance results obtained using TF-IDF in tests of 78,53% with a test data of 10%. As for classification without using TF-IDF of 77,98% with test data of 10%.

Keyword: Rumor, TF-IDF, Naïve bayes Multinomial, Non Rumor, Media Social

1. Pendahuluan

Perkembangan teknologi telah mencapai pada titik tertinggi dengan membuat komunikasi menjadi sangat mudah dengan adanya sosial media, seperti Facebook, Instagram, WhatsApp, Twitter, Line dan masih banyak lagi. Media sosial merupakan alat komunikasi yang memerlukan internet sehingga tidak perlu ada batas jarak. Twitter adalah salah satu sosial media yang paling cepat berkembang di internet, sebuah jaringan *microblogging* yaitu suatu bentuk blog yang membatasi ukuran setiap post-nya, yang memberikan fasilitas bagi pengguna untuk dapat menuliskan pesan dalam twitter update hanya berisi 140 karakter[1]. Pengguna dari sosial media twitter mencapai 300 juta pengguna kurang lebih terdapat 700 juta *tweet* perhari., dengan rata-rata 7.500 *tweet* per detik. Dengan jumlah yang sangat besar Maka twitter bisa disebut sebagai gudang data yang terdapat milyaran data dari berbagai negara[2]. Dengan informasi yang begitu banyak terdapat beberapa informasi yang mengandung positif dan negatif, selain untuk berinteraksi twitter dapat membuat penggunanya menyalahgunakan informasi (*rumor*, penipuan, *hoax*, pencemaran nama baik).

Rumor adalah hipotesis yang ditawarkan dengan tidak adanya informasi yang dapat diverifikasi mengenai keadaan yang tidak pasti bagi orang-orang, kemudian terjadinya kecemasan yang tidak terkontrol yang dihasilkan dari ketidakpastian ini[3]. Banyak sekali berita-berita diluar sana yang simpang siur atau berita *rumor* yang meresahkan masyarakat dengan informasi yang tidak bisa dipastikan kebenarannya. Maka diperlukan mengenali ciri-ciri berita *rumor* pada media sosial twitter dengan metode yang dapat membaca ciri-ciri tersebut.

Naïve Bayes Multinomial merupakan model multinomial mengambil jumlah kata yang muncul pada sebuah dokumen, pada model multinomial terdapat beberapa dokumen yang terdiri dari beberapa kata yang di asumsikan panjang dokumen tidak bergantuk pada kelasnya[4]. Dengan menggunakan asumsi dari *Naïve Bayes* kemungkinan tiap kejadian kata dalam sebuah dokumen tidak terpengaruhi dengan konteks kata dalam dokumen[4].

Dalam penelitian ini penulis menggunakan pembobotan *Term frequency Inverse Document Frequency* (TFIDF) pada deteksi berita *rumor* ini, TFIDF adalah metode statistik numerik yang memungkinkan penentuan bobot untuk setiap istilah (atau kata) dalam setiap dokumen[5]. Sedangkan klasifikasi menggunakan *Naïve Bayes Multinomial*, diharapkan dari penelitian yang dilakukan dapat membantu mengatasi masalah berita yang tidak benar.

Dalam latar belakang yang telah disampaikan di atas, masalah yang dapat diambil dalam tugas akhir ini yaitu berita *rumor*, kemudian tingkat akurasi dari klasifikasi *Naïve Bayes.* Kemudian batasan pekerjaan dalam tugas akhir ini adalah perilaku pengguna yang dimasukan ke dalam fitur *user-based* dan *tweet-based* dalam Bahasa Indonesia. Tujuan dalam penelitian ini untuk mengimplementasikan pembobotan TF-IDF dan metode NB untuk mendeteksi berita *rumor* pada Twitter, dengan mengetahui seberapa besar hasil akurasi yang dihasilkan oleh sistem yang dibuat.

2. Studi Terkait

2.1. Media Sosial

Media sosial telah digunakan banyak orang untuk kepentingan masing-masing, di mana orang membuat konten, berbagi, penunjuk dan jaringan di tingkat yang sangat besar. Sehingga dapat digunakan organisasi ataupun individu dengan berbagai kepentingannya. Dengan begitu kita bisa bertemu dengan orang yang baru saja dikenal[1].

2.2. *Rumor*

Rumor sering sekali kita jumpai pada berita-berita yang beredar diluar sana, sehingga menimbulkan kecemasan kepada orang-orang yang menerima berita tersebut[3]. Pada umumnya *rumor* menyebar dari mulut ke mulut dan informasi tidak diketahui secara jelas, berbeda dengan *hoax* yang sebenarnya bisa berisi fakta namun direkayasa[3]. Menurut statistik dari Dewan *anti-rumor*, sekitar 88 *rumor* di dokumentasikan dalam periode dari Januari sampai April 2017. Ini sejumlah besar *rumor* dalam jangka waktu yang singkat dapat memiliki hasil yang merusak bagi individu maupun masyarakat[15].

2.3. *Pre-processing Data*

Pre-processing atau praproses data merupakan proses untuk mempersiapkan data mentah sebelum dilakukan proses lain. Pada umumnya, preproses data dilakukan dengan cara mengeliminasi data yang tidak sesuai atau mengubah data menjadi bentuk yang lebih mudah diproses oleh sistem[6]. Macam-macam metode *preprocessing* meliputi *case folding*, *tokenizing*, *cleaning*, *stopword*, *stemming*. Pada penelitian ini *preprocessing* yang digunakan *case folding*, *cleaning*, normalisasi, *stopword*, dan *stemming*[13].

2.4. *N-Gram* Model

Ide penggunaan *N-gram* sudah banyak diterapkan dalam berbagai masalah seperti prediksi kata, koreksi ejaan, pengenal suara, koreksi kata terjemahan dan pencarian *string*[11]. Salah satu keuntungan dari metode *N-gram* ini adalah bahwa bahasa bersifat independen. Dalam koreksi ejaan, *N-gram* merupakan urutan sebanyak N huruf dalam sebuah kata atau *string*[11]. Dengan menggunakan dataset twitter yang tersedia untuk umum, kami melatih model *classifier* kami menggunakan *frekuensi n-gram* dan *term frequency-inverse document frequency* (TF-IDF) sebagai fitur[12].

Tabel 1. Contoh Ngram

Data	Jokowi maju pilpres lagi
Uni-gram	|Jokowi| |maju| |pilpres| |lagi|
Bi-gram	|Jokowi maju| |pilpres lagi|
Tri-gram	|Jokowi maju pilpres| |maju pilpres lagi|

2.5. TF-IDF

Metode *Term Frequency Invers Document Frequency* (TF-IDF) merupakan metode yang digunakan menentukan seberapa jauh keterhubungan kata (term) terhadap dokumen dengan memberikan bobot setiap kata[7]. Pembobotan ini menggabungkan dua konsep yaitu frekuensi kemunculan sebuah kata di dalam sebuah dokumen dan inverse frekuensi dokumen yang mengandung kata tersebut[7].

Dalam perhitungan bobot menggunakan TF-IDF, dihitung terlebih dahulu nilai TF perkata dengan bobot masing-masing kata adalah 1 jika f(x) x = t dan 0 jika x tidak sama dengan t. Untuk mencari nilai TF dapat diketahui dengan rumus dibawah ini[16].

$$TFt,d = \sum_{x} \in \text{[illegible]}(x) \quad (1)$$

Sedangkan nilai IDF diformulasikan pada Persamaan (2)[7].

$$IDF \text{[illegible]} = \log\left(\frac{D}{\text{[illegible]}}\right) \quad (2)$$

IDF adalah nilai IDF dari setiap kata yang akan di cari, td adalah jumlah keseluruhan dokumen yang ada, df jumlah kemuculan kata pada semua dokumen.

$$\text{[illegible]}dt = tf_{\text{[illegible]}} * I\text{[illegible]} \quad (3)$$

[illegible]*t* adalah hasil kali nilai TF dengan IDF, bobot term t terhadap dokumen d, [illegible]*t* adalah jumlah kemunculan term t dalam dokumen d dan i[illegible]*t* adalah nilai Inverse Document Frequency.[7]

2.6. *Naïve Bayes Multinomial*

Naive bayes merupakan salah satu metode klasifikasi yang berakar pada teorema Bayes. Pada *Naive Bayes*, setiap atribut dalam data dianggap independen antara satu dan lainnya[8]. Keuntungan penggunaan *Naive Bayes* adalah bahwa metode ini hanya membutuhkan jumlah data pelatihan (*Training Data*) yang kecil untuk menentukan estimasi paremeter yang diperlukan dalam proses pengklasifikasian. *Naive Bayes* sering bekerja jauh lebih baik dalam kebanyakan situasi dunia nyata yang kompleks dari pada yang diharapkan[9].

Ada beberapa macam model pada *Naïve Bayes*, pada penelitian ini penulis menggunakan *Naïve Bayes Multinomial*. Model *multinomial* memperhitungkan frekuensi setiap kata yang muncul pada dokumen. Misal terdapat dokumen d dan himpunan kelas c. Untuk memperhitungkan kelas dari dokumen d, maka dapat dihitung dengan rumus[14] :

$$P(c|\text{term dokumen d}) = P(c) \times P(t1|c) \times P(t2|c) \times P(t3|c) \times \ldots \times P(tn|c) \quad (1)$$

P(c|term dokumen d) adalah probabilitas suatu dokumen termasuk kelas c. tn yaitu kata dokumen d ke-n. P(tn|c) merupakan probabilitas kata ke-n dengan diketahui kelas c. Sedangkan P(c) adalah probabilitas *prior* dari kelas c. Untuk mencari nilai P(c) dapat dihitung dengan rumus[14]:

$$P(c) = \frac{N\text{[illegible]}}{N} \quad (2)$$

P(c) adalah jumlah kelas c pada seluruh dokumen, dan N jumlah seluruh dokumen. Sementara rumus Multinomial yang digunakan dengan pembobotan kata TF-IDF adalah sebagai berikut:

$$\text{[illegible]}(t\text{[illegible]}|\text{[illegible]}) = \frac{\text{[illegible]} + 1}{(\sum \text{[illegible]}' \in VW' ct) + B'} \quad (3)$$

P(tn|c) adalah probabilitas kata ke-n dengan diketahui kelas c. Wct adalah nilai pembobotan TF-IDF atau di kategorikan (1/0) yaitu jumlah kata unik (nilai 1 jika tidak dikan dengan tf) pada seluruh dokumen yang berada di kategori c. (1') yaitu jumlah kata unik (nilai 1 jika tidak dikan dengan tf) pada seluruh dokumen.

2.7. Confusion Matrix

Confusion Matrix adalah tahap analisis dan evaluasi terhadap performansi sistem yang dirancang. Performansi diukur dengan nilai akurasi, *precision*, *recall* dan F1. *Confusion matrix* adalah suatu metode yang digunakan untuk melakukan perhitungan akurasi pada konsep *data mining*[10]. Tabel *Confusion matrix* dapat dilihat dalam Tabel:

Tabel 2. Contoh *Confusion matrix*

	Actual Class Yes (Rumor)	Actual Class No (Non Rumor)
Predicted Class Yes (Rumor)	True Positive (TP)	False Positive (FP)
Predicted Class No (Non Rumor)	False Negative (FN)	True Negative (TN)

True Positive (TP) merupakan hasil klasifikasi dan hasil *actual* sama-sama *rumor*. *False Positive* (FP) merupakan hasil klasifikasi *rumor* dan hasil *actual non rumor*. *True Negative* (TN) merupakan hasil klasifikasi dan hasil *actual* sama-sama *non rumor*. *False Negative* (FN) merupakan hasil klasifikasi *non rumor* dan hasil *actual rumor*[10].

1. Akurasi
   Akurasi adalah tingkat kedekatan antara nilai prediksi dengan nilai sesungguhnya. Rumus dari perhitungan Akurasi yaitu :

$$\text{Akurasi} = \frac{TP + TN}{TP + TN + F\text{□□} + FN} \quad (1)$$

2. *Precision*
   Precision adalah tingkat ketepatan antara informasi yang diminta oleh pengguna dengan jawaban yang diberikan oleh sistem. Rumus dari perhitungan Precision yaitu:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (2)$$

3. *Recall*
   Recall adalah jumlah pengguna yang dengan benar diklasifikasikan dalam sebuah kelas dibagi dengan jumlah total pengguna dalam kelas tersebut. Rumus dari perhitungan Recall yaitu:

$$\textit{Recall} = \frac{TP}{TP + FN} \quad (3)$$

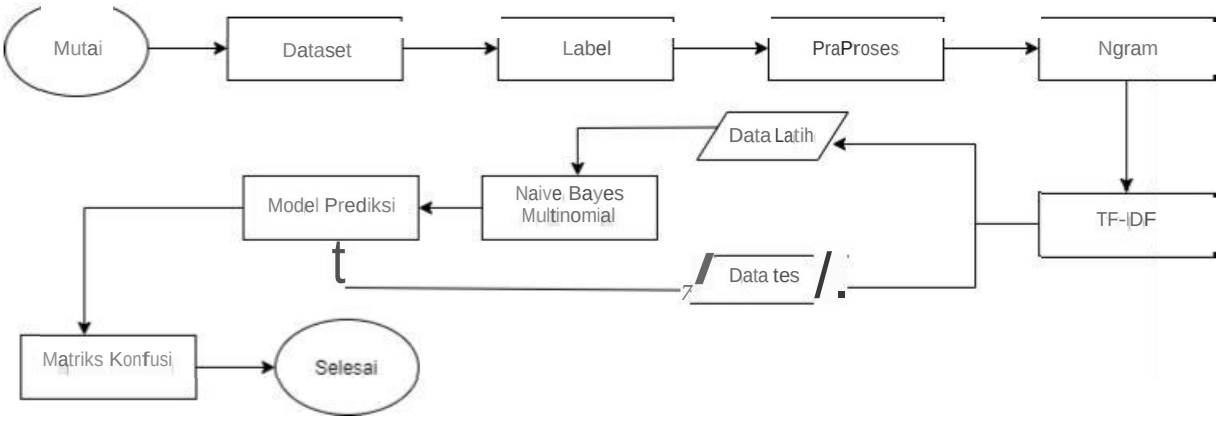
4. F1
   F1 adalah nilai rata rata dari *precision* dan *recall* yang di bobotkan. Rumus dari perhitungan F1 yaitu:

$$\text{F1} = \frac{2 \times \text{□□ □□□□□□ □ } \times}{\text{□ □□ □□}} \quad (4)$$

3. Sistem yang Dibangun

3.1. *Flowchart*

Berdasarkan pada gambar 1, untuk menghasilkan performansi dari sistem akan melalui beberapa tahap. Pertama dilakukan pengumpulan data dengan *crawling* yaitu pengumpulan data dari twitter lalu data disortir sampai akhirnya dikelompokan. Kemudian data dibagi menjadi dua yaitu data *training* dan data *testing* kemudian data dimasukkan ke tahap *preprocessing* untuk di olah agar tidak ada data yang duplikat. Kemudian data di beri bobot nilai TF-IDF untuk mendapatkan kata-kata yang sering muncul. Berikutnya data training yang telah di olah akan di proses pada klasifikasi dari *Naïve Bayes* dan data *testing* untuk melakukan prediksi untuk menguji model yang telah dibuat dan mendapatkan hasil akurasi yang diperoleh sistem.



Gambar 1. Model Deteksi Rumor dengan Naive Bayes Multinomial

3.2. *Crawling*

Pada penelitian ini penulis mengambil data dengan teknik *Crawling* pada API twitter yang telah dikembangkan oleh Jaka Eka Sembodo dkk, dengan menggunakan *Keyword*, pengambilan data maksimal sebanyak 200 *tweet* dalam 1 kali *crawling*. Cara pengambilan data dengan melihat *trending* twitter yang diperkirakan adanya berita *rumor*, *Crawling* dimulai pada September 2019 – Januari 2020. Data terkumpul sebanyak 47449 *tweet*.

Tabel 3. Daftar *Keyword*

Keyword	Jumlah Tweet
#Corona	7.449
#PelantikanJokowiAmin	10.000
#CongratsJokowiMarufAmin	10.000
#KabinetKerjaSemangatBaru	10.000
#PelantikanPresiden	10.000

3.3. *Pre-Processing*

Pre-processing adalah adalah tahapan dimana aplikasi melakukan seleksi data yang akan diproses pada setiap dokumen. Proses *preprocessing* ini meliputi (1) *case folding*, (2) *cleaning*, (3) normalisasi, (4) *stopword* dan (5) *Stemming*. *Case folding* adalah proses manipulasi *case-sensitive*, semua input teks data akan diubah menjadi huruf kecil/*lower-case*. *Cleaning* adalah suatu proses menghilangkan karakter, *url*, angka, dan *hastag*. Normalisasi adalah suatu proses pengubahan seluruh kata yang disingkat berdasarkan kamus bahasa yang akan disesuaikan dengan aturan penulisan dan kamus bahasa yang benar. Sedangkan *stopword removal* adalah proses penghapusan kata yang tidak relevan dalam teks. *Stemming* adalah pengubahan kata menjadi kata dasar.

Tabel 4. Contoh *preprocessing*

Preprocessing	Sebelum	Sesudah
Casefolding	Membka jalur baru demi kelancaran perekonomian warga desa Dengan dana Desa masyarakat semakin meningkat perekonomiannya https://t.co/8pEwFJsqyc	membka jalur baru demi kelancaran ekonomian warga desa dengan dana desa masyarakat semakin meningkat perekonomiannya https://t.co/8pEwFJsqyc
Cleaning	membka jalur baru demi kelancaran ekonomian warga desa dengan dana desa masyarakat semakin meningkat perekonomiannya https://t.co/8pEwFJsqyc	membka jalur baru demi kelancaran ekonomian warga desa dengan dana desa masyarakat semakin meningkat perekonomiannya
Normalisasi	membka jalur baru demi kelancaran ekonomian warga desa dengan dana desa masyarakat semakin meningkat perekonomiannya	membuka jalur baru demi kelancaran ekonomian warga desa dengan dana desa masyarakat semakin meningkat perekonomiannya
StopWord	membuka jalur baru demi kelancaran ekonomian warga desa dengan dana desa masyarakat semakin meningkat perekonomiannya	jalur baru kelancaran ekonomian warga desa dengan dana masyarakat semakin meningkat perekonomian
Stemming	jalur baru kelancaran ekonomian warga desa dengan dana masyarakat semakin meningkat perekonomian	jalur baru lancar ekonomi warga desa dana masyarakat tingkat perekonomian

3.4. TFIDF

Pada proses pembobotan tiap data yang telah di proses dengan *preprocessing* akan diberikan nilai atau bobot dengan menggunakan metode TF-IDF. Pembobotan ini berpengaruh untuk mengukur kata dari suatu dokumen. Pembobotan ini menggunakan *Ngram* yang terdiri dari *Unigram*, *Bigram*, *Trigram* kemudian di kombinasikan menjadi *Unigram* dan *Bigram*, *Bigram* dan *Trigram*, terakhir digabungkan semuanya menjadi *Unigram*, *Bigram*,*Trigram*.

4. Evaluasi

4.1 Hasil Pengujian dan Skenario

Pada penelitian ini menggunakan data hasil *crawling* dengan dataset berjumlah 47.449 *tweet* yang telah di *label rumor* dan *non rumor*. Pembagian dataset bisa dilihat di tabel 5 dibawah.

Tabel 5. Pembagian Dataset

NO	Data Training	Data test
1	90%	10%
2	80%	20%
3	70%	30%
4	60%	40%
5	50%	50%

Pengujian ini dilakukan untuk mengetahui tingkat keberhasilan dari sitem yang telah dibuat untuk identifikasi berita *rumor* dibagi menjadi 2 pengujian yang pertama menggunakan pembobotan TF-IDF lalu tidak menggunakan TF-IDF. Pada pengujian TF-IDF data dipecah karena penggunaan *N-gram*. Pengujian yang digunakan sebagai berikut:

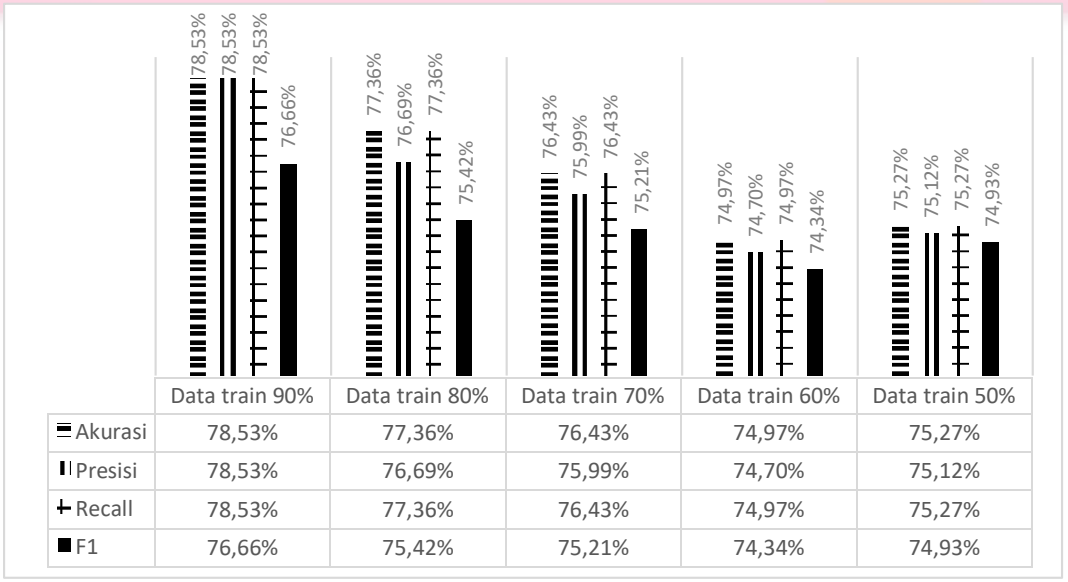
1. Non TF-IDF
2. Unigram
3. Bigram
4. Trigram
5. Unigram & Bigram

6. Unigram & Trigram
7. Unigram & Bigram & Trigram

1. Skenario 1
   Pengujian terhadap performa sistem dengan *Naïve Bayes Multinomial* dengan fitur TF-IDF dan tanpa TF-IDF. Pengujian ini menggunakan data set sebanyak 5 kali dengan data training 90%, 80%, 70%, 60%, 50% dengan rata rata dari 5 kali pengambilan data tersebut.
2. Skenario 2
   Pengujian tehadap performa sistem dengan *Naïve Bayes Multinomial* dengan *Ngram* menggunakan fitur TF-IDF dan tanpa TFIDF. Pengujian ini dilakukan sebanyak 5 kali dengan data training 90%, 80%, 70%, 60%, 50% untuk mengetahui pengaruh *Ngram* terhadap akurasi.

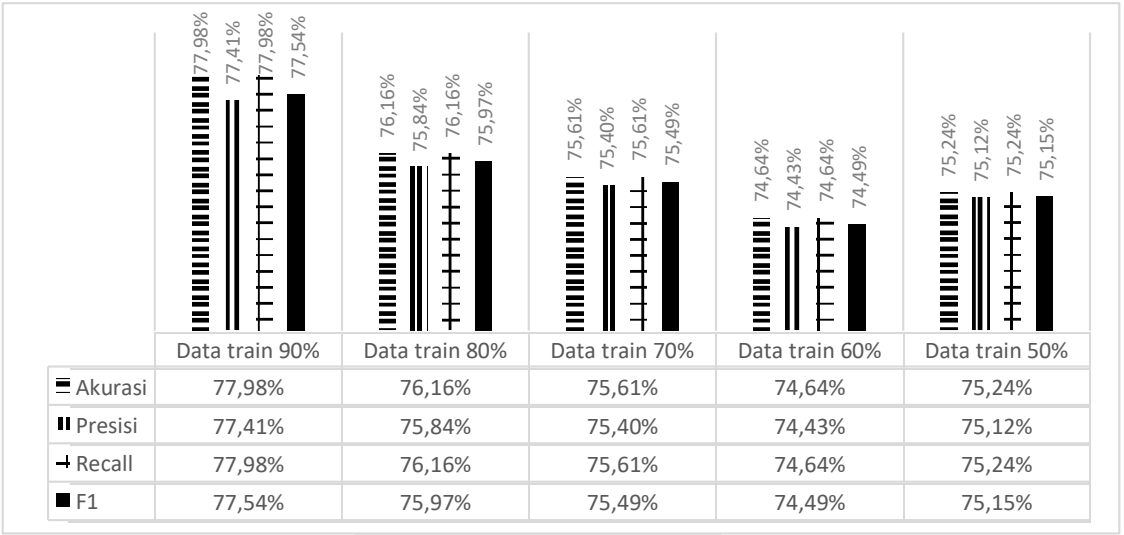
4.2 Analisis hasil pengujian

Pengujian untuk klasifikasi dengan rasio data uji yang berbeda dapat dilihat pada gambar dibawah



	Data train 90%	Data train 80%	Data train 70%	Data train 60%	Data train 50%
Akurasi	78,53%	77,36%	76,43%	74,97%	75,27%
Presisi	78,53%	76,69%	75,99%	74,70%	75,12%
Recall	78,53%	77,36%	76,43%	74,97%	75,27%
F1	76,66%	75,42%	75,21%	74,34%	74,93%

Gambar 2 Perbandingan *Confusion Matrix* dengan TF-IDF



	Data train 90%	Data train 80%	Data train 70%	Data train 60%	Data train 50%
Akurasi	77,98%	76,16%	75,61%	74,64%	75,24%
Presisi	77,41%	75,84%	75,40%	74,43%	75,12%
Recall	77,98%	76,16%	75,61%	74,64%	75,24%
F1	77,54%	75,97%	75,49%	74,49%	75,15%

Gambar 3 Perbandingan Confusion Matrix tanpa TF-IDF

Dari hasil pengujian yang dilakukan dapat dilihat hasil pembagian data yang di uji 90% memperoleh akurasi sebesar 78,53% dengan fitur TF-IDF, dan tanpa fitur TF-IDF akurasi sebesar 77,98%. Dapat kita lihat rasio

pembagian data pun berpengaruh dengan hasil akurasi. Semakin besar data *training* yang di uji, semakin besar nilai akurasi. Dapat kita lihat hasil yang diperoleh dengan adanya TF-IDF lebih besar dibandingkan dengan tanpa TF-IDF, dengan demikian pembobotan TF-IDF berpengaruh pada pengujian yang dilakukan. Rata – rata akurasi TF-IDF dan tanpa TF-IDF sebesar 0,7%, sedangkan *Precision* sebesar 1,4%, *recall* 0,7%, dan F1 sebesar 0,1%.

Tabel 6. Data *Train* 90% Perbandingan *Confusion Matrix*

	TFIDF	Tanpa TFIDF
Accuracy	78,53%	77,98%
Presicion	78,53%	77,41%
Recall	78,53%	77,98%
F1	76,66%	77,54%

Hasil dari analisis perbandingan menggunakan TF-IDF dengan tanpa TF-IDF pada data train 90% dan data *test* 10% menunjukan perbandingan yang tidak jauh terlalu berbeda.

4.3. Analisis Pengaruh *Ngram* terhadap Akurasi

Tabel 7. Data *Train* Akurasi *Ngram*

Fitur	TFIDF
Unigram	0,7796
Bigram	0,7096
Trigram	0,6937
Unigram + Bigram	0,7839
Bigram + Trigram	0,7053
Unigram + Bigram + Trigram	0,7842

Hasil *Ngram* terhadap klasifikasi memiliki nilai rata-rata *Ngram* menggunakan TF-IDF mendapatkan nilai 77,96%. Perbandingan hasil *Ngram* keseluruhan *Unigram+Bigram+Trigram* dengan fitur TF-IDF memperoleh nilai tertinggi sebesar 78,42%. Hasil akurasi *Unigram* dan kombinasi sangat kompetitif akan tetapi hasil tertinggi adalah kombinasi *Ngram* merupakan gabungan *Unigram*, *Bigram*, dan *Trigram*. Penggabungan *Ngram* tersebut memilki kelebihan informasi lebih banyak dibandingkan *Ngram* lainnya sehingga dapat meminimalisir kesalahan.

Nilai yang dihasilkan dari performansi sistem kurang maksimal, dikarenakan pengaruh pada proses *labeling* yang dilakukan terdapat kata yang sama dan data yang hilang ketika data di proses di *preprocessing* sehingga sistem tidak dapat menguji secara maksimal yang mengakibatkan peurunan nilai akurasi.

Tabel 8. Fitur pada Ngram

Label	Unigram	Bigram	Trigram
Non rumor	“betul”	“baca berita”	“infeksi virus orona”
	“baik”	“cegah virus”	“bapak Jokowi jadi”
	“fakta”	“gedung mpr”	“dewan wakil rakyat”
	“gitu”	“jokowi perintah”	“kena virus corona”
	“bangga”	“benah dulu”	“kerja kabinet Indonesia”
Rumor	“bakal”	“aku malah”	“kalau aku malah”
	“coba”	“tahu kan”	“sebar virus corona”
	“kayak”	“kalau bukan”	“ibu kota baru”
	“masa”	“kalau jadi”	“coret daftar negara”
	“calon”	“bisa jadi”	“kerja kabinet Indonesia”

5. Kesimpulan dan saran

Berdasarkan hasil pengujian dan analisis yang dilakukan penulis, dapat diberi kesimpulan bahwa metode TF-IDF dan klasifikasi Naïve bayes dapat digunakan dalam pendeteksi berita *rumor* pada twitter. Hasil akurasi yang diperoleh dalam pengujian dengan menggabungkan fitur *user-based* dan *tweet-based* sebesar 78,53% dengan fitur TF-IDF, sedangkan tanpa TF-IDF akurasi sebesar 77,98%. Hasil *Ngram* terbesar dengan TF-IDF *Unigram+Bigram+Trigram* rasio perbandingan 90:10 menghasil akurasi sebesar 78,42% sedangkan Tanpa TF-IDF sebesar 77,96%. Dengan adanya fitur TF-IDF, *Confusion Matrix* yang diperoleh pun lebih stabil dibandingkan tanpa TF-IDF memperoleh hasil yang naik turun. Pengaruh *Ngram* pun dapat dibilang sangat berpengaruh karena hasil yang diperoleh lebih besar dibanding tanpa *Ngram*. Hasil yang didapatkan dapat dikatakan kurang maksimal dikarenakan pada saat proses *labeling* terdapat beberapa kata yang sama pada kata *rumor* dan *non rumor*, dan juga terdapat *tweet* yang hilang atau bisa disebut data NaN sehingga proses menjadi kurang maksimal.

Saran untuk penelitian selanjutnya adalah melakukan perbaikan pada tahap *preprocessing* bagian *stopword* dan juga kamus normalisasi, sehingga data yang di olah lebih baik lagi dan menghasilkan nilai akurasi yang lebih optimal. Kemudian pada bagian *labeling* lebih di maksimalkan kembali karena proses *labeling* mempengaruhi hasil yang diperoleh.

Daftar Pustaka

[1] Asur, Sitaran dan Bernardo A. Huberman. Predicting The Future With Social Media. Proceeding WI-IAT ‘10 Proceeding of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology. Volume 01. Halaman 492-499. 2010.

[2] Prama Yoga Saputra. 2017. Implementasi Teknik Crawling untuk Pengumpulan Data dari Media Sosial Twitter. Jurnal Dinamika Dotcom . Volume 8 Nomor 2

[3] H. B. Dunn and C. A. Allen, (2005) "Rumors, urban legends and internet hoaxes," in Proceedings of the Annual Meeting of the Association of Collegiate Marketing Educators, p. 85.

[4] I. Destuardi.(2009). Klasifikasi Emosi Untuk Teks Bahasa Indonesia Menggunakan Metode Naive Bayes. Seminar Nasional Pascasarjana IX – ITS, Surabaya 12 Agustus 2009 ISBN No.

[5] Bruno Trstenjaka,Sasa Mikacb, Dzenana Donkoc. 2013. KNN with TF-IDF Based Framework for Text Categorization. Procedia Engineering 69 (2014) 1356 – 1364.

[6] Clark, A. (2003). Pre-processing Very Noisy Text. Proceedings of Workshop on Shallow Processing of Large Corpora (pp. 12-22). Lancaster: Lancaster University.

[7] Fitri, Meisya. (2013). Perancangan Sistem Temu Balik Informasi Dengan Metode Pembobotan Kombinasi Tf-Idf Untuk Pencarian Dokumen Berbahasa Indonesia. Universitas Tanjungpura : Semarang.

[8] Lowd, D., Domingos, P., Naive Bayes Models for Probability, Department of Computer Science and Engineering, University of Washington, Seattle.

[9] Pattekari, S. A., Parveen, A., 2012, Prediction System for Heart Disease Using Naive Bayes, International Journal of Advanced Computer and Mathematical Sciences, ISSN 2230-9624, Vol. 3, No 3, Hal 290-294.

[10] J. Han, M. Kamber and J. Pei, 2012, Data Mining Concepts and Techniques Third Edition, San Fransisco: Morgan Kauffman Publishers.

[11] Arina Indana Fahma.2018. Identifikasi Kesalahan Penulisan Kata (Typographical Error) pada Dokumen Berbahasa Indonesia Menggunakan Metode N-gram dan Levenshtein Distance. Vol. 2, No. 1, Januari 2018, hlm. 53-62

[12] Aditya Gaydhani. 2018. Detecting Hate Speech and Offensive Language on Twitter using Machine Learning: An N-gram and TFIDF based Approach. Department of Computer Engineering, Maharashtra Institute of Technology, Pune Pune, India. arXiv:1809.08651v1

[13] Reynald Karisma Wibowo. 2016. Penerapan Algoritma Winnowing Untuk Mendeteksi Kemiripan Teks Pada Tugas Akhir Mahasiswa. Techno.COM, Vol. 15, No. 4, November 2016 : 303-311

[14] Amelia Rahman. 2017. Online News Classification Using Multinomial Naive Bayes. ITSMART: Jurnal Ilmiah Teknologi dan Informasi. Vol. 6, No. 1, June 2017

[15] V. Qazvinian, E. Rosengren, D. R. Radev, and Q. Mei, (2011) "Rumor has it: Identifying misinformation in microblogs," Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 1589-1599.

[16] E. B. S. Z. K. A. B. Achmad Fauzi, “Deteksi Berita Hoax di Twitter dengan Metode Term Frequency Inverse Document Frequency dan Support Vector Machine,” Universitas Telkom, p. 2, 2019.