PENERAPAN ALGORITMA *SUPPORT VECTOR MACHINE* (SVM) DENGAN TF-IDF N-GRAM UNTUK *TEXT CLASSIFICATION*

Nur Arifin [1], Ultach Enri [2], Nina Sulistiyowati [3]
Program Studi Teknik Informatika, Universitas Singaperbangsa Karawang [123]
nur.arifin17013@student.unsika.ac.id[1], ultach@staff.unsika.ac.id[2], nina.sulistio@unsika.ac.id[3]

Submitted July 12, 2021; Revised November 26, 2021; Accepted November 26, 2021

Abstrak

Syntax Jurnal Informatika merupakan salah satu sistem informasi yang berisikan sekumpulan artikel ilmiah yang dikelola oleh Program Studi Teknik Informatika, Universitas Singaperbangsa Karawang. Saat ini Syntax Jurnal Informatika tidak memiliki fitur untuk kategorisasi artikel ilmiah berdasarkan fokus dan ruang lingkupnya. Penelitian ini dilakukan untuk mengklasifikasi artikel ilmiah ke dalam kategori sesuai dengan fokus dan ruang lingkup yang terdapat pada laman Syntax Jurnal Informatika secara otomatis dengan memanfaatkan proses *text mining*. *Text mining* merupakan proses yang bertujuan untuk mendapatkan informasi penting dari teks. Metodologi penelitian yang digunakan adalah *Knowledge Discovery Database* (KDD) dengan tahapan *data selection, preprocessing, transformation, modeling* dan *evaluation*. Penelitian ini akan membandingkan klasifikasi berdasarkan judul pada artikel. Adapun algoritma yang digunakan adalah *Support Vector Machine* (SVM) dengan menggunakan empat *kernel* SVM, diantaranya adalah *kernel linear, kernel polynomial, kernel sigmoid* dan *kernel RBF*. Pembagian data menggunakan *traintestsplit* dibagi menjadi empat skenario yaitu 60:40, 70:30, 80:30 dan 90:10. Hasil penelitian setelah dilakukan pengujian terhadap model diukur dengan nilai *Accuracy, Precision, Recall* dan *F-measure*. Hasil terbaik adalah *accuracy* sebesar 70%, *precision* sebesar 75%, *recall* sebesar 69% dan *f-measure* sebesar 71% pada skenario perbandingan 90:10 dan *kernel linear*.
Kata Kunci : *Support Vector Machine, Text Classification, TF-IDF, N-Gram*

Abstract

Syntax Journal of Informatics is an information system that contains a collection of scientific articles managed by the Informatics Study Program of Singaperbangsa Karawang University. Currently, Syntax Journal of Informatics does not have a feature for categorizing scientific articles based on their focus and scope. The research is conducted to classify scientific articles into categories according to focus and scope contained on Syntax Journal of Informatics' page automatically by utilizing the text mining process. Text mining is a process that aims to get important information from the text. The method used in the research is Knowledge Discovery in Database (KDD) with stages of data selection, preprocessing, transformation, modeling and evaluation. This study will compare the classifications based on the title of the article. The algorithm used is the Support Vector Machine (SVM) using four SVM kernels, including the linear kernel, polynomial kernel, sigmoid kernel and RBF kernel. Data are divided into four scenarios by using traintestsplit, namely 60:40, 70:30, 80:30 and 90:10. The results of the study after testing the model are measured by of Accuracy, Precision, Recall and F-measure. The best results are accuracy of 70%, precision of 75%, recall of 69% and f-measure of 71% in the 90:10 comparison scenario and linear kernel.
Keywords : *Support Vector Machine, Text Classification, TF-IDF, N-Gram*

1. PENDAHULUAN

Syntax Jurnal Informatika merupakan sebuah sistem informasi yang berisikan sekumpulan artikel ilmiah dan dikelola oleh Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Singaperbangsa Karawang. Syntax Jurnal Informatika saat ini sudah memuat 95

artikel ilmiah dan akan terus terjadi pertambahan. Berbagai macam hasil penelitian yang sudah diterbitkan pada laman Syntax Jurnal Informatika terdiri dari campuran kategori fokus dan ruang lingkup. Penelitian ini dilakukan untuk mengklasifikasi judul dari artikel ilmiah tersebut ke dalam kategori sesuai dengan fokus dan ruang lingkupnya secara otomatis dengan memanfaatkan proses *text mining*.

Text mining merupakan salah satu ilmu pencabangan *data mining*, *Text mining* merupakan sebuah proses yang memiliki tujuan untuk mendapatkan informasi penting dari suatu teks. Informasi penting tersebut umumnya dapat diperoleh dengan memperhatikan pola dan tren yang dipelajari dari pola statistik. Pola *text mining* tersebut terdapat pembobotan kata yang memiliki tujuan yaitu memberikan nilai atau bobot pada term yang ada pada dokumen tersebut [1].

Dalam penelitian lain menyebutkan bahwa algoritma *Support Vector Machine* (SVM) memiliki nilai akurasi terbaik [2]. Algoritma *Support Vector Machine* (SVM) adalah salah satu metode regresi atau klasifikasi data berdasarkan data sebelumnya dan modelnya dilakukan supervisi terlebih dahulu [3]. *Support Vector Machine* (SVM) merupakan sebuah metode yang membandingkan suatu seleksi parameter standar nilai diskrit yang disebut kandidat set [4]. *Support Vector Machine* (SVM) umumnya digunakan untuk *binary classifier* yang melakukan klasifikasi dengan membagi data menjadi 2 *class* yaitu dengan menggunakan *hyperplane*. Untuk memaksimalkan ruang antar kelas, ruang *input* asli diubah menjadi ruang yang berdimensi sangat tinggi disebut ruang fitur. Kernel digunakan untuk mentransformasi data ke ruang dimensi yang lebih tinggi, dan disebut ruang *kernel*, berguna untuk memisahkan data secara linear [5].

Data yang diambil berupa teks yaitu data judul dari situs Syntax Jurnal Informatika dan diolah menggunakan metode *Knowledge Discovery Database* (KDD). *Knowledge Discovery Database* (KDD) merupakan salah satu metode yang sering digunakan dalam *text mining* [6]. Tahapan *Knowledge Discovery Database* (KDD) yang pertama adalah *data selection*, kemudian *preprocessing*, dan *transformation* untuk meningkatkan hasil klasifikasi yang lebih akurat. Setelah dilakukan *preprocessing* data akan dilatih dan diuji dengan beberapa *kernel* yang ada pada algoritma *Support Vector Machine* (SVM) lalu tahapan ter akhir adalah *evaluation*.

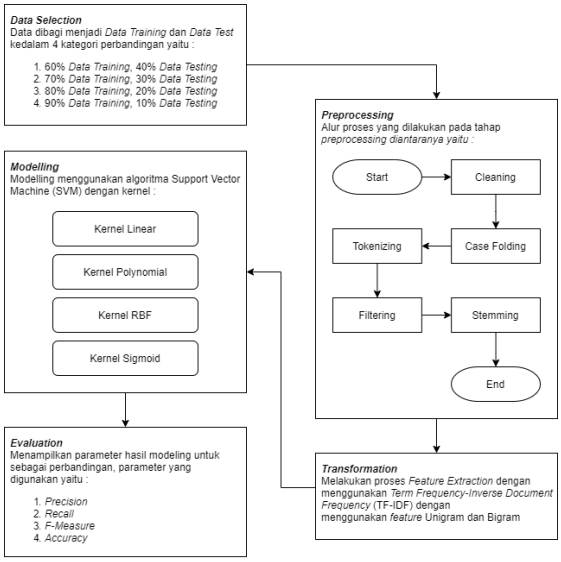
Pada tahapan *transformasi*, N-Gram diterapkan dengan melakukan modifikasi pemisahan atau pemecahan berdasarkan kata. Referensi [7] menjelaskan bahwa N-Gram adalah suatu potongan n-karakter yang didapatkan dari suatu *string*. Metode N-Gram ini biasanya diaplikasikan untuk pembangkitan kata atau karakter. Untuk memudahkan perhitungan dan menghindari kesalahan pada ekstraksi *Term Frequency Inverse Document Frequency* (TF-IDF), maka perlu digunakannya N-Gram. Penggunaan N-Gram juga dapat memberikan keuntungan karena hasil yang diperoleh menjadi lebih akurat dan efektif [8].

Berdasarkan penelitian Liani yang berjudul "Analisis Perbandingan Kernel Algoritma Support Vector Machine dalam Mengklasifikasikan Skripsi Teknik Informatika Berdasarkan Abstrak" didapatkan menggunakan algoritma *Support Vector Machine* (SVM) dengan *default parameter* dan 4 *kernel* yaitu *Linear, Polynomial, Radial Basis Function* (RBF) dan *Sigmoid* menghasilkan akurasi yang cukup tinggi [9], sehingga pada penelitian ini akan menggunakan *kernel* yang sama yaitu *kernel Linear, Sigmoid, Polynomial* dan *Radial Basis Function*

(RBF). Pengujian pada penelitian ini akan dilakukan untuk mencari nilai *recall*, *precision*, dan *f-measure* untuk menghitung nilai akurasi ketepatan klasifikasi kategori artikel ilmiah terhadap fokus dan ruang lingkupnya.

2. METODE PENELITIAN

Metodologi yang digunakan adalah *Knowledge Discovery in Database* (KDD). *Knowledge Discovery in Database* (KDD) merupakan salah satu metode yang sering digunakan dalam *text mining*. *Knowledge Discovery in Database* (KDD) merupakan proses kegiatan yang meliputi pengumpulan data, pemakaian data, historis untuk menemukan keteraturan, pola atau hubungan dalam set data yang memiliki ukuran besar [6]. Tahapannya dapat dilihat pada Gambar 1.



Gambar 1. Rancangan Penelitian

Dari Gambar 1, diketahui tahap penelitian yang dilakukan adalah sebagai berikut :

Data Selection

Penelitian ini menggunakan set data berupa judul artikel ilmiah yang terdapat pada laman Syntax Jurnal Informatika sebanyak 95 data. Pada tahap *data selection*, judul artikel ilmiah yang sudah dikumpulkan dibagi menjadi 2 set data, yaitu set *data training* dan set *data testing*. Pembagian data menggunakan *Split Data* dengan rasio perbandingan *data training* dan *data testing* sebesar 60:40, 70:30, 80:20 dan 90:10.

Preprocesing

Pengolahan data pada tahapan *preprocessing* yang diharapkan dapat meningkatkan hasil pengukuran yang baik pada tahap *evaluation* nanti. Adapun tahapan *preprocessing* sebagai berikut:

1. *Data Cleaning*

Pada tahap *data cleaning*, setiap kalimat yang mengandung *noise* yang berupa tautan situs, simbol seperti !,@,#,$,%,^,&,*,(,),/,”,[,. dan simbol lainnya atau angka akan dihapus.

	Judul	text_cleaning
0	Analisis Pemanfaatan Cloud Computing Berbasis ...	Analisis Pemanfaatan Cloud Computing Berbasis ...
1	Penerapan Sistem Viewboard Penilaian Pembimbin...	Penerapan Sistem Viewboard Penilaian Pembimbin...
2	Data Mining Pengelompokan Bidang Keahlian Maha...	Data Mining Pengelompokan Bidang Keahlian Maha...
3	Analisis dan Prediksi Kinerja Mahasiswa Menggu...	Analisis dan Prediksi Kinerja Mahasiswa Menggu...
4	Perancangan Program Pengolahan dan Pengelolaan...	Perancangan Program Pengolahan dan Pengelolaan...

Gambar 2. Hasil Proses *Data Cleaning*

2. *Case Folding*

Pada tahap *case folding*, setiap huruf diubah menjadi huruf kecil, karena model yang dibangun bersifat *case sensitif*, sedangkan penggunaan huruf kapital bisa jadi tidak konsisten pada semua kata.

	text_cleaning	case_folding
0	Analisis Pemanfaatan Cloud Computing Berbasis ...	analisis pemanfaatan cloud computing berbasis ...
1	Penerapan Sistem Viewboard Penilaian Pembimbin...	penerapan sistem viewboard penilaian pembimbin...
2	Data Mining Pengelompokan Bidang Keahlian Maha...	data mining pengelompokan bidang keahlian maha...
3	Analisis dan Prediksi Kinerja Mahasiswa Menggu...	analisis dan prediksi kinerja mahasiswa menggu...
4	Perancangan Program Pengolahan dan Pengelolaan...	perancangan program pengolahan dan pengelolaan...

Gambar 3. Hasil Proses *Case Folding*

3. *Tokenizing*

Tahapan *tokenizing* dilakukan untuk memecah kalimat menjadi kata berdasarkan spasi. *Tokenizing* akan digunakan untuk tahap selanjutnya yaitu *filtering* dan *stemming*.

	case_folding	text_tokenization
0	analisis pemanfaatan cloud computing berbasis ...	[analisis, pemanfaatan, cloud, computing, berb...
1	penerapan sistem viewboard penilaian pembimbin...	[penerapan, sistem, viewboard, penilaian, pemb...
2	data mining pengelompokan bidang keahlian maha...	[data, mining, pengelompokan, bidang, keahlian...
3	analisis dan prediksi kinerja mahasiswa menggu...	[analisis, dan, prediksi, kinerja, mahasiswa, ...
4	perancangan program pengolahan dan pengelolaan...	[perancangan, program, pengolahan, dan, pengel...

Gambar 4. Hasil Proses *Tokenizing*

4. *Filtering*

Filtering dilakukan untuk menghapus kata sambung seperti "dan", "atau", "yang" atau kata-kata yang kurang berpengaruh pada penelitian.

	text_tokenization	text_stopwordremove
0	[analisis, pemanfaatan, cloud, computing, berb...	[analisis, pemanfaatan, cloud, computing, berb...
1	[penerapan, sistem, viewboard, penilaian, pemb...	[penerapan, sistem, viewboard, penilaian, pemb...
2	[data, mining, pengelompokan, bidang, keahlian...	[data, mining, pengelompokan, bidang, keahlian...
3	[analisis, dan, prediksi, kinerja, mahasiswa, ...	[analisis, prediksi, kinerja, mahasiswa, tekni...
4	[perancangan, program, pengolahan, dan, pengel...	[perancangan, program, pengolahan, pengelolaan...

Gambar 5. Hasil Proses *Filtering*

5. *Stemming*

Stemming digunakan untuk mengubah seluruh kata menjadi kata dasarnya, salah satunya adalah menghapus imbuhan.

	text_stopwordremove	text_stemmed
0	[analisis, pemanfaatan, cloud, computing, berb...	[analisis, manfaat, cloud, computing, bas, sof...
1	[penerapan, sistem, viewboard, penilaian, pemb...	[terap, sistem, viewboard, nilai, bimbing, bas...
2	[data, mining, pengelompokan, bidang, keahlian...	[data, mining, kelompok, bidang, ahli, mahasis...
3	[analisis, prediksi, kinerja, mahasiswa, tekni...	[analisis, prediksi, kerja, mahasiswa, teknik,...
4	[perancangan, program, pengolahan, pengelolaan...	[ancang, program, olah, kelola, data, administ...

Gambar 6. Hasil Proses *Stemming*

Transformation

Proses *transformasi* bertujuan untuk mendapatkan representasi dokumen sesuai dengan yang diharapkan. Proses ini dilakukan untuk mengubah data menjadi vektor agar data mudah untuk dilakukan proses *data mining*. Menurut Fitri pada Referensi [5], Pembobotan Term *Frequency Inverse Document Frequency* (TF-IDF) adalah metode yang umumnya dipakai untuk menentukan hubungan kata (*term*) terhadap dokumen atau kalimat dengan memberikan bobot atau nilai pada masing-masing kata. Metode *Term Frequency Inverse Document Frequency* (TF-IDF) menggabungkan konsep *Frequency Inverse* sebuah kata di dalam sebuah dokumen *dan Inverse Document Frequency* yang mengandung kata tersebut. Perhitungan bobot menggunakan *Term Frequency Inverse Document Frequency* (TF-IDF) yaitu dilakukan perhitungan terlebih dahulu nilai TF per kata dengan bobot masing-masing kata. Adapun perhitungan *Term Frequency Inverse Document Frequency* (TF-IDF) seperti yang dijelaskan pada persamaan berikut:

1. Nilai *Term Frequency* (TF) diperoleh dari nilai frekuensi kemunculan fitur *t* pada dokumen *d.*

$$TF_t = (t, d) \quad (1)$$

2. Nilai *Inverse Document Frequency* diperoleh dari logaritma banyaknya dokumen *n* dibagi dokumen *df* yang mengandung fitur t.

$$IDF_t = \log \frac{n}{df(t)} + 1 \quad (2)$$

3. Nilai *Term Frequency Inverse Document Frequency* (W_t) didapatkan dengan mengalihkan nilai *TF* dengan *IDF*.

$$W_t = TF_t \times IDF_t \quad (3)$$

Pada proses *transformasi*, fitur N-Gram digunakan dengan menerapkan modifikasi pemisahan atau pemecahan kalimat menjadi kata. Referensi [8] menjelaskan bahwa N-Gram adalah suatu potongan n-karakter yang didapatkan dari suatu *string*. Metode N-Gram ini biasanya diaplikasikan untuk pembangkitan kata atau karakter [7].

Beberapa kali pengujian dilakukan untuk mendapatkan akurasi terbaik, diantaranya adalah pengujian yang dilakukan untuk mendapatkan kata yang digunakan pada proses *transformation* dengan menentukan jumlah minimal dokumen yang memiliki kata tersebut. Nilai minimal tersebut diuji dengan 3 pengujian yaitu 1, 2 dan 3 dengan menghasilkan akurasi seperti yang ditunjukkan pada Tabel 1.

Tabel 1. Perbandingan Jumlah Minimal Kata terhadap Akurasi

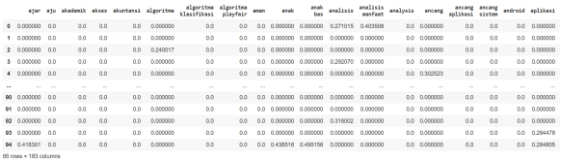
Minimal Kata	Akurasi
1	20%
2	60%
3	30%

Selain itu dilakukan pula pengujian menggunakan fitur N-Gram pada proses *transformation*. Perbandingan perbedaan akurasi dengan menggunakan fitur N-Gram dengan *Unigram*, *Unigram* + *Bigram*, dan *Bigram* pada Skenario dan *kernel* terbaik dipaparkan pada Tabel 2.

Tabel 2. Perbandingan Fitur N-Gram terhadap Akurasi

Fitur N-Gram	Akurasi
Unigram	60%
Unigram + Bigram	70%
Bigram	60%

Dari percobaan tersebut menghasilkan kesimpulan yaitu. Dan menerapkan fitur *Unigram* dan *Bigram* mampu meningkatkan nilai akurasi. Hasil dari dilakukannya proses TF-IDF dapat dilihat pada Gambar 7.



Gambar 7. Hasil proses *transformation* dengan TF-IDF dan fitur N-Gram

Modeling

Pada tahap *modeling*, data terpilih dilakukan proses untuk didapatkan informasi yang menarik dengan menggunakan algoritma tertentu. Pemilihan algoritma yang tepat sangat menentukan tujuan dan proses dari metodologi penelitian secara keseluruhan. Algoritma *Support Vector Machine* (SVM) masih memiliki kekurangan, secara teoritis algoritma *Support Vector Machine* (SVM) dikembangkan untuk masalah klasifikasi data dengan dua *class*, untuk mengatasi hal ini perlu digunakan teknik tambahan yang disebut dengan Kernel. Adapun *kernel* yang digunakan dalam penelitian ini yaitu *kernel Linear*, *Polynomial*, *RBF* dan *Sigmoid*. Berikut persamaan empat *kernel* SVM :

1. *Linear*

Fungsi kernel Linear merupakan fungsi kernel yang paling sederhana yaitu perkalian titik dua vektor.

$$K(x_i, x_j) = X_i^T X_j \quad (4)$$

2. *Polynomial*

Fungsi *kernel Polynomial* merupakan fungsi *kernel* yang memiliki derajat d, dimana d dan r merupakan parameter yang didefinisikan.

$$K(x_i, x_j) = (\gamma . X_i^T X_j + r)^d, \ \gamma > 0 \quad (5)$$

3. *Radial Basis Function (RBF)*

Fungsi *kernel Radial Basis Function* (RBF) biasa disebut juga sebagai fungsi *kernel Gaussian*. Dimana γ merupakan parameter untuk mengatur jarak.

$$K(x_i, x_j) = exp(-\gamma \left| X_i^T X_j \right|^2), \ \gamma > 0 \quad (6)$$

4. *Sigmoid*

Pada fungsi *kernel Sigmoid*, dimana *tanh* $(a) = 2\sigma(a) - 1$ dan $(a) = \frac{1}{(1+\exp(a))}$.

$$K(x_i, x_j) = \tanh(\gamma . X_i^T X_j + r) \quad (7)$$

Kernel dibutuhkan untuk membuat dimensi baru sehingga dapat memberi batas dengan membuat *hyperplane*-nya, *Hyperplane* yang baik merupakan *hyperplane* yang terletak di tengah-tengah antara dua objek dari dua kelas atau dengan kata lain ekuivalen dengan memaksimalkan margin atau jarak antara kedua sel objek dari kelas yang berbeda. Untuk melakukan klasifikasi dengan algoritma *Support Vector Machine* (SVM) terdapat beberapa tahapan diantaranya yaitu :

1. Menentukan *hyperplane* atau garis pembatas antara dua *support vector*.
2. Menentukan margin atau garis jarak antara *support vector* dan *hyperplane*.

3. Pemetaan *support vector* ke dalam suatu kelas dalam *class* dimensi yang sama.

Evaluation

Evaluation merupakan tahap akhir dari proses *Knowledge Discovery in Database* (KDD), evaluasi ini dilakukan untuk menampilkan pengetahuan agar mudah dibaca dan dipahami maka diperlukan penyajian yang menarik. Hasil penelitian yang dilakukan perlu ada pengujian dan evaluasi untuk mengetahui tingkat akurasi dan ketepatan dari hasil klasifikasi. Metode pengujian *data mining* yang paling banyak digunakan adalah mencari nilai *precision*, *recall*, *f-measure* dan *accuracy* [10] Nilai-nilai yang dicari pada tahap evaluasi diantaranya adalah :

1. *Precision*

Precision adalah hasil perbandingan dari jumlah data bernilai positif dengan hasil jumlah data benar bernilai positif dan data salah bernilai positif [1].

$$Precision = \frac{TP}{TP+FP} \quad (8)$$

2. *Recall*

Recall ialah hasil dari perbandingan nilai data benar bernilai positif dengan hasil jumlah data benar yang bernilai positif dan data salah bernilai negatif [1]

$$Recall = \frac{TP}{TP+FN} \quad (9)$$

3. *F-measure*

F-measure adalah parameter ukuran keberhasilan *retrieval* yang menggabungkan *precision* dan *recall*. Nilai tersebut didapat dari perhitungan perkalian nilai *precision* dan *recall* yang kemudian dibagi dengan hasil penjumlahan *precision* dan *recall* dan dikalikan dua [1]

$$F\text{-}Measure = \frac{Precision * Recall}{Precision+Recall} \quad (10)$$

4. *Accuracy*

Accuracy digunakan untuk mengukur akurasi dari masing-masing *kernel*. Dimana semakin besar nilai akurasi maka *kernel* yang digunakan semakin bagus [1]

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (11)$$

3. HASIL DAN PEMBAHASAN

Hasil yang diperoleh dari seluruh pengujian akan dibandingkan untuk mengetahui model mana yang terbaik dalam proses pengklasifikasian. Pada Tabel 3 dipaparkan hasil perbandingan evaluasi dari setiap skenario.

Tabel 3. Hasil Perbandingan Evaluasi Model

Skenario/ Pembagian	*Kernel*	*Accuracy*	*Precision*	*Recall*	*F-Measure*
1 / 60:40	*Linear*	39%	33%	34%	33%
	Sigmod	34%	26%	21%	20%
	Polynomial	11%	18%	15%	10%
	RBF	31%	2%	8%	4%
2 / 70:30	*Linear*	55%	51%	52%	50%
	Sigmod	34%	22%	21%	20%
	Polynomial	14%	12%	13%	8%
	RBF	31%	2%	8%	4%
3 / 80:20	*Linear*	53%	44%	43%	42%
	Sigmod	32%	16%	20%	17%
	Polynomial	21%	6%	14%	6%
	RBF	32%	3%	10%	5%
4 / 90:10	*Linear*	70%	75%	69%	71%
	Sigmod	40%	34%	38%	35%
	Polynomial	14%	12%	13%	8%
	RBF	20%	3%	14%	5%

Dari Tabel 3 dapat diketahui hasil akurasi menggunakan Skenario 4 dengan pembagian 90:10, *kernel linear* menghasilkan nilai akurasi paling tinggi sebesar 70%, *precision* sebesar 75%, *recall* sebesar 69% dan *f-measure* sebesar 71%. Sedangkan untuk nilai akurasi terkecil terdapat pada proses dengan Skenario 1, perbandingan 60:40 dan *kernel polynomial* sebesar 11%, *precision* sebesar 18%, *recall* sebesar 15% dan *f-measure* sebesar 10%. Adapun contoh hasil prediksi dari klasifikasi menggunakan Skenario 4 dengan pembagian 90:10 dan *kernel linear* dapat dilihat pada Gambar 8.

	Judul	actual	predict
0	Data Mining Pengelompokan Bidang Keahlian Maha...	Data Mining	Data Mining
1	Penerapan Metode SAW Pemilihan Siswa/i Berpres...	Neuro Fuzzy	Kecerdasan Buatan
2	KOMPARASI ALGORITMA KLASIFIKASI NAIVE BAYES DA...	Machine Learning	Machine Learning
3	PERANCANGAN E-COMMERCE ACCESSORIES HANDPHONE B...	E-Commerce	E-Commerce
4	Alat Pendeteksi Asap Rokok pada Ruangan Menggu...	Embeded Systems	Embeded Systems
5	Analisa Pola Penjualan Obat Menggunakan Algori...	Kecerdasan Buatan	Rekayasa Perangkat Lunak
6	Sistem Pendukung Keputusan Penanggulangan Hama...	Kecerdasan Buatan	Kecerdasan Buatan
7	Sistem Informasi Pengontrolan Persediaan Baran...	Rekayasa Perangkat Lunak	Rekayasa Perangkat Lunak
8	Optimasi Fuzzy C-Means Clustering Untuk Data B...	Neuro Fuzzy	Neuro Fuzzy
9	E-Kuesioner Pengukuran Kepuasan Pelanggan deng...	Rekayasa Perangkat Lunak	E-Government

Gambar 8. Hasil Prediksi Model yang Menghasilkan Akurasi Terbaik

Dari contoh hasil prediksi yang ditampilkan pada Gambar 8, terdapat beberapa Judul mendapatkan hasil prediksi yang akurat, namun tentu saja masih terdapat beberapa perbedaan antara kategori yang diharapkan atau *actual* dengan kategori yang diprediksikan oleh model atau *predict*.

4. SIMPULAN

Berdasarkan penelitian yang telah dilakukan dapat ditarik beberapa kesimpulan diantaranya yaitu :

1. *Text mining* dan algoritma *Support Vector Machine* (SVM) dapat digunakan untuk proses klasifikasi pada artikel ilmiah Syntax Jurnal Informatika,
2. Penentuan jumlah minimal kata yang diproses dan fitur N-Gram berupa *Unigram* dan *Bigram* pada tahapan *transformation* dengan *Term Frequency Inverse Document Frequency* (TF-IDF), dinilai mampu meningkatkan nilai akurasi pada proses klasifikasi artikel ilmiah Syntax Jurnal Informatika. Nilai akurasi semula yaitu 30% dengan menentukan jumlah minimal kata yang diproses naik menjadi 60% dan dengan menerapkan fitur *Unigram* dan *Bigram* nilai akurasi berubah menjadi 70%,
3. Data yang diolah yaitu data Judul pada Syntax Jurnal Informatika menghasilkan nilai performa terbaik terdapat pada *kernel linear* dengan dan Skenario pembagian data 4 yaitu 90:10 serta dengan menerapkan modifikasi pada pembobotan TF-IDF yaitu menerapkan TF-IDF *Unigram* dan TF-IDF *Bigram*. Kernel linear memiliki nilai akurasi terbaik sebesar 70%, *recall* sebesar 75%, precision sebesar 69% dan *f-measure* sebesar 71%. Untuk *kernel sigmoid* memiliki nilai akurasi terbaik sebesar 34%, precision sebesar 26%, *recall* sebesar 21% dan *f-measure* sebesar 20%. Untuk *kernel polynomial* memiliki nilai akurasi terbaik sebesar 32%, precision sebesar 3%, *recall* sebesar 10% dan f-*measure* sebesar 5%. Dan untuk *kernel* RBF memiliki nilai akurasi terbaik sebesar 32%, precision sebesar 3%, *recall* sebesar 10% dan *f-measure* sebesar 5%,
4. Dari hasil pengujian terbukti *kernel* pada algoritma *Support Vector Machine* (SVM) yang memiliki nilai performa terbaik antara keempatnya adalah *kernel linear* dengan skenario 4, pembagian 90:10 yang memiliki nilai *accuracy* sebesar 70%, *recall* sebesar 75%, precision sebesar 69% dan *f-measure* sebesar 71%.

DAFTAR PUSTAKA

[1] W. Anggraini, M. Utami, J. Berlianty, and E. Sellya, "*Klasifikasi Sentimen Masyarakat Terhadap Kebijakan Kartu Prakerja di Indonesia*," *Faktor Exacta*, vol. 13, no. 4, pp. 255–261, 2021, doi: 10.30998/faktorexacta.v13i4.7964.

[2] A. Deolika, K. Kusrini, and E. T. Luthfi, "*Analisis Pembobotan Kata Pada Klasifikasi Text Mining*," *Jurnal Teknologi Informasi*, vol. 3, no. 2, p. 179, 2019, doi: 10.36294/jurti.v3i2.1077.

[3] V. Gupta and G. S. Lehal, "*A survey of text mining techniques and applications*," *Journal of Emerging Technologies in Web Intelligence*,

vol. 1, no. 1, pp. 60–76, 2009, doi: 10.4304/jetwi.1.1.60-76.

[4] J. Han, M. Kamber, and J. Pei, *Data Mining Concepts and Techniques*. 2012. doi: 10.1016/b978-0-12-381479-1.00001-0.

[5] B. Herwijayanti, D. E. Ratnawati, and L. Muflikhah, "*Klasifikasi Berita Online dengan menggunakan Pembobotan TF-IDF dan Cosine Similarity*," *Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 2, no. 1, pp. 306–312, 2018.

[6] A. Liani, U. Enri, and Y. Umaidah, "*Analisis Perbandingan Kernel Algoritma Support Vector Machine dalam Mengklasifikasikan Skripsi Teknik Informatika berdasarkan Abstrak*," *JOINS (Journal of Information System)*, vol. 5, no. 2, pp. 240–249, 2020, doi: 10.33633/joins.v5i2.3715.

[7] S. Mardianti, M. Zidny, and I. Hidayatulloh, "*Ekstraksi tf-Idf n-gram dari komentar pelanggan produk smartphone pada website e-commerce*," *Seminar Nasional Teknologi Informasi dan Multimedia*, vol. 6, no. , pp. 79–84, 2018.

[8] A. A. Prasanti, M. A. Fauzi, and M. T. Furqon, "*Klasifikasi Teks Pengaduan Pada Sambat Online Menggunakan Metode N- Gram dan Neighbor Weighted K-Nearest Neighbor (NW-KNN)*," *Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 2, no. 2, pp. 594–601, 2018.

[9] I. Riadi, R. Umar, and F. D. Aini, "*Analisis Perbandingan Detection Traffic Anomaly Dengan Metode Naive Bayes Dan Support Vector Machine (Svm)*," *ILKOM Jurnal Ilmiah*, vol. 11, no. 1, pp. 17–24, 2019, doi: 10.33096/ilkom.v11i1.361.17-24.

[10] O. Rahman, G. Abdillah, and A. Komarudin, "*Klasifikasi Ujaran Kebencian pada Media Sosial Twitter Menggunakan Support Vector Machine*," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 1, pp. 17–23, 2021, doi: 10.29207/resti.v5i1.2700.