KLASIFIKASI JURNAL ILMU KOMPUTER BERDASARKAN PEMBAGIAN *WEB OF SCIENCE* DENGAN MENGGUNAKAN *TEXT MINING*

Sri Widaningsih[1], Agus Suheri[2]
[1,2]Program Studi Teknik Informatika, Fakultas Teknik, Universitas Suryakancana Cianjur
Jl. Pasirgede Raya Cianjur 43216
E-mail : sriwida@unsur.ac.id, agussuheri@unsur.ac.id

ABSTRAK
Pada penelitian ini dilakukan klasifikasi jurnal-jurnal ilmiah untuk bidang ilmu komputer berbahasa Inggris berdasarkan pada pembagian kategori penelitian berdasarkan web of science yaitu artificial intelligent , cybernetics, hardware and architecture, information systems, interdisciplinary applications, software engineering, theory and method, dan image science and photography. Dari setiap kategori diambil sebanyak 50 jurnal sehingga terdapat total 400 jurnal. Sebanyak 360 jurnal menjadi data latih dan 40 jurnal sebagai data uji. Proses untuk klasifikasi ini menggunakan tahapan text mining yang merupakan bagian dari data mining. Teknik klasifikasi menggunakan dua metode yaitu Naïve Bayes Classifier dan Support Vector Machine. Proses teks mining menggunakan bantuan software rapidminer 8.0.. Ukuran performansi yang digunakan yaitu recall, precision, F-Measure dan accuracy. Metode Naïve Bayes Classifier memberikan nilai performansi yang lebih baik dibandingkan dengan metode Support Vector Machine dengan nilai recall 64,90% , precision 69,23% , F-Measure 66,99% dan accuracy 64,42%.
Kata Kunci: *klasifikasi, jurnal, text mining, Naive Bayes, Support Vector Machine*

1. PENDAHULUAN

1.1 Latar Belakang

Jurnal ilmiah merupakan salah satu bukti karya tertulis yang dihasilkan oleh seorang peneliti yang melakukan penelitian pada suatu topik atau bidang tertentu. Jurnal-jurnal tersebut umumnya diterbitkan oleh lembaga-lembaga penyedia informasi ilmiah maupun institusi perguruan tinggi. Saat ini, jumlah jurnal ilmiah meningkat sangat cepat terutama dengan adanya perkembangan internet karena untuk menerbitkan jurnal diluar institusi baik dalam maupun luar negeri saat ini lebih mudah dengan proses yang lebih cepat. Sebagai contoh menurut www.elsevier.com, jumlah artikel penelitian yang telah diterbitkan oleh elselvier setiap tahunnya adalah 420.000 sehingga sampai sekarang secara total terdapat sekitar lebih dari 25 juta dokumen. Dengan jumlah data yang begitu besar akan sangat sulit untuk melakukan klasifikasi dokumen secara manual karena akan membutuhkan waktu yang lama dan dibutuhkan tingkat ketelitian yang sangat besar. Untuk itu dibutuhkan suatu teknik dalam bidang pemrosesan dokumen untuk menentukan klasifikasi topik tertentu secara otomatis. Salah satu teknik yang dapat melakukan klasifikasi secara cepat adalah dengan *text mining*. *Text mining* merupakan bagian dari bidang *data mining* yang bertujuan untuk menggali dan menemukan informasi-informasi , pola, dan tren yang tersembunyi dari jumlah data yang besar.

Klasifikasi dokumen merupakan salah satu bagian dari *machine learning* dalam bidang *natural language processing* (NLP). Tujuan dari klasifikasi dokumen adalah untuk menentukan suatu dokumen termasuk ke dalam suatu kategori tertentu yang telah ditetapkan sebelumnya. Selain penentuan kategori pada jurnal-jurnal ilmiah, hal ini sangat berguna terutama untuk aktivitas-aktivitas yang membutuhkan penentuan kategori dokumen seperti penerbit buku, surat kabar, blog , atau majalah yang umumnya menerima berbagai macam berita maupun tulisan yang terkadang harus dimasukkan ke dalam suatu topik tertentu sebelum diterbitkan. Klasifikasi tersebut tidak hanya dapat dilihat pada judul dokumen saja, tetapi juga harus melihat pada isi dokumen agar dapat dilakukan klasifikasi secara tepat.

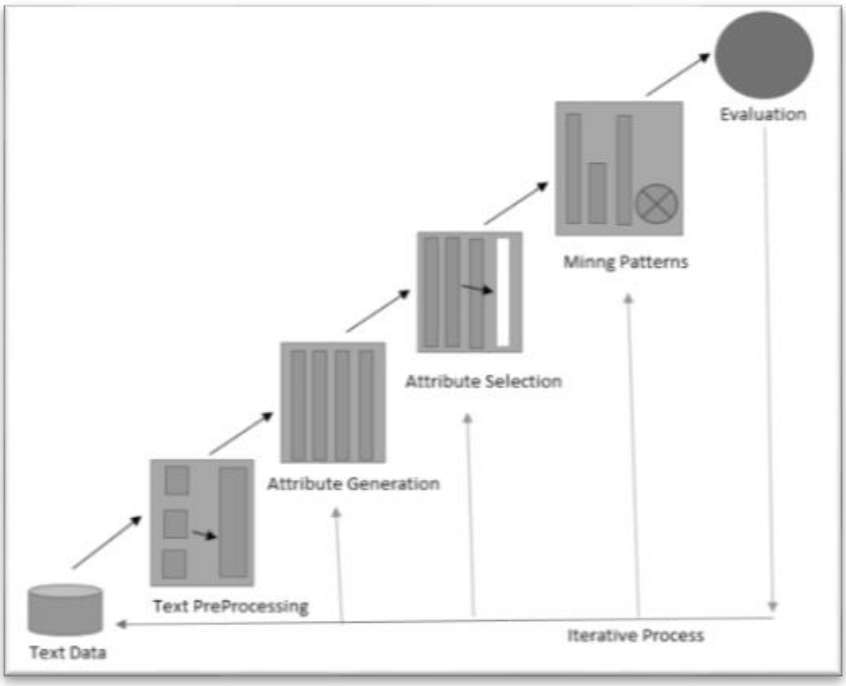
Jurnal-jurnal ilmiah secara umum dibagi menjadi beberapa kategori misalkan untuk kategori yang bersifat sosial dan teknik. Setiap kategori juga akan dibagi-bagi lagi berdasarkan subjek topik yang lebih mendetail yang menggambarkan isi dari jurnal tersebut. Setiap lembaga penyedia informasi memiliki dasar dalam pengklasifkasian jurnal yang akan terbit. Dalam penelitian ini, kategori klasifikasi yang digunakan berdasarkan pada area penelitian yang terdapat pada *Web of Science* . Subjek yang diambil yaitu *computer science* (ilmu komputer) yang terdiri dari delapan kategori yaitu *artificial intelligent , cybernetics, hardware and architecture, information systems, interdisciplinary applications, software engineering, theory and method*, dan *image science and photography.*

1.2 Tinjauan Pustaka

1.2.1 *Text Mining*

Text mining merupakan suatu teknik dalam ilmu komputer yang digunakan untuk memecahkan permasalahan informasi yang sangat banyak dengan mengkombinasikan teknik dari *data mining, machine learning, natural language processing, information retrieval* dan *knowledge management.* Seperti pada *data mining, text mining*

berusaha mengekstrak informasi yang bermanfaat dari sumber data melalui identifikasi dan eksplorasi pola menarik. Pada *text mining* sumber data adalah kumpulan dokumen, ini berarti dapat berupa koran, majalah, artikel, surat, ataupun laporan-laporan penelitian seperti jurnal , tugas akhir, atau tesis (Feldman dan Sanger, 2007). Dengan *text mining* dapat diketahui pengelompokkan dokumen yang serupa, memperkirakan penulis dari sebuah dokumen, atau mengklasfikasikan dokumen ke dalam suatu kategori. Tetapi tidak seperti data yang pada *data mining* yang umumnya sudah terstruktur, dokumen merupakan data yang tidak terstruktur. Proses *text mining* sama dengan tahapan *Knowledge Discovery in Database* (KDD) pada *data mining.*



Gambar 1. Proses *text mining* (Kumar dan Bhatia, 2013)

Berikut ini adalah proses *text mining* mengacu pada KDD

a. *Text Data*

Dokumen adalah syarat utama dalam *text mining.* Data berupa teks merupakan suatu fragmen yang dianggap sebagai unit. Dapat berupa buku, paragraph, abstrak, atau pun judul. Untuk *web*, fragmen teks adalah halaman *web*.

b. *Text PreProcessing*

Pada tahap ini dilakukan pembersihan data dan tokenisasi. Pembersihan data dilakukan untuk menghilangkan informasi yang tidak perlu atau tidak diinginkan seperti tabel, gambar atau rumus-rumus. Pada tahap tokenisasi, dokumen diperlakukan sebagai bentuk *string* dan dipecah menjadi token. Pada token kalimat-kalimat yang ada pada dokumen dipisahkan setiap kata dan karakter yang membentuknya. Dengan membentuk token, maka dapat dilakukan analisis teks lebih lanjut (Bhumika, et al , 2013).

Kata-kata yang bersifat *stopword* juga dihilangkan. Kata-kata *stopword* seperti "*a", "the"* ," *but*" adalah yang dibutuhkan pada struktur *grammer* dalam bahasa Inggris tetapi tidak memiliki arti dan tidak dibutuhkan dalam *text mining* . Pada tahap ini juga dilakukan proses *stemming* yang mengubah kata-kata dalam bentuk dasar. Misal : *connection* menjadi *connect*.

c. *Feature Transformation (Attribute Generation*)

Sebuah dokumen teks diwakili oleh kata-kata (fitur) yang dikandungnya dan kejadiannya. Representasi dokumen merupakan salah satu teknik yang digunakan untuk mengurangi kompleksitas dokumen sehingga lebih mudah untuk ditangani. Dokumen harus ditransformasikan dari versi teks lengkap ke dalam bentuk vektor dokumen. Representasi dokumen yang umum digunakan disebut *vector space model* dimana dokumen direpresentasikan menjadi vektor dari kata-kata. Beberapa keterbatasannya adalah: representasi dimensi yang tinggi, hilangnya korelasi dengan kata-kata yang berdekatan dan hilangnya hubungan semantik yang ada didalam dokumen. Untuk mengatasi masalah ini, metode pembobotan kata digunakan untuk menetapkan bobot yang sesuai dengan kata tersebut. Pada model ruang vektor, dokumen diwakili oleh vektor kata-kata karena kata-kata yang membentuk dokumen akan menentukan isi dari dokumen tersebut (Bhumika et al, 2013).

d. *Attribute Selection / Feature Selection*

Tahap ini adalah teknik pengurangan dimensi yang efektif untuk menghilangkan fitur *noise*. Fitur *noise* merupakan informasi-informasi yang tidak berguna yang dapat mengganggu hasil penelitian dalam *text mining* , seperti tulisan *copyright*, menu navigasi pada halaman *web*, dan lain-lain (Ting et al, 2011).

Pemilihan fitur juga dikenal sebagai pemilihan variabel, adalah proses pemilihan subset dari fitur yang penting untuk digunakan dalam pembuatan model. Hal ini dilakukan karena data mengandung banyak fitur yang berlebihan atau tidak relevan. Misalkan fitur yang berulang adalah salah satu yang tidak memberikan informasi tambahan. Fitur yang tidak relevan tidak memberikan informasi yang berguna atau relevan dalam konteks apapun.. Pemilihan fitur dilakukan dengan menyimpan kata-kata dengan bobot tertinggi sesuai dengan ukuran yang telah ditentukan terhadap pentingnya kata tersebut.

e. *Mining Pattern*

Pada tahap ini teknik yang digunakan *text mining* digabungkan dengan teknik klasik pada *data mining*, seperti klasifikasi, pengelompokkan atau asosiasi. Teknik-teknik tersebut dapat mengolah data yang telah terstruktur sebagai hasil pengolahan dari tahapan *text mining* yang telah dilakukan.

f. *Evaluation*

Pada tahap ini dilakukan evaluasi dari hasil *mining pattern*. Hasil evaluasi umumnya menggunakan suatu nilai performansi sesuai dengan teknik *data mining* yang digunakan.

1.2.2 Klasifikasi Teks

Klasifikasi teks didefinisikan sebagai pengkategorian teks secara otomatis ke dalam satu atau lebih kelas yang telah ditentukan berdasarkan isinya, misalkan pada koran jika ada isi berita mengenai kenaikkan tingkat suku bunga, atau penurunan bea masuk maka akan dimasukkan kedalam artikel ekonomi (Sebastiani, 2002). Tujuan dari kategorisasi teks adalah menguji pengklasfikasian teks yang belum diketahui kategorinya, jadi jika ada teks yang baru dapat lebih mudah diklasfikasikan pada suatu kategori berdasarkan teks-teks yang telah ada sebelumnya (Gaikwad et al, 2014). Dokumen dapat diklasifikasi kedalam satu kelas yang disebut "*single-label*" maupun beberapa kelas dan disebut sebagai "*multi-label*". *Single label* misalkan suatu artikel hanya dimasukkan kedalam satu kategori berita, sedangkan artikel yang multi label, dapat dimasukkan kedalam dua kategori berita /tidak saling terpisah (Wang dan Chiang, 2011). Beberapa algoritma klasifikasi yang biasanya digunakan untuk klasifikasi teks yaitu *Naïve Bayes Classifier* (NBC) , *Support Vector Machine* (SVM) , *neural network* , *decision tree* dan *K-nearest neighbor* (kNN*)*.

Perbandingan banyaknya penelitian yang menggunakan algoritma klasifikasi dilakukan Jindal ,et al (2015) terhadap 132 paper, dan dari paper tersebut algoritma yang paling banyak digunakan adalah SVM , kNN, dan NBC. Pengklasifikasian dokumen menggunakan NBC dilakukan pada klasifikasi kategori cerita pendek yang dibagi ke dalam dua kategori yaitu dongeng dan cerita anak dan menghasilkan tingkat akurasi sebesar 78,95% (Somantri, 2017). Indranandita, dkk (2008) menggunakan algoritma NBC untuk sistem klasifikasi dan pencarian jurnal. NBC menghasilkan prediksi yang baik jika vektor yang digunakan mewakili semua kategori. Algoritma klasifikasi NBC , SVM, dan kNN digunakan dalam penelitian Shrihari dan Desai (2015) dan dihasilkan bahwa SVM memberikan nilai presisi dan *recall* yang paling tinggi. Penelitian yang dilakukan Purohit (2015) menghasilkan bahwa teknik *decision tree* yang memberikan nilai akurasi yang lebih tinggi dibandingkan dengan NBC. Ariadi dan Fithriasari (2015) menggunakan NBC dan SVM untuk klasifikasi berita berbahasa Indonesia. Dari keseluruhan performa, teknik SVM kernel linier lebih baik dari NBC. Kasifikasi berita juga dilakukan oleh Kurniawan ,dkk (2015) dengan menggunakan NBC.

1.2.3 *Naïve Bayes Classifier* (NBC)

NBC merupakan teknik klasifikasi berdasarkan pada probabilitas bersyarat. Algoritma NBC merupakan metode klasifikasi yang sederhana dan sudah lama tetapi sangat berguna pada *text mining* terutama dapat menangani jumlah atribut yang cukup besar dengan baik. Meskipun kurang akurat dibandingkan metode diskriminatif lainnya (seperti SVM), banyak peneliti membuktikan bahwa cukup efektif untuk mengklasifikasikan teks di banyak domain . Model *Naïve Bayes* memungkinkan setiap atribut berkontribusi terhadap keputusan akhir secara merata dan independen dari atribut lainnya, di mana komputasinya lebih efisien bila dibandingkan dengan pengklasifikasi teks lainnya.. Berikut ini adalah rumus teorama Bayes pada persamaan (1) (Feldmand dan Sanger, 2007) .

$$P(c|d) = \frac{P(d|c)xP(c)}{P(d)} \qquad (1)$$

P(c|d) berarti probabilitas dokumen d termasuk kedalam kategori c. P(d) tidak pernah diperhitungkan karena nilainya tetap untuk semua kategori. Untuk menghitung P(d|c) perlu dibuat suatu asumsi mengenai struktur dokumen d. Dokumen direpresentasikan dengan bentuk vektor d = (w_1, w_2, ..) dan diasumsikan semua koordinat adalah independen. Sehingga menjadi persamaan (2) :

$$P(c|d) = \prod_i P(wi|c).P(c) \qquad (2)$$

Dimana P(wi|c) adalah probabilitas kondisi kata w_i terjadi pada dokumen d pada kelas c. Interpretasi P(wi|c) adalah ukuran berapa banyak wi berkontribusi dimana c adalah kelas yang benar. (w1, w2, …wn) adalah token pada dokumen d dan bagian dari kosa kata yang digunakan pada klasifikasi dan "n" adalah jumlah token pada dokumen d. P(c) adalah probabilitas dokumen dalam c sejumlah dokumen (Gogoi dan Sarma, 2015). Berikut ini adalah probabilitas dokumen pada persamaan (3).

$$P(c) = \frac{jumlah\ dokumen\ dalam\ kategori\ c}{jumlah\ dokumen} \qquad (3)$$

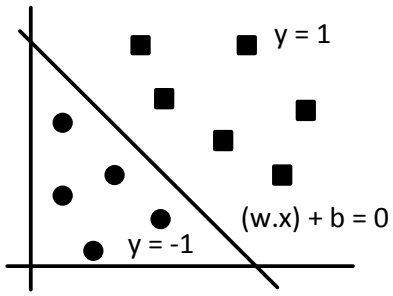
Tujuan klasifikasi teks adalah mencari kelas terbaik untuk dokumen. Kelas terbaik dalam NBC adalah *most likely* atau *maximum a posteriori* (MAP) yang dinotasikan dalam persamaan (4) :

$$C_{MAP} = \underset{c \in C}{argmax}\ P(c|d) = \underset{c \in C}{argmax}\ P(d|c).P(c) \qquad (4)$$

Klasifikasi NBC memprediksi kelas C_{MAP} dengan *posterior probability* paling besar. Klasifikasi yang dihasilkan dari asumsi ini disebut sebagai *Naïve Bayes Classifiers*. Disebut "naif" karena terdapat asumsi independensi antar kejadian itu tidak pernah diverifikasi . Padahal terdapat kemungkinan depedensi antar kejadian pada teorama ini. Tetapi asumsi "naif" pada model probabilistik dengan adanya depedensi tidak mempengaruhi performansi terlalu besar , sehingga dengan atau tanpa adanya independensi hal ini tidak terlalu mempengaruhi hasil yang diperoleh (Feldmand dan Sanger, 2007) .

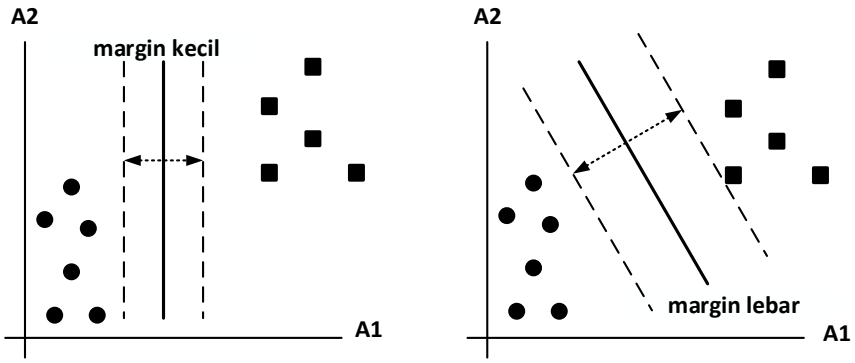
1.2.4 *Support Vector Machine* (SVM)

Algoritma SVM merupakan metode yang paling cepat dan efektif untuk permasalahan klasifikasi (Feldmand dan Sanger, 2007). Algoritma ini juga merupakan *machine learning* paling populer yang paling akurat untuk klasifikasi teks (Shrihari dan Desai, 2015). Awalnya, *Support vector machines* (SVM) dikembangkan untuk membangun pengklasifikasian biner (dua kelas) yang optimal namun kemudian teknik ini diperluas ke masalah regresi dan pengelompokan. Dalam istilah geometris, pengklasifikasian biner SVM dapat dilihat sebagai *hyperplane* di ruang fitur yang memisahkan titik yang mewakili contoh positif dari titik yang mewakili kejadian negatif seperti pada gambar 2 di bawah.



Gambar 2. Bidang pemisah linier

Klasifikasi *hyperplane* dipilih selama pelatihan sebagai *hyperplane* unik yang memisahkan contoh positif yang diketahui dari contoh negatif yang diketahui dengan margin maksimal. Perbandingan antara margin kecil dengan marign besar dapat dilihat pada gambar 3. Margin adalah jarak dari *hyperplane* ke titik terdekat dari yang positif dan set negatif. SVM *hyperplane* ditentukan sepenuhnya oleh sebagian kecil dari contoh pelatihan secara relatif, yang disebut vektor pendukung (*support vector*). Sisanya data pelatihan tidak berpengaruh terhadap pengklasifikasian terlatih (Feldmand dan Sanger, 2007).



Gambar 3. SVM *hyperplane* dengan margin kecil dan margin lebar

1.2.5 Ukuran Performansi Klasfikasi

Ukuran performasi termasuk ke dalam tahapan evaluasi. Terdapat beberapa ukuran performansi untuk teknik klasifikasi yaitu *recall, precision, F-Measure* dan *accuracy*. Berikut ini adalah penjelasan dari ukuran performansi evaluasi (Hossin dan Sulaiman, 2015) :

a. *Recall* : *recall* digunakan untuk mengukur fraksi pola positif yang diklasifikasikan dengan benar
b. *Precision* : presisi digunakan untuk mengukur pola positif yang diprediksi dengan benar dari total pola yang diprediksi di kelas positif
c. *F-Measure* : suatu ukuran yang menggambarkan rata-rata harmonis antara *recall* dan nilai presisi
d. *Accuracy* : suatu ukuran rasio prediksi yang benar terhadap total jumlah sampel dievaluasi

1.3 Metodologi Penelitian

Metodologi dalam penelitian klasifikasi jurnal ilmu komputer mengikuti tahapan yang terdapat pada proses *text mining* . Berikut ini adalah penjelasan setiap proses, sedangkan tahapan proses klasifikasi yang digunakan pada penelitian ini dapat dilihat pada gambar 4.

a. Data Teks

Data dalam bentuk dokumen jurnal berbahasa Inggris merupakan data yang digunakan pada penelitian ini. Jurnal-jurnal yang diambil berasal dari internet sebanyak 50 data untuk setiap kategori berdasarkan *Web of Science* yaitu *artificial intelligent , cybernetics, hardware and architecture, information systems, interdisciplinary applications, software engineering, theory and method*, dan *image science and photography*.

b. Text PreProcessing

Pada tahap ini dilakukan proses *tokenizing, case folding , filtering* dan *stemming*. Tahap *tokenizing* atau *parsing*, adalah memotong string kalimat-kalimat yang ada di dalam jurnal berdasarkan tiap kata penyusunnya Pada proses *case folding*, isi jurnal diubah semuanya ke dalam huruf kecil menjadi huruf ‘a’ hingga ‘z’.. Pada tahap *filtering* kata-kata *stopword/stop list* dibuang. *Stemming* adalah proses mengubah kata menjadi kata dasar dengan membuang imbuhan yang ada.

c. Feature Transformation

Agar dapat diolah pada proses klasifikasi, kata-kata yang dihasilkan pada tahap sebelumnya diubah bentuknya menjadi *vektor space model*.

d. Feature selection

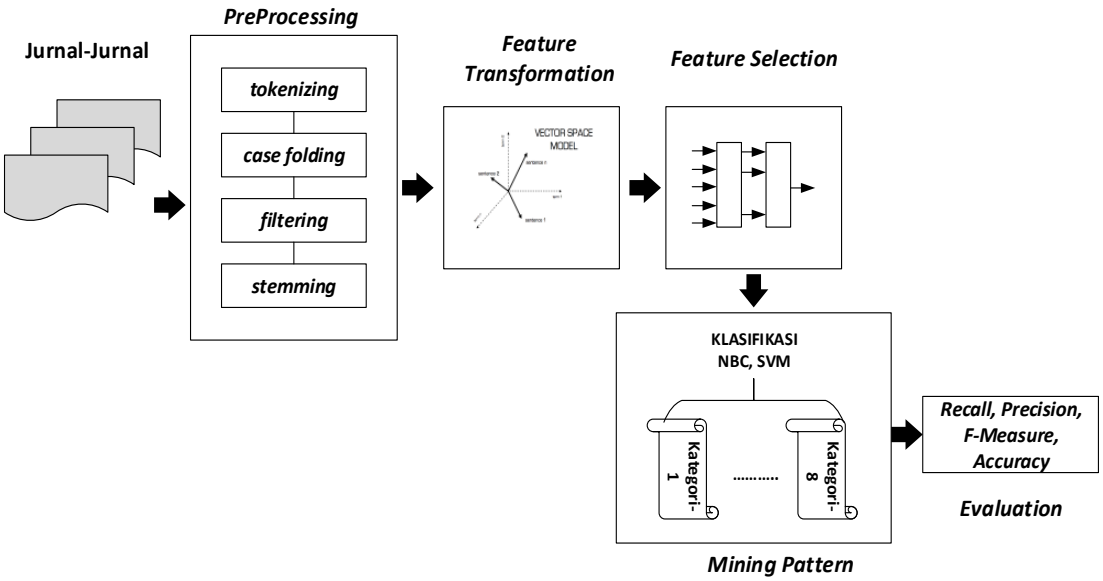
Salah satu teknik dalam *feature selection* adalah dengan algoritma TF-IDF (Khan et al, 2010). Algoritma TF-IDF adalah suatu algoritma yang berdasarkan nilai statistik menunjukkan kemunculan suatu kata di dalam dokumen. TF (*Term Frequency*) menyatakan banyaknya suatu kata muncul dalam sebuah dokumen. Dan DF (*Document Frequency*) menyatakan banyaknya dokumen yang mengandung suatu kata dalam satu segmen publikasi. TF-IDF adalah nilai bobot dari suatu kata yang diambil dari nilai IF dan nilai lnverse DF (Feldman dan Sanger,2007).

e. Mining Pattern

Algoritma yang digunakan untuk proses klasifikasi ini ada dua yaitu *Naïve Bayes Classifier* dan *Support Vector Machine.* Tipe kernel pada SVM menggunakan tipe dot.

f. Evaluasi

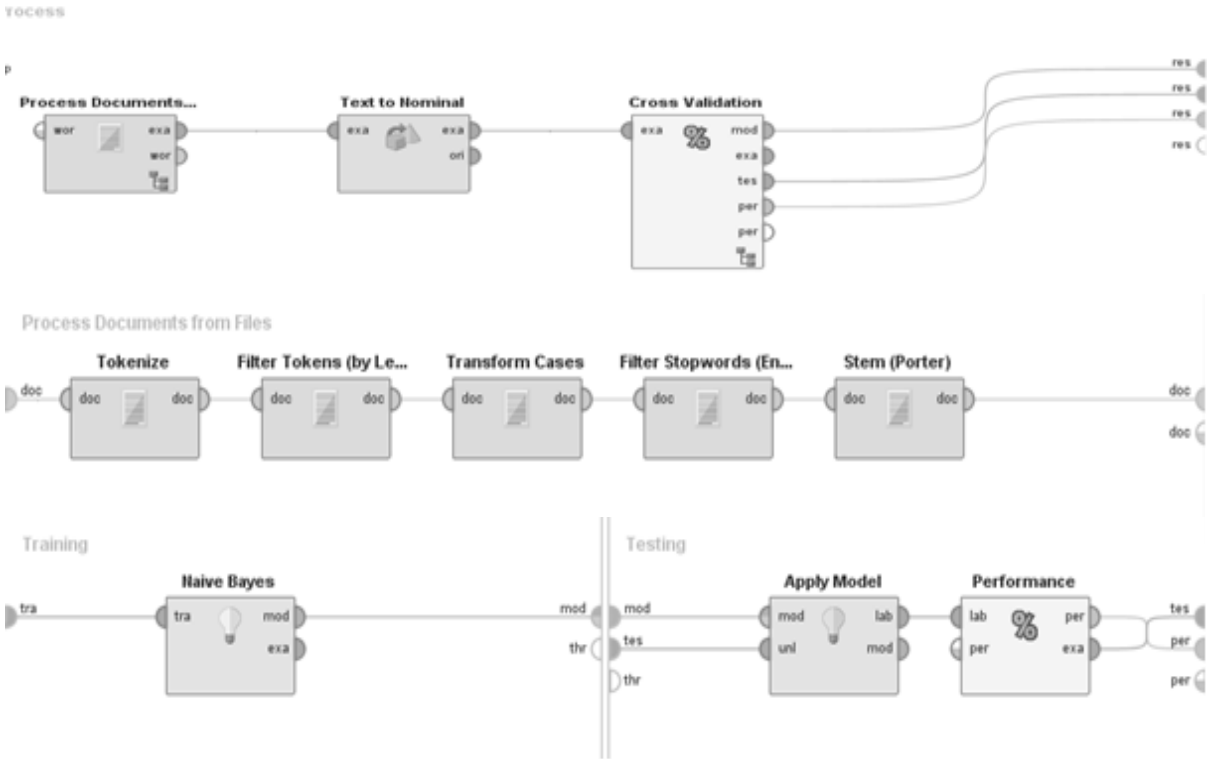
Ukuran performasi yang digunakan untuk mengevaluasi hasil dari klasifikasi NBC dan SVM menggunakan empat ukuran yaitu *recall, precision, F-measure* dan *accuracy*.



Gambar 4. Diagram proses klasifikasi

2. PEMBAHASAN

Untuk melakukan klasifikasi jurnal secara otomatis, pada penelitian ini menggunakan bantuan *software* Rapidminer 8.0 studio. Pada versi ini telah tersedia fasillitas *text processing*. Pada Rapidminer dapat dilakukan pemodelan data, proses data dan visualisasi. Serta menampilkan *output* berupa hasil evaluasi yang mempermudah untuk dilakukan interpretasi dari hasil klasifikasi yang menggunakan NBC dan SVM. Berikut ini adalah proses pemodelan klasifikasi pada *software* Rapidminer



Gambar 5. Proses klasifikasi pada rapidminer

2.1 Validasi

Jumlah total jurnal yang diambil sebanyak 400 dokumen. Teknik validasi yang digunakan untuk pada proses klasifikasi adalah *k-Fold Cross Validation. K-Fold Cross-validation* adalah metode statistik yang dapat digunakan untuk mengevaluasi kinerja model klasifikasi dimana data dipisahkan menjadi dua bagian yaitu data proses latihan (*training*) dan data uji . *k-Fold Cross Validation* digunakan karena dapat mengurangi waktu

komputasi dengan tetap menjaga keakuratan estimasi. Nilai k diambil 10 *fold* sehingga dari 400 dokumen akan menjadi 10 subset data dengan ukuran sama yaitu 40. Dari masing-masing 10 subset tersebut, 360 data menjadi data training dan 40 data menjadi data uji.

2.2 Hasil Evaluasi Setiap Kategori Klasifikasi

Tabel 1 merupakan *confusion matrix* untuk menjelasakan ukuran-ukuran performansi klasifikasi Untuk klasifikasi dengan kelas lebih dari dua (*multi class*), maka nilai *recall, precision, F-Measure* dan *accuracy* dapat dihitung dengan nilai rata-rata ukuran-ukuran tersebut dari setiap kelas (Sokolave dan Laplame, 2009).

Tabel 1. *Confusion matrix*

Aktual	Prediksi	
	Positive	*Negative*
Positive	*True Positive*	*False Negative*
Negative	*False Positive*	*True Negative*

$$Recall = \frac{\sum_{i=1}^{l} TP_i}{\sum_{i=1}^{l}(TP_i + FN_i)} \quad (5)$$

$$Precision = \frac{\sum_{i=1}^{l} TP_i}{\sum_{i=1}^{l}(TP_i + FP_i)} \quad (6)$$

$$F - Measare = \frac{2\ x\ recall\ x\ precision}{recall + precision} \quad (7)$$

$$Accuracy = \frac{\sum_{i=1}^{l} \frac{TP_i + TN_i}{TP_i + TN_i + FP_i + FN_i}}{l} \quad (8)$$

TPi : yaitu jumlah dokumen positif yang terklasifikasi dengan benar oleh sistem untuk kelas ke-i.
FNi : yaitu jumlah dokumen negatif namun terklasifikasi salah oleh sistem untuk kelas ke-i.
FPi : yaitu jumlah dokumen positif namun terklasifikasi salah oleh sistem untuk kelas ke-i
TNi : jumlah dokumen negatif yang terklasifikasi dengan benar oleh sistem untuk suatu kelas ke-i
l : jumlah kelas

Tabel 2. Ukuran performansi setiap kategori pada klasifikasi NBC

Kategori	*Recall*	*Precision*	*F-Measure*
Artificial intelligent	64%	60.38%	62.14%
Cybernetics	58%	64.44%	61.05%
Hardware and architecture	73.47%	65.45%	69.23%
Information systems	82%	42.71%	56.17%
Interdisciplinary applications	56%	63.64%	59.58%
Software engineering	38%	82.61%	52.06%
Theory and method	66%	91.67%	76.75%
Image science and photography	78%	82.98%	80.41%

Hasil output dari Rapidminer diperoleh nilai *recall* dan *precision* untuk setiap kelas pada kategori jurnal untuk klasifikasi dengan metode *naïve bayes classifier* seperti pada tabel 2 di atas. *Recall* berhubungan dengan kemampuan sistem klasifikasi untuk memanggil dokumen yang relevan. Nilai *recall* tertinggi ada pada pengklasifikasian jurnal sistem informasi sebesar 82%. Ini berarti sistem dapat mengklasifikasikan sebesar 82% jurnal sistem informasi dari jumlah jurnal yang telah diklasifikasikan sebagai jurnal sistem informasi (aktual). Nilai *recall* terendah ada pada klasifikasi jurnal *software engineering* (rekayasa perangkat lunak) sebesar 38%.

Presisi adalah jumlah kelompok dokumen relevan dari total jumlah dokumen yang ditemukan oleh sistem serta dapat mengukur tingkat efektivitas sistem. Nilai Presisi tertinggi ada pada pengklasfikasian jurnal teori dan metode sebesar 91,67%. Ini berarti model klasifikasi dapat mengklasifikasikan jurnal kategori teori dan metode sebesar 91,67% dari jumlah yang dokumen yang diprediksi / diambil dalam pencarian sebagai jurnal teori dan metode. Nilai presisi terendah ada pada pengklasifikasian jurnal sistem informasi.

F-measure merupakan salah satu perhitungan evaluasi yang mengkombinasikan antara nilai *recall* dan *precision*. Nilai *recall* dan presisi dapat memiliki nilai yang berbeda. Dengan demikian, dapat digunakan *F-Measure* dengan nilai *false negative* lebih kuat dari nilai *false positive*. Nilai *F-Measure* tertinggi terdapat pada pengklasifikasian jurnal dengan kategori *image science and photography* sebesar 80,41%, sedangkan untuk nilai terendah ada pada klasifikasi jurnal kategori *software engineering* sebesar 52,06%.

Tabel 3. Ukuran performansi setiap kategori pada klasifikasi SVM

Kategori	*Recall*	*Precision*	*F-Measure*
Artificial intelligent	64%	62.75%	63.37%
Cybernetics	48%	51.06%	49.48%
Hardware and architecture	59.18%	90.62%	71.60%
Information systems	62%	43.06%	50.82%
Interdisciplinary applications	46%	36.51%	40.71%
Software engineering	48%	42.11%	44.86%
Theory and method	66%	86.84%	75.00%
Image science and photography	70%	89.74%	78.65%

Nilai performansi untuk metode *Support Vector Mac*hine dapat dilihat pada tabel 3 di atas. Dari tabel tersebut dapat terlihat bahwa nilai *recall* tertinggi ada pada pengklasifikasian jurnal *image science and photography* sebesar 70%. Ini berarti sistem dapat mengklasfikasikan jurnal imange *sceiene and photography* sebesar 70% dari jumlah jurnal yang memang terklasfikasi sebagai kategori tersebut. Nilai *recall* terendah terdapat pada pengklasifikasian jurnal *interdisciplinary application*s sebesar 46%.

Nilai presisi tertinggi ada pada pengklasifikasian jurnal untuk kategori *hardware and architecture* sebesar 90,62%. Ini berarti model klasifikasi dapat mengklasifikasikan jurnal kategori *hardware and architecture* sebesar 91,67% dari jumlah yang dokumen yang diprediksi / diambil dalam pencarian sebagai jurnal *hardware and architecture*. Nilai presisi terendah ada pada pengklasifikasian jurnal kategori *interdisciplinary applications* sebesar 36,51%.

Nilai *F-Measure* tertinggi terdapat pada pengklasifikasian jurnal dengan kategori *image science and photography* sebesar 78,65%, sedangkan untuk nilai terendah ada pada klasifikasi jurnal kategori *interdisciplinary applications* sebesar 40,71%.

2.3 Perbandingan Metode Klasifikasi

Dengan menggunakan persamaan (5), (6) ,(7), dan (8) dapat dihitung nilai *recall, precision, F-measure* dan *accuracy* untuk mengetahui perbandingan performansi setiap metode. Dari tabel 4 perbandingan performansi klasifikasi, diperoleh hasil bahwa metode *Naïve Bayes* memiliki perfomansi lebih baik dibandingkan dengan metoda SVM karena memiliki nilai performasi yang lebih tinggi untuk semua ukuran performansi, walaupun tidak terlalu besar perbedaannya. Nilai akurasi tertinggi ada pada metode NBC dengan nilai akurasi 64,42%, ini berarti model klasifikasi yang dibuat dapat mengklasifikasikan jurnal secara benar ke dalam kategori sebesar 64,42% dari data keseluruhan. Nilai yang tidak terlalu tinggi ini dapat disebabkan karena isi teks yang terdapat pada suatu ketegori jurnal yang dijadikan sebagai data latih memiliki kesamaan dengan isi teks yang terdapat pada kategori jurnal yang lain.

Tabel 4. Perbandingan performansi klasifikasi

Metode	*Recall*	*Precision*	*F-Measure*	*Accuracy*
Naïve Bayes	64.90%	69.23%	66.99%	64.42%
Support Vector Machine	57.89%	62.83%	60.26%	57.90%

3. KESIMPULAN

Dari hasil proses *text mining* untuk klasifikasi jurnal ilmu komputer berdasarkan pembagian *Web of Science*, diperoleh bahwa klasifikasi dengan metode *Naïve Bayes Classifier* memberikan nilai performansi yang lebih tinggi dibandingkan dengan metode *Support Vector Machine*. Dari ukuran performansi yaitu *recall, precision*, F-*measure* dan *accuracy*, semua nilai performansi metode *Naïve Bayes* lebih besar. Berdasarkan pada nilai-nilai tersebut maka dapat dilakukan proses otomatisasi pengklasifikasian jurnal-jurnal lainnya dengan menggunakan metode *Naïve Bayes*.

Agar model klasifikasi dapat lebih mengklasifikasikan jurnal ke dalam kategori yang tepat dapat dicoba penggunaan teknik lain dalam proses *feature selection* selain yang digunakan dalam penelitian seperti *gain ratio* atau *chi square* sehingga dapat dipilih fitur yang paling tepat. Selain itu dapat dilakukan pengklasifikasian *multi label* sehingga satu jurnal dapat masuk kedalam beberapa kategori yang berbeda.

PUSTAKA

Ariadi, D., & Fithriasari, K. 2015. Klasifikasi Berita Indonesia Menggunakan Metode Naive Bayesian Classification dan Support Vector Machine dengan Confix Stripping Stemmer . *Jurnal Sains Dan Seni ITS*,(Online), Vol. 4, No.2, http://ejurnal.its.ac.id, diakses 15 Januari 2018

Bhumika, Sehra,S.S, & Nayyar,A. 2013. A Review Paper On Algorithms Used For Text Classification. *International Journal of Application or Innovation in Engineering & Management (IJAIEM)(Online), Volume 2 , No 3,* www.ijaiem.org , diakses 13 Januari 2018

Feldman , R., & Sanger, J .2007. *The Text Mining Handbook Advanced Approaches In Analyzing Unstructured Data* . New York : Cambridge University Press

Gaikwad,S.V., Chaugule, A., & Patil ,P.T 2014. Text Mining Methods and Techniques.*International Journal of Computer Applications* ,(Online) , Vol 85 ,No 17 , http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.428.8805&rep=rep1&type=pdf, diakses 9 Januari 2018

Gogoi, M.,& Sarma, S.K. 2015. Document Classification of Assamese Text Using Naïve Bayes Approach . *International Journal of Computer Trends and Technology (IJCTT)*, (Online), Vol 30, No 4, http://www.ijcttjournal.org, diakses 15 Januari 2018

Hossin, M. & Sulaiman, M.N. 2015. A Review On Evaluation Metrics For Data Classification Evaluations . *International Journal of Data Mining & Knowledge Management Process (IJDKP)*, (Online), Vol 5, No 2,https://www.researchgate.net/publication/275224157_A_Review_on_Evaluation_Metrics_for_Data_Classification_Evaluations, diakses 15 Januari 2018

Indranandita, A., Susanto, B., & Rahmat, A. 2008. Sistem Klasifikasi Dan Pencarian Jurnal Dengan Menggunakan Metode Naive Bayes Dan Vector Space Model. *Jurnal Informatika* , (Online) Vol 4, No 2 . https://www.researchgate.net/publication/265276639_SISTEM_KLASIFIKASI_DAN_PENCARIAN_JURNAL_DENGAN_MBNGGUNAKAN_METODE_NAIVE_BAYES_DAN_VECTOR_SPACE_MODEL, diakses 9 Januari 2018

Jindal, R., Malhotra, R., & Jain, A.2015. Techniques For Text Classification: Literature Review And Current Trends. *Webology*, (Online), Volume 12, No 2, *http://www.webology.org/2015/v12n2/a139.pdf*, *diakses 20 Januari 2018*

Kumar, L. & Bhatia, P.K. 2013.Text Mining: Concepts, Process And Applications. *Journal Of Global Research In Computer Science* ,(Online) ,Vol 4, No. 3, http://www.jgrcs.info, diakses 13 Januari 2018)

Kurniawan, B., Effendi, S. & Sitompul, O.S. 2012. Klasifikasi Konten Berita Dengan Metode Text Mining , *Jurnal Dunia Teknologi Informasi* , (Online) Vol. 1, No. 1, https://jurnal.usu.ac.id, diakses 15 Januari 2018

Khan, A., Baharudin, B., Lee, L.H., & Khan, K. 2010. A Review of Machine Learning Algorithms for Text-Documents Classification. *Journal Of Advances In Information Technology*, (Online), Vol. 1, No 1, http://www.jait.us/uploadfile/2014/1223/20141223050800532.pdf, diakses 12 Januari 2018

Purohit, A., Atre, D., Jaswani, P. ,& Asawara, P. 2015. Text Classification in Data Mining. International *Journal of Scientific and Research Publications*, (Online), Vol 5, No 6, www.ijsrp.org, diakses 9 Januari 2018

Sokolova, M. & Lapalme , G. 2009. A Systematic Analysis Of Performance Measures For Classification Tasks . *Information Processing and Management*, (Online), 45, http://atour.iro.umontreal.ca/rali/sites/default/files/publis/SokolovaLapalme-JIPM09.pdf, diakses 10 Januari 2018

Sebastiani, F. 2002. Machine Learning in Automated Text Categorization. *ACM Computing Surveys*, (Online) Vol 34, No 1 , http://nmis.isti.cnr.it/sebastiani/Publications/ACMCS02.pdf. Diakses 10 Januari 2018

Somantri, O. 2017. Text Mining Untuk Klasifikasi Kategori Cerita Pendek Menggunakan Naïve Bayes (NB). *Jurnal Telematika*, (Online), volume 12 , No. 1, https://www.researchgate.net/publication/320371563_Text_Mining_Untuk_Klasifikasi_Kategori_Cerita_Pendek_Menggunakan_Naive_Bayes_NB, diakses 10 Januari 2018

Shrihari, C., & Desai, A. 2015. A Review on Knowledge Discovery using Text Classification Techniques in Text Mining. *International Journal of Computer Applications* , (Online), Vol 111, No 6, https://pdfs.semanticscholar.org/1522/d935e6ff7fa185571f9c26c6ac9aef270bd8.pdf, diakses 10 Januari 2018

Ting,S.L., Ip, W.H.,& Tsang A.H.C. 2011. Is Naïve Bayes a Good Classifier for Document Classification? **.** *International Journal of Software Engineering and Its Applications*. (Online) Vol. 5, No. 3, https://www.researchgate.net/publication/266463703_Is_Naive_Bayes_a_Good_Classifier_for_Document_Classification, diakses 10 Januari 2018

Wang, T.Y., & Chiang, H.M. 2011. Solving Multi- Label Text Categorization Problem Using Support Vector Machine Approach With Membership Function. *Neurocomputing*, (Online), 74, https://www.journals.elsevier.com/neurocomputing, diakses 8 Januari 2018, diakses 20 Januari 2000).