SENTIMENT ANALYSIS PADA TEKS BAHASA INDONESIA MENGGUNAKAN *SUPPORT VECTOR MACHINE* (SVM) DAN *K-NEAREST NEIGHBOR* (K-NN)

Syahfitri Kartika Lidya[1], Opim Salim Sitompul[2], Syahril Efendi[3]
[123]*Program Studi Magister Teknik Informatika, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Sumatera Utara*
Jl. Doktor Mansyur 9 Medan 20155
Telp. (061) 8210122
E-mail: syahfitrikl@gmail.com, opim@usu.ac.id, syahril1@usu.ac.id

ABSTRAKS

Analisis Sentimen adalah proses menganalisis, memahami, dan mengklasifikasi pendapat, evaluasi, penilaian, sikap, dan emosi terhadap suatu entitas seperti produk, jasa, organisasi, individu, peristiwa, topik, secara otomatis untuk mendapatkan informasi. Penelitian ini menggunakan teks Bahasa Indonesia yang terdapat di website berupa artikel berita, kemudian metode K-Nearest Neighbor akan mengklasifikasi langsung pada data pembelajaran agar dapat menentukan model yang akan dibentuk oleh metode Support Vector Machine untuk menentukan kategori dari data baru yang ingin ditentukan kategori tekstual, yaitu kelas sentimen positif, negatif dan netral. Berdasarkan seluruh hasil pengujian, bahwa pengaruh nilai k pada k-fold cross validation yang terlalu kecil menghasilkan akurasi yang rendah, sedangkan nilai k yang terlalu besar menghasilkan nilai akurasi yang besar, kemudian Pengaruh nilai k pada K-NN terhadap akurasi, jika n memiliki akurasi rendah pada saat nilai k kecil. Hal ini dikarenakan, data yang masuk pada k tetangga terdekat terlalu sedikit dan tidak dapat merepresentasikan kelas pada data uji.

Kata Kunci: Sentiment, Analysis, SVM, K-NN

ABSTRACT

Sentiment analysis is the process of analyzing, understanding, and classifying opinions, evaluations, assessments, attitudes, and emotions to an entity such as products, services, organizations, individuals, events, topics, automatically to obtain information. This study uses Indonesian text contained on the website in the form of news articles, then the K-Nearest Neighbor method will classify directly to the learning data in order to determine the model that will be established by Support Vector Machine method to determine the category of the new data to be determined textual categories, namely the class of positive sentiment, negative and neutral. Based on the test results, that influence the value of k in k-fold cross validation is too small yield low accuracy, while the value of k is too large produce great accuracy value, then the value of k effect on K-NN for accuracy, if n has an accuracy lower when the value of k is small. This is because, the data are entered in the k nearest neighbors too little and can not represent a class on test data.

Keyword: Sentiment, Analysis, SVM, K-NN

1. PENDAHULUAN

1.1 Latar Belakang

Besarnya pengaruh dan manfaat dari *Sentiment Analysis*, menyebabkan penelitian ataupun aplikasi mengenai *Sentiment Analysis* berkembang pesat, bahkan di Amerika ada kurang lebih 20-30 perusahaan menggunakan *Sentiment Analysis* untuk mendapatkan informasi tentang sentimen masyarakat terhadap pelayanan perusahaan (Sumartini, 2011).

Adapun penelitian-penelitian terdahulu yang terkait dengan *Sentiment Analysis*, antara lain adalah penelitian (Abbasi *et al*, 2008) mendeteksi situs website palsu atau asli dengan klasifikasi artikel berita pada website. Penelitian (Han *et al*, 2013) menganalisis sentimen pada teks twitter, dengan menggunakan karakter bahasa n-gram model dan SVM untuk mengatasi variasi leksikal tinggi dalam teks Twitter. Penelitian (Vinodhini & Chandrasekaran, 2012) mengembangkan sistem yang dapat mengidentifikasi dan mengklasifikasikan sentimen masyarakat untuk memprediksi produk yang menarik dalam pemasaran.

Support Vector Machine (SVM) dan *K-Nearest Neighbor* (K-NN) dapat melakukan menganalisis dengan cara belajar dari sekumpulan contoh dokumen yang telah

diklasifikasi sebelumnya. Keuntungan dari metode ini adalah dapat menghemat waktu kerja dan memperoleh hasil yang lebih baik, tetapi pada *Support Vector Machine* untuk ekstraksi informasi dari dokumen teks tidak terstruktur karena jumlah fitur jauh lebih besar daripada jumlah sampel, metode ini memiliki performansi yang kurang baik, terhadap domain tertentu, oleh karena itu perlunya *K-Nearest Neighbor* untuk meminimalkan jumlah fitur yang akan digunakan untuk analisis sehingga lebih akurat. Kemudian SVM tidak memperhatikan distribusi data, karena hanya berdasarkan kelas yang memiliki pola berbeda dan dipisahkan oleh fungsi pemisah, sehingga analisis yang dihasilkan kemungkinan salah, sehingga K-NN akan mendistribusikan data tersebut dengan berdasarkan jarak data ke beberapa data terdekat, sehingga analisis yang dihasilkan lebih akurat. Penelitian ini diharapkan dapat mempercepat upaya mendapatkan informasi yang akurat tentang sentimen pemberitaan media massa pada suatu hal.

2. TINJAUAN PUSTAKA

2.1 *Text Mining*

Text mining, pada proses mengambil informasi dari teks. Informasi biasanya diperoleh melalui peramalan pola dan kecenderungan pembelajaran pola statistik. *Text mining* yaitu parsing, bersama dengan penambahan beberapa fitur linguistik turunan dan penghilangan beberapa diantaranya, dan penyisipan *subsequent* ke dalam database, menentukan poladalam data terstruktur, dan akhirnya mengevaluasi dan menginterpretasi output, *text mining* biasanya mengacu ke beberapa kombinasi relevansi, kebaruan, dan *interestingness*. Proses *text mining* yang khas meliputi kategorisasi teks, *text clustering*, ekstraksi konsep/entitas, produksi taksonomi granular, *sentiment analysis*, penyimpulan dokumen, dan pemodelan relasi entitas yaitu, pembelajaran hubungan antara entitas (Bridge, 2011).

2.2 *Sentiment Analysis*

Tugas dasar dalam analisis sentimen adalah mengelompokkan polaritas dari teks yang ada dalam dokumen, kalimat, atau fitur/tingkat aspek – apakah pendapat yang dikemukakan dalam dokumen, kalimat atau fitur entitas/aspek Ekspresi atau *sentiment* mengacu pada fokus topik tertentu, pernyataan pada satu topik mungkin akan berbeda makna dengan pernyataan yang sama pada *subject* yang berbeda. Sebagai contoh, adalah hal yang baik untuk mengatakan alur film tidak terprediksi, tapi adalah hal yang tidak baik jika 'tidak terprediksi' dinyatakan pada kemudi dari kendaraan. Bahkan pada produk tertentu, kata-kata yang sama dapat menggambarkan makna kebalikan, contoh adalah hal yang buruk untuk waktu *start-up* pada kamera digital jika dinyatakan "lama", namun jika "lama" dinyatakan pada usia baterai maka akan menjadi hal positif. Oleh karena itu pada beberapa penelitian, terutama pada review produk, pekerjaan didahului dengan menentukan elemen dari sebuah produk yang sedang dibicarakan sebelum memulai proses *opinion mining* (Ian *et al*, 2011).

2.3 *Support Vector Machine* (SVM)

Support Vector Machines (SVM) adalah seperangkat metode pembelajaran terbimbing yang menganalisis data dan mengenali pola, digunakan untuk klasifikasi dan analisis regresi. Algoritma SVM asli diciptakan oleh Vladimir Vapnik dan turunan standar saat ini *Soft Margin* (Cortes & Vapnik, 1995). SVM standar mengambil himpunan data input, dan memprediksi, untuk setiap masukan yang diberikan, kemungkinan masukan adalah anggota dari salah satu kelas dari dua kelas yang ada, yang membuat sebuah SVM sebagai penggolong nonprobabilistik linier biner. Karena SVM adalah sebuah pengklasifikasi, kemudian diberi suatu himpunan pelatihan, masing-masing ditandai sebagai milik salah satu dari dua kategori, suatu algoritma pelatihan SVM membangun sebuah model yang memprediksi apakah data yang baru jatuh ke dalam suatu kategori atau yang lain

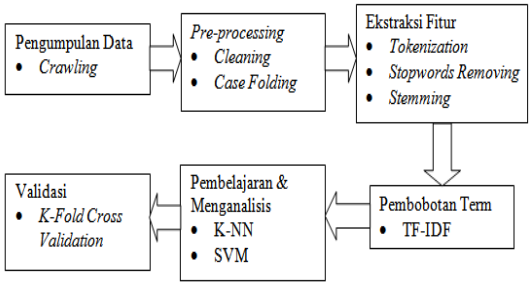
2.4 *K-Nearest Neighbor* (K-NN)

Tujuan dari algoritma ini adalah mengklasifikasikan obyek berdasarkan atribut dan *training sample. Clasifier* tidak menggunakan apapun untuk dicocokkan dan hanya berdasarkan pada memori. Diberikan titik *query*, akan ditemukan sejumlah *k* obyek atau (titik *training*) yang paling dekat dengan titik *query.* Klasifikasi menggunakan *voting* terbanyak diantara klasifikasi dari k obyek. Algoritma *K-Nearest Neighbor* (K-NN) menggunakan klasifikasi ketetanggaan sebagai nilai prediksi dari *query instance* yang baru.

3. METODOLOGI PENELITIAN

3.1 Proses Analisis Sentimen pada Dokumen

Berikut ini adalah metode yang digunakan untuk proses analisis sentimen yang digunakan dalam penelitian ini.



Gambar 1. Proses Analisis Sentimen

3.1.1 Pengumpulan Data

Penelitian dilakukan untuk teks artikel berita berbahasa indonesia. Data terbagi atas opini positif, opini negatif dan opine netral. Sebagian data akan dijadikan data latih dan sebagian sebagai data uji untuk digunakan metode *support vector machine* dan *k-nearest neighbor*.

Data yang digunakan artikel berita teks berbahasa indonesia diambil dari beberapa *website* berita yang terbaik di Indonesia, seperti detik.com, tribunnews.com, metrotvnews.com, kompas.com, dan website berita di Indonesia lainnya. Penelitian ini menggunakan kumpulan artikel berita yang dibentuk dalam file dokumen yang telah di *crawling* menggunakan *crawler4j* yaitu *open source web crawler for java* (Ganjisaffar, 2013). Data terbagi menjadi opini positif, opini negatif, dan opini netral.

3.1.2 *Pre-Processing*

Tahapan yang dilakukan pada dokumen *pre-processsing* adalah *Cleaning*, untuk mengurangi *noise* saat menganalisis sentimen. Proses ini menggunakan *jsoup 1.7.4-Snapshot API* (Hedley, 2013) dan *Case Folding*, pada proses ini akan menggunakan *S-Space package* (Jurgens, 2009).

3.1.3 Ekstraksi Fitur

Proses ekstraksi fitur yang akan digunakan sebagai dasar proses klasifikasi, *Tokenization*, penelitian ini menggunakan *S-Space package* (Jurgens, 2009), dengan fitur yang digunakan dalam memecah text adalah *unigram* yaitu token yang terdiri hanya satu kata. Dalam fitur unigram, kata dalam dokumen direpresentasikan ke dalam bentuk vektor, dan tiap kata dihitung sebagai satu fitur. *Stopwords Removing*, pada tahap ini menghilangkan kata tidak penting sesuai kamus data yang digunakan, agar memperbesar akurasi dari pembobotan term. *Stemming*, pada penelitian ini digunakan algoritma *stemming* Bahasa Indonesia yaitu algoritma Nazief & Adriani yang menyimpulkan sebuah kata dasar dapat ditambahkan imbuhan. (Nazief & Adriani, 1996)

3.1.4 Pembobotan Term

Pembobotan term merupakan *term documents matrix* yang representasi kumpulan dokumen yang digunakan untuk melakukan proses klasifikasi dokumen teks. Pada penelitian ini akan digunakan metode TF-IDF sebagai proses pembobotan, yaitu akan dilakukan pembobotan pada tiap term berdasarkan tingkat kepentingan tersebut di dalam sekumpulan dokumen masukan.

3.1.5 Pembelajaran dan Analisis

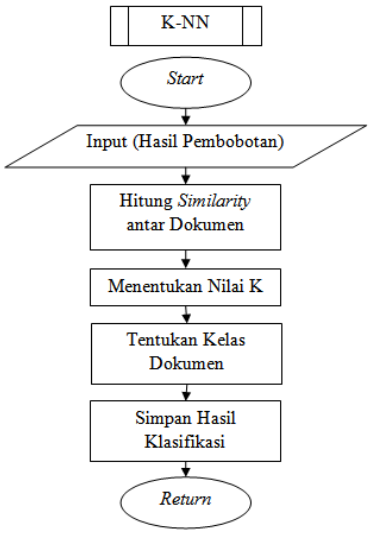
Pada proses ini akan dilakukan menggunakan K-NN dan SVM. Setelah dokumen dalam bentuk matriks kata-dokumen dan telah diberi pembobotan TF-IDF, maka proses selanjutnya K-NN menganalisis, yaitu setiap dokumen akan diberi tanda positif atau negatif, jika jumlah positif < jumlah negatif maka skor sentimen:

$$-1 \times \left(\frac{Jumlah\ Negatif}{Jumlah\ Kata}\right) \quad (1)$$

Jika, jumlah positif > jumlah negatif maka skor sentimen :

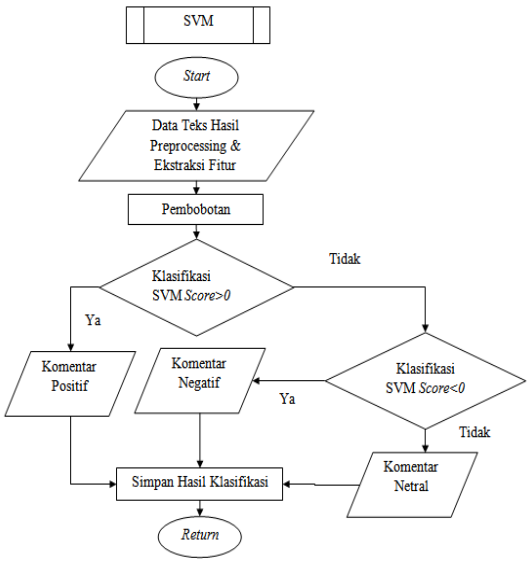
$$\left(\frac{Jumlah\ Positif}{Jumlah\ Kata}\right) \quad (2)$$

Jika selain kriteria diatas, maka sentimen adalah 0 atau disebut netral. (Khushboo *et al*, 2012) Kemudian akan dilakukan proses *K-Nearest neighbor* untuk menghitung nilai kemiripan antara data uji dengan semua data latih pada dokumen artikel dengan menggunakan metode *cosine similarity*.



Gambar 2. Diagram Alir *K-NN*

Kemudian, proses analisis menggunakan SVM dimulai mengubah text menjadi data vektor. Vektor dalam penelitian ini memiliki dua komponen yaitu dimensi (*word id*) dan bobot. Bobot ini adalah nilai tf-idf, tujuan dari model ruang vektor digunakan untuk memberikan setiap kata dalam dokumen sebuah ID (dimensi) dan sebuah bobot berdasarkan seberapa penting keberadaannya dalam dokumen (posisi dokumen dalam dimensi itu). SVM mencoba untuk menemukan garis yang terbaik membagi dua kelas, dan kemudian mengklasifikasikan dokumen uji berdasarkan di sisi mana dari garis tersebut mereka muncul. Diagram alir proses klasifikasi dengan SVM ditunjukkan oleh gambar 3.



Gambar 3. Diagram Alir *SVM*

3.1.6 Validasi dengan *K-Fold Cross Validation*

Pada tahap validasi dengan melihat akurasi (ketepatan), suatu dokumen yang telah direpresentasikan sebagai vektor dalam ruang *term*, dimana *k* adalah ukuran dari kumpulan *term*.

Pada penelitian ini digunakan *k-fold cross validation* untuk menghilangkan bias kata. *K-fold cross validation* membagi kumpulan dokumen menjadi *k* bagian. Dalam satu set percobaan akan dilakukan *k* buah percobaan klasifikasi dokumen dengan tiap percobaan menggunakan satu bagian sebagai data *testing, (k-1)/2* bagian sebagai *labeled documents*, dan *(k-1)/2* bagian lainnya sebagai *unlabeled documents* yang akan ditukar setiap percobaan sebanyak k kali. Kumpulan dokumen yang dimiliki terlebih dahulu diacak urutannya sebelum dimasukkan ke dalam sebuah *fold.* Hal ini dilakukan untuk menghindari pengelompokkan dokumen-dokumen yang berasal dari satu kategori tertentu pada sebuah *fold*

4. PENGUJIAN DAN PEMBAHASAN

Dokumen diproses untuk menyeragamkan bentuk kata, menghilangkan karakter-karakter selain huruf, dan mengurangi volume kosakata, proses ini terdiri dari lima tahapan. Kelima tahapan tersebut adalah proses *cleaning*, *case folding*, *tokenization*, *stopwords removing*, *stemming* dan pembuatan program analisis dokumen teks dengan menggunakan K-NN dan SVM.

Penelitian ini menggunakan spesifikasi perangkat keras, seperti pada tabel 1 berikut:

Tabel 1. Spesifikasi Perangkat Keras

Perangkat keras	*Spesifikasi*
Sistem Operasi	Ubuntu 14.04 LTS
Tipe Sistem	32 – bit
Prosesor	Intel® Core™ 2 Duo CPU T5550 @ 1.83GHz x 2
Memori (RAM)	992,4 MiB
Harddisk	30,0 GB
Perangkat Lunak	• NetBeans IDE 7.4 • MongoDB Manual 2.6.4 • Crawler4j • Jsoup 1.7.4-Snapshot API • S-Space Package • WEKA (*Waikato Environment for Knowledge Analysis*) Version 3.6.5

4.1 Persiapan Pengujian

Pada tahap persiapan ini pengambilan *data sample* masing-masing data untuk melakukan fitur baik dataset positif, negatif maupun dataset netral dilakukan dengan mengcrawling data teks dokumen menggunakan *crawler4j*. Dari metode crawling data, menggunakan *crawler4j* perlu dilakukan beberapa tahapan seperti melakukan installasi pada linux dan kemudian membangun beberapa program untuk melakukan eksekusi perintah, di dalam program ini dapat diberikan berupa alamat url yang diproses. Setelah dilakukan *crawling data* dari berbagai situs yang membahas topic yang telah ditentukan, data akan disimpan *k-nearest neighbor* adalah file teks dengan setiap *record* berisi sebuah kalimat dengan sentimen positif, negatif, atau netral.

Proses ini membaca tiap kata yang ada dalam *file* data latih dan mengelompokkannya sebagai *variable index*. Proses dilanjutkan dengan menghilangkan *stopwords* lalu mengubah padanan kata untuk kalimat yang memiliki unsur pembalik seperti kata ”tidak”. Untuk data yang masuk ke dalam mesin *support vector machine* maka terlebih dahulu diubah ke dalam bentuk data vektor, yaitu dilakukan dengan membaca kata satu persatu dan menghitung nilai tf-idf. Nilai tf-idf adalah kemunculan kata (*term frequency*) dalam kalimat dikalikan log jumlah *record* dibagi jumlah *record* yang mengandung kata yang dimaksud.

4.2 Hasil dan Pembahasan Percobaan dengan Metode K-NN dan SVM untuk Data Berbahasa Indonesia

Proses pembelajaran dan analisis dengan metode K-NN dan SVM menggunakan data yang sama untuk kedua metode. Pada penelitian ini dataset yang digunakan dengan tema pemilu 2014 di Indonesia, sehingga keyword yang digunakan adalah yang berhubungan dengan pemilu 2014 di Indonesia, dokumen hasil crawling 6025 terdiri dari 62545 kata, kemudian jumlah data uji positif, negatif dan netral tidak sama karena setelah melakukan crawling data tidak dipilih secara manual, sehingga hasil crawling data langsung diproses oleh sistem, dan sistem akan memilih menjadi 3 kategori yaitu data yang mana yang termasuk positif, negatif, dan netral. Hasil percobaan dengan K-NN dapat dilihat pada tabel 2, 3 dan 4.

Tabel 2. Persentase (%) Analisis Sentimen K-NN

No	*Keyword*	*Positif (%)*	*Negatif (%)*	*Netral (%)*
1	Prabowo	95.20 %	3.8 %	1 %
2	Jokowi	93.15 %	5.1 %	1.75 %
3	Pemilu 2014	77.80 %	15.2 %	7 %
4	Hatta Rajasa	89.16 %	5.2 %	5.64 %
5	Jusuf Kalla	90.35 %	8 %	1.62 %

Tabel 3. Jumlah Dokumen Hasil Analisis Sentimen K-NN

No	*Keyword*	*Jumlah Data Uji Positif*	*Jumlah Data Uji Negatif*	*Jumlah Data Uji Netral*	*Jumlah Data Hasil Uji Positif*	*Jumlah Data Hasil Uji Negatif*	*Jumlah Data Hasil Uji Netral*
1	Prabowo	600	313	87	845	116	39
2	Jokowi	600	274	126	800	93	107
3	Pemilu 2014	600	306	94	910	18	72
4	Hatta Rajasa	600	302	98	731	165	104
5	Jusuf Kalla	600	166	73	729	44	66

Tabel 4. Akurasi dan Waktu Proses K-NN dalam Menganalisis Sentimen

No	*Keyword*	*Akurasi (%)*	*Waktu Proses (Detik)*
1	Prabowo	62.50 %	33.81 detik
2	Jokowi	63.10 %	32.79 detik
3	Pemilu 2014	60.30 %	75.21 detik
4	Hatta Rajasa	64.70 %	106.51 detik
5	Jusuf Kalla	72.47 %	100.06 detik

Dari hasil percobaan pada tabel 2 dan 3 bahwa dengan jumlah data uji positif yang lebih besar maka hasil analisis akan cenderung menunjukkan sebagai opini positif. Demikian pula sebaliknya, jika data uji negatif lebih besar hasil analisis akan cenderung sebagai opini negatif. Pengolahan kata pada penelitian ini adalah kata bersifat independen tanpa memperhatikan kebergantungan satu dengan yang lainnya. K-NN memberikan performa yang baik untuk data yang independen, dapat dilihat hasil uji K-NN jika data uji lebih kecil dari data latih maka akurasi yang dihasilkan lebih baik, tetapi jika data uji lebih besar dari data latih maka akurasi yang dihasilkan kurang baik.

Percobaan dengan data yang sama menggunakan metode SVM, dilihat pada tabel 5, 6 dan 7

Tabel 5. Persentase (%) Analisis Sentimen SVM

No	*Keyword*	*Positif (%)*	*Negatif (%)*	*Netral (%)*
1	Prabowo	99.10 %	0.4 %	0.5 %
2	Jokowi	98.15 %	0.85 %	1 %
3	Pemilu 2014	95.80 %	4 %	0.2 %
4	Hatta Rajasa	94.20 %	2.8 %	3 %
5	Jusuf Kalla	95.75 %	2.25 %	2 %

Tabel 6. Jumlah Dokumen Hasil Analisis Sentimen SVM

No	*Keyword*	*Jumlah Data Uji Positif*	*Jumlah Data Uji Negatif*	*Jumlah Data Uji Netral*	*Jumlah Data Hasil Uji Positif*	*Jumlah Data Hasil Uji Negatif*	*Jumlah Data Hasil Uji Netral*
1	Prabowo	600	313	87	851	148	1
2	Jokowi	600	274	126	840	160	0
3	Pemilu 2014	600	306	94	775	224	1
4	Hatta Rajasa	600	302	98	780	220	0
5	Jusuf Kalla	600	166	73	633	205	1

Tabel 7. Akurasi dan Waktu Proses SVM dalam Menganalisis Sentimen

No	*Keyword*	*Akurasi (%)*	*Waktu Proses (Detik)*
1	Prabowo	66.77 %	57.80 detik
2	Jokowi	66.10 %	41.75 detik
3	Pemilu 2014	66.90 %	106.42 detik
4	Hatta Rajasa	70.10 %	103.86 detik
5	Jusuf Kalla	73.54 %	78.77 detik

Dari hasil percobaan yang ditunjukkan pada tabel 5 dan 6 bahwa metode SVM memberikan proses yang baik karena pola metode SVM mengacu pada ketersediaan *support vector* untuk membentuk *hyperlane*. Data opini berbahasa Indonesia pada penelitian ini cenderung terdistribusi linier dimana SVM memiliki keunggulan dalam menganalisis data yang terdistribusi linier. SVM memiliki rata-rata tingkat akurasi yang baik. Kemudian banyaknya data uji tidak berpengaruh banyak terhadap hasil generalisasinya. Akan tetapi hasilnya akan lebih baik apabila banyaknya data pembelajaran lebih banyak atau sama dengan banyaknya data uji. Dari hasil simulasi komputer tersebut diperoleh bahwa ternyata SVM memberikan hasil memprediksi data uji dengan kemampuan yang baik.

Semua data matriks diuji dengan fungsi kernel linear, formula SVM mentransformasikan data ke dalam dimensi ruang fitur dengan menggunakan fungsi kernel. Proses pengujian ini bertujuan membangun model dan menghitung tingkat akurasi SVM dalam memprediksi data uji.

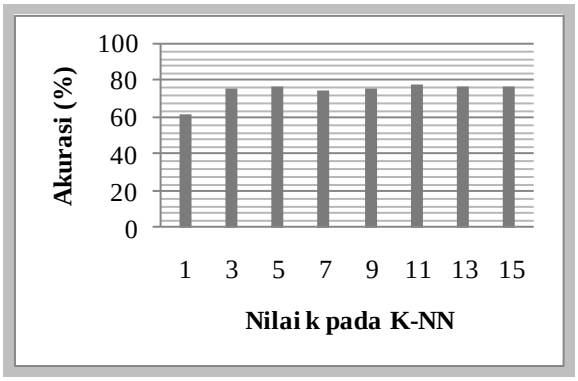
Pelatihan SVM membutuhkan parameter sesuai dengan kernelnya. Setiap proses pelatihan SVM yang menggunakan fungsi kernel diperlukan parameter terbaik untuk mendapatkan akurasi yang terbaik jika mencapai rataan nilai tertinggi. Namun pada penelitian ini didapatkan rataan nilai hampir sama di setiap iterasi pemodelan sehingga pengambilan parameter dilakukan pada nilai akurasi tertinggi pertama.

Hasil proses algoritma K-NN dan SVM dalam menganalisis sentimen pada dokumen. Jumlah dokumen yang digunakan untuk dianalisis adalah 1000 dokumen berupa sentimen yang masih acak.

SVM memiliki akurasi yang baik untuk menganalisis teks, karena pada penelitian ini SVM menggunakan kernel linear yaitu kernel yang paling sederhana dari semua fungsi kernel. Kernel ini memiliki keunggulan dalam kasus analisis teks, tetapi SVM Sulit jika digunakan pada jumlah data berskala besar, dalam hal ini dimaksudkan dengan jumlah sampel yang diolah, sedangkan pada K-NN perlu menentukan nilai dari parameter *k* (jumlah dari tetangga terdekat), *training* berdasarkan jenis jarak yang digunakan dan atribut mana yang akan digunakan untuk mendapatkan hasil terbaik, dan biaya komputasi cukup tinggi karena diperlukan perhitungan jarak dari tiap *query instance* pada keseluruhan *training sample*, tetapi K-NN memiliki keunggulan terhadap data uji yang memiliki banyak *noise* dan efektif apabila jumlah data uji banyak.

4.3 Pengaruh Pemilihan Nilai K pada K-NN

Pada pengujian ini akan dianalisis pengaruh parameter nilai k terhadap tingkat keberhasilan proses K-NN untuk menganalisis dokumen. Untuk pengujian digunakan nilai k yaitu 1, 3, 5, 7, 9, 11, 13, dan 15.



Gambar 4. Pengaruh Nilai k pada K-NN terhadap Akurasi

Nilai k pada K-NN yang terbaik tergantung pada data. Nilai k berpengaruh terhadap tingkat akurasi. Persentase akurasi yang tertinggi berada pada nilai k = 11 sampai k = 15 dengan nilai akurasi rata-rata 77,97 %. Persentase akurasi terendah berada pada nilai k = 1 sampai k = 9 dengan nilai akurasi rata-rata 70,90 %. Semua skenario pengujian pertama menggambarkan pola yang sama, metode K-NN memiliki akurasi rendah pada saat nilai k kecil. Hal ini dikarenakan pada k kecil, data yang masuk pada k tetangga terdekat terlalu sedikit dan belum bias merepresentasikan kelas pada data uji. Dokumen diekstrak menggunakan kata kunci positif, negatif, dan netral yang dilihat dari kamus dan dihitung jumlah kata tersebut dalam suatu dokumen. Hal tersebut menyebabkan adanya irisan fitur pada setiap data yang dapat merepresentasikan suatu dokumen dalam sentimen positif , negatif dan netral.

4.4 Akurasi *K-Fold Cross Validation*

Ada beberapa teknik untuk mengestimasi tingkat kesalahan, salah satunya adalah *k-fold cross validation*. Cara kerjanya adalah melakukan pengelompokkan antara data latih dan data uji, kemudian dilakukan proses pengujian yang diulang sebanyak k kali. Hasil pengujian itu kemudian dirata-ratakan untuk menghasilkan sebuah nilai. Pengaruh nilai k yang terlalu kecil adalah dapat menghasilkan akurasi yang rendah. Hal ini disebabkan dengan kecilnya nilai k, maka analisis akan lebih terpengaruh oleh *noise*. Sedangkan, pengaruh nilai k yang terlalu besar adalah dapat menghasilkan akurasi yang besar. Hal ini disebabkan dengan besarnya nilai k, maka analisis akan lebih terpengaruh oleh *noise*.

Hasil pengujian dari *K-Fold Cross Validation* yang dilakukan sebanyak 4 kali, yaitu 4 fold, 5 fold, 6 fold, 10 fold, 11 fold, 12 fold, 13 fold, 14 fold dan 15 fold.

Tabel 8. Hasil Rata-Rata Semua *Fold Cross Validation* pada K-NN dan SVM terhadap Akurasi

Fold Cross Validation	*Akurasi K-NN (%)*	*Akurasi SVM (%)*
4	59.50 %	66.90 %
5	60.10 %	67.90 %
6	60.30 %	66.90 %
10	60.30 %	67.90 %
11	60.30 %	67.90 %
12	60.30 %	67.90 %
13	60.30 %	67.90 %
14	60,20 %	67.90 %
15	60,20 %	67.90 %

Metode pengujian dengan *K-Fold Cross Validation* dengan nilai k = 4, 5, 6, 10, 11, 12, 13, 14 dan 15. Dari hasil pengujian dilakukan, bahwa pada nilai k = 10 adalah nilai yang optimal terhadap akurasi sebesar 67,90 % karena hasil dari berbagai percobaan yang ekstensif dan pembuktian teoritis, menunjukkan bahwa *10-fold cross validation* adalah pilihan terbaik untuk mendapatkan hasil validasi yang akurat *10-fold cross validation* yaitu mengulang pengujian sebanyak 10 kali dan hasil pengukuran adalah nilai rata-rata dari 10 kali pengujian.

5. KESIMPULAN

Adapun kesimpulan dari penelitian ini, yaitu :

1. Jumlah dokumen positif, negatif dan netral hasil dari *crawling* tidak sama banyak, karena dataset yang digunakan website berita, data yang keadaannya masih acak, tidak dipilih secara manual dokumen positif, negatif dan netral oleh user.
2. Waktu proses pada SVM lebih lama daripada K-NN, karena pada saat validasi menggunakan *K-Fold Cross Validation*, matriks vektor pada SVM cukup besar sehingga iterasi untuk validasi cukup lama.
3. Bahwa pengujian pada *k-fold cross validation* dilakukan dengan nilai k = 10 adalah pilihan yang terbaik untuk mendapatkan hasil validasi yang akurat karena pengujian dilakukan sebanyak 10 kali dan kemudian hasil pengukuran dari nilai rata-rata dari 10 kali pengujian.
4. Pengaruh nilai k pada *k-fold cross validation* yang terlalu kecil

menghasilkan akurasi yang rendah, sedangkan nilai k yang terlalu besar menghasilkan nilai akurasi yang besar. Karena analisis tergantung oleh *noise*.

5. Besarnya data latih tidak meningkatkan akurasi sistem. Hal ini dikarenakan sistem menggunakan kamus untuk merepresentasikan fitur suatu dokumen yang menyebabkan adanya irisan fitur pada setiap data yang dapat merepresentasikan suatu dokumen dalam sentimen positif, negatif dan netral.
6. Metode K-Nearest Neighbor (K-NN) dan Support Vector Machine (SVM) dapat diterapkan pada proses analisis sentimen berbahasa Indonesia. Sebelum dilakukan proses analisis, dokumen melalui berbagai tahapan meliputi *pre-processing*, ekstraksi fitur. Dokumen juga diekstrak menggunakan kata kunci positif, negatif dan netral, yang dilihat dari kamus dan dihitung jumlah kata tersebut dalam suatu dokumen.
7. Pengaruh nilai k pada K-NN terhadap akurasi, jika n memiliki akurasi rendah pada saat nilai k kecil. Hal ini dikarenakan pada k kecil, data yang masuk pada k tetangga terdekat terlalu sedikit dan belum bisa merepresentasikan kelas pada data uji.

PUSTAKA

Abbasi, Ahmed. Zhang, Zhu & Chun, Hsinchun. 2008. A Statistical Learning Based System for Fake Website Detection. *The Workshop on Secure Knowledge Management*. Dallas: Texas.

Bridge, C. 2011. Unstructured Data and the 80 Percent Rule. (Online) http://www.clarabridge.com/default.aspx?tabid=137&ModuleID=635&ArticleID=551 (5 Nov 2013).

Cortes, C. & Vapnik, V. 1995. *Support-Vector Networks Machine Learning, 20*. (Online). http://www.springerlink.com/content/k238jx04hm87j80g/. (8 Nov 2013).

Ganjisaffar, Yasser. 2013. *crawler4j : open source web crawler for java*. (Online). https://code.google.com/p/crawler4j/(1 Maret 2014)

Han, qi. Guo, junfei & Schuetze, hinrich. 2013. Combining an SVM Classifier and Character N-gram Language Models for Sentiment Analysis on Twitter Text. *Association for Computational Linguistics: Human Language Technologies.*

Hedley, Jonathan. 2013. *jsoup 1.7.4-Snapshot API* : *Open Source Java HTML parser that makes sense of real-world HTML soup.* (Online). http://jsoup.org/apidocs/ (13 Mei 2014)

Ian, H,Witten. Frank, Eibe & Mark A, Hall. 2011. Data mining: Practical Machine Learning Tools and Techniques 3rd Edition. Elsevier.

Jurgens, David. 2009. *S-Space : free software*. (Online). http://github.com. (13 Mei 2014)

Khushboo, N, Trivedi. Swati, K. Vekariya. & Prof. Shailendra, Mishra. 2012. Mining of Sentences Level Opinion Using Supervised Term Weighted Approach of Naïve Bayesian Algorithm. *int. J. Computer Technology & Applications*, Volume 3, pp. 987-991.

Nazief, B. A. A. & Adriani, M. 1996. *Confix Stripping: Approach to Stemming Algorithm for Bahasa Indonesia*. Internal Publication, Faculty of Computer Science, Jakarta: University of Indonesia.

Sumartini, Ni Wayan. 2011. *Text Mining Classifier dengan Metode Naïve Bayes dan Support Vector Machines untuk Sentiment Analysis*. Tesis. Universitas Udayana.

Vinodhini, G. Chandrasekaran, RM. 2012. Sentiment Analysis and Opinion Mining: A Survey. *International Journal of Advanced Research in Computer Science and Software Engineering*: pp. 283-293.