



Perbandingan Algoritma Sentimen Analisis Media Data Twitter Pilgub Jabar 2018

Amali

Program Studi Teknik Informatika, Universitas Pelita Bangsa
Korespondensi email: amali@pelitabangsa.ac.id

Abstrak	Informasi Artikel
<p><i>This paper is published to review and compare the performance of the SVM algorithms and CNN algorithms as an update composition in analyzing sentiment with tweeter attributes, the comparison of these algorithms using the Python application as a tool to support machine learning. The classification of negative, neutral and positive sentiments in the tweet dataset is tested and to determine and measure the accuracy, precision, recall, f_Measure and configuration matrix weights of both the SVM algorithms and CNN algorithms. The tools used with the Python Jupyter application, Tensorflow, Noto ++ are applied to the Indonesian language Twitter classification, the measurement results are precise and accurate according to human measurement parameters related to tweet data sentiment on Twitter social media commenting on the election of the Governor of West Java with the candidate for governor and deputy governor for the period 2018-2023. The results of this study, Testing Experiments before stemming with the SVM Algorithm was carried out seven times with an average accuracy rate of around 67%, and the CNN Algorithm before stemming also with seven trials with an average accuracy of around 67%, then the Testing Experiment after stemming. with SVM conducted seven trials the average accuracy rate was around 67%, while CNN algorithms before stemming was also carried out with seven trials with an average accuracy of about 52% lower than SVM algorithms.</i></p>	<p>Diterima: 25-02-2020 Direvisi: 27-13-2020 Dipublikasikan: 30-04-2020</p> <hr/> <p>Keywords Keyword : Sentiment Analysis, Twitter, Governor Election, SVM (Support Vector Machine), (CNN) Convolutional Neural Network</p>

I. Pendahuluan

Berbagai latar belakang pendidikan, strata sosial dan beragam profesi pekerjaan dalam lapisan masyarakat yang terbuka mudah berinteraksi di media sosial Twitter menjadi salah satu media untuk menjangkau kemenangan pemilihan kepala daerah Gubernur dan Wakil Gubernur dengan pembentukan opini, sehingga jika yang muncul nya untuk opini negatif, opini netral atau opini positif sangat berpengaruh kepada Calon Gubernur dan Wakil Gubernur Jawa barat periode 2018-2023

II Metodologi

Sentimen Analisis adalah studi komputasi terkait pada pendapat, penilaian, emosi, sikap, pandangan dari sekumpulan text, ekspresi gambar dan kombinasi yang ada di dalam media yang berfokus dalam mengekstrasi, mengidentifikasi, menilai karakteristik dalam unit teks menggunakan machine learning. Penelitian ini menggunakan dan membandingkan kinerja algoritma SVM dan CNN dalam Proses pemilihan kepala daerah provinsi Jawa Barat tahun 2018 dengan data resource tweeter, bentuk kombinasi yang di lakukan untuk melihat hasil kebaruan dari penelitian sejenis yang pernah ada di Indonesia.

Hasil yang didapatkan sebagai berikut :

Step 1 Percobaan Pengujian sebelum proses stemming dengan menggunakan Algoritma SVM dilakukan sebanyak Tujuh Kali percobaan dengan tingkat akurasi rata-rata di sekitar 69.98%, begitupun untuk Percobaan dengan Menggunakan CNN sebelum proses stemming dengan sebanyak Tujuh kali percobaan dengan akurasi rata-rata disekitar 66.08 %

Step 2 Percobaan Pengujian setelah proses stemming dengan menggunakan Algoritma SVM dilakukan sebanyak Tujuh Kali percobaan dengan tingkat akurasi rata-rata di sekitar 68.61%,

begitupun untuk Percobaan dengan Menggunakan CNN sebelum proses stemming dengan sebanyak Tujuh kali percobaan dengan akurasi rata-rata disekitar 67.41% sedikit lebih rendah dari SVM

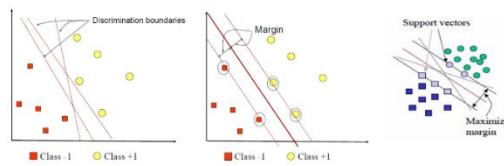
A. Dari Tujuh kali pengujian dengan menggunakan Algoritma SVM akurasi tertinggi di dengan data Training (latih) sebanyak 20% dan data Uji sebanyak 80% dengan nilai akurasi 72% pada saat sebelum proses stemming tetapi menjadi sedikit lebih rendah pada saat setelah stemming menjadi 71%, tetapi berbeda dengan Percobaan dengan menggunakan CNN akurasi tertinggi didapatkan dengan komposisi Data Latih 20% dan Data Uji sebanyak 80% pada saat setelah proses Stemming dengan Nilai akurasi 71% pada saat sebelum stemming nilai akurasi nya hanya di sekitar 67%.

B. Algoritma CNN akan memiliki akurasi yang lebih baik jika data set yang digunakan semakin banyak dibandingkan dengan Algoritma SVM yang cenderung tidak banyak perubahan meski ada penambahan data set dari setiap proses pengujian yang dilakukan.

Untuk mengukur kinerja SVM dan CNN ada beberapa proses yang di perlukan dalam memproses data latih (data training) dan data uji (data test).

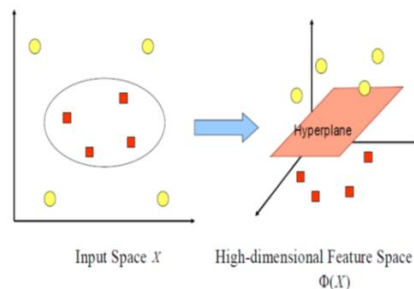
Algoritma SVM (Support Vector Machine) adalah pertama kali dikembangkan oleh Boser, Guyon dan Vapnik pada tahun 1992 ketika dilakukan Workshop on Computational Learning Theory.

SVM merupakan pembelajaran terbimbing yang pengklasifikasiannya menggunakan hipotesis berupa fungsi fungsi linier dalam sebuah ruang fitur (feature space) berdimensi tinggi [4][5]



Gambar 1. Model Klasifikasi SVM (*Support Vector Machine*)

Data yang berada pada bidang pembatas disebut dengan support vector. Dalam Gambar 1 dua kelas dapat dipisahkan oleh sepasang bidang pembatas yang sejajar. Bidang pembatas pertama membatasi kelas pertama sedangkan bidang pembatas kedua membatasi kelas kedua.



Gambar 2. Fungsi Φ memetakan data ke ruang vector yang berdimensi tinggi

Support Vector Machine adalah Algoritma teknik pembelajaran mesin (machine learning) paling mutakhir dibandingkan dengan mesin pembelajaran sebelumnya yang dikenal dengan *neural network*, keduanya berhasil digunakan dalam pengenalan pola dan pembelajaran. [8]. Beberapa kelemahan yang dimiliki SVM antara lain:

1. Sulit dipakai dalam problem berskala besar. Skala besar dalam hal ini dimaksudkan dengan jumlah sample yang diolah.
2. SVM secara teoritik dikembangkan untuk problem klasifikasi dengan dua class. SVM saat ini telah dimodifikasi agar dapat menyelesaikan masalah dengan class lebih dari dua, antara lain strategi *One versus rest* dan strategi

Tree Structure. Namun demikian, masing-masing strategi ini memiliki kelemahan, sehingga dapat dikatakan penelitian dan pengembangan SVM pada *multiclass-problem* masih merupakan tema penelitian yang masih terbuka untuk terus dikembangkan.[7]

Kebijakan dari Algoritma SVM (Support Vector Machine) dalam Generalisasi, yang mampu mengklasifikasi yang di dalam fase pembelajaran Algoritma tersebut dengan data yang tidak dipakai dan juga mampu meminimalkan *error* pada *training dataset* serta meminimalkan faktor yang di pengaruhi dimensi VC (*Vapnik-Chervokinensi*). proses estimasi ini menjadi sangat sulit.

Kekurangan dari Algoritma SVM (Support Vector Machine) diantaranya :

- a. Dalam hal jumlah sample yang diolah memiliki problem berskala besar maka proses dengan SVM menjadi lebih sulit
- b. Secara teoritis SVM dikembangkan untuk mengatasi permasalahan problem klasifikasi dengan dua kelas dan SVM saat ini telah di modifikasi untuk bisa menyelesaikan masalah dua kelas (*class*) dengan dua macam strategi yaitu : *Strategi One versus Rest* dan *Strategi Tree Structure*. Di sini masih ada kelemahan [7]

1. Algoritma CNN (Convolutional Neural Network) adalah untuk mengolah data dua dimensi dengan *Multilayer Perceptron (MLP)*, juga termasuk dalam jenis *DNN (Deep Neural Network)* yang cocok diaplikasikan pada kasus citra dimana DNN memiliki kedalaman jaringan yang tinggi serta banyak diaplikasikan pada data citra.

Di dalam kasus Klasifikasi Citra dan MLP karena tidak menyimpan informasi spasial ditambah dengan tidak menganggap setiap

piksel adalah suatu independen fitur MLP menghasilkan hasil yang kurang baik.

Convolutional Neural Network (CNN) terdiri dari satu atau lebih lapisan konvolusi dengan arsitektur yang dirancang untuk memanfaatkan struktur 2D dari gambar masukan (atau input 2D lainnya seperti sinyal ucapan). Manfaat lain dari CNN adalah bahwa mereka lebih mudah untuk melatih dan memiliki lebih sedikit parameter daripada jaringan yang terhubung sepenuhnya dengan jumlah unit tersembunyi yang sama.

Tabel 1 Perbandingan Algoritma SVM dan CNN

SVM	CNN
SVM adalah pengklasifikasi linier	CNN tidak linier
SVM dengan kernel adalah klasifikasi non linier	CNN bekerja dengan baik dengan data yang memiliki pola berulang secara special, gambar, ucapan
Tidak banyak pembatasan pada jenis data yang akan bekerja dengannya	CNN classifier fitur vector berguna dalam banyak masalah lainnya
Tidak ada cara untuk meningkatkan kompleksitas model	Representasi yang sangat bagus

2. Pengambilan data dan Algoritma yang di gunakan

Penelitian ini secara garis besar mirip dengan penelitian sentimen analisis lain yang pernah di lakukan, kebaruannya adalah untuk melihat hasil perbandingan kinerja kedua Algoritma SVM dan CNN terhadap klasifikasi sentimen yang ada pada tweeter dalam bahasa indonesia yang datasetnya diambil dari media sosial twitter dan hanya terkait hanya pada seputar pemilihan umum kepala daerah provinsi jawa barat 2018.

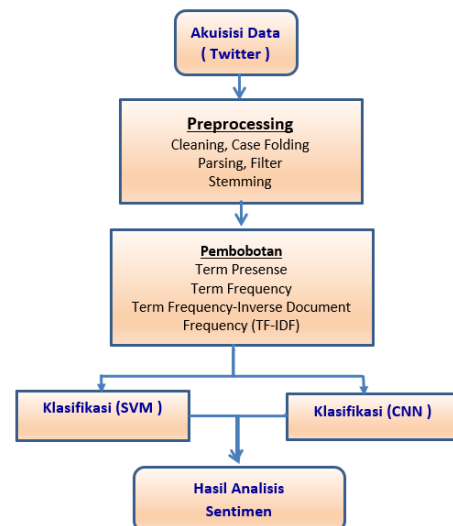
III Analisis Perancangan Sistem

3.1 Perancangan Sistem

Pada penelitian ini, perancangan sistem digambarkan dalam kerangka atau pola dasar sistem yang akan di bangun

Parameter Klasifikasi penentuan awal sentimen menggunakan standard kamus: <https://github.com/masdevi/ID-OpinionWords/blob/master/positive.txt> dan <https://github.com/masdevi/ID-OpinionWords/blob/master/negative.txt>.

Proses dilakukan dengan formula excel file.



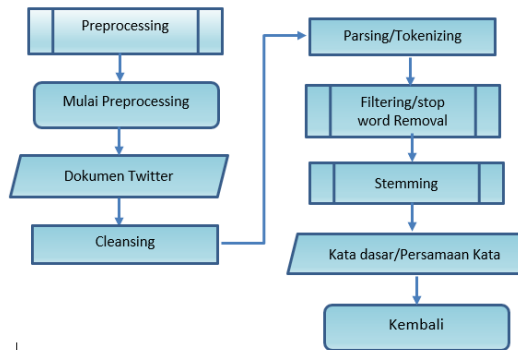
Gambar 3. Rancangan Sistem Klasifikasi

Dalam gambar 3 dijelaskan proses untuk tahapan perancangan sistem penelitian dimulai dengan tahapan akuisisi *Twitter* dilanjutkan pada proses *Preprocessing* (tahapan memproses data atau dokumen) pada tahapan ini data di akuisisi atau di kumpulkan dari situs jejaring media sosial *twitter*.

Dalam tahapan *preprocessing* terdapat bagian *cleansing*, *case folding*, *parsing / tokenizing*, *filtering* kemudian tahapan *stemming* untuk mendapatkan kata dasar yang akan di klasifikasi kedalam ketiga kelas. Tahapan selanjutnya adalah

pembobotan dengan menggunakan term of presense (TP), term of Frequency (TF) dan term of frequency-inverse document frequency (TF-IDF). Tahapan berikutnya proses klasifikasi SVM dan CNN

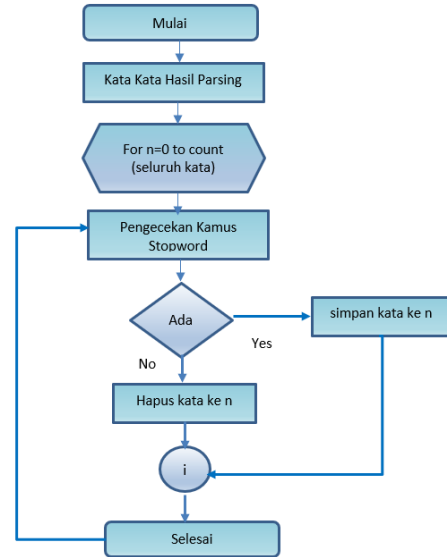
3.2 Diagram Alir Processing



Gambar 4. Diagram alir processing

Pada Gambar 3.2 dijelaskan tahapan data atau dokumen setelah di ekstrasing selanjutnya ke pembersihan (*cleansing*) untuk mengurangi *noise*, berikutnya di lakukan *case folding* menyeragamkan huruf huruf kemudian dilakukan proses *parsing tokenizing*, dokumen menjadi *term-term* berdasarkan spasi *stopword* kemudian dilakukan *filtering / stopwords removal* untuk menyaring kata atau dokumen, lalu dilakukan proses *stemming* dilakukan berulang kali untuk mendapatkan kata dasar sesuai KBBI.

3.3 Diagram Alur Filtering/Stopword Removal



Gambar 5. Diagram Alur Filtering/Stopword Removal

Pada Gambar 3.3 tweet yang sudah *parsing /tokenizing* di lakukan *filtering* untuk menghilangkan kata kata yang tidak relevan dengan *stopword removal*, kemudian menghapus kata yang tidak ada dalam data base. Dan untuk kata sebaliknya disimpan kedalam *database*.

3.4 Pembobotan

3.4.1 Term Weighting

Pembobotan bertujuan untuk mendapatkan nilai data *term /kata (TF-IDF method)* setelah melewati proses ekstrasi dan *preprocessing*, kemudian dokumen diubah menjadi sebuah vektor dengan banyak *term* (proses pebobotan).

3.4.2 Term Presense (TP)

Proses ini untuk menghitung jumlah kemunculan kata dalam dokumen. *Term Frequency* di dasari pada aspek *TF-IDF monolicity*, mendapatkan nilai TF dapat dilakukan dengan cara sebagai berikut:

- *Raw TF TF*, sebuah term dilakukan

perhitungan berdasarkan probabilitas kemunculan *term*/kata tersebut dalam dokumen

- *Logarithmic T*, nilai TF didapatkan dengan menggunakan fungsi logaritmik pada

$$tf = 1 + \log(tf) \quad (1)$$

dimana *tf* merupakan kemunculan *term*/kata dalam sebuah dokumen

- *Binary TF* *TF*, nilai yang diberikan berdasarkan kemunculan *term*/kata pada sebuah dokumen maka *term* bernilai 1 sehingga dengan kemunculan *term* yang semakin banyak tidak akan berpengaruh
- *Augmented TFidf* = $0.5 + 0.5 \left(\frac{tf}{\max(tf)} \right)$

$$(2)$$

dimana *tf* = jumlah *term* dalam sebuah dokumen

$\max(tf)$ = jumlah kemunculan terbanyak *term* pada dokumen yang sama

3.4.3 Inverse Document Frequency (IDF)

IDF merupakan jumlah dokumen yang berisikan *term* yang di cari dalam dokumen dataset. *IDF* sendiri biasa disebut *Global Weigh* dimana fungsinya adalah mendefinisikan kontribusi dari *term* kedalam dokumen :

$$Idf = \text{Log} \left(\frac{N}{df} \right) \quad (3)$$

dimana *N*: jumlah dokumen yang ada dalam kumpulan dokumen

df : jumlah dokumen yang terdapat *term*.

Hasil dari *TF-IDF* adalah dengan perkalian antara *TD* dan *IDF*

3.4.4 TF - IDF

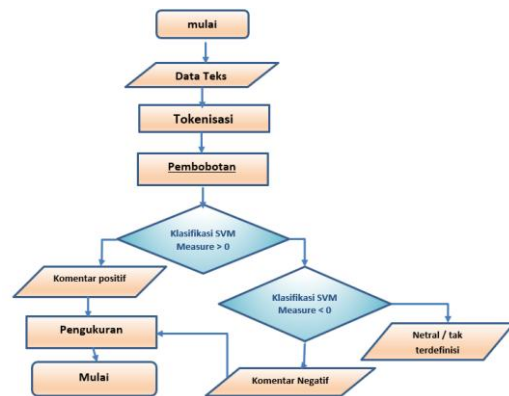
Adalah Algoritma pembobotan yang di integrasikan dari *term frequency* (*tf*) dan *inverse document frequency* (*idf*) yang di jabarkan dengan rumus :

$$tfidf(t,d,D) = tf(t,d) \times idf(t,D) \quad (4)$$

Dimana : *t* = *term*, *d* = *dokumen*
D = *koleksi dokumen*

Algoritma ini fungsinya untuk mencari representasi nilai dari kumpulan *datatraining* yang hasilnya merupakan bentuk vektor antara dokumen dan kata, di cluster berdasarkan kesamaan antara dokumen dan *term*/kata.

Proses Klasifikasi Support Vector Machine (SVM)



Gambar 6. Diagram alur proses data

Proses data berupa kata-kata yang sudah melalui *case folding*, *cleansing* dan *filtering* akan lanjutkan kedalam proses *tokenizing* akan dihitung pembobotannya.

Klasifikasi *machine learning* dengan SVM dimulai dengan mengubah teks menjadi data vektor, kemudian data vektor memiliki dua dimensi yaitu (*word id*) dan bobot di bagi untuk menentukan garis terbaik yang memisahkan diantaranya dua kelas dengan *margin* terbesar, klasifikasi pembobotan kata dihitung similaritasnya selanjutnya menentukan sentimen dokumen yang diuji sehingga menghasilkan dokumen *dataset*

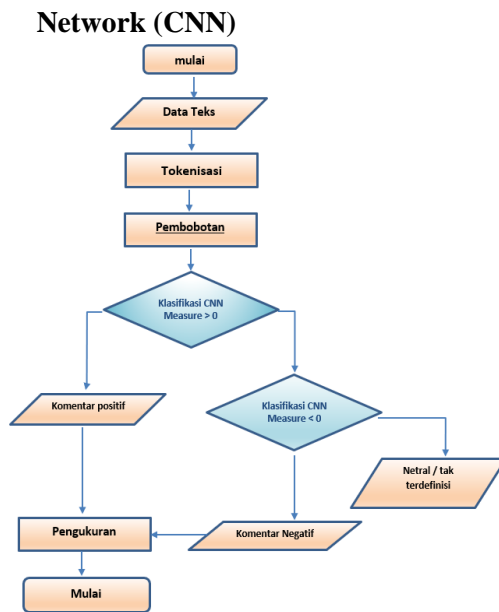
Support Vector Machine (SVM) yang menggunakan 2 bagian data percobaan yaitu data pelatihan dan data pengujian yang dibagi dalam empat tahapan *dataset* dengan kombinasi antara data pelatihan dan data pengujian sebagaimana tabel 2 dibawah ini

Tabel 2 Data Set

Perbandingan Data	
Data Pelatihan	Data Pengujian
80%	20%
70%	30%
60%	40%
50%	50%
40%	60%
30%	70%
20%	80%

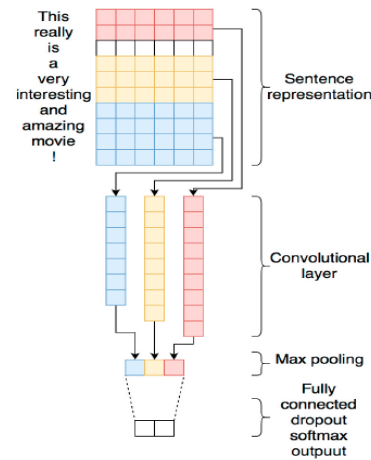
Secara umum *dataset* tidak dipisahkan secara linier, karena itu *dataset* harus dipetakan ke spasi fitur (*feature space*) yang lebih tinggi melalui mapping sehingga didapatkan hyperplane pemisah yang lebih optimal dengan menggunakan fungsi *kernel*

3.5 Proses Klasifikasi Convolutional Neural



Gambar 7. Diagram Alir Klasifikasi dengan Algoritma CNN

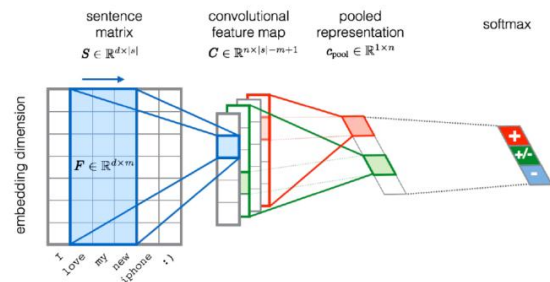
Pada gambar 7 Klasifikasi prosedi CNN setelah melewati proses data akusisi sampai dengan pembobotan. Cara kerja CNN memiliki kesamaan pada MLP, namun dalam CNN setiap neuron dipresentasikan dalam bentuk dua dimensi, tidak seperti MLP yang setiap neuron hanya berukuran satu dimensi



Gambar 8. Diagram Alir Klasifikasi dengan Algoritma CNN

CNN adalah salah satu model koneksi yang paling umum digunakan untuk klasifikasi. Model Connectionism berfokus pada belajar dari rangsangan lingkungan dan menyimpan informasi ini dalam bentuk koneksi antar neuron. Bobot dalam jaringan saraf disesuaikan menurut data pelatihan dengan beberapa algoritma pembelajaran.

Semakin besar perbedaan dalam data pelatihan, semakin sulit bagi algoritma pembelajaran untuk menyesuaikan data pelatihan, dan hasil klasifikasi yang buruk.



Gambar 9. Arsitektur model deep learning untuk klasifikasi sentimen

3.5.1 Matriks Kalimat (Sentence matrix)

Inputan data berupa tweet yang di gunakan sebagai urutan kata $w_1, \dots, w_{|s|}$, dimana setiap kata diambil dari kosakata V , Kata-kata diwakili oleh vektor distribusi

$w \in \mathbb{R}^{1 \times d}$ mencari kata yang terdapat dalam matriks $W \in \mathbb{R}^{d \times |V|}$.

Matriks ini dibentuk dengan hanya menggabungkan embeddings dari semua kata dalam V , Untuk kenyamanan dan kemudahan pengoperasian lookup di W kata-kata dipetakan ke dalam indeks $1, \dots, |V|$

Untuk setiap input tweet S , membangun matriks kalimat $S \in \mathbb{R}^{d \times |S|}$ dimana setiap kolom i mewakili embedding kata w_i pada yang sesuai posisi i dalam kalimat. Untuk belajar menangkap dan menyusun fitur kata-kata individual dalam sebuah kalimat yang diberikan dari inialisasi kata tingkat rendah ke dalam konsep semantik tingkat yang lebih tinggi, jaringan syaraf tiruan menerapkan serangkaian transformasi ke input matriks kalimat S menggunakan konvolusi, operasi non linier dan penyatuan, yang kami jelaskan selanjutnya.

3.5.2 Peta fitur konvolusi (*Convolutional Feature Maps*)

Tujuan lapisan konvolusi adalah mengekstrak pola, yaitu diskriminatifurutan kata yang ditemukan di dalam tweet masukan yang umum terjadi selama contoh pelatihan. Secara lebih formal, operasi konvolusi, $*$, di antara matriks masukan $S \in \mathbb{R}^{d \times |S|}$ dan sebuah filter $F \in \mathbb{R}^{d \times m}$ dimana lebar m menghasilkan sebuah vektor $c \in \mathbb{R}^{|S|+m-1}$ dimana masing-masing komponen dihitung sebagai berikut :

$$c_i = (S * F)_i = \sum_{k,j} (S_{[i-m+1:i]} \otimes F)_{kj}, \quad (5)$$

dimana \otimes penggandaan unsur-bijak dan element-wise multiplication dan $S_{[i-m+1:i]}$ adalah sebuah irisan matrik berukuran sepanjang kolom. Perhatikan bahwa filter konvolusi memiliki dimensi yang sama d sebagai matriks kalimat masukan, sebagaimana diperlihatkan dalam Gambar 9 slide sepanjang dimensi kolom

menghasilkan keluaran sebuah vektor $c \in \mathbb{R}^{1 \times (|S|-m+1)}$. Masing masing komponen c_i adalah hasil perhitungan sebuah elemen produk antara kolom slide S dan sebuah filter matrik F yang kemudian di jumlahkan menjadi nilai tunggal. Setelah tahapan menjelaskan cara untuk menghitung konvolusi antara matriks kalimat input dan satu filter. Untuk membentuk representasi data yang lebih kaya, model pembelajaran yang dalam menerapkan seperangkat filter yang bekerja secara paralel menghasilkan beberapa peta fitur (juga ditunjukkan pada Gambar 9).

Satu set filter membentuk bank filter $c \in \mathbb{R}^{n \times (|S|-m+1)}$ yang secara berurutan dipecahkan dengan matriks kalimat S dan menghasilkan sebuah fitur peta matrik

Dalam prakteknya, juga perlu menambahkan vektor bias $b \in \mathbb{R}^n$ untuk hasil konvolusi bias tunggal b_i , nilai untuk setiap peta fitur c_i . Ini memungkinkan jaringan untuk mempelajari ambang yang sesuai

3.5.3 Unit aktivasi (*Activation units*)

Untuk memungkinkan pembelajaran batasan keputusan non-linear, masing-masing lapisan konvolusi biasanya diikuti oleh fungsi aktivasi non linier $\alpha(\cdot)$ di terapkan pada element-wise. Di antara yang paling umum pilihan fungsi aktivasi adalah sigmoid (or logistic), hyperbolic tangent tanh, dan a rectified linear (ReLU) fungsi didefinisikan dengan sederhana $\max(0, x)$ untuk memastikan bahwa peta fitur selalu positif. Penggunaan ReLU dalam model karena, ini mempercepat pelatihan dan terkadang menghasilkan hasil yang lebih akurat

3.5.4 Pooling (*Pooling*)

Pooling yang biasa digunakan adalah *Max Pooling* dan *Average Pooling* dimana penyatuan maksimum adalah proses

diskretisasi berbasis sampel. Tujuannya adalah untuk menurunkan sampel representasi masukan (gambar, matriks keluaran lapisan tersembunyi, dll.), ini dilakukan untuk membantu penyesuaian secara berlebihan dengan menyediakan bentuk representasi yang disarikan. Selain itu, ini mengurangi biaya komputasi dengan mengurangi jumlah parameter untuk dipelajari dan memberikan inventarisasi translasi dasar ke representasi internal. Penyatuan maksimum dilakukan dengan menerapkan filter maks ke (biasanya) subregulasi non-tumpang tindih dari representasi awal.

3.5.5 Softmax

Output dari lapisan *convolutional* dan *pooling* kedua dari belakang x dilewatkan ke lapisan *softmax* yang terhubung penuh. Ini menghitung distribusi probabilitas di atas label

$$P(y = j|x, s, b) = \text{softmax}_j(\mathbf{x}^T \mathbf{w} + b) = \frac{e^{\mathbf{x}^T \mathbf{w}_j + b_j}}{\sum_{k=1}^K e^{\mathbf{x}^T \mathbf{w}_k + b_k}} \quad (6)$$

Dari persamaan diatas dimana w_k dan b_k bobot *vector* k dan nilai *bias* k pada kelas

3.5.6 Phrase-level sentiment analysis

Proses analisis sentimen tingkat *frase*, mengurut masukan tambahan lokasi *frase target* dalam sebuah *tweet*. Elemen dikodekan dengan dua jenis kata: *tokenfrase* diprediksi dengan kode 1s dan yang lainnya dengan 0s. Setiap jenis kata dikaitkan dengan *embedding*. Saat klasifikasi sentimen tingkat *frase*, membentuk matriks kalimat S yaitu setiap *token* dalam *tweet*, harus ada kata yang sesuai dengan *embedding* pada kata *matrix* W, dan *embedding* untuk salah satu dari dua tipe kata. Untuk melakukan perhitungan pada CNN untuk melihat serta membandingkan *performance* pada penggunaan data latih dan data uji serta komposisi yang sama. Dalam penelitian ini CNN tidak dibahas secara mendetail karena dalam aplikasi alat

bantu *python* yang digunakan yang telah terpadu antara SVM dan CNN.

IV. Hasil Pengujian Algoritma

Dari data yang di hasilkan dengan alat bantu github untuk mengambil data tweet terkait pilgub jabar, proses ekstrasi data tweet untuk mengklasifikasi kata kata yang bernilai negatif, netral dan positif, di mulai pengumpulan data, Preprocessing Data, Klasifikasi dengan SVM dan CNN, Perbandingan Kedua metode yang dipakai SVM dan CNN

Tabel 3 Hasil Pengujian

Sebelum Stemming		Nilai Prosentase Pengukuran							
		Akurasi		Precisi Data		Recall Data		F_Measure	
		SVM	CNN	SVM	CNN	SVM	CNN	SVM	CNN
Rata-Rata 7 kali Percobaan		67.11	67.17	67.42	67.51	67.19	67.20	67.08	67.26

Setelah Stemming		Nilai Prosentase Pengukuran							
		Akurasi		Precisi Data		Recall Data		F_Measure	
		SVM	CNN	SVM	CNN	SVM	CNN	SVM	CNN
Rata-Rata 7 kali Percobaan		67.17	52.99	67.51	52.54	67.20	51.70	67.26	50.22

Tabel 4 Hasil Pengujian

Sebelum Stemming		Nilai Prosentase Pengukuran							
		Akurasi		Precisi Data		Recall Data		F_Measure	
		SVM	CNN	SVM	CNN	SVM	CNN	SVM	CNN
Komposisi Data		73	69	74	70	73	70	73	70
Latih 80% - Uji 20%		72	71	72	70	72	71	72	71
Latih 70% - Uji 30%		66	67	66	67	66	67	66	67
Latih 60% - Uji 40%		68	68	68	69	68	68	68	68
Latih 50% - Uji 50%		66	67	66	67	66	67	66	67
Latih 40% - Uji 60%		63	66	64	66	63	66	63	66
Latih 30% - Uji 70%		62	63	62	63	62	62	62	63
Latih 20% - Uji 80%									

Setelah Stemming		Nilai Prosentase Pengukuran							
		Akurasi		Precisi Data		Recall Data		F_Measure	
		SVM	CNN	SVM	CNN	SVM	CNN	SVM	CNN
Komposisi Data		69	46	70	46	70	44	70	42
Latih 80% - Uji 20%		71	45	70	42	71	43	71	40
Latih 70% - Uji 30%		67	58	67	59	67	57	67	57
Latih 60% - Uji 40%		68	51	69	49	68	49	68	47
Latih 50% - Uji 50%		67	59	67	60	67	59	67	58
Latih 40% - Uji 60%		66	67	66	67	66	66	66	66
Latih 30% - Uji 70%		63	46	63	44	62	44	63	42
Latih 20% - Uji 80%									

V. Kesimpulan

Berdasarkan hasil dari data *twitter* yang diambil dan dilakukan percobaan terhadap data *texttwitter* yang diolah menjadi sentimen analisis dengan menggunakan dua Algoritma SVM (*Support vector Machine*) dan CNN (*Convolutional Neural Network*) dapat diambil kesimpulan sebagai berikut :

1. Percobaan Pengujian sebelum *stemming* dengan Algoritma SVM dilakukan tujuh kali percobaan tingkat akurasi rata-rata di sekitar 67%, Dan Algoritma CNN sebelum *stemming* juga dengan tujuh kali percobaan dengan akurasi rata-rata disekitar 67 %
 2. Percobaan Pengujian setelah *stemming* dengan SVM dilakukan tujuh kali percobaan tingkat akurasi rata-rata di sekitar 67%, sementara CNN sebelum *stemming* juga dengan tujuh kali percobaan dengan akurasi rata-rata disekitar 52% lebih rendah dari SVM
 3. Dengan Tujuh kali pengujian menggunakan SVM sebelum *stemming* akurasi tertinggi dengan data Training sebanyak 80% dan data Uji sebanyak 20% dengan nilai akurasi 73%, tetapi saat setelah *stemming* menjadi 69%, **lebih rendah akurasinya.** begitupun halnya Percobaan CNN akurasi tertinggi didapatkan sebelum *stemming* dengan komposisi Data Latih 80% dan Data Uji sebanyak 20% dengan Nilai akurasi 69% kemudian nilai nya menjadi lebih rendah setelah proses *stemming* menjadi 46%.
 4. Proses *stemming* sangat di berpengaruh terhadap hasil karena klasifikasi setelah *stemming* mempunyai pengaruh terhadap akurasi.
- DAFTAR PUSTAKA**
- [1] Ghulam Asrofi Buntoro : "Analisis Sentimen Calon Gubernur DKI Jakarta 2017 di *Twitter*" dari Program Studi Teknik Informatika, Fakultas Teknik, Universitas Muhammadiyah Ponorogo
 - [2] Ronen Feldman, James Sanger " The Text Mining Handbook". Cambridge University Press 2007
 - [3] K.L.Sumathy (Research and Development Center Bharathiyar University Coimbatore) & M.Chidambaram, Ph.d (Assistant Professor/CS Rajah Serfoji Govt. College Thanjavur) " Text Mining: Concepts, Applications, Tools and Issues – An Overview" International Journal of Computer Applications (0975 – 8887) Volume 80 – No.4, October 2013
 - [4] Christianini, Nello. Support Vector and Kernel Machines. ICML tutorial, 2001.
 - [5] Budi Santosa. Tutorial Support Vector Machines, Profesor di Teknik Industri, ITS Kampus ITS, Sukolilo Surabaya 2015
 - [6] I Wayan Suartika E.P, Arya Yudhi Wijaya dan Soelaiman : " Klasifikasi Citra menggunakan CNN (Convolutional Neural Network) pada Caltech 101". 2016 . Surabaya : Jurnal Teknik ITS Vol 5 No 1, 2016 ISSN : 2337-3539 (23-9271 Print).
 - [7] Anto Satriyo Nugroho, Arief Budi Witarto, Dwi Handoko : Support Vector Machine, Teori dan Aplikasinya dalam Bioinformatika, Kuliah Umum IlmuKomputer.Com. Copyright © 2003 IlmuKomputer.Com ; <http://www.asnugroho.net/papers/ikcsvm.pdf>
 - [8] J. Moody, C. Darken, Neural Computation 1 (1989) 281.
 - [9] H. Ogawa, Proceeding of International Conference on Intelligent Information Processing System , Beijing, RRC, 1992.
 - [10] Agnes Rossi Trisna Lestari1, Rizal Setya Perdana2, M. Ali Fauzi : "Analisis Sentimen Tentang Opini Pilkada DKI 2017 Pada Dokumen Twitter Berbahasa Indonesia Menggunakan Naïve Bayes dan Pembobotan Emoji" Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
 - [11] F. Luo, C. Li and Z Cao : " Affective-feature-based sentiment analysis using SVM classifier". 2016. IEEE 20th International Conference Computer Support Cooperation Work. Des. pp. 276281-2016.

- [12] An Introduction to conventional Neural Network, Visine Imaging Science and Technology Lab, Stanford University [Online]. Available at http://white.stanford.edu/teach/index.php/An_introduction_to_convolution_neural_networks.
- [13] Deep learning Tutorial, , Lisa Lab . [Online] Available <http://deeplearning.net/tutorial/contents.html>
- [14] Noviah Dwi Putranti dan Edi Winarko : "Analisis Sentimen Twitter untuk Teks Berbahasa Indonesia dengan Maximum Entropy dan Support Vector Machine", Jogjakarta 2014 : Jurusan Ilmu Komputer, FMIPA UGM, Yogyakarta Jurusan Ilmu Komputer dan Elektronika, FMIPA UGM, Yogyakarta
- [15] Abien Fred M. Agarap : "An Architecture Combining Convolutional Neural Network (CNN) and Support Vector Machine (SVM) for Image Classification". Department of Computer Science Adamson University Manila, Philippines
- [16] Siti Nur Asiyah dan Kartika Fithriasari : "Klasifikasi Berita Online Menggunakan Metode Support Vector Machine dan K-Nearest Neighbor"