

## Peringkasan Sentimen Ekstraktif di Twitter Menggunakan Hybrid TF-IDF dan Cosine Similarity

Devid Haryalesmana Wahid\*<sup>1</sup>, Azhari SN<sup>2</sup>

<sup>1</sup> Prodi S2 Ilmu Komputer; FMIPA UGM, Yogyakarta

<sup>2</sup> Departemen Ilmu Komputer dan Elektronika, FMIPA UGM, Yogyakarta

e-mail: \*<sup>1</sup>devid.wahid@gmail.com, <sup>2</sup>arism.sottcomp@gmail.com

### Abstrak

Antusias masyarakat yang memberikan perhatian lebih terhadap akun resmi selebriti di Twitter memunculkan tren penggunaan Twitter sebagai upaya manajemen kesan. Penggalan reaksi masyarakat di media sosial merupakan upaya strategis untuk memperoleh umpan balik, namun tidak mudah dilakukan. Pengguna membutuhkan waktu yang lama untuk membaca ribuan tweet sekaligus memilah sentimennya, sehingga dibutuhkan peringkasan sentimen ekstraktif secara otomatis. Penelitian terdahulu umumnya tidak memasukkan informasi sentimen yang terkandung pada sebuah tweet sebagai bobot peringkat kalimat, sehingga hasil ringkasan masih berupa topik umum yang dibicarakan masyarakat.

Penelitian ini bertujuan mengkombinasikan metode SentiStrength, Hybrid TF-IDF dan Cosine Similarity untuk mengekstraksi ringkasan sentimen positif dan negatif masyarakat terhadap topik selebriti di Twitter secara otomatis, dengan artis Agnes Monica sebagai studi kasus. Metode SentiStrength digunakan untuk mendapatkan skor kekuatan sentimen dan mengklasifikasi tweet ke dalam kelas positif, negatif dan netral. Tweet bersentimen positif dan negatif diringkas dengan cara pemeringkatan tweet menggunakan Hybrid TF-IDF yang dikombinasi dengan skor kekuatan sentimen, kemudian menghilangkan tweet yang mirip menggunakan Cosine Similarity.

Hasil pengujian memperlihatkan bahwa kombinasi SentiStrength, Hybrid TF-IDF, dan Cosine Similarity mampu menghasilkan ringkasan sentimen dengan akurasi yang lebih baik dibandingkan menggunakan Hybrid TF-IDF saja, dengan perolehan akurasi rata-rata sebesar 60% dan f-measure sebesar 62%. Hal ini disebabkan karena penambahan kekuatan sentimen sebagai bobot peringkasan.

**Kata kunci**— peringkasan sentimen ekstraktif, analisis sentimen, klasifikasi, peringkasan teks otomatis, SentiStrength, Hybrid TF-IDF, Cosine Similarity

### Abstract

The using of Twitter by celebrities has become a new trend of impression management strategy. Mining public reaction in social media is a good strategy to obtain feedbacks, but extracting it are not trivial matter. Reads hundred of tweets while determine their sentiment polarity are time consuming. Extractive sentiment summarization machine are needed to address this issue. Previous research generally do not include sentiment information contained in a tweet as weight factor, as a results only general topics of discussion are extracted.

This research aimed to do an extractive sentiment summarization on both positive and negative sentiment mentioning Indonesian selebrity, Agnes Monica, by combining SentiStrength, Hybrid TF-IDF, and Cosine Similarity. SentiStrength is used to obtain sentiment strength score and classify tweet as a positive, negative or neutral. The summarization of posisitve and negative sentiment can be done by rank tweets using Hybrid TF-IDF summarization and sentiment strength score as additional weight then removing similar tweet by using Cosine Similarity.

*The test results showed that the combination of SentiStrength, Hybrid TF-IDF, and Cosine Similarity perform better than using Hybrid TF-IDF only, given an average 60% accuracy and 62% f-measure. This is due to the addition of sentiment score as a weight factor in sentiment summarization.*

**Keywords**—*extractive sentiment summarization, sentiment analysis, classification, automatic text summarization, SentiStrength, Hybrid TF-IDF*

## 1. PENDAHULUAN

Kemunculan jejaring sosial Facebook di tahun 2004 telah berhasil menarik minat jutaan orang diberbagai belahan dunia untuk bergabung menggunakan layanannya. Hal ini menyebabkan munculnya berbagai jejaring sosial lainnya yang memiliki keunggulan masing-masing untuk menarik perhatian pengguna Internet. Salah satunya adalah layanan *microblog* Twitter. Layanan *microblog* ini memungkinkan siapa saja untuk berkomunikasi tanpa batas jarak dan waktu. Tidak hanya dimanfaatkan untuk menjalin pertemanan, Twitter juga dimanfaatkan oleh fans maupun haters selebriti sebagai tempat untuk berkumpul dan berinteraksi langsung dengan artis idolanya.

Antusias masyarakat yang memberikan perhatian lebih terhadap akun resmi artis di Twitter memunculkan tren penggunaan Twitter sebagai strategi presentasi diri dengan melakukan manajemen kesan (*Impression Management*). Trend tersebut dipelopori oleh Ashton Kutcher sebagai selebriti pertama yang memiliki satu juta pengikut atau *followers* di Twitter. Trend tersebut juga mempengaruhi selebriti Indonesia, salah satunya adalah Agnes Monica atau Agnez Mo yang memiliki 14 juta *followers* Twitter dan menempatkan dirinya sebagai akun Twitter dengan jumlah *followers* terbanyak di Indonesia. Selain itu, Agnes Monica juga masuk dalam peringkat ketiga *Key Influencer* pada *Shorty Awards 2012*. *Key influencer* memiliki pengertian bahwa orang tersebut disadari sebagai seorang pakar pada bidangnya, dan memiliki banyak pengikut yang bergantung padanya dalam pemilihan keputusan. *Shorty Awards* merupakan penghargaan bagi orang atau perusahaan yang berpengaruh di Twitter.

Dalam upaya manajemen kesan, *Impression Management* digunakan untuk mendapatkan pujian atas pertunjukan, wawancara, mencari respon balik, dan kesuksesan dalam karir, sehingga menggali opini *fans* dan *haters* di media sosial dapat menjadi upaya untuk memperoleh umpan balik berupa apa yang disukai dan tidak disukai oleh *fans* maupun *haters* hingga pujian, dan kritik [1]. Walaupun terdapat banyak informasi yang terkandung pada *microblog* Twitter, mengekstraksi umpan balik di Twitter bukanlah pekerjaan yang mudah. Terbatasnya isi pesan berdampak pada rendahnya keragaman pesan tekstual pada sebuah *tweet*, banyak penyingkatan kata, kesalahan penulisan, kata-kata tidak baku dan susunan kalimat yang tidak terstruktur (*ungrammatical*). Membaca ribuan *tweet* satu per satu dan memilah opini positif dan negatif membutuhkan waktu yang lama, melelahkan dan membosankan dikarenakan jumlah *tweet* yang besar serta banyak *tweet* yang sama sekali tidak mengandung informasi yang berguna, sehingga dibutuhkan sebuah peringkasan otomatis oleh mesin untuk mengekstraksi informasi penting di Twitter.

Membuat ringkasan otomatis teks atau *Text Summarization* (TS) telah lama diteliti semenjak tahun 1958. Seiring dengan munculnya subjektifitas pada sebuah kata, kalimat atau dokumen, maka memungkinkan bentuk baru TS yaitu TS berbasis sentimen atau *Sentiment Summary*. Analisis sentimen mengidentifikasi sentimen pada sebuah dokumen di berbagai level (dokumen, fragmen, kalimat, atau bahkan pada level kata) [2]. Sedangkan TS mengidentifikasi bagian paling relevan pada sebuah dokumen tunggal maupun banyak dokumen dan kemudian membuat ringkasan yang mewakili, sehingga TS berbasis sentimen dilakukan dengan cara melakukan klasifikasi polaritas kalimat kedalam kelompok positif, negatif atau netral, kemudian menentukan kalimat mana yang paling penting atau menonjol untuk dijadikan ringkasan final.

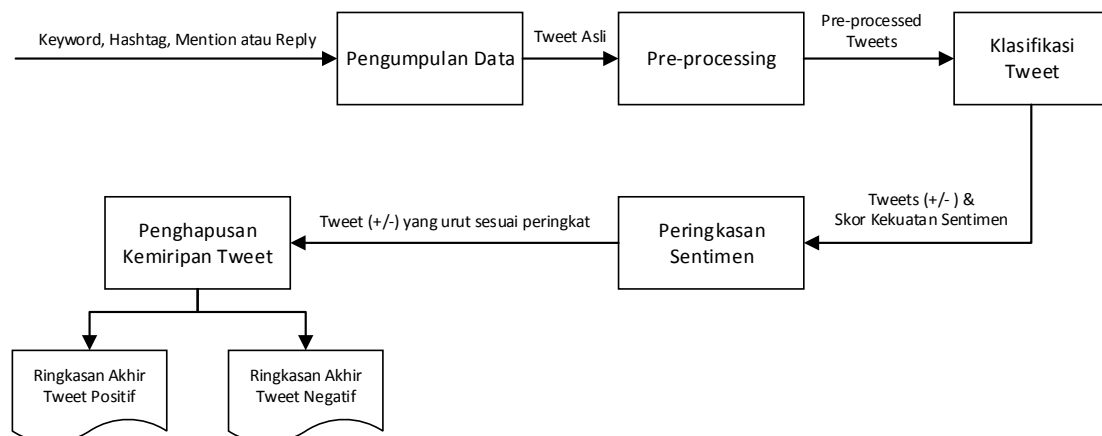
Walaupun *Text Summarization* dan *Sentiment Analysis* sudah lama diteliti dengan baik, penelitian kedua bidang tersebut pada area *microblog* Twitter masih relatif baru. Tidak banyak algoritma yang tersedia untuk melakukan peringkasan opini di Twitter yang mana dokumen Twitter memiliki karakteristik berbeda dibandingkan dokumen-dokumen pada umumnya. Literatur yang relevan di bidang peringkasan opini saat ini umumnya hanya berfokus pada domain peringkasan ulasan produk, *movie*, dan hotel pada sebuah situs-situs *marketplace*.

Pada penelitian ini, dilakukan kombinasi metode analisis sentimen menggunakan *SentiStrength* ke dalam metode peringkasan teks di Twitter (*Hybrid TF-IDF*), dan metode *Cosine Similarity* untuk mengeliminasi *tweet* hasil peringkasan yang memiliki kemiripan isi pesan. Dengan menggabungkan metode-metode tersebut diharapkan dapat meningkatkan kemampuan *Hybrid TF-IDF* dalam menghasilkan peringkasan opini di Twitter, khususnya pada masalah peringkasan opini *fans* dan *haters* selebriti.

## 2. METODE PENELITIAN

### 2.1 Rancangan Proses Umum

Rancangan proses merupakan penggambaran urutan alur kerja dari sistem mulai dari sistem berjalan hingga sistem tersebut berhenti. Secara umum proses yang terjadi di dalam sistem digambarkan oleh Gambar 1. Sistem dikembangkan berbasis *web* dimana pengguna akan memasukkan kata kunci yang berkaitan dengan selebriti kemudian memilih bulan yang akan dilakukan peringkasan.



Gambar 1 Proses Umum Peringkasan Sentimen Ekstraktif

### 2.2 Data Input

Data *input* merupakan data-data yang digunakan sebagai masukan pada sistem yang dibangun. Data tersebut kemudian akan diolah dan digunakan lebih lanjut untuk mendapatkan data *output* yang diinginkan. Data *input* yang akan digunakan antara lain:

1. *Tweet* Berbahasa Indonesia.

Data *tweet* diperoleh dengan melakukan *scraping* pada halaman *search.Twitter.com* sesuai dengan *keyword* yang diinputkan oleh *user*. *Tweet* hasil *scraping* akan disimpan ke dalam *database*.

2. Kamus Sentimen.

Kamus sentimen berisi kumpulan kata yang telah diberi bobot dengan kekuatan sentimen 1 (tidak memiliki sentimen positif) s.d. 5 (memiliki sentimen positif yang sangat kuat), dan -1 (tidak memiliki sentimen negatif) s.d. -5 (memiliki sentimen negatif yang sangat kuat). Kamus sentimen didapatkan dari hasil translasi kamus

sentimen berbahasa Inggris yang telah mengalami penambahan dan pengurangan kata berdasarkan dari hasil pengamatan dalam proses pengembangan sistem ini. Penelitian ini melibatkan 3 responden ahli, mahasiswa S2 Linguistik Universitas Gadjah Mada untuk memberikan bobot kekuatan sentimen. Nilai tengah dari ketiga responden yang dijadikan bobot akhir kekuatan sentimen.

3. Kamus Emosikon.

Emosikon adalah simbol atau kombinasi dari simbol-simbol yang biasanya digunakan untuk menggambarkan ekspresi wajah manusia yang mengandung emosi atau perasaan dalam bentuk pesan atau tulisan. Misalnya, simbol “:D” menunjukkan ungkapan emosi tertawa. Data ini dihasilkan secara manual berdasarkan pengamatan dalam proses pengembangan sistem ini. Kamus emosikon ini juga diberi bobot yang akan menentukan perubahan bobot sentimen pada sebuah kalimat.

4. Kamus *Idiom*.

Kamus *Idiom* adalah kata ungkapan yang memiliki makna berbeda dengan kata aslinya. Misalnya, kata “*kumpul kebo*” bukan berarti peristiwa berkumpulnya kerbau melainkan ungkapan pengganti kegiatan hubungan seks bebas yang dilakukan lebih dari 2 orang bersama-sama. Kamus *Idiom* ini juga diberi bobot yang akan menentukan perubahan bobot sentimen pada sebuah kalimat. Data ini dihasilkan secara manual berdasarkan pengamatan dalam proses pengembangan sistem ini.

5. *Slangwords* dan Konversinya.

*Slangwords* adalah kata-kata tidak baku yang sering digunakan seperti penggunaan kata yang disingkat, bahasa pergaulan yang biasa digunakan sehari-hari serta kata yang memiliki kesalahan penulisan. Data *slangwords* dan konversinya didapatkan dari penelitian Riandaru [3] dan telah mengalami penambahan kata yang didapatkan secara manual berdasarkan pengamatan dalam proses pengembangan sistem ini.

6. *Boosterwords*.

*Boosterwords* adalah kata yang dapat meningkatkan atau mengurangi intensitas sentimen kata disebelahnya. Misal, kata “*sangat bahagia*” lebih positif daripada kata “*bahagia*”, kata “*kurang bahagia*” tidak lebih positif daripada kata “*bahagia*”. Kata ini diberi bobot 1-2 untuk menambah atau mengurangi skor kata disampingnya. Data ini dihasilkan secara manual berdasarkan pengamatan dalam proses pengembangan sistem ini.

7. Kata Negasi.

Kata negasi merupakan kata yang terdapat dalam suatu kalimat yang dapat mengubah orientasi dari suatu opini. Misalnya, kata “*nakal*” merupakan kata yang bermakna negatif, akan tetapi jika kata “*nakal*” didahului kata negasi “*tidak*” maka susunan kata menjadi “*tidak nakal*” yang bermakna positif. Kata ini disimpan dalam bentuk file *.txt*, tidak disimpan dalam *database*. Data ini dihasilkan secara manual berdasarkan pengamatan dalam proses pengembangan sistem ini.

8. Kata Tanya.

Kata tanya merupakan kata yang terdapat dalam suatu kalimat yang dapat mengubah orientasi dari suatu opini. Misalnya, “*apakah kamu marah?*”, walaupun terdapat kata “*marah*”, kalimat ini tidak memiliki orientasi positif maupun negatif, sehingga diklasifikasikan sebagai kalimat netral. Kata ini disimpan dalam bentuk file *.txt*, tidak disimpan dalam *database*. Data ini dihasilkan secara manual berdasarkan pengamatan dalam proses pengembangan sistem ini.

9. Kamus Bahasa Inggris.

Kamus bahasa Inggris berisi 10.000 kata umum berbahasa Inggris yang dibutuhkan untuk melakukan penyaringan *term* berbahasa Inggris pada tahap *pre-processing*. Data ini didapatkan dari repositori Josh Kaufman di [github.com](https://github.com) yang bersumber dari analisis *n-gram frequency* oleh Google’s Trillion Word Corpus.

### 2.3 Pengumpulan Data

Pada tahap pengumpulan data dilakukan dengan cara *scrapping* halaman pencarian Twitter. Proses *scrapping* dimulai dengan *user* memasukkan *query* pencarian Twitter yang dapat berupa *keyword*, *mention*, *hashtag*, *reply* dan rentang tanggal *tweet* dibuat.

### 2.4 Pre-processing

Data *tweet* yang telah didapatkan akan mengalami *pre-processing* untuk mendapatkan data bersih agar proses klasifikasi dan peringkasan *tweet* menjadi lebih akurat. Proses *pre-processing* terdiri dari beberapa tahapan, antara lain:

- Penghapusan *newline*, contoh: “\n”.
- Penghapusan HTML *special character*.
- Penghapusan karakter *Unicode* yang berada diluar jangkauan ASCII.
- Penghapusan URL, seperti “http://site.com”, “www.site.com”.
- Penghapusan *mention*, karena umumnya tidak mengandung informasi yang penting, seperti “@agnezmo”.
- Penghapusan *hashtag*, seperti “#hutindosiar”
- Penghapusan kalimat yang berupa kutipan, karena umumnya kutipan berupa lirik lagu, *headline* berita, *retweet*, atau bahkan kutipan ayat kitab suci yang dapat mengurangi akurasi proses berikutnya.
- Pemisahan gabungan kata dalam bentuk *camel-case*. Gabungan kata ini umumnya terdapat dalam *hashtag*, seperti #AkuCintaKamu.
- Konversi bentuk-bentuk penyingkatan kata majemuk ke bentuk yang baku, umumnya penyingkatan menggunakan simbol angka 2 dan tanda petik dua.
- Penyisipan spasi antara huruf dan bukan huruf agar *emoticon* yang melekat pada sebuah kata dapat terpisah.
- Penyisipan spasi antara tanda koma dan titik.
- Perbaikan *emosikon* yang tidak terpisah dengan baik sebagai efek dari tahap sebelumnya.
- *Case Folding* atau konversi semua karakter huruf besar (*uppercase*) menjadi karakter huruf kecil (*lowercase*).
- Konversi kata tidak baku menjadi bentuk baku dan memperbaiki kesalahan penulisan.
- Penghapusan karakter selain *alphanumeric*, tanda seru, tanda tanya, titik, koma, yang tidak membentuk sebuah kalimat majemuk atau *emoticon*.
- Perhitungan jumlah kata dalam bahasa Inggris yang terkandung dalam *tweet*.
- Deteksi *tweet* yang mengandung kata dalam daftar hitam. Daftar hitam berisi kata yang mengindikasikan *tweet* tersebut relevan atau tidak. Contoh *tweet* yang tidak relevan yaitu sebuah *retweet*, ajakan *follow*, *invite pin bbm*, *headline* berita, kata khas bahasa sunda, kutipan lirik lagu, *request* lagu pada akun *Twitter* radio, dsb.
- Penghapusan *tweet* yang kurang dari 10 kata, mengandung lebih dari 2 kata dalam bahasa Inggris, dan mengandung kata dalam daftar hitam.

### 2.5 Klasifikasi Tweet

Klasifikasi *tweet* menggunakan algoritma *SentiStrength*. *SentiStrength* adalah algoritma klasifikasi yang menggunakan pendekatan berbasis leksikon yang menggunakan aturan-aturan dan informasi linguistik tambahan (non-leksikal) untuk mendeteksi kekuatan sentimen di sebuah teks singkat dalam bahasa Inggris [4], [5]. *SentiStrength* menggunakan sistem *dual-scale (positif-negatif)*, karena menurut penelitian psikologi, manusia dapat merasakan emosi positif dan negatif secara bersamaan hingga batas tertentu secara mandiri [6]. *SentiStrength* akan menghasilkan nilai positif dan negatif, dimana jangkauan nilai dimulai dari angka 1 sampai 5. Nilai 1 menunjukkan kalimat tersebut tidak memiliki sentimen positif maupun negatif,

sedangkan nilai 5 menunjukkan kalimat tersebut memiliki sentimen sangat positif atau sangat negatif.

Skor akhir pada sebuah kalimat ditentukan dari skor positif tertinggi dan skor negatif tertinggi dari kata-kata penyusunnya. Contoh, “agnezmo is horrible and nasty but lovely.”, maka hasil *parsing* sebagai berikut, “agnezmo is horrible [-4] and nasty [-3] but lovely [2]. <skor: 2, -4>”. Angka di dalam tanda “[...]” menunjukkan skor kekuatan sentimen setiap kata penyusun kalimat yang sesuai dengan kamus sentimen kata, sedangkan angka di dalam tanda “<skor: ..., ...>” mengindikasikan skor akhir kalimat. Dengan demikian, skor akhirnya adalah nilai positif tertinggi dan nilai negatif tertinggi, yaitu 2 dan -4. Skor akhir tersebut dapat berubah jika memenuhi aturan-aturan tambahan pada ekstraksi fitur. Kemudian keputusan akhir sentimen didasarkan pada aturan:

- **if** positive value > negative value **then** positive sentiment.
- **if** positive value < negative value **then** negative sentiment.
- **if** positive value = negative value **then** neutral sentiment.

Selain menggunakan kamus sentimen dengan pembobotan *term* oleh manusia, *SentiStrength* juga menggunakan kamus emosikon dan kamus ungkapan yang juga diberi bobot oleh manusia. Adapun aturan-aturan tambahan yang dapat mengubah skor akhir pada tahap ekstraksi fitur ditunjukkan pada Tabel 1.

Tabel 1 Standar ekstraksi fitur *SentiStrength*

No.	Fitur	Keterangan	Contoh
1.	F1	Menggunakan kamus negasi untuk membalik polaritas skor <i>term</i> didepannya	Cantik = 4 <b>Tidak Cantik</b> = -4
2.	F2	Menggunakan kamus penguat kata ( <i>Booster Words</i> ) untuk menambah atau mengurangi skor <i>term</i> didepan atau dibelakangnya	Cantik = 4 <b>Sangat Cantik</b> = 5
3.	F3	Menggunakan kamus emosikon	:) = 2
4.	F4	Menggunakan kamus ungkapan ( <i>idiom</i> )	Besar Kepala = -4
5.	F5	Menambah skor sentimen jika terjadi perulangan huruf lebih dari 2. Pada <i>term</i> positif/negatif +1, <i>term</i> netral +2	Keren = 4 Kereeeeeen = 5
6.	F6	Kalimat dengan tanda seru mendapatkan skor minimal 2	Hai semua! = 2
7.	F7	<i>Term</i> yang diikuti lebih dari satu tanda seru	Cantik = 4 Cantik!!!! = 5
8.	F8	<i>Term</i> yang berurutan dengan polaritas yang sama dan skor minimal +3/-3 meningkatkan skor pada <i>term</i> berikutnya.	Banci = 3 Nakal = 4 Banci [3] Nakal [5]
9.	F9	Mengabaikan emosi negatif pada kalimat tanya menjadi netral	Apakah kamu marah [-4] ? = -1 (Netral)
10.	*F10	Menetralkan kata sentimen yang diikuti kata “jika” dan “kalau”	Berani = 4 Kalau berani = 1

Catatan. \*) fitur tambahan

## 2.6 Peringkasan Sentimen

*Term Frequency – Inverse Document Frequency (TF-IDF)* adalah teknik pembobotan berbasis statistik yang sering diterapkan di berbagai permasalahan penggalian informasi. Namun, secara umum *TF-IDF* tidak banyak dikenal sebagai algoritma untuk peringkasan teks otomatis.

Pada peringkasan teks otomatis menggunakan *TF-IDF*, ide dasarnya adalah memberikan bobot pada setiap kalimat dalam sebuah dokumen. Setelah masing-masing kalimat diberikan bobot, kalimat akan diurutkan berdasarkan bobot dimana kalimat  $k$  teratas dengan bobot paling besar akan dipilih sebagai hasil akhir ringkasan. Bobot kalimat diperoleh dari penjumlahan bobot *term* pada sebuah kalimat, dimana *term* dapat berupa kata, frasa atau tipe sintatik lainnya.

Dalam konteks peringkasan teks *tweet* di Twitter, Sharifi *et al*, mengusulkan algoritma *Hybrid TF-IDF*. Menurut algoritma tersebut masing-masing *tweet* dianggap sebagai dokumen terpisah, hanya saja dalam perhitungan frekuensi *term* dilakukan pada keseluruhan *tweet*, sehingga nilai TF menjadi normal sekaligus tidak kehilangan properti IDF [7], [8]. Persamaan (1), (2), dan (3) menunjukkan bagaimana perhitungan *Hybrid TD-IDF* dilakukan, dimana  $w_i$  adalah *term* ke- $i$ , maka  $W(w_i)$  adalah bobot pada masing-masing *term*.

$$W(w_i) = tf(w) * \log_2(idf(w_i)) \quad (1)$$

$$tf(w_i) = \frac{\text{Jumlah kemunculan term di semua tweet}}{\text{Jumlah term di semua tweet}} \quad (2)$$

$$idf(w_i) = \frac{\text{Jumlah tweet}}{\text{Jumlah tweet yang memuat term}} \quad (3)$$

Berdasarkan persamaan (1), (2), dan (3), maka perhitungan untuk melakukan *microblog summarization* menggunakan *Hybrid TF-IDF* ditunjukkan pada persamaan (4) dan (5), dimana  $S$  adalah kalimat,  $W(S)$  adalah pembobotan pada masing-masing kalimat,  $nf(S)$  adalah faktor normalisasi kalimat. Nilai *ThresholdMinimal* diperoleh dari menghitung rata-rata jumlah kata dalam kalimat pada semua koleksi *tweet*.

$$W(S) = \frac{\sum_{i=0}^{\#KataDalamKalimat} W(w_i)}{nf(S)} \quad (4)$$

$$nf(S) = \max[\text{MinThreshold}, \#KataDalamKalimat] \quad (5)$$

Dalam konteks peringkasan sentimen, skor hasil normalisasi akan ditambah dengan skor kekuatan sentimen, sehingga *tweet* dengan kekuatan sentimen tertinggi akan memiliki bobot lebih besar.

## 2.7 Penghapusan Tweet yang Mirip

Pada tahap penghapusan *tweet* yang mirip, pertama-tama mengambil *tweet* pada *summary collection* berdasarkan bulan, polaritas tweet serta diurutkan berdasarkan skor tertinggi. *Tweet* pertama dengan skor tertinggi otomatis menjadi ringkasan final. *Tweet* kedua, ketiga dan seterusnya akan dihitung terlebih dahulu skor kemiripannya dengan daftar ringkasan final. Perhitungan kemiripan *tweet* dilakukan menggunakan metode *Cosine Similarity*. Perhitungan kemiripan akan dilakukan berulang-ulang hingga didapatkan ringkasan akhir sebanyak yang ditentukan, sehingga hasil ringkasan final yang dihasilkan akan berbeda antar *tweet* satu dengan yang lainnya.

Perhitungan *cosine similarity* akan menghitung jarak sudut antara dua buah vektor. Semakin kecil sudut yang dibentuk antara dua buah vektor, maka tingkat kesamaan antara dokumen dan *query* tersebut semakin besar. Perhitungan *cosine similarity* ditunjukkan oleh persamaan (6).

$$S(D_i, Q_i) = \frac{\sum(D_i * Q_i)}{\sqrt{(\sum D_i^2 * \sum Q_i^2)}} \quad (6)$$

Nilai  $D_i$  merupakan nilai *term frequency* dari *term* suatu dokumen yang sesuai dengan *term* suatu *query*, sedangkan nilai  $Q_i$  adalah nilai *term frequency* dari *term* suatu *query* yang sesuai dengan *term* suatu dokumen. Ketika dokumen dan *query* tersebut identik, sudutnya adalah  $0^\circ$  dan nilai kesamaannya adalah 1, sedangkan ketika dokumen dan *query* tidak identik sama sekali, sudutnya adalah  $90^\circ$  dan nilai kesamaannya adalah 0 [9].

### 2.8 Pengujian

Pengujian merupakan hal penting untuk memastikan bahwa suatu algoritma yang telah dirancang dapat berjalan sesuai dengan harapan. Dalam penelitian ini, pengujian terbagi dua yaitu: pengujian keberhasilan *SentiStrength* dalam melakukan klasifikasi sentimen sebuah *tweet*, dan pengujian perbandingan antara peringkasan sentimen yang dikembangkan menggunakan kombinasi metode *SentiStrength* dan *Hybrid TF-IDF* dengan peringkasan yang hanya menggunakan metode *Hybrid TF-IDF*. Skenario pengujian perbandingan ketiga variasi metode tersebut meliputi pengujian pada target ringkasan sebesar 30% dan 50% dari 10 dan 20 *tweet* pada masing-masing kelas.

Pengujian klasifikasi sentimen dilakukan dengan menguji aplikasi yang telah dibangun dengan membandingkan antara data prediksi dan data aktual [10]. Data prediksi berupa hasil klasifikasi yang dihasilkan oleh aplikasi yang dibangun, sedangkan data aktual berupa hasil klasifikasi yang dihasilkan dari oleh manusia. Data aktual didapatkan melalui pelabelan *tweet* ke dalam kelas positif, negatif, dan netral. Algoritma *SentiStrength* yang digunakan bersifat *unsupervised*, tidak membutuhkan data latih untuk mengklasifikasi sentimen, sehingga pengujian pengujian klasifikasi dilakukan dengan membagi data uji berdasarkan jumlah bulan pada koleksi *tweet* yang berhasil dikumpulkan, yaitu sebanyak 8 bulan. Hasil pengujian pada masing-masing bulan kemudian akan dirata-rata untuk memperoleh hasil pengujian klasifikasi sentimen secara keseluruhan.

Pengujian peringkasan teks otomatis secara umum terdapat dua metode, yaitu secara intrinsik atau ekstrinsik [11]. Pengujian secara intrinsik mengukur kualitas sistem peringkasan otomatis itu sendiri yang difokuskan pada seberapa koheren dan informatif dari suatu ringkasan, umumnya dibandingkan dengan bantuan ringkasan yang dibuat oleh manusia. Sedangkan metode ekstrinsik mengukur bagaimana pengaruhnya (efektivitas dan akseptabilitas) hasil ringkasan tersebut pada sistem lain misalnya pencarian dokumen, *indexing*, atau klasifikasi dokumen.

Pada penelitian ini akan dilakukan pengujian secara intrinsik dengan menghitung akurasi dan *F-Measure* berdasarkan perhitungan *precision* dan *recall* merupakan standar evaluasi dalam penghitungan *information retrieval*. Perhitungan akurasi, *precision* dan *recall* ditunjukkan pada persamaan 7 s.d. 10.

$$\text{Akurasi} = \frac{\# \text{ ringkasan relevan}}{\sum \text{ ringkasan}} \quad (7)$$

$$\text{Recall} = \frac{\# \text{ Kalimat ringkasan sistem} \cap \text{ ringkasan ideal}}{\sum \text{ kalimat ringkasan ideal}} \quad (8)$$

$$\text{Precision} = \frac{\# \text{ Kalimat ringkasan sistem} \cap \text{ ringkasan ideal}}{\sum \text{ kalimat ringkasan sistem}} \quad (9)$$

$$F - \text{measure} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{(\text{precision} + \text{recall})} \quad (10)$$



Akurasi didapatkan dengan cara membagi jumlah ringkasan sistem yang relevan dengan jumlah ringkasan yang dihasilkan. Ringkasan ideal diperoleh dengan memanfaatkan *tool* formulir Google yang akan dibagikan kepada responden. Formulir dapat diakses responden secara *online*. Formulir berisi instruksi kepada responden untuk memilih *tweet* yang dianggap paling penting dan sesuai dengan kelas sentimen masing-masing. Dalam menentukan *tweet* terpenting sebagai ringkasan ideal pasti terdapat perbedaan pendapat antar subyek pembuat ringkasan dalam memilih *tweet*. Misal subyek A menganggap *tweet* x sebagai *tweet* terpenting tetapi bisa saja subyek B berpendapat sebaliknya bahwa *tweet* x kurang begitu penting. Untuk mengatasi hal ini metode suara terbanyak (*majority opinion*) digunakan sebagai penentuan *tweet* ringkasan ideal.

### 3. HASIL DAN PEMBAHASAN

Total *tweet* asli hasil pengumpulan data sebanyak 7.421. Total *tweet* yang digunakan untuk pengujian sebanyak 1.860 data *tweet* hasil *pre-processing*, 25% dari data asli. *Tweet* terbagi dalam 8 bulan dengan distribusi yang diperlihatkan pada Tabel 2. Jumlah kamus yang digunakan sentimen sebanyak 1.729 *term*, kata tidak baku atau *slang words* sebanyak 4.266, kamus emosikon sebanyak 89 *term*, kamus kata penguat sebanyak 29 *term*, kamus kata ungkapan sebanyak 92 *term*, kamus kata negasi sebanyak 10 *term*, dan kamus kata tanya sebanyak 20 *term*.

Tabel 2 Distribusi data *tweet*

Bulan	Jumlah <i>Tweet</i> Asli	Jumlah <i>Tweet</i> Hasil <i>Pre-processing</i>
1	1.948	558
2	771	150
3	785	177
4	533	112
5	728	195
6	773	227
7	855	204
8	1.028	237
<b>Total</b>	<b>7.421</b>	<b>1.860</b>

Berdasarkan Tabel 3, terlihat bahwa hasil klasifikasi sentimen menggunakan *SentiStrength* memberikan akurasi secara keseluruhan sebesar 74,13%. Pada pengujian presisi, *recall* dan *F1-score* masing-masing kelas dirata-rata, kemudian didapatkan presisi secara keseluruhan sebesar 73,08%, *recall* sebesar 71,77%, dan *F1-score* sebesar 71,69%.

Tabel 3 Hasil pengujian klasifikasi sentimen menggunakan *SentiStrength*

Month	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
1	63,98	60,75	61,12	60,53
2	76,00	76,75	74,03	75,04
3	80,23	80,08	72,20	74,70
4	75,00	71,32	66,76	68,16
5	71,28	73,18	71,17	71,91
6	67,40	66,50	67,71	66,57
7	76,47	76,09	76,13	75,91
8	82,70	79,98	85,06	80,66
<b>Average</b>	<b>74,13</b>	<b>73,08</b>	<b>71,77</b>	<b>71,69</b>

Dalam pembuatan ringkasan opini, skor akhir *Hybrid TF-IDF* dikombinasikan dengan skor kekuatan sentimen sebagai bobot tambahan. Dalam penelitian ini, penjumlahan skor *Hybrid TF-IDF* dan *SentiStrength* (SS) menjadi *baseline* pengujian yang akan dibandingkan dengan menggunakan skor *Hybrid TF-IDF* saja, dan mengalikan skor *Hybrid TF-IDF* dengan skor *SentiStrength*.

Tabel 4 Perbandingan pengujian peringkasan sentimen

Dokumen Uji				HTFIDF & CS				SS+HTFIDF & CS				SS*HTFIDF & CS			
M	CR	#T	C	A	P	R	F	A	P	R	F	A	P	R	F
1	30%	10	+	67	67	67	67	33	33	33	33	67	67	67	67
1	30%	10	-	75	75	100	86	50	50	67	57	75	75	100	86
1	30%	20	+	43	43	50	46	57	57	67	62	57	57	67	62
1	30%	20	-	50	50	50	50	67	67	67	67	67	67	67	67
1	50%	10	+	60	60	60	60	80	80	80	80	60	60	60	60
1	50%	10	-	60	60	60	60	80	80	80	80	80	80	80	80
1	50%	20	+	50	50	50	50	70	70	70	70	50	50	50	50
1	50%	20	-	30	30	30	30	50	50	50	50	40	40	40	40
2	30%	10	+	67	67	67	67	67	67	67	67	67	67	67	67
2	30%	10	-	20	20	33	25	60	60	100	75	40	40	67	50
2	30%	20	+	0	0	0	0	50	50	50	50	0	0	0	0
2	30%	20	-	43	43	50	46	43	43	50	46	43	43	50	46
2	50%	10	+	60	60	60	60	60	60	60	60	60	60	60	60
2	50%	10	-	40	40	40	40	60	60	60	60	40	40	40	40
2	50%	20	+	50	50	50	50	60	60	60	60	50	50	50	50
2	50%	20	-	64	64	70	67	73	73	80	76	64	64	70	67
<b>Average</b>				<b>49</b>	<b>49</b>	<b>52</b>	<b>50</b>	<b>60</b>	<b>60</b>	<b>65</b>	<b>62</b>	<b>54</b>	<b>54</b>	<b>58</b>	<b>56</b>

Keterangan: M (bulan), CR (Compression Ratio), #T (Jumlah tweet), C (Kelas sentimen), SS (*SentiStrength*), CS (Cosine Similarity), A (Akurasi), P (Precision), R (Recall), F (F-measure)

Berdasarkan Tabel 4, dapat diketahui bahwa kombinasi bobot kekuatan sentimen dengan bobot *Hybrid TF-IDF* dapat menghasilkan ringkasan jauh lebih relevan dibandingkan hanya menggunakan skor *Hybrid TF-IDF* saja dengan perolehan akurasi rata-rata sebesar 60% dan *f-measure* sebesar 62%. Kombinasi bobot kekuatan sentimen dan *Hybrid TF-IDF* dengan menjumlahkan bobot jauh lebih baik dibandingkan dengan mengalikan bobot.

#### 4. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, maka diperoleh kesimpulan:

1. Peringkasan sentimen ekstraktif di Twitter dapat dilakukan dengan cara pemeringkatan *tweet* berdasarkan penjumlahan skor kekuatan sentimen dengan skor *Hybrid TF-IDF* yang telah dinormalisasi dan dihilangkan hasil ringkasan yang mirip.
2. Kombinasi *SentiStrength*, *Hybrid TF-IDF*, dan *Cosine Similarity* dengan penambahan skor kekuatan sentimen sebagai bobot peringkasan mampu menghasilkan ringkasan yang relevan dengan tingkat akurasi yang lebih baik dibandingkan menggunakan *Hybrid TF-IDF* saja dengan perolehan akurasi rata-rata sebesar 60% dan *f-measure* sebesar 62%.
3. Hasil ringkasan dengan *compression ratio* yang lebih kecil memperoleh akurasi lebih rendah.
4. Hasil ringkasan dengan jumlah dokumen yang lebih besar memperoleh akurasi lebih kecil.

#### 5. SARAN

Pada penelitian selanjutnya, diharapkan sistem dapat dikembangkan dengan menggunakan algoritma lain yang menggunakan pendekatan *supervised learning* dan mampu mendeteksi sarkasme. Sebaiknya metode *SentiStrength* hanya digunakan sebagai metode untuk mendapatkan skor kekuatan sentimen sebagai pembobotan tambahan pada peringkasan opini.

#### DAFTAR PUSTAKA

- [1] Alim, C. A., 2015, Impression Management Agnes Monica Melalui Akun Instagram (@agnezmo), *Jurnal e-Komunikasi*, vol. 2, no. 3.
- [2] Pang, B. dan Lee, L., 2008, Opinion mining and sentiment analysis, *Foundations and trends in information retrieval*, vol. 2, no. 1-2, pp. 1-135.
- [3] Riandaru, V., 2016, Penggolongan Program Sinetron Berdasarkan Opini Masyarakat di Twitter dengan Cosine Similarity, *Tesis*, Master of Computer Science, Universitas Gadjah Mada, Indonesia.
- [4] Thelwall, M., Buckley, K., Paltoglou, G., Cai, D. & Kappas, A., 2010, Sentiment Strength Detection in Short Informal Text, *Journal of the American Society for Information Science and Technology*, 61(12), 2544–2558.
- [5] Thelwall, M., Buckley, K. & Paltoglou, G., 2012, Sentiment Strength Detection for the Social Web, *Journal of the American Society for Information Science and Technology*, 63(1), 163-173.
- [6] Norman, G. J., Norris, C., Gollan, J., Ito, T., Hawkey, L., Larsen, J., Berntson, G. G., 2011, Current emotion research in psychophysiology: The neurobiology of evaluative bivalence, *Emotion Review*, 3, 3349-359.
- [7] Sharifi, B., Hutton, M. A. & Kalita, J. K., 2010, Experiments in Microblog Summarization, *Social Computing (SocialCom)*, 2010 IEEE Second International Conference, IEEE, 49-56.

- 
- [8] Sharifi, B. P., Inouye, D. I. & Kalita, J. K., 2013, Summarization of Twitter Microblogs, *The Computer Journal*, bxt109.
- [9] Imbar, R.V., Adelia., Ayub, M., dan Rehatta, A., 2014, Implementasi Cosine Similarity dan Algoritma Smith-Waterman untuk Mendeteksi Kemiripan Teks, *Jurnal Informatika*, Vol.10, No.1, pp.31-42.
- [10] Han, J dan Kamber, M., 2006, *Data Mining: Concepts and Technique 2nd Edition*, Morgan Kauffman Publisher, San Fransisco.
- [11] Pratama, F., 2014, Rancang Bangun Aplikasi Peringkat Teks Otomatis Artikel Berbahasa Indonesia Menggunakan Metode Term Frequency Inverse Document Frequency (TF-IDF) dan K-Mean Clustering, *Skripsi*, Jurusan Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Sultan Syarif Kasim, Pekanbaru.