

ANALISIS SENTIMEN APLIKASI MICROSOFT TEAMS BERDASARKAN ULASAN GOOGLE PLAY STORE MENGGUNAKAN MODEL NEURAL NETWORK DENGAN OPTIMASI ADAPTIVE MOMENT ESTIMATION (ADAM)

Nefa Andriani^{1*}, Budi Warsito², Rukun Santoso³

^{1,2,3} Departemen Statistika, Fakultas Sains dan Matematika, Universitas Diponegoro

*e-mail: nevaandriani0404@gmail.com

DOI: 10.14710/j.gauss.13.1.168-179

Article Info:

Received: 2023-05-18

Accepted: 2024-10-09

Available Online: 2024-10-10

Keywords:

Microsoft Teams; Google Play Store; Sentiment Analysis; Neural Network; Adaptive Moment Estimation

Abstract: The *Covid-19* virus pandemic, with its rapid spread, has made the government enact several policies that can reduce community activities that can potentially cause crowds, including social distancing, work-from-home (WFH), and online or hybrid learning. One of the platforms that were widely used during the WFH and online learning period was Microsoft Teams. Microsoft Teams is an app that lets people and organizations collaborate and interact online. Microsoft Teams can be downloaded through the Google Play Store, and users can leave reviews. Sentiment analysis is used to classify user sentiment towards Microsoft Teams into positive and negative sentiments. Review data obtained from January to November 2022. The classification is carried out using the *Neural Network*, a computational system inspired by biological neural networks. Weight optimization is done using *Adaptive Moment Estimation*, which is popular in deep learning because it quickly achieves good results. The classification model in this paper was built with a training test ratio of 80:20. The best hyperparameter combination is node hidden layer 10, learning rate 0,01, and batch size 64. The accuracy of the ANN classification model was obtained by 87%.

1. PENDAHULUAN

Pandemi virus *Covid-19* berdampak signifikan terhadap kehidupan masyarakat di seluruh dunia tak terkecuali Indonesia. Pemerintah berupaya mengurangi penyebaran virus *Covid-19* dengan memberlakukan beberapa kebijakan yang dapat mengurangi kegiatan masyarakat yang berpotensi menimbulkan kerumunan diantaranya *social distancing*, *work from home* (WFH) dan pembelajaran daring maupun *hybrid*. Microsoft Teams merupakan salah satu platform yang banyak digunakan selama masa WFH dan pembelajaran daring. Microsoft Teams merupakan aplikasi yang memungkinkan orang-orang dan organisasi berkolaborasi maupun berinteraksi secara *online*. Pengguna aplikasi Microsoft Teams berdasarkan Statista.com (2022) berjumlah 20 juta pengguna selama tahun 2019 dan meningkat menjadi 270 juta pengguna aktif pada tahun 2022. Microsoft Teams dapat diunduh melalui Google Play Store. Selain mengunduh pengguna juga dapat memberikan *rating* dan *review*/ulasan. Berdasarkan pernyataan sebelumnya, maka penulis ingin meneliti respon pengguna terhadap performa aplikasi Microsoft Teams di Google Play Store.

Analisis sentimen digunakan dalam penelitian ini untuk mengetahui respon pengguna terhadap aplikasi Microsoft Teams. Sentimen akan diklasifikasikan menjadi sentimen positif dan sentimen negatif. Model *Artificial Neural Network* (ANN) digunakan untuk proses klasifikasi yang merupakan model pemrosesan data berdasarkan cara kerja sistem saraf biologis. ANN merupakan salah satu metode klasifikasi yang populer. ANN banyak digunakan pada masalah visi komputer untuk pengenalan pola dan memiliki kinerja yang baik (Patra dan Singh, 2013). Metode optimasi yang umumnya digunakan pada model ANN adalah *Stochastic Gradient Descent* (SGD). SGD mempunyai performa yang baik dalam

berbagai aplikasi. Namun, penggunaan optimasi SGD dapat merugikan kinerja ANN apabila skala gradien kecil karena SGD menetapkan *learning rate* secara seragam ke segala arah (Wang *et al.*, 2018). Untuk memperbaiki kekurangan SGD, maka diusulkan metode adaptif yang secara diagonal menetapkan *learning rate* melalui perkiraan kelengkungan fungsi. Metode yang dimaksud adalah *Adaptive Moment Estimation* atau Adam (Kingman dan Ba, 2015).

2. TINJAUAN PUSTAKA

Microsoft Teams merupakan platform digital untuk sebuah tim. Platform ini menyatukan konten, percakapan, dan aplikasi dalam satu tempat. Pengguna Microsoft Teams dapat berkomunikasi secara efektif melalui obrolan, percakapan, panggilan, rapat *online* dan konferensi web. Microsoft Teams diciptakan untuk kemudahan dan fleksibilitas dalam berkomunikasi serta berkolaborasi. Microsoft Teams dapat diunduh melalui Google Play Store. Google Play Store merupakan toko digital yang menyediakan layanan produk *online* milik Google. Google Play Store memiliki beragam fitur salah satunya yaitu fitur ulasan atau *review*. Melalui fitur tersebut pengguna dapat memberikan ulasan terhadap layanan yang tersedia dalam Google Play Store (Herlinawati *et al.*, 2020).

Ulasan pada Google Play Store diekstraksi menggunakan teknik *web scraping*. Tujuan *web scraping* adalah untuk mengekstrak informasi dari satu atau banyak situs web menjadi data terstruktur sederhana yang dapat disimpan dan dianalisis dalam bentuk *spreadsheet*, *database* atau CSV (Diouf *et al.*, 2019).

Analisis sentimen atau *opinion mining* mengacu pada bidang yang luas dari pengolahan bahasa alami, *text mining* dan komputasi linguistik yang memiliki tujuan menganalisa sikap, sentimen, pendapat, penilaian, evaluasi dan emosi seseorang pembicara atau penulis berkenaan dengan suatu topik, layanan, produk, individu, organisasi, ataupun kegiatan tertentu (Liu, 2012). Analisis sentimen bertujuan untuk mengelompokkan teks yang terdapat dalam dokumen merupakan opini positif atau negatif. *Text mining* berguna untuk mengetahui lingkup atau topik permasalahan dalam suatu teks (Manning *et al.*, 2009). Data pada *text mining* memiliki format yang tidak terstruktur sehingga harus dilakukan *pre-processing* terlebih dahulu agar dapat digunakan pada tahap selanjutnya menggunakan *text pre-processing*. Tahap *text pre-processing* terdiri dari *case folding*, *remove number*, *remove punctuation*, *remove emoticon*, *remove whitespace*, normalisasi kata, dan *tokenizing*. *Sentiment scoring* merupakan proses pemberian skor pada setiap kata yang terdapat dalam sebuah dokumen (Handayanto *et al.*, 2021). *Sentiment scoring* bekerja dengan cara melakukan klasifikasi polaritas kalimat ke dalam kelompok positif dan negatif. Dalam pemberian skor sentimen diperlukan kamus sentimen, *boosterwords*, dan kata negasi (Wahid dan Azhari, 2016). *Stopword* merupakan kumpulan kata yang sering muncul dalam suatu dokumen tetapi tidak memiliki arti deskriptif sehingga dapat dihilangkan. *Stemming* adalah proses mengubah kata berimbuhan menjadi kata dasar. *Stemming* didasarkan pada asumsi bahwa istilah-istilah yang mempunyai akar atau kata dasar yang sama biasanya akan mempunyai makna yang sama (Tala, 2003).

Metode TF-IDF merupakan suatu metode pemberian bobot dari hubungan suatu kata (*term*) terhadap dokumen yang banyak digunakan pada bidang *text mining* (Kim dan Gil, 2019). TF-IDF termasuk dalam transformasi data yang merupakan proses mengubah token menjadi vektor numerik. Proses ini bertujuan agar algoritma klasifikasi dapat memproses data karena algoritma klasifikasi tidak dapat memproses dataset *original* (Hamzah, 2021). Metode TF-IDF memberikan nilai bobot untuk setiap kata pada dokumen dengan

menghitung *Term Frequency* dan *Inverse Document Frequency*. Rumus perhitungan *term frequency* dinyatakan pada persamaan (1).

$$TF_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \quad (1)$$

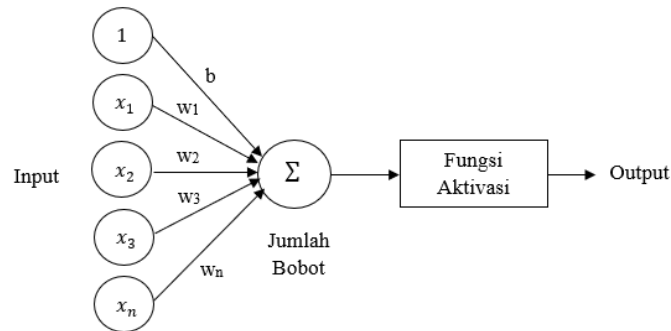
Rumus perhitungan IDF dinyatakan pada persamaan (2).

$$IDF_i = \log_e \left(\frac{N}{df(i)} \right) + 1 \quad (2)$$

Perhitungan TF-IDF didapatkan dari perkalian persamaan (1) dan (2) sehingga TF-IDF dinyatakan sebagai berikut:

$$TFIDF_{i,j} = TF_{i,j} \times IDF_i \quad (3)$$

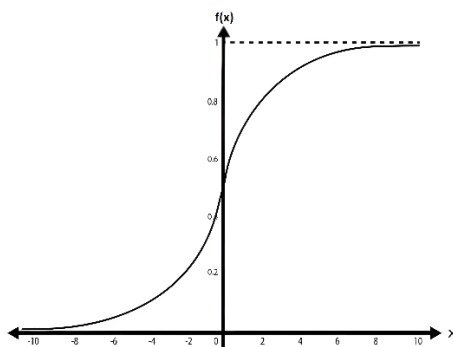
Artificial Neural Network (Jaringan Saraf Tiruan) atau *Neural Network* merupakan algoritma pemodelan yang terinspirasi oleh otak manusia, meniru cara pensinyalan antara neuron biologis (Warsito *et al.* 2021). ANN terdiri dari neuron yang merupakan sejumlah prosesor sederhana dan saling berhubungan. Sinyal melewati neuron satu ke neuron yang lain dengan masing-masing neuron terhubung oleh pembobotan (*weight*). Struktur neuron pada ANN ditunjukkan pada Gambar 1.



Gambar 1. Struktur Neuron *Artificial Neural Network*

Gambar 1 menunjukkan neuron ANN terdiri dari beberapa komponen. Input berperan layaknya dendrit di dalam otak manusia sebagai penerima informasi yang masuk dari neuron lainnya. *Input* berfungsi sebagai masukan yang terdiri dari fitur ($x_1, x_2, x_3, \dots, x_n$) yang akan diproses dalam ANN. Setiap nilai input melewati sebuah hubungan berbobot w ($w_1, w_2, w_3, \dots, w_n$). Bobot berfungsi seperti synopsis pada otak manusia yaitu untuk menunjukkan kekuatan keterkaitan antara satu neuron dengan neuron yang lain. Pada jaringan saraf tiruan, sering kali dimasukkan sebuah unit input yang selalu memiliki nilai satu, yang dikenal sebagai unit bias b . Bias berfungsi untuk mengubah nilai *threshold* menjadi nol. Semua nilai kemudian digabungkan. Gabungan nilai diproses oleh fungsi aktivasi untuk menghasilkan sinyal keluaran dari jaringan yang telah dimasukkan atau bahkan menjadi masukan bagi neuron yang lain.

Fungsi aktivasi merupakan fungsi pada jaringan saraf yang digunakan untuk menetapkan nilai *output* suatu neuron (Siang, 2009). Fungsi aktivasi yang sering digunakan pada ANN adalah sigmoid biner yang memiliki nilai antara 0 sampai 1. Kurva dari fungsi aktivasi sigmoid biner dapat dilihat pada Gambar 2.



Gambar 2. Fungsi Aktivasi Sigmoid Biner

Rumus fungsi sigmoid biner secara sistematis dinyatakan pada persamaan (4).

$$\sigma(x) = f(x) = \frac{1}{(1+e^{-x})}, -\infty < x < \infty \quad (4)$$

Fungsi turunan dari sigmoid biner ditunjukkan pada persamaan (5).

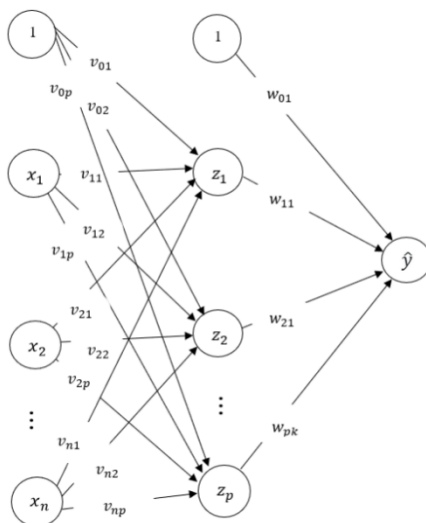
$$\sigma'(x) = f'(x) = f(x)(1 - f(x)) \quad (5)$$

Cross entropy merupakan *loss function* yang berguna untuk mengukur kebaikan performa kinerja model dalam memprediksi target. *Loss function* yang optimal adalah fungsi yang menghasilkan *error* atau kesalahan yang diharapkan paling rendah. Perhitungan *cross entropy* menggunakan nilai y dengan y bernilai 0 atau 1 dan \hat{y} dengan $(0 < \hat{y} < 1)$. *Loss function* dengan *cross entropy* dapat dihitung berdasarkan persamaan (6).

$$E = -(y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})) \quad (6)$$

dengan y nilai sebenarnya/target, \hat{y} nilai prediksi

Backpropagation merupakan salah satu bagian dari *neural network* yang biasa digunakan pada jaringan *multilayer*. Pelatihan *backpropagation* terdiri dari tiga tahapan, yakni *feedforward* (umpan maju), *backpropagation of error* (umpan mundur/propagasi error), serta *adjustment* (modifikasi bobot dan bias) (Fausett, 1994). Arsitektur *backpropagation* ditunjukkan pada Gambar 3.



Gambar 3. Arsitektur ANN *Backpropagation*

Gambar 3 mengilustrasikan arsitektur ANN dengan unit *input* satu sampai n dan unit konstan/bias, satu *hidden layer* dengan unit neuron satu sampai p dan unit konstan/bias, serta satu unit *output*. Model *feedforward* ANN dengan satu *output*, satu *hidden* dan *input* x_1, \dots, x_n ditulis dalam persamaan (7).

$$\hat{y} = f^o\{w_{0k} + \sum_{j=1}^p w_{jk} f^h(v_{0j} + \sum_{i=1}^n v_{ij} x_i)\} \quad (7)$$

dengan \hat{y} vektor nilai-nilai *output* jaringan, w_{0k} bobot antara unit bias dan *output*, w_{jk} bobot koneksi *hidden* dengan *output*, v_{0j} bobot antara unit bias dan *hidden*, v_{ij} bobot koneksi *input* dengan *hidden*, f^o fungsi aktivasi pada unit *output*, f^h fungsi aktivasi pada unit *hidden*, x_i unit *input*. Berdasarkan persamaan (7) maka Model ANN dengan satu *output*, satu *hidden* dan *input* x_1, \dots, x_n menggunakan fungsi aktivasi sigmoid biner dituliskan pada persamaan (8).

$$\hat{y} = \sigma^o\{w_{0k} + \sum_j w_{jk} \sigma^h(v_{0j} + \sum_i v_{ij} x_i)\} \quad (8)$$

Tahapan *FeedForward* satu *hidden layer* dengan fungsi aktivasi sigmoid biner adalah sebagai berikut:

1. Langkah 0 : Inisialisasikan nilai bobot dengan bilangan acak antara -1 sampai 1.
2. Langkah 1 : Selama kondisi berhenti bernilai salah, maka lakukan langkah 2-9.
3. Langkah 2 : Lakukan langkah 3-8 untuk setiap *data training*.
4. Langkah 3 : Setiap unit *input layer* ($x_i, i = 1, 2, 3, \dots, n$) menerima sinyal x_i dan meneruskan sinyal tersebut ke semua unit pada lapisan yang ada di atasnya (*hidden layer*).
5. Langkah 4 : Setiap unit *hidden layer* ($z_j, j = 1, 2, 3, \dots, p$) menjumlahkan sinyal-sinyal *input* terbobot berdasarkan persamaan (9).

$$z_{in j} = v_{0j} + \sum_{i=1}^n \sum_{j=1}^p x_i v_{ij} \quad (9)$$

Aplikasikan fungsi aktivasi untuk mendapatkan sinyal *output* dari unit *hidden* berdasarkan persamaan (10).

$$z_j = f(z_{in j}) = \frac{1}{(1+e^{-z_{in j}})} \quad (10)$$

Sinyal *output* unit *hidden* kemudian diteruskan ke unit *output*.

6. Langkah 5 : Setiap unit *output* $y_k (k = 1, 2, 3, \dots, m)$ menjumlahkan semua sinyal *input* dari unit *hidden* yang sudah terboboti termasuk biasnya.

$$y_{in k} = w_{0k} + \sum_{j=1}^p \sum_{k=1}^m z_j w_{jk} \quad (11)$$

Aplikasikan fungsi aktivasi untuk mendapatkan sinyal *output* dari unit *output* berdasarkan persamaan (12).

$$\hat{y}_k = f(y_{in k}) = \frac{1}{(1+e^{-y_{in k}})} \quad (12)$$

Sinyal *output* tersebut kemudian diteruskan ke seluruh unit *output*.

Tahapan *Backwardpass/Backpropagation* satu *hidden layer* adalah sebagai berikut (Sadowski, 2016):

7. Langkah 6 : Hitung derivatif error terhadap bobot unit *hidden layer* ke unit *output layer* w_{jk} .

$$\frac{\partial E}{\partial w_{jk}} = \frac{\partial E}{\partial \hat{y}_k} \frac{\partial \hat{y}_k}{\partial y_{in k}} \frac{\partial y_{in k}}{\partial w_{jk}} = \frac{\hat{y}_k - y_k}{\hat{y}_k(1-\hat{y}_k)} \hat{y}_k(1-\hat{y}_k) z_j = (\hat{y}_k - y_k) z_j \quad (13)$$

Hitung derivatif error terhadap bobot bias w_{ok} .

$$\frac{\partial E}{\partial w_{ok}} = \frac{\partial E}{\partial \hat{y}_k} \frac{\partial \hat{y}_k}{\partial y_{in k}} \frac{\partial y_{in k}}{\partial w_{ok}} = \frac{\hat{y}_k - y_k}{\hat{y}_k(1-\hat{y}_k)} \hat{y}_k(1-\hat{y}_k)1 = (\hat{y}_k - y_k) \quad (14)$$

8. Langkah 7 : Hitung derivatif error terhadap bobot koneksi unit *input layer* ke unit *hidden layer* v_{ij} .

$$\frac{\partial E}{\partial v_{ij}} = \frac{\partial E}{\partial z_{in j}} \frac{\partial z_{in j}}{\partial v_{ij}} = \sum_{k=1}^n (\hat{y}_k - y_k)(w_{jk})(z_j(1-z_j))x_i \quad (15)$$

Hitung derivatif error terhadap bobot bias v_{oj} .

$$\frac{\partial E}{\partial v_{oj}} = \frac{\partial E}{\partial z_{in j}} \frac{\partial z_{in j}}{\partial v_{oj}} = \sum_{k=1}^n (\hat{y}_k - y_k)(w_{jk})(z_j(1-z_j))1 \quad (16)$$

Tahapan Modifikasi bobot dan bias (*Adjustment*) adalah sebagai berikut:

9. Langkah 8: Hitung perubahan bobot koneksi unit *hidden layer* ($k = 1,2,3, \dots, m$) menuju *output layer* ($j = 1,2,3, \dots, p$) dengan *learning rate* α .

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) - \alpha \frac{\partial E}{\partial w_{jk}} \quad (17)$$

Hitung perubahan bobot koneksi di unit *input layer* menuju unit ($i = 1,2,3, \dots, n$) *hidden layer* ($j = 1,2,3, \dots, p$) dengan *learning rate* α .

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) - \alpha \frac{\partial E}{\partial v_{ij}} \quad (18)$$

10. Langkah 9 : Apabila kondisi telah terpenuhi, maka proses pelatihan berhenti. Namun jika belum terpenuhi maka lakukan langkah 3 sampai 9.

Optimasi adalah algoritma untuk memperbarui bobot pada proses pembelajaran *neural network* yang berfungsi untuk mengurangi jumlah *error* atau selisih antara nilai keluaran jaringan dengan target. Adam (*Adaptive Moment Estimation*) merupakan metode yang hanya memerlukan nilai gradien pada urutan pertama sehingga penerapannya efisien. Fungsi gradien diperoleh dari nilai turunan *loss function*. Algoritma optimasi Adam menghitung *learning rate* secara adaptif untuk setiap parameter bobot dari estimasi momen pertama dan momen kedua dari gradien. Rumus tahapan proses algoritma adam sebagai berikut:

$$t = t + 1 \quad (19)$$

$$g_t = \nabla_{\theta} f_t(\theta_{t-1}) \quad (20)$$

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \quad (21)$$

$$u_t = \beta_2 u_{t-1} + (1 - \beta_2) g_t^2 \quad (22)$$

$$\hat{m}_t = \frac{m_t}{(1-\beta_1^t)} \quad (23)$$

$$\hat{u}_t = \frac{u_t}{(1-\beta_2^t)} \quad (24)$$

$$\theta_t = \theta_{t-1} - \alpha \frac{\hat{m}_t}{\sqrt{\hat{u}_t + \epsilon}} \quad (25)$$

dengan t iterasi, $\nabla_{\theta} f_t(\theta_{t-1})$ derivatif parsial fungsi *error/loss* terhadap bobot θ pada iterasi ke- t , \hat{m}_t bias dari estimator m_t , \hat{u}_t bias dari estimator u_t , θ_t parameter yang akan diperbaiki, α *learning rate*, ϵ epsilon, β_1 dan β_2 *exponential decay of the rate* untuk perkiraan momen pertama dan kedua.

Prediksi tidak dapat mengklasifikasikan dengan benar 100% semua dokumen sehingga perlu dilakukan pengujian kinerja sistem klasifikasi menggunakan *confusion matrix*. *Confusion matrix* digunakan untuk mengukur kinerja klasifikasi dengan perhitungan akurasi. Akurasi adalah persentase dokumen data test diklasifikasikan dengan benar. Tabel 1 menunjukkan tabel *confusion matrix*.

Tabel 1. *Confusion Matrix*

		Kelas Prediksi	
		Positif	Negatif
Kelas Aktual	Positif	TP	FN
	Negatif	FP	TN

Word cloud adalah sebuah metode visualisasi teks dokumen. Metode ini menampilkan kata-kata yang paling sering muncul pada suatu dokumen dalam bentuk grafis dua dimensi. Ukuran kata menunjukkan frekuensi kemunculan kata. Semakin besar ukuran kata, maka kemunculan kata tersebut dalam dokumen semakin banyak (Castella dan Sutton, 2014).

3. METODE PENELITIAN

Data kualitatif merupakan jenis data dalam penelitian ini. Data kualitatif yang dimaksud berupa data ulasan atau *review* penggunaan Microsoft Teams. Data dari penelitian ini bersumber dari data sekunder berupa data ulasan pengguna Microsoft Teams yang tersedia pada aplikasi Google Play Store yang diperoleh dari bulan Januari hingga November 2022. Data diperoleh dengan metode *web scraping* menggunakan *python*. Dari proses *scraping* diperoleh data sebanyak 3714 data. Data ulasan yang diperoleh selanjutnya dibersihkan dari duplikat dan data ulasan yang tidak bermakna sehingga tersisa 3587 data. Penelitian ini terdiri dari dua variabel yaitu variabel bebas dan variabel terikat. Variabel bebas berupa *review* atau ulasan aplikasi Microsoft Teams pada Google Play Store dan variabel terikat berupa sentimen pengguna mengenai aplikasi Microsoft Teams. Analisis data dilakukan dengan menggunakan Google Colaboratory dan Microsoft Excel. Analisis data dilakukan dengan urutan langkah-langkah sebagai berikut:

1. *Scraping data review*
2. *Text pre-processing*
3. *Sentiment scoring*
4. *Stopword removal*
5. *Stemming*
6. Pembobotan TF-IDF.
7. Klasifikasi menggunakan Algoritma *Artificial Neural Network* dengan optimasi Adam.
 - a. Memisah data menjadi dua yaitu data *training* dan data *testing*.
 - b. Membangun model ANN dengan optimasi Adam.
 - c. Mengevaluasi hasil kinerja klasifikasi menggunakan *confusion matrix*.
8. Visualisasi *word cloud*

4. HASIL DAN PEMBAHASAN

Data *review* Microsoft Teams pada *website* Google Play Store diekstraksi menggunakan teknik *web scraping*. Data ulasan yang telah diperoleh selanjutnya disimpan untuk dilakukan pengolahan data. Data ulasan yang diperoleh dari *website* Google Play Store dengan cara *scraping* merupakan data teks berupa format *csv* yang tidak terstruktur sehingga belum sepenuhnya dapat diolah untuk proses klasifikasi. Data tak terstruktur diubah menjadi format

data terstruktur dengan tahap *pre-processing*. Metode *text mining* digunakan pada penelitian ini untuk proses pengolahan data dengan bantuan *library* NLTK yang tersedia pada Google Colab. Berikut merupakan tahapan dari proses *pre-processing*:

1. *Case folding*

Tahap *case folding* akan mengubah seluruh huruf kapital menjadi huruf kecil.

2. *Remove number*

Tahap *remove number* akan menghilangkan karakter angka dalam dokumen. Angka pada sebuah dokumen perlu dihapus karena tidak menunjukkan suatu perasaan atau sentimen.

3. *Remove punctuation*

Tahap *remove punctuation* akan menghapus simbol atau tanda baca pada dokumen karena simbol atau tanda baca dapat mengganggu proses analisis sentimen.

4. *Remove emoticon*

Tahap *remove emoticon* akan menghilangkan karakter emoji pada dokumen.

5. *Remove whitespace*

Tahap *remove whitespace* akan mengganti spasi berturut-turut pada dokumen dengan satu spasi. *Remove whitespace* berguna untuk mempermudah proses pembuatan token dan proses klasifikasi.

6. Normalisasi kata

Tahap normalisasi kata akan mengubah kata non-formal menjadi kata formal sehingga program dapat memahami teks dengan mudah.

7. *Tokenizing*

Tahap *tokenizing* akan memisahkan teks menjadi per kata dengan menggunakan spasi sebagai pemisah.

Data ulasan yang diperoleh tidak mengandung label sehingga perlu dilakukan pelabelan sentimen agar dapat diolah menggunakan metode klasifikasi. *Sentiment scoring* merupakan suatu metode memberikan label pada data. Ketentuan dalam *sentiment scoring* berdasarkan program yang digunakan adalah sebagai berikut:

1. Setiap kata pada dokumen akan diberikan skor sesuai skor kata yang terdapat dalam kamus sentimen dan tidak diberi skor jika kata pada dokumen tidak terdapat dalam kamus sentimen.
2. Setiap kata yang diikuti oleh kata yang terdapat dalam kamus negasi akan mendapatkan nilai yang berlawanan dari nilai yang terdapat dalam kamus sentimen.
3. Kata yang bernilai > 0 berdasarkan kamus sentimen dan sebelum kata tersebut terdapat kata yang termasuk dalam kamus *boosterwords*, maka nilai tersebut akan ditambahkan dengan nilai yang terdapat pada kamus *boosterwords*.
4. Kata yang bernilai < 0 berdasarkan kamus sentimen dan sebelum kata tersebut terdapat kata yang termasuk dalam kamus *boosterwords*, maka nilai tersebut akan dikurangi dengan skor yang terdapat pada kamus *boosterwords*.
5. Setiap dokumen yang tidak memiliki kata dengan skor positif akan diberi skor maksimal positif sebesar 1, jika dokumen tidak memiliki kata dengan skor negatif akan diberi skor maksimal negatif sebesar -1.

Kelas dokumen ditentukan berdasarkan nilai positif tertinggi dan nilai negatif tertinggi yang diperoleh. Jika suatu dokumen memiliki nilai positif tertinggi \geq nilai negatif tertinggi maka dokumen diberi label positif. Sebaliknya, jika nilai positif tertinggi $<$ nilai negatif tertinggi maka dokumen diberi label negatif. Berdasarkan hasil pelabelan dengan *sentiment scoring* diperoleh jumlah dokumen dengan sentimen positif sebanyak 2638 dokumen dan jumlah dokumen dengan sentimen negatif sebanyak 949.

Data ulasan pada penelitian memungkinkan terdapat kata-kata umum yang kerap muncul tetapi tak bermakna atau tak memiliki pengaruh sehingga perlu dihapus. Proses penghapusan kata tidak bermakna disebut *stopword removal*. Proses *stopword removal* dilakukan menggunakan kamus *stopword* Indonesia dan *stopword* manual. *Stemming* merupakan tahapan mengubah kata berimbuhan menjadi kata dasar. Kata yang memiliki akar kata yang sama cenderung mempunyai makna yang sama, adanya proses *stemming* akan meminimumkan indeks kata sehingga memudahkan program dalam melakukan proses klasifikasi analisis sentimen.

Data yang akan melalui tahap klasifikasi analisis sentimen harus berbentuk numerik. Metode pembobotan kata TF-IDF adalah metode mengubah data teks menjadi bilangan numerik yang dapat dihitung dan diolah. Jumlah kata yang terdeteksi sebanyak 1124. Tabel 2 menunjukkan hasil perhitungan bobot kata menggunakan TF-IDF untuk ulasan aplikasi Microsoft Teams.

Tabel 2. Hasil TF-IDF

Sampel ke-	aplikasi	bagus	jaring	Masuk	ribet	Stabil	...	zoom
1	0,6786	0	1,4460	0	0	3,3101	...	0
2	0	0	0	1,3568	0	0	...	0
3	0	0	0	0	0,9383	0	...	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮		⋮
3587	0,6786	0,5808	0	0	0	0	...	0

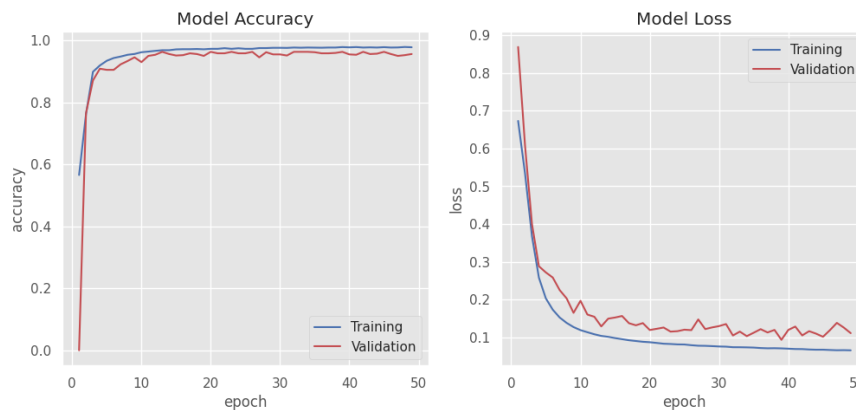
Data dokumen yang telah melalui tahap *pre-processing* dan diketahui nilai pembobotannya dibagi menjadi data *train* dan data *test*. Data *train* adalah data yang digunakan untuk melatih model klasifikasi yang sudah dibentuk. Data *test* adalah data yang digunakan untuk mengukur keberhasilan model dalam melakukan klasifikasi dengan benar. Data *train* harus dipisah dari data *test* untuk mencegah model mengingat data yang digunakan untuk melatihnya. Perbandingan data *train* dan data *test* dalam penelitian ini adalah 80%:20%. Data kelas positif dan negatif yang tidak *imbalance* ditangani dengan *resampling* menggunakan ROSE.

Proses klasifikasi diawali dengan memproses input pada arsitektur *Artificial Neural Network* berupa vektor matriks hasil TF-IDF dengan baris adalah dokumen dengan jumlah sebanyak 3587 dan kolom adalah kata dalam seluruh dokumen sebanyak 1124. Masing-masing kata diproses ke dalam propagasi maju ANN. Model ANN dibangun dengan cara *trial and error* pada *hyperparameter* jumlah node *hidden layer*, *learning rate* dan jumlah *batch size* yang mungkin untuk mendapatkan model terbaik. *Epoch* ditentukan dengan sebesar 1000 *epoch* untuk semua model ANN. Namun, dapat berhenti sebelum mencapai 1000 karena menggunakan *early stopping* untuk menghindari terjadinya *overfitting* atau kondisi saat model memiliki nilai akurasi tinggi pada data *train*, tetapi nilai akurasi rendah pada data *test* yang terlalu parah. *Batch size* yang digunakan sebesar 32 dan 64. *Learning rate* yang digunakan sebesar 0,01, 0,001, serta 0,0001. Propagasi maju ANN dihubungkan menggunakan jaringan *hidden layer* sebanyak satu unit dengan fungsi aktivasi *sigmoid biner* serta optimasi menggunakan Adam untuk *update* bobot dan *cross entropy* untuk mengetahui *loss* yang dihasilkan pada saat *training data*. *Trial and error* model ANN terdiri dari 18 model *trial* dengan kombinasi *hyperparameter* berbeda. Tabel 3 menunjukkan hasil proses *training* model ANN masing-masing *trial*.

Tabel 3. Hasil Percobaan Model ANN

Trial	Node Hidden Layer	Learning Rate	Batch Size	Sebelum Resampling	Setelah Resampling
				Validation Loss	
1	10	0,01	32	0,2185	0,0945
2	20			0,2202	0,1061
3	30			0,2201	0,1001
4	10	0,001		0,2161	0,5578
5	20			0,2170	0,1182
6	30			0,2174	0,1270
7	10	0,0001		0,2122	0,3682
8	20			0,2131	0,3275
9	30			0,2133	0,2977
10	10	0,01	64	0,2234	0,0939
11	20			0,2246	0,0959
12	30			0,2233	0,1014
13	10	0,001		0,2207	0,4553
14	20			0,2212	0,4820
15	30			0,2212	0,5117
16	10	0,0001		0,2158	0,3583
17	20			0,2156	0,3108
18	30			0,2157	0,2759

Model terbaik berdasarkan Tabel 3 yaitu model *trial* 10 setelah *resampling* dengan *validation loss* terkecil yaitu 0,0939. Model *trial* 10 diperoleh dengan kombinasi 10 node *hidden layer*, *learning rate* 0,01 dan *batch size* sebanyak 64. Gambar 4 memperlihatkan plot loss dan akurasi dari Model *Trial* 10 setelah *resampling*.



Gambar 4. Plot Loss dan Akurasi pada Model *Trial* 10 setelah *resampling*

Penilaian kinerja akan dilakukan terhadap model klasifikasi terbaik yang telah diperoleh. *Confusion matrix* digunakan sebagai alat evaluasi kinerja klasifikasi pada penelitian ini. *Confusion matrix* membandingkan hasil klasifikasi prediksi dengan klasifikasi sebenarnya. Tabel 4 menunjukkan hasil dari *confusion matrix* algoritma *Artificial Neural Network* Model *Trial* 10 setelah *resampling*.

Tabel 4. Hasil *Confusion Matrix*

Kelas Aktual	Kelas Prediksi	
	Positif	Negatif
Positif	487	43
Negatif	51	137

Hasil *confusion matrix* kemudian digunakan untuk memperoleh tingkat akurasi:

$$accuracy = \frac{TP + TN}{TP + FP + TN + FN} = \frac{487 + 137}{487 + 51 + 137 + 43} = 0,8690$$

Tingkat akurasi diperoleh sebesar 0,8690 atau 87% yang artinya menunjukkan bahwa model ANN dengan optimasi adam yang dibangun mampu mengklasifikasikan 87% data ulasan pengguna terhadap aplikasi Microsoft Teams secara tepat.

Topik yang paling sering dibicarakan oleh pengguna aplikasi Microsoft Teams pada Google Play Store berdasarkan frekuensi kemunculan kata divisualisasikan dengan *word cloud*. Hasil visualisasi ulasan positif dan negatif dari hasil ekstraksi informasi ditunjukkan pada Gambar 5.



Gambar 5. Hasil *Word Cloud* Sentimen Positif (kiri) dan *Word Cloud* Sentimen Negatif (kanan)

Berdasarkan Gambar 5 dapat diketahui bahwa pada sentimen positif, kata yang banyak muncul yaitu kata “bagus”, “aplikasi”, “oke”, “mantap” yang menunjukkan bahwa pengguna aplikasi Microsoft Teams merasa aplikasi tersebut memiliki performa yang baik dilihat dari kata bagus, oke dan mantap serta indikasi kepuasan lainnya yang ditunjukkan dari kata-kata yang ukurannya lebih kecil yaitu kata “bantu” dan lain-lain. Hasil *word cloud* sentimen negatif menampilkan kata “aplikasi”, “masuk”, “eror”, “unduh” yang berukuran besar mengartikan bahwa kata tersebut sering muncul dalam ulasan sentimen negatif. Hal tersebut menunjukkan bahwa kendala yang dialami pengguna aplikasi Microsoft Teams saat menggunakan aplikasi tersebut diantaranya kendala eror terutama saat proses masuk aplikasi dan kendala saat proses pengunduhan serta kendala lain yang ditunjukkan oleh kata-kata lain yang ukurannya lebih kecil yaitu kata “susah”, “jelek”, dan lain-lain.

5. KESIMPULAN

Model klasifikasi *Artificial Neural Network* dengan optimasi Adam menggunakan data *resampling* dengan kombinasi *hyperparameter* node *hidden layer* sebanyak 10, *learning rate* sebesar 0,01 dan *batch size* sebanyak 64 menghasilkan akurasi sebesar 87%.

Ulasan Aplikasi Microsoft Teams pada Google Play Store cenderung banyak memiliki sentimen positif. Kata-kata yang sering dibicarakan pengguna terkait ulasan sentimen positif Microsoft Teams yaitu kata “bagus”, “aplikasi”, “oke”, “mantap” yang menunjukkan bahwa ulasan sentimen positif pengguna cenderung terkait tentang kepuasan terhadap performa aplikasi Microsoft Teams. Pada ulasan sentimen negatif, kata-kata yang sering muncul adalah “aplikasi”, “masuk”, “eror”, “unduh” menunjukkan topik yang sering dibicarakan pengguna pada sentimen negatif adalah kendala eror saat masuk aplikasi dan masalah pada pengunduhan aplikasi Microsoft Teams serta kendala-kendala lainnya.

DAFTAR PUSTAKA

- Castella, Q., Sutton, C. 2014. *Word Storms: Multiples of Word Cloud for Visual Comparison of Documents*. Seoul, s.n.
- Diouf, R. *et al.* 2019. *Web Scraping: State-of-the-Art and Areas of Application*. IEEE International Conference on Big Data (Big Data) Vol. 2019 : Hal. 6040-6042.
- Fausett, L. V., 1994. *Fundamentals of Neural Network. Achitectures, Algorithms, and Applications*. New Jersey: Prentice Hall.
- Hamzah, B. M. 2021. *Classification of Movie Review Sentiment Analysis Using Chi-Square and Multinomial Naïve Bayes with Adaptive Boosting*. Journal of Advances in Information Systems and Technology, Hal : 67-74.
- Handayanto, R, T. *et al.* 2021. Analisis Sentimen Pada Situs *Google Review* dengan *Naïve Bayes* dan *Support Vector Machine*. Jurnal Komtika (Komputer dan Informatika) Vol. 5, No. 2 : Hal. 153-163.
- Herlinawati, N., Yuliani, Y., Faizah, S., Gata, W., Samudi. 2020. Analisis Sentimen *Zoom Cloud Meetings* di *Play Store* Menggunakan *Naïve Bayes* dan *Support Vector Machine*. Journal of Computer Engineering System and Science Vol. 2, No. 5 : Hal. 293-298.
- Kim, S. W., Gil, J. M. 2019. *Research paper classifcation systems based on TF-IDF and LDA schemes*. Human-centric Computing and Information Science Vol. 9, No. 1. doi: 10.1186/s13673-019-0192-7.
- Kingma, D. P., dan Ba, J. 2014. *Adam: A Method for Stochastic Optimization*. 3rd International Conference on Learning Representations, ICLR 2015.
- Liu, B. 2012. *Sentiment Analysis and Opinion Mining*. California: Morgan & Claypoll.
- Manning, C. D., Raghavan, P., Schutze, H. 2009. *An Introduction to Information Retrieval*. Cambridge: Cambridge University Press.
- Patra, A., Singh, D. 2013. *Neural Network Approach for Text Classification using Relevance Factor as Term Weighing Method*. International Journal of Computer Applications Vol.68, No.17 : Hal. 37-41.
- Sadowski, P. 2016. *Notes on Backpropagation*. Department of Computer Science: University of California Irvine.
- Siang, J. J., 2009. Jaringan Syaraf Tiruan dan Pemrogramannya Menggunakan *Matlab*. Yogyakarta: Andi.
- Tala, F. 2003. *A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia*. Amsterdam Institute for logic Language, and Computation, Universeit Van Amsterdam.
- Vailshery, L. 2022. *Number of Daily Active Users (DAU) of Microsoft Teams Worldwide As of April 2021 (In Millions)*. <https://www.statista.com/statistics/1003742/worldwide-microsoft-teams-daily-and-monthly-users/>.
- Wahid, D. H., Azhari, S. N. 2016. Peringkasan Sentimen Esktraktif di Twitter Menggunakan Hybrid TF-IDF dan Cosine Similarity. IJCCS Indonesian Journal of Computing and Cybernetics Systems Vol.10, No. 2 : Hal. 207-218.
- Wang, Y., Zhou, P., Zhong, W. 2018. *An Optimization Strategy Based on Hybrid Algorithm of Adam and SGD*. International Conference on Electronic Information Technology and Computer Engineering (EITCE 2018).
- Warsito, B., Santoso, R., Yasin, H. 2021. *Metaheuristic Optimization in Neural Network Model for Seasonal Data*. TELKOMNIKA Telecommunication Computing Electronics and Control Vol. 19, No. 6 : Hal. 1892-1901.