ANALISA SENTIMEN PADA TINJAUAN BUKU DENGAN ALGORITMA K-NEAREST NEIGHBOUR

Luvia Friska Narulita
Program Studi Teknik Informatika, Fakultas Teknik
Universitas 17 Agustus 1945 Surabaya
Email : luvianarulita@gmail.com

ABSTRAK

Analisa sentimen pada tinjauan buku dapat digunakan untuk pengklasifikasian dokumen tinjauan sehingga pembagian sentimen positif dan negatif dapat dilakukan secara sistemis. Penggunaan metode *k-nearest neighbor* dan digabungkan dengan metode pembobotan istilah dan penghitungan tingkat kemiripan memberikan hasil yang cukup baik pada penelitian yang telah dilakukan.

Kata Kunci: *analisa sentimen, similarity, k nearest neighbor, term frequency*

1. Pendahuluan

Menurut hasil survey Asosiasi Penyelenggara Jasa Internet Indonesia (APJII) [1], penggunaan internet di Indonesia semakin meluas, dimana sebanyak 132,7 juta orang dari 256,2 juta penduduk Indonesia atau sekitar 51,8 persen penduduk Indonesia telah mengenal internet. Penggunaan internet oleh masyarakat Indonesia sangat beragam, diantaranya adalah untuk berbisnis, sosialisasi, pekerjaan, mengisi waktu luang dan untuk pendidikan.

Internet menjadi rujukan bagi masyarakat untuk mendapatkan informasi tentang berbagai hal, termasuk tinjauan dari pengguna mengenai buku atau barang – barang yang akan dibeli. Adanya tinjauan terhadap barang dan buku mempengaruhi keputusan seseorang untuk melakukan pembelian barang [2].

Salah satu situs yang dapat diakses untuk membaca tinjauan mengenai buku yang telah atau sedang dibaca adalah goodreads.com. Tinjauan yang dituliskan oleh pengguna situs goodreads.com dapat berisi tinjauan negatif atau positif. Hal tersebut mendasari peneliti untuk melakukan analisa sentimen terhadap tinjauan buku pada situs goodreads.com yang berbahasa Indonesia.

Penelitian mengenai analisa sentimen telah banyak dilakukan. Firmansyah dkk [3] menggunakan metode Naïve Bayes dan Query Expansion untuk mengklasifikasi tinjauan pada aplikasi mobile. Tingkat akurasi hasil penelitian tersebut pada saat menggunakan metode Naïve Bayes adalah sebesar 95% sedangkan pada saat menggunakan kombinasi Naïve Bayes dan Query Expansion mencapai 98%.

Penelitian yang dilakukan oleh Gülen Toker [4] menggunakan algoritma k-Nearest Neighbor untuk melakukan klasifikasi terhadap teks. Dalam penelitian tersebut disimpulkan bahwa algoritma k-Nearest Neighbor merupakan algoritma yang efisien dan sederhana untuk melakukan klasifikasi teks.

Berdasarkan uraian yang telah dituliskan, maka peneliti mengusulkan untuk menggunakan algoritma k –

Nearest Neighbor sebagai algoritma klasifikasi pada tinjauan buku. Pada penelitian ini dilakukan proses pembobotan kata dengan metode *tfidf* (*term frequency – inverse document frequency*) dan melakukan proses penghitungan tingkat kemiripan dari kata - kata dalam dokumen tinjauan sebelum dilakukan proses klasifikasi dengan algoritma k-Nearest Neighbor.

Data tinjauan dari situs goodreads.com digunakan dalam penelitian ini. Data tinjuan hanya berupa data tinjauan yang berbahasa Indonesia dan diambil dari sembilan buku berbeda yang diterbitkan di Indonesia. Data yang diambil sebanyak seratus data tinjauan dengan pembagian enam puluh lima (65) data latih dan tiga puluh lima (35) data uji.

2. Tinjauan Pustaka

Analisa Sentimen merupakan bidang yang mempelajari dan menganalisa opini publik, sentimen, evaluasi, penilaian, sikap dan emosi terhadap suatu produk, pelayanan, organisasi, individu, isu, kejadian, topik dan atribut – atribut yang menyertai [5].

Penelitian mengenai analisa sentimen telah dilakukan oleh Buntoro [6] dengan melakukan analisa dokumen teks twitter untuk menganalisas sentimen pada calon gubernur DKI Jakarta tahun 2017, menggunakan metode Naïve Bayes Classifier. Data yang digunakan adalah cuitan dalam bahasa Indonesia dengan kata kunci AHY, Ahok, Anies dengan dataset sebanyak tiga ratus (300) cuitan. Tingkat akurasi rata – rata yang diperoleh adalah 95%.

Penelitian selanjutnya adalah analisa sentimen pada dokumen twitter dengan KNN dan SVM yang dilakukan oleh [7]. Data yang digunakan pada penelitian tersebut adalah sebanyak seribu (1000) cuitan. Hasil dari penelitian tersebut menunjukkan bahwa variasi algoritma KNN mempunyai performa akurasi yang lebih baik dibandingkan dengan variasi algoritma SVM.

Gülen Toker [4] pada penelitian yang telah dilakukan untuk klasifikasi teks pada bahasa Turki menggunakan metode k-Nearest Neighbor juga menerapkan algoritma *tfidf* (*term frequency – inverse document frequency*) untuk pembobotan, melakukan penghitungan tingkat kemiripan dan menerapkan teknik klasifikasi *k*-Nearest Neighbor. Data yang digunakan untuk pengujian adalah sebanyak 50 dokumen berbeda dan hasil yang diperoleh adalah 46 dokumen benar dan 4 dokumen salah. Tingkat akurasi dari penelitian tersebut adalah sebesar 92%. Faktor – faktor yang mempengaruhi tingkat akurasi diantaranya adalah jumlah data latih, kata – kata yang digunakan pada data latih dan konteks dari data contoh.

Riloff dkk [8] mengajukan menyajikan algoritma bootstrap baru yang secara otomatis mempelajari daftar frasa sentimen positif dan frasa situasi negatif dari tweet sarkastik. Hasil penelitiannya menunjukkan bahwa mengidentifikasi konteks yang kontras menggunakan frase yang dipelajari melalui bootstrap menghasilkan peningkatan daya ingat untuk pengenalan sarkasme.

Sementara itu Farra dkk [9] menyelidiki penambangan sentimen teks Arab di tingkat kalimat dan tingkat dokumen. Penelitian yang ada di pertambangan sentimen Arab masih sangat terbatas. Untuk klasifikasi tingkat kalimat, digunakan dua pendekatan. Yang pertama adalah pendekatan gramatikal baru yang menggunakan penggunaan struktur

umum untuk kalimat bahasa Arab. Pendekatan kedua didasarkan pada orientasi kata semantik dan frekuensi yang sesuai; untuk melakukan ini, kami membangun kamus semantik pembelajaran interaktif yang menyimpan polaritas akar kata-kata yang berbeda dan mengidentifikasi polaritas baru berdasarkan pada akar ini. Untuk klasifikasi tingkat dokumen, digunakan kalimat dari kelas yang dikenal untuk mengklasifikasikan seluruh dokumen, menggunakan pendekatan baru di mana dokumen dibagi secara dinamis menjadi potongan-potongan dan klasifikasi didasarkan pada kontribusi semantik dari potongan yang berbeda dalam dokumen. Pendekatan chunking dinamis ini juga dapat diselidiki untuk penambangan sentimen dalam bahasa lain. Akhirnya, sebuah skema klasifikasi hierarkis yang menggunakan hasil classifier tingkat kalimat sebagai input ke classifier level dokumen diusulkan. Dalam penelitiannya memberikan hasil yang menjanjikan dengan pendekatan tingkat kalimat, dan percobaan tingkat dokumen dengan akurasi tinggi, dapat disimpulkan bahwa layak untuk mengekstraksi sentimen dokumen Arab berdasarkan kelas kalimatnya.

3. Metode

Metode yang digunakan pada penelitian ini adalah sebagai berikut:

a. Pengumpulan data
   Data diambil dari situs goodreads.com secara acak dan hanya mengambil data dengan bahasa Indonesia. Data tinjauan buku terdiri dari data yang mendapatkan satu bintang, dua bintang, tiga bintang, empat bintang dan lima bintang. Tinjauan buku yang digunakan adalah tinjauan dari sembilan (9) judul buku berbeda, diantaranya adalah Laskar Pelangi, Critical Eleven, Sunshine Becomes You, Jingga untuk Matahari dan Sang Pemimpi. Contoh tinjauan positif adalah "Yang jelas ga nyesel baca buku ini karena aku bisa menangis dan tertawa dalam waktu berdekatan". Sedangkan contoh tinjauan negatif adalah "Ceritanya terlalu datar dan gampang ketebak deh, jadi bosan bacanya".
b. Tahap *Preprocessing*
   Tahap *preprocessing* terdiri dari tahap pemisahan kata – kata dalam dokumen atau disebut juga *tokenization*, perbaikan kata tidak baku, dan penghapusan *stopword.* Tahap *tokenization* dilakukan dengan cara memisahkan setiap dokumen menjadi kata – kata, menghilangkan tanda baca, dan mengubah kata dengan huruf besar menjadi huruf kecil.
c. Pembobotan istilah (*term*)
   Metode yang digunakan untuk proses pembobotan adalah dengan menggunakan algoritma *tfidf* (*term frequency – inverse document frequency*). Rumus yang digunakan dalam pembobotan adalah sebagai berikut [10] :

$$wtf(t) = tf * idf(t) \quad (1)$$

Dengan *wtf* adalah bobot untuk setiap istilah, *t* adalah istilah yang dihitung bobotnya. Kemudian *tf* adalah jumlah kemunculan istilah pada satu dokumen. Sedangkan *idf* adalah nilai *inverse document frequency* untuk istilah *t.* Nilai *tf* dihitung dengan cara menghitung jumlah kemunculan setiap kata yang telah dipecah dalam satu dokumen dan dibandingkan dengan jumlah keseluruhan istilah pada

dokumen. Nilai *idf* dihitung dengan rumus sebagai berikut [4]:

$$idf = \log 2 \frac{N}{n} \quad (2)$$

Dengan N adalah jumlah keseluruhan dokumen dan n adalah jumlah dokumen dengan istilah *t.*

d. Penghitungan tingkat kemiripan.
Metode yang digunakan untuk menghitung tingkat kemiripan melibatkan hasil pembobotan yang telah dilakukan pada tahap sebelumnya. Setiap kata yang telah dibobotkan dihitung tingkat kemiripannya terhadap keseluruhan data latih. Rumus yang digunakan adalah sebagai berikut [4]:

$$Sim(X, Dj) = \frac{[\sum ti \in (X \cap Dj)\ xi * dij]}{[\|X\| * \|Dj\|]} \quad (3)$$

X adalah tingkat kesamaan dokumen dengan data latih. *ti* adalah istilah yang ditemukan pada kedua dokumen yang dibandingkan, *xi* dan *dij* adalah bobot dari setiap term yang ditemukan. Penghitungan kemiripan dilakukan berulang untuk semua data latih. Hasil dari penghitungan tingkat kemiripan berfungsi untuk mendapatkan *neighborhood* sehingga dapat dilakukan proses klasifikasi pada tahap selanjutnya.

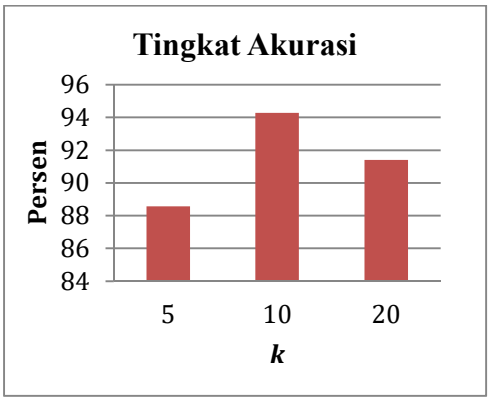
e. Klasifikasi
Setelah mendapatkan nilai bobot untuk setiap dokumen data latih, maka klasifikasi dilakukan dengan menggunakan data uji. Sistem yang digunakan melakukan pembobotan, penghitungan tingkat kemiripan dan klasifikasi dibangun dengan menggunakan bahasa pemrograman PHP dan dijalan pada server web Apache. Nilai k yang digunakan pada proses klasifikasi adalah 5, 10 dan 20.

f. Pengujian
Pengujian dilakukan dengan membandingkan data uji terhadap data latih dan membandingkan secara langsung hasil yang didapatkan dengan nilai aktual.

4. Hasil dan Pembahasan

Hasil dari penelitian yang telah dilakukan dibagi menjadi dua, yaitu hasil pengujian dengan menggunakan sistem yang dibangun sendiri oleh peneliti dan hasil pengujian dengan menggunakan aplikasi Weka 3.8.

Hasil dengan menggunakan sistem yang dibangun sendiri ditunjukkan pada grafik berikut



Gambar 1. Tingkat akurasi dengan menggunakan sistem yang dibangun sendiri

Dari Gambar 1 menunjukkan tingkat akurasi yang didapatkan dengan menggunakan sistem penghitungan dengan bahasa pemrograman PHP adalah 88,57% untuk nilai k = 5. 94,28% untuk nilai k = 10 dan 91,4% untuk nilai k = 20. Pada Tabel 1 dan Tabel 2 ditunjukkan jumlah hasil predikis dengan menggunakan sistem dengan bahasa pemrograman PHP.

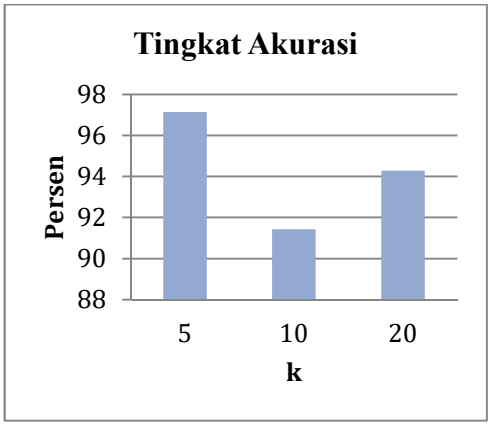
Tabel 1. Tabel Hasil Prediksi untuk Klasifikasi Positif

k	Jumlah Positif	
	Aktual	Prediksi
5	16	18
10	16	18
20	16	19

Tabel 2. Tabel Hasil Prediksi untuk Klasifikasi Negatif

k	Jumlah Negatif	
	Aktual	Prediksi
5	19	17
10	19	17
20	19	16

Pengujian dengan aplikasi Weka 3.8 menggunakan *classifier* iBK dengan *percentage split* sebesar 65%. Tingkat akurasi yang didapatkan dengan menggunakan aplikasi Weka 3.8 ditunjukkan pada Gambar 2. Pada gambar tersebut tingkat akurasi yang didapatkan adalah 97,14% untuk nilai *k*=5, 91,42% untuk nilai *k* = 10 dan 94,28% untuk nilai *k*=20.



Gambar 2. Tingkat akurasi dengan menggunakan aplikasi Weka 3.8

Hasil penghitungan dengan dengan menggunakan aplikasi Weka 3.8 dapat ditunjukkan pada Tabel 3.

Tabel 3. Hasil Prediksi dengan Menggunakan Aplikasi Weka 3.8

k	Jumlah benar	Jumlah salah
5	34	1
10	32	3
20	33	2

Dalam Tabel 3 ditunjukkan bahwa jumlah kesalahan pengklasifikasian bernilai 1 untuk *k = 5*. Kesalahan pengklasifikasian untuk *k* = 10 adalah 3 kesalahan dan kesalahan pengklasifian untuk *k = 20* adalah 2.

5. Penutup

Dari penelitian yang dilakukan dapat disimpulkan bahwa metode klasifikasi k-Nearest Neighbor dan penggabungan algoritma *tfidf* dan penghitungan tingkat kemiripan yang digunakan pada penelitian ini menghasilkan tingkat akurasi yang cukup baik yaitu sekitar 90% untuk nilai *k* dengan variasi 5, 10 dan 20.

Untuk penelitian berikutnya disarankan bahwa langkah *preprocessing* dapat dilakukan dengan mempertimbangkan adanya kata – kata dalam bahasa asing yang dituliskan pada tinjauan buku. Penggunaan data latih yang lebih banyak juga disarankan untuk penelitian berikutnya. Pembagian klasifikasi untuk penelitian berikutnya dapat dikembangkan, sehingga tidak hanya klasifikasi positif dan negatif, tetapi dapat dikembangkan mejadi positif, netral dan negatif.

6. Daftar Pustaka

[1] Asosiasi Penyelenggara Jasa Internet Indonesia, “Penetrasi dan Perilaku Pengguna Internet Indonesia,” 2016.

[2] E. Maslowska, E. C. Malthouse, and V. Viswanathan, “Do customer reviews drive purchase

decisions? The moderating roles of review exposure and price," *Decis. Support Syst.*, vol. 98, pp. 1–9, 2017.

[3] R. F. N. Firmansyah, M. A. Fauzi, and T. Afirianto, "Sentiment Analysis pada Review Aplikasi Mobile Menggunakan Metode Naive Bayes dan Query Expansion," *Doro Ptiik*, vol. 8, no. December, p. 14, 2016.

[4] G. Toker and Ö. Kırmemiş, "Text Categorization Using k-Nearest Neighbor Classification," *Middle East Tech. Univ.*, 2013.

[5] B. Liu, *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers, 2012.

[6] G. A. Buntoro, "Analisis Sentimen Calon Gubernur DKI Jakarta 2017 Di Twitter," *Integer J.*, vol. 2, no. 1, pp. 32–41, 2017.

[7] M. Rezwanul, A. Ali, and A. Rahman, "Sentiment Analysis on Twitter Data using KNN and SVM," *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 6, pp. 19–25, 2017.

[8] E. Riloff, A. Qadir, P. Surve, L. De Silva, N. Gilbert, and R. Huang, "Sarcasm as contrast between a positive sentiment and negative situation," in *EMNLP 2013 - 2013 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, 2013, no. October, pp. 704–714.

[9] N. Farra, E. Challita, R. A. Assi, and H. Hajj, "Sentence-level and document-level sentiment mining for arabic texts," *Proc. - IEEE Int. Conf. Data Mining, ICDM*, pp. 1114–1119, 2010.

[10] I. Rahal, H. Najadat, and W. Perrizo, "A P-tree based K-Nearest Neighbor Text Classifier using Intervalizaion."