Peringkasan dan Support Vector Machine pada Klasifikasi Dokumen

Article · November 2017

DOI: 10.20895/infotel.v9i4.312

3 authors:

Nelly Indriani Widiastuti
Universitas Komputer Indonesia
3 PUBLICATIONS **3** CITATIONS
SEE PROFILE

Ednawati Rainarli
Universitas Komputer Indonesia
6 PUBLICATIONS **1** CITATION
SEE PROFILE

Kania Dewi
Universitas Komputer Indonesia
4 PUBLICATIONS **1** CITATION
SEE PROFILE

JURNAL INFOTEL
Informatika - Telekomunikasi - Elektronika

Website Jurnal : http://ejournal.st3telkom.ac.id/index.php/infotel
ISSN : 2085-3688; e-ISSN : 2460-0997

Peringkasan dan Support *Vector Machine* pada Klasifikasi Dokumen

Nelly Indriani[1], Ednawati Rainarli[2], Kania Evita Dewi[3]
[1,2,3]Informatika, Teknik dan Ilmu Komputer, UNIKOM
[1,2,3]Jl. Dipati Ukur no. 112-116, 40132, Bandung, Indonesia
Email korespondensi : nelly.indriani@email.unikom.ac.id

Dikirim 08 Oktober 2017, Direvisi 29 Oktober 2017, Diterima 09 November 2017

Abstrak – Klasifikasi adalah proses pengelompokkan objek yang memiliki karakteristik atau ciri yang sama ke dalam beberapa kelas. Klasifikasi dokumen secara otomatis dapat dilakukan dengan menggunakan ciri atau fitur kata yang muncul pada dokumen latih. Jumlah dokumen yang besar dan banyak mengakibatkan jumlah kata yang muncul sebagai fitur akan bertambah. Oleh karena itu, peringkasan dipilih untuk mereduksi jumlah kata yang digunakan dalam proses klasifikasi. Untuk proses klasifikasi digunakan metode *Support Vector Machine* (SVM) untuk multikelas. SVM dipilih karena dianggap memiliki reputasi yang baik dalam klasifikasi. Penelitian ini menguji penggunaan ringkasan sebagai seleksi fitur dalam klasifikasi dokumen. Peringkasan menggunakan kompresi 50 %. Hasil yang diperoleh menunjukkan bahwa proses peringkasan tidak mempengaruhi nilai akurasi dari klasifikasi dokumen yang menggunakan SVM.Akan tetapi, penggunaan peringkasan berpengaruh pada peningkatan hasil akurasi dari metode klasifikasi*Simple Logistic Classifier* (SLC). Hasil pengujian metode klasifikasi menunjukkan bahwa penggunaan metode *Naïve Bayes Multinomial* (NBM) menghasilkan akurasi yang lebih baik dari pada metode SVM.

Kata kunci – peringkasan, klasifikasi, SVM, seleksi fitur, dokumen

Abstract - Classification is the process of grouping objects that have the same features or characteristics into several classes. The automatic documents classification use words frequency that appears on training data as features.The large number of documents cause the number of words that appears as a feature will increase.Therefore, summaries are chosen to reduce the number of words that used in classification.The classification uses multiclass Support Vector Machine (SVM) method.SVM was considered to have a good reputation in the classification. This research tests the effect of summary as selection features into documents classification.The summariesreduce text into 50%.A result obtained that the summaries did not affect value accuracy of classification of documents that use SVM.But, summaries improve the accuracy of Simple Logistic Classifier. The classification testing shows that the accuracy of Naïve Bayes Multinomial (NBM) better than SVM.

Keywords – summarization, clasification, SVM, features selection, documents

I. PENDAHULUAN

Klasifikasi adalah proses pengelompokkan objek yang memiliki karakteristik atau ciri yang sama ke dalam beberapa kelas. Pada umumnya klasifikasi dokumen dilakukan dengan menentukan ciri-ciri atau fitur-fitur yang diwakili oleh kalimat-kalimat penting. Dalam dokumen yang berukuran besar, klasifikasi akan menjadi tantangan sistem. Jumlah kata yang menyusun kalimat meningkat secara eksponensial[1]. Diperlukan teknik yang dapat mereduksi beban komputasi secara signifikan. Peringkasan dapat dianggap sebagai cara untuk memilih fitur sekaligus mengurangi beban komputasi.

Beberapa penelitian telah melakukan peringkasan sebagai fitur ekstraksi untuk klasifikasi dokumen. Aguinano-Hernandez dkk. melakukan pengujian pada beberapa ukuran data set. Hal tersebut menunjukkan bahwa peringkasan adalah pendekatan yang kompetitif untuk pemilihan fitur dibandingkan dengan teknik Informasi Gain yang biasa dilakukan[2]. Penelitian serupa juga dilakukan pada bahasa lain yaitu oleh Eman Al-thwaib. Penelitian ini mengklasifikasikan dokumen berbahasa Arab ke dalam bidang ekonomi,

agama, olahraga, dan politik. Hasilnya menunjukkan bahwa peringkasan memberikan nilai akurasi, *precision* dan *recall* yang lebih tinggi tetapi menghasilkan waktu komputasi yang lebih lama[3]. Berbeda dengan penelitian *opinion mining* yang dilakukan oleh Savita Harer dkk. Mereka mengklasifikasi sentimen terhadap *review* film. Sistem yang dibuat berbasis *mobile*,menggunakan LSA untuk peringkasan dan *Random Forest* untuk klasifikasi. Penelitian ini menghasilkan klasifikasi yang baik[4]. Klasifikasi sentimen terhadap konten dalam *web* yang mengulas tentang *handphone* juga telah dilakukan. Jumlah data yang besar dan struktur halaman situs yang berbeda-beda menyebabkan mereka perlu meringkas informasi atau data berkaitan[5]. Hyoungil Jeong telah menunjukkan bahwa peringkasan dan klasifikasi dapat saling membantu kinerja sistem. Mereka mengajukan sebuah *framework* yang efektif untuk kerjasama antara peringkasan dan klasifikasi sehingga menghasilkan alat analisa teks yang lebih baik [6].

Dalam penelitian ini, peringkasan digunakan sebagai seleksi fitur sebelum dilakukan klasifikasi dokumen berbahasa Indonesia. Dokumen yang digunakan merupakan latar belakang sebuah karya ilmiah. Dokumen tersebut dapat dikatakan cukup besar sehingga kebutuhan untuk mereduksi jumlah fitur klasifikasi menjadi tinggi.Metode peringkasan yang digunakan adalah skema pembobotan berbasis kata*Term frequency-Inverse document frequency*atau Tf-Idf[7]. Setelah melalui proses peringkasan, dokumen akan diklasifikasi menggunakan metode yang cukup terkenal yaitu *Support Vector Machines*.

A. Tf-Idf

Term frequency-Inverse document frequency atau lebih dikenal sebagai Tf-Idf digunakan untuk mengekstrak kalimat dengan cara memberikan nilai atau bobot pada kalimat. Ekstraksi kalimat adalah cara untuk mengkomputasi suatu kalimat sehingga dapat ditentukan nilai penting atau tidak pentingnya suatu kalimat[8][9]. Metode ini sering digunakan sebagai faktor pembobotan dalam *information retrieval*, *text mining*, dan *recommendation system.* Nilai Tf-Idf menentukan besar bobot kalimat. Penentuan nilai bobot dilakukan dengan cara menghitung frekuensi kemunculan kata dalam dokumen.

Joeran Beel dkk. melakukan survei terhadap sejumlah artikel yang melakukan penelitian tentang sistem rekomendasi, seperti rekomendasi film, *gadget*, dan lain-lain. Sistem tersebut menentukan rekomendasi berdasarkan opini. Sejumlah penelitian tersebut menggunakan TF-Idf sebagai cara untuk mengekstraksi kalimat atau opini yang terkumpul. Dari 200 artikel, sebanyak 70% menggunakan TF-Idf untuk menentukan bobot suatu kalimat[10].

Dalam Tf-Idf, hasil pra proses yang berupa token dan telah difilter sesuai kebutuhan. Token tersebut dihitung frekuensinya dalam kalimat dan dokumen. Gambar 3 adalah proses perhitungan Tf-Idf. Rumus (1) menghitung Tf yang mana $f_{t,d}$ adalah frekuensi kata t dalam dokumen d.

$$Tf = f_{t,d} \quad (1)$$

Rumus (2) untuk menghitung Idf yang digunakan

$$IDF = \log\left(\frac{N}{n_i}\right) \quad (2)$$

dimana N mewakili jumlah kalimat yang digunakan, sedangkan untuk n_i untuk mewakili jumlah kalimat yang mana kata ke i muncul.

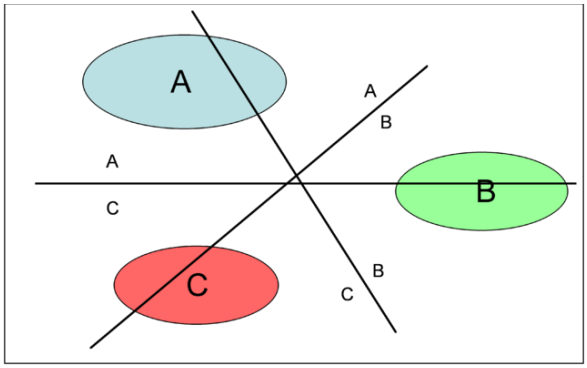
B. Support Vector Machines

Metode klasifikasi SVM adalah salah satu metode diskriminatif yang paling tepat yang digunakan dalam klasifikasi. Metode ini bekerja berdasarkan pada *Structural Risk Minimization*, yang merupakan prinsip induktif penggunaan dalam pembelajaran mesin[11]. Menurut Auria, beberapa kelebihan dari metode SVM adalah metode ini bekerja baik untuk sekumpulan data yang tidak dapat dipisahkan secara linear. Pada penggunaan kernel *Gaussian*, pemilihan nilai parameter C dan r yang tepat dapat membuat metode SVM bekerja dengan baik walaupun data yang dilatih memiliki nilai bias. Parameter yang diperoleh dari hasil pelatihan dengan metode SVM dijamin adalah parameter optimal. Hal ini berbeda jika dibandingkan dengan metode *Neural Network* dimana bisa terjadi solusi yang diperoleh terjebak dalam minimum lokal [12].

Metode *Support Vector Machine* (SVM) banyak digunakan untuk melakukan klasifikasi otomatis. Beberapa penelitian telah menggunakan SVM untuk berbagai penerapan, diantaranya adalah pada pengenalan citra, analisis medik, ataupun untuk melakukan prediksi. Secara spesifik, Wang merangkum beberapa penelitian yang berkaitan dengan perkembangan SVM beserta penggunaannya [12]. Dalam beberapa penelitian ditunjukkan bahwa SVM adalah metode yang efisien [13][14][15]. Mathias Ring dkk. menggunakan Kernel *Gaussian* RBF untuk SVM sehingga menghasilkan waktu proses yang lebih baik tanpa kehilangan akurasi [16]. Bissan Ghaddar dan Joe Naoum-Sawaya menguji SVM terhadap kasus sentimen film dan klasifikasi penyakit. Dengan jumlah fitur yang sangat tinggi, SVM telah menunjukkan hasil yang baik. Pendekatan klasifikasi dan seleksi fitur yang diajukan, sederhana, dan alur yang dapat ditelusuri, dan mencapai rata-rata *error* yang rendah [17]. Dalam penelitian yang dilakukan oleh Xuchan Ju dkk. SVM ditingkatkan kemampuannya dengan memodifikasi *nonparallel hyperplanes* untuk *multiclass classifications*. Hasilnya menunjukkan efisiensi dan akurasi yang baik [18]. Gambar 1 memperlihatkan SVM *multiclass* satu ke satu [19].

Metode SVM dipilih karena SVM termasuk metode yang populer selain KNN dalam mengklasifikasikan dokumen [20]. Umumnya SVM membagi ruang vektor menjadi 2 yaitu kelas positif

dan kelas negatif. Hal tersebut tidak menutup kemungkinan untuk menggunakan SVM untuk keperluan membagi menjadi lebih dari 2 kelas. Pada penelitian yang dilakukan oleh Chih-Wei Hsu and Chih-Jen Lin menunjukkan SVM multi kelas lebih baik dibandingkan dengan metode pembanding lainnya [21]. Berdasarkan permasalahan yang telah disampaikan, maka tujuan penelitian ini adalah untuk mengukur kinerja peringkasan bila digunakan dalam klasifikasi dokumen berbahasa Indonesia dan menguji hasil klasifikasi dokumen dengan metode SVM multi kelas pada data set yang digunakan.



Gambar 1. SVM Multi Class

II. METODE PENELITIAN

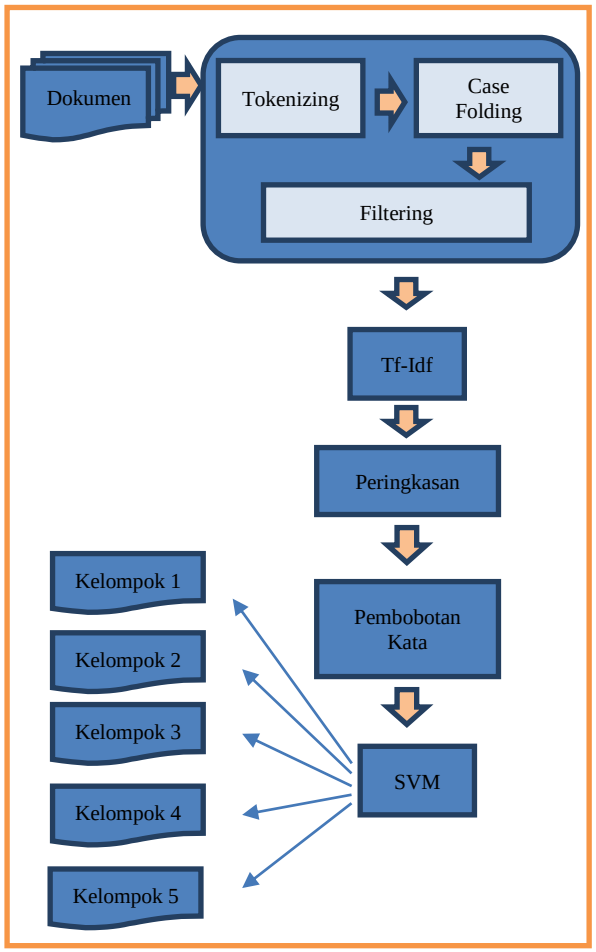
Metode penelitian yang digunakan analisis deskriptif. Dalam paper ini dideskripsikan tahap-tahap menuju klasifikasi dokumen. Seluruh dokumen melalui tahap pra proses, pembobotan kata, peringkasan, dan klasifikasi.

Dalam penelitian ini setiap dokumen melalui seluruh tahapan yang digambarkan pada Gambar 2. Pertama dokumen melalui tahap pra proses. Tahapan pra proses terbagi menjadi tiga bagian yaitu *tokenizing*, *case folding* dan *filtering*. Tahap berikutnya setiap kalimat dalam dokumen diekstraksi menggunakan metode *Term frequency-Inverse document frequency* (Tf-Idf). Tf-Idf dipilih berdasarkan hasil penelitian yang telah dilakukan sebelumnya. Penelitian Kania, dkk. membandingkan ekstraksi kalimat atau pembobotan kata Tf-Idf dengan skema LGN[7]. Hasil tahap ekstraksi kalimat adalah bobot setiap kalimat dalam dokumen yang kemudian dilakukan pemeringkatan dari yang terbesar hingga yang terkecil. Dengan menggunakan kompresi 50%, maka diperoleh hasil ringkasan. Setelah melalui proses peringkasan, dokumen diklasifikasi menggunakan *Support Vector Machine* (SVM). Sebelum diklasifikasi, hasil ringkasan kembali melalui tahap pra proses dan Tf-Idf untuk memperoleh nilai vektor. Nilai vektor tersebut akan menjadi masukan ke dalam proses klasifikasi menggunakan SVM. Gambar 2 adalah metode penelitian yang dilakukan. Dokumen sudah dikonversi dari format pdf menjadi format txt. Konversi dilakukan di luar sistem.

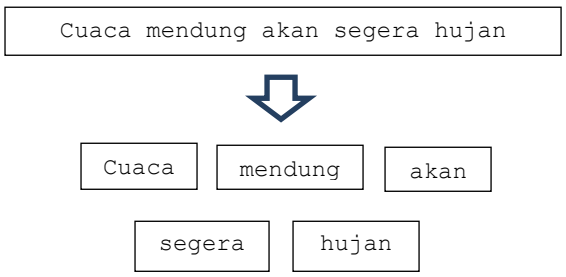
A. *Pra Proses*

Tahapan pra proses yang digunakan adalah *tokenizing, case folding* dan *filtering.* Secara umum *tokenizing* adalah memecah kalimat menjadi satuan terkecilnya yaitu kata, huruf, atau simbol. Gambar 3 adalah ilustrasi *tokenizing*.

Setelah setiap kalimat diproses oleh *tokenizer*, selanjutnya adalah mengubah setiap huruf kapital menjadi huruf kecil. Proses terakhir adalah memfilter hasil *case folding*. Dalam penelitian ini *filtering* dilakukan terhadap simbol-simbol dan angka.



Gambar 2. Metode Penelitian

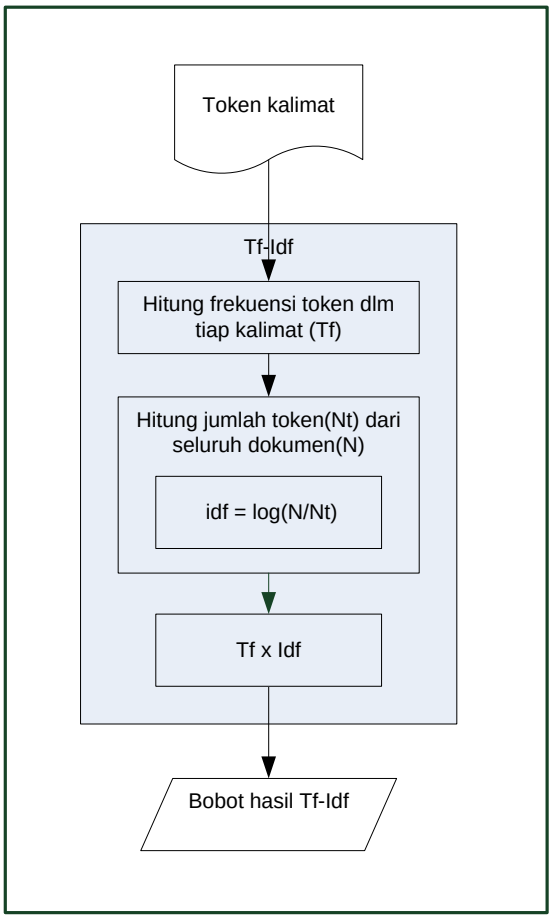


Gambar 3. Ilustrasi Tokenizing

B. Ekstraksi Kalimat

Metode ekstraksi kalimat yang digunakan dalam penelitian ini adalah metode *Term frequency-Inverse document frequency* (Tf-Idf). Dalam penelitian yang dilakukan oleh Kania dkk. menunjukkan bahwa Tf-Idf lebih baik daripada skema LGN meskipun tidak terlalu signifikan [7]. Pada Gambar 4 menunjukkan tahap penghitungan Tf-Idf.

Dalam paper ini dicontohkan dokumen yang memiliki 8 buah kalimat, diantaranya terdapat kalimat "rencana warga bukit duri dibatalkan". Setelah melalui tahap pra proses, kemudian dihitung frekuensi kata terhadap kalimat dan dokumen. Berdasarkan proses tersebut menghasilkan nilai Tf dan IDF kedua nilai tersebut dikalikan untuk memperoleh nilai bobot $Tf - Idf$.



Gambar 4. Ilustrasi Tf-Idf

Tabel 1 menunjukkan nilai n_i = 1 untuk i = 1 berarti kata "Rencana"hanya terdapat 1 dalam seluruh dokumen sedangkan kata "Warga" berjumlah 4 dalam seluruh dokumen dan seterusnya. Nilai N adalah jumlah kalimat dalam dokumen. Pada tabel 1 dokumen yang digunakan memiliki jumlah kalimat 8. Nilai N/$\boldsymbol{n_i}$untuk kata "Rencana" = 8, kata "Warga"= 8/4 = 2 dan seterusnya. Kolom Idf merupakanhasil $\log\left(\frac{N}{n_i}\right)$.

Tabel 1. Contoh Nilai Idf

Kata	n_i	N/n_i	Idf
Rencana	1	8	0.903
Warga	4	2	0.301
Bukit	3	2.67	0.426
Duri	3	2.67	0.426
Dibatalkan	1	8	0,903

Nilai Tf-Idf yang diperoleh dengan mengalikan nilai Tf dengan nilai Idf, sehingga dapat dilihat pada tabel 2. Jumlah nilai Tf-Idf setiap kalimat diurutkan dari yang terbesar hingga terkecil. Dengan mengambil 50% dari jumlah seluruh kalimat dalam dokumen, maka jumlah dokumen yang menjadi ringkasan akan berjumlah 4 kalimat. Jadi berdasarkan Tabel 2, maka yang menjadi hasil ringkasan adalah {S6, S1, S7, S3}.

Tabel 2. Contoh Hasil Perangkingan

kalimat	TF-IDF	Ranking
S1	10.91	2
S2	5.316	8
S3	9.155	4
S4	9.145	5
S5	8.678	6
S6	15.653	1
S7	10.257	3
S8	5.969	7

Kalimat-kalimat yang terpilih sebagai ringkasan kembali melalui tahap pra proses dan ekstraksi kalimat menggunakan Tf-Idf. Hasil tahap ini adalah vektor dokumen yang menjadi masukan ke tahap selanjutnya.

C. Klasifikasi Dokumen

Dokumen yang telah diringkas diklasifikasi menggunakan *Support Vector Machine* (SVM). SVM adalah metode klasifikasi yang membagi ruang vektor menjadi 2 bagian yaitu kelas positif dan kelas negatif oleh hyperplan. Dalam penelitian ini dokumen akan dibagi menjadi 5 kelas. Berdasarkan hal tersebut maka digunakan *multiclass* SVM.

Pada *multiclass* SVM terdapat 2 cara yaitu *one-against-one* dan *one-against-all*. Dalam penelitian ini dokumen diklasifikasi menjadi 5 kelompok menggunakan Weka. Dalam Weka, libsvm mengklasifikasi *multiclass* dengan cara 1 kelas terhadap 1 kelas yang lain yang disebut *one-against-one*. Setiap kelas akan dibandingkan antara satu kelas dengan kelas yang lain. Metode *one-against-one* pertama kali dipublikasikan dalam buku [22]. Data training dalam suatu kelas ditentukan apakah termasuk kelas 1 atau kelas 0. Demikian sampai seluruh data training dilatih. Contohnya dalam penelitian ini 100 dokumen akan dikelompokkan

menjadi 5 kelas {A, B, C, D, E}. Setiap dokumen uji akan ditentukan apakah termasuk kelas A atau bukan, kemudian diuji apakah termasuk kelas B atau bukan, dan seterusnya hingga kelas E.

Keputusan akhir apakah suatu dokumen uji termasuk kelas yang mana diantara 5, ditentukan oleh fungsi keputusan. Fungsi keputusan menggunakan cara *voting* untuk setiap dokumen menggunakan cara *voting*. Jika memenuhi fungsi keputusan, maka dokumen tersebut menjadi bagian dari kelas 1. Setelah semua dokumen *training* dilatih, maka yang memiliki nilai keputusan yang paling tinggi menjadi anggota kelas tersebut. Rumus (3) adalah fungsi keputusan.

$$class\ x \equiv \arg\max_{i=1,..,k} [(w^i)^T \phi(x) + b^i] \qquad (3)$$

Dimana$k$adalah jumlah klasifikasi SVM biner yang dilakukan, $\emptyset$ adalah parameter penalti, b adalah *treshold,w* adalah bobot. Jumlah k dapat ditentukan berdasarkan rumus (4), dengan n adalah jumlah kelas yang akan dikelompokkan.

$$k = \frac{n(n-1)}{2} \qquad (4)$$

Berdasarkan hal tersebut dikatakan bahwa sebuah data *training* menjadi anggota kelas x jika memenuhi fungsi keputusan[21].

III. HASIL PENELITIAN

Metode pengujian yang digunakan adalah *K fold cross validation*. Pengujian dilakukan terhadap dokumen latar belakang karya ilmiah berjumlah 100 buah yang akan dikategorikan ke dalam kelas A, B, C, D dan E. Setiap kelas memiliki data sejumlah 25 buah. Dengan menggunakan 10 *fold*, maka tiap fold akan berjumlah 10 dokumen yang berbeda.

Untuk menguji pengaruh peringkasan maka pengujian akurasi mempertimbangkan dua kondisi yaitu dengan proses peringkasan dan tanpa peringkasan. Sedangkan untuk melihat performansi dari SVM maka hasil klasifikasi dari metode SVM akan dibandingkan dengan beberapa metode klasifikasi yang sering digunakan yaitu: *Naïve Bayes Multinomial* (NBM), *Naïve Bayes Classifier* (NBC), dan *Simple Logistic Classifier* (SLC). NBM, NBC, SLC dipilih karena karakterikstik metode ini bekerja baik untuk data set yang dapat dipisahkan secara linear[23][24].Joachims menyebutkan bahwa masalah klasifikasi teks sering termasuk ke dalam jenis data yang dapat dikelompokkan secara linear [25]. Menurut Zhu, NBM dan NBC bekerja dengan baik untuk data latih yang sedikit sedangkan SLC bekerja baik untuk data latih yang besar.

Tabel 3 adalah hasil pengujian dari penggunaan metode klasifikasi *Support Vector Machine* (SVM), *Naïve Bayes Multinomial* (NBM), *Naïve Bayes Classifier* (NBC), dan *Simple Logistic Classifier* (SLC). Kata yang digunakan sebelum dan sesudah peringkasan berkurang sebanyak 645 kata. Akan tetapi, pengaruh peringkasan terhadap peningkatan hasil akurasi tidak terlalu besar. Hal ini terlihat pada setiap metode klasifikasi SVM dan NBC justru mengalami penurunan akurasi sebesar 2%, sedangkan NBM dan SLC justru mengalami peningkatan sebesar 1-2%. Dari semua metode yang digunakan dalam pengujian tanpa proses seleksi kata, metode *Naïve Bayes Multinomial* memiliki nilai akurasi paling besar yaitu 79% tanpa ringkasan dan 80% dengan proses peringkasan terlebih dahulu.

Tabel 3. Hasil Akurasi Klasifikasi Dokumen

Cara	Jumlah kata	Akurasi (dalam %)			
		SVM	NBM	NBC	SLC
tanpa peringkasan	4555	78	79	73	67
peringkasan	3910	77	80	72	69

IV. PEMBAHASAN

Berdasarkan hasil yang diperoleh pada Tabel 3 maka dapat dilihat penggunaan tahapan peringkasan dan seleksi fitur pada SLC akan membuat nilai akurasi bertambah. Akan tetapi, penambahan proses peringkasan tidak membuat akurasi SLC lebih tinggi dibandingkan dengan SVM, NBM dan NBC.

Untuk metode SVM yang biasanya merupakan metode yang memiliki kinerja yang lebih baik untuk data yang berdimensi tinggi, pada kasus ini ternyata tidak lebih baik dari pada NBM dan NBC. Khusus metode SVM, penggunaan proses peringkasan tidak meningkatkan performansinya. Hasil yang diperoleh ini berbeda dengan yang didapatkan oleh Ghaddar, B., & Naoum-Sawaya, J. yang menunjukkan performasi SVM semakin baik dengan adanya proses seleksi terlebih dahulu pada kasus analisis sentiment [17]. Nilai akurasi SVM yang justru berkurang setelah dilakukan proses peringkasan memunculkan dugaan bahwa seleksi fitur dengan peringkasan justru telah menghilangkan fitur kata yang relevan dalam menentukan kelas dari data yang diuji.

Dari pengujian yang dilakukan, ternyata untuk kasus pengelompokan dokumen karya ilmiah menggunakan proses peringkasan terlebih dahulu tidak membuat hasil klasifikasi SVM, NBM, dan NBC meningkat. Hasil pengujian ini berbeda dengan yang diperoleh dari Hernández dkk yang menyampaikan bahwa peringkasan teks bisa digunakan sebagai metode seleksi fitur [2]. Kemungkinan hal ini terjadi disebabkan oleh proses peringkasan otomatis yang belum berhasil mengurangi jumlah penggunaan kata secara signifikan. Hal ini terlihat pada Tabel 3, dimana setelah proses peringkasan sebesar 50% dilakukan, jumlah kata berkurang hanya sebesar 645 dari 4555 kata yang digunakan sebagai fitur.

V. PENUTUP

A. Kesimpulan

Penelitian ini telah memperlihatkan bahwa penggunaan peringkasan pada metode SVM tidak mengakibatkan nilai akurasi dari klasifikasi dokumen meningkat, khususnya untuk data set yang digunakan dalam penelitian ini. Pada kasus klasifikasi dokumen metode SVM menunjukkan nilai akurasi yang baik namun nilainya tidak lebih baik dari penggunaan metode NBM. Untuk observasi beberapa metode klasifikasi ternyata seleksi fitur dengan menggunakan peringkasan meningkatkan nilai akurasi pada metode SLC.

B. Saran

Untuk meningkatkan pengaruh peringkasan dalam klasifikasi, disarankan menggunakan fitur ekstraksi seperti kedekatan kalimat dengan judul, mempertimbangkan kalimat pertama atau kalimat terakhir sebagai kalimat penting, dan lain-lain.

UCAPAN TERIMA KASIH

Penelitian ini merupakan bagian dari penelitian dosen pemula (PDP) tahun 2016 dengan judul Klasifikasi Dokumen Skripsi Menggunakan Support Vector Machine.

Ucapan terima kasih kepada Ristekdikti khususnya DIPA Direktorat Penelitian Pengabdian kepada Masyarakat Kementrian Pendidikan dan Kebudayaan yang memberikan dukungan materil atas terlaksananya penelitian ini dan kepada Balai Bahasa Jawa Barat atas bantuannya dalam memvalidasi hasil pengujian ringkasan yang telah dilakukan.

DAFTAR PUSTAKA

[1] A. Kotcz, V. Prabakarmurthi, and J. Kalita, "Summarization as Feature Selection for Text Categorization," in *CIKM'01*, 2001, pp. 365–370.

[2] E. Anguiano-hernández and L. Villaseñor-pineda, "Summarization as Feature Selection for Document Categorization on Small Datasets," pp. 39–44, 2010.

[3] E. Al-thwaib, "Text Summarization as Feature Selection for Arabic Text Classification," vol. 4, no. 7, pp. 101–104, 2014.

[4] S. Harer, "Sentiment Classification and Feature based Summarization of Movie Reviews in Mobile Environment," vol. 100, no. 1, pp. 30–35, 2014.

[5] P. Bharambe and P. S. Deokar, "Classification and Summarization on rating of Mobiles features," vol. 5, no. 9, pp. 1–5, 2015.

[6] H. Jeong, Y. Ko, and J. Seo, "How to Improve Text Summarization and Classification by Mutual Cooperation on an Integrated Framework," *Expert Syst. Appl.*, 2016.

[7] K. E. Dewi, N. Indriani, and E. Rainarli, "Evaluasi Sentence Extraction pada Peringkasan Dokumen Otomatis," 2017.

[8] E. Chisholm and T. G. Kolda, "NEW TERM WEIGHTING FORMULAS FOR THE VECTOR SPACE METHOD IN INFORMATION RETRIEVAL," *Comput. Sci. Math. Div.*, pp. 1–16, 1999.

[9] J. D. Rajaraman, A.; Ullman, "Data Mining," in *Mining of Massive Datasets*, Cambridge University Press, 2011, pp. 1–17.

[10] J. Beel, B. Gipp, S. Langer, C. Breitinger, and C. Breitinger, "Research-paper recommender systems : a literature survey," *Int. J. Digit. Libr.*, no. June, 2015.

[11] K. Spärck-Jones, "'What Might be in Summary?,'" *Inf. Retrieval'93*, p. 9–26., 1993.

[12] L. Auria and R. A. Moro, "Support Vector Machines (SVM) as a Technique for Solvency Analysis," *DIW Berlin Discuss. Pap.*, vol. 811, no. August, 2008.

[13] H. Brcher, G. Knolmayer, and M.-A. Mittermayer, "Document classification methods for organizing explicit knowledge," in *the Third European Conference on Organizational Knowledge, Learning, and Capabilities*, 2002.

[14] A. Govada, S. Ranjani, A. Viswanathan, and S. K. Sahay, "A Novel Approach to Distributed Multi-Class SVM," Zuarinagar, Goa, PIN - 403726, India, 2011.

[15] S. Chakrabarti, S. Roy, and M. V. Soundalgekar, "Fast and accurate text classification via multiple linear discriminant projections," in *Proceedings of the 28th VLDB Conference*, 2002.

[16] M. Ring and B. M. Eskofier, "An approximation of the Gaussian RBF kernel for efficient classification with SVMs," *Pattern Recognit. Lett.*, 2016.

[17] B. Ghaddar and J. Naoum-sawaya, "High Dimensional Data Classification and Feature Selection using Support Vector Machines," *Eur. J. Oper. Res.*, 2017.

[18] X. Ju, Y. Tian, D. Liu, and Z. Qi, *Nonparallel Hyperplanes Support Vector Machine for Multi-class Classification*, vol. 51. Elsevier Masson SAS, 2015.

[19] B. Aisen, "A Comparison of Multiclass SVM Methods," 2006. [Online]. Available: http://courses.media.mit.edu/2006fall/mas622j/Projects/aisen-project/. [Accessed: 08-Oct-2017].

[20] R. Jindal, "Techniques for text classification : Literature review and current trends," *Webology*, vol. 12, no. 2, pp. 1–28, 2015.

[21] C. Hsu and C. Lin, "A Comparison of Methods for Multiclass Support Vector Machines," *IEEE Trans. NEURAL NETWORKS*, vol. 13, no. 2, pp. 415–425, 2002.

[22] U. Kreßel, *"Pairwise classification and support vector machines"- in Advances in Kernel Methods—Support Vector Learning*. Cambridge, MA: MIT Press, 1999.

[23] S. Raschka, "Introduction and Theory." pp. 1–20, 2014.

[24] X. Zhu, "CS838-1 Advanced NLP : Text Categorization with Logistic Regression," no. 3. pp. 1–3, 2007.

[25] T. Joachims, "Text Categorization with Support Vector Machines : Learning with Many Relevant Features," in *Machine Learning: ECML-98*, 1998, pp. 2–7.