Jurnal Elektronik Ilmu Komputer Udayana
Volume 14, No 2. November 2025

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Analisis dan Visualisasi Sentimen Analisis Data Twitter Menggunakan Support Vector Machine dan Social Network Analysis

Getzbie Alfredo Tpoy[a1], Ida Ayu Gde Suwiprabayanti Putra[a2], Anak Agung Istri Ngurah Eka Karyawati[a3], I Putu Gede Hendra Suputra[a4]

[a]Informatics, Faculty of Match and Science, University of Udayana
South Kuta, Badung, Bali, Indonesia
[1]getzbiealfredo123@gmail.com
[2]iagsuwiprabayantiputra@unud.ac.id
[3]eka.karyawati@unud.ac.id
[4]hendra.suputra@unud.ac.id

Abstract

The dissemination of information in today's era is extremely rapid due to the development of various social media platforms. However, the information circulating is sometimes inaccurate or invalid. One of the platforms commonly used to spread information is Twitter. Among the vast amount of circulating information, some content is deliberately spread—either positive or negative. This sparked the idea to develop a sentiment analysis program that can be visualized using Support Vector Machine (SVM) and Social Network Analysis (SNA). The model was built using a Twitter dataset and achieved an accuracy of 86.8%, a precision of 86.2%, a recall of 87%, and an F1-score of 86.6%. The model effectively classifies sentiment in Twitter data into appropriate categories. The visualization is presented by displaying nodes and edges interconnected to form a graph. Each node varies in size and color to represent the relationships found in the data.

Keywords: *Support Vector Machine, Sentiment, Social Network Analysis, Visualization, Graph*

1. Pendahuluan

Penyebaran informasi semakin mudah dan cepat dengan menggunakan media sosial. Isu mudah diangkat dan menjadi trending topik, kemudian memicu berbagai reaksi. Ada yang menganggapnya secara positif, setuju dengan isu atau respon. Ada yang tidak terima, memberikan kritikan, bahkan menebarkan ujaran kebencian. Percakapan di media sosial didapat dari komunikator yang menggunakan sosial media dan mempengaruhi khalayak dalam jumlah besar melahirkan komunikasi massa. Topik atau percakapan yang terjadi di media sosial dapat disebut sebagai sebuah sentimen, yang kemudian dapat dianalisis.

Analisis sentimen merupakan proses mengekstrak dan menganalisa data tekstual untuk memperoleh informasi sentimen dalam suatu kalimat [1] Analisis sentimen menghasilkan isi opini pada sebuah isu atau permasalahan yang memiliki kecenderungan negatif atau positif. Analisis sentimen dilakukan dengan mengolah bahasa alami (*natural language processing)* dan klasifikasi sentimen dengan menggunakan algoritma pembelajaran mesin, dan evaluasi.

Salah satu algoritma yang dapat digunakan untuk klasifikasi pada analisis sentimen adalah *Support Vector Machine*(SVM). SVM akan mencari hyperplane optimal yang memisahkan kelompok data dengan margin maksimum. Beberapa penelitian sebelumnya yang membahas tentang analisis sentimen dengan menggunakan SVM diantaranya . Analisis Sentimen Terhadap Layanan Indihome Berdasarkan Twitter Dengan Metode Klasifikasi *Support Vector Machine* (SVM)[2] menghasilkan akurasi sebesar 87% dengan ketepatan antara hasil prediksi dengan data sebenarnya (*precision*) sebesar 86%, tingkat keberhasilan sistem dalam memprediksi sebuah data (*recall*) sebesar 95%,

tingkat kesalahan semua data yang diprediksi (*error rate*) sebesar 13%, sedangkan untuk nilai perbandingan rata-rata *precision* dan *recall* (*f1- score*) adalah sebesar 90%.

Analisis sentimen yang dilakukan pada penelitian – penelitian sebelumnya memberikan output berupa hasil sentimen seperti jumlah dan akurasi opini negatif atau positif. Masih terdapat informasi yang bisa dianalisis lebih dalam seperti sumber isu dan opini disekitar sumber isu(negatif atau positif). Informasi yang ada bisa dianalisis lebih dalam dengan memvisualisasikan menggunakan network (graph). Visualisasi didapatkan dengan mencari keterhubungan diantara sumber informasi. Penelitian ini akan membahas implementasi analisis sentiment beserta dengan visualisasinya.

2. Metode

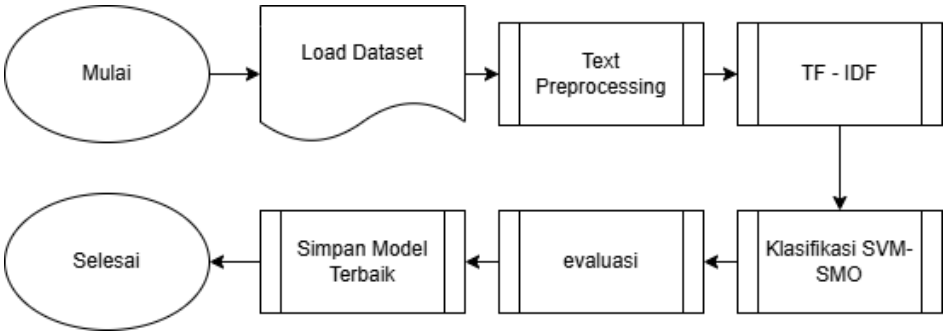
Tahapan dalam penelitian ini dibagi menjadi dua bagian, yaitu pembuatan *machine learning* dengan menggunkan *support vector machine* dan dilanjutkan viasualisasi menggunakan *social network analysis.*

2.1. Pengumpulan data

Data yang digunakan adalah data twitter yang didapat dari Kaggle dengan judul dataset "*Indonesia Presidential Candidate's Dataset, 2024*" berisikan twitter dengan topik pemilu. Data tersebut dikumpulkan sebelum periode pemilu, yang berisikan opini mengenai calon presiden Indonesia. Total data berjumlah 30.000 record, yang dikumpulkan dari Oktober 2022 – hingga April 2023. Data lainnya adalah data twitter asli yang diperoleh melalui API twitter, data ini dikumpulkan sebagai sampel oleh situs drone emprit, sebanyak 3000 *record* data twitter dengat format JSON. Karena keterbatasan API twitter yang sudah tidak dapat diakses bebas, hanya data ini saja yang dapat digunakan untuk visualisasi SNA.

2.2. Rancangan Model SVM

langkah – langkah secara umum dalam membuat machine learning. Alur secara umum ditunjukan pada gambar 1



Gambar 1. Alur pembuatan Machine Learning

a. Load Dataset
   Langkah awal yang dilakukan adalah meload dataset yang situs Kaggle. Tidak semua bagian dari dataset digunakan, yang diambil adalah kolom yang diperlukan yaitu kolom Text dan Label. Kemudian akan dilakukan penyesuaian pada label, karena SVM hanya mengenali label 1 dan -1, maka label yang berisikan char harus diubah menjadi numerik.
b. Text Preprocessing
   Preprocessing merupakan tahapan yang mencakup semua proses, metode, dan rutinitas yang dilakukan untuk menyiapkan data yang digunakan dalam text mining untuk menemukan suatu pengetahuan. Preprocessing mengkonversi informasi dalam setiap sumber data asli ke dalam format kanonik sebelum menerapkan berbagai metode ekstraksi fitur [3].
   1. Tokenization

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Tokenization dilakukan untuk memecah teks menjadi daftar potongan yang di inginkan seperti token kata, frasa, simbol untuk mengahasilkan data teks yang dapat dikerjakan dengan lebih efektif. *Tokenization* perlu dilakukan karena hasilnya akan dijadikan input untuk tahap selanjutnya. Token yang didapatkan dari *Tokenization* dapat dipisahkan satu dengan lainnya menggunakan *whitespace, punctuation,* atau *line break*.

2. Removing punctuation
   Tanda baca dan emotikon tidak diberlakukan sebagai token untuk klasifikasi sentimen. Menghapus emotikon dan tanda baca dapat meningkatkan akurasi klasifikasi karena dianggap sebagai dimensi dalam set fitur untuk setiap kata.
3. Stowpword elimination
   Stopword merupakan kata-kata paling umum dalam Bahasa contohnya seperti “di”, “adalah”, “sebuah” dan daianggap tidak perlu dala penambangan data tekstual. Kata – kata inti dapat berupa kata ganti , kata depan, kata sambung, dan kata kerja bantu.
4. Stemming
   Stemming dilakukan untuk mengubah kata – kata yang ada penjadi kata dasar dengan menghilangkan afiks(awalan kata) dan sufiks(akhiran kata) sesuai dengan aturan tata Bahasa yang berlaku.
5. Lowercasing
   Lowercasing dilakukan untuk mengkonversi semua huruf menjadi huruf kecil untuk mencegah sensivitas terhadap huruf besar atau kecil. Sehingga kinerja klasifikasi dapat ditingkatkan

c. TF-IDF

Term Frequency-Inverse Document Frequency(TF-IDF) merupakan pengukuran formal tentang seberapa terkonsentrasi kata ke dalam dokumen yang relatif sedikit [4]. Misalkan terdapat sebuah dokumen *N*, tentukan f_{ij} sebagai frekuensi (jumlah kemunculan) istilah kata *i* dalam dokumen *j* ditunjukan dalam persamaan (1)

$$TF_{ij} = \frac{f_{ij}}{max_k f_{kj}} \quad (1)$$

Pada persamaan diatas frekuensi suku *i* dalam dokumen *j* adalah f_{ij}, dinormalisasi dengan membaginya dengan jumlah maksimum kemunculan istilah apapun dalam dokumen yang sama. Sehingga dapat diperoleh *TF*.

IDF dapat didefinisikan, misalkan istilah *i* muncul dalam n_i pada dokumen *N* maka *IDF* dapat dituliskan dalam persamaan (2).

$$IDF_i = log_2(\frac{N}{n_i}) \quad (2)$$

Untuk menghitung *TF-IDF*, skor dalam *term i* dalam dokumen *j* dapat didefinisikan sebagai (3).

$$TF - IDF = TF_{ij} \times IDF \quad (3)$$

Dengan *TF* tertinggi dan skor *IDF* menjadi istilah yang paling mencirikan dokumen.

d. SVM-SMO

Tujuan utama dari *Support Vector Machine* adalah memilih *hyperplane* (4)

$$w.x + b = 0 \quad (4)$$

Yang memaksimalkan jarak γ antara *hyperplane* dari titik manapun dari set data[4]. Sebuah *hyperplane* akan memisahkan data berbeda di seberang *hyperplane* tersebut. Secara intuituf, kelas data yang berada jauh dari *hyperplane* (pada ruang yang benar) memiliki kelas yang lebih jelas dibandikan dengan kelas data yang berada dekat dengan *hyperplane*. Titik yang dekat dengan *hyperplane* ini disebut sebagai *support vector*.

Dapat dirumuskan sebagai *quadratic* problem untuk mendapat titik minimal ditunjukan dalam persamaan (5) dengan memperhatikan *constrait* atau kendala persamaannya (6).

$$\min_{\vec{w}} \tau(w) = \frac{1}{2}\|\vec{w}\| \quad (5)$$

$$yi(w \cdot x_i + b) \geq -1, \forall i \quad (6)$$

Persamaan (5) dan (6) dapat direduksi dengan menggunakan fungsi *lagrange* yang terdiri dari jumlah fungsi objektif dan m kendala dikalikan dengan pengganda *langrange*, persamaan ditunjukan (7) dapat disebut sebagai sequential minimal optimization.

$$L(w,b) = \frac{1}{2}(w.w) - \sum_{i=1}^{m} a_i\,(y_i(w.x_i + b) - 1 \quad (7)$$

a_i merupakan *langrange multipliers* dan nilai $a_i \geq 0$ [5]

e. Evaluasi
Metriks evaluasi yang digunakan diantaranya Akurasi, *Precision*, *Recall*, dan *F-1 Score*. Metriks ini dapat digunakan untuk mengevaluasi kualitas pengklasifikasi. Sehingga dapat diketahui seberapa baik pengklasifikasi dapat menegenali tuple dari berbagai kelas [6].

$$AKURASI = \frac{TP+TN}{TP+FP+TN+FN} \times 100\% \quad (8)$$

$$PRECISION = \frac{TP}{TP+FP} \quad (9)$$

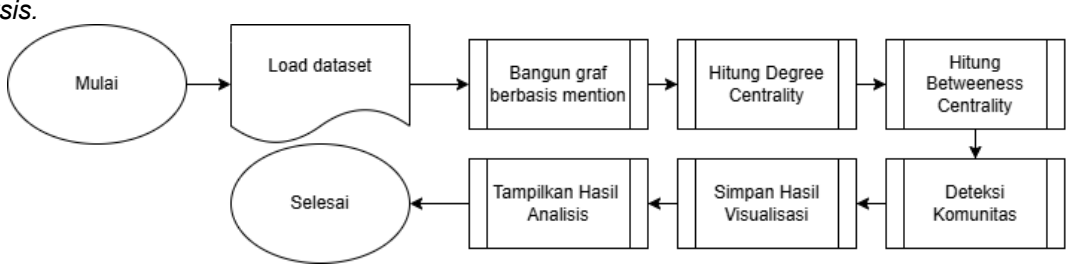
$$RECALL = \frac{TP}{FP+FN} \quad (10)$$

$$F-1\,SCORE = \frac{2 \times PRECISION \times RECALL}{PRECISION + RECALL} \quad (11)$$

f. Simpan Model Terbaik
Setelah melakukan beberapa kali proses training dan pengujian, dapat dipilih model dengan kinerja terbaik. Model akan disimpan dalam format pickle. Beberapa parameter yang disimpan diantaranya model, jumlah fitur, vector pembobotan, dan mean normalisasi. Beberapa parameter ini harus disimpan, sehingga ketika model diload tidak perlu melakukan perhitungan ulang.

2.3. Rancangan SNA

Pada gambar 2 akan dijelaskan langkah – langkah secara umum dalam membuat *Social Network Analysis.*



Gambar 2. Alur Pembuatan SNA

a. Load Dataset
Dataset yang digunakan dalam SNA adalah dataset asli twitter dengan format JSON, sehingga masih menyimpan informasi seperti username, twit *reply*, dan re-twit.

b. Bangun Graf
Struktur graf diinisialisasi dengan menggunakan sebuah dict yang menampung hubungan antar pengguna. kemudian buat sebuah list untuk menyimpan teks twit, author. Untuk mencari twit berupa reply, cari semua data yang memuat simbol mention “@” dengan menggunakan regex.

c. Degree Centrality

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Untuk menghitung *degree centrality*, hitung jumlah *edge* keluar untuk setiap *node* (*degree out*). Untuk setiap *node* yang menjadi target, tambahkan nilai 1(*degree in*).

d. Betweeness Centrality
Metode yang digunakan dalam menghitung betweness adalah metode *Breath first search* untuk menelusuri semua *node* yang ada, dan mencari *node* perantara yang memiliki rute terpendek.

e. Deteksi Komunitas
Deteksi komunitas menggunakan label yang memetakan setiap *node* ke ID label, kemudian acak urutan *node*, dan untuk setiap *node* Kumpulkan tetangga (baik dari *edge* keluar maupun *edge* masuk: *node* yang memiliki *node* sebagai target). Ambil label tetangga yang sudah ada di labels.

f. Visualisasi
Hasil graph yang sudah dibuat, akan disimpan dalam bentuk svg. Menyimpan dengan format SVG merupakan alternatif yang dapat dignakan untuk mengantikan fitu library seperti matplotlib, dengan cara menggambar *node* dan *edge* yang sudah diperoleh ke kanvas SVG.

g. Hasil Analisis
Beberapa hasil analisis yang ditampilkan adalah *degree centrality, betweenness centrality, dan Community summary.*

3. Hasil dan Pembahasan

3.1. Hasil Model SVM

a. Text preprocessing dan TF-IDF
Dataset yang digunakan merupakan dataset twitter yang ditunjukan pada gambar 3 berisikan opini yang dilakukan pengguna twitter dengan topik pemilihan presiden.

	Tweet	Tweet_tokens
0	info anies presiden	[info, anies, presiden]
1	politisi partai gerindra sandiaga uno menjawab soal diduetkan kembali dengan mantan gubernur dki jakarta anies baswedan dalam pemilihan presiden mendatang	[politisi, partai, gerindra, sandiaga, uno, menjawab, soal, diduetkan, kembali, dengan, mantan, gubernur, dki, jakarta, anies, baswedan, dalam, pemilihan, presiden, mendatang]
2	lanjut pak anies kita kawal sampai jadi presiden	[lanjut, pak, anies, kita, kawal, sampai, jadi, presiden]
3	semoga allah swt menyelamatkan bangsa dan negara republik indonesia dari para penghianat amanat rakyat yaa allah yang maha pengasih lagi maha penyayang angkatlah derajat bang anies rasyid baswedan menjadi presiden republik indonesia tahun dan seterusnya aamiin yra	[semoga, allah, swt, menyelamatkan, bangsa, dan, negara, republik, indonesia, dari, para, penghianat, amanat, rakyat, yaa, allah, yang, maha, pengasih, lagi, maha, penyayang, angkatlah, derajat, bang, anies, rasyid, baswedan, menjadi, presiden, republik, indonesia, tahun, dan, seterusnya, aamiin, yra]

Gambar 3. Dataset *Preprocessing*

Setelah dilakukan penyeimbangan, total dataset yang ada sebanyak 3453 data positif dan 3453 data negatif. Kemudian data akn derepresentasikan ke dalam bentuk angka dengan menghitung kemunculuan setiap kata dalam dokumen tertentu (*Term Frequention*), melakukan perhitungan jumlah dokumen dengan kata tertentu (*Document Frequention*), melakukan perhitungan *Inverse Document Frequention*, dan melakukan perhitungan TF-IDF.

b. Klasifikasi dan evaluasi model

Pada pembuatan model menggunakan dua fungsi *kernel* yaitu *kernel linear* dan *kernel Radial Basis Function*. Data dapat ditransformasikan ke dalam fitur yang lebih tinggi dengan menggunakan *kernel*. Parameter pada *kernel linear* adalah nilai C yaitu nilai yang mengontrol penalti untuk kesalahan pada klasifikasi. Parameter pada *kernel RBF* adalah nilai *gamma* yang mengontrol pengaruh suatu titik terhadap sekitarnya.

Pada proses training, model divalidasi dengan menggunakan metode *k – fold – cross validation* dengan menggunakan nilai k = 5 iterasi sebanyak 10000. Pengujian pertama dilakukan pada *kernel linear* dengan menggunakan nilai $C = 0.001, 0.05, 0.1, 1.$

Tabel 1. Pengujian dengan nilai $C = 0.001$

Fold	Akurasi (%)	Precision (%)	Recall (%)	F1-Score (%)
1	81	87	72	79
2	82	78	88	83

3	82	86	75	80
4	82	89	75	81
5	83	90	74	81
Rata-rata	82	86	76.8	80.8

Hasilnya pada tabel 1 model dapat mencapai akurasi 82% dengan nilai parameter kecil nilai. Namun hasil *Recall* masih cukup kurang jika dibandingkan dengan hasil *Precision* menunjukan model masih kurang baik terhadap data positif

Pengujian selanjutnya menggunakan nilai C yang diperbesar menjadi 0.05 hasil yang ditunjukan pada tabel 2 Hasilnya ditunjukan pada tabel model dapat mencapai akurasi 86.8% dengan nilai parameter C yang ditingkatkan

Tabel 2. Pengujian menggunakan nilai $C = 0.05$

Fold	Akurasi (%)	Precision (%)	Recall (%)	F1-Score (%)
1	85	84	87	85
2	87	88	86	87
3	87	87	85	86
4	87	87	86	87
5	88	85	91	88
Rata-rata	86.8	86.2	87	86.6

Hasil *Precision, Recall*, dan *F – 1 Score* pada tabel 2 menunjukan hasil yang lebih stabil dibandingkan parameter sebelumnya dengan hasil *Precision* sebesar 86%, hasil *Recall* sebesar 87%, dan *F-1 Score* 86.6%.

Tabel 3. Pengujian dengan nilai $C = 0.1$

Fold	Akurasi (%)	Precision (%)	Recall (%)	F1-Score (%)
1	84	83	85	84
2	86	86	86	86
3	87	86	88	87
4	86	86	86	86
5	87	85	90	88
Rata-rata	86	85.2	87	86.2

Hasilnya ditunjukan pada tabel 3 model dapat mencapai rata - rata akurasi 86% dengan nilai parameter C yang ditingkatkan . Hasil *Precision, Recall*, dan *F1 - Score* juga menunjukan hasil sudah cukup baik, namun tidak lebih baik daripada parameter $C = 0.05$

Selanjutnya dilakukan percobaan terhadap *kernel RBF* dengan menggunakan parameter *gamma*. Percobaa dilakukan untuk melihat bagimana data yang tidak dapat dipisahkan di ruang *linear*.

Tabel 4. Pengujian menggunakan kernel RBF

Fold	Akurasi (%)	Precision (%)	Recall (%)	F1-Score (%)
1	82	78	88	83

p-ISSN: 2301-5373
e-ISSN: 2654-5101

2	84	81	89	85
3	82	79	86	82
4	84	80	89	84
5	84	81	90	85
Rata-rata	83.2	79.8	88.4	83.8

Hasilnya ditunjukan pada tabel 4.2 model dapat mencapai rata - rata akurasi 83.2% nilai *gamma* yang digunakan adalah nilai default dengan menggunkan variabel *scale* pada perintah *kernel*. Model sudah memiliki akurasi yang cukup baik, namun masih kurang stabil karena nilai *Precision* 79.8 % dan nilai *Recall* 88.4% memiliki selisih yang cukup jauh, menunjukan model yang kurang stabil.

3.2. Hasil SNA

a. Pembuatan *Graph*

Tabel 5. Pembuatan *Graph*

Pembuatan *Graph*

graph = defaultdict(set)
*edge*_colors = {}
for item in data_list:
  content = item.get("content", "")
  sentiment = item.get("sentiment_color", "gray")
  author_data = json.loads(item.get("author", "{}"))
  author = author_data.get("scr_name", "").lower()
 	
  mentions = re.findall(r'@(\w+)', content)
  for mention in mentions:
  target = mention.lower()
  graph[author].add(target)
  # Simpan warna *edge* berdasarkan sentimen
  *edge*_colors[(author, target)] = "green" if sentiment == "green" else "red"

Struktur *graph* di inisialisasi seperti pada tabel 5 sebagai deafultdict(set) untuk menyimpan hubungan antar pengguna seperti *node*, *edge* berdasarkan mention. Kemudian sebuah dictionary *edge*_color digunakan untuk menyimpan label warna. Untuk setiap item di data list, akan dilakukan pencarian semua mention dalam konten, kemudian tambahkan *edge* dari author ke target (*mention*) dan simpan warna *edge* di *edge*_colors(author, target).

b. Centrality dan Komunitas

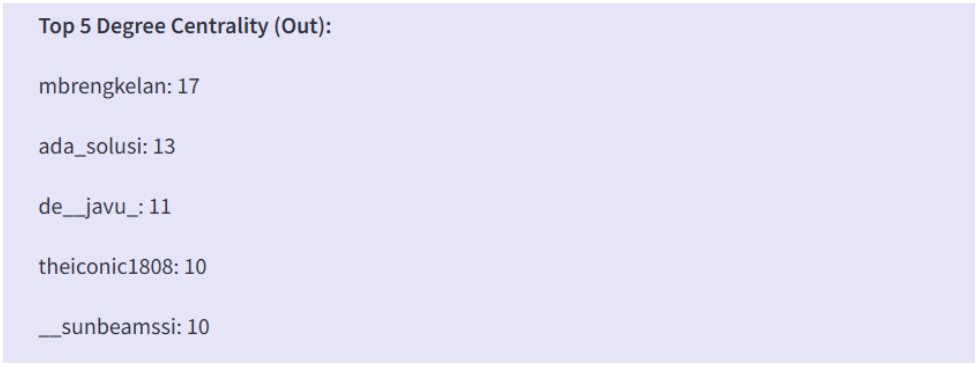
Tabel 6. Perhitungan degree

Perhitungan *degree centrality* dan komunitas

degree_out = {*node*: len(graph[*node*]) for *node* in all_*node*s}
degree_in = defaultdict(int)
for source in graph:
  for target in graph[source]:
  degree_in[target] += 1

perhitungan *degree centrality*, dengan menggunakan perhitungan *degree in* dan *degree out* yang ditunjukan pada tabel 4.8. *Degree centrality* mengkur pentingnya sebuah *node* dalam graph berdasarkan koneksi langsungnya.

Hasil perhitungan degree centrality yang diperoleh ditunjukan pada gambar 4. diperoleh lima akun yang memiliki degree tertinggi.



Gambar 4. Hasil *Degree Centrality*

Hasil gambar 4 menunjukan nilai *centrality* tertinggi yang diperoleh pada jaringan twitter dengan topik pemilu. Terdapat lima akun dengan hubungan tertinggi dengan akun lainnya dalam jaringan dengan koneksi terbanyak oleh akun @mbrengkelan. Nilai *centrality* yang tinggi menunjukan pengaruh dengan akun lainnya dalam topik pemilu di twitter, baik dalam menyebarkan opini positif dan negatif.

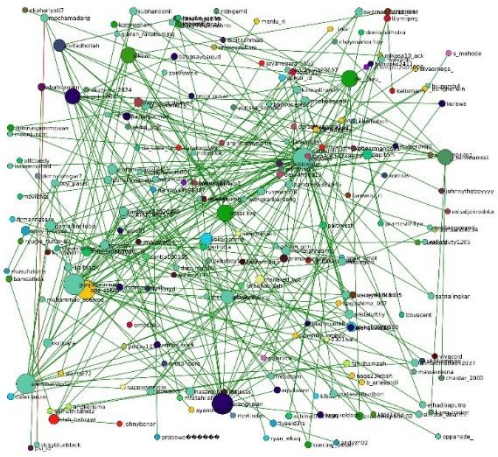
Gambar 5. Komunitas

Komunitas yang terbentuk juga dapat dilihat pada output pada gambar 5 *Node* yang berada pada satu komunitas, akan memiliki warna yang sama. Komunitas dengan jumlah akun terbanyak adalah komunitas 130, 122, dan 17. Komunitas berisikan diskusi yang muncul dari sebuah kmentar atau *quote retweet* pada *twitter*.. Pada gambar 3 hasil visualisasi menunjukan topik pembahasan cenderung mengarah kepada opini positif, mengindikasikan penyeberan positif dari setiap calon dalam rangka menaikan elektabilitas calon citra presiden.

c. Hasil Visualisasi

SNA juga sudah dapat diaplikasikan, hasil visualisasi graph yang diperoleh dari data twitter ditunjukan pada gambar 6.

p-ISSN: 2301-5373
e-ISSN: 2654-5101



Gambar 6. Hasil Visualisasi SNA

Visualisasi pada gambar 6 menunjukan topik pembahasan cenderung mengarah kepada opini positif, mengindikasikan penyeberan positif dari setiap calon dalam rangka menaikan elektabilitas calon citra presiden.

4. Kesimpulan

Dari hasil penelitian yang dilakukan dalam membuat program analisis sentiment menggunakan support vector machine dan visualisasi dengan social network analysis, diperoleh beberapa Kesimpulan :

a. Parameter yang digunakan untuk menghasilkan model dengan kinerja terbaik menggunakan nilai C = 0.05, iterasi = 10000, *kernel* linear, menghasilkan nilai akurasi sebesar 86.8%, *Precision* 86.2%, *Recall* 87%, dan *F1 - score* 86.6%Model mampu mengkasifikasikan sentimen data twitter ke dalam kelassnya dengan baik.

b. Visualisasi diapat dibuat dengan menampilkan *node* dan *edge* yang terhubung membentuk sebuah graph. *Node* memiliki ukuran dan warna yang melambangkan hubungan yang dimiliki dalam data. Hasil perhitungan *centrality* menampilkan 5 akun dengan *degree centraility* tertinggi yaitu mbrengkelan: jumlah koneksi 17, ada_solusi: jumlah koneksi 13, de__javu_: jumlah koneksi 11, basaillive: jumlah koneksi 10, dan __sunbeamssi: jumlah koneksi 10. Nilai tersebut menunjukan pengaruh dalam penyebaran opini pada jaringan yang digunakan dalam penelitian ini. Hasil visualisasi menunjukan komunitas yang terbentuk dalam jaringan, komunitas ini membentuk sub-topik yang dibahas dalam jaringan. Komunitas dengan jumlah akun terbanyak adalah komunitas 130 sebanyak 42 akun yang terlibat, komunitas122 sebanyak 20 akun yang terlibat, dan komunitas 17 sebanyak 15 akun yang terlibat.

References

[1] F. V. Sari and A. Wibowo, “ANALISIS SENTIMEN PELANGGAN TOKO ONLINE JD.ID MENGGUNAKAN METODE NAÏVE BAYES CLASSIFIER BERBASIS KONVERSI IKON EMOSI,” *Jurnal SIMETRIS*, vol. 10, no. 2, 2019.

[2] R. Tineges, A. Triayudi, and I. D. Sholihati, “Analisis Sentimen Terhadap Layanan Indihome Berdasarkan Twitter Dengan Metode Klasifikasi Support Vector Machine (SVM),” *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 4, no. 3, p. 650, Jul. 2020, doi: 10.30865/mib.v4i3.2181.

[3] R. Feldman and J. Sanger, *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. Cambridge University Press, 2007. [Online]. Available: https://books.google.co.id/books?id=U3EA_zX3ZwEC

[4] J. Leskovec, A. Rajaraman, and J. D. Ullman, *Mining of Massive Datasets*. Cambridge University Press, 2014. [Online]. Available: https://books.google.co.id/books?id=16YaBQAAQBAJ

[5] C. Campbell and Y. Ying, *Learning with Support Vector Machines*. in Synthesis lectures on artificial intelligence and machine learning. Morgan & Claypool, 2011. [Online]. Available: https://books.google.co.id/books?id=uhqmlu0lgf8C

[6] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*. in The Morgan Kaufmann Series in Data Management Systems. Elsevier Science, 2011. [Online]. Available: https://books.google.co.id/books?id=pQws07tdpjoC