SEMINAR NASIONAL TEKNOKA

Vol. 7, 2022
ISSN No. **2502-8782**

Klasifikasi Komentar *Abusive* Dan *Hate Speech* Teks Twitter Menggunakan Metode *Convolutional Neural Network*

Indri Pangestuti[1)], Surya Agustian[2)]
[1,2)]Teknik Informatika , Fakultas Sains & Teknologi, UIN Sultan Syarif Kasim Riau
[1,2)]Jl. H.R. Soebrantas No. 155 Km 15, Simpang Baru, Panam, Pekanbaru
Website: , E-mail: [1)]11850124817@students.uin-suska.ac.id, [2)]surya.agustian@uin-suska.ac.id

Abstrak

Twitter salah satu media sosial yang banyak digunakan saat ini, terutama untuk mengeluarkan pendapat secara bebas. Tidak adanya mekanisme penyeleksian kata-kata dan kalimat pada twitter, menyebabkan siapa saja dapat melontarkan ujaran kebencian maupun penggunaan bahasa kasar terhadap orang atau golongan lain. Ujaran kebencian dan bahasa kasar sering ditemukan pada twitter dalam berbagai kasus maupun topik percakapan, seperti perseteruan antar kelompok, ketidakpuasan terhadap produk, sampai kepada protes terhadap kebijakan pemerintah. Penelitian ini mengusulkan penggunaan deep learning untuk mengklasifikasi apakah tweet mengandung ujaran kebencian atau bahasa kasar. Metode yang digunakan Convolutional Neural Network dengan input fitur teks word embedding word2vec. Beberapa skenario pengujian dilakukan untuk mendapatkan hasil optimal dengan melakukan training pada 90% data. Model final yang dipilih diterapkan terhadap data testing sebanyak 10% dari data set, memperoleh akurasi untuk kelas hate speech sebesar 84,92%, dan untuk kelas abusive 91,47%. Hasilnya sangat baik dan kompetitif bila dibandingkan dengan metode-metode machine learning konvensional.

***Keyword**: deep learning, klasifikasi, convolutional neural network, ujaran kebencian, bahasa kasar*

Abstract

Twitter is one of the most widely used social media today, especially for expressing opinions freely. There is no mechanism for selecting words and sentences on Twitter, causing anyone to utter hate speech or use offensive language against other people or groups. We often find hate speech and offensive language on Twitter in various cases and topics of conversation, such as inter-group feuds, product dissatisfaction, to protests against government policies. This study proposes the use of deep learning to classify whether tweets contain hate speech or offensive language. The method used is the Convolutional Neural Network with the text feature word embedding word2vec input. Several test scenarios were carried out to obtain optimal results by conducting training on 90% of the data. The selected final model was applied to 10% of the data testing data, obtaining an accuracy of 84.92% for the hate speech class, and 91.47% for the abusive class. These results are very good and competitive when compared to conventional machine learning methods.

***Kata kunci**: deep learning, classification, convolutional neural network, hate speech, abusive language*

1 PENDAHULUAN

Twitter adalah salah satu jenis media *social microblogging* dimana penggunanya dapat menulis dan mempublikasikan aktivitas dan opini mereka. Secara historis, keberadaan dan kemunculan media *social* telah memberi twitter ruang yang tetap, hingga 140 karakter. Seperti media *social* lainnya, pengguna twitter dapat terhubung dengan orang lain, menyebarkan informasi yang mendukung pendapat dan pandangan orang lain, berpartisipasi dalam diskusi tentang topik yang sedang tren, dan menggunakan tagar tertentu. Orang-orang bisa menjadi bagian dari topik tersebut dengan berpartisipasi dalam tweet [1].

Jumlah akun media sosial sekarang ini mengantongi kurang lebih 4,2 miliar orang, dan dari kisaran tersebut pengguna media sosial sering mengarahkan bahasa yang menyinggung atau bisa disebut dengan ujaran kebencian *(Hate Speech)* dan menurut argumennya tidak berpikir panjang memakai bahasa kasar *(Abusive)*, ini dikarenakan tidak terlepas pada pribadi yang pada umumnya mempunyai watak positif dan negatif [2].

Ujaran kebencian adalah ungkapan kebencian terhadap individual atau kelompok tertentu dan digunakan untuk mempermalukan atau mengekspresikan ungkapan kebenciannya bisa dengan membenci atau memaki. *Hate speech* juga termasuk dalam raut wajah yang menyulut kebencian rasial,

Dipresentasikan pada Tanggal 3 Desember 2022

kebencian yang berkaitan dengan SARA berbasis intoleransi, dan diskriminasi terhadap kelompok minoritas, pendatang bahkan masyarakat pendatang khususnya [13]. *Abusive language* memiliki makna yakni bahasa jorok atau bahasa kasar yang berasal dari sebuah keadaan, bagian tubuh, kegiatan, binatang, pekerjaan, dan makhluk astral [2]

Penelitian tentang ujaran kebencian telah banyak dilakukan sebelumnya, antara lain menggunakan metode *Convolutional Neural Network* (CNN) dengan modifikasi pada *word embedding* dengan menggunakan *fasttext* [4]. Pada tahap *prepocessing* dilakukan beberapa kombinasi menggunakan *slang words, stop words*, dan *stemming*. Hasil terbaik digunakan untuk membentuk *word embeddings* menggunakan *fasttext* dan *library keras*. Hasilnya, *embedding* dari *fasttext* menunjukkan kinerja yang lebih baik dengan akurasi hingga 86%, keras hanya mencapai 71%, dengan proses klasifikasi yang sama menggunakan *convolutional neural network* (CNN)

Contoh penelitian lainnya yang telah dilakukan untuk menyelesaikan kasus klasifikasi ujaran kebencian dan kata kasar menggunakan beragam algoritma. Mendeteksi *hate speech* dan kata-kata kasar di twitter Indonesia menggunakan algoritma *decision tree* dan menggunakan fitur khusus dan fitur tekstual. Masing-masing hasil akurasi adalah buat pembagian data latih dan data uji 90:10 rata-rata mengalami kenaikan untuk ke-3 kelas sebesar 69,77% menjadi 70,48% kemudian untuk pembagian data latih dan data uji 80:20 rata-rata akurasi juga mengalami kenaikan dari 69,35% menjadi 69,54% [3]

Rujukan lainnya yaitu klasifikasi dengan algoritma *refresi logistic* dan fitur *word embedding*. Salah satu eksperimen dilakukan buat memperoleh model terbaik sehingga pengujian dapat diperoleh secara maksimal. Tingkat akurasi rata-rata ketiga kelas tersebut adalah 75,59%, buat kelas *hate speech* 75,86%, kelas *abusive* 80,05%, kelas level 70,86% dengan komposisi 90:10. [4]

Berdasarkan permasalahan tersebut, penelitian ini menggunakan metode *convolutional neural network* untuk mencari fitur dan parameter yang optimal untuk meningkatkan hasil akurasi klasifikasi,

2 LANDASAN TEORI

2.1 Hate Speech (HS)

Hate speech adalah ucapan, ekspresi atau ucapan kecurigaan, perselisihan, dan kejelekan yang diarahkan sama seseorang atau sekelompok orang atas dasar mengungkapkan perasaan yang sebenarnya. Ujaran kebencian adalah berbagai macam bentuk ekspresi, baik lisan maupun tulisan yang menyebarkan, mendorong, mendukung atau membenarkan kebencian atas dasar intoleransi atau dasar agama[7]. Ujaran kebencian biasanya menyebar dengan cepat lewat media sosial dan bisa menyebabkan mis-informasi atau masyarakat biasa menyebutnya dengan gosip belaka dan ini menimbulkan asumsi yang salah. Hal ini mungkin disebabkan oleh lambatnya respon negara terhadap regulasi kemajuan teknologi informasi, yang paling utama media sosial sebagai benih ujaran kebencian [8].

2.2 Abusive Language (AB)

Menurut KBBI, kata-kata *abusive* terkandung dalam klausa kata sifat (abjectives) dan berarti *abusive* ini diartikan dengan kata/kalimat yang tidak sopan dan sadis. Sedangkan pendapat kamus Bahasa inggris terjemahan Bahasa Indonesia, kata *abusive* berarti kekejaman, keburukan, kerugian dan penghinaan. Bahasa *abusive* adalah kata atau frasa yang menyinggung, melukai hati, dan perbuatan yang menyerang. Bahasa *abusive* adalah bahasa yang biasa digunakan untuk menyerang orang lain baik secara individual atau sekelompok dengan kalimatnya yang menyakiti hati,[11]. Ujaran kebencian juga dapat mencakup ungkapan atau kata-kata kasar/menyinggung yang bisa membangkitkan emosi dan memancing permusuhan[7].

2.3 Twitter

Berdasarkan buku yang ditulis oleh Hadi, pengertian twitter adalah situs *microblogging* yang memungkinkan bagi pengguna untuk mengirimkan teks hingga 140 kata karakter melalui SMS, instan *messenger*, dan email. Inti dari twitter adalah tweet, yaitu tulisan yang panjangnya maksimal 140 karakter yang diposting ke twitter [10]

2.4 Klasifikasi

Klasifikasi yaitu kata resapan dari bahasa Belanda, *classificatie*, yang berawal dari bahasa Prancis *classification*. Sebutan ini mengarah pada sebuah metode buat merapikan data secara sistematis dan menurut beberapa ketentuan atau norma yang sudah ditentukan [11].

Dalam KBBI, klasifikasi adalah pengaturan secara sistematis ke dalam kelompok-kelompok atau kelas-kelas menurut peraturan atau kriteria yang telah ditentukan. Secara harfiah dapat dikatakan bahwa klasifikasi adalah pembagian sesuatu ke dalam kelas-kelas. Menurut science, klasifikasi yaitu tahapan pengelompokan objek bersumber pada perumpamaan dan perbedaannya[11].

2.5 Komentar

Komentar yaitu bagaimana seseorang mengeluarkan argumennya, bisa juga dengan mengekspresikan pendapatnya. Raut muka biasa menjadi masalah yang paling unggul karena ditemui raut wajah yang melampaui dan cuma fokus pada

ISSN No. 2502-8782

faktor kedengkian yang tidak ada alasannya. Raut wajah yang fokus pada ketidakpuasan pengguna[12].

2.6 Text Processing

Text processing adalah suatu proses untuk menyeleksi data text agar menjadi lebih terstruktur lagi dengan melalui serangkaian tahapan meliputi tahapan *case folding, cleaning, tokenizing, normalization, stop removal,* dan *stemming.*

a) Case Cleaning
Case cleaning merupakan tahapan yang bertujuan untuk menghilangkan kata yang tidak diperlukan, seperti menghapus tanda baca dan angka [13]

b) Case Folding
Case folding merupakan tahapan yang bertujuan untuk menjadikan semua teks/kalimat menjadi huruf kecil semua [14].

c) Tokenizing
Tokenizing merupakan tahapan pemotongan urutan karakter dan sebuah set dokumen berdasarkan tiap kata yang menyusunnya sehingga kalimat menjadi kata sesuai kebutuhan *system*.

d) Normalization
Normalization adalah tahap mengubah kata yang tidak sesuai ejaan ke dalam bentuk sebenarnya atau kata baku.

e) Stopword Removal
Stopword removal merupakan tahap menghapus kata yang tidak diperlukan atau tidak penting atau mempunyai ketergantungan pada suatu topik, contoh: kata konjungsi, kata preposisi, dan kata artikel [15].

f) Stemming
Stemming merupakan suatu proses mengubah kata ke kata dasar [15].

2.7 Word Embeddings Word2vec

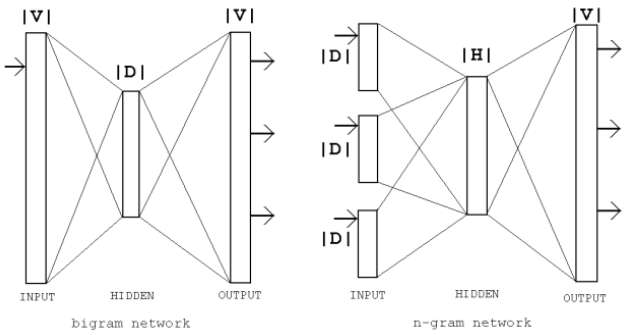
Word2vec adalah suatu metode untuk mempresentasikan setiap kata di dalam konteks sebagai *vector* dengan *N* dimensi. *word2vec* merupakan nama *word embedding* yang dibuat oleh google. Ada 2 jenis arsitektur *neural network* dari *word2vec* yaitu *skip-gram* dan *continous bag of word* (CBOW) [16].

Penelitian ini menggunakan *word2vec* untuk membentuk *word embeddings* dari kata-kata yang ada pada korpus. Karena kasus yang ditangani saat ini adalah topik spesifik, yaitu mengenai ujaran kebencian pada twitter, maka menggunakan *pre-trained word embeddings* yang dilatih menggunakan korpus Bahasa baku, seperti artikel Wikipedia, akan memiliki hasil yang kurang signifikan. Selain itu, dari segi efisiensi juga rendah, karena akan banyak kata-kata yang tidak pernah terlihat pada saat diimplementasikan pada dataset ujaran kebencian ini. Hal ini disebabkan Bahasa yang digunakan dalam cuitan twitter lebih banyak menggunakan Bahasa non formal, dan lebih lagi pada kasus ujaran kebencian, akan banyak Bahasa-bahasa yang tidak layak dan mungkin tidak pernah terlihat pada artikel berita seperti Wikipedia.

Dataset yang digunakan untuk pembentukan *word embeddings* adalah kata-kata yang ada pada data *training*. Tahap *preprocessing* yang sama sebagaimana diterangkan pada bagian 2.6 dilakukan terhadap setiap tweet. Kemudian diproses menggunakan *library gensim*[1] *word2vec* dalam bahasa pemrograman *python*.

Dari seluruh tweet pada data *training*, dikumpulkan kata-kata unik hasil *text* preprocessing yang disebut dengan *Bag of Words*. Input *word2vec* kemudian diubah menjadi bentuk *one-hot encoded vector* dengan dimensi sebanyak jumlah kata unik tersebut.

Kemudian input kata-kata yang sudah menjadi vektor dengan elemen 1 dan 0 ini, dilatih dalam arsitektur *word2vec* untuk membentuk vektor kata-kata (*word embeddings*) dengan dimensi yang jauh lebih kecil daripada dimensi *Bag of Words*. Arsitektur *word2vec* berbentuk jaringan syaraf tiruan, yang berbasis *log-linear model* (*neural network log-linear model* NNLM) [16], yang dikembangkan dari model jaringan n-gram [17] sebagaimana gambar Gambar 1.



***Gambar 1** Arsitektur dasar word2vec [17]*

Dari model tersebut, dikembangkan model dengan bentuk *Continous Bag of Words* (CBOW), dan *skip-gram* (seperti pada Gambar 2). Penelitian ini menggunakan model skip-gram untuk menghasilkan word embeddings.

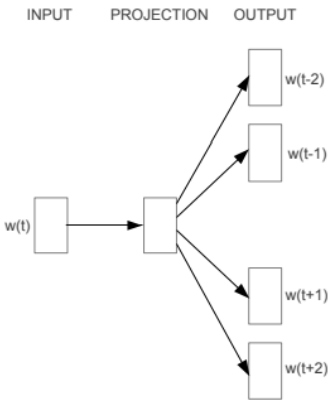
Tujuan dari skip-gram adalah untuk memprediksi konteks atau kata-kata (*output*) di sekitar *current word (input)*. Dari diagram pada gambar 1, misalnya kata-kata yang dievaluasi adalah "Pertemuan G20 dihadiri pemimpin dunia". Maka ketika *w(t)* berada di posisi "pertemuan", maka output *w(t+1)* dan *w(t+2)* akan memprediksi kata-kata "G20" dan "dihadiri". Token sebelumnya, yaitu *w(t-1)* dapat berupa tag "<s>" yang berarti permulaan kalimat. Sedangkan *w(t-2)* dapat berisi token kosong.

Pada saat *w(t)* mengevaluasi kata "G-20", maka output yang diprediksi setelahnya adalah *w(t+1)* dan *w(t+2)* yang secara berturut-turut adalah "dihadiri" dan "pemimpin". Sedangkan token sebelumnya yang

[1] https://radimrehurek.com/gensim/models/word2vec.html

ISSN No. 2502-8782

diprediksi adalah *w(t-2)* dan *w(t-1)* secara berturut-turut adalah token "<s>" dan "pertemuan".



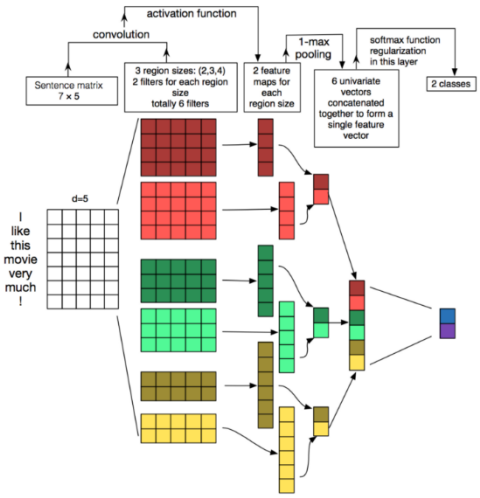
Gambar 2 *Arsitektur Word2vec Skip-gram* [15]

2.8 Algoritma CNN (Convolutional Neural Network)

Convolution Neural Networks (CNN) adalah jaringan saraf tiruan berbasis konvolusi, yaitu perkalian *element wise* pada matriks input sesuai pergerakan *window* matriks filternya. CNN mampu mendeteksi fitur kompleks dalam data, misalnya, mengekstraksi fitur dalam data gambar dan teks. CNN pada awalnya dikembangkan untuk tugas-tugas *image processing* seperti klasifikasi gambar, deteksi objek, dan segmentasi gambar, dan dapat memberikan hasil *state-of-the-art* yang jauh melampaui teknik konvensional dan *machine learning* biasa. CNN juga diuji pada masalah teks, dan dapat bekerja cukup baik pada tugas klasifikasi, namun masih belum dapat memecahkan masalah lain yang lebih kompleks seperti *text generation*, yang menjadi dasar tugas peringkasan dokumen secara abstraktif, mesin penerjemah, sistem *question answering* dan lainnya.

CNN terdiri dari dua bagian utama, yaitu layer konvolusi untuk mendapatkan fitur dari data, dan layer *pooling* untuk mengurangi ukuran dimensi fitur. Layer konvolusi bertugas untuk mendeteksi fitur yang penting dari data. Outputnya berupa dapat *feature map***,** pada gambar misalnya berbentuk hasil deteksi tepi pada bagian wajah, atau bagian tubuh, Gedung, pesawat, alat transportasi, bunga dan daun dan lainnya. Fitur map dihasilkan dari perkalian *element wise* antara matriks filter yang ukurannya lebih kecil, yang dijalankan di seluruh posisi elemen matriks inputnya. Pada gambar, matriks input adalah bit-bit yang menyusun gambar dalam koordinat x dan y. Dalam pemrosesan *image*, pendeteksi fitur sering juga disebut sebagai kernel atau filter.

Pada pemrosesan teks, inputnya adalah kalimat atau dokumen yang direpresentasikan sebagai matriks. Setiap baris pada matriks merupakan token (kata atau sejumlah karakter *n-gram*) yang direpresentasikan dalam bentuk vektor. Seperti ilustrasi pada Gambar 3, kalimat yang akan dijadikan input adalah "I like this movie very much !". Kalimat ini dijadikan matriks dengan ukuran 7x5, yaitu 7 kata dengan vektor kata berdimensi 5.



Gambar 3 *Ilustrasi arsitektur CNN untuk klasifikasi kalimat* [2]

Input ini akan dilewatkan pada lapisan konvolusi yang terdiri atas 3 pasang filter, yaitu masing-masing berukuran 2, 3 dan 4 kata. Proses konvolusi pada setiap filter di sepanjang matriks kalimat, menghasilkan *featuremaps* untuk setiap area. *Max pooling* memilih 1 nilai dari setiap *featuremaps*, dan disatukan dalam bentuk vektor *output* dari layer konvolusi CNN. Vektor fitur ini kemudian dikalkulasi oleh sebuah *neural network* (*dense layer*) sederhana untuk menghasilkan kelas klasifikasi.

3 METODOLOGI PENELITIAN

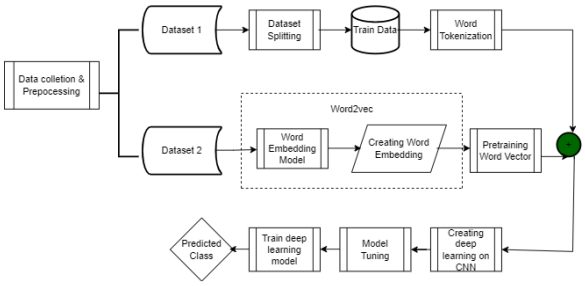
Penelitian ini mendeteksi 2 kelas dari setiap tweet, yaitu apakah suatu tweet mengandung Hate Speech atau ujaran kebencian, dan apakah di dalamnya juga terdapat Bahasa kasar atau *Abusive Language*. Tidak selalu ujaran kebencian berisi Bahasa kasar, dan tidak pula setiap tweet yang mengandung Bahasa kasar mengandung ujaran kebencian yang ditujukan kepada seseorang. Oleh karena itu, antara kedua kelas ini saling berdiri sendiri dan tidak saling mempengaruhi. Oleh sebab itu, proses klasifikasi dilakukan secara terpisah, dengan model CNN yang terpisah pula. Proses klasifikasi dilakukan melalui tahapan seperti dapat dilihat pada gambar 4.

3.1 Data Collection

Penelitian ini menggunakan dataset dari Ibrahim dan Budi [18] yang dikumpulkan dari twitter

[2] https://medium.com/jatana/report-on-text-classification-using-cnn-rnn-han-f0e887214d5f

menggunakan teknik *crawling*. Data tersebut diberikan label oleh 30 orang annotator. Terkumpul sebanyak 13.126 data tweet yang valid, dengan kelas yang digunakan dalam penelitian ini adalah kelas *Hate Speec*h dan *Abusive*, dengan statistik sebagaimana terlihat pada Tabel 1 berikut.



Gambar 4 *Blok Diagram Tahapan Klasifikasi Tweet*

3.2 Data Collection

Penelitian ini menggunakan dataset dari Ibrahim dan Budi [18] yang dikumpulkan dari twitter menggunakan teknik *crawling*. Data tersebut diberikan label oleh 30 orang annotator. Terkumpul sebanyak 13.126 data tweet yang valid, dengan kelas yang digunakan dalam penelitian ini adalah kelas *Hate Speech* dan *Abusive*, dengan statistik sebagaimana terlihat pada Tabel 1 berikut.

Tabel 1 *Tabel Statistika Dataset*

No	Kelas	Label	Jumlah
1	Hate Speech	HS	5.552
		Non Hs	7.574
2	Abusive	Absive	5.034
		Non Abusive	8.092

Text preprocessing sebagaimana dijelaskan pada bagian 2.6 diterapkan di sini, untuk menghasilkan token kata yang sama antara data yang dipakai untuk klasifikasi dan data yang dipakai untuk pembentukan *word embeddings*.

Dalam penelitian ini, dataset digandakan menjadi dataset 1 yang digunakan untuk klasifikasi, dan dataset 2 yang digunakan untuk membuat *word embeddings* dengan word2vec. Porsi data *training* yang sama yang digunakan pada dataset 1, digunakan pula untuk membangkitkan *word embeddings* pada dataset 2. Porsi data testing tidak boleh terlihat selama *training* model, baik model klasifikasi CNN maupun model *word embeddings word2vec*.

3.3 Data Splitting

Blok *data splitting* memisahkan data set menjadi *data training* dan *data testing*, dengan komposisi 90:10. Data *training* atau data latih digunakan untuk melatih mesin dan mengembangkan model dari sistem *machine learning*. Proses *training* dilakukan untuk mengestimasi bobot atau parameter-parameter pada metode *machine learning* yang digunakan. Set parameter dan bobot ini menjadi model sebuah *machine learning* untuk digunakan dalam mengklasifikasi data testing.

3.4 Tokenisasi Kata (*Word Tokenization*)

Tokenisasi adalah proses pemisahan kata-kata dari kalimat tweet, dengan pemisah adalah spasi dan tanda baca (titik, koma, tanda seru, tanda tanya). Selanjutnya, setiap kata-kata akan direpresentasikan dengan vektor *word embeddings* yang telah dihasilkan dari model *word2vec*.

3.5 Word2vec

Blok *Word2vec* melakukan *training* terhadap data set untuk menghasilkan *word embeddings* yang digunakan untuk merepresentasikan kata dalam bentuk vektor dengan ukuran dimensi tertentu. *Language model* yang dihasilkan jauh lebih kecil dimensinya dan seringkali lebih baik daripada menggunakan model *Bag of Words* yang *sparsity*-nya sangat tinggi (banyak komponen vektor yang bernilai 0).

Daftar vektor kata (blok *pretrained word vector*) yang dihasilkan oleh *word2vec* disimpan dalam suatu model bahasa (*language model*) dengan kosa kata sebanyak 13000 kata, dengan dimensi vektor sebesar 100. Model ini akan mengeluarkan vektor kata sesuai dengan yang diminta oleh blok *word tokenization* pada Gambar 5.

3.6 Deep Learning CNN

Blok CNN memproses input kalimat tweet yang sudah diubah ke dalam bentuk matriks vektor kata, dengan ukuran maksimum 35 kata dikali 100 ukuran vektor setiap kata. Apabila kata-kata pada tweet tidak mencapai 35 kata, maka diberikan padding 0 di depannya, sehingga ukuran matriks input CNN tetap fix pada 35x100.

Selanjutnya dibentuk 3 filter, dengan ukuran dimensi 2x100 (filter *bigram*), 3x100 (*trigram*) dan 4x100 (*fourgram*). Dari masing-masing filter, hasil konvolusi *elemen wise* terhadap matriks input akan menghasilkan beberapa nilai dengan fungsi aktifasi *relu*, sesuai dengan ukuran filternya. Diambil nilai tertinggi dengan *maxpooling* pada masing-masing filter. Ketiga filter ini dijalankan di sepanjang kata, mulai dari kata pertama sampai kata terakhir. Akan dihasilkan sebanyak 34 nilai dari filter *bigram*, 33 nilai dari filter *trigram*, dan 31 nilai dari filter *fourgram*.

Output dari ketiga filter digabungkan dalam satu vektor (*flattening*), dan kemudian diproses ke dalam sebuah *multilayer perceptron* (*neural network* konvensional) dengan komposisi *hidden layer* berdimensi 256 *node*. Lalu outputnya dihitung dengan fungsi aktifasi *sigmoid* menghasilkan satu nilai. Bila nilai tersebut lebih besar atau sama dengan 0.5, maka kelasnya adalah 1 (kelas HS atau AB), dan kelasnya 0 selainnya (tidak mengandung HS atau AB).

Parameter tuning dilakukan untuk mengoptimasi hasil klasifikasi. Proses training

dilakukan berulang-ulang sampai diperoleh hasil akurasi yang paling optimal. Apabila hasil kembali memburuk, maka model yang dipilih adalah model optimal yang dicapai sebelumnya.

Model terpilih, selanjutnya akan digunakan dalam memprediksi data testing yang belum pernah terlihat sebelumnya selama proses training.

3.6 *Set Up* Eksperimen

Untuk kebutuhan klasifikasi pada penelitian ini, arsitektur CNN yang digunakan adalah seperti dideskripsikan pada gambar 4, dengan jumlah layer CNN sebagaimana diterangkan dalam bagian 3.5. Seleksi fitur biasa digunakan untuk mengkombinasikan proses *text preprocessing* pada kalimat menjadi vektor yang nantinya akan digunakan untuk fitur dasar.

Seleksi fitur untuk membentuk kata-kata input, dilakukan mengikuti hasil-hasil terbaik yang telah dicapai di dalam [5, 6, 19, 20, 21, 22], yaitu komposisi penerapan *stop word removal* (STW), *case folding* (CF), dan *punctuation removal* (PCT) sebagaimana Tabel 2 berikut. Adapun komposisi lainnya yang bersifat spesifik dari metode masing-masing, tidak diujikan dalam penelitian ini.

Tabel 2 *Seleksi fitur yang digunakan*

Exp ID	STW	CF	PCT
Exp1	Ya	Ya	Tidak
Exp2	Ya	Tidak	Tidak
Exp3	Ya	Ya	Ya
Exp4	Tidak	Ya	Tidak

3.7 Riset terkait

Dataset yang sama juga diteliti pada [5, 6, 19, 20, 21, 22], dengan mengembangkan metode *machine learning* tertentu dan mengoptimasinya. Metode *random forrest* pada [19] menggunakan komposisi fitur *selection* terbaik dengan *word embeddings FastText*, dan pada parameter optimalnya mendapatkan hasil akurasi sebesar 77.3% dan 80.3% untuk kelas HS dan AB.

Decision Tree [5] mendapatkan akurasi terbaik berbagai variasi fitur *text preprocessing* untuk membentuk *word embeddings FastText*, dan tambahan fitur khusus, untuk kelas HS dan AB adalah 71.29% dan 77.99%. Metode *k-nearest neighbour* (K-NN) pada [20] dan menggunakan *feature engineering* akurasi HS dan AB sebesar 79.13% dan 83.54%.

Metode *logistic regression* pada [6] melakukan beberapa kali percobaan pada input vektor *word embeddings FastText.* Akurasi yang diperoleh untuk kelas HS adalah 75,86%, dan kelas AB 80,05%, dengan komposisi train:test adalah 90:10. Sedangkan *Naïve Bayes* di dalam [21] yang hanya bekerja berdasarkan probabilitas kata-kata di dalam *Bag of Words*, berhasil mendapatkan akurasi yang cukup tinggi sebesar 82.99% dan 86.48% untuk kelas HS dan AB.

Satu-satunya metode klasifikasi berdasarkan *deep learning*, yaitu *transfer learning* BERT pada [22] menghasilkan akurasi yang sangat tinggi untuk kelas HS dan AB, yaitu 88.91% dan 93.24%. Hasil ini membuktikan bahwa saat ini metode *deep learning* sudah dapat melampaui metode *machine learning* konvensional menjadi metode *state-of-the-art* di bidang klasifikasi teks.

3.8 Evaluation Metric

Evaluation metric atau metrik/ukuran evaluasi yang digunakan dalam penelitian ini adalah akurasi, yaitu seberapa banyak data uji yang berhasil diklasifikasikan benar, dibagi jumlah data. Akurasi dapat dihitung melalui sebuah tabel *confusion matriks* seperti pada Tabel 3. Ia menyajikan jumlah data-data yang benar terklasifikasi sesuai kelasnya (TP dan TN), jumlah data yang seharusnya terklasifikasi benar (positif) namun tidak dapat ditemukan (FP), dan jumlah data yang salah (*negative*) namun terklasifikasi benar (FN). Akurasi dihitung menurut persamaan (1) dalam persen.

Tabel 3 *Confusion Matrix*

Kelas	Terklasifikasi positif	Terklasifikasi negatif
Positif	TP (True Positif)	FP (False Positif)
Negatif	FN (False Negatif)	TN (True Negatif)

$$\text{Akurasi} : \frac{(TP+TN)}{TP+TN+FN+FP} x\ 100 \qquad (1)$$

4 HASIL DAN PEMBAHASAN

4.1 Optimasi Model

Sebelum penerapan CNN, penelitian ini melakukan training terlebih dahulu untuk membangkitkan *word embedding* dari kata-kata, yaitu *word2vec* sebagaimana dijelaskan pada bagian 3.4. Kemudian input tweet diubah bentuknya ke dalam representasi vektor *word embeddings* dengan dimensi 35x100 sebagaimana diterangkan pada bagian 3.5. Input data training ini diproses oleh jaringan CNN dengan komposisi 90% untuk training dan 10% untuk validasi. Proses optimasi training menggunakan akurasi sebagai metrik pengukuran validasinya, optimizer Adam, dan loss validasi menggunakan *binary cross entropy* untuk mendapatkan model. Training terpisah antara kelas HS dan AB dilakukan untuk mendapatkan model terbaik masing-masing.

4.2 Evaluasi Model

Evaluasi model bertujuan untuk menghitung akurasi dari klasifikasi pada data testing menggunakan metode *convolutional neural network* (CNN) dari

ISSN No. 2502-8782

model yang paling optimal. Dari komposisi seleksi fitur yang dilakukan sebagaimana terlihat pada Tabel 4, pada eksperimen ini didapatkan hasil yang sangat baik, yaitu di atas 80% untuk kelas HS, dan di atas 90% untuk kelas AB.

***Tabel 4** Percobaan dengan seleksi fitur*

Fitur Selection			Akurasi (%)		Rata-rata (%)
CF	**STW**	**PCT**	**HS**	**AB**	
Ya	Tidak	Ya	**84,92**	**91,47**	88,19
Tidak	Tidak	Ya	83,93	90,25	87,09

4.2 Analisa

Hasil terbaik dari metode CNN dalam peneltian ini adalah bila menggunakan komposisi fitur selection CF dan PCT, yaitu menerapkan *case folding* dan penghapusan tanda baca (*punctuation*). Sedangkan *stopword removal* dan lainnya tidak dilakukan.

Hasil ini bila dibandingkan dengan hasil-hasil penelitian yang menggunakan dataset yang sama, menunjukkan bahwa metode *deep learning* dapat melampaui metode *machine learning conventional* yang ada. Satu-satunya hasil yang lebih tinggi dari hasil yang dicapai pada penelitian ini adalah menggunakan metode *transfer learning* (*transformer model*) dengan BERT[3], yang juga merupakan metode berbasis *deep learning*, sebagaimana ditunjukkan pada Tabel 5.

Tabel 5 Perbandingan Hasil Klasifikasi

Metode	Akurasi	
	Kelas HS (%)	**Kelas AB (%)**
Decision Tree [5]	71,29	77,99
KNN [20]	79,13	83,54
Random Forest [19]	77,29	80,30
Logistic Regression [6]	75,86	80,05
Naïve Bayes [21]	82,99	86,48
BERT [22]	**88,91**	**93,24**
CNN (penelitian ini)	84,92	91,47

Hasil yang dicapai CNN sudah cukup baik bila dibandingkan dengan BERT [22], mengingat metode ini hanya menggunakan *word embeddings* yang dilatih dari data input yang terbatas, yaitu sejumlah data tweet pada data training. Sedangkan metode BERT, menggunakan *word embedding* yang dilatih dari korpus yang sangat besar yang berasal dari koleksi dokumen artikel berita, media sosial, blog dan website, di mana semua jenis kalimat percakapan, kalimat berita, sampai kepada ungkapan-ungkapan bermuatan emosi sangat banyak terdapat di sana.

5 SIMPULAN

Metode Convolutional Neural Network yang dikembangkan dalam penelitian ini, telah berhasil mencapai akurasi klasifikasi kelas Hate Speech dan Abusive Language dengan hasil yang sangat baik, bila dibandingkan dengan metode machine learning konvensional, yaitu berturut-turut adalah sebesar 84.92% dan 91.47%.

Namun demikian, metode CNN masih belum dapat melampaui hasil yang diperoleh dari metode transformer BERT. Walaupun begitu, hasil ini tetap menggembirakan karena dengan *data training* yang berukuran kecil, proses *training deep learning* untuk mendapatkan model optimal cukup cepat, dan dapat menghasilkan akurasi yang tinggi.

KEPUSTAKAAN

[1] Nasrullah, Rulli, *Media Sosial: perspektif komunikasi, budaya, dan sositeknologi*, ISBN 978-602-7973-25-1. Bandung, 2015

[2] L. P. A. S. Tjahyanti, Pendeteksian Bahasa Kasar (Abusive Language) dan Ujaran Kebencian (Hate Speech) dari Komentar Di Jejaring Sosial. *Journal of Chemical Information and Modeling*," *Jurnal Pendidikan*, vol. 07 No. 01, pp. 1-14, 2020.

[3] D. T. McGonagle, "The Council of Europe against online hate speech:Conundrums and challenges," pp. 1-40, 2013.

[4] W. N. Dewani, P. S. A. and Y. Azhar, "Klasifikasi Multi-label Ujaran Kasar dan Kebencian (Hate Speech & Abusive) Pada Media Sosial Twitter di Indonesia," pp. 1-11.

[5] F. Ihsan, I. Iskandar, N. S. Harahap and S. Agustian, "Algoritme decision tree untuk mendeteksi ujaran kebencian dan bahasa kasar," *Jurnal Teknologi dan Sistem Komputer*, pp. 1-6, 2021.

[6] A. Fransiska, S. Agustian, F. Insani, M. Fikry and Pizaini, "Algoritme Logistic Regression untuk Mendeteksi Ujaran Kebencian dan Bahasa Kasar Multilabel pada Twitter Berbahasa Indonesia," *Jurnal Nasional Komputasi dan Teknologi Informasi*, vol. V, pp. 1-5, 2022.

[7] H., Setiawandari, H.I., Mundandar, Hate Speech In Election 2019: Case Study Of Youth Organizations, J. Strateg. Glob., vol. 4 No.1, 2021.

[8] B. J, "Regulating hate speech online," *Int. Rev. Law, Comput. Technol*, vol. 24 No.3, pp. 233-239, 2010.

[3] https://huggingface.co/docs/transformers/model_doc/bert

[9] A. F. Hidayatullah, A. A. F. Yusuf, K. P. Juwairi and R. A. N. Nayoan, "Identifikasi Konten Kasar pada Tweet Bahasa Indonesia," *Jurnal Linguistik Komputasional,* vol. 2 No.1, pp. 1-5, 2019.

[10] M. Hadi, Twitter Untuk orang Awam, Palembang, 2010.

[11] W. E. Bebas, "Klasifikasi," Wikipedia project, 6 September 2022. [Online]. Available: https://id.wikipedia.org/wiki/Klasifikasi.

[12] I. Gamayanto, F. E. Nilawati and Suharnawi, "Pengembangan dan Implementasi dari Wise Netizen (EComment) di Indonesia," *Tecno.com,* vol. 16 No.1, pp. 1-16, 2017.

[13] E. Retnawiyati, Fatoni and E. S. Negara, "Analisis Sentimen Pada Data Twitter dengan Menggunakan Text Mining terhadap Suatu Produk," 2015.

[14] A. C. Pradikdo and A. Ristyawan, "Model Klasifikasi Abstrak Skripsi Menggunakan Text Mining untuk Pengkategorian Skripsi Sesuai Bidang Kajian," *Simetris,* vol. 8 No.2, pp. 1-8, 2018.

[15] A. Ayedh, G. TAN, K. Alwesabi and H. Rajeh, "The Effect of Preprocessing on ArabicDocument Categorization," *Algorithms,* pp. 1-17, 2016.

[16] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *1st Int. Conf. Learn. Represent. ICLR 2013 - Work. Track Proc.*, pp. 1–12, 2013.

[17] T. Mikolov, J. Kopecky, L. Burget, O. Glembek and J.H. Cernocky, "Neural Network Based Language Models for Highly Inflective Languages", in Proc.: *ICASSP* 2009.

[18] M. O. Ibrohim and I. Budi, "Multi-label Hate Speech and Abusive Language Detection in Indonesian Twitter," *Proceedings of the Third Workshop on Abusive Language Online,* pp. 46-57, 2019.

[19] A. Amri, "Implementasi Algoritma *Random Forest* Untuk Mendeteksi *Hate Speech* Dan *Abusive Language* Pada Twitter Bahasa Indonesia", thesis report, *UIN Suska Riau*, 2020

[20] A. Fadhilah, "Penerapan Algoritma K-Nearest Neighbor untuk Mendeteksi Ujaran Kebencian dan Bahasa Kasar Pada Twitter Bahasa Indonesia", thesis report, *UIN Suska Riau*, 2021

[21] T. Ghassani, "Klasifikasi Hate Speech dan Abusive Language pada Twitter Bahasa Indonesia dengan Metode Naive Bayes Classifier", thesis report, *UIN Suska Riau*, 2021

[22] R. Saputra, "Implementasi Bidirectional Encoder Representations From Transformers (BERT) untuk Mendeteksi Hatespeech dan Abusive Language pada Twitter Bahasa Indonesia", thesis report, *UIN Suska Riau*, 2022

ISSN No. 2502-8782