Terakreditasi SINTA Peringkat 2
Surat Keputusan Dirjen Penguatan Riset dan Pengembangan Ristek Dikti No. 10/E/KPT/2019
masa berlaku mulai Vol. 1 No. 1 tahun 2017 s.d Vol. 5 No. 3 tahun 2021

Terbit online pada laman web jurnal: **http://jurnal.iaii.or.id**

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 4 No. 5 (2020) 915 - 922 ISSN Media Elektronik: 2580-0760

Klasifikasi Jenis Pantun dengan Metode *Support Vector Machines* (SVM)

Helena Nurramdhani Irmanda[1], Ria Astriratma[2]
[1,2]Jurusan Sistem Informasi, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional Veteran Jakarta
[1]helenairmanda@upnvj.ac.id, [2]astriratma@upnvj.ac.id

Abstract

This study aims to create a model for categorizing pantun types and analyze the accuracy of support vector machines (SVM). The first stage is collecting pantun that have been labeled with pantun category. The pantun categories consist of pantun for children, pantun for young people, and pantun for elder. After collecting data, the next stage is pre-processing. This pre-processing stage makes data ready to be processed on the extraction stage. The pre-processing stage consists of text segmentation, case folding, tokenization, stop word removal, and stemming. The feature extraction stage is intended to analyze potential information and represent terms as a vector. Separating training data and testing data is necessary to be conducted before the classification process. Then the classification process is done by using multiclass SVM. The results of the classification are evaluated to obtain accuracy and will be analyzed whether the classification model is proper to be used. The results showed that SVM classified the types of pantun with accuracy of 81,91%.

Keywords: classification, SVM, pantun, text mining, data mining.

Abstrak

Penelitian ini bertujuan untuk membuat model untuk mengkategorikan jenis pantun dan menganalisis hasil akurasi dari metode *support vector machines* (SVM). Tahapan penelitian diawali dengan pengumpulan data yaitu *dataset* pantun yang sudah dilabeli dengan jenis pantunnya masing-masing. *Dataset* pantun yang digunakan terdiri atas tiga kategori yaitu pantun anak-anak, pantun muda, dan pantun orang tua. Setelah data terkumpul, dilakukan tahap pra proses. Tahap pra proses ini bertujuan untuk membuat data sehingga siap untuk diolah di tahapan ekstraksi fitur. Tahap pra proses terdiri atas *text segmentation, case folding, tokenization, stopword removal*, dan *stemming*. Tahapan ekstraksi fitur mempunyai tujuan menggali informasi potensial serta merepresentasikan kata-kata sebagai vektor fitur. Tahapan selanjutnya yaitu memisahkan data latih dan data uji. Kemudian pada proses klasifikasi dilakukan dengan menggunakan metode *multiclass* SVM untuk mendapatkan hasil akhir dari pembuatan sistem. Hasil klasifikasi kemudian dievaluasi untuk mendapatkan nilai akurasi dan akan dianalisis apakah model klasifikasi yang dibuat layak digunakan. Hasil penelitian menunjukkan bahwa SVM dapat dengan baik mengklasifikasi jenis pantun dengan akurasi sebesar 81,91%.

Kata kunci: klasifikasi, SVM, pantun, text mining, data mining.

1. Pendahuluan

Pantun merupakan suatu jenis puisi lama dari kesusastraan Indonesia. Pantun digunakan sebagai salah satu alat komunikasi, menyelipkan nasehat, bahkan sebagai kritik sosial yang ramah, dan dapat dilakukan oleh siapapun untuk menambah semarak suatu kegiatan. Adapun salah satu manfaat pantun yaitu melatih seseorang berpikir mengenai arti dari kata sebelum diucapkan. Selain itu, pantun dapat memberi pandangan pada seseorang agar mampu berpikir bahwa suatu kata bisa memiliki kaitan dengan kata yang lain [1]. Pantun merupakan suatu jenis puisi lama dari kesusastraan Indonesia. Pantun digunakan sebagai salah satu alat komunikasi, menyelipkan nasehat, bahkan sebagai kritik sosial yang ramah, dan dapat dilakukan oleh siapapun untuk menambah semarak suatu kegiatan. Adapun salah satu manfaat pantun yaitu melatih seseorang berpikir mengenai arti dari kata sebelum diucapkan. Selain itu, pantun dapat memberi pandangan pada seseorang agar mampu berpikir bahwa suatu kata bisa memiliki kaitan dengan kata yang lain [2]. Dalam buku Redaksi Balai Pustaka dijelaskan bahwa pembagian pantun itu dapat dibagi menjadi pantun tua, pantun muda, dan pantun anak-anak [3].

Dengan memahami definisi, syarat pantun, dan jenis-jenis pantun, diharapkan masyarakat tidak hanya

Diterima Redaksi : 11-08-2020 | Selesai Revisi : 11-09-2020 | Diterbitkan Online : 30-10-2020

mengenal mengenai pantun yang bersifat profan, namun pantun-pantun dengan jenis yang lain juga. Sehingga, penggunaan pantun di masyarakat dapat lebih berkembang dan sesuai dengan jiwa dari pantun itu sendiri. Pengetahuan jenis-jenis pantun ini masih terbatas diketahui oleh pakar di bidang kesusastraan, oleh karena itu dibutuhkan suatu model yang dapat secara otomatis mengklasifikasikan suatu pantun masuk ke dalam jenis pantun apa. Klasifikasi jenis pantun dapat ditentukan dengan teknik pengolahan teks (*text processing*). Klasifikasi teks merupakan proses untuk mengklasifikasikan dokumen ke dalam satu atau beberapa kategori yang sudah ditentukan sebelumnya [4]. Klasifikasi teks merupakan jenis pembelajaran *supervised* (terpandu). Klasifikasi teks bertujuan membantu mengorganisasikan informasi dalam jumlah banyak sehingga dapat dipahami oleh pengguna [5]. Didasari latar belakang tersebut, dalam penelitian ini akan dibangun suatu model yang dapat melakukan klasifikasi teks pantun ke dalam beberapa kategori pantun yaitu pantun anak-anak, pantun muda, dan pantun tua.

Beberapa jenis algoritma klasifikasi teks yang sering digunakan yaitu *Naïve bayes, Support Vector Machines, Decision Tree,* dan KNN. Dari algoritma-algoritma tersebut, SVM merupakan salah satu algoritma yang menghasilkan nilai akurasi baik. Hasil ini dibuktikan oleh beberapa penelitian sebelumnya mengenai klasifikasi teks juga antara lain, penelitian mengenai klasifikasi topik berita menggunakan SVM. Hasil penelitian ini menyimpulkan bahwa SVM sebagai metode klasifikasi memberikan hasil prediksi yang paling akurat dibandingkan dengan metode yang lainnya yaitu sebesar 94,24% [6]. Penelitian lainnya yaitu penelitian mengenai klasifikasi jenis tiket helpdesk menggunakan SVM, akurasi yang diperoleh cukup tinggi yaitu sebesar 81% [7]. Selain itu, SVM juga pernah digunakan digunakan untuk klasifikasi tugas akhir dengan akurasi sebesar 85,38% [8]. Berdasarkan latar belakang tersebut, dalam penelitian ini akan diimplementasikan metode *support vector machines* untuk klasifikasi jenis pantun.

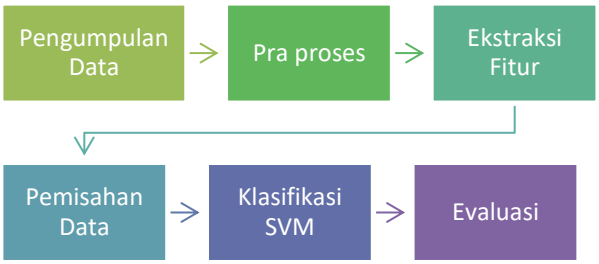
2. Metode Penelitian

Metode penelitian yang digunakan untuk melakukan klasifikasi jenis pantun adalah dengan menerapkan pengolahan *dataset* pantun menggunakan metode *Support Vector Machines* (SVM), rancangan sistem yang akan dibangun dapat dilihat pada Gambar 1.

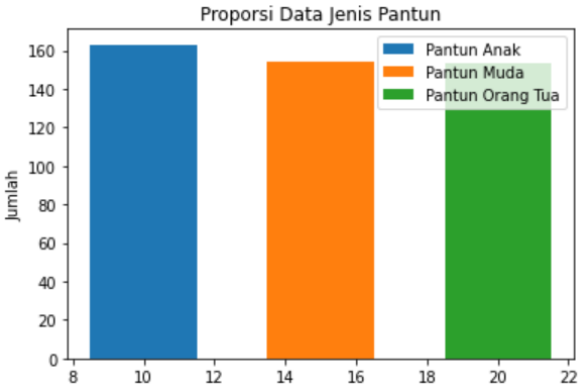
Berdasarkan Gambar 1, terdapat 6 (enam) tahapan dalam penelitian ini antara lain pengumpulan data, pra proses, ekstraksi fitur, pemisahan data latih dan data uji, proses klasifikasi dengan SVM dan terakhir yaitu evaluasi.

2.1. Pengumpulan Data

Dataset yang digunakan sebagai input dari sistem ini yaitu *dataset* pantun yang sudah dilabeli dengan jenis pantunnya masing-masing. Jenis pantun yang digunakan merujuk pada referensi buku Redaksi Balai Pustaka yaitu pantun tua, pantun muda, dan pantun anak-anak [3]. *Dataset* yang dikumpulkan sebanyak 470 *record* data pantun yang telah diberi label pantun anak-anak, pantun muda, dan pantun orang tua [9]. Proporsi data pantun digambarkan pada Gambar 2.



Gambar 1 Rancangan Sistem

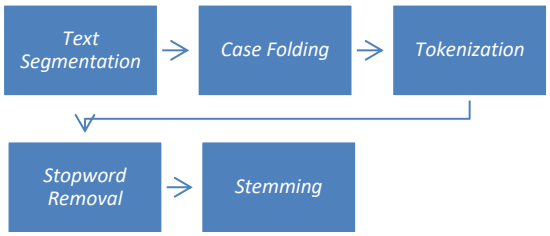


Gambar 2 Proporsi Data Jenis Pantun

Data pantun yang digunakan terdiri atas pantun anak-anak sebanyak 163 data, pantun muda sebanyak 154 data, dan pantun orang tua sebanyak 153 data.

2.2. Pra proses

Setelah data terkumpul, dilakukan tahap pra proses sehingga siap untuk diolah di tahapan selanjutnya. Tahap pra proses terdiri atas *text segmentation, case folding, tokenization, stopword removal, dan stemming*.



Gambar 3 Blok diagram praproses

Berdasarkan Gambar 3, pra proses tahap pertama yaitu *text segmentation* yang bertujuan untuk memecah teks pantun yang utuh ke dalam setiap baris kalimat. Pada proses ini yang akan diambil yaitu bagian isi pantun saja atau baris ke 3 dan baris ke 4. Tahap kedua yaitu *case folding* yang mengubah semua kata dalam teks isi pantun menjadi berhuruf kecil dan karakter lain selain huruf

dihilangkan. Tahap ketiga yaitu *tokenization* yang berfungsi untuk memecah kalimat menjadi token-token atau kata-kata tunggal. Pemecahan ini dilakukan berdasarkan tanda spasi. Tahap keempat yakni *stopword removal* atau menghilangkan kata-kata *stopword*. *Stopword* adalah kumpulan kata-kata yang dianggap tidak memiliki makna. Kemudian tahap terakhir yaitu *stemming*, yang berfungsi untuk mengembalikan kata ke dalam kata dasarnya.

2.3. Ekstraksi Fitur

Setelah melakukan tahapan pra proses, maka tahapan berikutnya adalah melakukan ekstraksi fitur. Dalam mengolah teks diperlukan ekstraksi kata menjadi numerik karena pada prinsipnya komputer tidak dapat mengolah data selain data numerik. Ekstraksi fitur digunakan untuk menggali informasi potensial serta merepresentasikan kata-kata sebagai vektor fitur. Vektor ini akan digunakan sebagai input untuk metode klasifikasi di tahap selanjutnya. Salah satu Teknik dalam ekstraksi fitur yaitu menggunakan TF IDF. *Term Frequency* atau TF dihitung berdasarkan jumlah kemunculan setiap kata dalam tiap dokumen, sedangkan *Inverse Document Frequency* atau IDF dihitung berdasarkan jumlah kemunculan kata dalam keseluruhan dokumen. Nilai TF dibandingkan terhadap nilai IDF [10]. Hasil akhirnya berupa matriks o*utput* dari ekstraksi fitur direpresentasikan dalam bentuk matriks yang berisi kata-kata unik dan nilai-nilai fitur TF-IDF dari setiap kata pada seluruh data pantun. Perhitungan bobot kata dapat dilihat pada Persamaan 1 dan Persamaan 2 [11].

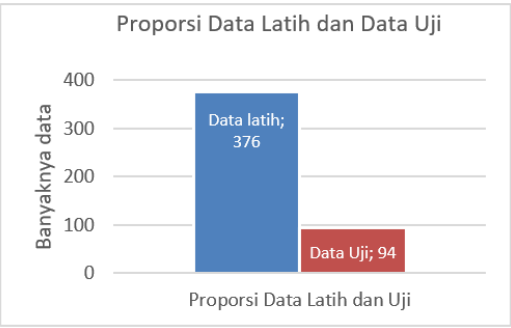
$$w_{ij} = tf_{ij}\, x\, idf_j \quad (1)$$

$$w_{ij} = tf_{ij}\, x \log\,(D/df_j) \quad (2)$$

Berdasarkan Persamaan 1 dan Persamaan 2 dapat dijelaskan bahwa w_{ij} merupakan bobot dari kata dari semua dokumen, tf_{ij} merupakan frekuensi kemunculan suatu kata dalam dokumen, sedangkan idf_j adalah *inverse* dari jumlah dokumen yang berisi suatu kata.

2.4. Pemisahan Data

Tahapan selanjutnya memisahkan data training dan data testing. Pada penelitian ini digunakan data latih sebanyak 80% dan data uji sebanyak 20% dari total data pantun 470 *record*.



Gambar 4 Proporsi data latih dan data uji

Gambar 4 menunjukan proporsi data latih dan data uji. Didapatkan data latih sebanyak 376 data, dan data uji sebanyak 94 data. Penentuan data latih dan data uji dilakukan random sehingga proporsi antar kategori pantun tetap seimbang.

2.5. Klasifikasi dengan SVM

Klasifikasi dilakukan dengan menggunakan metode SVM untuk mendapatkan hasil akhir dari pembuatan sistem. *Support Vector Machine* (SVM) merupakan salah satu metode klasifikasi dengan prinsip mencari *hyperplane* yang memiliki *margin* terbesar. *Hyperplane* yaitu suatu garis yang memisahkan data antar kelas/kategori. Sedangkan margin merupakan jarak antara *hyperplane* dengan data terdekat yang berada pada masing-masing kelas. Data yang paling dekat dengan *hyperplane* disebut *support vector* [8]. Berikut merupakan prsamaan untuk perhitungan *hyperplane* ditunjukan oleh Persamaan 3 [12].

$$w.x + b = 0 \quad (3)$$

Berdasarkan Persamaan 3, dapat disimpulkan bahwa w merupakan parameter *hyperplane* yang dicari, sementara x adalah data input SVM (x1 = index kata, x2 = bobot kata), dan b adalah nilai bias atau parameter *hyperplane* yang dicari. Cara memaksimalkan margin yaitu dengan Persamaan 4 [12]:

$$ax + by + c = 0 \quad (4)$$

Persamaan garis tersebut diubah x menjadi $x1$, y menjadi $x2$, a menjadi $w1$, b menjadi $w2$. Asumsi sekarang berada di dimensi d, kemudian dijadikan persamaan yang lebih umum sehingga menjadi Persamaan 5 dan Persamaan 6 [12].

$$\sum_{j=1}^{d} = w_j x_j + c = 0 \quad (5)$$

$$g(x) = < w, x > + c = 0 \quad (6)$$

SVM pada prinsipnya dapat digunakan untuk b*inary classification* dan *multiclass classification*. Untuk *binary classification*, aturannya ditunjukan pada Persamaan 7 [12].

$$(x) \begin{cases} +1, jika\ g(x) \geq 1 \\ -1, jika\ g(x) < 1 \end{cases} \quad (7)$$

Nilainya akan positif jika hasil g(x) lebih besar atau sama dengan 1, dan bernilai negatif jika hasil g(x) lebih kecil dari 1. Akan tetapi, pada penelitian ini kelas yang digunakan terdiri dari 3 kelas yakni pantun anak-anak, pantun muda, dan pantun orang tua. Oleh karena itu metode SVM yang digunakan adalah *multiclass classification.*

Beberapa metode untuk melakukan *multiclass classification* salah satunya adalah *one against all*. Metode *one against all* berisi *N binary classifiers*, di mana *N* adalah jumlah kelas dalam set data. SVM biner ke-i dilatih dengan semua contoh data di kelas dengan label positif, dan semua contoh data lainnya dilabeli

dengan negatif. Untuk membangun satu model SVM *multiclass* dengan metode *one against all*, diadopsi konsep *binary classifier*, kelas-kelas dibagi menjadi dua kelompok: kelompok pertama dibentuk oleh satu kelas, dan kelompok kedua adalah untuk kelas-kelas lainnya. Pengklasifikasi biner SVM yang diperoleh dilatih untuk memutuskan apakah kelas tersebut berasal dari kelompok pertama atau milik kelompok kelas lainnya. Proses ini diulangi untuk grup kedua yang berisi lebih dari dua kelas hingga hanya memiliki satu kelas untuk setiap grup [13]. Gambaran SVM *multiclass* pada klasifikasi jenis pantun dengan metode *one against all* dapat dilihat pada Tabel 1.

Tabel 1 Gambaran metode *one against all*

yi = 1	yi = -1	Hipotesis
Kelas 1	Bukan kelas 1	$f^1(x) = (w^1)x + b^1$
Kelas 2	Bukan kelas 2	$f^2(x) = (w^2)x + b^2$
Kelas …	Bukan kelas …	$f^{\cdots}(x) = (w^{\cdots})x + b^{\cdots}$
Kelas n	Bukan kelas n	$f^n(x) = (w^n)x + b^n$

Berdasarkan Tabel 1 metode *one against all* akan membangun sebanyak *n* buah model SVM (n adalah jumlah kelas). Setiap model klasifikasi dilatih dengan menggunakan keseluruhan data untuk mencari kelas yang tepat [14]. Jadi, dengan mengikuti cara ini, *multiclass* SVM ditransformasikan ke beberapa *classifier binary* SVM. Setiap *classifier binary* SVM dilatih menggunakan matriks data pelatihan, di mana setiap baris sesuai dengan fitur yang diekstraksi sebagai pengamatan dari kelas. Setelah fase pelatihan, model multiclass SVM dapat memutuskan kelas yang benar untuk vektor fitur input. Untuk mengklasifikasikan objek, vektor fitur inputnya disajikan secara iteratif ke-*i* terhadap semua classifier biner dari yang pertama ke classifier *N* ketika hasil negatif. Ketika *classifier* biner ke-*i* memberikan hasil positif, proses dihentikan. Ini berarti bahwa objek tersebut milik kelas ke-*i* [13].

2.5. Evaluasi

Hasil klasifikasi kemudian dievaluasi untuk mendapatkan nilai akurasi yang akan dianalisis apakah model klasifikasi yang dibuat layak digunakan. Untuk menggambarkan kinerja dari model klasifikasi, dibuat tabel *confusion matrix* 3x3 diterapkan untuk penelitian ini, yang ditunjukkan pada Tabel 2 [15].

Tabel 2 *Confusion Matrix* 3x3 [15]

		Hasil Prediksi		
		A	B	C
Hasil Aktual	A	A	FB1	FC1
	B	FA1	TB	FC2
	C	FA2	FB2	TC

Berdasarkan Tabel 2, *confusion matrix* 3x3 terdiri atasa 2 bagian yaitu hasil prediksi dan hasil aktual. Hasil aktual adalah data yang didapatkan berdasarkan kenyataan. Hasil aktual terdiri atas tiga kelas yaitu kelas A, B, dan C. Sedangkan hasil prediksi adalah hasil klasifikasi yang didapatkan oleh model klasifikasi SVM yang juga terdiri atas tiga kelas yaitu kelas A, B, dan C. Kasus dibagi menjadi sembilan nilai: TA, FA1, FA2, FB1, TB, FB2, FC1, FC2, dan TC. TA adalah kelas A yang diklasifikasikan dengan benar, TB adalah kelas B yang diklasifikasikan dengan benar, TC adalah kelas C yang diklasifikasikan dengan benar. FA1 adalah kelas B yang diklasifikasikan ke dalam kelas A, FA2 adalah kelas C yang diklasifikasikan ke dalam kelas A. FB1 adalah kelas A yang diklasifikasikan ke dalam kelas B, FB2 adalah kelas C yang diklasifikasikan ke dalam kelas B. FC1 adalah kelas A yang diklasifikasikan ke dalam kelas C, FC2 adalah kelas B yang diklasifikasikan ke dalam kelas C [15].

Setelah dibuat *confusion matrix* 3x3, nilai akurasi dihitung menilai kinerja SVM untuk mengklasifikasikan, ditentukan dalam Persamaan 8 [15].

$$Akurasi = \frac{T}{T+FA1+FA2+FB1+FB2+FC1+FC2}\ x\ 100\% \quad (8)$$

Nilai T dalam Persamaan 8 yaitu penjumlahan TA+TB+TC. Nilai akurasi adalah nilai yang didapatkan dari hasil bagi dari semua jumlah data uji yang benar dengan jumlah data uji keseluruhan [16].

Selain akurasi untuk mengevaluasi keberhasilan model prediksi, dilakukan juga perhitungan *precision, recall* (sensitivitas), dan spesifisitas untuk setiap kelas jenis pantun dengan Persamaan 9 [16], Persamaan 10 [16], dan Persamaan 11 [16].

$$Precision = \frac{TP}{TP+FP}\ x\ 100\% \quad (9)$$

$$Recall = \frac{TP}{TP+FN}\ x\ 100\% \quad (10)$$

$$Spesifisitas = \frac{TN}{TN+FP}\ x\ 100\% \quad (11)$$

Precision pada Persamaan 9 digunakan untuk mendapat perbandingan jumlah data yang diprediksi positif benar dibandingkan dengan keseluruhan data yang prediksi positif. Untuk kasus dengan 3 kelas, disesuaikan dengan *confusion matrix* pada Tabel 2, maka TP untuk kelas A yaitu TA, FP untuk kelas A yaitu FA1+FA2. TP untuk kelas B yaitu TB, FP untuk kelas A yaitu FB1+FB2. TP untuk kelas C yaitu TC, FP untuk kelas C yaitu FC1+ FC2.

Recall atau sensitivitas digunakan untuk mendapat perbandingan jumlah prediksi positif yang benar dibandingkan dengan keseluruhan jumlah kelas yang positif. Untuk kasus dengan 3 kelas, disesuaikan dengan *confusion matrix* pada Tabel 2, maka TP untuk kelas A yaitu TA, FN untuk kelas A yaitu FB1+FC1. TP untuk kelas B yaitu TB, FN untuk kelas B yaitu FB1+FC2. TP untuk kelas C yaitu TC, FN untuk kelas C yaitu FA2+FB2.

Sedangkan spesifisitas merupakan perbandingan data jumlah prediksi negatif yang benar dibandingkan dengan keseluruhan jumlah kelas negatif. Untuk kasus dengan 3

kelas, disesuaikan dengan *confusion matrix* pada Tabel 2, maka TN untuk kelas A yaitu TB+FB2+FC2+TC, FP untuk kelas A yaitu FA1+FA2. TN untuk kelas B yaitu TA+FA2+FC1+TC, FP untuk kelas B yaitu FB1+FB2. TN untuk kelas C yaitu TA+FA1+FB1+TB, FP untuk kelas C yaitu FC1+FC2.

3. Hasil dan Pembahasan

Hasil dan pembahasan dari klasifikasi pantun dengan menggunakan SVM dijelaskan berdasarkan blok diagram yang telah direncanakan sebelumnya di bab 2 bagian metode penelitian.

3.1. Pra proses

Pra proses yang terdiri dari 6 tahapan yaitu *text segmentation, case folding, tokenization, stopword removal,* dan *stemming,* dilakukan dengan menggunakan Bahasa pemrograman *python* dengan *library* NLTK, dan sastrawi.

Tahap t*ext segmentation* atau memecah ke dalam baris-baris, mengambil isi dari pantun (baris ke-3 dan baris ke-4) hasilnya yaitu:

Contoh *input* = "Akar alang entah menghilang. Tumbuh bukan sebagai tanaman. Hati ini senang bukan kepalang. Bermain bola bersama teman."

Contoh *output* = Hati ini senang bukan kepalang. Bermain bola bersama teman.

Tahap *case folding* atau mengubah teks isi pantun menjadi huruf kecil dan menghilangkan karakter selain huruf, hasilnya yaitu:
Contoh *output* = hati ini senang bukan kepalang bermain bola bersama teman

Tahap *tokenization* atau pemecahan menjadi token-token/kata, hasilnya yaitu:
Contoh *output* = [hati, ini, senang, bukan, kepalang, bermain, bola, bersama, teman]

Tahap *stopword removal* atau menghapus kata-kata yang dianggap tidak memiliki makna menggunakan *library* sastrawi yang bisa diakses melalui https://github.com/sastrawi/sastrawi. Total *stopword* dari *library* sastrawi ini sebanyak 758 kata [17]. Beberapa kata-kata *stopword* dalam Sastrawi antara lain: ['yang', 'untuk', 'pada', 'ke', 'para', 'namun', 'menurut', 'antara', 'dia', 'dua', 'ia', 'seperti', 'jika', 'sehingga', 'kembali', 'dan', 'tidak', 'ini', …]. Jika diimplementasikan dalam contoh kata 'ini' menjadi hilang, sehingga hasilnya yaitu:

Contoh *output* = [hati, senang, bukan, kepalang, bermain, bola, bersama, teman]

Tahap *stemming* atau mengubah kata menjadi kata dasarnya, hasilnya yaitu:

Contoh *output* = [hati, senang, bukan, kepalang, main, bola, sama, teman]

Keenam tahapan telah dilakukan sehingga kita mendapatkan teks pantun yang sudah bersih dan siap digunakan pada tahap ekstraksi fitur.

3.2. Ekstraksi Fitur

Ekstraksi fitur dalam penelitian ini menggunakan pembobotan TF-IDF dengan rumus seperti pada Persamaan 1 dan Persamaan 2. TF-IDF akan menilai seberapa penting sebuah kata di dalam dokumen. Untuk melakukan perhitungan TF-IDF digunakan *library* di *Sklearn python* yaitu *TfidfVectorizer()*. Setelah mendapatkan bobot setiap term/kata dengan menggunakan TF-IDF, didapatkan ranking bobot *term*/kata secara keseluruhan dokumen yaitu:

Tabel 3 Ranking bobot *term* keseluruhan dokumen pantun

No	*Term*
1	hati
2	jangan
3	jadi
4	kalau
5	orang
6	anak
7	badan
8	hendak
9	main
10	tak

Banyaknya *term*/fitur yang akan digunakan, dipilih berdasarkan *top* fitur maksimum yang diurutkan berdasarkan *term frequency* seluruh korpus pantun. Pada penelitian ini dibutuhkan jumlah fiturnya tidak terlalu besar, tetapi memungkinkan model klasifikasi tetap dapat berfungsi dengan baik. Oleh karena itu, diambil maksimum fitur yang digunakan adalah 1000 fitur.

Untuk ranking *term* setiap kategori (pantun anak-anak, pantun muda, dan pantun orang tua) ditunjukan pada Tabel 4, Tabel 5, dan Tabel 6.

Tabel 4 Ranking bobot *term* keseluruhan dokumen pantun anak-anak

No	*Term*
1	hati
2	jadi
3	anak
4	main
5	jangan
6	teman
7	senang
8	sama
9	ibu
10	ayah

Tabel 5 Ranking bobot *term* keseluruhan dokumen pantun muda

No	*Term*
1	hati
2	kalau
3	badan
4	orang
5	rindu
6	hendak
7	rasa
8	tak
9	abang
10	kasih

Tabel 6 Ranking bobot *term* keseluruhan dokumen pantun orang tua

No	Term
1	jangan
2	kalau
3	orang
4	jadi
5	hendak
6	banyak
7	ajar
8	allah
9	ilmu
10	mati

3.2. Klasifikasi SVM dan Evaluasi

Proses klasifikasi dilakukan dengan SVM diawali pada data latih yaitu sebanyak 376 teks isi pantun. Pada penelitian ini proses klasifikasi menggunakan *library* SVM pada *Sklearn Python*. Oleh karena klasifikasi ini bukan *binary classification* melainkan *multiclass classification* (3 kelas/kategori) sehingga pada penelitian ini juga menggunakan *library OneVsRestClassifier* (sama dengan metode *one against all*) dari *Sklearn* di *python*. Gambaran SVM *multiclass* pada klasifikasi jenis pantun dengan metode *one against all* dapat dilihat pada Tabel 7.

Tabel 7 Gambaran one against all pada klasifikasi jenis pantun

Problem	Kelas 1	Kelas -1
Binary Classification 1	Pantun anak-anak	Pantun muda, pantun orang tua
Binary Classification 2	Pantun muda	Pantun anak-anak, pantun orang tua
Binary Classification 3	Pantun orang tua	Pantun anak-anak, pantun muda

Berdasarkan Tabel 7 bisa diartikan bahwa dibuat 3 *binary classification* dengan mengikuti aturan pada Persamaan 7. B*inary classification* 1 untuk menyaring nilai positif yang merupakan pantun anak-anak, dan kelas yang lainnya pantun muda, pantun orang tua merupakan kelas negatif. *Binary classification* 2 untuk menyaring nilai positif yang merupakan pantun muda dan kelas yang lainnya pantun anak-anak, pantun orang tua merupakan kelas negatif. Sedangkan *binary classification* 3 untuk menyaring nilai positif yang merupakan pantun orang tua dan kelas yang lainnya pantun anak-anak, pantun muda merupakan kelas negatif.

Model yang didapatkan dari proses *training*, akan disimpan dan digunakan pada proses *testing*. Proses *testing* dilakukan pada 20% dari total data atau sebanyak 94 teks isi pantun. Pada proses *testing* ini akan diperoleh kelas yang diprediksi oleh model. Kelas ini akan dibandingkan dengan kelas sebenarnya. Untuk mengetahui apakah model yang dibuat sukses atau tidak, diperlukan evaluasi. Cara evaluasi salah satunya dapat digunakan perhitungan nilai akurasi. Seperti dijelaskan pada Tabel 2, data hasil *testing* disajikan dalam bentuk *confusion matrix* di Tabel 8.

Tabel 8 Confusion matrix hasil klasifikasi

		Hasil Prediksi		
		Anak	Muda	Tua
Hasil Aktual	Anak	29	1	3
	Muda	1	26	6
	Tua	2	4	22

Dari hasil klasifikasi pada Tabel 8 terdapat 4 kemungkinan kasus yang terjadi antara lain:

TP (*True Positive*) untuk kelas pantun anak-anak (TA) adalah pantun diprediksi sebagai pantun anak-anak dan nilai sebenarnya juga pantun anak-anak sebanyak 29. TN (*True Negative*) untuk kelas pantun anak-anak (TB+FB2+FC2+TC) adalah pantun yang diprediksi sebagai bukan pantun anak-anak dan nilai sebenarnya juga bukan pantun anak-anak sebanyak 26+6+4+22 = 58. FP (*False Positive*) untuk kelas pantun anak (FC1+ FC2) adalah pantun yang diprediksi sebagai pantun anak-anak, namun kelas pantun sebenarnya bukan pantun anak-anak sebanyak 1+2 = 3. FN (*False Negative*) untuk kelas pantun anak-anak (FB1+FC1) adalah pantun yang diprediksi sebagai bukan pantun anak-anak, namun kelas pantun sebenarnya adalah pantun anak-anak sebanyak 1+3 = 4.

TP (*True Positive*) untuk kelas pantun muda (TB) adalah pantun diprediksi sebagai pantun muda dan nilai sebenarnya juga pantun muda dengan sebanyak 26. TN (*True Negative*) untuk kelas pantun muda (TA+FA2+FC1+TC) adalah pantun yang diprediksi sebagai bukan pantun muda dan nilai sebenarnya juga bukan pantun muda sebanyak 29+3+2+22 = 56. FP (*False Positive*) untuk kelas pantun muda (FB1+FB2) adalah pantun yang diprediksi sebagai pantun muda, namun kelas pantun sebenarnya bukan pantun muda sebanyak 1+4 = 5. FN (*False Negative*) untuk kelas pantun muda (FB1+FC2) adalah pantun yang diprediksi sebagai bukan pantun muda, namun kelas pantun sebenarnya adalah pantun muda sebanyak 1+6 = 7.

TP (*True Positive*) untuk kelas pantun orang tua (TC) adalah pantun diprediksi sebagai pantun orang tua dan nilai sebenarnya juga pantun orang tua dengan sebanyak 22. TN (*True Negative*) untuk kelas pantun orang tua (TA+FA1+FB1+TB) adalah pantun yang diprediksi sebagai bukan pantun orang tua dan nilai sebenarnya juga bukan pantun orang tua sebanyak 29+1+1+26 = 57. FP (*False Positive*) untuk kelas pantun orang tua (FC1+FC2) adalah pantun yang diprediksi sebagai pantun orang tua, namun kelas pantun sebenarnya bukan pantun orang tua sebanyak 3+6= 9. FN (*False Negative*) untuk kelas pantun orang tua (FA2+FB2) adalah pantun yang diprediksi sebagai bukan pantun orang tua, namun kelas pantun sebenarnya adalah pantun orang tua sebanyak 2+4= 6.

Berdasarkan nilai TP, TN, FP, dan FN untuk masing-masing kelas pantun, dihitung *precision*, *recall*, dan spesifisitas dengan menggunakan Persamaan 9, Persamaan 10, dan Persamaan 11. Hasil dari perhitungan *precision*, *recall*, dan spesifisitas untuk kelas pantun anak, pantun muda, dan pantun tua ditunjukan pada Tabel 9.

Tabel 9 Nilai *precision*, *recall*, dan spesifisitas setiap kelas

Kelas	*Precision*	*Recall*	Spesifisitas
Pantun Anak	90,63%	87,88%	95,08%
Pantun Muda	83,87%	78,79%	91,80%
Pantun Tua	70,97%	78,57%	86,36%

Pada Tabel 9 terlihat nilai *precision* menjelaskan bahwa persentase prediksi pantun anak yang benar dari keseluruhan pantun yang diprediksi pantun anak adalah sebesar 90,63%, persentase prediksi pantun muda yang benar dari keseluruhan pantun yang diprediksi pantun muda adalah sebesar 83,87%, dan persentase prediksi pantun tua yang benar dari keseluruhan pantun yang diprediksi pantun tua adalah sebesar 70,97%. Nilai *precision* tertinggi diperoleh oleh pantun anak. Nilai *recall* (sensitifitas) menjelaskan bahwa persentase prediksi pantun anak yang benar dibandingkan jumlah keseluruhan kelas pantun anak adalah sebesar 87,88%, persentase prediksi pantun muda yang benar dibandingkan jumlah keseluruhan kelas pantun muda adalah sebesar 78,79%, dan persentase prediksi pantun tua yang benar dibandingkan jumlah keseluruhan kelas pantun tua adalah sebesar 78,57%, Nilai *recall* tertinggi diperoleh oleh pantun anak. Nilai spesifisitas menjelaskan bahwa persentase keseluruhan jumlah prediksi bukan pantun anak yang benar dibandingkan dengan keseluruhan jumlah kelas yang bukan pantun anak adalah sebesar 95,08%, persentase keseluruhan jumlah prediksi bukan pantun muda yang benar dibandingkan dengan keseluruhan jumlah kelas yang bukan pantun muda adalah sebesar 91,80%, dan persentase keseluruhan jumlah prediksi bukan pantun orang tua yang benar dibandingkan dengan keseluruhan jumlah kelas yang bukan pantun orang tua adalah sebesar 86,36%. Nilai spesifisitas tertinggi juga diperoleh oleh pantun anak. Nilai *precision*, *recall*, dan spesifisitas pantun anak yang tinggi terjadi karena jumlah *dataset* dengan label pantun anak lebih banyak dibandingkan pantun muda dan pantun orang tua.

Selain *precision*, *recall*, dan spesifisitas, berdasarkan Persamaan 8, pada pantun anak-anak, pantun muda, pantun orang tua, dihitung nilai akurasi secara keseluruhan untuk model klasifikasi jenis pantun dengan menggunakan SVM ini yaitu penjumlahan TA+TB+TC atau 29+26+22=77 dibagi dengan jumlah seluruh data uji TA+TB+TC+FA1+FA2+FB1+FB2+FC1+FC2 atau 29+26+22+1+3+1+6+2+4=94 sehingga menghasilkan nilai akurasi sebesar 81,91%. Dari nilai akurasi tersebut dapat disimpulkan bahwa model klasifikasi jenis pantun dengan metode SVM dapat bekerja dengan baik.

4. Kesimpulan

Berdasarkan hasil penelitian klasifikasi jenis pantun dengan SVM dengan fitur maksimum sebanyak 1000 fitur dan jumlah *dataset* 470 data pantun yang terdiri dari pantun anak, pantun muda, dan pantun orang tua (data latih 376 *record*, data uji 94 *record*) dapat disimpulkan SVM dapat dengan baik mengklasifikasi jenis pantun dengan akurasi sebesar 81,91%. Selain itu pantun anak-anak memiliki nilai *precision*, *recall*, dan spesifisitas lebih tinggi dibandingkan pantun tua dan pantun muda yaitu sebesar 90,63%, 87,88%, 95,08%. Meskipun demikian, nilai *precision*, *recall*, dan spesifisitas untuk pantun muda maupun pantun orang tua memiliki nilai yang juga baik dan dapat diterima. Nilai *precision*, *recall*, dan spesifisitas pantun anak tertinggi dipengaruhi oleh jumlah data pantun anak yang lebih banyak dibanding pantun muda dan pantun orang tua. Saran untuk penelitian berikutnya adalah melakukan penambahan data lebih banyak lagi baik untuk kategori pantun anak, pantun muda, dan pantun orang tua. Dengan demikian, diharapkan nilai akurasi, *precision*, *recall*, dan spesifisitas akan meningkat.

Ucapan Terima Kasih

Ucapan terima kasih disampaikan tim penulis kepada Direktorat Penelitian Dan Pengabdian Masyarakat Direktorat Jenderal Perguruan Tinggi, KEMENDIKBUD, serta Lembaga Penelitian dan Pengabdian Masyarakat (LPPM) Universitas Pembangunan Nasional Veteran Jakarta yang telah mendanai penelitian ini melalui skim Penelitian Dosen Pemula.

Daftar Rujukan

[1] D. E. Maulina, "Keanekaragaman Pantun Di Indonesia," *Semantik*, vol. 1, no. 1, pp. 107–121, 2015.

[2] F. N. Murti, W. Siswanto, and H. Suwignyo, "Model Threshold Pantun untuk Pembelajaran Memproduksi Pantun Kelas XI," Tesis tidak diterbitkan). Pascasarjana Universitas Negeri Malang, Malang, 2015.

[3] T. Andriani, "Pantun Dalam Kehidupan Melayu (Pendekatan historis dan antropologis)," *Sos. Budaya*, vol. 9, no. 2, pp. 195–211, 2012.

[4] L. Mutawalli, M. T. A. Zaen, and W. Bagye, "Klasifikasi Teks Sosial Media Twitter Menggunakan Support Vector Machine (Studi Kasus Penusukan Wiranto)," *J. Inform. dan Rekayasa Elektron.*, vol. 2, no. 2, pp. 43–51, 2019.

[5] C. Darujati and A. B. Gumelar, "Pemanfaatan teknik supervised untuk klasifikasi teks bahasa indonesia," *J. Bandung Text Min.*, vol. 16, no. 1, pp. 1–5, 2012.

[6] L. G. Irham, A. Adiwijaya, and U. N. Wisesty, "Klasifikasi Berita Bahasa Indonesia Menggunakan Mutual Information dan Support Vector Machine," *J. Media Inform. Budidarma*, vol. 3, no. 4, pp. 284–292, 2019.

[7] S. H. Kusumahadi, H. Junaedi, and J. Santoso, "Klasifikasi Helpdesk Menggunakan Metode Support Vector Machine," *J. Inform.*, vol. 4, no. 01, pp. 55–60, 2019.

[8] O. Somantri, S. Wiyono, and D. Dairoh, "Metode K-Means untuk Optimasi Klasifikasi Tema Tugas Akhir Mahasiswa Menggunakan Support Vector Machine (SVM)," *Sci. J. Informatics*, vol. 3, no. 1, pp. 34–45, 2016.

[9] E. Waridah, *Kumpulan Majas, Pantun, dan Peribahasa Plus*

Kesusastraan Indonesia. Ruang Kata, 2014.

[10] P. M. Prihatini, "Implementasi Ekstraksi Fitur Pada Pengolahan Dokumen Berbahasa Indonesia," *Matrix J. Manaj. Teknol. dan Inform.*, vol. 6, no. 3, pp. 174–178, 2017.

[11] A. Riyani, M. Z. Naf'an, and A. Burhanuddin, "Penerapan Cosine Similarity dan Pembobotan TF-IDF untuk Mendeteksi Kemiripan Dokumen," *J. Linguist. Komputasional*, vol. 2, no. 1, pp. 23–27, 2019.

[12] A. Fikriani, I. Asror, and Y. R. Murti, "Klasifikasi Kepribadian Berdasarkan Data Twitter dengan Menggunakan Metode Support Vector Machine," *eProceedings Eng.*, vol. 6, no. 3, pp. 10436–10450, 2019.

[13] M. Oujaoura, B. Minaoui, M. Fakir, R. El Ayachi, and O. Bencharef, "Recognition of isolated printed tifinagh characters," *Int. J. Comput. Appl.*, vol. 85, no. 1, pp. 1–13, 2014.

[14] D. Retnowati, E. Ernawati, and K. Anggriani, "Penerapan Support Vector Machine Untuk Pendeteksian dan Klasifikasi Motif Pada Citra Batik Besurek Motif Gabungan Berdasarkan Fitur Histogram Of Oriented Gradient," *Pseudocode*, vol. 5, no. 2, pp. 75–84, 2018.

[15] A. Faricha, M. Rivai, M. A. Nanda, D. Purwanto, R. R. P. Anhar, and others, "Design of electronic nose system using gas chromatography principle and Surface Acoustic Wave sensor," *Telkomnika*, vol. 16, no. 4, pp. 1457–1467, 2018.

[16] S. Adinugroho and Y. A. Sari, *Implementasi Data Mining Menggunakan Weka*. Universitas Brawijaya Press, 2018.

[17] A. N. Rohman, R. D. Handayani, and K. Kusrini, "Deteksi Emosi Media Sosial Menggunakan Term Frequency-Inverse Document Frequency," *CSRID (Computer Sci. Res. Its Dev. Journal)*, vol. 11, no. 3, pp. 140–148, 2020.