

## PENERAPAN ALGORITMA *SUPPORT VECTOR MACHINE* UNTUK MENDETEKSI EMOSI DARI TEKS BAHASA INDONESIA

Bhayu Aji Seno<sup>1</sup>, Widodo<sup>2</sup>, Bambang Prasetya Adhi<sup>3</sup>

<sup>1</sup> Mahasiswa Prodi Pendidikan Teknik Informatika dan Komputer, Teknik Elektro, FT – UNJ

<sup>2,3</sup> Dosen Prodi Pendidikan Teknik Informatika dan Komputer, Teknik Elektro, FT – UNJ

<sup>1</sup> [bhayuajiseno@gmail.com](mailto:bhayuajiseno@gmail.com), <sup>2</sup> [widodo@unj.ac.id](mailto:widodo@unj.ac.id), <sup>3</sup> [bambangpadhi@unj.ac.id](mailto:bambangpadhi@unj.ac.id)

### Abstrak

Pertumbuhan penggunaan internet khususnya sosial media yang sangat pesat telah menyebabkan perubahan interaksi antar manusia yang sebelumnya terjadi secara langsung di dunia nyata kemudian berubah menjadi tidak langsung melalui sosial media. Ketika manusia berinteraksi melalui sosial media, terkadang terdapat kesalahpahaman informasi yang didapatkan dibandingkan secara langsung di dunia nyata. Hal ini karena kurangnya faktor penting ketika manusia berinteraksi melalui sosial media, yaitu emosi. Penelitian ini bertujuan untuk mendeteksi emosi dari teks bahasa Indonesia dengan menerapkan algoritma *Support Vector Machine* (SVM). Penelitian dimulai dengan pembuatan 4 jenis data sampel yaitu, data sampel dengan jumlah 1000, 2500, dan 4000 tweet untuk proses training, dan data sampel dengan jumlah 120 tweet untuk proses testing. Kemudian seluruh data sampel tersebut melalui proses *Natural Language Processing* (NLP) untuk diekstrak informasinya. Setelah itu dilakukan proses training dengan algoritma SVM menggunakan 3 tipe kernel yang berbeda, yaitu *Polinomial*, *Gaussian RBF*, dan *Sigmoid*. Kemudian dilakukan pengujian dengan hasil tingkat akurasi yang cukup variatif untuk setiap jenis data sampel serta tipe kernel yang digunakan. Hasil penelitian mendapatkan kesimpulan semakin banyak jumlah data yang digunakan, maka semakin tinggi tingkat akurasi yang dihasilkan, namun waktu proses training yang dibutuhkan lebih lama. Sementara untuk tipe kernel *Sigmoid* memiliki tingkat akurasi yang tinggi, serta waktu proses training yang lebih cepat dibandingkan tipe kernel lainnya.

**Kata kunci :** *Support Vector Machine, Natural Language Processing, Emosi*

### 1. Pendahuluan

Pertumbuhan penggunaan internet khususnya sosial media yang sangat pesat telah menyebabkan perubahan interaksi antar manusia yang sebelumnya terjadi secara langsung di dunia nyata kemudian berubah menjadi tidak langsung melalui sosial media. Ketika manusia berinteraksi melalui sosial media, terkadang terdapat kesalahpahaman informasi yang didapatkan dibandingkan secara langsung di dunia nyata. Hal ini karena kurangnya faktor penting ketika manusia berinteraksi melalui sosial media, yaitu emosi (Lopatovska dan Arapakis, 2011). Kesalahpahaman informasi ini dapat menyebabkan permasalahan baru di dunia nyata.

Sejalan dengan pertumbuhan sosial media tersebut juga menghasilkan data teks yang berjumlah semakin banyak (Paltoglou, dkk., 2013). Jumlah data teks tersebut dapat diaplikasikan pada berbagai bidang serta menyimpan berbagai informasi, salah satu informasi tersebut adalah emosi. Kemampuan mendeteksi dan menanggapi emosi dengan baik dapat mengurangi kesalahpahaman informasi yang didapatkan serta meningkatkan kecerdasan buatan sistem komputer (Picard, 1997). Kebutuhan terhadap sistem komputer yang dapat melakukan deteksi dan memberi tanggapan terhadap emosi telah memunculkan bidang *affective computing*.

*Affective computing* merupakan bidang ilmu yang menggabungkan ilmu psikologi, ilmu kognitif dan ilmu komputer dalam pengembangan teknologi, sistem dan alat yang dapat mengenali, memproses dan meniru efek manusia seperti emosi dan perasaan (Picard, 2007). Deteksi emosi merupakan kemampuan utama dan memiliki peran penting dalam *affective computing* (Quan dan Ren, 2016).

Deteksi emosi dapat dilakukan berdasarkan ekspresi wajah (Rahulamathavan, dkk., 2012), suara (Pfister dan Robinson, 2011), gerak tubuh (Kleinsmith dan Bianchi-berthouze, 2013), sinyal *electroencephalogram* – EEG (Soleymani, dkk., 2012), dan teks (Calvo dan D'Mello, 2010). Deteksi emosi dari teks dapat digunakan pada berbagai aplikasi seperti sistem tutor cerdas, analisis kepuasan pelanggan, sistem dialog, dan perilaku blogger (Mohammad dan Turney, 2013). Oleh karena itu deteksi emosi dari teks menjadi penting untuk dilakukan.

Berbagai penelitian dengan beberapa pendekatan telah dilakukan untuk menyelesaikan masalah deteksi emosi dari teks. Pendekatan yang digunakan untuk menyelesaikan masalah deteksi emosi dari teks dapat dikelompokkan menjadi tiga yaitu *keyword-spotting*, *learning-based* dan *rule-based* (Neviarouskaya, dkk., 2011).

Pendekatan *learning-based* merupakan pendekatan yang banyak digunakan. Namun demikian, pendekatan *learning-based* memiliki kelemahan utama yaitu membutuhkan data dalam jumlah yang besar (Neviarouskaya, dkk., 2011).

Penelitian deteksi emosi dari teks telah dilakukan terhadap isi blog, email, *tweet*, cerita anak dan pesan singkat (Paltoglou dan Thelwall, 2013). Mayoritas penelitian tersebut menggunakan data teks bahasa Inggris. Pada sisi yang lain, pengguna internet dan media sosial di Indonesia mengalami pertumbuhan pesat dalam beberapa tahun terakhir. Namun, penelitian deteksi emosi dari teks bahasa Indonesia belum banyak dilakukan, serta belum tersedianya kumpulan data teks bahasa Indonesia dalam jumlah yang besar untuk digunakan dalam penelitian.

Berdasarkan sejumlah uraian sebelumnya maka penelitian deteksi emosi dari teks bahasa Indonesia perlu dilakukan. Deteksi dilakukan dengan menggunakan pendekatan berbasis *learning-based*. Pendekatan *learning-based* menggunakan metode *Support Vector Machine* (SVM) yang memiliki beberapa kombinasi parameter yang harus diatur sehingga dapat menghasilkan algoritma SVM dengan hasil akurasi terbaik dan memiliki *learning time* yang cepat. Sementara untuk kumpulan data teks bahasa Indonesia dalam jumlah besar yang digunakan dalam penelitian mengambil data *tweet* dari situs media sosial Twitter.

## 2. Dasar Teori

### 2.1. Model Emosi

Terdapat berbagai model emosi yang dapat diekspresikan oleh manusia. Model emosi tersebut menjadi dasar dari deteksi emosi. Berbagai model emosi telah diusulkan oleh para peneliti dalam bidang psikologi dan telah digunakan dalam penelitian deteksi emosi. Meskipun terdapat berbagai model emosi, terdapat dua model emosi yang banyak digunakan dalam deteksi emosi yaitu *categorical* dan *dimensional* (Calvo & Kim, 2013).

Pada model emosi *categorical*, emosi dikelompokkan menjadi sejumlah emosi dasar (Peter & Herbon, 2006). Emosi dasar tersebut menjadi kategori emosi yang saling berbeda satu sama lain dan setiap kategori memiliki label (Binali & Potdar, 2012). Kategori emosi pada model *categorical* berjumlah 2 hingga 18 (Peter & Herbon, 2006). Terdapat enam kategori emosi dasar yaitu senang, sedih, marah, takut, terkejut dan sedih (Ekman, 1972). Jenis emosi seperti “bosan (*boredom*)”, “frustrasi (*frustration*)” dan “bingung (*confusion*)” telah digunakan dalam sistem tutor cerdas “*AutoTutor*” (D’Mello, dkk., 2007). Sementara pada model *dimensional*, emosi dimodelkan dalam bentuk dimensi. Suatu jenis emosi menempati suatu titik pada ruang dimensi tersebut. Jenis emosi pada model *dimensional* saling terkait satu dengan yang lain (Peter & Herbon, 2006).

Model emosi yang digunakan pada penelitian ini yaitu model *categorical*. Model emosi ini terdiri dari enam kategori emosi yaitu senang, sedih, terkejut, marah, takut dan jijik (Ekman, 1972).

### 2.2. Affective Computing

*Affective computing* merupakan bidang ilmu yang menggabungkan ilmu psikologi, ilmu kognitif dan ilmu komputer dalam pengembangan teknologi, sistem dan alat yang dapat mengenali, memproses dan meniru efek manusia seperti emosi dan perasaan (Picard, 1997). *Affective computing* terbagi menjadi empat kategori yang dapat diilustrasikan sebagai berikut:

Tabel 2.1. Ilustrasi Empat Kategori Affective Computing

Komputer	Tidak dapat mengekspresikan emosi	Dapat mengekspresikan emosi
Tidak dapat memahami emosi	I	II
Dapat memahami emosi	III	IV

Pada kategori pertama (I), komputer tidak dapat memahami emosi dan tidak dapat berinteraksi lebih baik dari manusia dengan manusia. Pada kategori kedua (II), sistem komputer tidak dapat memahami emosi, tetapi dapat mengekspresikan emosi. Sebagai contoh, Siri pada perangkat Iphone, dimana Siri dapat memberikan tanggapan dari berbagai emosi, tetapi tidak dapat memahami emosi dari pengguna.

Pada kategori ketiga (III), sistem komputer dapat memahami dan memberikan respon terhadap emosi pengguna, tetapi tidak dapat mengekspresikan emosi itu sendiri. Pada kategori keempat (IV), sistem komputer dapat memahami dan mengekspresikan emosi. Apabila suatu sistem komputer telah mencapai kategori ini, maka sistem komputer tersebut bersifat “personal” dan bersifat “*user friendly*”.

### 2.3. Deteksi Emosi

Salah satu persoalan utama dalam penelitian *affective computing* adalah deteksi emosi (Calvo & D'Mello, 2010; Quan & Ren, 2014). Deteksi emosi bertujuan untuk mengenali keadaan emosi dari pengguna. Komputer dapat mengolah dan memberikan tanggapan yang tepat jika komputer dapat mengenali terlebih dahulu keadaan emosi pengguna (Calvo & D'Mello, 2010). Maka, deteksi emosi menjadi proses pertama dan utama sebelum komputer dapat mengolah emosi pengguna.

Deteksi emosi dari pengguna dapat dilakukan melalui berbagai cara dan media seperti suara, ekspresi wajah dan teks. Teks merupakan media utama dalam berkomunikasi menggunakan komputer (Anusha & Sandhya, 2015). Selain berisi informasi, teks juga dapat berisi sentimen ataupun emosi. Berdasarkan hal tersebut maka teks dapat digunakan untuk mendeteksi emosi.

Deteksi emosi dari teks dapat diterapkan pada berbagai bidang seperti *customer service*, media pembelajaran online, *video game* dan antar muka (Anusha & Sandhya, 2015). Deteksi emosi dari teks juga merupakan persoalan penelitian yang menarik, berbagai pendekatan telah digunakan untuk menyelesaikan persoalan deteksi emosi dari teks. Berdasarkan cara deteksi maka pendekatan tersebut dapat dikelompokkan menjadi *keyword-spotting*, *learning-based* dan *rule-based* (Neviarouskaya, et al., 2011; Anusha & Sandhya, 2015).

Deteksi emosi dari teks merupakan persoalan untuk mencari hubungan antara suatu teks dengan suatu jenis emosi tertentu. Proses deteksi emosi dari teks menerima masukan berupa teks dan label emosi. Selanjutnya menggunakan suatu fungsi maka akan dihasilkan pasangan teks dan label emosi yang sesuai.

### 2.4. Pendekatan *Learning Based*

Pada pendekatan *learning based*, deteksi emosi dipandang sebagai klasifikasi (Anusha & Sandhya, 2015). Tujuan utamanya adalah mencari fungsi  $f$  yang dapat memetakan suatu teks  $s$  dengan jenis emosi  $em$  seperti pada persamaan berikut:

$$f: s \rightarrow em \quad (1)$$

Hasil dari fungsi  $f$  adalah pasangan teks  $s$  dengan jenis emosi  $em$  yang sesuai ( $s, em$ ). Fungsi  $f$  tersebut juga disebut sebagai *classifier*. Fungsi ini diperoleh dengan melakukan pembelajaran menggunakan sejumlah data latih yang disebut *corpus*. Sebelum melakukan proses pembelajaran, *corpus* yang akan digunakan terlebih dahulu melewati pengolahan bahasa alami atau biasa disebut dengan *Natural Language Processing* (NLP). Hal ini untuk membuat komputer mampu memahami teks dalam bentuk bahasa sehari-hari yang ada di dalam *corpus*.

Pendekatan *learning-based* bergantung pada metode dan *corpus* yang digunakan dalam proses pembelajaran (Binali & Potdar, 2012). Secara umum metode pembelajaran yang digunakan terdiri dari dua yaitu terbimbing (*supervised*) dan tidak terbimbing (*unsupervised*). Pada pembelajaran terbimbing (*supervised*), *corpus* yang digunakan adalah data teks yang memiliki label emosi (Anusha & Sandhya, 2015). *Corpus* ini digunakan untuk melatih *classifier* seperti *Support Vector Machine* (SVM). Sedangkan pada pembelajaran tidak terbimbing (*unsupervised*) *corpus* yang digunakan tidak memiliki label emosi (Anusha & Sandhya, 2015). Metode seperti *Latent Semantic Analysis* (LSA) digunakan untuk membangun model deteksi dari data yang tidak memiliki label emosi.

### 2.5. *Natural Language Processing* (NLP)

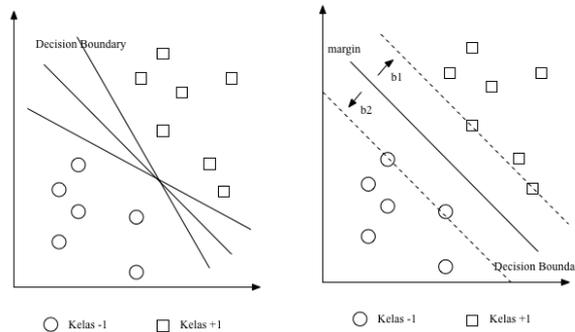
Pengolahan bahasa alami atau biasa disebut dengan *Natural Language Processing* adalah bidang ilmu komputer dan linguistik yang berkaitan dengan interaksi antara komputer dan manusia. *Natural Language Processing*, biasanya disingkat dengan NLP, membuat komputer mampu memahami suatu perintah yang dituliskan dalam bentuk bahasa sehari-hari dan diharapkan komputer juga mampu merespon dalam bahasa yang mirip dengan bahasa sehari-hari.

Sebuah sistem NLP harus memperhatikan pengetahuan terhadap bahasa itu sendiri seperti bagaimana kata-kata tersebut digabung untuk menghasilkan suatu kalimat, apa arti suatu kata, apa fungsi sebuah kata dalam kalimat dan sebagainya. NLP tidak bertujuan untuk mentransformasikan bahasa yang diterima dalam bentuk teks atau suara menjadi data digital atau sebaliknya, tapi bertujuan memahami arti dari kalimat dalam bahasa alami dan memberikan respon yang sesuai. Untuk mencapai tujuan ini dibutuhkan *text preprocessing* dan *feature extraction*.

### 2.6. *Support Vector Machine* (SVM)

Ide dasar SVM adalah memaksimalkan batas *hyperplane*, yang diilustrasikan pada Gambar 2.1. Pada gambar (a) ada sejumlah pilihan *hyperplane* yang mungkin untuk set data, sedangkan gambar (b) merupakan *hyperplane* dengan margin maksimal. Meskipun sebenarnya pada gambar (a) bisa juga menggunakan *hyperplane* sembarang, tetapi *hyperplane* dengan margin yang maksimal akan memberikan generalisasi yang lebih baik pada metode klasifikasi (Prasetyo, 2014). Kekuatan SVM berasal dari kemampuannya untuk mempelajari pola klasifikasi data

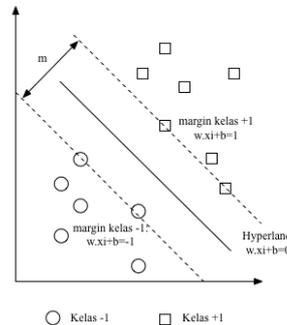
dengan akurasi yang seimbang. SVM telah menjadi alat yang banyak digunakan untuk klasifikasi, dengan fleksibilitas tinggi yang mencakup berbagai skenario ilmu data, termasuk penelitian gangguan otak (Pisner & Schnyer, 2020).



Gambar 2.1. Ilustrasi Batas Keputusan yang Mungkin untuk Set Data (Prasetyo, 2014)

Konsep klasifikasi dengan SVM dapat dijelaskan secara sederhana sebagai usaha mencari *hyperplane* terbaik yang berfungsi sebagai pemisah dua buah kelas data pada *input space*. Gambar 2.1 memperlihatkan beberapa data yang merupakan anggota dari dua buah kelas data, yaitu +1 dan -1. Data yang tergabung pada kelas -1 disimbolkan dengan bentuk lingkaran, sedangkan data pada kelas +1, disimbolkan dengan bujur sangkar (Prasetyo, 2014).

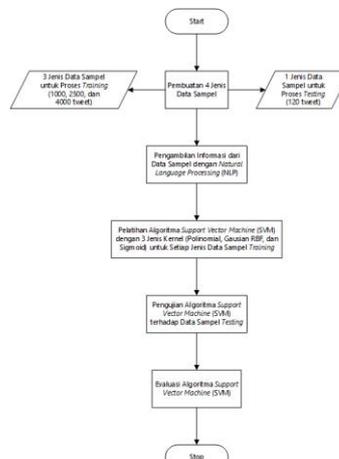
*Hyperplane* (batas keputusan) pemisah terbaik antara kedua kelas dapat ditemukan dengan mengukur margin *hyperplane* tersebut dan mencari titik maksimalnya. Margin adalah jarak antara *hyperplane* tersebut dengan data terdekat dari masing – masing kelas. Data yang paling dekat ini disebut sebagai *support vector*. Garis solid pada gambar 2.1 (b) sebelah kanan menunjukkan *hyperplane* yang terbaik, yaitu yang terletak tepat di tengah – tengah kedua kelas, sedangkan data lingkaran dan bujur sangkar yang dilewati garis batas margin (garis putus-putus) adalah *support vector*. Usaha untuk mencari lokasi *hyperplane* ini merupakan inti dari proses pelatihan pada SVM (Prasetyo, 2014).



Gambar 2.2. Ilustrasi Margin Hyperplane (Prasetyo, 2014)

### 3. Metodologi

#### 3.1 Diagram Alir Penelitian



Gambar 3.1. Diagram Alir Penelitian

### 3.2 Membuat Jenis Data Sampel

Pada Gambar 3.1 dapat dilihat bahwa proses pertama adalah pembuatan data sampel. Data yang akan diproses ke dalam algoritma merupakan data sampel berupa *tweet* dari Twitter dalam jumlah banyak dan telah diberikan kategori emosi (senang, sedih, terkejut, marah, takut dan jijik) untuk setiap *tweet*. Data sampel yang dibuat akan terdapat 4 jenis, yaitu data sampel untuk proses *training* sebanyak 3 jenis dengan jumlah data yang berbeda, yaitu 1000, 2500, dan 4000 *tweet*, serta data sampel untuk proses *testing* dengan jumlah data 120 *tweet*.

### 3.3 Mengekstrak Informasi dengan NLP

Data sampel yang telah dibuat akan melalui proses *Natural Language Processing* (NLP) terlebih dahulu dengan melalui tahapan berikut ini:

1. *Case Folding*  
Mengubah semua karakter huruf pada setiap *tweet* di data sampel menjadi huruf kecil (*lowercase*).
2. *Tokenizing*  
Setelah *case folding* setiap *tweet* di data sampel berupa kalimat-kalimat diubah menjadi kata-kata dan menghilangkan alamat link, nama akun pengguna, dan tanda baca seperti tanda titik (.), koma (,), spasi dan karakter angka yang ada pada kata.
3. *Stopword Removal*  
Setelah *tokenizing* kata-kata yang tidak mengandung emosi dihapus berdasarkan *stoplist* yang ada, contohnya “yang”, “dan”, “di”, “dari” dan lain-lain.
4. *Stemming*  
Setelah itu kata-kata diurai menjadi bentuk kata dasarnya (*stem*), bertujuan menghilangkan imbuhan-imbuhan berupa prefiks, sufiks, dan konfiks di setiap kata.
5. *Feature Extraction*  
Terakhir kata-kata tersebut dengan metode TF-IDF diekstrak informasi yang penting untuk digunakan dalam proses *training* dengan algoritma SVM.

### 3.4 Melatih Algoritma SVM

Kemudian algoritma SVM akan melalui proses *training* dengan cara berikut:

1. Informasi yang diekstrak pada proses *Natural Language Processing* (NLP) akan dipetakan dengan menggunakan salah satu dari 3 tipe kernel, yaitu *Polynomial*, *Gaussian RBF*, atau *Sigmoid*.
2. Lalu algoritma SVM akan melakukan proses *training* dengan menggunakan salah satu dari 3 jenis data sampel dengan jumlah data yang berbeda, yaitu 1000, 2500, dan 4000 *tweet*.
3. Ulangi tahap 1 dan 2 dengan kombinasi tipe kernel dan jenis data sampel yang berbeda.

### 3.5 Menguji Algoritma SVM

Pengujian akurasi algoritma SVM bertujuan untuk menguji berapa akurat algoritma SVM setelah proses *training* terhadap data *testing* yang dimasukkan. Proses pengujian tersebut adalah sebagai berikut:

1. Algoritma SVM yang sudah melalui proses *training* akan dilakukan proses *testing* terhadap data *testing* dengan jumlah data yaitu 120 *tweet*.
2. Keluaran yang dihasilkan dari proses *testing* pada tahap 1 akan dibandingkan dengan keluaran yang seharusnya didapatkan, kemudian data keluaran yang benar akan dihitung.
3. Hasil pengujian algoritma SVM akan menghasilkan sebuah nilai berbentuk persentase keakuratan algoritma SVM. Hasil uji 100% berarti algoritma sangat akurat, hasil uji 0% berarti algoritma sangat tidak akurat.

### 3.6 Mengevaluasi Algoritma SVM

Setelah proses pelatihan dan pengujian, maka algoritma SVM dapat dievaluasi. Evaluasi terhadap algoritma SVM mencakup beberapa aspek, sebagai berikut:

1. Lama waktu *training*  
Lama waktu *training* dipengaruhi oleh kompleksitas kernel dan metode yang digunakan dalam konfigurasi algoritma.
2. Akurasi dari Algoritma SVM  
Algoritma akan menghasilkan klasifikasi dengan keakuratan tertentu. Akurasi tersebut dapat digunakan sebagai tolak ukur kebergunaan algoritma di kasus nyata.

Masing-masing aspek tersebut dipengaruhi oleh konfigurasi yang disebutkan dalam tahap sebelumnya sehingga perlu proses pelatihan dan pengujian algoritma SVM yang berulang-ulang agar seluruh data dapat dikumpulkan dan dibandingkan.

## 4. Hasil dan Analisis

### 4.1 Deskripsi Hasil penelitian

Penelitian dilakukan dengan mengambil sumber data dari *tweet* masyarakat yang dapat diakses secara mudah melalui situs *twitter.com*, data tersebut akan diolah sehingga menghasilkan data sampel untuk nantinya digunakan sebagai dasar mendeteksi emosi dari teks bahasa Indonesia. Penelitian ini akan menghasilkan analisis mengenai kinerja dari algoritma *Support Vector Machine (SVM)* dalam mendeteksi emosi dari teks bahasa Indonesia.

### 4.2 Hasil Penelitian dan Analisis

#### 4.2.1 Pengambilan Data

Dalam pembuatan data sampel peneliti menggunakan *library* Tweepy untuk melakukan penarikan data *tweet* berdasarkan 5 trending topik yang ada di platform Twitter sebagai sumber data mentah. Kemudian peneliti melakukan pelabelan kategori emosi untuk setiap *tweet* di sumber data mentah sebelumnya sehingga menjadi data sampel yang utuh.

Hasil dari data sampel yang utuh tersebut, kemudian peneliti membagi data sampel menjadi 3 jenis data sampel yang akan dijadikan sebagai data *training*, yaitu data sampel dengan jumlah 1000, 2500, dan 4000 *tweet*. Selain itu, peneliti juga membuat satu buah data sampel yang dijadikan sebagai data *testing* dengan jumlah 120 *tweet*. Sehingga data sampel yang akan dihasilkan berjumlah 4 jenis, yaitu 3 jenis untuk data *training*, serta 1 jenis untuk data *testing*.

#### 4.2.2 Mengekstrak Informasi dengan NLP

Keseluruhan jenis data sampel yang telah dibuat akan melalui proses *Natural Language Processing (NLP)* terlebih dahulu yaitu, *Case Folding*, *Tokenizing*, *Stopword Removal*, *Stemming*, dan *Feature Extraction*. Untuk proses *Stopword Removal*, dan *Stemming* menggunakan *library* Sastrawi, sementara untuk proses *Feature Extraction* menggunakan *library* NLTK.

Berikut adalah waktu yang dibutuhkan untuk menyelesaikan keseluruhan proses NLP berdasarkan jenis data sampel dalam Tabel 4.1.

Tabel 4.1. Perbandingan Waktu Proses NLP per Jenis Data Sampel

Jenis Data Sampel	Waktu (Detik)
Data <i>Training</i> (1000 <i>tweet</i> )	219.566
Data <i>Training</i> (2500 <i>tweet</i> )	511.798
Data <i>Training</i> (4000 <i>tweet</i> )	956.363
Data <i>Testing</i>	26.04

#### 4.2.3 Melatih Algoritma SVM

Pada tahapan ini 3 jenis data sampel untuk data *training* yang sudah diekstrak informasinya dengan proses NLP akan melalui proses *training* dengan algoritma SVM yang menggunakan 3 tipe kernel yang berbeda untuk setiap jenis data sampel, yaitu *Polinomial*, *Gaussian RBF*, dan *Sigmoid*. Berikut adalah waktu yang dibutuhkan untuk menyelesaikan proses *training* berdasarkan jenis data sampel dan tipe kernel yang digunakan dalam Tabel 4.2.

Tabel 4.2. Perbandingan Waktu Proses *Training* per Jenis Data Sampel dan Tipe Kernel

Jenis Data Sampel	Tipe Kernel	Waktu (Detik)
Data <i>Training</i> (1000 <i>tweet</i> )	<i>Polinomial</i>	0.154
	<i>Gaussian RBF</i>	0.154
	<i>Sigmoid</i>	0.142
Data <i>Training</i> (2500 <i>tweet</i> )	<i>Polinomial</i>	1.098
	<i>Gaussian RBF</i>	1.098
	<i>Sigmoid</i>	0.854
Data <i>Training</i> (4000 <i>tweet</i> )	<i>Polinomial</i>	5.209
	<i>Gaussian RBF</i>	4.696
	<i>Sigmoid</i>	3.134

#### 4.2.4 Menguji Algoritma SVM

Setelah proses *training* dengan algoritma SVM yang menggunakan 3 tipe kernel yang berbeda untuk setiap jenis data sampel selesai, kemudian peneliti melakukan pengujian akurasi algoritma SVM dengan jenis data sampel untuk data *testing* yang sudah diekstrak informasinya dengan proses NLP sebelumnya. Tahapan ini bertujuan untuk menguji seberapa akurat algoritma SVM setelah proses *training* terhadap data *testing* yang dimasukkan. Berikut adalah tingkat akurasi algoritma SVM yang didapatkan berdasarkan jenis data sampel dan tipe kernel yang digunakan dalam Tabel 4.3.

Tabel 4.3. Perbandingan Tingkat Akurasi Algoritma SVM per Jenis Data Sampel dan Tipe Kernel

Jenis Data Sampel	Tipe Kernel	Akurasi (%)
Data Training (1000 tweet)	<i>Polinomial</i>	59.166
	<i>Gaussian RBF</i>	65.833
	<i>Sigmoid</i>	70.833
Data Training (2500 tweet)	<i>Polinomial</i>	70.833
	<i>Gaussian RBF</i>	79.166
	<i>Sigmoid</i>	80
Data Training (4000 tweet)	<i>Polinomial</i>	82.5
	<i>Gaussian RBF</i>	84.1666
	<i>Sigmoid</i>	87.5

#### 4.3 Pembahasan

Penerapan algoritma *Support Vector Machine* (SVM) untuk mendeteksi emosi dari teks bahasa Indonesia dimulai dengan membuat data sampel. Dalam pembuatan data sampel digunakan *library* Tweepy untuk melakukan penarikan data *tweet* berdasarkan 5 trending topik yang ada di platform Twitter sebagai sumber data mentah. Kemudian data mentah tersebut dilakukan pelabelan kategori emosi untuk setiap *tweet* sehingga menjadi data sampel yang utuh. Dari data sampel yang utuh tersebut, kemudian dibagi 3 jenis data sampel yang akan dijadikan sebagai data *training*, yaitu data sampel dengan jumlah 1000, 2500, dan 4000 *tweet*. Selain itu, perlu juga dibuat satu buah data sampel yang dijadikan sebagai data *testing* dengan jumlah 120 *tweet*. Sehingga data sampel yang akan dihasilkan berjumlah 4 jenis, yaitu 3 jenis untuk data *training*, serta 1 jenis untuk data *testing*.

Kemudian seluruh jenis data sampel yang telah dibuat akan melalui proses *Natural Language Processing* (NLP) terlebih dahulu yaitu, *Case Folding*, *Tokenizing*, *Stopword Removal*, *Stemming*, dan *Feature Extraction*. Untuk proses *Stopword Removal*, dan *Stemming* menggunakan *library* Sastrawi, sementara untuk proses *Feature Extraction* menggunakan *library* NLTK. Proses NLP ini bertujuan untuk mengekstrak informasi dari setiap *tweet* untuk setiap jenis data sampel.

Setelah informasi dari setiap jenis data sampel diekstrak, lalu 3 jenis data sampel untuk data *training* akan melalui proses *training* dengan algoritma SVM yang menggunakan 3 tipe kernel yang berbeda untuk setiap jenis data sampel, yaitu *Polinomial*, *Gaussian RBF*, dan *Sigmoid*. Waktu yang dibutuhkan untuk proses *training* tersebut bervariasi tergantung dari jumlah data di setiap jenis data sampel serta tipe kernel yang digunakan. Dari keseluruhan proses *training*, dapat dilihat pada tabel 4.2. semakin banyak jumlah data akan membutuhkan waktu *training* yang lebih lama, serta dari setiap tipe kernel yang digunakan, tipe kernel *Sigmoid* membutuhkan waktu *training* yang lebih cepat dibandingkan tipe kernel lainnya.

Setelah proses *training* dengan algoritma SVM yang menggunakan 3 tipe kernel yang berbeda untuk setiap jenis data sampel selesai, kemudian dilakukan pengujian akurasi algoritma SVM dengan jenis data sampel untuk data *testing* yang sudah diekstrak informasinya. Dari keseluruhan pengujian, dapat dilihat pada tabel 4.3. semakin banyak jumlah data akan meningkatkan tingkat akurasi dari algoritma SVM tersebut, serta dari setiap tipe kernel yang digunakan, tipe kernel *Sigmoid* memiliki tingkat akurasi yang lebih baik dibandingkan dengan tipe kernel lainnya.

### 5. Kesimpulan dan Saran

#### 5.1 Kesimpulan

Setelah dilakukan penelitian dengan judul Penerapan Algoritma *Support Vector Machine* untuk Mendeteksi Emosi Dari Teks Bahasa Indonesia, yang dimulai dengan pembuatan 4 jenis data sampel, kemudian data sampel tersebut melalui proses NLP untuk diekstrak informasinya. Setelah itu dilakukan proses *training* dengan algoritma SVM menggunakan 3 tipe kernel yang berbeda, yaitu *Polinomial*, *Gaussian RBF*, dan *Sigmoid*. Kemudian dilakukan pengujian dengan hasil tingkat akurasi yang cukup variatif untuk setiap jenis data sampel serta tipe kernel yang digunakan. Sehingga dapat menghasilkan kesimpulan semakin banyak jumlah *tweet* pada data sampel yang digunakan, semakin tinggi tingkat akurasi yang dihasilkan, namun waktu proses *training* yang dibutuhkan

juga lebih lama. Sementara untuk tipe kernel *Sigmoid* memiliki tingkat akurasi lebih tinggi serta waktu proses *training* juga lebih cepat dibandingkan dengan tipe kernel lainnya.

## 5.2 Saran

Penulis memiliki beberapa saran untuk penelitian serupa yang berkaitan dengan penerapan algoritma untuk deteksi emosi pada teks bahasa Indonesia, yaitu:

1. Untuk penghematan waktu dalam penelitian, perlu adanya kumpulan data berukuran besar yang berisi teks bahasa Indonesia dengan label emosi dan dapat diakses secara terbuka yang dapat digunakan dalam penelitian lainnya.
2. Untuk penggunaan *library* Sastrawi perlu ditambahkan kata-kata *stopword* baru pada kode pemrogramannya untuk meningkatkan ketepatan dalam proses *Stopword Removal*, dan *Stemming*.

## Daftar Pustaka:

- Lopatovska, L., & Arapakis, I. (2011). *Theories, methods and current research on emotions in library and information science, information retrieval and human-computer interaction. Information Processing and Management*, 47:575-592.
- Paltoglou, G., Theunis, M., Kappas, A., & Thelwall, M. (2013). *Predicting emotional responses to long informal text. IEEE Transactions*, 4:106-115.
- Quan, C., & Ren, F. (2016). *Visualizing emotions from chinese blogs by textual emotion analysis and recognition techniques. International Journal of Information Technology & Decision Making*, 15:215-234.
- Rahulamathavn, Y., Phan, R., Chambers, J., & Parish, D. (2012). *Facial expression recognition in the encrypted domain based on local fisher discriminant analysis. IEEE Transactions*, 4:83-92.
- Pfister, T., & Robinson, P. (2011). *Real-time recognition of affective states from nonverbal features of speech and its application for public speaking skill analysis. IEEE Transactions*, 2:66-78.
- Kleinsmith, A., & Bianchi-Berthouze, N. (2013). *Affective body expression perception and recognition: a survey. IEEE Transactions*, 4:15-33.
- Soleymani, M., et al. (2012). *DEAP: a database for emotion analysis; using physiological signals. IEEE Transactions*, 3:18-31.
- Calvo, R.A., & D’Mello, S. (2010). *Affect detection: an interdisciplinary review of models, methods, and their applications. IEEE Transactions*, 1:18-37.
- Mohammad, S.M., & Turney, P.D. (2013). *Crowdsourcing a word–emotion association lexicon. Computational Intelligence*, 29:436-465.
- Neviarouskaya, A., Prendinger, H., & Ishizuka, M. (2011). *Affect analysis model: novel rule-based approach to affect sensing from text. Natural Language Engineering*, 17:95-135.
- Paltoglou, G., & Thelwall, M. (2013). *Seeing stars of valence and arousal in blog posts. IEEE Transactions*, 4:116-123.
- Ekman, P. (1972). *Universals and cultural differences in facial expressions of emotions. Nebraska Symposium*, 53:207-282.
- Calvo, R.A., & Kim, M. (2013). *Emotions in text: dimensional and categorical models. Computational Intelligence*, 29:527-543.
- Peter, C., & Herbon, A. (2006). *Emotion representation and physiology assignments in digital systems. Interacting with Computers*, 18:139-170.
- Anusha V., & Sandhya, B. (2015). *A learning-based emotion classifier with semantic text processing. Intelligent Informatics*, 320:371-382.
- D’Mello, S.K., Picard, R.W., & Graesser, A.C. (2007). *Towards an affect-sensitive autotutor. IEEE Intelligent Systems*, 22:53-61.
- Picard, R.W. (1997). *Affective Computing*. Massachusetts: MIT Press Cambridge.
- Pisner, D. A., & Schnyer, D. M. (2020). *Support vector machine. In Machine learning* (pp. 101-121). Academic Press.
- Luhulima, Yudasha, Y., Marji, & Muflikhah, L. (2015). *Sentiment Analysis pada Review Barang Berbahasa Indonesia dengan Metode K-Nearest Neighbor (KNN)*. Malang: Universitas Brawijaya.
- Prasetyo, E. (2014). *Data Mining*. Yogyakarta: Andi Offset.
- Vapnik, V. (2006). *Estimation of Dependences Based on Empirical Data*. New York: Springer-Verlag.
- Ghazi, D., Inkpen, D., & Szpakowicz, S. (2010). *Hierarchical Approach to Emotion Recognition and Classification in Texts*. Ottawa: University of Ottawa.
- Krcadinac, U., Jovanovic, J., Devedzic, V., & Pasquier, P. (2016). *Textual affect communication and evocation using abstract generative visuals. IEEE Transactions*, 46:370-379.

- Tiara. (2015). Analisis Sentimen pada Twitter untuk Menilai Performansi Program Televisi dengan Kombinasi Metode *Lexicon-Based* dan *Support Vector Machine*. Bandung: Telkom University.
- Utama, P., Widodo, & Ajie, H. (2014). *A Framework of Human Emotion Recognition Using Extreme Learning Machine*. Bandung: *International Conference of Advanced Informatics: Concept, Theory and Application (ICAICTA)*.
- Nugroho, R.A. (2019). Pengembangan Algoritma Jaringan Saraf Tiruan dengan Pendekatan *Deep Learning* untuk Sistem Rekomendasi Peminatan Program Studi Pendidikan Teknik.