TESIS

TEXT MINING DENGAN METODE *NAÏVE BAYES CLASSIFIER* DAN *SUPPORT VECTOR MACHINES* UNTUK *SENTIMENT ANALYSIS*

NI WAYAN SUMARTINI SARASWATI

PROGRAM PASCASARJANA
UNIVERSITAS UDAYANA
DENPASAR
2011

TESIS

TEXT MINING DENGAN METODE *NAÏVE BAYES CLASSIFIER* DAN *SUPPORT VECTOR MACHINES* UNTUK *SENTIMENT ANALYSIS*

NI WAYAN SUMARTINI SARASWATI
NIM 0991761024

PROGRAM MAGISTER
PROGRAM STUDI TEKNIK ELEKTRO
PROGRAM PASCASARJANA
UNIVERSITAS UDAYANA
DENPASAR
2011

TEXT MINING DENGAN METODE *NAÏVE BAYES CLASSIFIER* DAN *SUPPORT VECTOR MACHINES* UNTUK *SENTIMENT ANALYSIS*

Tesis untuk Memperoleh Gelar Magister

Pada Program Magister, Program Studi Teknik Elektro,

Program Pascasarjana Universitas Udayana

NI WAYAN SUMARTINI SARASWATI
NIM 0991761024

PROGRAM MAGISTER
PROGRAM STUDI TEKNIK ELEKTRO
PROGRAM PASCASARJANA
UNIVERSITAS UDAYANA
DENPASAR
2011

Lembar Pengesahan

TESIS INI TELAH DISETUJUI

TANGGAL 20 JULI 2011

Pembimbing I,	Pembimbing II,
DR. I. Ketut Gede Darma Putra, S.Kom., MT NIP. 19740424 199903 1 003	Ni Made Ary Esta Dewi Wirastuti, ST., M.Sc., Ph.D NIP. 19760327 200112 2 001

Mengetahui,

Ketua Program Magister Program Studi Teknik Elektro Program Pascasarjana Universitas Udayana	Direktur Program Pascasarjana Universitas Udayana
Prof. Ir. Ida Ayu Dwi Giriantari, M.Eng.Sc.,Ph.D NIP. 19651213 199103 2 001	Prof. Dr. dr. A.A. Raka Sudewi, Sp.S(K) NIP. 19590215 198510 2 001

Tesis Ini Telah Diuji pada

Tanggal 18 Juli 2011

Panitia Penguji Tesis Berdasarkan SK Rektor Universitas Udayana,

No. : 091/UN14.4/TU/TE/2011

Ketua : DR. I. Ketut Gede Darma Putra, S.Kom., MT

Anggota :

1. Ni Made Ary Esta Dewi Wirastuti, ST., M.Sc., Ph.D
2. Prof. Ir. Rukmi Sari Hartati, MT., PhD
3. Wayan Gede Ariastina, ST., MEngSc., PhD
4. Ir. Linawati, MengSc., PhD

PERNYATAAN KEASLIAN KARYA TULIS TESIS

Dengan ini saya menyatakan bahwa dalam tesis ini tidak terdapat karya tulis yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu perguruan tinggi, dan sepanjang pengetahuan saya tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan orang lain, kecuali secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila kemudian hari terbukti bahwa saya melakukan tindakan menyalin atau meniru tulisan orang lain sebagai hasil pemikiran saya sendiri, maka gelar dan ijasah yang telah diterbitkan oleh universitas batal saya terima.

Denpasar, 20 Juli 2011

Yang menyatakan

Ni Wayan Sumartini Saraswati

ABSTRAK

TEXT MINING DENGAN METODE *NAÏVE BAYES CLASSIFIER* DAN *SUPPORT VECTOR MACHINES* UNTUK *SENTIMENT ANALYSIS*

Text mining, mengacu pada proses mengambil informasi berkualitas tinggi dari teks. Informasi berkualitas tinggi biasanya diperoleh melalui peramalan pola dan kecenderungan melalui sarana seperti pembelajaran pola statistik. Proses *text mining* yang khas meliputi kategorisasi teks, *text clustering*, ekstraksi konsep/entitas, produksi taksonomi granular, *sentiment analysis*, penyimpulan dokumen, dan pemodelan relasi entitas.

Pada penelitian ini dibahas klasifikasi opini sebagai opini positif dan opini negatif pada data berbahasa Inggris dan data berbahasa Indonesia menggunakan metode *Naïve Bayes Classifier* (NBC) dan *Support Vector Machine* (SVM). Baik metode NBC maupun metode SVM memberikan unjuk kerja yang baik dalam *sentiment analysis* pengklasifikasian opini berbahasa Inggris dan berbahasa Indonesia pada penelitian ini. Hasil percobaan menunjukkan bahwa metode SVM memberikan unjuk kerja yang lebih baik daripada metode NBC untuk mengklasifikasikan opini berbahasa Inggris dan opini positif berbahasa Indonesia. Sedangkan NBC memberikan unjuk kerja yang lebih baik dalam mengklasifikasikan data uji opini negatif berbahasa Indonesia.

Kata kunci : *text mining*, *sentiment analysis*, *opinion mining*, NBC, SVM

ABSTRACT

TEXT MINING WITH NAÏVE BAYES CLASSIFIER DAN SUPPORT VECTOR MACHINES METHOD FOR SENTIMENT ANALYSIS

Text mining refers to the process of deriving high-quality information from text. High-quality information is typically derived through the divising of patterns and trends through means such as statistical pattern learning. Typical text mining tasks include text categorization, text clustering, concept/entity extraction, production of granular taxonomies, sentiment analysis, document summarization, and entity relation modeling.

This research discussed the opinions classification as positive opinions and negative opinions on English and Indonesian language data using the Naive Bayes Classifier (NBC) *and Support Vector Machine* (SVM). *In this study, both* NBC *and* SVM *method gives a good sentiment analysis performance in classifying opinion for English and Indonesia language. The experimental results showed that* SVM *method gives better performance than* NBC *method for classifying English opinions. Whether NBC method gives better performance for classifying negative opinions experimental data on Indonesian language.*

Keywords : *text mining, sentiment analysis, opinion mining,* NBC, SVM

KATA PENGANTAR

Segenap puja dan puji syukur penulis panjatkan kehadapan Ida Sang Hyang Widhi Wasa sebagai sumber dari segala sumber pengetahuan, karena atas asung kertha wara nugrahaNya tesis yang berjudul ***"TEXT MINING* DENGAN METODE *NAÏVE BAYES CLASSIFIER* DAN *SUPPORT VECTOR MACHINES* UNTUK *SENTIMENT ANALYSIS"*** ini dapat diselesaikan.

Dalam penyusunan tesis ini, penulis banyak memperoleh petunjuk dan bimbingan dari berbagai pihak. Sehubungan dengan hal tersebut, maka pada kesempatan ini penulis menyampaikan ucapan terima kasih dan penghargaan yang sebesar-besarnya kepada :

1. Ibu Prof. Dr. dr. A.A. Raka Sudewi, Sp.S(K), sebagai Direktur Program Pascasarjana Universitas Udayana.
2. Ibu Prof. Ir. Ida Ayu Dwi Giriantari, M.Eng.Sc.,Ph.D, sebagai Ketua Program Magister Program Studi Teknik Elektro Program Pascasarjana Universitas Udayana.
3. Bapak DR. I. Ketut Gede Darma Putra, S.Kom., MT, selaku dosen pembimbing I yang telah banyak membantu dalam memberikan ide, saran, motivasi, bimbingan selama perkuliahan dan pengerjaan tesis ini.
4. Ibu Ni Made Ary Esta Dewi Wirastuti, ST., M.Sc., Ph.D, selaku dosen pembimbing II yang telah banyak membantu dalam memberikan ide, saran, motivasi, bimbingan selama pengerjaan tesis ini.
5. Ibu Prof. Ir. Rukmi Sari Hartati, MT., PhD, Bapak Wayan Gede Ariastina, ST., MEngSc., PhD, dan Ibu Ir. Linawati, MengSc., PhD, selaku dosen penguji yang telah memberikan masukan demi kesempurnaan tesis ini.
6. PT. Bali Post atas ketersediaan data yang digunakan dalam penelitian ini.
7. Suami dan anak anak tercinta yang telah dengan sabar memberikan dukungan dan kesempatan dalam menyelesaikan tesis ini
8. Ibu, Bapak dan segenap keluarga besar yang telah memberikan dukungan moril maupun material hingga terselesaikannya tesis ini.

9. Bapak Gde Iwan Setiawan SE, MKom, selaku Ketua STMIK Denpasar atas ijin belajar dan beberapa kemudahan selama penulis menempuh pendidikan pada program magister ini.
10. Teman – teman seperjuangan di program pasacasarjana Elektro Udayana angkatan 2009.

Seperti kata pepatah tak ada gading yang tak retak, maka tesis ini tentu saja masih memiliki kekurangan. Harapan penulis, semoga karya ini bermanfaat bagi penelitian – penelitian selanjutnya.

Juli, 2011

Penulis

DAFTAR ISI

Halaman

DAFTAR TABEL

Halaman

DAFTAR GAMBAR

Halaman

BAB I

PENDAHULUAN

1.1 Latar Belakang

Web adalah tempat yang baik bagi orang - orang untuk mengekspresikan pendapat mereka, pada berbagai topik. Bahkan pemberi opini secara profesional, seperti reviewer film, memiliki *blog* dimana publik dapat mengomentari dan merespon apa yang mereka pikirkan. Kemampuan untuk mengekstrak pendapat tersebut dari baris-baris teks dapat menjadi sangat berguna, dan ini adalah area studi yang banyak dikaji, tidak diragukan karena kemungkinan nilai komersialnya (Ian Barber, 2010). Dikarenakan kebanyakan informasi (perkiraan umum mengatakan lebih dari 80%) saat ini disimpan sebagai teks, *text mining* diyakini memiliki potensi nilai komersial tinggi (Clara Bridge, 2011).

Text mining, mengacu pada proses mengambil informasi berkualitas tinggi dari teks. Informasi berkualitas tinggi biasanya diperoleh melalui peramalan pola dan kecenderungan melalui sarana seperti pembelajaran pola statistik. 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 bernama) (Wikipedia, 2011).

Salah satu metode klasifikasi yang dapat digunakan adalah metode Naïve Bayes yang sering disebut dengan *Naïve Bayes Classifier* (NBC). Kelebihan NBC adalah sederhana tetapi memiliki akurasi yang tinggi. Berdasarkan hasil

eksperimen, NBC terbukti dapat digunakan secara efektif untuk mengklasifikasikan berita secara otomatis dengan akurasi mencapai 90.23%. Algoritma NBC yang sederhana dan kecepatannya yang tinggi dalam proses pelatihan dan klasifikasi membuat algoritma ini menarik untuk digunakan sebagai salah satu metode klasifikasi (Yudi Wibisono, 2008).

Teknik klasifikasi yang telah memperoleh perhatian serius adalah *support vector machine* (SVM). Teknik ini berakar pada teori pembelajaran statistik dan telah menunjukkan hasil empiris yang menjanjikan dalam berbagai aplikasi praktis dari pengenalan digit tulisan tangan sampai kategorisasi teks. SVM juga bekerja sangat baik pada data dengan banyak dimensi dan menghindari kesulitan dari permasalahan dimensionalitas (Pang-Ning Tan, dkk, 2006).

Kedua metode tersebut banyak digunakan dalam kategorisasi teks. Pada hasil eksperimen (Wulandini, F. & Nugroho, A. N. 2009) untuk kategorisasi teks berbahasa Indonesia didapatkan bahwa SVM menunjukkan performansi yang sedikit lebih baik dengan akurasi 92,5% dibandingkan metode NBC dengan akurasi 90% padahal metode NBC adalah metode yang jauh lebih konvensional dan lebih sederhana. Sehingga pada penelitian ini ingin diketahui metode yang mana memiliki performansi yang lebih baik untuk diimplementasikan dalam *sentiment analysis* opini berbahasa Inggris dan berbahasa Indonesia.

Beberapa metode lain yang digunakan pula untuk proses teks mining adalah C45, K-Nearest Neighbor, K-Means dan algoritma genetika. Dari hasil penelitian (Wulandini, F. & Nugroho, A. N. 2009) untuk kategorisasi teks berbahasa Indonesia didapatkan hasil akurasi 29,17% untuk *K-Nearest Neighbor*

dan 77,5% untuk metode C45. Oleh karena itu pada penelitian ini dipilih metode SVM dan NBC untuk menyelesaikan masalah *sentiment analysis*.

Sentiment Analysis atau opinion mining adalah studi komputasional dari opini-opini orang, *appraisal* dan emosi melalui entitas, *event* dan atribut yang dimiliki (Biu, L. 2010). 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 bersifat positif , negatif atau netral (Dehaff, M., 2010).

1.2 Rumusan Masalah

Ada beberapa pokok masalah yang akan dikaji dalam penelitian ini, antara lain :

1. Bagaimana unjuk kerja metode *Naïve Bayes Classifier* dalam mengklasifikasikan opini berbahasa Inggris dan berbahasa Indonesia.
2. Bagaimana unjuk kerja metode *Support Vector Machine* dalam mengklasifikasikan opini berbahasa Inggris dan berbahasa Indonesia.
3. Diantara dua metode NBC dan SVM, manakah yang memberikan unjuk kerja lebih baik untuk mengklasifikasikan opini berbahasa Inggris dan opini berbahasa Indonesia.
4. Dapatkah metode NBC dan SVM mengklasifikasikan sebuah paragraf opini yang terdiri dari beberapa kalimat dalam bahasa Inggris dan bahasa Indonesia.

1.3 Tujuan Penelitian

Penelitian ini memiliki tujuan untuk mengembangkan software yang dapat mengklasifikasikan opini berbahasa Inggris dan opini berbahasa Indonesia sebagai opini positif ataupun opini negatif menggunakan metode *naive bayes classification* dan metode *support vector machine* serta menganalisis performansi kedua metode tersebut untuk mengklasifikasikan opini berbahasa Indonesia dan opini berbahasa Inggris.

1.4 Manfaat Penelitian

Dengan adanya software yang mampu mengklasifikasikan opini maka proses klasifikasi opini dalam jumlah yang besar dapat dilakukan secara terkomputerisasi sebagai ganti dari proses klasifikasi manual. Hal ini terutama akan dirasakan manfaatnya untuk analisis layanan dan produk dalam kaitannya dengan respon pasar. Kita dapat melacak produk-produk, merek dan orang-orang misalnya dan menentukan apakah mereka dilihat positif atau negatif di web.

Dengan didapatkannya hasil pengukuran performansi metode *naive bayes classification* dan metode *support vector machine* dalam mengklasifikasikan teks opini maka akan diketahui metode yang lebih unggul sehingga dapat dijadikan metode acuan dalam *sentiment analysis*.

1.5 Ruang Lingkup Penelitian

Penelitian dilakukan untuk teks opini film dalam bahasa Inggris. Data terbagi atas opini positif dan opini negatif. Sebagian data akan dijadikan data latih dan sebagian sebagai data uji untuk mengukur performansi metode *naive bayes classification* dan *support vector machine*.

Data opini berbahasa Indonesia diambil dari harian Bali Post yang terbit di provinsi Bali pada rubrik Bali Terkini dari Januari 2010 sampai Februari 2011.

Sentiment Analysis dilakukan dalam lingkup fungsi dasar yaitu mengklasifikasikan kalimat opini berbahasa Inggris dan berbahasa Indonesia pada data tersebut di atas sebagai opini positif atau opini negatif berdasarkan isinya dengan dua metode yaitu *Naïve Bayes Classifier* dan *Support Vector Machine*.

1.6 Keaslian Penelitian

Penelitian yang berkaitan dengan *sentiment analysis* telah dilakukan dengan metode *Naïve Bayes Classifier* oleh Ian Barber dalam artikelnya *Bayesian Opinion Mining* tahun 2010. Sedangkan (Wulandini, F. & Nugroho, A. N., 2009) dalam penelitiannya membandingkan beberapa metode teks *mining* yaitu C45, *K-Nearest Neighbor*, *Naïve Bayes Classifier* dan *Support Vector Machine* dalam menyelesaikan permasalahan kategorisasi teks berbahasa Indonesia. Penelitian yang lain menggunakan metode NBC dan telah menunjukkan hasil yang baik dilakukan oleh (Wibisono, Y., 2005) untuk mengklasifikasikan berita yang termuat di harian kompas.

Penelitian lain tentang adaptasi domain pada *sentiment analysis* dilakukan oleh (Blitzer, J., Dredze, M. & Pereira, F., 2006). Penelitian ini menggunakan algoritma *structural correspondence learning* (SCL).

Dari beberapa penelitian di atas belum ditemukan penelitian yang mengungkap bagaimana *sentiment analysis* diselesaikan dengan metode SVM. Untuk perbandingan performansi maka digunakan metode NBC dengan beberapa variasi jumlah data latih dan data uji. Beberapa penelitian teks mining dalam

bidang kategorisasi teks untuk data berbahasa Indonesia telah dibahas namun untuk teks mining dalam bidang *sentiment analysis* data berbahasa Indonesia belum dilakukan.

BAB II

KAJIAN PUSTAKA

2.1 *State Of Art Review*

Ian Barber dalam *Bayesian Opinion Mining*, 2010 telah melakukan eksperimen untuk data review film dan menghasilkan tingkat akurasi 80% menggunakan metode NBC. Tingkat akurasi diujikan untuk ketepatan menentukan klas opini dengan 5000 record opini negatif dan 5000 record opini positif sebagai data latih. Data uji adalah 333 opini negatif. Ian Barber belum melakukan eksperimen untuk data uji *sentiment* positif dan pengaruh variasi jumlah data latih dan data uji terhadap performansi metode NBC. Ian Barber juga belum melakukan eksperimen menggunakan data tersebut untuk metode klasifikasi teks yang lain misalkan SVM.

Yudi Wibisono dalam Klasifikasi Berita Berbahasa Indonesia menggunakan *Naïve Bayes Classifier*, 2006 telah melakukan klasifikasi pada 582 dokumen berbahasa Indonesia menggunakan metode NBC dan memperoleh hasil eksperimen seperti pada tabel 2.1 :

Tabel 2.1
Hasil Eksperimen Yudi Wibisono pada klasifikasi dokumen berbahasa Indonesia

Jumlah Dokumen Contoh	Jumlah Dokumen Uji Coba	Akurasi (%)
524 (90%)	58 (10%)	89,47
407 (70%)	175 (30%)	90,23
291 (50%)	291 (50%)	86,90
175 (30%)	407 (70%)	85,47
58 (10%)	524 (90%)	68,64

Dari tabel 2.1 terlihat bahwa nilai akurasi NBC tinggi, terutama jika dokumen contoh yang digunakan besar (≥400 dokumen). Hal yang menarik adalah akurasi tidak menunjukkan peningkatan yang signifikan walaupun dokumen contoh telah meningkat banyak dari 70% menjadi 90% serta akurasi masih relative tinggi walaupun dokumen contoh secara ekstrim dikurangi hanya 58 dokumen (10%).

Fatimah Wulandini dan Anto Satriyo Nugroho dalam (*Text Classification Using Support Vector Machine for Webmining Based Spation Temporal Analysis of the Spread of Tropical Diseases*, 2009) mendapatkan hasil bahwa metode SVM menunjukkan hasil paling baik pada kategorisasi teks berbahasa Indonesia. Eksperimen dilakukan pada 3713 *features* dan 360 *instances*. Data tersebut dibagi menjadi 120 *instances* sebagai data uji dan 240 *instances* sebagai data latih. Hasilnya serti ditunjukkan oleh tabel 2.2 :

Tabel 2.2
Hasil Eksperimen Fatimah Wulandini dan Anto Satriyo Nugroho

Metode	Akurasi
SVM	92,5 %
K- Nearest Neighbor	29,17 %
Naïve Bayes Classifier	90 %
C45	77,5 %

Tabel 2.2 tersebut menunjukkan performansi yang tidak berbeda jauh antara metode SVM dan NBC walaupun metode NBC adalah metode yang lebih konvensional dan lebih sederhana.

Fabrice Colas & Pavel Brazdil dalam penelitiannya berjudul *Comparison of SVM and Some Older Classification Algorithms in Text Classification Tasks*

mendapatkan bahwa metode NBC memiliki performansi yang lebih baik dibandingkan KNN dan SVM untuk menyelesaikan *binary classification* pada dokumen berbahasa Inggris. Hasil penelitiannya juga menyebutkan waktu komputasi yang jauh lebih pendek oleh metode NBC dan KNN. Waktu komputasi SVM berkembang secara kuadratik seiring dengan perkembangan jumlah data latih.

Namun penelitian oleh Jason D. M. Rennie & Ryan Rifkin yang berjudul *Improving Multiclass Text Classification with the Support Vector Machine* menunjukkan hasil bahwa SVM menghasilkan performansi yang lebih baik dalam menyelesaikan klasifikasi teks multi kelas dibandingkan metode NBC. Hal ini sesuai dengan pernyataan Fabrice Colas & Pavel Brazdil bahwa SVM unggul dalam klasifikasi *multiclass*.

Penelitian mengenai *sentiment analysis* yang dilakukan oleh Ahmed Abbasi, Hsinchun Chen, & Arab Salem berjudul *Sentiment Analysis in Multiple Languages: Feature Selection for Opinion Classification in Web Forums,* menemukan bahwa metode hibridisasi algoritma genetika EWGA mendapatkan hasil yang lebih baik dibandingkan metode SVM weight untuk *feature selection*. Sedangkan untuk proses klasifikasi sendiri dilakukan dengan metode SVM. Metode EWGA merupakan gabungan antara metode heuristic *Information Gain* (IG) dengan metode random Algoritma Genetika.

Untuk adaptasi pergantian domain dalam proses *sentiment analysis* , Blitzer, J., Dredze, M. & Pereira dalam penelitiannya berjudul *Biographies, Bollywood, Boom-boxes and Blenders:Domain Adaptation for Sentiment Classification* digunakan metode structural correspondence learning (SCL),

baseline dan SCLMI. Hasil ekperimen menunjukkan SCLMI mendapatkan hasil terbaik diantara metode yang dipakai.

Dari beberapa penelitian sebelumnya mengenai *text mining* maka dapat dibuatkan rangkuman seperti ditunjukkan oleh tabel 2.3

Tabel 2.3
Rangkuman penelitian *text mining* sebelumnya

No	Judul, penulis	Kategori	Metode	Deskripsi / hasil / kesimpulan
1.	*Bayesian Opinion Mining*, Ian Barber	*Sentiment analysis*	NBC	Dilakukan pada data review film berbahasa Inggris dan diujikan untuk 5000 record opini negatif dan 5000 record opini positif sebagai data latih dan 333 record opini negatif sebagai data uji serta menghasilkan akurasi sebesar 80%
2.	Klasifikasi Berita Berbahasa Indonesia menggunakan *Naïve Bayes Classifier,* yudi Wibisono	Kategorisasi teks	NBC	Dilakukan pada 582 dokumen berbahasa Indonesia dan memperoleh hasil akurasi tertinggi 90.23% untuk persentase data latih dan data uji sebesar 70% dan 30%
3.	*Text Classification Using Support Vector Machine for Webmining Based Spation Temporal Analysis of the Spread of Tropical Diseases,* Fatimah Wulandini dan Anto Satriyo Nugroho	Kategorisasi teks	SVM, NBC, K-Nearest Neighbor, C45	Dilakukan pada 3713 feature dan 360 instance. 360 instance sebagai data latih dan 120 instance sebagai data uji. SVM dan NBC menunjukkan hasil yang jauh lebih baik 92,5% dan 90%.
4.	*Comparison of SVM and Some*	Binary kategorisasi	SVM, KNN, NBC	NBC menunjukkan performansi yang paling

	Older Classification Algorithms in Text Classification Tasks, Fabrice Colas & Pavel Brazdil	teks		baik melebihi SVM dan KNN. Waktu komputasi NBC dan KNN jauh lebih pendek daripada SVM.
5.	*Improving Multiclass Text Classification with the Support Vector Machine,* Jason D. M. Rennie & Ryan Rifkin	Kategorisasi teks	SVM dan NBC	SVM menghasilkan performansi yang lebih baik dibandingkan NBC
6.	*Sentiment Analysis in Multiple Languages: Feature Selection for Opinion Classification in Web Forums,* Ahmed Abbasi, Hsinchun Chen, & Arab Salem	*Sentiment Analysis*	Entropy Weighted Genetic Algorithm, SVM Weight	EWGA menunjukkan performansi yang lebih baik dari SVM Weight
7.	*Biographies, Bollywood, Boom-boxes and Blenders:Domain Adaptation for Sentiment Classification,* Blitzer, J., Dredze, M. & Pereira	*Sentiment Analysis*	structural corres pondence learning (SCL), baseline, SCLMI	SCL-MI menunjukkan performansi yang lebih baik untuk adaptasi domain.

2.2 *Text Mining*

Text mining, mengacu pada proses mengambil informasi berkualitas tinggi dari teks. Informasi berkualitas tinggi biasanya diperoleh melalui peramalan pola

dan kecenderungan melalui sarana seperti pembelajaran pola statistik. *Text mining* biasanya melibatkan proses penataan teks input (biasanya parsing, bersama dengan penambahan beberapa fitur linguistik turunan dan penghilangan beberapa diantaranya, dan penyisipan *subsequent* ke dalam database), menentukan pola dalam data terstruktur, dan akhirnya mengevaluasi dan menginterpretasi output. 'Berkualitas tinggi' di bidang *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 bernama) (Wikipedia, 2011).

Klasifikasi / kategorisasi dokumen adalah masalah dalam ilmu informasi. Tugas kita adalah untuk menetapkan dokumen elektronik masuk dalam satu atau lebih kategori, berdasarkan isinya. Tugas klasifikasi dokumen dapat dibagi menjadi dua macam yaitu klasifikasi dokumen terawasi di mana beberapa mekanisme eksternal (seperti feedback manusia) memberikan informasi mengenai klasifikasi yang tepat untuk dokumen, dan klasifikasi dokumen tak terawasi, dimana klasifikasi harus dilakukan sepenuhnya tanpa merujuk ke informasi eksternal. Ada juga klasifikasi dokumen semi-diawasi, dimana bagian dari dokumen diberi label oleh mekanisme eksternal (Wikipedia, 2011).

Pendekatan manual *text mining* secara intensif dalam laboratorium pertama muncul pada pertengahan 1980-an, namun kemajuan teknologi telah memungkinkan ranah tersebut untuk berkembang selama dekade terakhir. *Text mining* adalah bidang interdisipliner yang mengacu pada pencarian informasi,

pertambangan data, pembelajaran mesin, statistik, dan komputasi linguistik. Dikarenakan kebanyakan informasi (perkiraan umum mengatakan lebih dari 80%) saat ini disimpan sebagai teks, *text mining* diyakini memiliki potensi nilai komersial tinggi (Clara Bridge, 2011).

Saat ini , *text mining* telah mendapat perhatian dalam berbagai bidang :

1. Aplikasi keamanan.

Banyak paket perangkat lunak *text mining* dipasarkan terhadap aplikasi keamanan, khususnya analisis *plain text* seperti berita internet. Hal ini juga mencakup studi enkripsi teks.

2. Aplikasi biomedis.

Berbagai aplikasi *text mining* dalam literatur biomedis telah disusun. Salah satu contohnya adalah PubGene yang mengkombinasikan *text mining* biomedis dengan visualisasi jaringan sebagai sebuah layanan Internet. Contoh lain *text mining* adalah GoPubMed.org. Kesamaan semantik juga telah digunakan oleh sistem *text mining*, yaitu, GOAnnotator.

3. Perangkat Lunak dan Aplikasi

Departemen riset dan pengembangan perusahaan besar, termasuk IBM dan Microsoft, sedang meneliti teknik *text mining* dan mengembangkan program untuk lebih mengotomatisasi proses pertambangan dan analisis. Perangkat lunak *text mining* juga sedang diteliti oleh perusahaan yang berbeda yang bekerja di bidang pencarian dan pengindeksan secara umum sebagai cara untuk meningkatkan performansinya.

4. Aplikasi Media Online

Text mining sedang digunakan oleh perusahaan media besar, seperti perusahaan Tribune, untuk menghilangkan ambigu informasi dan untuk memberikan pembaca dengan pengalaman pencarian yang lebih baik, yang meningkatkan loyalitas pada site dan pendapatan. Selain itu, editor diuntungkan dengan mampu berbagi, mengasosiasi dan properti paket berita, secara signifikan meningkatkan peluang untuk menguangkan konten.

5. Aplikasi Pemasaran

Text mining juga mulai digunakan dalam pemasaran, lebih spesifik dalam analisis manajemen hubungan pelanggan. Coussement dan Van den Poel (2008) menerapkannya untuk meningkatkan model analisis prediksi untuk *churn* pelanggan (pengurangan pelanggan).

6. *Sentiment Analysis*

Sentiment Analysis mungkin melibatkan analisis dari review film untuk memperkirakan berapa baik review untuk sebuah film. Analisis semacam ini mungkin memerlukan kumpulan data berlabel atau label dari efektifitas kata-kata. Sebuah sumber daya untuk efektivitas kata-kata telah dibuat untuk WordNet.

7. Aplikasi Akademik

Masalah *text mining* penting bagi penerbit yang memiliki database besar untuk mendapatkan informasi yang memerlukan pengindeksan untuk pencarian. Hal ini terutama berlaku dalam ilmu sains, di mana informasi yang sangat spesifik sering terkandung dalam teks tertulis. Oleh karena itu, inisiatif telah diambil seperti *Nature's proposal* untuk *Open Text Mining Interface* (OTMI) dan *Health's common Journal Publishing* untuk *Document Type Definition* (DTD) yang akan

memberikan isyarat semantik pada mesin untuk menjawab pertanyaan spesifik yang terkandung dalam teks tanpa menghilangkan *barrier* penerbit untuk akses publik.

Sebelumnya, website paling sering menggunakan pencarian berbasis teks, yang hanya menemukan dokumen yang berisi kata-kata atau frase spesifik yang ditentukan oleh pengguna. Sekarang, melalui penggunaan web semantik, *text mining* dapat menemukan konten berdasarkan makna dan konteks (daripada hanya dengan kata tertentu).

Text mining juga digunakan dalam beberapa filter email spam sebagai cara untuk menentukan karakteristik pesan yang mungkin berupa iklan atau materi yang tidak diinginkan lainnya.

2.3 *Sentiment Analysis*

Sentiment analysis atau *opinion mining* mengacu pada bidang yang luas dari pengolahan bahasa alami, komputasi linguistik dan *text mining*. Secara umum, bertujuan untuk menentukan *attitude* pembicara atau penulis berkenaan dengan topik tertentu. *Attitude* mungkin penilaian atau evaluasi mereka, pernyataan afektif mereka (pernyataan emosional penulis saat menulis) atau komunikasi emosional dimaksud (efek emosional penulis inginkan terhadap pembaca) (Wikipedia, 2011).

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

bersifat positif , negatif atau netral (Dehaff, M., 2010). Lebih lanjut *sentiment analysis* dapat menyatakan emosional sedih, gembira, atau marah.

Beberapa penelitian mengklasifikasikan polaritas dokumen pada skala multi-arah, yang dicoba oleh (Pang, B. & Lee, L. 2005) dan (Snyder B. & Barzilay R. 2007) antara lain : memperluas tugas dasar klasifikasi review film sebagai positif atau negatif terhadap memprediksi peringkat bintang baik skala 3 atau bintang 4, sementara (Snyder B. & Barzilay R. 2007) melakukan analisa mendalam tentang review restoran, memprediksi peringkat untuk berbagai aspek dari restoran yang diberikan, seperti makanan dan suasana (dalam skala bintang lima).

Sebuah metode yang berbeda untuk menentukan sentimen adalah penggunaan sistem skala dimana kata-kata umumnya terkait memiliki sentimen negatif, netral atau positif dengan mereka diberi nomor pada skala -5 sampai +5 (paling negatif hingga yang paling positif) dan ketika sepotong teks terstruktur dianalisis dengan pemrosesan bahasa alami, konsep selanjutnya dianalisis untuk memahami kata-kata ini dan bagaimana mereka berhubungan dengan konsep. Setiap konsep kemudian diberi skor berdasarkan bagaimana kata-kata sentimen berhubungan dengan konsep, dan skor yang terkait. Hal ini memungkinkan gerakan untuk pemahaman yang lebih canggih dari sentimen berdasarkan skala 11 titik.

Penelitian dengan arah berbeda adalah identifikasi subjektivitas / objektivitas. Tugas ini biasanya didefinisikan sebagai menggolongkan suatu teks yang diberikan (biasanya kalimat) ke salah satu dari dua kelas: objektif atau subjektif (Pang, B. & Lee, L, 2008). Masalah ini kadang-kadang dapat lebih sulit

daripada klasifikasi polaritas (Mihalcea, R. & dkk, 2007) subjektivitas kata-kata dan frase mungkin tergantung pada konteks dan dokumen objektif mungkin berisi kalimat subjektif (misalnya, sebuah artikel berita mengutip pendapat orang). Selain itu, seperti yang disebutkan oleh (Su, F. & Markert, K. 2008), hasilnya sangat tergantung pada definisi subjektivitas digunakan ketika memberikan keterangan pada teks. Namun, (Pang, B. & Lee, L. 2004) menunjukkan bahwa menghapus kalimat objektif dari sebuah dokumen sebelum mengelompokkan polaritasnya membantu meningkatkan kinerja.

Kita dapat melacak produk-produk, merek dan orang-orang misalnya dan menentukan apakah mereka dilihat positif atau negatif di web. Hal ini memungkinkan bisnis untuk melacak:

a. Deteksi Flame (rants buruk)

b. Persepsi produk baru.

c. Persepsi Merek.

d. Manajemen reputasi.

Hal ini juga memungkinkan individu untuk mendapatkan sebuah pandangan tentang sesuatu (review) pada skala global (Jenkins, M. C., 2011).

Orang sering kali menyatakan lebih dari satu opini "*the movie was terrible, but DeNiro's performance was superb, as always*", sebuah sarkasme "*this is probably the best laptop Dell could come up with*", atau menggunakan negasi dan banyak elemen kompleks sehingga sulit untuk diparsing "*not that I'm saying this was a bad experience*".

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 batere 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 Barber, 2010).

Hal pertama dalam pemrosesan dokumen adalah memecah kumpulan karakter ke dalam kata atau token, sering disebut sebagai tokenisasi. Tokenisasi adalah hal yang kompleks untuk program komputer karena beberapa karakter dapat dapat ditemukan sebagai *token delimiters*. Delimiter adalah karakter spasi, tab dan baris baru "*newline"*, sedangkan karakter ( ) < > ! ? " kadangkala dijadikan delimiter namun kadang kala bukan tergantung pada lingkungannya (Wulandini, F. & Nugroho, A. N. 2009).

2.4 *Naïve Bayes Classifier*

Sebuah *bayes classifier* adalah *classifier* probabilistik sederhana berdasarkan penerapan teorema Bayes (dari statistik Bayesian) dengan asumsi independen (naif) yang kuat. Sebuah istilah yang lebih deskriptif untuk model probabilitas yang digaris bawahi adalah " model fitur independen".

Dalam terminologi sederhana, sebuah NBC mengasumsikan bahwa kehadiran (atau ketiadaan) fitur tertentu dari suatu kelas tidak berhubungan dengan kehadiran (atau ketiadaan) fitur lainnya. Sebagai contoh, buah mungkin dianggap apel jika merah, bulat, dan berdiameter sekitar 4 inchi. Bahkan jika fitur ini bergantung satu sama lain atau atas keberadaan fitur lain,. Sebuah NBC menganggap bahwa seluruh sifat-sifat berkontribusi mandiri untuk probabilitas bahwa buah ini adalah apel.

Tergantung pada situasi yang tepat dari model probabilitas, NBC dapat dilatih sangat efisien dalam *supervised learning*. Dalam aplikasi praktis, parameter estimasi untuk model NBC menggunakan metode *likelihood* maksimum, dengan kata lain, seseorang dapat bekerja dengan model Naïve Bayes tanpa mempercayai probabilitas Bayesian atau menggunakan metode Bayesian lainnya.

Dibalik desain naifnya dan asumsi yang tampaknya terlalu disederhanakan, NBC telah bekerja cukup baik dalam banyak situasi dunia nyata yang kompleks. Pada tahun 2004, analisis masalah klasifikasi Bayesian telah menunjukkan bahwa ada beberapa alasan teoritis untuk keberhasilan yang tampaknya tidak masuk akal dari NBC (Zhang, H., 2004). Selain itu, perbandingan yang komprehensif dengan metode klasifikasi lainnya pada tahun 2006 menunjukkan bahwa klasifikasi Bayes mengungguli pendekatan terbaru, seperti *boosted tree* atau *random forest* (Caruana, R. & Niculescu-Mizil, A, 2006).

Sebuah keuntungan dari NBC adalah bahwa ia memerlukan sejumlah kecil data pelatihan untuk mengestimasi parameter (rata-rata dan varian dari variabel) yang diperlukan untuk klasifikasi. Karena variabel diasumsikan independen,

hanya varian dari variabel-variabel untuk setiap kelas yang perlu ditentukan dan bukan keseluruhan *covariance matrix.*

2.4.1 Model Probabilistic *Naïve Bayes*

Model probabilitas untuk classifier adalah model kondisional

$$p(C|F_1,\ldots,F_n) \quad (2.1)$$

terhadap variabel kelas dependen C dengan sejumlah kecil hasil atau kelas, tergantung pada beberapa variabel fitur F1 sampai Fn. Masalahnya adalah bahwa jika jumlah fitur n besar atau bila fitur bisa mengambil sejumlah besar nilai, maka membuat sebuah model pada tabel probabilitas adalah tidak mungkin. Oleh karena itu kita mereformulasi model untuk membuatnya lebih fleksibel.

Menggunakan teorema Bayes , kita menulis

$$p(C|F_1,\ldots,F_n) = \frac{p(C)p(F1,\ldots,Fn|C)}{p(F1,\ldots,Fn)} \quad (2.2)$$

Dalam bahasa Inggris persamaan di atas dapat ditulis sebagai

$$\text{posterior} = \frac{prior\ X\ likelihood}{evidence} \quad (2.3)$$

Dalam prakteknya kita hanya tertarik pada pembilang dari persamaan tersebut, karena penyebut tidak tergantung pada C dan nilai-nilai fitur Fi diberikan, sehingga penyebut secara efektif konstan. Pembilang ini setara dengan model probabilitas gabungan $p(C|F_1,\ldots,F_n)$ yang dapat ditulis ulang sebagai berikut, menggunakan penggunaan berulang dari definisi probabilitas bersyarat:

$$p(C|F_1,\ldots,F_n)$$
$$= p(C)\, p(F_1,\ldots,F_n|C)$$

$= p(C)\ p(F_1|\mathrm{C})\ p(F_2,\ldots,F_n|C, F_1)$

$= p(C)\ p(F_1|\mathrm{C})\ p(F_2|C,F_1)\ p(F_3,\ldots,F_n|C, F_1, F_2)$

$= p(C)\ p(F_1|\mathrm{C})\ p(F_2|C,F_1)\ p(F_3|C,F_1,F_2)\ p(F_4,\ldots,F_n|C, F_1, F_2, F_3)$

$= p(C)\ p(F_1|\mathrm{C})\ p(F_2|C,F_1)\ p(F_3|C,F_1,F_2)\ldots\ p(F_n|C, F_1, F_2, F_3,\ldots, F_{n-1})$ (2.4)

Sekarang asumsi kemandirian bersyarat yang "naif" memegang peranan. Menganggap bahwa setiap fitur Fi adalah secara kondisi independen terhadap setiap fitur lainnya Fj untuk j ≠ i. Ini berarti bahwa

$$p(F_i|\mathrm{C}, F_j) = p(F_i|\mathrm{C}) \quad (2.5)$$

untuk i ≠ j , sehingga *joint model* dapat dinyatakan sebagai

$p(C|F_1,\ldots,F_n) = p(C)\ p(F_1|\mathrm{C})\ p(F_2|C)\ p(F_3|C)\ldots$

$$=p(C)\ \prod_{i=1}^{n} p(Fi|C) \quad (2.6)$$

Ini berarti bahwa di bawah asumsi independen di atas, distribusi bersyarat dari variabel kelas C dapat dinyatakan seperti ini :

$$p(C|F_1,\ldots,F_n) = \frac{1}{Z}\ p(C)\ \prod_{i=1}^{n} p(Fi|C) \quad (2.7)$$

dimana Z (bukti) adalah faktor skala tergantung hanya pada $F_1,\ldots,F_n$, yaitu, sebuah konstanta jika nilai dari variabel fitur diketahui.

Model dari bentuk ini jauh lebih mudah dikelola, karena mereka memecah menjadi *class prior* p(C) dan distribusi probabilitas independen $p(F_i|\mathrm{C})$. Jika ada k kelas dan jika model untuk masing-masing $p(F_i|\mathrm{C} = c)$ dapat dinyatakan dalam bentuk parameter, maka model naif Bayes yang sesuai memiliki (k - 1) + n r k parameter. Dalam prakteknya, sering k = 2 (klasifikasi biner) dan r = 1 (variabel

Bernoulli sebagai fitur) yang umum, sehingga jumlah parameter model Bayes naif adalah 2n + 1, dimana n adalah jumlah fitur biner yang digunakan untuk klasifikasi dan prediksi.

2.4.2 Estimasi Parameter

Semua model parameter (yaitu, prior kelas dan distribusi probabilitas fitur) dapat didekati dengan frekuensi relatif dari himpunan pelatihan. Ini merupakan perkiraan kemungkinan maksimum dari probabilitas. Sebuah *prior class* dapat dihitung dengan asumsi kelas *equiprobable* (yaitu, prior = 1 / (jumlah kelas)), atau dengan menghitung perkiraan probabilitas kelas dari himpunan pelatihan (yaitu, (*prior* untuk kelas tertentu) = (jumlah sampel di kelas) / (jumlah sampel)). Untuk memperkirakan parameter untuk distribusi fitur ini, seseorang harus mengasumsikan distribusi atau menghasilkan model nonparametrik untuk fitur-fitur dari training set. Jika seseorang berhadapan dengan data kontinu, asumsi khas adalah distribusi Gaussian, dengan parameter model dari mean dan varians. Mean, μ, dihitung dengan

$$\mu = \frac{1}{N} \sum_{i=1}^{N} xi \tag{2.8}$$

dimana N adalah jumlah sampel dan xi adalah nilai dari suatu contoh yang diberikan.

Varian, σ^2, dihitung dengan $\sigma^2 = \frac{1}{(N-1)} \sum_{i=1}^{N} (xi - \mu)^2$ (2.9)

Jika sebuah kelas tertentu dan nilai fitur tidak pernah terjadi bersama-sama dalam himpunan pelatihan maka estimasi probabilitas berbasis frekuensi akan menjadi nol. Hal ini bermasalah karena akan menghapus seluruh informasi dalam

probabilitas lain ketika mereka dikalikan. Oleh karena itu sering diinginkan untuk memasukkan koreksi sampel kecil dalam semua perkiraan probabilitas bahwa tidak ada probabilitas untuk menjadi persis nol.

2.4.3 Membangun sebuah *classifier* dari model probabilitas.

Diskusi sejauh ini telah menurunkan model fitur independen, yaitu, model probabilitas *naïve bayes*. NBC mengkombinasikan model ini dengan aturan keputusan. Sebuah aturan yang umum adalah untuk memilih hipotesis yang paling mungkin, ini dikenal sebagai posteriori maksimum atau aturan keputusan MAP. *Classifier* terkait adalah fungsi yang didefinisikan sebagai berikut:

$$\text{classify } (f_1,\ldots,f_n) = \underset{c}{argmax}\ p(C=c) \prod_{i=1}^{n} p(Fi = fi \mid C = c) \qquad (2.10)$$

2.4.4 *Naïve Bayes Classifier* untuk klasifikasi dokumen.

Berikut ini adalah sebuah contoh dari NBC untuk permasalahan klasifikasi dokumen. Masalah mengklasifikasikan dokumen adalah berdasarkan konten, misalnya spam dan non-spam e-mail. Bayangkan bahwa dokumen yang diambil dari beberapa kelas dokumen yang dapat dimodelkan sebagai set kata-kata dimana probabilitas (independen) bahwa kata ke-i dari suatu dokumen tertentu terjadi dalam dokumen dari kelas C dapat ditulis sebagai

$$p(w_i|C) \qquad (2.11)$$

Untuk perlakuan ini, kita menyederhanakan hal-hal lebih lanjut dengan mengasumsikan bahwa kata-kata secara acak terdistribusi dalam dokumen - yaitu, kata-kata tidak tergantung pada panjang dokumen, posisi dalam dokumen ,dengan hubungannya dengan kata lain, atau dokumen-konteks yang lain.

Maka probabilitas suatu dokumen D , kelas C, adalah

$$\prod_i p(wi \mid C)$$

$p(D|C)$ = (2.12)

Pertanyaan yang ingin dijawab adalah: "berapa probabilitas bahwa dokumen D adalah milik kelas C ?" Dengan kata lain, berapa $p(C|D)$?

Sekarang menurut definisi

$$p(D|C) = \frac{p(D \cap C)}{p(C)} \quad (2.13)$$

dan

$$p(\mathrm{C}|D) = \frac{p(D \cap C)}{p(D)} \quad (2.14)$$

Teorema Bayes memanipulasi ini ke dalam pernyataan dari probabilitas dalam bentuk *likelihood*.

$$p(\mathrm{C}|D) = \frac{p(C)}{p(D)}\, p(D|C) \quad (2.15)$$

Asumsikan untuk saat ini bahwa hanya ada dua kelas yang saling eksklusif, S dan ¬ S (misalnya spam dan bukan spam),

$$p(D|S) = \prod_i p(wi\,|S) \quad (2.16)$$

dan

$$p(D|\neg S) = \prod_i p(wi\,|\neg S) \quad (2.17)$$

Menggunakan hasil Bayesian di atas, kita bisa menulis

$$p(S|D) = \frac{p(S)}{p(D)} \prod_i p(wi\,|S) \quad (2.18)$$

$$p(\neg S|D) = \frac{p(\neg S)}{p(D)} \prod_i p(wi\,|\neg S) \tag{2.19}$$

Membagi satu dengan yang lain memberikan:

$$\frac{p(S|D)}{p(\neg S|D)} = \frac{p(S)\prod_i p(wi|S)}{p(\neg S)\prod_i p(wi|\neg S)} \tag{2.20}$$

Yang bisa difaktorisasi ulang sebagai :

$$\frac{p(S|D)}{p(\neg S|D)} = \frac{p(S)}{p(\neg S)} \prod_i \frac{p(wi|S)}{p(wi|\neg S)} \tag{2.21}$$

Dengan demikian, rasio probabilitas *p* (*S* | *D*) / *p* (¬ *S* | *D*) dapat dinyatakan dalam serangkaian rasio kemungkinan. *p* probabilitas aktual (*S* | *D*) dapat dengan mudah dihitung dari log (*p* (*S* | *D*) / *p* (¬ *S* | *D*)) berdasarkan pengamatan dimana *p*(*S* | *D*) + *p* (¬ *S* | *D*) = 1.

Mengambil logaritma dari semua rasio, kita memiliki:

$$\ln \frac{p(S|D)}{p(\neg S|D)} = \ln \frac{p(S)}{p(\neg S)} + \sum_i \ln \frac{p(wi|S)}{p(wi|\neg S)} \tag{2.22}$$

Akhirnya, dokumen dapat diklasifikasikan sebagai berikut. Dikategorikan spam jika *p*(*S*|*D*) > *p*(¬*S*|*D*) (yaitu $\ln \frac{p(S|D)}{p(\neg S|D)} > 0$), jika tidak memenuhi maka bukan spam.

2.4.5 *Naïve Bayes Classification* untuk klasifikasi dokumen multikelas

Pada NBC setiap record direpresentasikan dalam pasangan atribut < a_1, $a_{2,\ldots,}$ a_n > dimana a_1 adalah kata pertama, a_2 adalah kata kedua dan seterusnya. Sedangkan V adalah himpunan kategori dokumen.

Pada saat klasifikasi, pendekatan Bayes akan menghasilkan label kategori yang paling tinggi probabilitasnya (V_{MAP}) dengan masukan atribut < a_1, $a_{2,...,}$ a_n >

$$V_{MAP} = \arg_{vj \epsilon V} \max P(v_j \mid a_1, a_{2,...,} a_n) \quad (2.23)$$

Teorema Bayes menyatakan

$$P(B \mid A) = \frac{P(A|B)P(B)}{P(A)} \quad (2.24)$$

Menggunakan teorema Bayes ini, persamaan (2.23) ini dapat ditulis :

$$V_{MAP} = \arg_{vj \epsilon V} \max \frac{P(a_1,a_2,...a_n|v_j)P(v_j)}{P(a_1,a_2,...a_n)} \quad (2.25)$$

$P(\ a_1,\ a_{2,...,}\ a_n\ )$ nilainya konstan untuk semua v_j sehingga persamaan ini dapat ditulis sebagai berikut:

$$V_{MAP} = \arg_{vj \epsilon V} \max P(a_1, a_{2,...,} a_n| v_j)\ P(v_j) \quad (2.26)$$

Tingkat kesulitan menghitung $P(a_1, a_{2,...,} a_n| v_j)$ menjadi tinggi karena jumlah term $P(a_1, a_{2,...,} a_n| v_j)$ bisa jadi akan sangat besar. Ini disebabkan jumlah term tersebut sama dengan jumlah kombinasi posisi kata dikali dengan jumlah kategori.

Naïve Bayes Classifier menyederhanakan hal ini dengan mengasumsikan bahwa dalam setiap kategori, setiap kata independen satu sama lain. Dengan kata lain :

$$P(a_1, a_{2,...,} a_n| v_j) = \prod_i P(a_i| v_j) \quad (2.27)$$

Substitusi persamaan ini dengan persamaan 3.4 akan menghasilkan :

$$V_{MAP} = \arg_{vj \epsilon V} \max P(v_j) \prod_i P(a_i| v_j) \quad (2.28)$$

$P(v_j)$ dan probabilitas kata w_k untuk setiap kategori $P(w_k| v_j)$ dihitung pada saat pelatihan.

$$P(v_j) = \frac{|\ docs\ j\ |}{|\ Contoh\ |} \quad (2.29)$$

$$P(w_k| v_j) = \frac{n_k+1}{n+|\ kosakata|} \quad (2.30)$$

Di mana $|docs_j|$ adalah jumlah dokumen pada kategori j dan $|Contoh|$ adalah jumlah dokumen yang digunakan dalam pelatihan. Sedangkan n_k adalah jumlah kemunculan kata w_k pada kategori v_j dan $|kosakata|$ adalah jumlah kata yang unik (*distinct*) pada semua data latihan. Jumlah kata dalam tiap kelas dinyatakan sebagai n.

Ringkasan algoritma untuk *Naïve Bayes Classifier* adalah sebagai berikut :

A. Proses pelatihan. Input adalah dokumen-dokumen contoh yang telah diketahui kategorinya.

1. Kosakata ← himpunan semua kata yang unik dari dokumen-dokumen contoh
2. Untuk setiap kategori v_j lakukan :
   a. $Docs_j$ ← Himpunan dokumen-dokumen yang berada pada kategori v_j
   b. Hitung $P(v_j)$ dengan persamaan 2.29
   c. Untuk setiap kata w_k pada kosakata lakukan :
      1. Hitung $P(w_k |v_j)$ dengan persamaan 2.30

B. Proses klasifikasi. Input adalah dokumen yang belum diketahui kategorinya :

Hasilkan v_{MAP} sesuai dengan persamaan 2.28 dengan menggunakan $P(v_j)$ dan $P(w_k | v_j)$ yang telah diperoleh dari pelatihan.

2.5 *Support Vector Machine*

Support Vector Machines (SVMs) 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 (margin lunak) diusulkan oleh Corinna Cortes dan Vapnik Vladimir (Cortes, C. & Vapnik, V, 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 non-probabilistik linier biner. Karena sebuah 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.

Secara intuitif, model SVM merupakan representasi dari data sebagai titik dalam ruang, dipetakan sehingga kategori contoh terpisah dibagi oleh celah jelas yang selebar mungkin. Data baru kemudian dipetakan ke dalam ruang yang sama dan diperkirakan termasuk kategori berdasarkan sisi mana dari celah data tersebut berada.

Lebih formal, *Support Vector Machine* membangun *hyperplane* atau himpunan *hyperplane* dalam ruang dimensi tinggi atau tak terbatas, yang dapat

digunakan untuk klasifikasi, regresi atau tugas-tugas lainnya. Secara intuitif, suatu pemisahan yang baik dicapai oleh *hyperplane* yang memiliki jarak terbesar ke titik data training terdekat dari setiap kelas (margin fungsional disebut), karena pada umumnya semakin besar margin semakin rendah error generalisasi dari pemilah.

Ketika masalah asal mungkin dinyatakan dalam dimensi ruang terbatas, sering terjadi bahwa dalam ruang, himpunan tidak dipisahkan secara linear. Untuk alasan ini diusulkan bahwa ruang dimensi terbatas dipetakan ke dalam sebuah ruang dimensi yang jauh lebih tinggi yang mungkin membuat pemisahan lebih mudah dalam ruang itu. Skema SVM menggunakan pemetaan ke dalam ruang yang lebih besar sehingga *cross product* dapat dihitung dengan mudah dalam hal variabel dalam ruang asal membuat beban komputasi yang wajar. *Cross product* di ruang yang lebih besar didefinisikan dalam hal fungsi kernel K (x, y) yang dapat dipilih sesuai dengan masalah. Sekumpulan *hyperplane* dalam ruang besar yang didefinisikan sebagai himpunan titik-titik yang *cross product* dengan vektor dalam ruang yang konstan. Vektor mendefinisikan *hyperplanes* dapat dipilih untuk menjadi kombinasi linear dengan parameter αi dari gambar vektor fitur yang terjadi pada database. Dengan pilihan ini sebuah hyperplane di titik x di ruang fitur yang dipetakan ke hyperplane ini ditentukan oleh relasi:

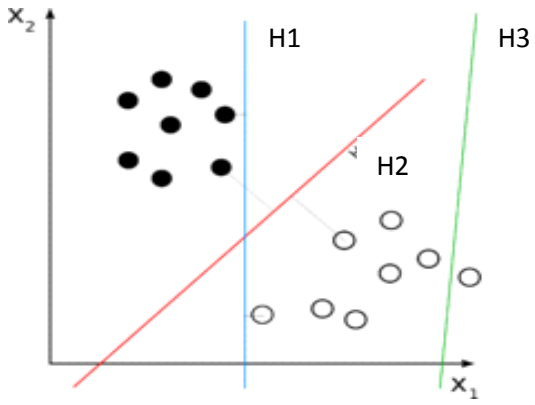
$$\sum_i \alpha i K(xi, x) = constant \qquad (2.31)$$

Perhatikan bahwa jika K (x, y) menjadi kecil ketika y tumbuh lebih lanjut dari x, setiap elemen dalam pengukuran penjumlahan dari tingkat kedekatan titik uji x ke titik xi pada database yang sesuai. Dengan cara ini jumlah kernel di atas dapat digunakan untuk mengukur kedekatan relatif masing-masing titik uji dengan titik data yang berasal dalam satu atau yang lain dari himpunan yang akan

dikelompokkan. Perhatikan fakta bahwa himpunan titik x dipetakan ke hyperplane yang manapun , dapat cukup rumit sebagai akibat mengijinkan pemisahan yang lebih kompleks antara himpunan yang jauh dari *convex* di ruang asli.

2.5.1 Motivasi

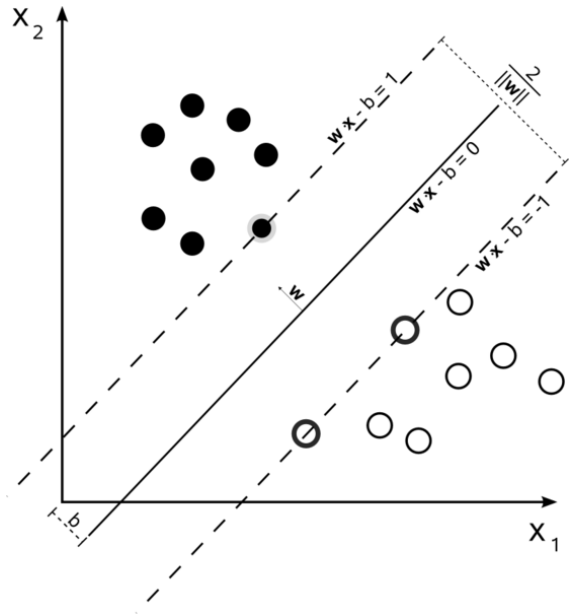
Ide utama dari metode SVM adalah konsep dari *hyperplane* margin maksimal. Dengan ditemukannya *hyperplane* margin maksimal maka vector tersebut akan membagi data menjadi bentuk klasifikasi yang paling optimum. Beberapa contoh hyperplane yang mungkin muncul untuk mengklasifikasi data ditunjukkan oleh gambar 2.1



Gambar 2.1 Contoh beberapa hyperlane

Dari gambar 2.1 didapat bahwa garis H3 (hijau) tidak memisahkan dua kelas. Garis H1 (biru) memisahkan, dengan margin kecil dan garis H2 (merah) dengan maksimum margin. Mengklasifikasi data adalah tugas umum dalam pembelajaran mesin. Misalkan beberapa titik data yang diberikan masing-masing milik salah satu dari dua kelas, dan tujuannya adalah untuk menentukan kelas suatu titik data baru akan masuk. Dalam kasus SVM, titik data dipandang sebagai vektor p-dimensi (a list dari p jumlah), dan kami ingin tahu apakah kita dapat

memisahkan titik-titik tersebut dengan (p - 1) *hyperplane* dimensional. Ini disebut linear classifier. Ada banyak *hyperplane* yang mungkin mengklasifikasikan data. Satu pilihan yang wajar sebagai *hyperplane* terbaik adalah salah satu yang mewakili pemisahan atau margin terbesar, antara dua kelas. Jadi kita memilih *hyperplane* sehingga jarak dari dan ke titik data terdekat di setiap sisi dimaksimalkan. Jika *hyperplane* tersebut ada, itu dikenal sebagai *hyperplane* maksimum margin dan *linier classifier* yang didefinisikannya dikenal sebagai pengklasifikasi margin maksimal. Ilustrasi dari *hyperplane* margin maksimal ditunjukkan oleh gambar 2.2 berikut.



Gambar 2.2 Hyperplane margin maksimal

Maksimum-margin *hyperplane* dan margin untuk suatu SVM dilatih dengan sampel dari dua kelas. Sampel pada margin disebut sebagai *support vector*.

2.5.2 *Formalization*

Kita diberikan beberapa data pelatihan $\mathcal{D}$, satu set dari n titik dalam bentuk

$$\mathcal{D} = \{(x_i, y_i) \mid x_i \in R^p , y_i \in \{ -1,1\}\}_{i=1}^{n} \quad (2.32)$$

dimana yi adalah 1 atau -1, menunjukkan kelas mana titik x_i itu berada. Masing-masing x_i adalah vektor nyata p-dimensi. Kami ingin mencari *hyperplane* maksimum margin yang membagi poin untuk poin yang memiliki $y_i = 1$ dari yang memiliki $y_i = -1$. *Hyperplane* apapun dapat ditulis sebagai himpunan titik-titik x memuaskan

$$w. x – b = 0, \quad (2.33)$$

dimana menunjukkan dot product. Vektor w adalah vektor normal: adalah tegak lurus *hyperplane* tersebut. Parameter $\frac{b}{\|w\|}$ menentukan offset *hyperplane* dari asal sepanjang vektor normal w.

Kami ingin memilih w dan b untuk memaksimalkan margin, atau jarak antara *hyperplane* paralel yang terpisah sejauh mungkin sementara masih memisahkan data. Hyperplanes ini dapat digambarkan oleh persamaan

$$w . x – b = 1 \quad (2.34)$$

dan

$$w . x – b = -1 \quad (2.35)$$

Perhatikan bahwa jika data pelatihan terpisah linier, kita bisa pilih dua *hyperplane* dari margin dengan sebuah cara : tidak ada poin antara mereka dan kemudian mencoba untuk memaksimalkan jaraknya. Dengan menggunakan geometri, kita menemukan jarak antara kedua *hyperplane* adalah $\frac{2}{\|w\|}$, jadi kita ingin

meminimalkan $\|w\|$. Seperti kita juga harus mencegah titik data jatuh ke dalam margin, kita menambahkan batasan berikut: untuk setiap i baik

$$w \cdot x_i - b \geq 1 \quad \text{untuk } x_i \text{ sebagai kelas pertama} \tag{2.36}$$

atau

$$w \cdot x_i - b \leq -1 \quad \text{untuk } x_i \text{ sebagai kelas kedua} \tag{2.37}$$

Hal ini dapat ditulis ulang sebagai :

$$y_i (w. x_i - b) \geq 1 \text{ , untuk semua } 1 \leq i \leq n \tag{2.38}$$

Kita bisa menempatkan ini bersama-sama untuk mendapatkan masalah optimasi:

Minimalkan (dalam w, b) $\|w\|$

subject dari (untuk setiap) i = 1,…, n

$$y_i (w. x_i - b) \geq 1 \tag{2.38}$$

2.5.3 *Primal form*

Masalah optimasi yang disajikan pada bagian sebelumnya sulit untuk diselesaikan karena bergantung pada $\|w\|$, bentuk normal w, yang melibatkan akar kuadrat. Untungnya adalah mungkin untuk mengubah persamaan dengan menggantikan $\|w\|$ dengan $\frac{1}{2}\|w\|^2$ (faktor dari 1 / 2 digunakan untuk kenyamanan matematis) tanpa mengubah solusi (nilai minimum yang asli dan persamaan dimodifikasi memiliki w dan b yang sama) . Ini adalah masalah optimasi pemrograman kuadratik (QP) Lebih jelasnya:

Minimalkan (dalam w, b) $\frac{1}{2}\|w\|^2$

dikenakan (untuk setiap) i = 1, … , n

y_i (w. x_i – b) ≥ 1 (2.38)

Seseorang bisa mencoba untuk mengungkapkan masalah sebelumnya dengan menggunakan pengali Lagrange non-negatif α_i sebagai

$$\min_{w,b,\alpha} \{\frac{1}{2}\|w\|^2 - \sum_{i=1}^{n} \alpha_i [y_i (w \cdot x_i - b) - 1] \} \quad (2.39)$$

tapi ini akan salah. Alasannya adalah sebagai berikut: misalkan kita dapat menemukan keluarga *hyperplane* yang membagi poin. Maka semua

$y_i(w \cdot x_i - b) - 1 \geq 0.$ (2.40)

Oleh karena itu kita bisa menemukan nilai minimal dengan mengirimkan semua α_i untuk +∞, dan minimum ini akan dicapai untuk semua anggota keluarga, tidak hanya untuk yang terbaik yang dapat dipilih sebagai pemecahan masalah asal.

Namun constrain sebelumnya dapat dinyatakan sebagai

$$\min_{w,b} \max_{\alpha} \{\frac{1}{2}\|w\|^2 - \sum_{i=1}^{n} \alpha_i [y_i (w \cdot x_i - b) - 1] \} \quad (2.41)$$

yang kita cari adalah titik pelana. Dalam melakukan hal tersebut semua poin yang dapat dipisahkan sebagai $y_i(w \cdot x_i - b) - 1 > 0.$ (2.42)

tidak penting karena kita harus mengatur αi yang bersesuain, dengan nol.

Masalah ini sekarang dapat diatasi dengan teknik dan pemrograman kuadratik standar. Solusi ini dapat dinyatakan dengan istilah kombinasi linier dari vektor pelatihan sebagai

$$w = \sum_{i=1}^{n} \alpha_i \, y_i \, x_i \quad (2.43)$$

Hanya beberapa α_i akan lebih besar dari nol. x_i yang sesuai adalah vektor pendukung, yang terletak di margin dan memenuhi $y_i(w \cdot x_i - b) = 1.$

Dari hal tersebut didapat bahwa vektor pendukung juga memenuhi

$$w \cdot x_i - b = 1/\ y_i = y_i \Leftrightarrow b = w \cdot x_i - y_i \quad (2.44)$$

yang memungkinkan seseorang untuk menentukan offset b. Secara praktis, itu lebih kuat untuk merata-rata semua vektor dukungan NSV:

$$b = \frac{1}{NSV} \sum_{i=1}^{NSV} (w \cdot x_i - y_i) \quad (2.45)$$

2.5.4 *Dual form*

Menulis aturan klasifikasi dalam bentuk *unconstrained dual form* mengungkapkan bahwa *hyperplane* margin maksimum dan oleh karena itu tugas klasifikasi adalah hanya fungsi dari vektor pendukung, data pelatihan yang terletak pada margin.

Menggunakan fakta, bahwa $\|w\|^2 = w.w$ dan menggantikan

$$w = \sum_{i=1}^{n} (\propto_i y_i x_i) \quad (2.46)$$

seseorang dapat menunjukkan bahwa dual dari SVM mengurangi untuk masalah optimasi berikut:

Maksimalkan (dalam α_i)

$$\tilde{L}(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j {x_i}^T x_j = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j k(x_i, x_j) \quad (2.47)$$

dikenakan (untuk setiap i = 1,…,n)

$\alpha_i \geq 0$, dan constraint dari minimisasi di b

$$\sum_{i=1}^{n} \alpha_i y_i = 0 \quad (2.48)$$

Di sini kernel didefinisikan oleh $k(x_i, x_j) = x_i \cdot x_j$. (2.49)

Istilah α merupakan representasi ganda untuk vektor berat dalam hal training set :

$$w = \sum_{i} \alpha_i y_i x_i \quad (2.50)$$

2.5.5 *Hyperplanes* bias dan tidak bias

Untuk alasan kesederhanaan, kadang-kadang diperlukan bahwa *hyperplane* melewati sistem koordinat asli. *Hyperplane* seperti ini disebut *unbiased*, sedangkan *hyperplane* umum yang tidak harus melewati titik asal disebut bias. Sebuah hyperplane tidak bias dapat dilaksanakan dengan menetapkan b = 0 dalam masalah optimasi primal.

$$\sum_{i=1}^{n} \alpha_i y_i = 0 \quad (2.48)$$

2.5.6 *Properties*

SVM milik keluarga pengklasifikasi linear umum. Mereka juga dapat dianggap sebagai kasus khusus dari regularisasi Tikhonov. Sebuah properti khusus adalah bahwa mereka secara simultan meminimumkan kesalahan klasifikasi empiris dan memaksimalkan margin geometrik, maka mereka juga dikenal sebagai pengklasifikasi margin maksimal.

2.5.7 *Soft Margin*

Di tahun 1995, Corinna Cortes dan Vladimir Vapnik menyarankan ide margin maksimal dimodifikasi yang memungkinkan untuk contoh mislabeled. Jika terdapat hyperplane yang dapat memecah contoh "ya" dan "tidak ", metode Margin Soft akan memilih hyperplane yang membagi contoh-contoh sebersih mungkin. Metode ini memperkenalkan variabel slack, ξi, yang mengukur tingkat kesalahan klasifikasi dari datum xi

$$y_i\ (\mathrm{w}\ .\ w_i - b) \geq 1 - \varepsilon_i \quad 1 \leq \mathrm{i} \leq \mathrm{n} \tag{2.51}$$

Fungsi objektif ini kemudian meningkat dengan fungsi yang menghukum non-nol ε_i, dan optimisasi menjadi trade off antara margin yang besar, dan denda kesalahan kecil. Jika fungsi penalty adalah linier, masalah optimasi menjadi:

$$\min_{w,\varepsilon} \{\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \varepsilon i \ \} \tag{2.52}$$

dikenakan (untuk setiap i = 1,…,n)

$$y_i\ (\mathrm{w}\ .\ w_i - b) \geq 1 - \varepsilon_i \quad \varepsilon_i \geq 0 \tag{2.53}$$

Constraint dalam (2.43) bersama dengan tujuan untuk meminimalkan ‖w‖ dapat diselesaikan dengan menggunakan pengali Lagrange seperti yang dilakukan di atas. Satu telah lalu untuk menyelesaikan masalah berikut

$$\min_{w,\varepsilon,b} \max_{\alpha,\beta} \{\frac{1}{2}\|w\|^2 + \quad C\sum_{i=1}^{n} \varepsilon_i - \sum_{i=1}^{n} \alpha_i\,[y_i(w\ .x_i - b) - 1 + \ \varepsilon_i] - \sum_{i=1}^{n} \beta_i \varepsilon_i\} \tag{2.54}$$

dengan $\alpha_i, \beta_i \geq 0$.

Keuntungan kunci dari fungsi penalti linier adalah bahwa variabel slack lenyap dari masalah ganda, dengan konstanta C muncul hanya sebagai constrain tambahan pada pengganda Lagrange. Untuk formulasi di atas dan dampak besar dalam prakteknya, Cortes dan Vapnik menerima 2008 ACM Paris Award Kanellakis. Fungsi penalty nonlinear telah digunakan, terutama untuk mengurangi efek outlier pada classifier.

2.5.8 *Non linier kernel*

Dua keluarga kernel yang umum digunakan adalah kernel polynomial dan fungsi basis radial. Kernel polynomial adalah bentuk dari $K(x,z) = (1 + x^T z)^d$. Dalam kasus d = 1 adalah kernel linier, yang telah dibicarakan sebelumnya.

Dalam kasus d = 2 akan memberikan kernel kuadratik, dan sangat umum digunakan.

BAB III
METODE PENELITIAN

3.1 Data Review Film

Data review film dalam teks berbahasa Inggris diambil dari situs http://www.cs.cornell.edu/people/pabo/movie-review-data/ . Ada beberapa data review film yang tersedia dalam situs tersebut, untuk penelitian ini diambil data dengan ukuran file terbesar. Data terbagi menjadi opini positif dan opini negatif. Opini tersebut ditulis dalam bahasa Inggris yang dikumpulkan dari beberapa blog *moview reviewer*. Contoh dari data review film untuk opini positif teks berbahasa Inggris adalah sebagai berikut :

"the rock is destined to be the 21st century's new " conan " and that he's going to make a splash even greater than arnold schwarzenegger , jean-claud van damme or steven".

Contoh untuk opini negatif teks berbahasa Inggris adalah sebagai berikut :

"exploitative and largely devoid of the depth or sophistication that would make watching such a graphic treatment of the crimes bearable".

Data opini berbahasa Indonesia diambil dari rubrik Bali Terkini yang dimuat di harian Bali Post dari bulan Januari 2010 sampai Februari 2011. Tahap pengolahan awal adalah proses tokenisasi terhadap baris-baris kalimat opini.

Contoh opini positif dalam teks berbahasa Indonesia adalah sebagai berikut :

"Salut terhadap langkah inovatif jajaran Satlantas Polres Gianyar, mengadakan kontrak kerja sama dengan rumah sakit Ganesa dan Rumah Sakit Hari Santhi." Sedangkan contoh untuk opini negatif adalah "Belum genap tiga tahun dibangun,

jalan yang menghubungkan Desa Tanglad dengan Desa Sekartaji, Nusa Penida rusak berat."

Dalam penelitian ini tokenisasi secara garis besar dilakukan dengan memecah kalimat menjadi token (kata) dengan mengabaikan karakter non alphabet. Semua huruf kapital diubah menjadi huruf kecil sehingga token tersebut dapat diurutkan secara alfabetik dan diproses selanjutnya.

3.2 Perangkat Keras dan Perangkat Lunak Pendukung

Aplikasi klasifikasi teks menggunakan metode NBC dan metode SVM dibangun menggunakan bahasa pemrograman PHP sehingga dibutuhkan beberapa perangkat lunak sebagai berikut :

1. Web browser (Mozzila Firefox atau Internet Explorer)
2. Web server (Apache dalam Xampp)
3. Editor (Macromedia Dreamweaver)
4. PHP
5. Borland C++

Sedangkan perangkat keras yang digunakan adalah personal komputer dengan prosesor Intel core i5 2.40 GHz dan memory 2 GB, VGA ATI Radeon.

3.3 Rancangan Klasifikasi Teks dengan NBC

Berangkat dari aturan bayes P(X|Y) = $\frac{P(X)P(Y|X)}{P(Y)}$ maka kita membuat sebuah asumsi tentang bagaimana kita menghitung probabilitas dari kemunculan dokumen yang setara dengan perkalian (*product*) dari probabilitas kemunculan tiap kata di dalamnya. Hal ini menyebabkan tidak ada hubungan antara kata yang

satu dengan kata yang lainnya. Asumsi independen ini tidak sepenuhnya benar, banyak kata-kata lebih sering muncul bersamaan daripada muncul secara individual, tapi hal ini menyederhanakan proses klasifikasi.

Kita dapat mengestimasi probabilitas kemunculan sebuah kata sebagai sebuah sentiment positif atau negatif dengan melihat kumpulan data latih sentiment positif dan negatif dan menghitung seberapa sering kata tersebut muncul dalam setiap kelas. Hal ini yang membuat training ini sebagai pembelajaran terbimbing.

Sehingga persamaan untuk menyelesaikan permasalahan ini adalah sebagai berikut:

$$\text{P(sentiment | sentence)} = \frac{\text{P(sentiment) P(sentence | sentiment)}}{\text{P(sentence)}} \quad (3.1)$$

Kita dapat menghilangkan pembagi, karena nilainya konstan untuk setiap kelas, dan kita hanya perlu meranking daripada menghitung probabilitas persisnya. Kita dapat menggunakan asumsi independen untuk menyatakan P(sentence | sentiment) sebagai perkalian (*product*) dari P(token | sentiment) untuk semua token dalam kalimat tersebut. Sehingga kita mengestimasi P(token | sentiment) sebagai count(this token in class) + 1 / count(all tokens in class) + count(all tokens)

$$\text{P(token | sentiment)} = \frac{jumlah\ kemunculan\ token\ dalam\ kelas\ +1}{jumlah\ semua\ token\ dalam\ kelas + jumlah\ semua\ token} \quad (3.2)$$

Angka 1 dan jumlah semua dari semua token disebut *'add one'* atau penghalusan *Laplace*, dan menghentikan nilai 0 sebagai hasil perkalian. Jika kita tidak menggunakan hal ini maka kalimat dengan token yang tak terdefinisikan akan menghasilkan nilai nol.

Fungsi klasifikasi dimulai dengan menghitung probabilitas prior (kemungkinan kalimat menjadi positif atau negatif sebelum mengacu pada token) berdasarkan pada jumlah data latih positif ataupun negatif. Dalam persamaan 3.1 probabilitas prior dinyatakan sebagai P(sentiment).

$$P(\text{sentiment}_j) = \frac{jumlah\ record\ pada\ kelas_j}{jumlah\ semua\ record} \quad (3.3)$$

Ringkasan algoritma untuk *Naïve Bayes Classifier* adalah sebagai berikut :

C. Proses pelatihan. Input adalah dokumen-dokumen contoh yang telah diketahui kategorinya.

3. jumlah semua token ← jumlah semua kata yang unik dari dokumen-dokumen contoh
4. Untuk setiap kelas sentimen lakukan :
   d. Jumlah record pada $kelas_j$ ← jumlah record yang berada pada kelas j
   e. Hitung $P(\text{sentiment}_j)$ dengan persamaan 3.3
   f. Untuk setiap kata w_k pada daftar semua token lakukan :
      2. Hitung $P(\text{token}_k | \text{sentiment}_j)$ dengan persamaan 3.2

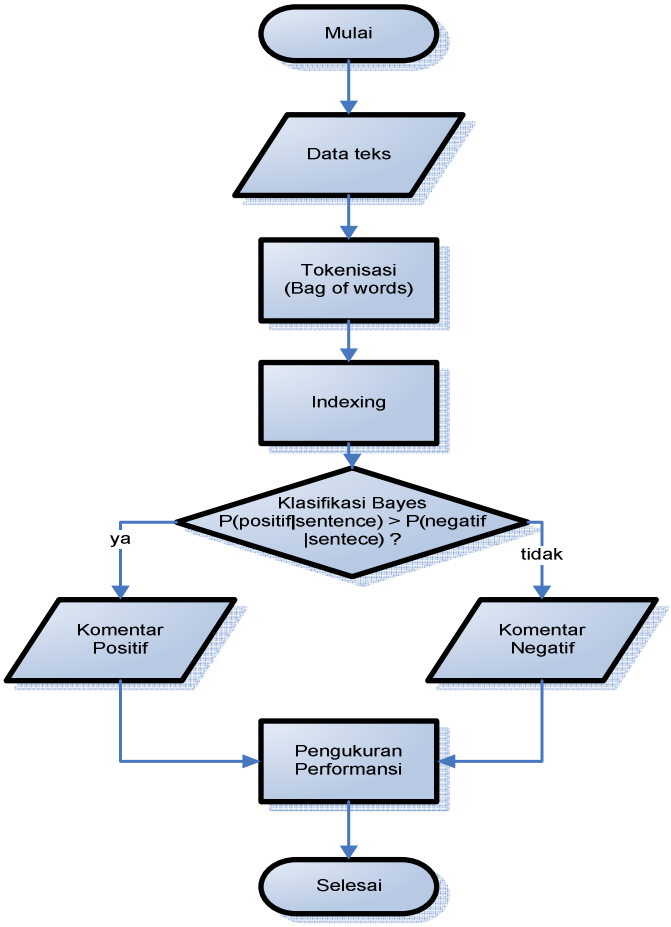
D. Proses klasifikasi. Input adalah dokumen yang belum diketahui kategorinya :

1. Hasilkan probabilitas untuk masing- masing kelas sesuai dengan persamaan 3.1 dengan menggunakan $P(\text{sentiment}_j)$ dan $P(\text{token}_k | \text{sentiment}_j)$ yang telah diperoleh dari pelatihan.
2. Probabilitas kelas maksimum adalah kelas sentiment terpilih hasil klasifikasi.

Akurasi dihitung dengan :

$$Akurasi = \frac{Jumlah\ Klasifikasi\ Benar}{Jumlah\ Dokumen\ Uji\ Coba} \text{ x } 100\%$$

Tahapan proses klasifikasi opini dengan NBC ditunjukkan dalam diagram alir seperti pada gambar 3.1



Gambar 3.1 Diagram Alir Klasifikasi dengan NBC

3.4 Rancangan Klasifikasi Teks dengan SVM

Proses klasifikasi menggunakan SVM dimulai dengan mengubah text menjadi data vector. Vector dalam penelitian ini memiliki dua komponen yaitu dimensi (*word id*) dan bobot. Bobot ini sering dikombinasikan ke dalam sebuah nilai tf-idf, secara sederhana dengan mengalikan mereka bersama-sama. Ada

banyak variasi pada gagasan dasar tf-idf, tetapi implementasi langsung akan terlihat seperti:

```
<?php
 $tfidf = $term_frequency *  // tf
 log( $total_document_count / $documents_with_term, 2);/idf
?>
```

idf adalah jumlah total dokumen atas hitungan yang berisi istilah tersebut. Jadi, jika ada 50 dokumen dalam koleksi, dan dua di antaranya terdapat istilah yang menjadi pertanyaan, IDF akan menjadi 50 / 2 = 25. Untuk menjadi akurat, kita harus memasukkan query dalam perhitungan IDF, jadi jika dalam koleksi ada 50 dokumen, dan 2 berisi istilah dari query, perhitungan yang sebenarnya akan (50 +1) /(2 +1) = 51 / 3.

Diambil log dari IDF untuk memberikan beberapa penghalusan. Jika sebuah istilah A direpresentasikan dalam x buah dokumen, dan istilah B sejumlah 2x kali, maka istilah A adalah istilah yang lebih spesifik yang harus memberikan hasil yang lebih baik, tetapi belum tentu dua kali lebih baik. Kelembutan dari log adalah pemecahan perbedaan-perbedaan ini.

Dokumen dapat dinyatakan sebagai list dari term. Sebuah contoh mapping dokumen untuk term (istilah) dalam bahasa php adalah sebagai berikut :

```
<?php

function getIndex() {
     $collection = array(
      1 => 'this string is a short string but a good string',
      2 => 'this one isn\'t quite like the rest but is here',
      3 => 'this is a different short string that\' not as short'
     );

    $dictionary = array();
    $docCount = array();

    foreach($collection as $docID => $doc) {
        $terms = explode(' ', $doc);
        $docCount[$docID] = count($terms);
```

```
        foreach($terms as $term) {
           if(!isset($dictionary[$term])) {
             $dictionary[$term] = array('df' => 0, 'postings' =>
array());
           }
      if(!isset($dictionary[$term]['postings'][$docID])) {
        $dictionary[$term]['df']++;
        $dictionary[$term]['postings'][$docID] = array('tf' => 0);
      }

      $dictionary[$term]['postings'][$docID]['tf']++;
   }
}

return array('docCount' => $docCount, 'dictionary' => dictionary);
}
?>
```

Kita kemudian menormalisasi tiap komponen dengan panjang dari vector sehingga bobot tersebut dinyatakan dalam 1 unit panjang.

Masalah klasifikasi adalah sesuatu yang telah kita bahas sebelumnya, tetapi pada umumnya adalah tentang belajar yang memisahkan dua set contoh, dan berdasarkan hal tersebut menempatkan dengan benar contoh-contoh yang tak terlihat ke salah satu himpunan. Contohnya bisa berupa filter spam, di mana, diberikan pelatihan himpunan mail spam dan non-spam diharapkan untuk mengklasifikasikan email sebagai spam atau bukan spam.

SVM adalah sistem untuk melakukan hal itu, tetapi mereka hanya peduli tentang titik dalam ruang, daripada email atau dokumen. Untuk tujuan ini 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.

Format data input untuk klasifikasi SVM dalam penelitian ini adalah :

+1 1:0.049 45:0.0294

Dengan masukan yang pertama +1 atau -1 menyatakan dua kelas (atau 0 untuk data yang akan diklasifikasi). Angka kedua menyatakan dimensi (*row id*) dan angka ketiga (setelah karakter ":") menyatakan bobot dari term tersebut, tiap term dalam sebuah dokumen dipisahkan dengan spasi.

Intuisi yang mendorong SVM sebagai garis terbaik yang memisahkan kedua kelas adalah yang memiliki margin terbesar diantaranya dan contoh titik pelatihan terdekat di kedua sisinya. Oleh karena itu, vektor contoh penting adalah vektor yang menentukan margin tersebut - yang paling dekat dengan *dividing lines*. Ini adalah *support vector*, dan merupakan kombinasi dari vector-vector yang memberikan keputusan fungsi (kelas atau bukan kelas) untuk *classifier* SVM. Fungsi klasifikasi dalam contoh kode adalah sebagai berikut:

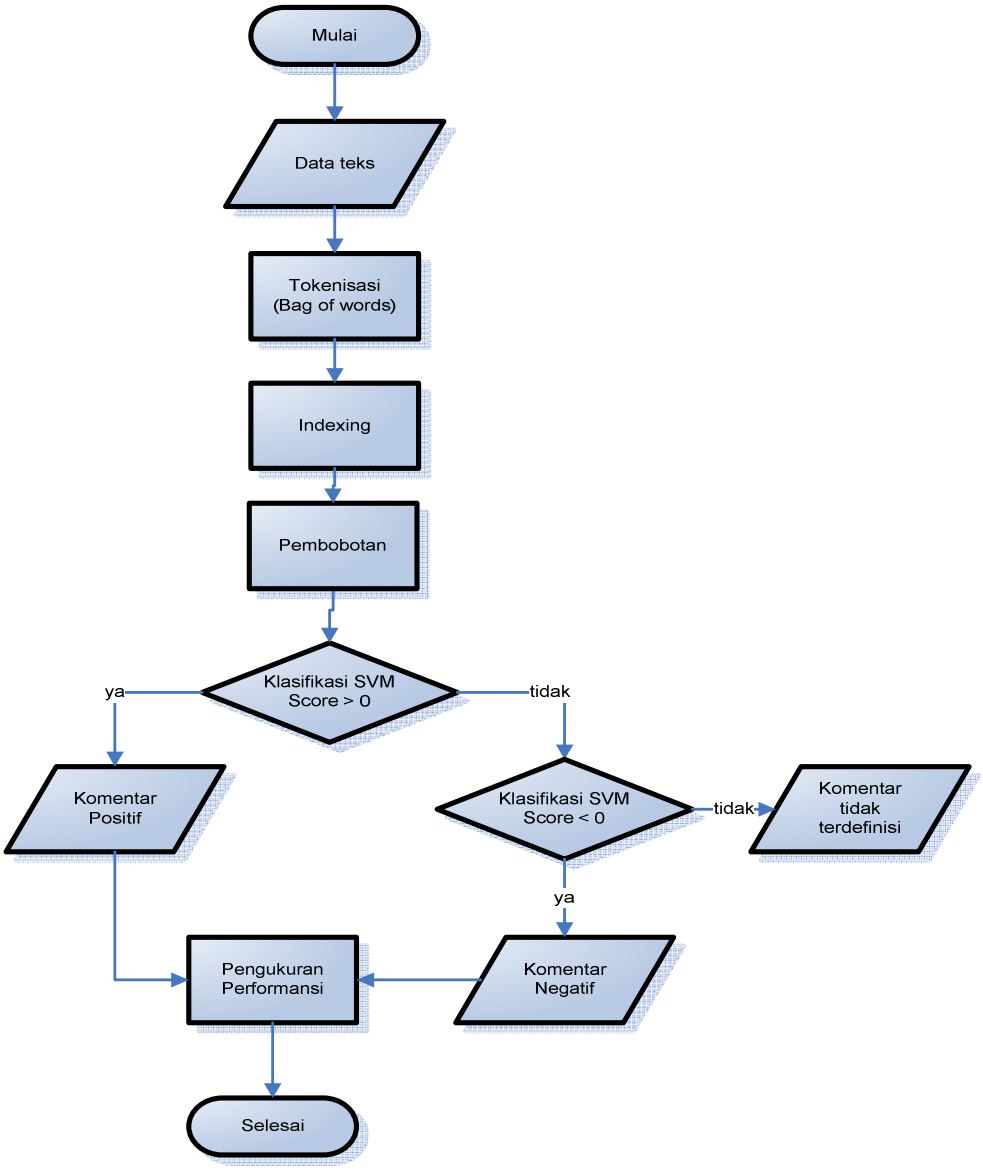
```
<?php
protected function classify($rowID) {
  $score = 0;
  foreach($this->lagrangeMults as $key => $value) {
    if($value > 0) {
      $score += $value * $this->targets[$key] * $this-
>kernel($rowID, $key);
    }
  }
  return $score - $this->bias;
}
?>
```

Penilaian kemudian dibuat dengan menilai *score* apakah positif atau negatif yang merepresentasikan di sisi mana dari garis pemisah dokumen berada. Sejauh ini fungsi kernel dapat diasumsikan sebagai *dot product* antara dua vector.

Namun, *dot product* dalam fungsi klasifikasi SVM tidak harus sebagai sebuah *dot product*, yang mengijinkan kita untuk memasang fungsi dengan tipe yang berbeda. Fungsi ini dapat menentukan secara efektif sebuah pemetaan

dimensi pada ruang lingkup permasalahan, dan beberapa ruang lingkup dimensional yang lebih tinggi dimana kemungkinan data akan dipisahkan secara linier. Trik ini mengijinkan hasil klasifikasi yang baik pada sumber berbeda dan bervariasi, meskipun tidak mendukung beban dalam hal kompleksitas tambahan dan biaya komputasi.

Diagram alir proses klasifikasi dengan SVM ditunjukkan oleh gambar 3.2



Gambar 3.2 Diagram Alir Klasifikasi dengan SVM

3.5 Pengelolaan data dalam percobaan.

Data dibagi menjadi data latih dan data uji. Data asli juga mengalami penukaran posisi record di dalamnya dengan harapan akan mendapatkan variasi data latih dan data uji. Akan dibentuk 7 variasi data. Untuk mengukur performansi dari metode NBC dan metode SVM maka pembagian data latih dan data uji dilakukan dengan proporsi seperti pada tabel 3.1:

Tabel 3.1
Pembagian data latih dan data uji untuk teks berbahasa Inggris

No	Data latih	Data uji
1	2000 opini positif dan 2000 opini negatif	3331 opini positif
2	2000 opini positif dan 2000 opini negatif	3331 opini negatif
3	3000 opini positif dan 3000 opini negatif	2331 opini positif
4	3000 opini positif dan 3000 opini negatif	2331 opini negatif
5	4000 opini positif dan 4000 opini negatif	1331 opini positif
6	4000 opini positif dan 4000 opini negatif	1331 opini negatif
7	5000 opini positif dan 5000 opini negatif	331 opini positif
8	5000 opini positif dan 5000 opini negatif	331 opini negatif
9	5200 opini positif dan 5200 opini negatif	131 opini positif
10	5200 opini positif dan 5200 opini negatif	131 opini negatif
11	5300 opini positif dan 5300 opini negatif	31 opini positif
12	5300 opini positif dan 5300 opini negatif	31 opini negatif

Masing-masing metode diujicoba menggunakan data pada tabel tersebut di atas dan diukur akurasinya.

Untuk data berbahasa Indonesia akan dilakukan pengujian dengan pembagian data latih dan data uji seperti pada tabel 3.2:

Tabel 3.2
Persentase data latih dan data uji pada opini berbahasa Indonesia

No	Data latih	Data uji
1	1817 opini positif dan 1817 opini negatif	3000 opini positif
2	1817 opini positif dan 1817 opini negatif	3000 opini negatif
3	2817 opini positif dan 2817 opini negatif	2000 opini positif
4	2817 opini positif dan 2817 opini negatif	2000 opini negatif
5	3817 opini positif dan 3817 opini negatif	1000 opini positif
6	3817 opini positif dan 3817 opini negatif	1000 opini negatif
7	4317 opini positif dan 4317 opini negatif	500 opini positif
8	4317 opini positif dan 4317 opini negatif	500 opini negatif
9	4717 opini positif dan 4717 opini negatif	100 opini positif
10	4717 opini positif dan 4717 opini negatif	100 opini negatif
11	4787 opini positif dan 4787 opini negatif	30 opini positif
12	4787 opini positif dan 4787 opini negatif	30 opini negatif

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi Metode *Naïve Bayes Classifier*

Sebagai data uji adalah sejumlah data yang ada pada file data dipotong dari *record* paling akhir untuk masing-masing data opini positif dan opini negatif. Data latih diperoleh dari penggabungan file data opini positif dan file data opini negatif sisa dari penggunaan untuk data uji. Dibentuk enam variasi data masing-masing untuk file data positif dan negatif dengan menukar susunan *record*, dengan demikian akan terdapat enam variasi data pengujian untuk tiap proporsi data uji.

Beberapa tahapan dalam proses klasifikasi pada penelitian ini dapat diuraikan sebagai berikut :

4.1.1 Persiapan data

Pada tahap persiapan ini data yang digunakan disimpan dalam file teks dimana setiap *record* berisi sebuah kalimat dengan *sentiment* positif atau negatif. Kumpulan *record* positif disimpan dalam file dengan nama rt-polarity.pos sedangkan kumpulan *record* negatif disimpan dalam file dengan nama rt-polarity.neg. Untuk menguji pengaruh probabilitas priori dalam menentukan akurasi maka disiapkan data dengan mengurangi 1500 *record* teratas dari kedua file yang disebutkan sebelumnya.

Untuk data berbahasa Indonesia, pengujian pengaruh probabilitas priori dalam menentukan akurasi dilakukan dengan menyiapkan data dengan mengurangi 1300 *record* teratas dari file data latih bahasa Indonesia.

4.1.2 Penyusunan *Bag of Words*

Proses ini membaca tiap kata yang ada dalam file data latih dan mengelompokkannya sebagai *token* dalam variabel *index*. Proses dilanjutkan dengan menghilangkan *stop words* lalu mengubah padanan kata untuk kalimat yang memiliki unsur pembalik seperti kata “not” dan kata “tidak”. Dalam proses ini pula dihitung juga properti-properti lain yang mengikuti perhitungan probabilitas Bayes seperti jumlah semua *record* atau dokumen, jumlah *record* pada masing-masing kelas, jumlah kemunculan *token* dalam masing-masing kelas, jumlah semua *token*, dan jumlah semua *token* pada masing-masing kelas.

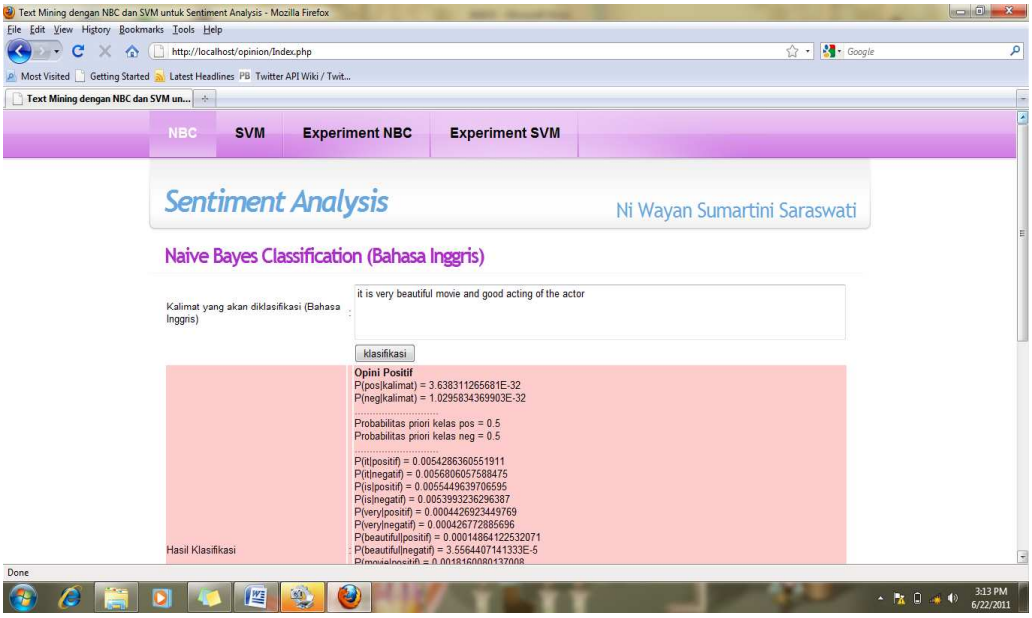
Baik pada data berbahasa Inggris maupun data berbahasa Indonesia berlaku bahwa tidak ada data latih yang sama dengan data uji.

4.1.3 Proses Klasifikasi

Inti dari proses klasifikasi adalah menentukan sebuah kalimat sebagai anggota kelas opini positif atau sebagai anggota kelas opini negatif berdasarkan nilai perhitungan probabilitas Bayes yang lebih besar ($classScores). Jika hasil probabilitas Bayes kalimat tersebut untuk kelas opini positif lebih besar maka kalimat tersebut masuk kategori opini positif demikian juga sebaliknya.

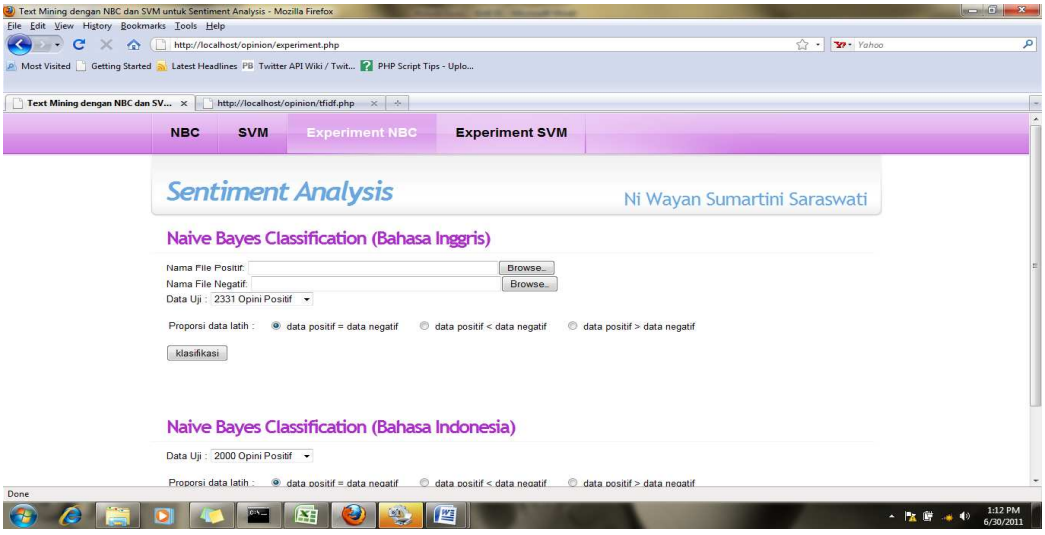
4.1.4 Antar muka sistem

Ada dua halaman untuk implementasi NBC seperti ditunjukkan oleh gambar 4.1 dan gambar 4.2



Gambar 4.1
Antar muka klasifikasi untuk 1 kalimat dengan NBC

Gambar 4.1 di atas adalah sebuah contoh klasifikasi kalimat yang secara tekstual bernada positif dan hasil klasifikasi NBC dari software menunjukkan hasil sebagai opini positif. Hasil disertai dengan probabilitas kata yang menyusun kalimat dan probabilitas kalimat sehingga bisa diambil kesimpulan apakah kalimat termasuk kelas opini positif atau opini negatif.



Gambar 4.2
Antar muka klasifikasi untuk data percobaan dengan NBC

Gambar 4.2 menunjukkan masukan untuk file data latih positif dan negatif, opsi data uji dan opsi keseimbangan data positif dan negatif dalam data latih. Proses klasifikasi dilakukan dengan menekan tombol klasifikasi.

4.2 Implementasi Metode *Support Vector Machine*

Dalam implementasi metode SVM digunakan data yang sama seperti yang digunakan dalam percobaan menggunakan metode NBC. Dalam percobaan ini digunakan SVM *light* hasil penelitian dari Thorsten Joachims. SVM *light* ini ditulis dalam bahasa C. Sebelum data masuk ke dalam mesin pengklasifikasi SVM maka terlebih dahulu data teks diubah ke dalam bentuk data vektor.

4.2.1 Data Teks Menjadi Data Vektor

Data latih dan data uji secara bersamaan akan diubah menjadi data vektor. Sebagai contoh dalam pengubahan data teks menjadi data vektor digunakan sebuah kalimat dari data opini positif seperti di bawah ini :

“the rock is destined to be the 21st century's new " conan " and that he's going to make a splash even greater than arnold schwarzenegger , jean-claud van damme or steven segal . “

Format data vektornya adalah sebagai berikut :

1 1:3.8883373704202 2:8.425994156363 3:1.8590808230985 4:10.38019046675 5:3.3113532272598 6:3.6004711116064 7:0.87239582655114 8:10.795227966029 9:13.38019046675 10:5.7291387755709 11:10.564316767579 12:12.38019046675 13:1.1166278269292 14:2.2651468165881 15:7.425994156363 16:7.0051510354029 17:5.298041425396 18:1.0795519127561 19:11.058262371862 20:5.0314623125188 21:10.572835544692 22:4.0515155394219 23:9.5728355446922 24:10.210265465308 25:0.78843439072684 26:13.38019046675 27:9.9207588481125 28:12.38019046675 29:4.4524125046675 30:8.8566285106928 31:13.38019046675 32:0.046615789291697 33:0.81057216330319

Angka 1 pada karakter pertama menyatakan data tersebut masuk dalam kelas data opini positif. Jika data adalah dari kelas opini negatif maka angka

tersebut diganti dengan -1. Kata “the” pada kalimat di atas digantikan dengan 1:3.8883373704202 pada data vektor yang berarti 1 sebagai id kata untuk “the” dan 3.8883373704202 sebagai bobot tf-idf untuk kata “the” dalam file. Begitu seterusnya sehingga semua kata dalam kalimat terwakili oleh data vektor. Untuk kata yang sama muncul lebih dari sekali dalam sebuah kalimat akan diwakili oleh sebuah data vektor saja dengan nilai tf-idf yang bersesuaian. Data vektor untuk sebuah kalimat diurutkan berdasarkan id kata dari terkecil hingga terbesar.

Proses pengubahan data teks menjadi data vektor dilakukan dengan membaca kata satu persatu dan menghitung nilai tf-idf. Nilai tf-idf adalah kemunculan kata (*term frequency*) dalam kalimat dikalikan log jumlah dokumen/*record* dibagi jumlah dokumen/*record* yang mengandung kata yang dimaksud.

4.2.2 Fungsi Kernel dan Nilai Bias

Dalam percobaan SVM digunakan dua jenis kernel yaitu model linier dan model *polynomial*. Model p*olynomial* menggunakan nilai *degree* 2. Untuk masing-masing tipe kernel diberikan variasi pengunaan bias dan yang tidak menggunakan bias.

4.2.3 Proses Klasifikasi

Pemrosesan SVM dilakukan dalam dua tahap yaitu pelatihan dan klasifikasi itu sendiri. Input dari proses pelatihan berupa data latih dalam bentuk file teks data vektor (train.txt). Output dari proses pelatihan ini adalah file model.dat yang berisi *support vector* (vektor terpilih yang membentuk hyperplane). Perintah untuk pelatihan adalah sebagai berikut “svm_learn [option]

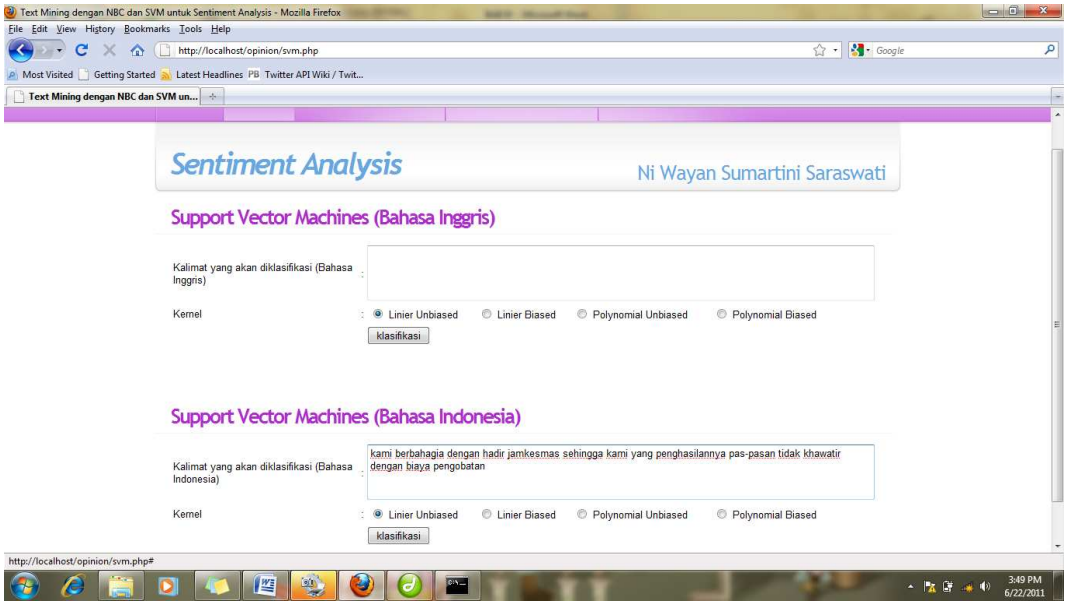
train.txt model.dat". Pada posisi [option] kita akan menggunakan variasi kernel dan variabel bias. Berikut adalah perintah masing-masing :

a. Linier tanpa bias : svm_learn
b. Linier dengan bias : svm_learn –b 0
c. *Polynomial* tanpa bias : svm_learn –t 1 –d 2
d. *Polynomial* dengan bias : svm_learn –t 1 –d 2 –b 0

Masukan dari proses klasifikasi adalah file teks data vektor. Proses klasifikasi dilakukan dengan perintah "svm_classify [option] test.txt model.dat output.dat". Hasil klasifikasi disimpan dalam file output.dat berupa *record* nilai real. *Record* pada output.dat bersesuaian dengan *record* pada test.txt dengan pengertian kalimat atau *record* pertama jika diklasifikasikan hasilnya disimpan pada *record* pertama di file output.dat. Nilai real pada file output.dat menyatakan kelas kalimat. Jika nilai tersebut lebih besar dari nol maka kalimat masuk kelas opini positif, dan jika nilai *record* lebih kecil dari nol maka kalimat masuk kelas opini negatif.

4.2.4 Antar Muka Sistem

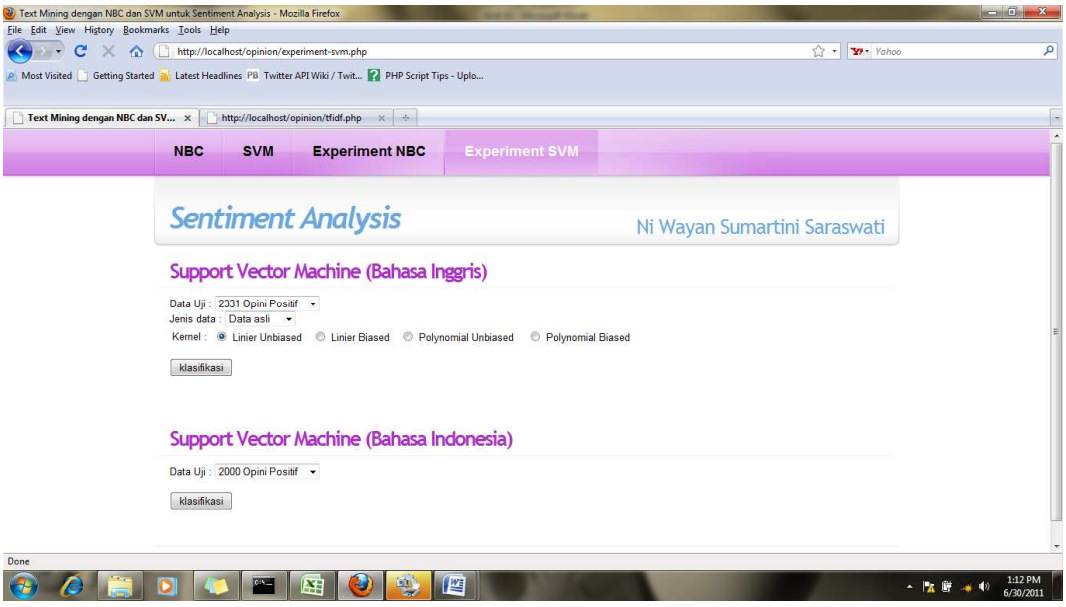
Untuk menerima masukan dari user digunakan dua halaman. Halaman pertama adalah klasifikasi untuk 1 kalimat dan halaman kedua adalah klasifikasi untuk data percobaan. Antar muka hasil implementasi ditunjukkan oleh gambar 4.3 dan gambar 4.4.



Gambar 4.3

Antar muka klasifikasi SVM untuk satu kalimat

Gambar 4.3 adalah contoh klasifikasi opini positif bahasa Indonesia untuk metode SVM. Ada opsi pilihan kernel yang digunakan. Klasifikasi dilakukan dengan menekan tombol klasifikasi.

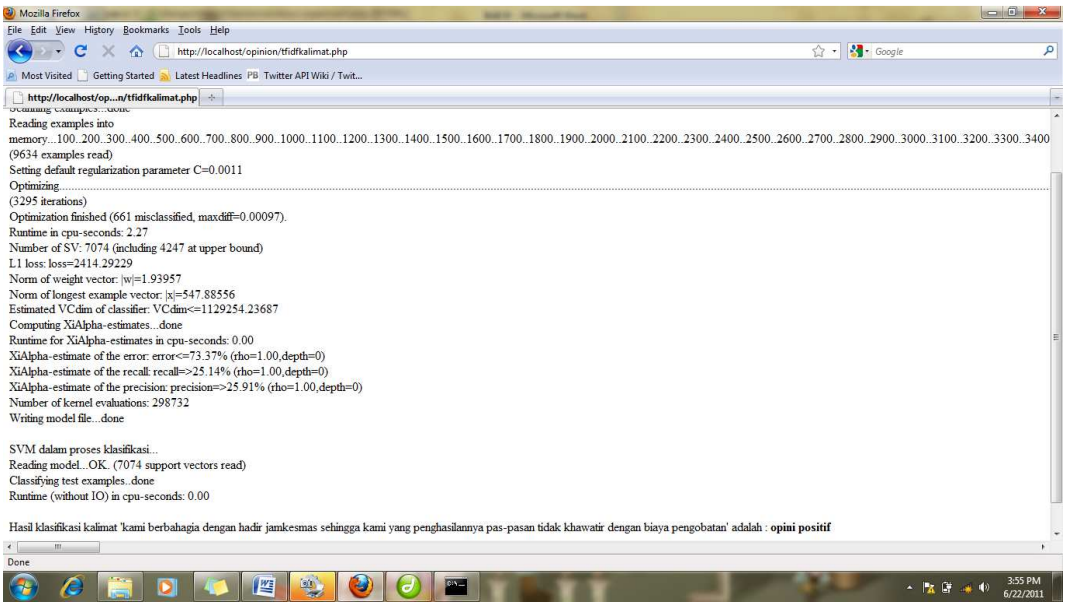


Gambar 4.4

Antar muka klasifikasi SVM untuk data percobaan

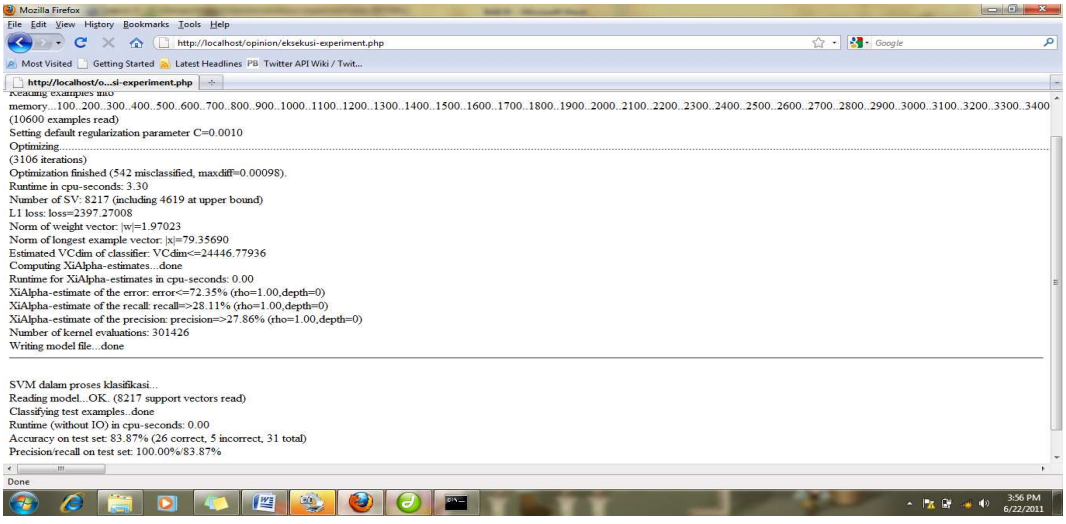
Gambar 4.4 adalah contoh eksperimen dengan 2331 data uji opini positif dimana pilihan kernel adalah *linier unbiased* untuk data asli.

Untuk hasilnya seperti ditunjukkan oleh gambar 4.5 dan gambar 4.6



Gambar 4.5
Antar muka hasil klasifikasi SVM untuk satu kalimat

Dari gambar 4.5 hasil klasifikasi dari kalimat yang diujikan ditunjukkan oleh kalimat terakhir yaitu opini positif.



Gambar 4.6
Antar muka hasil klasifikasi SVM untuk data percobaan

Dari gambar 4.6 merupakan contoh hasil klasifikasi untuk data uji dalam bentuk file.

4.3 Hasil Percobaan

4.3.1 Variasi keseimbangan data latih pada metode NBC

Untuk mengetahui pengaruh proporsi data latih antara data positif dan data negatif terhadap akurasi maka dilakukan percobaan dengan jumlah data positif dan data negatif yang berbeda.

4.3.1.1 Data berbahasa Inggris

Hasil eksperimen klasifikasi dengan metode NBC untuk data berbahasa Inggris ditunjukkan oleh tabel 4.1 dan tabel 4.2

Tabel 4.1
Hasil variasi keseimbangan data latih metode NBC
untuk data uji positif berbahasa Inggris

Data uji	data positif = data negatif	data positif <data negatif	data positif>data negatif
31 data uji positif	90,32	51,61	99,47
131 data uji positif	80,91	33,58	99,44
331 data uji positif	76,13	30,51	99,23
1331 data uji positif	75,58	21,26	99,54
2331 data uji positif	75,2	10,12	99.79

Tabel 4.2
Hasil variasi keseimbangan data latih metode NBC
untuk data uji negatif berbahasa Inggris

Data uji	data positif = data negatif	data positif <data negatif	data positif>data negatif
31 data uji negatif	93,54	99,67	35,48
131 data uji negatif	85,49	99,57	29
331 data uji negatif	82,47	99,61	32,02
1331 data uji negatif	80,09	99,39	17,43
2331 data uji negatif	77,9	99,66	6

Dari hasil percobaan pada tabel 4.1 dan tabel 4.2 diperoleh pemahaman bahwa dengan jumlah data latih positif yang lebih besar maka hasil klasifikasi akan cenderung menunjukkan sebagai opini positif. Demikian pula sebaliknya. Jika data latih negatif lebih besar hasil klasifikasi akan cenderung sebagai opini negatif. Dengan jumlah data latih yang seimbang maka hasil klasifikasi hanya ditentukan dari jumlah dan kata-kata di data uji tanpa pengaruh priori probability.

4.3.1.2 Data berbahasa Indonesia

Hasil eksperimen klasifikasi dengan metode NBC untuk data berbahasa Indonesia ditunjukkan oleh tabel 4.3 dan tabel 4.4

Tabel 4.3
Hasil variasi keseimbangan data latih metode NBC
untuk data uji positif berbahasa Indonesia

Data uji	data positif = data negatif	data positif <data negatif	data positif>data negatif
30 data uji positif	100	1,4	91,38
100 data uji positif	79	4,8	90,07
500 data uji positif	65,4	15,3	87,01
1000 data uji positif	60,4	18,59	86,9
2000 data uji positif	57,8	17,18	90,6

Tabel 4.4
Hasil variasi keseimbangan data latih metode NBC
untuk data uji negatif berbahasa Indonesia

Data uji	data positif = data negatif	data positif <data negatif	data positif>data negatif
30 data uji negatif	93,34	98,93	3,09
100 data uji negatif	90	98,58	9,34
500 data uji negatif	85	97,45	29,21
1000 data uji negatif	88,6	97,52	38,07
2000 data uji negatif	85,5	98,11	35,95

Dari tabel 4.3 dan tabel 4.4 didapatkan bahwa hal yang sama terjadi pada percobaan menggunakan data berbahasa Indonesia. Perbandingan jumlah data latih positif dan data latih negatif mempengaruhi kecenderungan hasil klasifikasi. Hasil terbaik jika terjadi keseimbangan jumlah data latih antara opini positif dan opini negatif.

4.3.2 Hasil Percobaan dengan Metode NBC dan SVM untuk Data Berbahasa Inggris.

Percobaan dilakukan dengan data yang sama antara metode SVM dan NBC. Untuk hasil percobaan dengan metode NBC ditunjukkan oleh tabel 4.5 dan tabel 4.6.

Tabel 4.5
Hasil NBC untuk data positif berbahasa Inggris

NBC data positif	data asli	data ke -1	data ke-2	data ke-3	data ke-4	data ke-5	data ke-6	**rata-rata**
31 data uji positif	90,32	67,74	93,54	70,96	83,87	74,19	80,64	**80,18**
131 data uji positif	80,91	77,09	80,91	74,8	77,86	73,28	83,2	**78,29286**
331 data uji positif	76,13	74,32	76,73	76,43	77,34	78,24	77,03	**76,60286**
1331 data uji positif	75,58	74,98	75,73	75,58	76,03	76,18	78,13	**76,03**
2331 data uji positif	75,2	74,6	74,68	74,38	74,94	75,94	77,13	**75,26714**

Hasil percobaan pada tabel 4.5 menunjukkan bahwa NBC memberikan unjuk kerja yang cukup baik pada klasifikasi opini positif berbahasa Inggris. Bahkan untuk jumlah data uji yang besar yaitu 2331 *record* mendekati 77% jumlah data latih, NBC memberikan akurasi 75%. Dari hasil percobaan itu pula didapatkan kecenderungan akurasi yang menurun seiring dengan berkurangnya data latih dan bertambahnya data uji. Hasil percobaan metode NBC untuk data negatif ditunjukkan oleh tabel 4.6

Tabel 4.6
Hasil NBC untuk data negatif berbahasa Inggris

NBC data negatif	data asli	data ke -1	data ke-2	data ke-3	data ke-4	data ke-5	data ke-6	**rata-rata**
31 data uji negatif	93,54	83,87	93,54	77,41	80,64	90,32	67,74	**83,86571**
131 data uji negatif	85,49	81,67	84,73	80,91	84,73	83,2	77,86	**82,65571**
331 data uji negatif	82,47	80,96	80,36	80,36	81,26	80,06	80,96	**80,91857**
1331 data uji negatif	80,09	79,78	79,48	78,58	78,36	78,21	77,53	**78,86143**
2331 data uji negatif	77,9	77,78	77,39	77,28	76,96	77,13	78,12	**77.50857**

Dari tabel 4.6 didapatkan bahwa NBC juga menunjukkan hasil yang baik untuk mengklasifikasikan data negatif berbahasa Inggris. Kecenderungan akurasi juga menurun untuk bertambahnya data latih dan berkurangnya data uji.

Hasil percobaan SVM untuk data berbahasa Inggris ditunjukkan oleh tabel 4.7 sampai tabel 4.14

Tabel 4.7
Hasil SVM Linier *Unbiased* untuk data positif berbahasa Inggris

Data Uji Positif dengan kernel linier *unbiased*	data asli	data ke -1	data ke-2	data ke-3	data ke-4	data ke-5	data ke-6	**rata-rata**
31 data uji positif	100	77,42	93,55	61,29	72,46	77,42	67,74	**78,55429**
131 data uji positif	100	76,34	77,86	74,05	73,33	73,28	74,05	**78,41571**
331 data uji positif	100	73,11	74,92	73,72	75,53	75,53	74,32	**78,16143**
1331 data uji positif	100	72,13	71,98	72,2	73,33	73,1	77,39	**77,16143**
2331 data uji positif	100	72,97	72,8	72,76	72,46	72,54	75,68	**77,03**
3331 data uji positif	99,94	70,25	70,28	70,4	70,34	70,76	71,78	**74,82143**

Tabel 4.8
Hasil SVM Linier Bias untuk data positif berbahasa Inggris

Data Uji Positif dengan kernel linier bias	data asli	data ke -1	data ke-2	data ke-3	data ke-4	data ke-5	data ke-6	**rata-rata**
31 data uji positif	100	77,42	93,55	64,52	77,42	77,42	67,74	**79,72429**
131 data uji positif	99,24	77,86	77,86	76,34	78,63	75,57	75,57	**80,15286**
331 data uji positif	99,4	73,41	75,23	73,72	76,13	76,13	75,53	**78,50714**
1331 data uji positif	99,55	73,18	73,33	73,7	73,92	74,15	77,16	**77,85571**
2331 data uji positif	99,74	74,09	73,92	74	73,75	73,79	76,66	**77,99286**
3331 data uji positif	99,73	71,78	71,72	71,51	71,78	72,23	74,18	**76,13286**

Tabel 4.9
Hasil SVM *Polynomial Unbiased* untuk data positif berbahasa Inggris

Data Uji Positif dengan kernel *Polynomial Unbiased*	data asli	data ke -1	data ke-2	data ke-3	data ke-4	data ke-5	data ke-6	**rata-rata**
31 data uji positif	100	80,65	96,77	70,97	77,42	77,42	74,19	**82,48857**
131 data uji positif	96,18	81,68	81,68	77,86	80,92	77,86	80,92	**82,44286**
331 data uji positif	96,07	80,06	80,36	78,85	79,76	78,85	80,97	**82,13143**
1331 data uji positif	96,39	81,44	80,99	80,92	81,44	80,92	81,44	**83,36286**
2331 data uji positif	96,05	80,87	80,22	79,71	79,84	79,41	78,34	**82,06286**
3331 data uji positif	95,08	77,78	77,48	76,52	77,03	75,89	69,08	**78,40857**

Tabel 4.10
Hasil SVM *Polynomial* Bias untuk data positif berbahasa Inggris

Data Uji Positif dengan kernel *Polynomial Bias*	data asli	data ke -1	data ke-2	data ke-3	data ke-4	data ke-5	data ke-6	**rata-rata**
31 data uji positif	100	80,65	93,55	67,74	77,42	77,42	74,19	**81,56714**
131 data uji positif	93,89	80,15	80,15	74,05	78,63	77,86	80,15	**80,69714**
331 data uji positif	93,35	76,74	77,34	76,74	77,95	77,64	77,95	**79,67286**
1331 data uji positif	93,61	76,56	76,26	75,43	75,73	75,88	79,56	**79,00429**
2331 data uji positif	92,02	92,36	75,2	74,56	74,95	74,95	77,43	**80,21**
3331 data uji positif	90,72	73,94	74,24	73,91	74,45	74,48	74,24	**76,56857**

Tabel 4.11
Hasil SVM Linier *Unbiased* untuk data negatif berbahasa Inggris

Data uji negatif dengan kernel linier *unbiased*	data asli	data ke -1	data ke-2	data ke-3	data ke-4	data ke-5	data ke-6	**rata-rata**
31 data uji negatif	93,55	100	100	96,77	93,55	93,55	96,77	**96,31286**
131 data uji negatif	96,95	97,71	96,18	96,18	96,95	98,47	98,47	**97,27286**
331 data uji negatif	97,89	98,49	97,89	97,58	97,28	97,89	98,79	**97,97286**
1331 data uji negatif	98,2	98,27	98,2	98,2	98,27	98,42	98,42	**98,28286**
2331 data uji negatif	98,11	98,2	98,2	98,2	98,24	98,24	98,37	**98,22286**
3331 data uji negatif	98,38	98,44	98,38	98,38	98,32	98,38	98,32	**98,37143**

Tabel 4.12
Hasil SVM Linier Bias untuk data negatif berbahasa Inggris

Data uji negatif dengan kernel linier bias	data asli	data ke -1	data ke-2	data ke-3	data ke-4	data ke-5	data ke-6	**rata-rata**
31 data uji negatif	96,77	100	100	96,77	96,77	96,77	96,77	**97,69286**
131 data uji negatif	98,47	98,47	97,71	97,71	98,47	99,24	98,47	**98,36286**
331 data uji negatif	98,79	99,09	98,79	98,49	98,49	98,79	99,4	**98,83429**
1331 data uji negatif	99,02	98,87	98,87	98,87	98,87	98,95	99,25	**98,95714**
2331 data uji negatif	98,84	98,93	98,93	98,88	98,93	99,01	98,84	**98,90857**
3331 data uji negatif	98,92	99,01	99,07	99,04	98,98	99,01	98,89	**98,98857**

Tabel 4.13
Hasil SVM *Polynomial Unbiased* untuk data negatif berbahasa Inggris

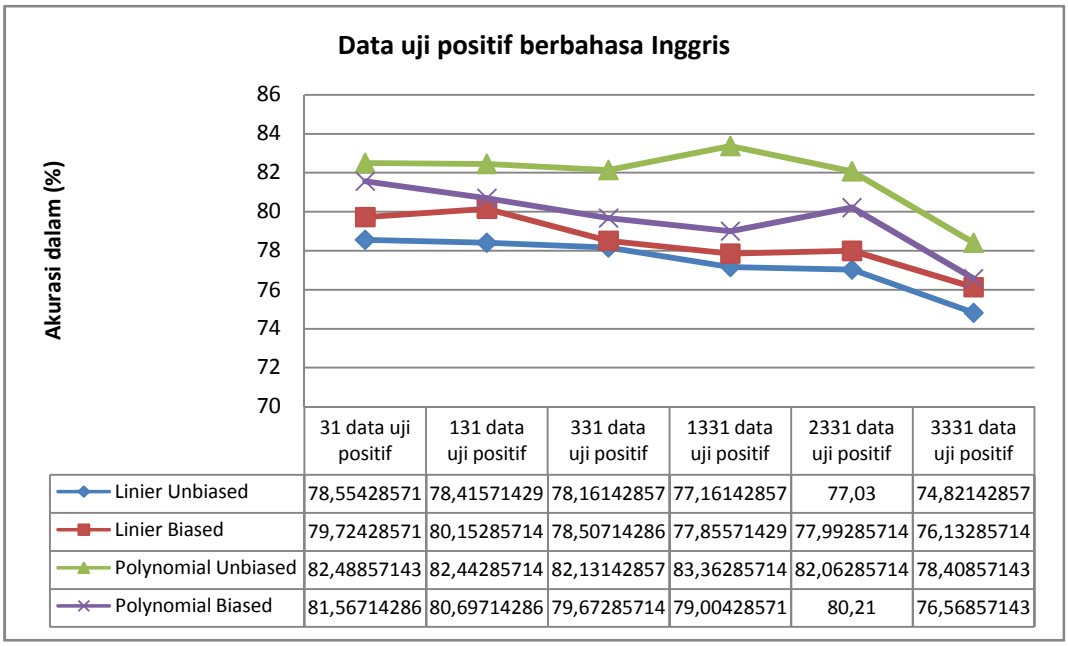
Data uji negatif dengan kernel *polynomial unbiased*	data asli	data ke -1	data ke-2	data ke-3	data ke-4	data ke-5	data ke-6	**rata-rata**
31 data uji negatif	96,77	93,55	93,55	90,32	90,32	83,87	74,19	**88,93857**
131 data uji negatif	93,13	89,31	88,55	87,02	87,02	87,79	87,79	**88,65857**
331 data uji negatif	96,07	87,92	87,31	87,01	88,22	88,82	88,52	**89,12429**
1331 data uji negatif	96,39	84,75	84,6	84,6	84,52	84,97	86,4	**86,60429**
2331 data uji negatif	96,05	81,6	81,51	81,34	81,81	81,85	77,82	**83,14**
3331 data uji negatif	82,02	80,07	78,87	79,32	79,86	79,05	71,99	**78,74**

Tabel 4.14
Hasil SVM *Polynomial* Bias untuk data negatif berbahasa Inggris

Data Uji negatif dengan kernel *polynomial* bias	data asli	data ke -1	data ke-2	data ke-3	data ke-4	data ke-5	data ke-6	**rata-rata**
31 data uji negatif	100	96,77	100	90,32	90,32	90,32	87,1	**93,54714**
131 data uji negatif	96,95	93,13	92,37	89,31	90,84	91,6	92,37	**92,36714**
331 data uji negatif	94,26	93,35	92,45	91,84	92,75	93,35	93,66	**93,09429**
1331 data uji negatif	93,46	93,54	92,94	92,94	93,16	93,39	93,91	**93,33429**
2331 data uji negatif	92,02	92,36	92,24	91,93	91,93	91,93	92,84	**92,17857**
3331 data uji negatif	90,51	90,27	90,15	89,85	89,76	90,03	90,15	**90,10286**

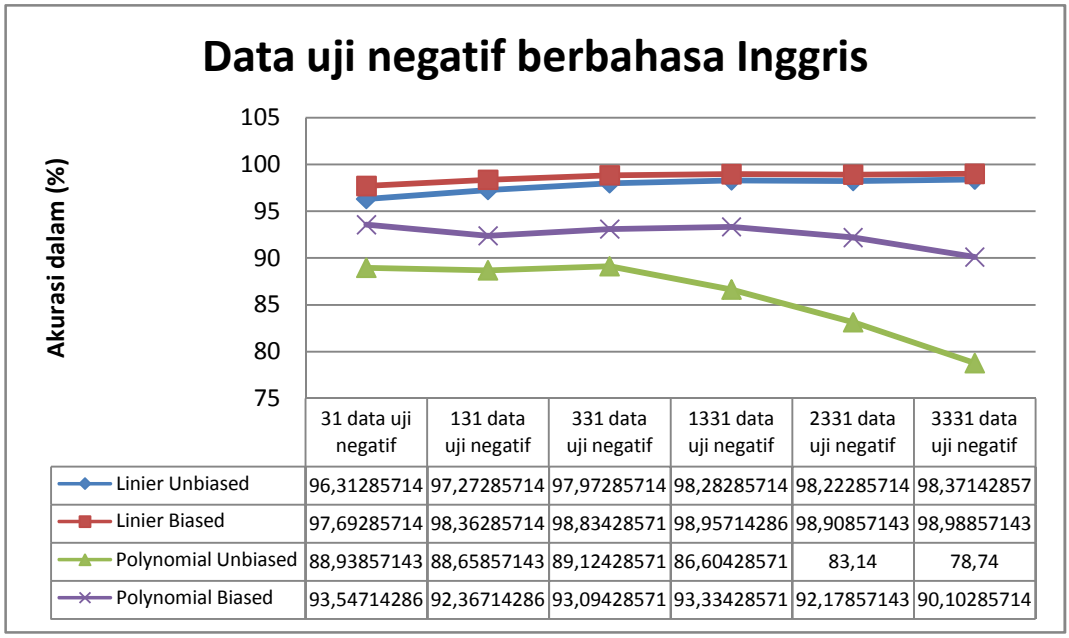
Dari hasil percobaan seperti ditunjukkan oleh tabel 4.7 sampai dengan tabel 4.14 untuk klasifikasi opini berbahasa Inggris dengan metode SVM diperoleh informasi bahwa metode SVM memberikan unjuk kerja yang baik. Unjuk kerja metode SVM secara umum menurun dengan berkurangnya jumlah data latih dan bertambahnya jumlah data uji. Pada hasil yang ditunjukkan oleh tabel 4.11 tidak menunjukkan penurunan dari segi persentase. Walau demikian hasil tetap menunjukkan peningkatan jumlah kesalahan klasifikasi seiring dengan bertambahnya jumlah data uji dan berkurangnya jumlah data latih. Hal ini relatif tidak berpengaruh disebabkan pola klasifikasi metode SVM mengacu pada ketersediaan *support vector* untuk membentuk *hyperplane*, jika terjadi kemiripan data latih maka ketersediaan *support vector* dianggap telah mewakili terbentuknya *hyperplane* walaupun dengan jumlah data latih yang lebih sedikit. Rangkuman

rata-rata hasil akurasi klasifikasi metode SVM dengan beberapa kernel ditunjukkan oleh gambar 4.7 dan 4.8.



	31 data uji positif	131 data uji positif	331 data uji positif	1331 data uji positif	2331 data uji positif	3331 data uji positif
Linier Unbiased	78,55428571	78,41571429	78,16142857	77,16142857	77,03	74,82142857
Linier Biased	79,72428571	80,15285714	78,50714286	77,85571429	77,99285714	76,13285714
Polynomial Unbiased	82,48857143	82,44285714	82,13142857	83,36285714	82,06285714	78,40857143
Polynomial Biased	81,56714286	80,69714286	79,67285714	79,00428571	80,21	76,56857143

Gambar 4.7
Grafik perbandingan unjuk kerja beberapa kernel pada data positif berbahasa Inggris

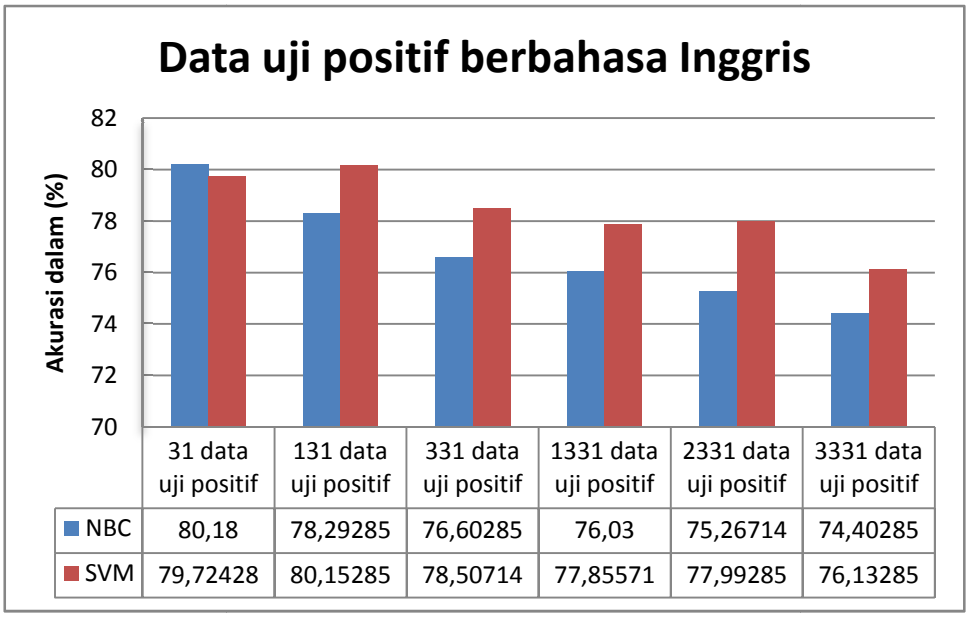


	31 data uji negatif	131 data uji negatif	331 data uji negatif	1331 data uji negatif	2331 data uji negatif	3331 data uji negatif
Linier Unbiased	96,31285714	97,27285714	97,97285714	98,28285714	98,22285714	98,37142857
Linier Biased	97,69285714	98,36285714	98,83428571	98,95714286	98,90857143	98,98857143
Polynomial Unbiased	88,93857143	88,65857143	89,12428571	86,60428571	83,14	78,74
Polynomial Biased	93,54714286	92,36714286	93,09428571	93,33428571	92,17857143	90,10285714

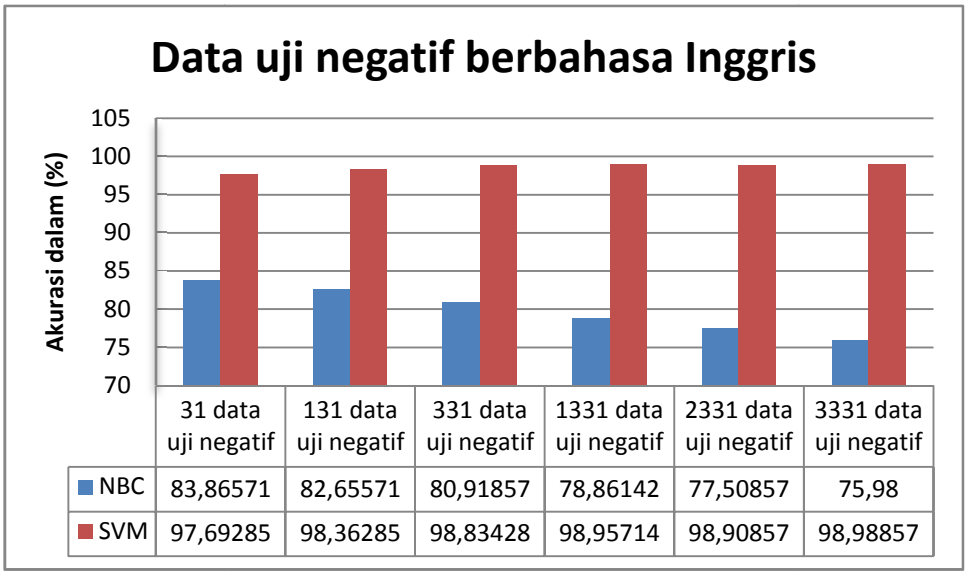
Gambar 4.8
Grafik perbandingan unjuk kerja beberapa kernel pada data negatif berbahasa Inggris

Grafik pada gambar 4.7 dan 4.8 menunjukkan untuk kernel linier *unbiased* dan linier bias memberikan hasil yang tidak jauh berbeda. Untuk kernel *polynomial unbiased* hasil cenderung ke arah opini positif sehingga akurasi untuk data uji positif lebih besar daripada kernel yang lain tetapi untuk data uji negatif lebih kecil daripada kernel yang lain. Hal ini disebabkan karena pada klasifikasi SVM dengan kernel *polynomial unbiased*, persamaan kuadrat melewati sistem koordinat asli dimana *hyperplane* akan bergeser yang berakibat bergeser pula kecenderungan hasil klasifikasi. Dari grafik juga kita dapatkan informasi bahwa data opini berbahasa Inggris cenderung terdistribusi secara linier didukung dengan hasil akurasi yang lebih baik untuk klasifikasi dengan kernel linier dibandingkan dengan kernel *polynomial*.

Perbandingan unjuk kerja metode SVM dan NBC ditunjukkan oleh gambar 4.9 dan 4.10.



Gambar 4.9
Grafik perbandingan unjuk kerja metode NBC dan SVM pada data positif berbahasa Inggris



Gambar 4.10
Grafik perbandingan unjuk kerja metode NBC dan SVM pada data negatif berbahasa Inggris

Dari grafik pada gambar 4.9 dan 4.10 kita dapatkan bahwa SVM memberikan unjuk kerja yang lebih baik dibandingkan metode NBC pada data berbahasa Inggris. Nilai akurasi metode SVM pada grafik tersebut adalah hasil percobaan menggunakan kernel liner bias yang merupakan hasil terbaik dari beberapa kernel yang diuji cobakan.

4.3.3 Hasil Percobaan dengan Metode NBC dan SVM untuk Data Berbahasa Indonesia.

Percobaan dilakukan dengan data yang sama antara metode SVM dan NBC. Untuk hasil percobaan dengan metode NBC ditunjukkan oleh tabel 4.15 dan tabel 4.16.

Tabel 4.15
Hasil NBC untuk data positif berbahasa Indonesia

NBC positif	data asli	data ke -1	data ke-2	data ke-3	data ke-4	data ke-5	data ke-6	**rata-rata**
30 data uji positif	100	73,34	70	76,67	73,34	63,34	63,34	**74,29**
100 data uji positif	79	69	70	64	65	59	74	**68,57143**
500 data uji positif	65,4	62,8	62,8	62,8	62,2	61,8	64,2	**63,14286**
1000 data uji positif	60,4	59,5	59,6	59,4	59	58,8	59,4	**59,44286**
2000 data uji positif	57,8	57,45	57,15	57,15	56,95	57,1	59,95	**57,65**
3000 data uji positif	60,13	60,13	59,76	59,46	59,16	59,43	63,8	**60,26714**

Tabel 4.16
Hasil NBC untuk data negatif berbahasa Indonesia

NBC negatif	data asli	data ke -1	data ke-2	data ke-3	data ke-4	data ke-5	data ke-6	**rata-rata**
30 data uji negatif	93,33	86,67	86,67	73,33	90	93,33	86,67	**87,14286**
100 data uji negatif	90	83	84	83	86	85	84	**85**
500 data uji negatif	85	86	86,4	86,8	88,6	88,8	76,4	**85,42857**
1000 data uji negatif	88,6	88,5	88,2	88	88,7	88	76,8	**86,68571**
2000 data uji negatif	85,5	85,25	84,7	84,8	84,6	84,25	72,45	**83,07857**
3000 data uji negatif	83,93	84,06	83,56	82,73	82,7	82,36	75,13	**82,06714**

Hasil yang ditunjukkan oleh tabel 4.15 dan tabel 4.16 menggambarkan bahwa metode NBC memberikan unjuk kerja yang cukup baik pada klasifikasi opini data berbahasa Indonesia. Nilai rata-rata pada tabel mengalami penurunan seiring dengan bertambahnya jumlah data uji yang disertai berkurangnya jumlah data latih. Hal ini menunjukkan bahwa proporsi jumlah data uji dan data latih mempengaruhi unjuk kerja dari metode NBC.

Hasil metode SVM ditunjukkan oleh tabel 4.17 sampai dengan tabel 4.24.

Tabel 4.17
Hasil SVM linier *unbiased* untuk data positif berbahasa Indonesia

Data uji positif dengan kernel linier *unbiased*	data asli	data ke -1	data ke-2	data ke-3	data ke-4	data ke-5	data ke-6	**rata-rata**
30 data uji positif	100	83,33	76,67	80	83,33	80	80	**83,33286**
100 data uji positif	83	77	80	81	85	82	86	**82**
500 data uji positif	78,8	76,2	77	77,6	77,4	78	81,8	**78,11429**
1000 data uji positif	76,4	75,7	75,8	75,9	76,4	76,7	80,2	**76,72857**
2000 data uji positif	75	75,3	74,75	75,1	75,15	74,5	74,8	**74,94286**
3000 data uji positif	77,87	77,3	76,43	76,27	76,3	76,47	62,27	**74,70143**

Tabel 4.18
Hasil SVM linier bias untuk data positif berbahasa Indonesia

Data uji positif dengan kernel linier bias	data asli	data ke -1	data ke-2	data ke-3	data ke-4	data ke-5	data ke-6	**rata-rata**
30 data uji positif	96,67	77,42	66,67	73,33	86,67	66,67	80	**78,20429**
100 data uji positif	78	72	73	74	76	71	86	**75,71429**
500 data uji positif	71,8	70	70,8	71,2	71,6	70,8	77	**71,88571**
1000 data uji positif	70,8	70,4	70,2	70	70	69,6	73,7	**70,67143**
2000 data uji positif	67,4	67,25	67,2	67,5	67,3	67,6	68,1	**67,47857**
3000 data uji positif	68,27	67,97	67	66,6	66,43	67,17	58,1	**65,93429**

Tabel 4.19
Hasil SVM *polynomial unbiased* untuk data positif berbahasa Indonesia

Data uji positif dengan kernel *polynomial unbiased*	data asli	data ke -1	data ke-2	data ke-3	data ke-4	data ke-5	data ke-6	**rata-rata**
30 data uji positif	100	93,33	93,33	96,67	93,33	86,67	96,67	**94,28571**
100 data uji positif	96	95	96	95	94	95	97	**95,42857**
500 data uji positif	94,8	93,8	93,8	94,2	93,6	93,8	93,8	**93,97143**
1000 data uji positif	92	91,9	92,2	91,9	91,6	91,2	92,6	**91,91429**
2000 data uji positif	93,9	93,5	93,6	93,6	93,3	93,55	94,35	**93,68571**
3000 data uji positif	96,77	96,53	96,3	96,2	95,93	95,83	94,13	**95,95571**

Tabel 4.20
Hasil SVM *polynomial* bias untuk data positif berbahasa Indonesia

Data uji positif dengan kernel *polynomial* bias	data asli	data ke -1	data ke-2	data ke-3	data ke-4	data ke-5	data ke-6	**rata-rata**
30 data uji positif	90	83,33	66,67	80	80	66,67	66,67	**76,19143**
100 data uji positif	80	75	74	74	75	69	82	**75,57143**
500 data uji positif	71,8	69,8	69	69,2	68,8	68,8	75	**70,34286**
1000 data uji positif	68,1	67,8	67,4	67,1	66,1	66,9	69,9	**67,61429**
2000 data uji positif	65,6	65,65	65,5	65,4	65,2	64,9	64,1	**65,19286**
3000 data uji positif	64,2	64,27	63,83	63,23	63,27	63,83	56,47	**62,72857**

Tabel 4.21
Hasil SVM linier *unbiased* untuk data negatif berbahasa Indonesia

Data uji negatif dengan kernel linier *unbiased*	data asli	data ke -1	data ke-2	data ke-3	data ke-4	data ke-5	data ke-6	**rata-rata**
30 data uji negatif	90	56,67	70	63,33	80	76,67	76,67	**73,33429**
100 data uji negatif	74	67	74	74	75	75	67	**72,28571**
500 data uji negatif	75,8	75,8	76,8	77,8	78,6	78,4	60,4	**74,8**
1000 data uji negatif	75,8	75,5	76	75,2	75,6	74,7	60,9	**73,38571**
2000 data uji negatif	71,1	70,5	70,8	70,55	70,4	75,1	61,2	**69,95**
3000 data uji negatif	62,07	61,43	60,93	60,3	59,93	58,57	53,23	**59,49429**

Tabel 4.22
Hasil SVM linier bias untuk data negatif berbahasa Indonesia

Data uji negatif dengan kernel linier bias	data asli	data ke -1	data ke-2	data ke-3	data ke-4	data ke-5	data ke-6	**rata-rata**
30 data uji negatif	86,67	56,67	76,67	66,67	80	80	80	**75,24**
100 data uji negatif	77	68	76	75	79	79	72	**75,14286**
500 data uji negatif	79,2	78,8	80,4	80	81,4	81	66,4	**78,17143**
1000 data uji negatif	79,8	79,1	80	79	79,9	78,7	66,2	**77,52857**
2000 data uji negatif	75,75	76,1	75,7	74,7	74,85	67,55	65,3	**72,85**
3000 data uji negatif	71,53	71,17	70,67	67,7	69,47	69,03	52,8	**67,48143**

Tabel 4.23
Hasil SVM *polynomial unbiased* untuk data negatif berbahasa Indonesia

Data uji negatif dengan kernel *polynomial unbiased*	data asli	data ke -1	data ke-2	data ke-3	data ke-4	data ke-5	data ke-6	**rata-rata**
30 data uji negatif	53,33	43,33	40	26,67	53,33	60	63,33	**48,57**
100 data uji negatif	44	35	43	45	49	48	44	**44**
500 data uji negatif	43,4	44,6	46	48	49,8	50,6	38	**45,77143**
1000 data uji negatif	44,5	44,9	45,3	46,2	47,6	48,1	37,6	**44,88571**
2000 data uji negatif	30,1	30,7	31,15	30,1	30,75	93,2	32,85	**39,83571**
3000 data uji negatif	20,03	20,2	21,37	21,93	21,97	27,3	27,3	**22,87143**

Tabel 4.24
Hasil SVM *polynomial* bias untuk data negatif berbahasa Indonesia

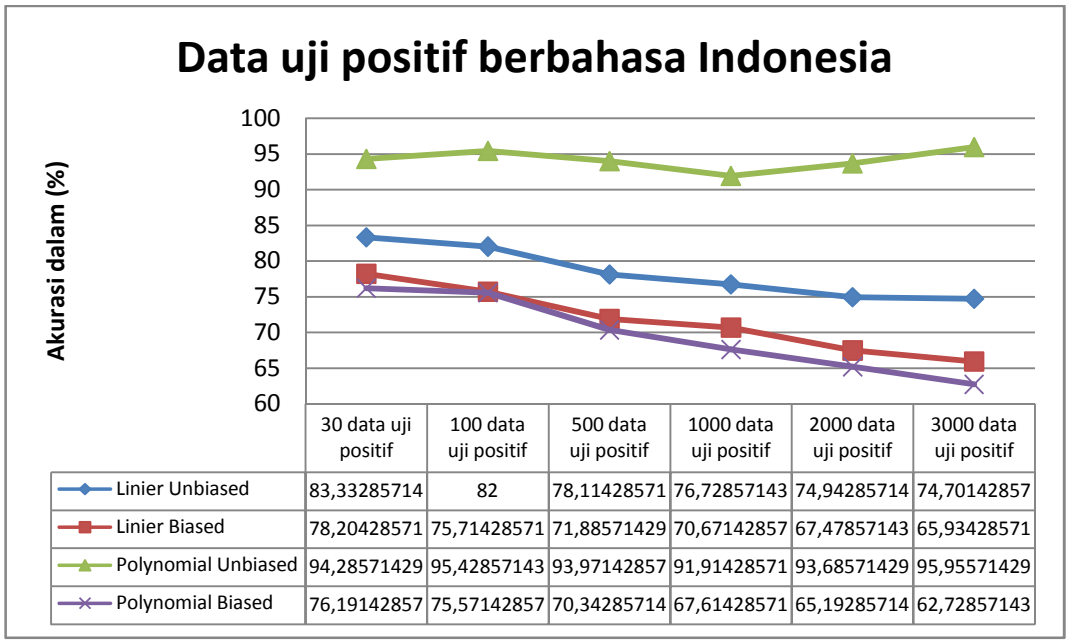
Data uji negatif dengan kernel *polynomial* bias	data asli	data ke -1	data ke-2	data ke-3	data ke-4	data ke-5	data ke-6	**rata-rata**
30 data uji negatif	90	63,33	76,67	70	76,67	86,67	73,33	**76,66714**
100 data uji negatif	78	71	76	77	78	79	73	**76**
500 data uji negatif	77,8	78,2	79,6	80,6	81,4	81,6	66,2	**77,91429**
1000 data uji negatif	80,3	80,3	80,4	80,1	79,6	79,3	67	**78,14286**
2000 data uji negatif	76,5	76,25	75,95	76,2	76	66	64,6	**73,07143**
3000 data uji negatif	75,03	75,27	74,87	74,17	74,13	74,07	58,23	**72,25286**

Unjuk kerja metode SVM dalam mengklasifikasikan data berbahasa Indonesia secara umum cukup baik ditunjukkan oleh nilai rata-rata pada tabel 4.17 sampai dengan tabel 4.18.

Sama halnya dengan yang terjadi pada uji coba dengan data berbahasa Inggris, pengurangan jumlah data latih yang diiringi bertambahnya data uji kurang berpengaruh dengan penurunan unjuk kerja metode SVM. Dalam uji coba dengan bahasa Indonesia didapatkan kecenderungan yang sama dengan uji coba

data berbahasa Inggris mengenai penggunaan kernel *polynomial unbiased.* Penggunaan kernel tersebut cenderung menggeser hasil klasifikasi kearah opini positif. Ditunjukkan oleh tabel 4.19 hasil akurasi dengan data uji opini positif memberikan hasil yang sangat baik, namun dari tabel 4.23 kita dapatkan unjuk kerja yang kurang baik untuk klasifikasi opini negatif.

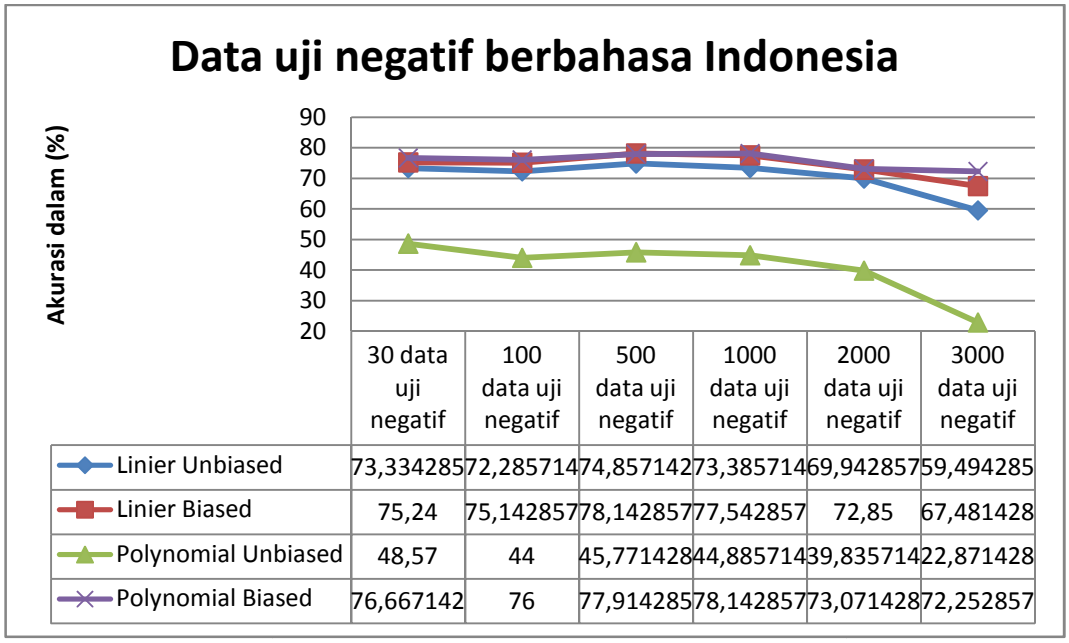
Rangkuman hasil klasifikasi data berbahasa Indonesia dengan metode SVM menggunakan beberapa variasi kernel ditunjukkan oleh gambar 4.11 dan 4.12



	30 data uji positif	100 data uji positif	500 data uji positif	1000 data uji positif	2000 data uji positif	3000 data uji positif
Linier Unbiased	83,33285714	82	78,11428571	76,72857143	74,94285714	74,70142857
Linier Biased	78,20428571	75,71428571	71,88571429	70,67142857	67,47857143	65,93428571
Polynomial Unbiased	94,28571429	95,42857143	93,97142857	91,91428571	93,68571429	95,95571429
Polynomial Biased	76,19142857	75,57142857	70,34285714	67,61428571	65,19285714	62,72857143

Gambar 4.11
Grafik perbandingan unjuk kerja beberapa kernel pada data positif berbahasa Indonesia

Data uji yang sedikit lebih besar dari data latih tidak menunjukkan penurunan unjuk kerja yang drastic. Hal ini disebabkan oleh telah terpenuhinya variasi kata dalam data latih yang berjumlah cukup besar.

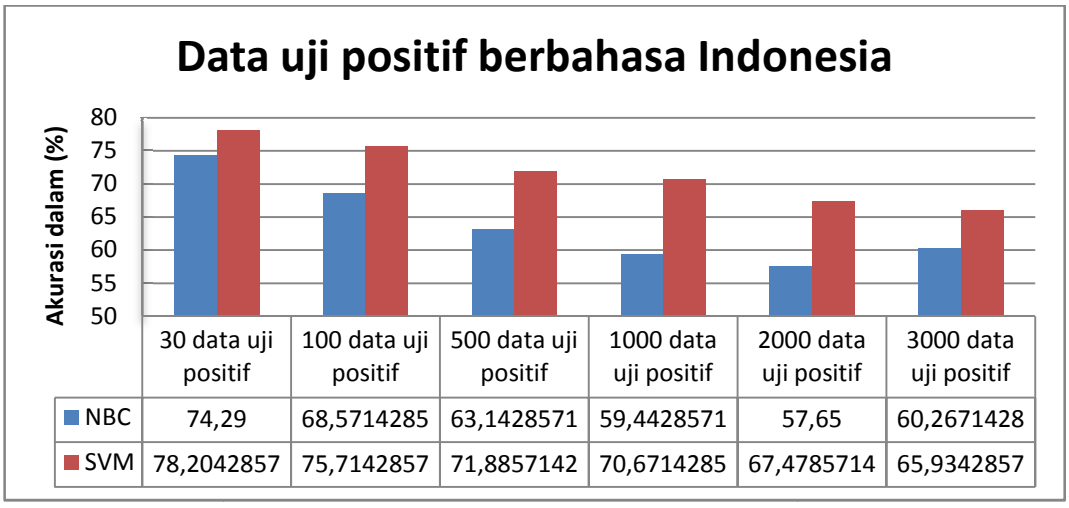


	30 data uji negatif	100 data uji negatif	500 data uji negatif	1000 data uji negatif	2000 data uji negatif	3000 data uji negatif
Linier Unbiased	73,33428	72,28571	74,85714	73,38571	69,94285	59,494285
Linier Biased	75,24	75,14285	78,14285	77,542857	72,85	67,481428
Polynomial Unbiased	48,57	44	45,77142	44,88571	39,83571	22,871428
Polynomial Biased	76,667142	76	77,91428	78,14285	73,07142	72,252857

Gambar 4.12
Grafik perbandingan unjuk kerja beberapa kernel pada data negatif berbahasa Indonesia

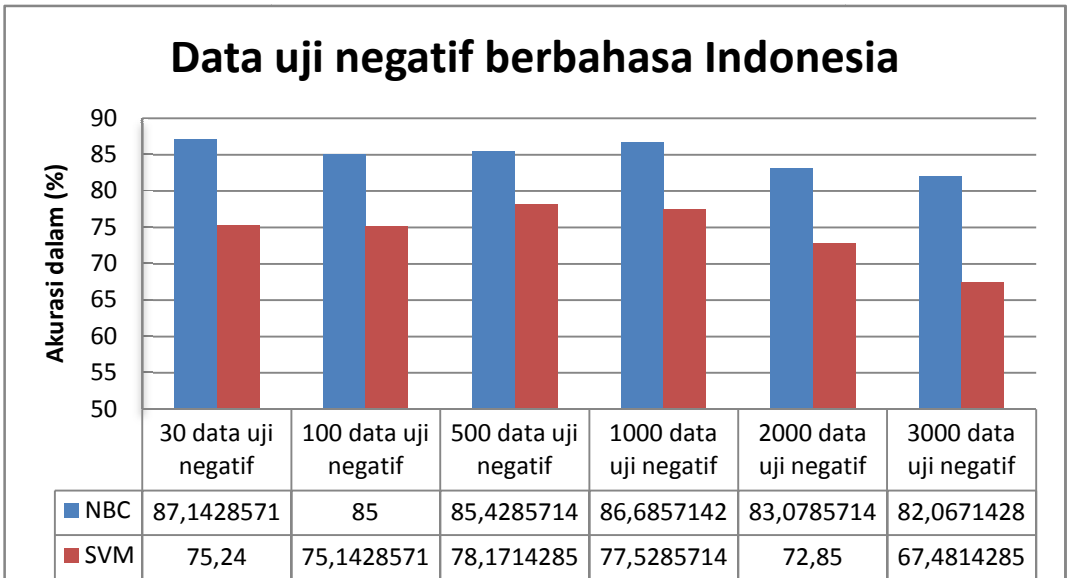
Gambar 4.11 dan 4.12 memberikan gambaran hasil yang lebih baik dengan menggunakan kernel linier, hal ini menunjukkan pola distribusi data opini berbahasa Indonesia cenderung terklasifikasi secara linier.

Perbandingan unjuk kerja metode SVM dan NBC pada data berbahasa Indonesia ditunjukkan oleh gambar 4.13 dan 4.14.



	30 data uji positif	100 data uji positif	500 data uji positif	1000 data uji positif	2000 data uji positif	3000 data uji positif
NBC	74,29	68,5714285	63,1428571	59,4428571	57,65	60,2671428
SVM	78,2042857	75,7142857	71,8857142	70,6714285	67,4785714	65,9342857

Gambar 4.13
Grafik perbandingan unjuk kerja metode NBC dan SVM pada data positif berbahasa Indonesia



	30 data uji negatif	100 data uji negatif	500 data uji negatif	1000 data uji negatif	2000 data uji negatif	3000 data uji negatif
NBC	87,1428571	85	85,4285714	86,6857142	83,0785714	82,0671428
SVM	75,24	75,1428571	78,1714285	77,5285714	72,85	67,4814285

Gambar 4.14
Grafik perbandingan unjuk kerja metode NBC dan SVM pada data negatif berbahasa Indonesia

Dari gambar 4.13 dan 4.14 kita dapatkan hasil percobaan pada teks bahasa Indonesia menunjukkan SVM memberikan akurasi yang lebih baik untuk data uji opini positif akan tetapi NBC menunjukkan akurasi yang lebih baik untuk data uji opini negatif. NBC memberikan hasil lebih baik disebabkan oleh beberapa faktor antara lain :

1. Beberapa kalimat data uji negatif berbahasa Indonesia pada penelitian menghasilkan probabilitas posteori sebagai opini negatif akan tetapi pemetaan ke ruang vektor berada pada kelas opini positif.
2. Metode pengolahan kata pada penelitian ini adalah *bag of words* sehingga kata bersifat independen tanpa memperhatikan kebergantungan satu dengan yang lainnya. Kelebihan dari metode NBC adalah memberikan performa yang baik untuk data yang independen (I. Rish).

3. Data opini berbahasa Indonesia pada penelitian ini cenderung terdistribusi linier dimana SVM memiliki keunggulan dalam mengklasifikasikan data yang terdistribusi *non* linier (Colas Fabrice & Brazdil Pavel).

4.3.4 Hasil Percobaan dengan Metode NBC dan SVM untuk Data Uji Paragraf

Dalam penelitian ini diujikan data teks panjang berupa kumpulan beberapa kalimat dalam sebuah paragraf. Hasil pengujian seperti ditunjukkan oleh tabel 4.25.

Tabel 4.25
Hasil Metode NBC dan SVM untuk Data Uji Paragraf

Data Uji	Kontekstual	NBC	SVM
"Joel Moore showing his range outside of his comedic work in Dodgeball An Underdog Story And the classy veteran actors CCH Pounder and Wes Studi who just simply do not work enough Avatar is one of the best films of the year The most exciting thrilling and superb work you will feast your eyes on in any theater this century Cinema forever will remember the benchmark that James Cameron placed not only for himself but for any man daring to change the game the way Cameron did Avatar is a movie experience to be remembered and please experience in a movie theater first"	positif	positif	Positif
"Forgive me I am going to jump from professional to fan boy for a while here I have not had the jitters after a film the way I have had for Avatar in quite sometimes James Cameron Avatar is the most entertaining and enthralling cinematic experiences of my life It is incredible simply put What Cameron has done here is the most passionate film project put out since Steven Spielberg released Schindlers List His attention to detail and his zeal for pushing the envelope is so admirable to any film maker or actor who will ever do another film from this point on"	positif	positif	Positif

“This is by far the weakest part of the movie the script For some odd reason they borrowed far too much from the novel whose pacing and themes far differ that of the Pirates brand and forgot that sometimes simplicity is best which is what made Curse of the Black Pearl such a great film It was the easiest to follow and On Stranger Tides did not learn from the previous two installments To add to that the script utterly separated everyone even those with the best onscreen chemistry Barbossa was barely with Jack Sparrow Sparrow was rarely with his ex lover and worst of all reliable Gibbs spent minimal time with Sparrow When they are together the humor the banter tension and the charm works well When they are not well the movie drags a bit”	negatif	negatif	Negatif
“Almost all the chase scenes or action sequences were done with very lowlighting and poor camera angles With the exception of the mesmerizing and chilling mermaid sequence and the opening chase all the action moments were missing that special touch While the bizarreness of Gore Verbinski will not be totally missed although his style worked perfectly in Rango his ability to crank out excellent stuntwork and fights was sorely missing here At least we got to see plenty of it from the opening chase to the final dramatic and short showdown Say whatever you want but there has yet to be anything that can top the infamous three way sword fight old mill showdown from Dead Man s Chest Bottom Line Pirates of the Caribbean On Stranger Tides is a mix of frustration and fun”	negatif	negatif	Negatif
“Di desa cemagi, kecamatan mengwi, hubungan harmonis antara masyarakat dan usaha pariwisata di kawasan setempat telah terjalin cukup lama. Salah satu program di tahun 2011 ini adalah program kursus bahasa inggris gratis kepada para pelajar di desa cemanggi. Kursus ini diikuti oleh pelajar SD. Kursus ini di gelar di kantor Perbekel Desa Cemagi. Respon positif dan antusias pelajar sangat terlihat meskipun saat itu akhir pekan. Para pelajar tampak bersemangat menyimak arahan dari para pembimbing berkompeten di bidangnya. Koordinator program ini mengatakan, para siswa SD tersebut	positif	positif	positif

merupakan sebagian dari 150 siswa SD, SMP, SMA yang mengikuti kursus. Penyelenggaraan kursus tahun ini memasuki tahun ke dua atau level ke dua."			
"Siswa-siswi kota denpasar kembali menunjukan keunggulannya dalam bidang pendidikan. Setelah siswa-siswi SMA berhasil meraih nilai UN tertinggi nasional, kini tingkat SMP, siswa-siswi denpasar kembali meraih nilai tertinggi nasional. Berdasarkan pengumuman hasil ujian nasional tingkat SMP yang diumumkan sabtu lalu, siswa SMPN 3 denpasar atas nama I Made Aditya Pramartha memperoleh nilai tertinggi 39,10. Sedangkan Putu Inda Pratiwi dari SMPN 10 dan I.B. Ari Sudewa, siswa SMP saraswati 1 denpasar, berhasil meraih peringkat ke-3 besar berdasarkan nilai akhir."	positif	positif	Positif
"Dari pemeriksaan inspektorat Kabupaten Bangli, ditemukan 50 orang pejabat yang didominasi kepala sekolah membuat rekomendasi bodong untuk meloloskan pegawai honorer disekolahnya. Akibat adanya pemeriksaan tersebut, mereka memilih menarik surat yang telah direkomendassikan sebelumnya. Hal ini disampaikan kepala inspektorat kabupaten bangle Drs. I Gede Suryawan, MM, rabu kemarin di bangli. Dari pemeriksaan inspektorat, kasek tersebut dengan polos mengakui surat yang dibuatnya itu adalah palsu agar bisa meloloskan pegawai pengabdian di sekolahnya dalam penjaringan CPNS Hal ini terkait dengan ancaman Bupati Made Gianyar bakal memasalahkan mereka yang tidak mau berkata jujur secara hukum dan administrasi. Suryawan mengatakan pihaknya telah melakukan verifkasi ke lapangan setelah menerima perintah bupati"	negatif	negatif	Negatif
"Maraknya Grab Boy yang berkedok menjadi penjual gelang lilit kulit di kawasan pariwisata kuta mulai meresahkan. Selain mengganggu ketertiban umum, tindakan mereka juga tergolong tindak kriminal yang dikhawatirkan mencoreng citra pariwisata. Demikian diungkapkan anggota DPRD Badung asal legian I Wayan Puspa Negara, minggu kemarin. Menurut Puspa, selain memaksa dan mengganggu wisatawan yang sedang menikmati liburannya, juga melakukan aksi	negatif	negatif	negatif

pencopetan. Modus operandinya anak-anak ini adalah menawakan gelang ikat kulit kepada wisman. Mereka mengerumuni wisatawan, khususnya wisatawan mancanegara, lalu diantara mereka ada yang memelas atau memaksa wisatawan untuk membeli. Di antara mereka juga ada yang membawa silet kemudian menorehkan tas wisman untuk kemudian mengambil barang-barang berharga.”			

Dari percobaan data paragraf seperti ditunjukkan oleh tabel 4.25 membuktikan bahwa metode NBC dan SVM mampu memberikan hasil yang baik dalam mengklasifikasikan lebih dari satu kalimat.

4.3.5 Contoh Kesalahan Hasil Klasifikasi

Contoh kesalahan pengenalan kalimat dalam klasifikasi dengan metode NBC adalah sebagai berikut :

Dipilih sebuah kalimat negatif berbahasa Indonesia sebagai data uji.

“Saya benar benar kecewa dengan promosi XL hanya dengan Rp 5 ribu bisa dapat 300 SMS gratis ke semua operator”;

Hasil program menunjukkan :

P(pos|kalimat) = 1.4241659268633E-74
P(neg|kalimat) = 5.8715681253752E-75
.....
Probabilitas priori kelas pos = 0.5
Probabilitas priori kelas neg = 0.5
.....
P(saya|positif) = 0.0048668780263425
P(saya|negatif) = 0.0045078135434754
P(benar|positif) = 0.00055889961074868
P(benar|negatif) = 0.00031006124902211
P(kecewa|positif) = 1.4838042763239E-5
P(kecewa|negatif) = 0.00014310519185636
P(dengan|positif) = 0.0036748885910289
P(dengan|negatif) = 0.0035919403155946
P(promosi|positif) = 3.4622099780892E-5
P(promosi|negatif) = 9.5403461237574E-6
P(xl|positif) = 4.9460142544131E-6
P(xl|negatif) = 9.5403461237574E-6
P(hanya|positif) = 0.00085071445175905
P(hanya|negatif) = 0.0010255872083039

P(rp|positif) = 0.00021267861293976
P(rp|negatif) = 0.00020511744166078
P(5|positif) = 0.00010386629934267
P(5|negatif) = 8.5863115113816E-5
P(ribu|positif) = 0.00012365035636033
P(ribu|negatif) = 8.1092942051938E-5
P(bisa|positif) = 0.0018745394024226
P(bisa|negatif) = 0.0013594993226354
P(dapat|positif) = 0.00054900758223985
P(dapat|negatif) = 0.00027667003758896
P(300|positif) = 4.9460142544131E-6
P(300|negatif) = 1.4310519185636E-5
P(sms|positif) = 2.4730071272065E-5
P(sms|negatif) = 8.1092942051938E-5
P(gratis|positif) = 0.00014838042763239
P(gratis|negatif) = 4.2931557556908E-5
P(ke|positif) = 0.0019932437445285
P(ke|negatif) = 0.00157415711042
P(semua|positif) = 0.00063803583881929
P(semua|negatif) = 0.00039115419107405
P(operator|positif) = 9.8920285088262E-6
P(operator|negatif) = 9.5403461237574E-6

Terlihat dari hasil bahwa ada beberapa kata yang menyumbangkan andil besar menghasilkan probabilitas positif untuk kalimat seperti kata “gratis” dan “promosi”. Sehingga kalimat teridentifikasi sebagai kalimat positif. Kesalahan juga terjadi untuk klasifikasi dengan metode SVM.

Dengan kalimat yang sama data vektor yang terbentuk adalah :

-1 1:3.7827080967546 18:5.8416017858082 24:2.0111243057189 33:5.910939500733 39:5.9119911136996 110:3.8654127470793 139:8.3759382134594 154:3.9554697503665 235:7.0051005180911 315:4.71035725253 339:13.2680127328 449:10.853128573059 1571:8.1895250892285 2583:8.4265642865294 3072:9.5334794904459 3190:10.426564286529 4559:8.5334794904459 8539:12.233919208587 10751:12.233919208587 15230:13.233919208587

Setelah dimasukkan ke dalam persamaan hyperplane score SVM memberikan hasil 0.326386 yang berarti kalimat dengan metode SVM diklasifikasikan sebagai opini positif.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

1. Metode NBC memberikan hasil dengan akurasi hingga 80,18% untuk data uji opini positif berbahasa Inggris, dan memberikan hasil dengan akurasi hingga 83,86% untuk data uji opini negatif berbahasa Inggris. Untuk data berbahasa Indonesia metode NBC memberikan hasil dengan akurasi hingga 74,29% pada data uji opini positif dan hingga 87,14% pada data uji opini negatif.
2. Metode SVM memberikan hasil dengan akurasi hingga 80,15% untuk data uji opini positif berbahasa Inggris, dan memberikan hasil dengan akurasi hingga 98,95% untuk data uji opini negatif berbahasa Inggris. Untuk data berbahasa Indonesia metode SVM memberikan hasil dengan akurasi hingga 78,20% pada data uji opini positif dan hingga 78,14% pada data uji opini negatif.
3. Metode SVM memberikan unjuk kerja yang lebih baik daripada metode NBC untuk mengklasifikasikan opini berbahasa Inggris dan opini positif berbahasa Indonesia. Sedangkan NBC memberikan unjuk kerja yang lebih baik dalam mengklasifikasikan data uji opini negatif berbahasa Indonesia.
4. Metode NBC dan SVM memberikan hasil yang tepat dalam mengklasifikasikan opini dalam bentuk paragraf yang terdiri dari beberapa kalimat.

5.2 Saran

Pada penelitian ini kalimat yang akan diklasifikasikan dipandang sebagai *bag of words* atau sekumpulan kata-kata. Faktor yang berpengaruh adalah frekuensi kemunculan kata pada kalimat tersebut. Kedepannya diharapkan dapat diteliti pengklasifikasian kalimat yang juga memperhitungkan faktor susunan kata-kata yang dapat dipisahkan dalam *subject – predicate – object* serta penanganan frase sehingga membentuk sebuah *sentence processor*.

Metode SVM pada penelitian ini adalah pengklasifikasian biner yang hanya menghasilkan dua kelas. Selanjutnya dapat diteliti bagaimana implementasi dan unjuk kerja metode SVM untuk pengklasifikasian teks *multiclass*.

DAFTAR PUSTAKA

Anonym. 2010. *Naïve Bayes Classifier*. [Online]. Tersedia di: http://en.wikipedia.org/wiki/Naive_Bayes_classifier. [diunduh : 8 Nov 2010].

Anonym. 2010. *Sentiment Analysis*. [Online]. Tersedia di: http://en.wikipedia.org/wiki/Sentiment_Analysis. [diunduh : 8 Nov 2010].

Anonym. 2010. *Support Vector Machine* [Online]. Tersedia di:http://en.wikipedia.org/wiki/Support_vector_machine. [diunduh : 8 Nov 2010].

Anonym. 2010. *Text Mining*. [Online]. Tersedia di: http://en.wikipedia.org/wiki/Text_mining. [diunduh : 8 Nov 2010].

Barber, I. 2010. *Bayesian Opinion Mining*. [Online]. Tersedia di: http://phpir.com/bayesian-opinion-mining [diunduh: 10 Nov 2010].

Barber, I. 2009. *Simple Search : The Vector Space Model*. [Online]. Tersedia di: http:// phpir.com/simple-search-the-vector-space-model [diunduh: 10 Nov 2010].

Barber, I. 2009. *Support Vector Machines In PHP*. [Online]. Tersedia di: http:// phpir.com/support-vector-machines-in-php [diunduh: 10 Nov 2010].

Biu, L. 2010. *Sentiment Analysis: A Multi-Faceted Problem*. [Online]. Tersedia di:

Blitzer, J., Dredze, M. & Pereira, F. 2006. *Biographies, Bollywood, Boom-boxes and Blenders: Domain Adaptation for Sentiment Classification*. [Online]. Tersedia di: *www.cs.jhu.edu/~mdredze/publications/sentiment_acl07.pdf.* [diunduh : 1 Februari 2011]

Bridge, C. 2011. *Unstructured Data and the 80 Percent Rule*. [Online]. Tersedia di: http://www.clarabridge.com/default.aspx?tabid=137&ModuleID=635&ArticleID=551 [diunduh : 5 Nov 2010].

Caruana, R. & Niculescu-Mizil, A. 2006. *An empirical comparison of supervised learning algorithms. Proceedings of the 23rd international conference on Machine learning, 2006*. [Online]. Tersedia di: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.122.5901&rep=rep1&type=pdf

Colas, F. & Brazdil, P. 2005. *Comparison of SVM and Some Older Classification Algorithms in Text Classification Tasks.*

Cortes, C. & Vapnik, V. 1995. *Support-Vector Networks. Machine Learning, 20*. [Online]. Tersedia di: http://www.springerlink.com/content/k238jx04hm87j80g/. [diunduh : 8 Nov 2010].

Coussement, K. & Poel, V. D. 2008. *Integrating the Voice of Customers through Call Center Emails into a Decision Support System for Churn Prediction*. [Online]. Tersedia di: http://www.textmining.ugent.be/ [diunduh : 5 Nov 2010].

Dehaff, M. 2010. *Sentiment Analysis, Hard But Worth It!*. [Online]. Tersedia di: http://www.customerthink.com/blog/sentiment_analysis_hard_but_worth_it [diunduh : 5 Nov 2010].

Jason D. M. Rennie & Ryan Rifkin. *Improving Multiclass Text Classification with the Support Vector Machine.*[Online]. Tersedia di: http://www.ai.mit.edu

Jenkins, M. C. 2011. *How Sentiment Analysis works in machines*. [Kuliah]. Tersedia di: http://www.slideshare.net/mcjenkins/how-sentiment-analysis-works/download [diunduh : 8 Februari 2011].

Mihalcea, R. , Banea, C. & Wiebe, J. 2007. *Learning Multilingual Subjective Language via Cross-Lingual Projections*. *Proceedings of the Association for Computational Linguistics (ACL).* pp. 976–983. [Online]. Tersedia di: http://www.cse.unt.edu/~rada/papers/mihalcea.acl07.pdf.

Pang, B. & Lee, L. 2004. *A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts*. *Proceedings of the Association for Computational Linguistics (ACL).* pp. 271–278. [Online]. Tersedia di: http://www.cs.cornell.edu/home/llee/papers/cutsent.home.html.

Pang, B. & Lee, L. 2005. *Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales*. *Proceedings of the Association for Computational Linguistics (ACL).* pp. 115–124. [Online]. Tersedia di: http://www.cs.cornell.edu/home/llee/papers/pang-lee-stars.home.html [diunduh : 8 Februari 2011].

Pang, B. & Lee, L. 2008. *Subjectivity Detection and Opinion Identification*. *Opinion Mining and Sentiment Analysis*. Now Publishers Inc. [Online]. Tersedia di: http://www.cs.cornell.edu/home/llee/opinion-mining-sentiment-analysis-survey.html.

Snyder B. & Barzilay R. 2007. *Multiple Aspect Ranking using the Good Grief Algorithm*. *Proceedings of the Joint Human Language Technology/North American Chapter of the ACL Conference (HLT-NAACL).* pp. 300–307. [online]. Tersedia di: http://people.csail.mit.edu/regina/my_papers/ggranker.ps.

Su, F. & Markert, K. 2008. From Words to Senses: a Case Study in Subjectivity Recognition. *Proceedings of Coling 2008, Manchester, UK*. [Online]. Tersedia di: http://www.comp.leeds.ac.uk/markert/Papers/Coling2008.pdf.

Tan, P. N., Steinbach, M. & Kumar, V. 2006. *Introduction to Data Mining*. Boston : Pearson Addison Wesley.

Wulandini, F. & Nugroho, A. N. 2009. *Text Classification Using Support Vector Machine for Webmining Based Spation Temporal Analysis of the Spread of Tropical Diseases. International Conference on Rural Information and Communication Technology 2009*. [Online] 189-192. Tersedia di: http://asnugroho.net/papers/rict2009_textclassification.pdf [diunduh:5 Nov 2010].

Wibisono, Y. 2005. Klasifikasi Berita Berbahasa Indonesia menggunakan *Naïve Bayes Classifier*. [Online]. Tersedia di: http:// *fpmipa.upi.edu/staff/yudi/yudi_0805.pdf [diunduh: 1 Nov 2010]*.

Wibisono, Y. 2005. *Clustering Berita Berbahasa Indonesia. [Online]*. *Tersedia di: http:// fpmipa.upi.edu/staff/yudi/KNSI_Clustering_yudi_masayu.pdf [diunduh : 1 Nov 2010]*.

Zhang, H. 2004. *The Optimality of Naive Bayes*. FLAIRS2004 conference. [Online]. Tersedia di: http://www.cs.unb.ca/profs/hzhang/publications/FLAIRS04ZhangH.pdf.