

ANALISIS SENTIMEN TWEET MENGGUNAKAN BACKPROPAGATION NEURAL NETWORK

Maulana Aziz Assuja¹⁾, Saniati²⁾

^{1,2)}Program Studi Teknik Informatika, STMIK Teknokrat Lampung
Jl. H. Zainal Abidin Pagaram No. 9 – 11 Kedaton Bandar Lampung 35141
Telp. (0721) 774061, 702022 (hunting)
aziz.maulana.assuja@gmail.com¹⁾, saniati@teknokrat.ac.id²⁾

Abstrak

Analisis sentimen tweet berkembang sebagai sebuah kajian pada bidang Pengolahan Bahasa Alami yang bermanfaat mengetahui opini masyarakat terhadap sebuah topik tertentu secara otomatis. Pada penelitian ini kami mengajukan teknik analisis tweet kedalam tiga kelas (positif, negatif dan netral) menggunakan algoritma Backpropagation Neural Network. Input jaringan merupakan sejumlah kata terpilih yang diranking menggunakan skor TF*IDF. Variasi praproses term dilakukan untuk menguji performa klasifikasi sentimen. Hasil pengujian menunjukkan metode yang kami ajukan berhasil melakukan klasifikasi dengan hasil terbaik dengan akurasi 78.34% dan presisi 84.21%.

Kata kunci: Analisis sentimen, Tweet, Backpropagation, Neural Network

I. Pendahuluan

Twitter merupakan *microblog* yang banyak digunakan oleh penggunanya untuk berkomunikasi dan mengungkapkan pendapatnya (*tweet*) tentang topik tertentu. Berdasarkan *tweet-tweet* komentar yang dihasilkan, dapat ditangkap beberapa opini dari *tweeter*, baik berupa pernyataan positif, negatif atau bahkan netral (non opini). Dengan jumlah data yang relatif banyak, *tweeter* yang berasal dari berbagai sumber (terdistribusi), dan waktu penyajian yang *realtime*, maka pendapat/sentimen dari para *tweeter* ini dapat dimanfaatkan untuk berbagai hal, diantaranya mengetahui kepuasan atau keluhan *user* terhadap suatu produk atau layanan, menganalisa tren produk yang populer, melihat tingkat popularitas artis/politikus/tokoh, dsb.

Berdasarkan manfaat-manfaat tersebut maka penelitian tentang sentimen analisis pun banyak dilakukan yang sejak lebih kurang tahun 2002 [4][13] dan menjadi pembahasan penting pada bidang Pengolahan Bahasa Alami. *Sentiment analysis* atau biasa disebut juga *opinion mining* bertujuan mempelajari tentang pendapat, sentimen dan emosi yang diekspresikan pada teks [3].

Banyak metode yang digunakan untuk menerapkan sentiment analisis diantaranya dengan pendekatan *supervised learning* seperti *Naive Bayes* [9][4][2][12], *Support Vector Machine* (SVM)[1][12][4][2][11], *Maximum Entropy* [4] [2], *Neural Network* [5][14] dan *unsupervised learning*[13][17][6]. *Task* utama dalam

sentimen analisis yaitu mengklasifikasi teks *tweet* kedalam salah satu kelas yaitu positif, negatif atau netral. Pada penelitian ini digunakan pendekatan *supervised learning* dengan algoritma *Backpropagation Neural Network* (BNN) yang merupakan varian dari *Artificial Neural Network* dan telah banyak berhasil diaplikasikan termasuk pada masalah klasifikasi[16]. BNN melakukan pembelajaran dengan menyesuaikan bobot dari tiap perceptron jaringannya sehingga sesuai (menggunakan batas toleransi atau iterasi) dengan data *trainingnya*.

Klasifikasi pada teks umumnya menggunakan beberapa fitur yang mampu menjadi ciri pengelompokan yang baik seperti leksikon atau *term*[4][10][2][5] baik secara *unigram*, *bigram*, *emoticon*[12][17], *POS tag*[4][2][1], *hash tag*[1] dan lain sebagainya. Leksikon atau *term* dalam teks merupakan fitur yang paling banyak digunakan, namun pemilihannya sebagai representasi tiap kelas target menjadi tidak konsisten jika jumlah data *trainingnya* berubah. Untuk mendapatkan *term-term* yang paling relevan tersebut dengan jumlah yang konsisten, maka pada penelitian digunakan perankingan *term* terbaik menggunakan teknik TF*IDF (*Term Frequency Invers Document Frequency*) seperti pada beberapa penelitian sebelumnya[10][12].

II. Tinjauan Pustaka

2.1. Sentimen Analisis

Sentimen analisis atau *opinion mining* didefinisikan sebagai bidang ilmu yang mempelajari bagaimana mengekspresikan sentimen, opini atau pendapat dan emosi yang diekspresikan didalam teks [3]. Terdapat beberapa topik pembahasan dalam sentimen analisis, salah satu yang paling sering diteliti yaitu klasifikasi sentimen. Topik ini berfokus pada kegiatan pengelompokan sentimen berdasarkan teks opini terhadap pembahasan masalah tertentu (*movie*, produk, tokoh, kejadian, dsb).

Beberapa penelitian melakukan pengelompokan menjadi beberapa jenis kelas target, 1) dengan dua kelas (positif dan negatif) [4][2][6][11] [5]; 2) tiga kelas (positif, negatif dan netral)[7][1][12], dan 3) lima kelas (sangat negatif, negatif, netral, positif dan sangat positif)[5]. Namun menurut riset Go [2] yang melakukan klasifikasi kedalam kelas negatif dan positif saja, menyimpulkan jika pengelompokan kedalam kelas netral tetap perlu dilakukan, karena pada beberapa *tweet*

ditemukan sentimen yang tidak cenderung bernilai positif maupun negatif.

2.2. Tweets

Sentimen analisis dilakukan terhadap opini yang diberikan terhadap review pembahasan tertentu, yang pada masa kekinian, banyak orang mengomentari banyak hal dengan memanfaatkan media sosial *online* seperti twitter. Pada twitter tiap orang yang terdaftar dapat mengirimkan dan membaca teks pendek yang biasa disebut *tweet*.

Tweet memiliki karakteristik yang unik berdasarkan laporan analisis Go [2] baik pada panjang teks, ketersediaan data, model bahasa, dan domain. Panjang teks pada *tweet* maksimum 140 karakter dengan rata-rata *tweet* terdiri dari 14 kata atau 78 karakter. Ketersediaan data, dengan Twitter API dapat dikumpulkan begitu banyak *tweet* sebagai data set. Model bahasa pada *tweet* tidak terstruktur bahkan terkadang ada kesalahan pengetikan. Serta domain pembahasan pada twitter sangat luas terhadap banyak sekali topik apapun, berbeda dengan penelitian lain pada domain terbatas seperti review *movie*.

2.3. TF*IDF (Term Frequency Invers Document Frequency)

TF*IDF merupakan teknik pembobotan kata yang berbasis pada statistik kemunculan kata dan tingkat kepentingan dokumen yang mengandungnya. Pembobotan yang diadaptasi dari pendekatan temu-kembali informasi (*information retrieval*)[8] ini merupakan hasil dari perkalian *term frequency* dan *inverse document frequency* yang tiap nilainya didapatkan dengan persamaan (1).

$$TF * IDF = TF(w_i, d) * IDF(w_i) \quad (1)$$

Dimana:

w_i = kata ke i

d = dokumen

$TF(w_i, d)$ =jumlah kemunculan kata w_i pada dokumen d

$IDF(w_i)$ = *Invers Document Frequency* dari kata w_i .

Pembobotan dihitung untuk tiap kata, *Term Frequency* menunjukkan bahwa semakin tinggi kemunculan kata pada sebuah dokumen maka semakin penting kata tersebut mewakili dokumen. Nilai *Inverse Document Frequency* memberikan indikasi bahwa jika kemunculan sebuah kata mempunyai frekuensi yang tinggi pada dokumen tertentu saja maka kata tersebut menjadi ciri penting dan mewakili dokumen tersebut. Sedangkan jika kata muncul pada seluruh dokumen maka kata tersebut bersifat umum atau tidak mewakili dokumen manapun dan memiliki nilai IDF 0. Dimana nilai IDF dapat didapatkan dengan persamaan (2).

$$IDF(w_i) = \log \frac{|D|}{DF(w_i)} \quad (2)$$

Dimana:

w_i = kata ke i

$|D|$ = jumlah seluruh dokumen

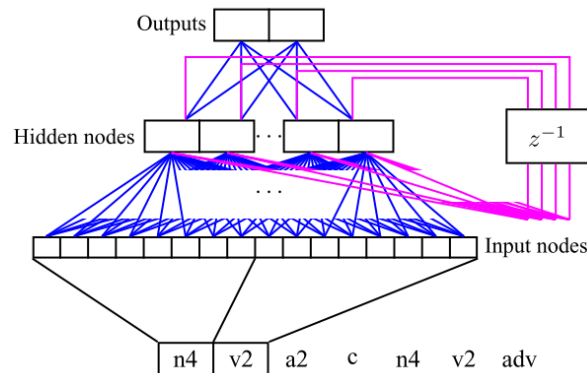
$IDF(w_i)$ = jumlah dokumen yang mengandung kata w_i .

Pada penelitian sentimen analisis, penggunaan dokumen pada persamaan diwakili oleh *tweet* yang digunakan. Pembobotan TF*IDF membantu mengambil kata-kata dari *tweet* yang paling mewakili kelasnya.

2.4. Neural Network

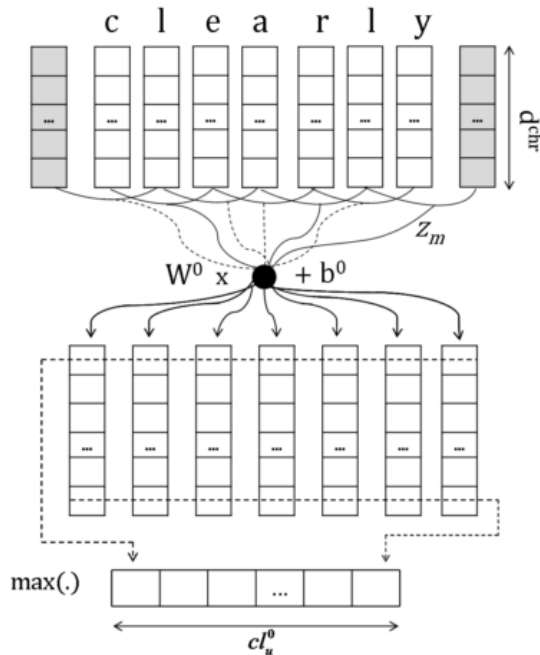
Penelitian sentimen analisis dengan pendekatan *supervised learning* dengan varian metode *Artificial Neural Network* [5][14] menunjukkan hasil yang baik lebih dari 80% bahkan mencapai 86%[5]. Pada penelitian klasifikasi lainnya (kelas *gramatical* dan *ungramatical*)[15] juga menunjukkan hasil yang mencapai lebih dari 95%. Hal ini menunjukkan jika pendekatan *Artificial Neural Network* memberikan hasil yang memuaskan karena termasuk metode mampu menangani *training sample* dengan *noise* yang banyak (kebutuhan akan sistem yang *robust* terhadap *noise*)[16].

Backpropagation Neural Network (BNN) adalah versi modifikasi *Artificial Neural Network* yang menambahkan propagasi *error* dengan bergerak mundur (dari lapisan *output* menuju lapisan *input*). Penggunaan propagasi balik dapat mempercepat konvergensi sehingga model dengan jumlah data input yang banyak pun dapat dengan cepat didapatkan. Hal ini berkaitan dengan jumlah layer dan jumlah data input pada arsitektur klasifikasi yang digunakan, karena penelitian sebelumnya data input dapat berjumlah banyak tergantung levelnya.



Gambar 1. Arsitektur dengan input level kata [15]

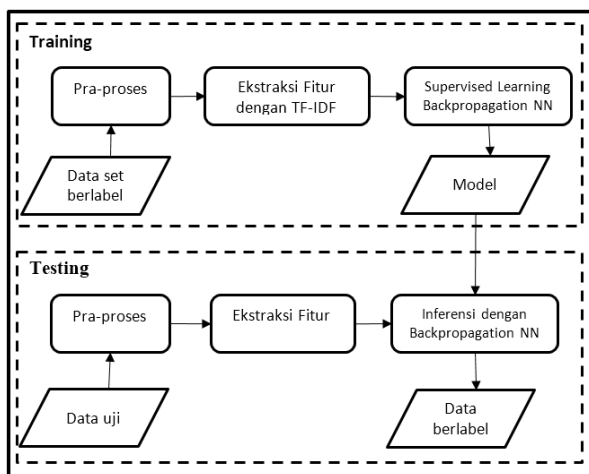
Ada yang input layernya menggunakan level kata[15][5][14] dengan arsitektur seperti pada gambar 1. bahkan pada level karakter[5] seperti pada gambar 2.



Gambar 2. Arsitektur dengan *input level karakter* [9]

III. Analisis Sentimen Tweet Menggunakan Backpropagation Neural Network

Pada penelitian sentimen analisis *tweet* ini digunakan arsitektur rancangan yang terlihat pada gambar 3.



Gambar 3. Arsitektur rancangan Sentimen Analisis Twitter menggunakan *Backpropagation Neural Network*

Berdasarkan gambaran rancangan arsitektur, ada dua bagian utama pada penelitian ini yaitu fase *Training* dan *Testing*. Pada fase *training*, data set berlabel melewati 4 proses diantaranya yaitu pra-proses, tokenisasi, ekstraksi fitur dengan TFIDF dan *supervised learning* dengan *Backpropagation Neural Network* sehingga menghasilkan model. Pada fase *testing*, data uji akan melewati beberapa proses seperti praproses, tokenisasi dan inferensi menggunakan *Backpropagation Neural Network* yang membutuhkan data model serta menghasilkan data berlabel. Berikut dibahas lebih detail untuk bagian pada tiap fase.

A. Praproses

Pada pengolahan teks, umumnya dilakukan tahap praproses yang ditujukan untuk membersihkan dan menyamakan format teks sehingga pada proses pengolahan berikutnya lebih terstruktur. Tahap praproses ini digunakan baik pada fase *training* (untuk data berlabel) maupun pada fase *testing* (untuk data uji). Beberapa praproses yang dilakukan yaitu transformasi kata/*term* gaul, *cleaning* dan *normalization*, *stopword removal*, *stemming*, *soundex* dan tokenisasi.

Transformasi kata gaul dilakukan karena *tweet* ditulis sebagai teks yang tidak baku, sehingga banyak ditemukan kata-kata yang tidak formal/gaul yang perlu dinormalisasi ke dalam kata yang sesuai dengan KBBI (Kamus Besar Bahasa Indonesia). Pada proses ini, digunakan kamus bahasa gaul dari situs <http://www.snipertechno.com/2012/12/kamus-alay-terbaru-2013.html> dan <http://adhitez12.blogspot.com/2012/12/kamus-bahasa-alay-lengkap.html> untuk melakukan transformasi ke penulisan kata yang baku.

Cleaning dan *normalization* dilakukan untuk menghapus “mention” (@), “retweet”(RT) dan digit pada data set yang dianggap tidak memiliki peran dalam penentuan kelas dari *tweet*. Normalisasi dilakukan dengan *me-lower case* kan semua huruf untuk mendapatkan format yang seragam, serta penggantian semua kata-kata tidak sopan dengan kata “makian” untuk mengelompokkan ungkapan-ungkapan kekesalan tersebut.

Stopword merupakan kata-kata yang tidak memiliki arti seperti “itu”, “ini”, “dan”, “yang”, dst. *Stopword removal* atau penghapusan *stopword* dilakukan karena frekuensi kemunculannya yang tinggi namun tidak memiliki kontribusi, dikhawatirkan dapat mengganggu proses klasifikasi. Daftar *stopword* didapatkan dari <http://yudiwbs.wordpress.com/2008/07/23/stop-words-untuk-bahasa-indonesia/>, juga dilakukan beberapa penambahan kata-kata yang dianggap perlu seperti “sih”, “nah”, “kok”, dsb.

Stemming atau penghilangan imbuhan dilakukan untuk mengambil bentuk dasar dari sebuah kata yang telah mengalami morphology karna penambahan imbuhan. Misalnya ‘membeli’ dan ‘belikan’ menjadi ‘beli’. Tujuan dari tahap ini yaitu menangani jika ada dua atau lebih kata berbeda namun memiliki makna yang hampir sama tetap diperlakukan sebagai 1 kata yang sama.

Soundex dilakukan untuk mengubah kata kedalam bentuk kode suara, sehingga untuk kata-kata yang pengucapannya mirip akan memiliki kode yang sama. Pada *soundex* juga hanya mengolah huruf-huruf konsonan dari kata, sedangkan huruf vokal dihilangkan. Berdasarkan analisa data, langkah ini dianggap perlu karena banyak kata-kata dalam data set yang dituliskan secara tidak lengkap/disingkat seperti ‘dapat’ ditulis ‘dpt’. Ada juga beberapa kata yang sama namun ditulis secara berbeda seperti ‘malas’ dan ‘males’. Langkah ini

akan menghasilkan kode yang sama baik untuk kata disingkat maupun kata sama yang ditulis sedikit berbeda.

Tokenisasi dilakukan untuk memenggal tiap kata atau *term* yang terpisah oleh tanda baca seperti spasi, titik, koma, tanda tanya, dsb. *Term-term* atau token-token tersebut dipilih kembali sebagai kata yang paling mewakili tiap label sentimen pada tahap ekstraksi fitur untuk fase *training*. Sedangkan pada fase *testing*, *term* selanjutnya diinferensi dengan algoritma *Backpropagation Neural Network*.

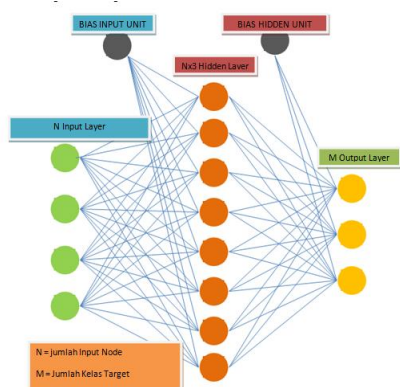
B. Ekstraksi Fitur dengan TF*IDF

Keluaran dari praproses dan tokenisasi menghasilkan jumlah *term* yang banyaknya sesuai dengan data set masukannya. *Term-term* tersebut akan mewakili atau menjadi fitur pada *text classification* menggunakan *backpropagation*. Namun tidak semua *term* dijadikan fitur, fitur didapatkan dengan cara mengambil 50 kata/*term* terbaik dari data *training* yang mewakili tiap kelas target (netral, positif, dan negatif). Dari setiap kelompok *tweet* berdasarkan kelas target akan memiliki 50 *term* yang masing-masing akan di*merge* secara *distinct*. Total *term* yang didapatkan akan menjadi fitur (*input perceptron*) pada jaringan *backpropagation*.

Perhitungan yang digunakan untuk mendapatkan top 50 *term* terbaik dari tiap kelompok *tweet* menggunakan pembobotan TF*IDF (*Term Frequency-Inverse Document Frequency*). Nilai TF*IDF yang tinggi menggambarkan bahwa *term* tersebut merupakan *term* yang penting dari suatu kelompok *tweet* (kelompok berdasar kelas target), dilihat dari kemunculannya yang intens dalam kelompoknya namun tidak umum untuk kelompok lain. Misalnya, kata “makian” akan menjadi penting untuk kelas negatif tetapi tidak banyak dijumpai pada kelas lainnya.

C. Supervised Learning Backpropagation Neural Network

Pada penelitian sentiment analisis ini, digunakan metode pembelajaran terbimbing dengan algoritma *Backpropagation Neural Network* (BNN) dengan arsitektur seperti gambar 4. Fungsi aktivasi untuk setiap *perceptron* yang dipilih ialah sigmoid.



Gambar 4. Arsitektur BNN untuk Sentimen Analisis Tweet

Setiap node pada input merepresentasikan satu buah kata atau term yang dijadikan fitur. Sehingga jika terdapat 50 term yang akan dijadikan fitur, maka input pada BNN merupakan array double sejumlah 50 elemen. Pada fase *training*, *term-term* tiap *tweet* dipetakan sesuai dengan 50 *term* terpilih pada node input, jika terdapat *term* yang sama maka akan bernilai 1 dan 0 jika sebaliknya. Selain itu, tiap-tiap *tweet* juga telah terlabeli sesuai kelasnya (negatif, positif atau netral). Berdasarkan data set berlabel tersebut akan dilakukan pembelajaran terbimbing sehingga diakhir didapatkan bobot yang sesuai dengan data pembelajaran yang kemudian disimpan sebagai model.

D. Ekstraksi Fitur

Tahap ekstraksi fitur ini dilakukan terhadap data uji (*tweet* tidak berlabel) dengan cara menyamakan term yang dihasilkan dari praproses dengan daftar term yang sudah diekstraksi dari fase *training* menggunakan TFIDF. Jika ada term yang sama maka elemen fitur bernilai 1, jika tidak maka bernilai 0.

E. Inferensi dengan Backpropagation Neural Network

Proses ini dilakukan untuk menginferensi atau mengklasifikasi fitur dari data uji, terhadap tiga kelas yang ada. Tahap ini membutuhkan model dari BNN dari hasil *training*. Pada tahap ini model mengandung bobot yang sudah sesuai dengan hasil *training*, yang kemudian fitur input dihitung terhadap bobot-bobot di tiap layer sehingga *outputnya* merepresentasikan kelas dari data uji. Jika hasil akhir *output* secara berurutan ialah 100 maka data uji berlabel kelas negatif, 010 kelas positif dan 001 kelas netral.

IV. Hasil dan Pembahasan

4.1. Skenario Eksperimen

Ketentuan penggunaan *Backpropagation Neural Network* pada eksperimen ini yaitu, misal

1. jumlah *term* yang diekstraksi fitur dengan TF*IDF (N) = 90
2. *input Layer* → N = 90 (+ 1 Bias Unit) = 91 *Perceptron*
3. *hidden Layer* → N x 3 = 90 * 3 (+ 1 Bias Unit) = 271 *Perceptron*
4. *output Layer* → M = 3 (Negatif, Positif, Netral)
5. *learning rate* → 0.1
6. *stopping criterion* untuk *training* menggunakan BNN adalah dengan galat 10% atau maksimal iterasi 500.

Pada eksperimen ini total data set yang digunakan yaitu 944 *tweet* dimana *tweet-tweet* tersebut terdiri dari kelas netral (500 *tweet*), positif (254 *tweet*), dan negatif (190 *tweet*). Seluruh *tweet* kemudian dibagi menjadi 2 bagian, 2/3 untuk data *training* (630 *tweet*) dan 1/3 untuk data *testing* (314 *tweet*).

Pada pengujian dilakukan uji coba terhadap beberapa jumlah fitur *term* yang diambil yaitu sebanyak 30, 50, 70 dan 100 *term* per kelas target. Skema selanjutnya yaitu

kedua data *training* dan data *testing* akan dibagi menjadi 5 grup yaitu:

- Grup I :data hanya di tokenisasi, tanpa praproses lain.
- Grup II :data dengan praproses transformasi kata gaul, *cleaning & normalization*, tokenisasi.
- Grup III :data dengan praproses transformasi kata gaul, *cleaning & normalization*, *stopword removal*, tokenisasi.
- GrupVI :data dengan praproses transformasi kata gaul, *cleaning & normalization*, *stopword removal*, *stemming*, tokenisasi.
- Grup V :data dengan praproses transformasi kata gaul, *cleaning & normalization*, *stopword removal*, *stemming*, *soundex*, tokenisasi.

Eksperimen ini diukur menggunakan 3 jenis pengukuran yaitu:

1. *Correct classified* (C), didapatkan dari prosentase hasil kata yang terlabeli dengan benar dibagi total seluruh kata yang diuji.
2. *Precision* (P), didapatkan dari prosentase penjumlahan *precision* semua kelas dibagi total kelas. Sedangkan nilai *precision* tiap kelas didapatkan dengan menghitung jumlah kata yang terklasifikasi benar terhadap sebuah kelas dibagi total kata yang terklasifikasi kekelas tersebut.
3. *Recall* (R), didapatkan dari prosentase penjumlahan *recall* semua kelas dibagi total kelas. Sedangkan nilai *recall* tiap kelas didapatkan dengan menghitung jumlah kata yang terklasifikasi benar terhadap sebuah kelas dibagi jumlah kata yang sebenarnya merupakan member dari kelas tersebut.

4.2. Hasil

Setelah dilakukan eksperimen didapatkan hasil untuk tiap grup dengan variasi jumlah *term* perkelas target seperti pada tabel-tabel berikut.

Tabel 1. Hasil pengujian Grup I

| | 30 term /kelas target (%) | 50 term /kelas target (%) | 70 term /kelas target (%) | 100 term /kelas target (%) | rata-rata (%) |
|----------|----------------------------------|----------------------------------|----------------------------------|-----------------------------------|----------------------|
| C | 60.82 | 69.11 | 68.47 | 73.56 | 67.99 |
| P | 59.81 | 72.04 | 68.17 | 76.99 | 69.25 |
| R | 57.08 | 62.24 | 63.28 | 66.73 | 62.33 |

Tabel 1. adalah hasil pengukuran untuk Grup I dimana semua data set tidak dilakukan praproses kecuali tokenisasi. Dari tabel tersebut terlihat jika hasil pengukuran terbaik terjadi pada pengujian dengan 100 *term*/kelas target. Sedang hasil terendah dari pengujian dengan 30 *term*/ kelas target.

Tabel 2. Hasil pengujian Grup II

| | 30 term /kelas target (%) | 50 term /kelas target (%) | 70 term /kelas target (%) | 100 term /kelas target (%) | rata-rata (%) |
|----------|----------------------------------|----------------------------------|----------------------------------|-----------------------------------|----------------------|
| C | 72.61 | 78.34 | 77.07 | 77.07 | 76.27 |
| P | 72.12 | 84.21 | 82.93 | 83.73 | 80.74 |
| R | 68.26 | 72.63 | 70.54 | 70.63 | 70.51 |

Hasil pengukuran untuk pengujian Grup II dimana data set melewati tahap praproses transformasi kata gaul, *cleaning & normalization*, tokenisasi ada pada tabel 2. Secara rata-rata hasil pada Grup II lebih baik dibandingkan dengan Grup I, dengan hasil paling baik dengan perbedaan tipis ditujukan pada pengujian dengan 50 *term*/kelas target.

Tabel 3. Hasil pengujian Grup III

| | 30 term /kelas target (%) | 50 term /kelas target (%) | 70 term /kelas target (%) | 100 term /kelas target (%) | ata-rata (%) |
|----------|----------------------------------|----------------------------------|----------------------------------|-----------------------------------|---------------------|
| C | 76.43 | 74.20 | 78.98 | 75.15 | 76.19 |
| P | 80.99 | 79.73 | 81.22 | 80.98 | 80.73 |
| R | 71.01 | 67.72 | 75.35 | 68.43 | 70.62 |

Skenario untuk Group III hampir sama dengan Grup II dengan penambahan praproses *stopword removal*. Hasil pengukuran secara umum pada Grup III tidak jauh berbeda dengan Grup II. Dimana hasil terbaik diperoleh pada pengujian 70 *term*/ kelas target.

Tabel 4. Hasil pengujian Grup IV

| | 30 term /kelas target (%) | 50 term /kelas target (%) | 70 term /kelas target (%) | 100term /kelas target (%) | rata-rata (%) |
|----------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------|
| C | 76.43 | 75.47 | 76.11 | 78.02 | 76.5 |
| P | 80.86 | 81.71 | 81.13 | 80.03 | 80.93 |
| R | 71.10 | 68.73 | 69.71 | 74.15 | 70.92 |

Pada eksperimen untuk Grup IV juga masih memiliki rata-rata hasil pengukuran yang mirip dengan Grup II dan III. Data set pada Grup IV melalui tahap praproses transformasi kata gaul, *cleaning & normalization*, *stopword removal*, *stemming*, dan tokenisasi. Hasil dengan *correct classified* dan *recall* terbaik yaitu pengujian untuk 100 *term*/kelas target.

Tabel 5. Hasil pengujian Grup V

| | 30 term /kelas target (%) | 50 term /kelas target (%) | 70 term /kelas target (%) | 100term /kelas target (%) | rata-rata (%) |
|----------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------|
| C | 76.43 | 75.47 | 76.11 | 78.02 | 76.5 |
| P | 80.86 | 81.71 | 81.13 | 80.03 | 80.93 |
| R | 71.10 | 68.73 | 69.71 | 74.15 | 70.92 |

| | | | | | |
|----------|-------|-------|-------|-------|-------|
| C | 71.33 | 72.93 | 71.33 | 72.61 | 72.05 |
| P | 73.13 | 73.64 | 74.05 | 76.53 | 74.33 |
| R | 66.38 | 69.43 | 66.47 | 66.43 | 67.17 |

Pengujian Grup terakhir pada tabel 5. mengalami penurunan hasil pengukuran, dimana data set mengalami semua praproses yaitu transformasi kata gaul, *cleaning & normalization*, *stopword removal*, *stemming*, *soundex*.

V. Simpulan

Hasil pengujian pada sentimen analisis menunjukkan jika tahap praproses diperlukan untuk meningkatkan akurasi hasil klasifikasi. Sedangkan bentuk praproses yang paling baik ditunjukkan oleh penggunaan transformasi kata gaul, *cleaning (retweet dan mention)* dan normalisasi (*case folding* dan perubahan makian). Sedangkan praproses *soundex* memberikan dampak penurunan akurasi. Sementara jumlah fitur atau *term* sebagai data *input* tidak banyak berpengaruh terhadap hasil. Secara keseluruhan, metode *Backpropagation Neural Network* memberikan performa yang baik.

Daftar Pustaka

- [1] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R. Passonneau, "Sentiment analysis of Twitter data," *Assoc. Comput. Linguist.*, 2011.
- [2] A. Go, R. Bhayani, and L. Huang, "Twitter Sentiment Classification using Distant Supervision," 2009.
- [3] B. Liu, "Sentiment Analysis and Subjectivity," 2010.
- [4] B. Pang, L. Lee, H. Rd, and S. Jose, "Thumbs up? Sentiment Classification using Machine Learning Techniques," 2002.
- [5] C. N. dos Santos and M. Gatti, "Deep Convolutional Neural Networks for Sentiment Analysis of Short Texts," *Coling-2014*, 2014.
- [6] C. Scheible, "Unsupervised Sentiment Analysis with a Simple and Fast Bayesian Model using Part-of-Speech Feature Selection," vol. 2012, 2012.
- [7] E. Kouloumpis, T. Wilson, and J. Moore, "Twitter sentiment analysis: The good the bad and the omg!," *Proc. Fifth Int. AAAI Conf. Weblogs Soc. Media (ICWSM 11)*, 2011.
- [8] G. Salton and M. McGill, *Introduction to modern Information Retrieval*. McGraw-Hill, 1983.
- [9] I. F. Rozi, S. Hadi, and E. Achmad, "Implementasi Opinion Mining (Analisis Sentimen) untuk Ekstraksi Data Opini Publik pada Perguruan Tinggi," vol. 6, no. 1, 2012.
- [10] J. Martineau, J. Martineau, T. Finin, T. Finin, C. Fink, C. Fink, C. Piatko, C. Piatko, J. Mayfield, J. Mayfield, Z. Syed, Z. Syed, Others, and Others, "Delta TFIDF: An Improved Feature Space for Sentiment Analysis," *Proc. Second Int. Conf. Weblogs Soc. Media (ICWSM)*, vol. 29, no. May, 2008.
- [11] K. Ghag, "SentiTFIDF – Sentiment Classification using Relative Term Frequency Inverse Document Frequency," vol. 5, no. 2, 2014.
- [12] P. Aliandu, "Sentiment analysis on indonesian tweet," 2013.
- [13] P. D. Turney, "Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews," no. July, 2002.
- [14] R. Socher, A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank," 2013.
- [15] S. Roa and F. Nino, "Classification of Natural Language Sentences using Neural Networks," 2003.
- [16] T. M. Mitchell, *Machine Learning*. McGraw-Hill, 1997.
- [17] X. Hu, J. Tang, H. Gao, and H. Liu, "Unsupervised Sentiment Analysis with Emotional Signals," 2013.

Maulana Aziz Assuja, lahir di Sindang Anom, 24 Nopember 1986. Memperoleh gelar D4 di Program Studi Teknik Informatika, Institut Teknologi Sepuluh November di Surabaya pada tahun 2009. Kemudian pada tahun 2015 memperoleh gelar S2 di Program Studi Informatika, Institut Teknologi Bandung. Saat ini bekerja aktif sebagai staf pengajar di STMIK-AMIK Teknokrat Lampung.

Saniati, lahir di Tarakan, 14 Februari 1988. Memperoleh gelar D4 di Program Studi Teknik Informatika, Institut Teknologi Sepuluh November di Surabaya pada tahun 2009. Kemudian pada tahun 2014 memperoleh gelar S2 di Program Studi Informatika, Institut Teknologi Bandung. Saat ini bekerja aktif sebagai staf pengajar di STMIK-AMIK Teknokrat Lampung.