

KLASIFIKASI LAPORAN KENDALA USER PADA SISTEM NEW OPERATION SUPPORT SYSTEM MENGGUNAKAN METODE SUPPORT VECTOR MACHINE

(Studi Kasus : Aplikasi IT Care Divisi L2 PT. Sigma Cipta Caraka)

Ika Fitri Nuryanti¹, Rizki Tri Prasetyo²

¹Universitas Adhirajasa Reswara Sanjaya
Jl. Sekolah Internasional No. 1-2 Antapani, Bandung, 022-7100124
e-mail: ika.16160073@ars.ac.id

²Universitas Adhirajasa Reswara Sanjaya
Jl. Sekolah Internasional No. 1-2 Antapani, Bandung, 022-7100124
e-mail: rizki@ars.ac.id

ABSTRAK

Sistem *New Operation Support System* (NOSS) yang bertujuan untuk meningkatkan efektifitas dan efisiensi dalam bussiness Telkom dengan sistem berbasis *Enterprise Resource Planning* (ERP) ini merupakan sistem yang cukup kompleks . Namun pada saat menggunakan sistem ini tidak jarang user menemukan kendala dalam penggunaannya, oleh karena itu user membutuhkan bantuan *Helpdesk* dalam mempermudah penggunaan aplikasinya. Guna mempermudah pengumpulan kedala yang diberikan user, Telkomsigma membuat aplikasi yang diberinama *IT Care* . Dikarenakan belum ada klasifikasi laporan kendala user, Sehingga pada penelitian ini akan dilakukan pengklasifikasian laporan kendala sistem NOSS pada kategori multiclass yaitu: *Unified Inventory Management* (UIM), *Order and Service Management* (OSM), *Automated Service Activation Program* (ASAP), *SmartCloud Control Desk* (SCCD), Dan lain-lain (DLL) . Klasifikasi ini dilakukan dengan menggunakan metode *Support VectorMachine* (SVM) dengan kernel RBF dan parameter yang digunakan adalah C dan γ . Dengan Jumlah dataset sebanyak 1553 data kendala, Pengujian dilakukan menggunakan aplikasi *RapidMiner*, diuji dengan menggunakan *10-Fold cross validation* dan diukur dengan *confussion matrix* untuk evaluasi hasil. Dari hasil uji coba aplikasi menunjukkan akurasi tertinggi sebesar 90.47% dengan pengujian tanpa menggunakan feature selection pada titik $C=3.0$, $\gamma=0.3$.

Kata Kunci : Klasifikasi , kendala , SVM

ABSTRACT

The New Operation Support System (NOSS) system which aims to increase the effectiveness and efficiency in Telkom business with an Enterprise Resource Planning (ERP) based system is a quite complex system. However, when using this system it is not uncommon for users to encounter problems in its use, because of that users need Helpdesk assistance in making it easier to use the application. In order to simplify the collection of information provided by the user, Telkomsigma created an application called IT Care. Because there is no classification of user constraint reports, so in this study classifications of NOSS system constraints will be classified in the multiclass category, namely: Unified Inventory Management (UIM), Order and Service Management (OSM), Automated Service Activation Program (ASAP), SmartCloud Control Desk (SCCD), and others (DLL). This classification is done using the Support Vector Machine (SVM) method with the RBF kernel and the parameters used are C and γ . With a total dataset of 1553 constraint data, the test was carried out using the Rapid Miner application, tested using 10-Fold cross validation and measured by a configuration matrix to evaluate the results. From the test results, the application shows the highest accuracy of 90.47% by testing without using feature selection at point $C = 3.0$, $\gamma = 0.3$.

Keyword : *Classification , Constraints , SVM*

1. Pendahuluan

Penggunaan sistem informasi sudah diterapkan diperusahaan sebagai media yang saling menghubungkan antara satu bagian dengan bagian lain untuk saling bertukar informasi baik berupa informasi yang sifatnya transaksi atau laporan (Agustina, 2018) . Dalam manajemen perusahaan terutama manajer menggunakan banyak laporan atau informasi untuk mencerminkan keadaan perusahaan (Fatta, 2007) .

Tak terkecuali Pada perusahaan *Internet Service Provider (ISP)* PT.Telkom Indonesia (Persero) Tbk menerapkan sistem teknologi informasi terintegrasi "*Enterprise Resources Planning*" (ERP) (Redaksi WE Online, 2017) . Salah satu sistem informasi PT.Telkom Indonesia yang menggunakan ERP adalah sistem NOSS / Next Generation OSS yang bertujuan untuk meningkatkan efektifitas dan efisiensi pengelolaan jaringan secara end to end serta mendukung operasional dan pemeliharaan layanan secara terpadu (PT.Telkom Indonesia, 2020) . Sistem yang berbasis ERP ini sendiri terdiri dari beberapa aplikasi yaitu *Unified Inventory Management (UIM)*, *Order and Service Management (OSM)*, *Automated Service Activation Program (ASAP)*, *SmartCloud Control Desk (SCCD)*, dan *Network Integrity7 (NI7)* aplikasi tersebut saling terintegrasi satu sama lain.

Namun pada saat menggunakan sistem ini tidak jarang *user* menemukan kendala dalam penggunaannya atau terbatasnya akses *user* pada menu aplikasi , oleh karena itu *user* membutuhkan bantuan admin dalam mempermudah penggunaannya , atau dengan meminta tolong pada *helpdesk*. Menurut website *help-desk-world* (Help Desk World, 2002) "*Helpdesk* sendiri merupakan titik sentral di mana masalah atau kendala dilaporkan dan selanjutnya dikelola dan dikoordinasikan"

Guna mempermudah pengumpulan kendala yang diberikan *user* Telkomsigma / PT. Sigma Cipta Caraka yang bertanggung jawab dalam pembuatan sistem NOSS sekaligus penanganan *operation* nya membuat aplikasi yang bernama *IT Care* . Kendala yang masuk pada aplikasi *IT Care* masuk terlebih dulu ke inbox *Helpdesk* L1 jika

kendala yg masuk pada inbox *Helpdesk* L1 tidak bisa dikerjakan maka kendala atau tiket pada aplikasi *IT Care* di assign ke *Helpdesk* L2.

Namun terkadang kendala yang masuk pada *inbox* L2 tidak menentu per hari nya , bisa dalam satu hari masuk kendala user banyak sekali, dikarenakan belum ada klasifikasi laporan kendala user, sehingga *helpdesk* L2 belum tau penyebab kendala ini muncul banyak dalam waktu tertentu, agar mengetahui juga penyebab dari kendala-kendala yang di laporkan *user* .

Dalam penelitian ini pengolahan data menggunakan *Data Mining* , *Data Mining* sendiri merupakan "metode yang dapat mengotomatisasi proses pengolahan data untuk mengekstraksi pengetahuan dari informasi yang tidak bisa diamati hanya dengan melihat data karena terlalu rumit atau multidimensi" (Dewi, 2017). Pada kasus data pada laporan kendala *IT Care* di menu *Helpdesk* L2 yang merupakan data *teks*, jenis metode *Data Mining* yang dapat digunakan adalah *Text Mining*. *Text Mining* memegang peran penting dalam analisis *Big Data* yang bersifat tidak terstruktur seperti data *teks* dan dalam jumlah yang sangat besar (Xiang, Schwart, Jr. c, & Uysal, 2015).

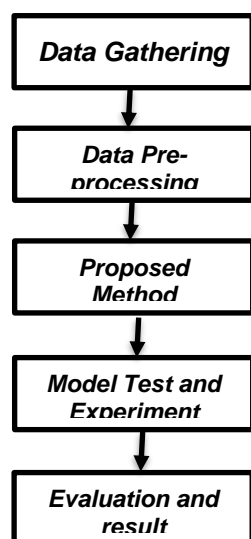
Pada penelitian ini akan dilakukan klasifikasi kendala user dengan menggunakan metode *Support Vector Machine (SVM)* . Metode tersebut dipilih karena *Support Vector Machine (SVM)* mampu mengidentifikasi *hyperplane* terpisah sehingga dapat memaksimalkan margin antara 2 kelas yang berbeda (Chou, Cheng, Wu, & Pham, 2014) . Selain itu penelitian terdahulu yang dilakukan oleh (Dewi, 2017) membuktikan bahwa metode *support vector machine* dengan diterapkan klasifikasi dapat menghasilkan sebuah model *text mining* identifikasi keluhan . Oleh sebab itu, penulis terdorong untuk membuat penelitian serupa dengan mengklasifikasi laporan kendala *user*.

Dengan mengembangkan penelitian menggunakan metode klasifikasi yang merupakan variasi dari *data mining* yaitu *Text mining* guna berusaha menemukan pola yang menarik dari sekumpulan data tekstual yang berjumlah besar" (Kurniawan, Effendi, & Sitompul, 2012) . Sedangkan menurut (Indriati & Ridok, 2016) "*Text mining* adalah

proses penemuan pola yang sebelumnya tidak terlihat pada dokumen atau sumber tertentu menjadi pola yang diinginkan untuk tujuan tertentu” .

2. Metode Penelitian

Terdapat empat metode penelitian yang umum digunakan, yaitu *Action Research*, *Experiment*, *Case Study*, dan *Survey* (Dawson, 2009). Pada penelitian kali ini yang digunakan adalah penelitian *Experiment*, yaitu penelitian yang melibatkan penyelidikan perlakuan pada parameter/variabel tergantung dari penelitiannya dan menggunakan tes yang dikendalikan oleh si peneliti itu sendiri. Berikut tahapan dari penelitiannya :



Gambar 1 Tahapan Desain Penelitian

2.1 Pengumpulan Data (*Data Gathering*)

Data yang digunakan pada penelitian ini merupakan data sekunder. Data sekunder adalah data yang tidak diperoleh langsung dari obyek penelitian, melainkan telah dikumpulkan oleh pihak lain atau data yang diambil dengan perantara . Data teks laporan yang digunakan adalah data laporan keluhan user yang dilaporkan pada aplikasi IT *Care* melalui Telegram yang terhitung mulai tanggal 1 Februari 2020 - 29 Februari 2020 . Dengan jumlah data keluhan sebanyak 1553 .

2.2 Pengolahan Data Awal (*Data Pre-processing*)

Pengolahan Data Awal atau *Text preprocessing* merupakan tahapan sangat penting dalam melakukan proses klasifikasi data teks. Tujuan dilakukannya *text preprocessing* yaitu untuk menghilangkan noise, menyeragamkan bentuk kata dan mengurangi volume kata (Berry & Kogan, 2010) . Data yang masuk sebelum diolah harus terlebih dahulu melalui tahapan preprocessing agar sistem mengerti sebelum melanjutkan proses *text mining* .

a. Pelabelan Data

Didalam permasalahan klasifikasi hal yang paling utama adalah melakukan pelabelan dengan benar agar data yang terkumpul atau terklasifikasi sesuai dengan kelas yang seharusnya (Wahjono, 2010). Pada penelitian ini akan dilakukan pelabelan terhadap laporan keluhan yang diambil berdasarkan keluhan user terkait aplikasi yang ada pada sistem NOSSF . Dan maka dari itu pelabelan keluhan hanya di bagi menjadi 5 kategori , yaitu : UIM, OSM,ASAP,SCCD,DLL

b. Tahap *Text Preprocessing*

Text Preprocessing merupakan langkah penting dalam melakukan analisa klasifikasi keluhan yang bertujuan untuk membersihkan data dari unsur-unsur yang ada yang tidak dibutuhkan. Berikut ini merupakan tahapan preprocessing: *Tokenizing*, *Transform cases*, *Filtering Filter Token (by Length)*, *Stopwords (Dictionary)*

c. Pembobotan

Merupakan proses merubah kata menjadi bentuk vektor, dimana tiap kata dihitung sebagai satu fitur. dengan menggunakan metode Term Frequency-Inverse Document Frequency (TF-IDF). Metode *term frequency inverse document frequency* (TF-IDF) merupakan cara pemberian bobot hubungan pada suatu kata (term) terhadap dokumen. (MA'ARIF, 2015) . Ukuran yang digunakan pada TF-IDF adalah ukuran statistik yang digunakan untuk mengevaluasi seberapa penting sebuah kata dalam sekelompok data. Frekuensi kemunculan kata di dalam dokumen menunjukkan seberapa penting kata di dalam dokumen tersebut (Putra, 2015) .

Pada algoritma TF-IDF digunakan rumus untuk menghitung bobot (W) masing – masing dokumen terhadap kata kunci dengan menggunakan rumus:

$$W_{dt} = tf_{dt} \times IDF_t$$

Gambar 2 Rumus Bobot

Keterangan :

W : Bobot dokumen ke-d terhadap kata ke-t

d : Dokumen ke-d

t : Kata ke-t dari kata kunci

tf : Banyaknya kata yang dicari pada sebuah dokumen

IDF : Inverse document frequency

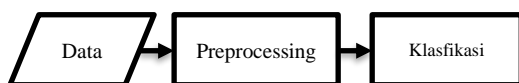
Persamaan Inverse document frequency:

$$IDF(t) = \log \frac{N}{df(t)}$$

Gambar 3 Persamaan IDF

2.3 Metode yang Diusulkan (*Proposed Method*)

Pemodelan pada penelitian ini menggunakan metode *Support Vector Machine* (SVM) dalam melakukan klasifikasi data. Proses klasifikasi akan menggunakan tools *RapidMiner* pada implementasi nantinya. Secara gambaran umum proses pada penelitian ini melalui tiga tahapan proses yaitu: proses pengambilan data , preprocessing, dan yang terakhir klasifikasi. Berikut gambaran tahapan dalam penelitian ini :



Gambar 4 Tahapan secara umum

Tahap *preprocessing* merupakan tahapan sangat penting dalam melakukan proses klasifikasi data teks. Guna mempersiapkan dan membersihkan data sebelum akan dilakukan proses klasifikasi dengan mempersiapkan data lalu melakukan pelabelan data berdasarkan aplikasi , lalu melakukan tahap-tahap text pre-processing setelah itu proses pembobotan dengan menggunakan metode TF-IDF lalu menghasilkan kata yang sudah dilakukan pembobotan pada file CSV .

Pada penelitian ini proses klasifikasi akan menggunakan tools *RapidMiner* sebagai tools pembantu . Dengan memasukan data CVS yang telah dilakukan pembobotan kata , lalu membagi data nya menjadi training dan testing .

Di dalam proses pengklasifikasian terdapat proses pembelajaran (*training*) dengan menggunakan data latih dan menghasilkan model yang akan dilakukan proses pengujian (*testing*) dengan beberapa data yang akan digunakan sebagai data uji

2.4 Eksperimen dan Pengujian Model (*Model Test and Experiment*)

Eksperimen pengujian model ini didukung dengan menggunakan tools *RapidMiner*, berikut tahapan eksperimen dan pengujian model berdasarkan metode yang diusulkan:

2.4.1 Tahap Preprocessing

1. Pada tahapan ini setelah membuat *document* baru pada *RapidMiner* menambahkan *operators* pada *panel process* berupa *operator read excel* dengan memasukan data keluhan yang telah diberikan label , *set role* , *nominal to text* , *Process Document* , *Set role* dan menyesuaikan attribute name dengan attribute yang tertera pada exce dan masukan juga *operator write CSV* . *Write CSV* sendiri nantinya berfungsi untuk menyimpan data *vector* yang telah dibuat.
2. Masuk kedalam *operator document* , *operator* ini berfungsi untuk menyimpan *process text processing*. Berupa *operator Tokenizing* , *Transform cases* , *Filtering Filter Token (by Length)* , *Stopwords (Dictionary)*. Tidak lupa untuk memasukan kamus *Stopwords* kedalam *operator Stopwords (Dictionary)*.
3. Setelah *process* memasukan *operator* selesai, selanjutnya *run document* , maka akan terbentuk pembobotan kata untuk keluhan pada sistem NOSSF.

2.4.2 Tahap Klasifikasi

1. Setelah tahap *preprocessing* selesai dan , dan pembobotan telah terbentuk maka selanjutnya dilakukan tahap klasifikasi , pertama buat *file document* baru. Lalu ubah file pembobotan yang telah terbentuk sebelum nya dari CSV ke format xlsx . Dan masukan *operator read excel* , dengan memasukan file pembobotan .

Dan juga tambahkan operator *Cross Validation*, operator ini merupakan metode *statistic* yang bertujuan untuk mengevaluasi dan membandingkan algoritma yang dibagi menjadi dua segmen yaitu: *testing* dan *training*.

2. *Double click* pada operator *cross validation*, lalu tampilan akan menjadi seperti gambar , lalu tambahkan operator *LibSVM* pada *training* dan operator *apply model* dan *performance* pada bagian *testing*. Operator *LibSVM* merupakan algoritma yang digunakan, Operator *apply model* menerapkan model pada dataset dan Operator *performance* memiliki fungsi sebagai pengirim hasil kinerja dari model yang digunakan.
3. Jalankan proses modelling dengan menggunakan *Process-Run* atau *Icon Run* guna mengetahui hasil *accuracy* paada process modeling ini

2.5 Evaluasi dan Validasi Hasil

Hasil evaluasi dengan menggunakan *software RapidMiner* dilakukan secara otomatis ketika menjalankan proses modelling seperti pada *process* klasifikasi. Hasil evaluasi akan tampil pada *Performance Vector* dan model yang telah digunakan akan tersimpan di result history. mendapatkan accuracy sebesar 90.47%

3. Hasil dan Pembahasan

Penerapan algoritma *Support Vector Machine* menggunakan *ekstraksi feature* dengan menggunakan *kernel RBF* dimaksudkan untuk mengetahui nilai akurasi dari klasifikasi keluhan pada sistem *NOSS* . Hasil akurasi akan terlihat pada hasil eksperimen. Eksperimen dilakukan dalam 2 (dua) tahap, tahap pertama adalah pengujian dengan data yang belum dilakukan pemilihan fitur dengan data yang sudah dilakukan pemilihan fitur Berikut merupakan contoh hasil dari data yang diperoleh.

Tabel 1 Pelabelan Data

No	Probability	Text
1	OSM	Cabutan ;#REQ #ASTINET ;1-9717922788 ;SOM IS » 185146735 ;SOM_HandleFailure Received
2	SCCD	#REQ #METRO ;Nomor NCX/SC : 1-9673241609 ;Order ID SOM - Task : 185687673-InstalNTETask ;Order ID TOM - Task : ;Reservation ID : ;Tiket - Deskripsi : ;Problem : ;Moban munculkan task selanjutnya dan pelurusan <i>workzonenya</i> ke BLK (BULUKUMBA - SULSEL (MAKASSAR))
3	UIM	#REQ #NOSSF ; ;SC502026299 ;IN62059465]The resource type PhysicalPort with Id 132476803 - 0247472027 is not available because the resource is already assigned. ;SOM 186461381 GetSOMTTResponse ;TOM - ; ;moban solusinya rekan
4	DLL	#REQ #NOSSF ;moban rekan wfm id kosong, mau di lanjut progres ;SC 501696538 issued ;WO kosong ;SOM 176190064 GetTechnicalOrderTask ;TOM 176190484 WorkForceTask ;order PDA ;terimakasih
5	ASAP	#REQ #WIFI Rekan mohon info apakah asap2 sedang gangguan? order2 kena <i>prevalidate</i> ... coba buka jg gabisa

Tahap selanjutnya yang dilakukan pada penelitian yaitu *text preprocessing*, yang merupakan pengolahan data awal yang belum terstruktur menjadi data yang terstruktur sehingga dapat diolah pada penelitian ini . Berikut tahapan *text preprocessing* yang digunakan pada penelitian ini:

- a. *Tokenizing*
Proses *tokenize* berfungsi untuk menghapus karakter-karakter tertentu seperti tanda baca, symbol dan karakter yang bukan berupa huruf serta memfilter berdasarkan panjang teks pada setiap keluhan.
- b. *Transform Case*
Pada tahap transform case, rapidminer akan mengubah kapitalisasi huruf menjadi lower case. Setelah melalui tahap ini, seluruh isi ulasan menjadi lower case.
- c. *Filter Token (by Length)*
Pada tahap ini dilakukan pemilihan token berdasarkan panjang karakter dengan ukuran panjang minimal 4 karakter (huruf) dan panjang maksimal 250 karakter (huruf).
- d. *Filter Stopword (by Dictionary)*
Tahap ini dilakukan untuk penghapusan kata-kata yang tidak berguna, dimana kata-kata yang tidak digunakan akan dihilangkan dari data.

Setelah semua tahapan *text preprocessing* dilakukan, hasil output merupakan *vektor* kata. Berikut merupakan hasil yang didapatkan setelah pengolahan *text preprocessing* :

Tabel 2 Vektor

Informasi Term Frequency Dokumen						
No	Kata	D1	D2	D3	D4	D5
1	<i>already</i>	0	0	1	0	0
2	<i>asap</i>	0	0	0	0	1
3	<i>assigned</i>	0	0	1	0	0
4	<i>available</i>	0	0	1	0	0
5	<i>because</i>	0	0	1	0	0
6	<i>buka</i>	0	0	0	0	1
7	<i>bulukumba</i>	0	1	0	0	0
8	<i>cabutan</i>	1	0	0	0	0
9	<i>coba</i>	0	0	0	0	1
10	<i>gangguan</i>	0	0	0	0	1
11	<i>getsomtresponse</i>	0	0	1	0	0
12	<i>gettechnicalordertask</i>	0	0	0	1	0
13	<i>handlefailure</i>	1	0	0	0	0
14	<i>info</i>	0	0	0	0	1
15	<i>instalntetask</i>	0	1	0	0	0

Setelah diperoleh hasil wordlist preprocessing dilakukan pembobotan kata menggunakan metode *Term Frequency Inverse Document Frequency (TF-IDF)*. Pembobotan itu sendiri merupakan proses merubah kata menjadi bentuk *vektor*, dimana tiap kata dihitung sebagai satu fitur. Dengan melakukan beberapa proses yaitu menghitung *Term Frequency (TF)*, *Document Frequency (DF)*, *Inverse Document Frequency (IDF)* dan mengalikan TF dengan IDF sebagai bentuk bobot dari kata suatu *keyword*. Dengan menggunakan rumus sebagai berikut :

$$IDF(\text{asap}) = \log \frac{N}{df(t)}$$

$$IDF(\text{asap}) = \log \frac{5}{1}$$

$$IDF(\text{asap}) = \log 5$$

$$IDF(\text{asap}) = 0,699$$

Tabel 3 TF IDF

Informasi Term Frequency Dokumen						
No	Kata	D1	D2	D3	D4	D5
1	<i>already</i>	0	0	0,699	0	0
2	<i>asap</i>	0	0	0	0	0,699
3	<i>assigned</i>	0	0	0,699	0	0
4	<i>available</i>	0	0	0,699	0	0
5	<i>because</i>	0	0	0,699	0	0
6	<i>buka</i>	0	0	0	0	0,699
7	<i>bulukumba</i>	0	0,699	0	0	0
8	<i>cabutan</i>	0,699	0	0	0	0
9	<i>coba</i>	0	0	0	0	0,699
10	<i>gangguan</i>	0	0	0	0	0,699
11	<i>getsomtresponse</i>	0	0	0,699	0	0
12	<i>gettechnicalordertask</i>	0	0	0	0	0,699
13	<i>handlefailure</i>	0,699	0	0	0	0
14	<i>info</i>	0	0	0	0	0,699

15	instalmentata sk	0	0, 69 9	0	0	0
----	---------------------	---	---------------	---	---	---

3.1. Hasil Penelitian

3.1.1 Eksperimen Tanpa Pemilihan Fitur

Tahap ini adalah proses penerapan tahap pertama yaitu pengujian model menggunakan *Software RapidMiner* terhadap algoritma *Support Vector Machine*

c / γ	0.01	0.03	0.06	0.1	0.2	0.3	0.6
0.0	78.24 %	80.43 %	80.62 %	81.65 %	85.90 %	87.19 %	88.15 %
0.1	78.24 %	78.24 %	78.24 %	78.24 %	78.24 %	80.04 %	80.43 %
0.3	78.24 %	78.24 %	78.30 %	80.43 %	80.55 %	80.62 %	81.65 %
0.9	78.24 %	80.43 %	80.62 %	81.26 %	85.26 %	86.99 %	87.64 %
1.0	78.24 %	80.43 %	80.62 %	81.65 %	85.90 %	87.19 %	88.15 %
1.5	78.24 %	80.62 %	81.52 %	85.00 %	87.57 %	88.93 %	90.02 %
2.0	79.78 %	80.68 %	83.39 %	86.87 %	88.99 %	90.09 %	90.15 %
2.5	80.36 %	80.88 %	85.45 %	87.51 %	89.96 %	90.21 %	90.08 %
3.0	80.43 %	81.65 %	86.48 %	88.03 %	90.21 %	90.47 %	90.47 %

tanpa pemilihan fitur. Eksperimen ini menggunakan 1553 data keluhan user yang telah disiapkan untuk diimplementasikan pada proses uji model kemudian diujikan pada algoritma *Support Vector Machine*. Pada penelitian ini menggunakan *cross validation* dengan *number of folds* 10. Model yang akan digunakan pada pengujian adalah model pembelajaran hasil dari training data latih. Karena klasifikasi pada penelitian ini bersifat *non linear* maka penelitian ini akan menggunakan kernel RBF dengan nilai parameter sebagai berikut $0 \leq c \leq 3.0$ dan $0.01 \leq \gamma \leq 10.0$:

Tabel 4 *Grid Search*

Setelah di uji menggunakan *grid search* mendapatkan nilai akurasi terbaik terdapat pada titik $c=3.0, \gamma=0.3$ mencapai nilai akurasi tertinggi 90.47%.

Tabel 5 *confusion matrix* tanpa pemilihan fitur

	true OSM	true UIM	true DLL	true SCC D	true AS AP	class precision
pre d. OS M	166	14	7	7	2	84.69 %

pre d. UIM	35	1195	33	34	4	91.85 %
pre d. DLL	4	3	3	0	0	30.00 %
pre d. SC CD	1	3	1	41	0	89.13 %
pre d. AS AP	0	0	0	0	0	0.00 %
clas s rec all	80.5 8%	98.3 5%	6.8 2%	50.0 0%	0.0 0%	

Dari *confusion matrix* pada Tabel 5 dapat diukur tingkat akurasi dari klasifikasi sebagai berikut:

$$Accuracy = \frac{TU+TO+TS+TA+TD}{TU+TO+TS+TA+TD+FU+FO+FS+FA+FD} \times 100$$

$$Accuracy = \frac{166+1195+3+41+0}{166+1195+3+41+0+14+7+7+2+35+33+34+4+4+3+3+1+3+1} \times 100$$

$$Accuracy = \frac{1405}{1553} \times 100$$

$$Accuracy = 0.9047 \times 100$$

$$Accuracy = 90.47$$

Hasil *accuracy* pengujian algoritma *support vector machine* pada laporan keluhan sistem NOSS tanpa menggunakan pemilihan fitur memperoleh hasil 90.47%. Untuk *class precision* prediksi aplikasi UIM memiliki nilai *class precision* prediksi tertinggi dari *class precision* yang lain sebanyak 91.85% dikarenakan untuk keluhan pada aplikasi UIM bulan february terhitung terbanyak dari aplikasi yang lain. Guna mengetahui *accuracy* terbaik pada klasifikasi ini dicoba dengan pemilihan fitur, maka dilakukan beberapa bentuk model pengujian dengan *ekstraksi feature* dengan jumlah *feature* yang berbeda.

3.1.2 Eksperimen dengan pemilihan Fitur

Pada pemilihan fitur dilakukan dengan 3 cara *ekstraksi feature* yaitu pemilihan *feature ekstraksi threshold, ranking selection threshold up, ranking selection threshold down*. Dengan mengelompokan fitur yang masuk kedalam 3 cara tersebut, berikut pengujian menggunakan pemilihan fitur:

- Ranking selection threshold up*
Merupakan *feature* yang mempunyai ranking tertinggi akan digunakan dan yang terendah akan dikesampingkan, berikut hasil percobaan pemilihan fitur *Ranking selection threshold up* dengan range *feature* 25-989. Mendapatkan hasil nilai akurasi terbaik terdapat pada

titik $c=3.0$, $\gamma=3.0$ mencapai nilai akurasi tertinggi 89.50% .

- b. *Ranking selection threshold down*
Ranking selection threshold down merupakan feature yang mempunyai ranking terendah akan digunakan dan yang tertinggi akan dikesampingkan , berikut hasil percobaan pemilihan fitur *Ranking selection threshold down* dengan range *feature 1-20*. Mendapatkan hasil nilai akurasi terbaik terdapat pada titik $c=2.5$, $\gamma=1.0$ mencapai nilai akurasi tertinggi 85.32%.
- c. *Feature ekstraksi threshold*
 Feature ekstraksi threshold merupakan titik dimana frekuensi kemunculan kata dan jumlah feature mulai konstan , berikut hasil percobaan pemilihan fitur *Ranking selection threshold down* dengan range 13-24 *feature* menggunakan *grid search*. Mendapatkan hasil nilai akurasi terbaik terdapat pada titik $c=1.0$, $\gamma=3.0$ mencapai nilai akurasi tertinggi

Hasil Pengujian	
Proses Pengujian	Accuracy
Tanpa Pemilihan Fitur	90.47%
Pemilihan Fitur	
- <i>Ranking selection threshold up</i>	89.50%
- <i>Ranking selection threshold down</i>	85.32%
- <i>Ekstrasi threshold</i>	80.36%

80.36%.

3.2. Pembahasan

Berdasarkan pengujian yang telah dilakukan menggunakan *RapidMiner* dengan dilakukan beberapa jenis pengujian yaitu pengujian tanpa pemilihan fitur dan pengujian menggunakan pemilihan fitur (*feature ekstraksi threshold, ranking selection threshold up* , *ranking selection threshold down*) dengan proses *grid search* dengan parameter C dan γ dengan nilai $C = 0.0 \leq C \leq 3.0$, $\gamma = 0.01 \leq \gamma \leq 10.0$, *cross validation* dengan *number of folds* 10 guna mencari akurasi terbaik , maka didapat hasil rekapitulasi keseluruhan dapat dilihat pada tabel berikut:

Tabel 6 Hasil Pengujian

Pada tabel 6 adalah hasil perbandingan akurasi dari pengujian tanpa pemilihan fitur dengan pengujian pemilihan fitur mendapatkan akurasi terbaik pada pengujian tanpa menggunakan pemilihan fitur sebesar 90.47% yang berarti *Excellent Classification* . Hal ini disebabkan pengujian dengan tanpa pemilihan fitur menggunakan seluruh fitur yang ada tanpa menyaring fitur yg akan digunakan , sedangkan pengujian menggunakan pemilihan fitur (*feature ekstraksi threshold, ranking selection threshold up* , *ranking selection threshold down*) hanya menguji fitur yang masuk kedalam kategori pada pemilihan fitur tersebut , maka mendapatkan hasil lebih rendah dibandingkan dengan pengujian tanpa pemilihan fitur.

4. Kesimpulan

Berdasarkan dari hasil analisa dan implementasi data keluhan terhadap sistem NOSS dengan menggunakan metode *Support Vector Machine* (SVM), maka dapat diambil beberapa kesimpulan diantaranya adalah :

- Algoritma *Support Vector Machine* berhasil diterapkan untuk melakukan klasifikasi data dalam kategori multiclass terhadap keluhan sistem NOSS yang dikategori berdasarkan aplikasi yang ada pada sistem NOSS dengan keluhan yang dilaporkan pada aplikasi *IT Care* , dengan menggunakan kernel RBF dan parameter yang digunakan adalah C dan γ .
- Penelitian ini berhasil mendapatkan nilai akurasi tertinggi sebesar 90.47% terdapat pada titik $C=3.0$ $\gamma=0.3$ dengan pengujian data tanpa *feature selection*, dan untuk data yang sudah dilakukan *feature selection* mencapai nilai akurasi tertinggi sebesar 89.50% pada titik $C=3.0$, $\gamma=3.0$ pada pemilihan fitur *Ranking selection threshold up* .

Referensi

- Agustina, N. (2018). Evaluasi Penggunaan Sistem Informasi ERP Dengan Metode Pieces Framework. *JURNAL INFORMATIKA*, 1.
- Berry, M. W., & Kogan, J. (2010). *Text Mining: Applications and Theory*. United Kingdom.
- Chou, J. S., Cheng, M. Y., Wu, Y. W., & Pham, A. D. (2014). Optimizing

- parameters of support vector machine using fast messy. *Expert Systems with Applications*.
- Dawson, C. (2009). *Projects in Computing and Information Systems A Student's Guide*. England: Addison-Wesley.
- Dewi, R. N. (2017). Model TEXT MINING UNTUK IDENTIFIKASI KELUHAN PELANGGAN PRODUK PERUSAHAAN PERANGKAT LUNAK. *Seminar Nasional Riset dan Inovasi Teknologi*.
- Fatta, H. A. (2007). KOMPUTER - SISTEM INFORMASI. Dalam A. H. Triyuliyana, *Analisis dan Perancangan Sistem Informasi untuk Keunggulan Bersaing Perusahaan dan Organisasi Modern* (hal. 2). DIY: Penerbit Andi.
- Help Desk World. (2002). What is a Help Desk? *The Directory of Help Desk Software and Information*. Dipetik 04 20, 2020, dari <http://www.help-desk-world.com/help-desk.htm>
- Indriati, & Ridok, A. (2016). SENTIMENT ANALYSIS FOR REVIEW MOBILE APPLICATIONS USING NEIGHBOR METHOD WEIGHTED K-NEAREST NEIGHBOR (NWKNN). *Journal of Environmental Engineering & Sustainable Technology*.
- Kurniawan, B., Effendi, S., & Sitompul, O. S. (2012). Klasifikasi Konten Berita Dengan Metode. *JURNAL DUNIA TEKNOLOGI INFORMASI Vol. 1, 1*.
- MA'ARIF, A. A. (2015). Penerapan Algoritma TF-IDF Untuk Pencarian Karya Ilmiah. *Dokumen Karya Ilmiah | Tugas Akhir | Program Studi Teknik Informatika - S1 | Fakultas Ilmu Komputer | Universitas Dian Nuswantoro Semarang |*.
- PT.Telkom Indonesia. (2020). https://www.telkom.co.id/sites/about-telkom/id_ID/page/profil-dan-riwayat-singkat. Dipetik Mei 20, 2020, dari https://www.telkom.co.id/sites/about-telkom/id_ID/page/profil-dan-riwayat-singkat
- Putra, A. A. (2015). Implementasi Text Summarization Menggunakan Metode Vector Space Model Pada Artikel Berita Berbahasa Indonesia. Diambil kembali dari <https://elib.unikom.ac.id/gdl.php?mo> d=browse&op=read&id=jbptunikomp-gdl-agungaulia-33595
- Redaksi WE Online. (2017, Mei 16). *Telkom Terapkan Sistem ERP, Apa itu?* Dipetik Mei 04, 2020, dari www.wartaekonomi.co.id: <https://www.wartaekonomi.co.id/read141216/telkom-terapkan-sistem-erp-apa-itu>
- Wahjono, I. S. (2010). *Manajemen Pemasaran Bank*. Yogyakarta: Graha Ilmu.
- Xiang, Z., Schwart, Z., Jr. c, J. H., & Uysal, M. (2015). What can big data and text analytics tell us about hotel guest experience and satisfaction? *International Journal of Hospitality Management*. Dipetik 04 21, 2020, dari <https://www.journals.elsevier.com/international-journal-of-hospitality-management>