

Jurnal Ilmiah

Informatika Komputer

ANALISIS PERINGKAT TOP BRAND OJEK ONLINE MENGGUNAKAN JEJARING SOSIAL PERCAKAPAN TWITTER Ilham Aminudin, Dyah Anggraini	88
APLIKASI KOREKSI KESALAHAN PENULISAN KATA DALAM BAHASA INGGRIS DENGAN MENGGUNAKAN ALGORITMA RABIN-KARP Sendy Agustian, Kenny, Kristen Margi Suryaningrum	105
PREDIKSI CURAH HUJAN DI JAKARTA BERBASIS ALGORITMA LEVENBERG MARQUARDT Gema Indah Merdekawati, Ismail	116
PUBLIC SENTIMENT ANALYSIS OF PASAR LAMA TANGERANG USING K-NEAREST NEIGHBOR METHOD AND PROGRAMMING LANGUAGE R Hustinawaty, Rama Al Azis Dwiputra, Tavipia Rumambi	129
IMPLEMENTASI METODE EUCLIDEAN DISTANCE UNTUK EKSTRAKSI FITUR JARAK PADA CITRA SKELETON Miftahul Jannah, Nurul Humaira	134
IMPLEMENTASI LEXICON BASED DAN NAIVE BAYES PADA ANALISIS SENTIMEN PENGGUNA TWITTER TOPIK PEMILIHAN PRESIDEN 2019 Gusti Nur Aulia, Eka Patriya	140
APLIKASI VIRTUAL REALITY TAMAN WISATA BAWAH LAUT SEA WORLD ANCOL BERBASIS ANDROID Deviliana, Nurma Nugraha	154

DEWAN REDAKSI JURNAL ILMIAH INFORMATIKA KOMPUTER

Penanggung Jawab

Prof. Dr. E.S. Margianti, S.E., M.M.
Prof. Suryadi Harmanto, SSI., M.M.S.I.
Drs. Agus Sumin, M.M.S.I.

Dewan Editor

Dr. Fitriarningsih, Universitas Gunadarma
Dr. Rodiah, Universitas Gunadarma
Dr. Ias Sri Wahyuni, S.Si, M.Si, Universitas Gunadarma

Mitra Bebestari

Prof. Dr. Rer. Nat. A. Benny Mutiara, Universitas Gunadarma
Prof. Dr. Achmad Nizar Hidayanto, Universitas Indonesia
Prof. Dr.-Ing. Adang Suhendra, S.Si, S.Kom, M.Sc, Universitas Gunadarma
Prof. Dr. Sarifuddin Madenda, Universitas Gunadarma
Surya Sumpeno, PhD, Institut Teknologi Sepuluh Nopember (ITS)
Noor Akhmad Setiawan, Ph.D, Universitas Gadjah Mada
Dr. Paulus Insap Santosa, Universitas Gadjah Mada
Dr. Agus Harjoko, M.Sc, Ph.D, Universitas Gadjah Mada
Dr. Ernastuti, Universitas Gunadarma
Dewi Agushinta R., Universitas Gunadarma
Dr. Eri Prasetyo, Universitas Gunadarma
Dr. Sunny Arief Sudiro, STMIK Jakarta STI&K
Dr. Tubagus Maulana Kusuma, S.Kom.,Mengsc, Universitas Gunadarma
Dr. Lussiana ETP, S.si., M.T, STMIK Jakarta STI&K
Harry Budi Santoso, Ph.D, Universitas Indonesia

Sekretariat Redaksi

Universitas Gunadarma
Jalan Margonda Raya No. 100 Depok 16424
Phone : (021) 78881112 ext 516.

JURNAL ILMIAH INFORMATIKA KOMPUTER

NOMOR 2, VOLUME 24, AGUSTUS 2019

DAFTAR ISI

ANALISIS PERINGKAT TOP BRAND OJEK <i>ONLINE</i> MENGUNAKAN JEJARING SOSIAL PERCAKAPAN TWITTER Ilham Aminudin, Dyah Anggraini	88
APLIKASI KOREKSI KESALAHAN PENULISANKATA DALAMBahasa INGGRIS DENGAN MENGUNAKAN ALGORITMA RABIN- KARP Sedy Agustian, Kenny, Kristien Margi Suryaningrum	105
PREDIKSI CURAH HUJAN DI JAKARTA BERBASIS ALGORITMA LEVENBERG MARQUARDT Gema Indah Merdekawati, Ismail	116
PUBLIC SENTIMENT ANALYSIS OF PASAR LAMA TANGERANG USING K-NEAREST NEIGHBOR METHOD AND PROGRAMMING LANGUAGE R Hustinawaty, Rama Al Azis Dwiputra, Tavipia Rumambi	129
IMPLEMENTASI METODE <i>EUCLIDEAN DISTANCE</i> UNTUK EKSTRAKSI FITUR JARAK PADA CITRA <i>SKELETON</i> Miftahul Jannah, Nurul Humaira	134
IMPLEMENTASI LEXICON BASED DAN NAIVE BAYES PADA ANALISIS SENTIMEN PENGGUNA TWITTER TOPIK PEMILIHAN PRESIDEN 2019 Gusti Nur Aulia, Eka Patriya	140
APLIKASI VIRTUAL REALITY TAMAN WISATA BAWAHLAUT SEA WORLD ANCOL BERBASIS ANDROID Deviliana, Nurma Nugraha	154

ANALISIS PERINGKAT TOP BRAND OJEK ONLINE MENGUNAKAN JEJARING SOSIAL PERCAKAPAN TWITTER

¹Ilham Aminudin, ²Dyah Anggraini

^{1,2}Jurusan Sistem Informasi, Universitas Gunadarma

Jl. Margonda Raya No 100, Pondok Cina, Depok 16424, Jawa Barat

¹ilhmdn@gmail.com, ²d_anggraini@staff.gunadarma.ac.id

Abstrak

Banyak bisnis mulai muncul dengan melibatkan pengembangan teknologi internet. Salah satunya adalah bisnis di aplikasi berbasis penyedia layanan di bidang moda transportasi berbasis online yang ternyata dapat memberikan solusi dan menjawab berbagai kekhawatiran publik tentang layanan transportasi umum. Kemacetan lalu lintas di kota-kota besar dan ketegangan publik dengan keamanan transportasi umum diselesaikan dengan adanya aplikasi transportasi online seperti Grab dan Gojek yang memberikan kemudahan dan kenyamanan bagi penggunanya. Penelitian ini dilakukan untuk menganalisa keaktifan percakapan brand jasa transportasi online di jejaring sosial Twitter berdasarkan properti jaringan. Penelitian dilakukan dengan mengambil data dari percakapan pengguna di social media Twitter dengan cara crawling menggunakan Bahasa pemrograman R programming dan software R Studio dan pembuatan model jaringan dengan software Gephy. Setelah itu data dianalisis menggunakan metode social network analysis yang terdiri berdasarkan properti jaringan yaitu size, density, modularity, diameter, average degree, average path length, dan clustering coefficient dan nantinya hasil analisis akan dibandingkan dari setiap properti jaringan kedua brand jasa transportasi Online dan ditentukan strategi dalam meningkatkan dan mempertahankan keaktifan serta tingkat kehadiran brand jasa transportasi online, Grab dan Gojek.

Kata Kunci: Analisis peringkat brand, Social Network Analysis, top brand, transportasi online, Twitter

Abstract

Many businesses began to emerge involving the development of internet technology. One of them is a business in an application based on service providers in the field of online-based transportation that turns out to be able to provide solutions and answer various public concerns about public transportation services. Traffic congestion in big cities and public tensions with public transportation security is solved by the presence of online transportation applications that provide convenience and comfort for its users. This research was conducted to analyze the activeness of brand conversations online transportation services on Twitter social networks based on network properties. The research was conducted by retrieving data from users' conversations on Twitter social media by crawling using R programming language and R Studio software and making network models with Gephy software. After that the data is analyzed using the social network analysis method based on network properties namely size, density, modularity, diameter, average degree, average path length, and clustering coefficient and later the results of the analysis will be compared from each network property of both Online transportation service brands and determined strategy in improving and maintaining the activity and presence level of online transportation service brands, Grab and Gojek.

Keywords: Brand ranking analysis, Social Network Analysis, top brand, online transportation, Twitter

PENDAHULUAN

Perkembangan teknologi internet yang cepat menyebabkan perubahan kebiasaan di masyarakat. Banyak bisnis mulai muncul dengan melibatkan pengembangan teknologi internet. Salah satunya adalah bisnis di aplikasi berbasis penyedia layanan di bidang moda transportasi berbasis *online* yang ternyata dapat memberikan solusi dan menjawab berbagai kekhawatiran publik tentang layanan transportasi umum. Kemacetan lalu lintas di kota-kota besar dan ketegangan publik dengan ke-amanan transportasi umum diselesaikan dengan adanya aplikasi transportasi *online* yang memberikan kemudahan dan Kenyamanan bagi penggunanya. Bisnis transportasi *online* yang akhir-akhir ini sangat dikenal adalah Grab dan Gojek. Kedua perusahaan transportasi berbasis aplikasi terbukti menyebabkan perubahan signifikan terutama untuk kehidupan masyarakat umum. Bahkan, perusahaan-perusahaan ini telah menjadi perusahaan layanan di bidang moda transportasi berbasis *online* sebagai pilihan utama publik terutama di kota-kota besar[1].

Dari hasil peringkat *Top Brand Award* tahun 2018 fase 2 pada peringkat pertama dalam kategori jasa transportasi *online* diraih oleh Grab dengan nilai 48% kemudian diikuti oleh Gojek pada peringkat kedua dengan nilai 44.9%. Dilakukan survey dengan menyebarkan kuesioner ke 11 kota di Indonesia secara manual (konvensional) untuk mendapatkan hasil *brand* mana yang paling diminati dan

dipilih oleh konsumen [2]. Metode *Social Network Analysis* (SNA) digunakan untuk mengetahui strategi layanan transportasi manakah yang memiliki keaktifan lebih tinggi dengan membandingkan properti jaringan yang melekat yang ada di media sosial Twitter. Data pengguna media sosial Twitter digunakan agar memberikan hasil keputusan lebih cepat, serta efisiensi biaya dan efektifitas waktu bagi perusahaan dalam pengambilan keputusan yang cepat.

Dalam penelitian [3] menunjukkan perbandingan nilai yang didapat dari masing-masing properti jaringan kedua ritel waralaba Alfamart dan Indomaret menggunakan metode *Social Network Analysis*, mendapatkan hasil yang sama dengan penentuan peringkat *brand* menggunakan *Top Brand Award* atau metode konvensional. Dari kedua ritel waralaba tersebut, Alfamart memiliki jaringan yang paling banyak mendapatkan nilai tertinggi dari masing-masing properti jaringan yang dihitung, dengan total empat dari tujuh perhitungan properti jaringan, yaitu pada: *density*, *diameter*, *average path length*, dan *clustering coefficient*. Sedangkan Indomaret memiliki jaringan dengan tiga nilai tertinggi pada properti jaringan yang dihitung, yaitu pada: *size*, *modularity*, dan *average degree*. Sehingga urutan peringkat *brand* yang dihasilkan pada ritel waralaba menggunakan metode SNA adalah Alfamart mendapat peringkat pertama dan Indomaret mendapat peringkat kedua dengan perbedaan sangat

tipis yaitu Alfamart unggul salah satu nilai dari properti jaringan yang dihitung.

Penelitian lain yaitu pada penelitian [4] juga menunjukkan bahwa hasil perhitungan dan analisis perbandingan berdasarkan properti jaringan didapat nilai untuk masing – masing jaringan *smartphone*. Setelah dilakukan perbandingan, didapat peringkat berdasarkan metriknya, yaitu Samsung unggul pada 2 properti yaitu *average degree* dan *clustering coefficient*. Berarti dengan unggul pada *average degree*, Samsung memiliki banyak jumlah *link* yang menghubungkan *node* maka penyebaran informasi mengenai Samsung akan semakin cepat. Sedangkan *clustering coefficient* menunjukkan hubungan *node* yang ada dalam jaringan Samsung itu kuat. Nokia unggul pada 2 properti juga yaitu diameter dan *average path length*, artinya jaringan Nokia memiliki *shortcut* dengan jalur terpendek. Nokia juga unggul pada *metric average path length* untuk menunjukkan jarak rata-rata antara suatu *node* dengan *node* lain, semakin kecil nilai *average path length* berarti semakin cepat penyebarannya. Blackberry juga unggul pada 2 properti yaitu *density* dan *modularity*. Dimana *density* menunjukkan kepadatan suatu jaringan, hal ini berarti Blackberry memiliki jaringan yang paling padat diantara yang lain. Sedangkan *modularity* menunjukkan kelompok yang terbentuk didalam jaringan Blackberry sangat tinggi, yang berarti jaringan tersebut membentuk kelompok-kelompok yang banyak. Iphone unggul pada 2 properti yaitu

size dan *connected component*. *Size* menunjukkan bahwa aktor yang berperan di jaringan Iphone banyak. Sedangkan *Connected component* merupakan kumpulan dari “pecahan” dalam satu graf yang saling terpisah. Artinya semakin tinggi nilai *connected component* dari Iphone menunjukkan bahwa jaringan tersebut dapat membentuk banyak kelompok yang saling terhubung satu sama lain. Dalam meningkatkan dan mempertahankan peringkat keaktifan *smartphone* di jejaring sosial percakapan di Twitter, yaitu dengan menentukan strategi dengan menarik perhatian pengguna dengan konten yang menarik dalam akun media sosial Twitter. Hal ini secara otomatis dapat menarik perhatian pengguna sehingga dapat menyebabkan banyak interaksi didalamnya.

Pada tahun 2017, upaya meningkatkan *brand awareness* transportasi *online* dilakukan [5] dengan pengambilan data melalui wawancara. Penelitian ini dilakukan melalui aktivitas *public relation* untuk mengatasi persaingan di antara perusahaan-perusahaan transportasi *online*, sehingga perlu diciptakannya *brand* yang kuat ditengah masyarakat. Kegiatan marketing *public relations* berlaku dalam meningkatkan *brand awareness* ke 7 strategi *marketing public relations*. Kegiatan seperti publikasi, melaksanakan kegiatan-kegiatan menarik, *establish good* hubungan dengan masyarakat, menjalin kerjasama dengan perusahaan lain dilakukan untuk memberikan pelayanan terbaik kepada konsumen.

Transportasi Online

Transportasi merupakan sarana yang sangat dibutuhkan pada zaman sekarang, karena dengan adanya transportasi dapat mengefektifkan pekerjaan dan membantu dalam memenuhi kebutuhan sehari-hari. Transportasi terbagi menjadi tiga bagian, yaitu: jalur darat, laut, dan udara. Ketertarikan masyarakat terhadap transportasi sangat tinggi, dengan alasan untuk mempersingkat waktu perjalanan. Kemajuan yang sangat pesat di bidang teknologi informasi memberikan pengaruh yang besar terhadap berbagai aspek kehidupan manusia. Salah satu hasil kemajuan teknologi informasi yang berkontribusi besar terhadap perubahan ini adalah internet. Kehadiran jasa transportasi berbasis aplikasi *online* yang menggunakan internet sangat berpengaruh bagi masyarakat dalam segala aktivitas secara cepat dan efisien. Salah satu bisnis yang sedang berkembang saat ini adalah bisnis jasa transportasi dengan sepeda motor atau yang dulu biasa disebut ojek. Jika dahulu ojek dikelola secara konvensional dan kepemilikan tunggal, sekarang muncul bisnis baru taxi motor yaitu suatu usaha komersial, yang menyediakan jasa transportasi bagi umum dan dikelola secara profesional. Berawal dari banyaknya jasa transportasi yang bermunculan mengakibatkan semakin kuat pula persaingan yang dihadapi oleh para pengemudi ojek. Hampir disetiap sudut jalan besar dan area kampus banyak ditemui para pengemudi ini. Apabila dicermati ada banyak

hal yang harus dibenahi dalam layanan ojek yang ada saat ini khususnya masalah keselamatan, kesopanan dan kewajaran harga layanan. Masalah harga yang paling sering menjadi ganjalan penumpang. Pengemudi ojek memasang tarif semau mereka sendiri. Tarif ojek untuk jarak dekat tidak terpaut jauh dengan taksi mobil biasa. Muncul ide perusahaan taksi motor, dengan *member value added* yang berbeda dengan ojek umumnya. Dikelola secara profesional, dengan menyediakan *call center* dan kantor. Pelayanan yang diberikan memang lebih standar seperti standar keselamatan, etika kesopanan pengendara dan yang termasuk penting tarif yang sudah baku [6].

Internet

Menurut data yang dihasilkan oleh APJII pengguna internet di Indonesia terus meningkat sejak tahun 1998 hingga tahun 2017 yaitu sebesar 143 juta pengguna. Jumlah pengguna internet di Indonesia diproyeksikan tembus 175 juta pada 2019, atau sekitar 65,3% dari total penduduk 268 juta. Peningkatan pengguna internet terutama ditopang oleh semakin meluasnya penggunaan ponsel pintar di Indonesia. Peningkatan jumlah pengguna internet yang signifikan itu pun diharapkan bisa memberikan dampak positif ke berbagai kegiatan produktif yang akan mendongkrak ekonomi nasional, terutama ekonomi digital termasuk di dalamnya layanan di bidang moda transportasi berbasis *online*[7].

Jejaring Sosial

Jejaring sosial bisa dikatakan sebagai sebuah teknologi yang kini meningkat penggunaannya sebagai sumber informasi karena teknologi ini memungkinkan orang-orang untuk dapat mengirim dan menerima informasi dengan cepat. Jejaring ini merupakan sebuah aplikasi yang para penggunanya dapat melakukan kolaborasi untuk menciptakan dan mendistribusikan sebuah konten dan jejaring sosial ini dibangun berdasarkan karakteristik web 2.0 [8].

Twitter

Twitter adalah sebuah jaringan informasi yang terdiri dari pesan 140 karakter yang disebut tweet [9]. Suh, Hong, Pirolli dan Chi [10] mengungkapkan Twitter adalah layanan *microblogging* yang memungkinkan pengguna untuk mengirim dan membaca pesan singkat sebanyak 140 karakter yang dikenal sebagai *tweets*, yang memungkinkan pengguna untuk berbagi dan menemukan topik yang menarik secara *real time*.

Social Network Analysis

Social Network Analysis (SNA) dapat dideskripsikan sebagai suatu ilmu yang mempelajari hubungan manusia dengan bantuan teori graf. SNA mempelajari struktur hubungan yang mengaitkan individu atau unit sosial lain dan ketergantungan dalam perilaku atau sikap yang berhubungan dengan susunan hubungan sosial. Hubungan tersebut digambarkan dengan *nodes*, atau bisa disebut

vertices yang melambangkan aktor atau user dan *ties* atau disebut juga *edges*, *links* atau *connections* yang melambangkan hubungan antar actor [3][4][5].

SNA bukan hanya sebuah metodologi namun sebuah perspektif yang unik tentang bagaimana fungsi masyarakat. Sedangkan dalam praktik bisnis, SNA dapat dimanfaatkan untuk menganalisis dan meningkatkan alur komunikasi dalam suatu organisasi atau dengan *network partner* dan pelanggan[5]. SNA menggunakan graf untuk menggambarkan hubungan ini, jenis graf ini disebut Sociogram [13].

Properti Jaringan

Jaringan memiliki beberapa atribut yang dapat dihitung dan dianalisis yang disebut properti jaringan. Properti jaringan ini dapat digunakan untuk menentukan model sebuah jaringan dengan menggunakan metode SNA. Adapun properti jaringan yang akan digunakan adalah sebagai berikut: *size*, *density*, *modularity*, *diameter*, *Average Degree*, *Average Path Length*, dan *Clustering Coefficient*.

Network size (N) menunjukkan jumlah komponen dalam suatu sistem [3] [8]. Secara sederhana, *network size* bisa didapatkan dengan menghitung jumlah *node* yang ada dalam suatu *network* [8]. Menurut [5] *network density* merupakan jumlah dari seluruh hubungan yang ada (*actual ties*), dibagi dengan jumlah hubungan yang mungkin terjadi (*possible ties*) dalam suatu *network*.

Density bisa dikatakan juga sebagai ukuran seberapa erat hubungan antar *node* dalam suatu *network*.

Modularity didefinisikan sebagai sekelompok *link* dalam suatu komunitas dikurangi nilai yang diharapkan dari kelompok tersebut apabila posisi *link* tersebut diacak[4][10]. *Modularity* digunakan untuk mengukur kekuatan dari pembagian jaringan dalam kelompok-kelompok [9][10]. Semakin tinggi nilai *modularity* dari suatu *network* tersebut berisikan kelompok-kelompok didalamnya. *Modularity* menunjukkan bagaimana group yang berbeda dibentuk dalam suatu jaringan. Koefisien *modularity* yang lebih besar berarti lebih jelas batas antara kelompok-kelompok dalam jaringan. Dengan demikian *modularity* yang lebih tinggi dapat meningkatkan kesadaran akan merk, jika lebih beragam komunitas maka penyebaran informasi produk lebih me-ningkat. *Diameter network* dilambangkan dengan d_{max} adalah jalur terdekat maksimal dalam suatu *network* atau bisa disebut juga jarak terbesar antara sepasang *node* [7][11].

Average degree merupakan salah satu bagian penting dalam properti jaringan, dimana *average degree* dari *network* memberikan derajat rata-rata dari jumlah *link* yang menghubungkan satu *node* dengan *node* yang lain [11]. Semakin banyak *link* yang menghubungkan suatu *node* kepada *node* yang lain, berarti penyebaran informasi akan semakin cepat dan mudah.

Average path length dilambangkan dengan $\langle d \rangle$, adalah jarak rata-rata antara satu *node* dengan *node* yang lain dalam suatu *network*[11]. Dalam implementasi di media sosial, *average path length* diterjemahkan sebagai jumlah rata-rata akun atau *node* yang harus dilewati oleh suatu akun untuk mencapai suatu akun tertentu. Semakin banyak akun yang dilewati maka semakin baik.

Clustering coefficient menggambarkan bagaimana suatu *node* berhubungan dengan *node* lain di sekitarnya. *Clustering coefficient* dilambangkan dengan C_i , apabila $C_i=0$, berarti *node* di sekitar *node* i tidak berhubungan satu sama lain. Sedangkan apabila $C_i=1$, berarti *node* di sekitar *node* i semuanya saling terhubung. Semakin besar nilai C_i , berarti hubungannya semakin padat dan mengindikasikan fenomena *small world* [11]. Apabila aktor dalam *network* saling mengenal (terhubung) satu sama lain, berarti informasi yang disebarkan akan lebih cepat diketahui.

Top Brand Award

Top Brand Award merupakan apresiasi merek yang dikategorikan sebagai merek teratas. Top Brand Award diberikan kepada merek di kategori produk tertentu yang memenuhi kriteria. Kriteria didasarkan pada survei yang dilakukan oleh Frontier Consulting Group. *Brand* teratas pada survey yang dilakukan oleh Frootier Consulting Group untuk menghasilkan Top Brand Award terdiri dari dua panel yaitu B2B responden

dan B2C responden. B2C responden melibatkan 7500 sampel acak dan 1600 sampel booster. Survei ini dilakukan di sebelas kota, yaitu Jakarta, Bandung, Semarang, Surabaya, Medan, Makassar, Pekanbaru, Balikpapan, Palembang, Samarinda dan Denpasar. Sementara itu, Top Brand survei untuk B2B responden melibatkan 1600 perusahaan berlokasi di Jakarta, Bandung, dan Surabaya[2]

Metode sampling untuk sampel acak B2C adalah sampling acak bertahap sedangkan metode untuk B2C booster sampel adalah sampel *purposive*. Survei dilakukan melalui wawancara pribadi tatap muka dengan menggunakan kuesioner terstruktur. Metode sampling untuk B2B adalah *stratified sampling acak*, dengan menggunakan jalur bisnis sebagai *database*.

R Programming

R adalah suatu bahasa komputer dan merupakan lingkungan pemrograman interaktif untuk analisis data dan grafik. Bahasa R adalah bahasa tingkat tinggi (*very high level language*) untuk komputasi. Bahasa R memungkinkan untuk menghitung, melihat data dan program secara interaktif dengan umpan balik yang cepat. R dapat digunakan pada berbagai bidang seperti analisis keuangan, penelitian statistika, manajemen, akademis, matematika, grafik dan analisis data. R Studio merupakan *Integrated Development Environment (IDE)* khusus bagi bahasa pemrograman R[21].

Gephi

Software Gephi adalah sebuah aplikasi yang bersifat *open source* untuk melakukan eksplorasi dan manipulasi jaringan. Sebuah modul jaringan yang akan dikembangkan dapat diolah dengan diimpor, divisualisasikan, dipetakan, difilter, dimanipulasi dan diekspor di dalam *softwareGephi* tersebut. *Gephi* didasarkan pada paradigma visualisasi dan manipulasi yang memungkinkan setiap pengguna untuk menemukan jaringan dan properti data. Selain itu, dirancang untuk mengikuti rantai studi kasus, dari file data ke peta yang dapat.

Crawling

Crawling adalah suatu teknik yang digunakan untuk mengumpulkan informasi yang ada dalam *web*. *Crawling* bekerja secara otomatis, dimana informasi yang dikumpulkan berdasarkan atas kata kunci yang diberikan oleh pengguna. Alat yang digunakan untuk melakukan *crawling* disebut dengan *crawler*. *Crawler* berbentuk program yang diprogram dengan algoritma tertentu, sehingga dapat melakukan pemindaian ke halaman-halaman *web*, sesuai dengan alamat web ataupun kata kunci yang diberikan pengguna. Pada saat melakukan pemindaian, *crawler* akan membaca teks yang ada, *hyperlink* dan berbagai tag yang digunakan di halaman web tersebut. Berdasarkan informasi tersebut, *crawler* akan mengindeks informasinya atau pun menyimpan informasi tersebut ke dalam sebuah *file* atau ke dalam *database*.

Text Mining

Text mining adalah sebuah penelitian yang mencoba memecahkan masalah informasi yang overload, dengan menggunakan teknik dari *data mining*, *machine learning*, *Natural Language Processing (NLP)*, *Information Retrieval (IR)*, dan *knowledge management*. *Text mining* melibatkan pre-processing koleksi dokumen (kategorisasi teks, ekstraksi informasi, ekstraksi istilah), penyimpanan representasi menengah, teknik untuk menganalisis representasi menengah ini (seperti analisis distribusi, pengelompokan, analisis tren, dan peraturan asosiasi), dan visualisasi hasilnya [13].

METODE PENELITIAN

Dalam penelitian ini penulis menggunakan metode *Social Network Analysis* yang terdiri dari beberapa tahapan. Pada tahap 1 menggunakan *Social Network Analysis*, karena tujuan utama dari penelitian ini adalah dapat menganalisis keaktifan percakapan *brand* jasa transportasi *online*, menentukan peringkat Top Brand jasa transportasi *online*, dan menentukan strategi dalam meningkatkan dan mempertahankan keaktifan dan tingkat kehadiran *brand* jasa transportasi *online* melalui percakapan di jejaring sosial Twitter dengan membandingkan properti jaringan.

Tahap 2, mengidentifikasi masalah yang ditemukan dari latar belakang. Penentuan peringkat dengan melihat keaktifan dan

tingkat kehadiran *brand* jasa transportasi *online* menggunakan jejaring sosial Twitter dengan metode *Social Network Analysis (SNA)* berdasarkan properti jaringan merupakan suatu strategi baru yang dapat digunakan. Data percakapan yang terbentuk dalam jejaring sosial belum dimanfaatkan secara maksimal untuk menghimpun informasi terhadap pola komunikasi jaringan secara jelas.

Pada tahap 3, Pengumpulan data berbasis pada konsep *User Generated Content (UGC)*, dimana data yang diambil berasal dari jejaring sosial yang berisikan konten-konten yang dibuat sendiri oleh pengguna. Media yang digunakan sebagai sumber data adalah media sosial Twitter dengan tweet sebagai konten data yang akan diambil. Data yang diambil adalah seluruh *tweet* baik berupa *retweet*, *reply*, maupun *mention* yang berisi kata kunci 'Grab dan 'Gojek'. Data yang diambil dalam penelitian diambil dengan rentang waktu dari tanggal 4 Juni 2019 sampai dengan tanggal 11 Juni 2019 dan dibatasi hanya 500 data. Pengumpulan data dilakukan menggunakan software R Studio dengan cara *crawling* menggunakan R *Programming*.

Selanjutnya data yang diperoleh dalam bentuk CSV yang masih memiliki dimensi yang tinggi, terdapat *noise* pada data, dan terdapat struktur teks yang tidak baik. Maka dilakukan proses *text mining* agar atribut-atribut yang tidak diperlukan dihilangkan agar sisa data sesuai dengan

kebutuhan penelitian sehingga data akan lebih mudah untuk dianalisis.

Analisis data dilakukan pada tahap ke 5, menggunakan metode *Social Network Analysis* (SNA) dengan membandingkan property jaringan yang dihasilkan otomatis melalui software Gephi. Setelah membandingkan properti jaringan pada kedua jaringan *brand* jasa transportasi *online*, maka akan dipaparkan kesimpulan dari hasil analisis yang diperoleh, selain itu dapat ditentukan peringkat keaktifan dan tingkat kehadiran *brand* melalui percakapan di jejaring sosial Twitter dan menentukan strategi agar kedua *brand* jasa transportasi *online* tersebut bisa mempertahankan atau bahkan meningkatkan kualitasnya dengan membuat keputusan yang cerdas mengenai kebutuhan, sikap, pendapat, tren terbaru dan berbagai faktor yang mempengaruhi pengguna dari hasil perbandingan properti jaringan yang telah dianalisis. Kemudian hasil penelitian akan dipaparkan agar dapat bermanfaat untuk penelitian selanjutnya.

HASIL DAN PEMBAHASAN

Jaringan Transportasi *Online*

Setelah melakukan proses *crawling* data, *text mining* dan pembuatan model jaringan dengan menggunakan *software* Gephy, selanjutnya dilakukan pembahasan mengenai setiap model jaringan dari setiap *brand* jasa transportasi *online* yaitu Grab dan Gojek. Jenis *graph* yang digunakan adalah

undirected graph dimana *graph* ini tidak mempertimbangkan arah hubungan antar *node*. Jenis *graph* tidak menunjukkan *indegree* (*node* yang dituju) dan *outdegree* (*node* asal).

Layout graf yang dipilih dalam pembuatan model jaringan ini adalah *layout* Yifan Hu Proportional. Algoritma tata letak Multilevel Yifan Hu Proportional adalah suatu algoritma yang menyatukan bagian-bagian yang baik dari algoritma yang diarahkan gaya dan algoritma multilevel untuk mengurangi kompleksitas algoritma. Ini adalah salah satu algoritma yang bekerja sangat baik dengan jaringan besar.

Jaringan Grab

Hasil visualisasi model jaringan Grab menunjukkan setiap *actor* atau *node* yang membicarakan Grab dalam media *social* Twitter diambil dari seluruh tweet baik berupa *retweet*, *reply*, maupun mention yang berisi kata kunci 'Grab' dengan rentang waktu dari tanggal 4 Juni 2019 sampai dengan tanggal 11 Juni 2019. Selain itu ditunjukkan juga antar *node* yang terhubung melalui garis atau *edge*. Selain itu ditunjukkan juga antar *node* yang terhubung melalui garis atau *edge* pada jaringan Grab. Hasil visualisasi model jaringan Grab dapat dilihat pada Gambar 1. Setelah dilakukan pembuatan visualisasi model jaringan, diperlukan perhitungan properti jaringan agar dapat dianalisa lebih lanjut. Perhitungan dapat dilakukan secara otomatis melalui *software* Gephy. Perhitungan tersebut didapatkan hasil nilai

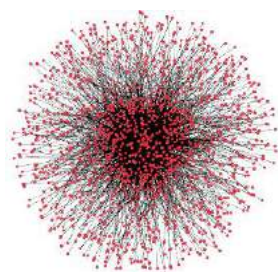
dari masing-masing properti jaringan pada jaringan Grab dapat dilihat pada Tabel 1. Properti jaringan pertama yaitu size terdiri dari *node* dan *edge* dengan hasil perhitungan properti size pada jaringan grab sebanyak 1446 *nodes* dan 3895 *edges*, hal ini berarti pada jaringan Grab dapat dikatakan cukup aktif, karena banyak aktor yang berinteraksi. Properti jaringan kedua yaitu *density*, didapatkan hasil yaitu jaringan grab memiliki nilai *density* yaitu 0.004, hal ini menunjukkan pada jaringan Grab memiliki hubungan yang erat antar *node* dan memiliki hubungan yang kuat.

Properti jaringan ketiga yaitu *modularity* dengan nilai yaitu 0.216, hal ini menunjukkan akan ada kelompok yang berbeda-beda terbentuk dalam jaringan Grab.

Setiap kelompok yang terbentuk dapat menjadi komunitas yang berbeda sehingga butuh spesifikasi lebih terhadap produk di setiap komunitas.

Properti jaringan keempat yaitu diameter sebesar 4, hal ini menunjukkan bahwa pada jaringan Grab memiliki diameter yang kecil sehingga akan memudahkan *node* untuk saling berkomunikasi karena jaraknya yang pendek.

Properti jaringan Kelima yaitu *average degree* dengan nilai sebesar 5.387, hal ini menunjukkan bahwa penyebaran informasi pada jaringan Grab akan semakin cepat dan mudah karena memiliki derajat rata-rata dari jumlah *link* yang menghubungkan *node* satu dengan *node* lain *link*.



Gambar 1. Visualisasi Model Jaringan Grab

Tabel 1. Perhitungan Jaringan Grab

No	Properti Jaringan	Hasil
1	Size	Nodes : 1446 Edges : 3895
2	Density	0.0004
3	Modularity	0.216
4	Diameter	4
5	Average Degree	5.387
6	Average Path Length	3.374
7	Clustering Coefficient	0

Properti jaringan keenam yaitu *average path length* dengan nilai sebesar 3.374, hal ini menunjukkan bahwa jaringan grab memiliki kecepatan informasi antar *actor* yang cukup cepat.

Properti jaringan ketujuh yaitu *clustering coefficient* yang 0, hal ini berarti dalam jaringan grab tidak ada *node* berhubungan dengan *node* lain di sekitar.

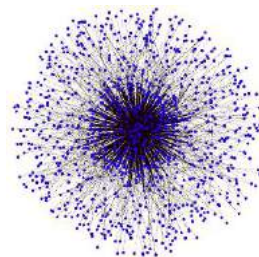
Jaringan Gojek

Hasil visualisasi model jaringan Gojek menunjukkan setiap *actor* atau *node* yang membicarakan Gojek dalam media social Twitter diambil dari seluruh tweet baik berupa retweet, reply, maupun mention yang berisi kata kunci 'Gojek dengan rentang waktu dari tanggal 4 Juni 2019 sampai

dengan tanggal 11 Juni 2019. Hasil visualisasi model jaringan Gojek dapat dilihat pada Gambar 2.

Setelah dilakukan pembuatan visualisasi model jaringan, diperlukan perhitungan properti jaringan agar dapat dianalisa lebih lanjut. Perhitungan dapat dilakukan secara otomatis melalui software Gephy Tabel 2 merupakan perhitungan tersebut didapatkan hasil nilai dari masing-masing properti jaringan pada jaringan Gojek.

Properti jaringan pertama yaitu size terdiri dari *node* dan *edge* dengan memiliki jaringan sebanyak 1073 *nodes* dan 2704 *edges*, hal ini berarti pada jaringan Gojek dapat dikatakan cukup aktif, karena banyak aktor yang berinteraksi.



Gambar 2. Visualisasi Model Jaringan Gojek

Tabel 2 Perhitungan Jaringan Gojek

No	Properti Jaringan	Hasil
1	Size	Nodes : 1073 Edges : 2704
2	Density	0.0005
3	Modularity	0.226
4	Diameter	4
5	Average Degree	5.04
6	Average Path Length	3.407
7	Clustering Coefficient	0

Properti jaringan kedua yaitu *density dengan* memiliki nilai 0.005, hal ini menunjukkan pada jaringan Grab memiliki hubungan yang sangat erat antar *node* dan memiliki hubungan yang sangat kuat.

Properti jaringan ketiga yaitu *modularity* dengan memiliki nilai yaitu 0.226, hal ini menunjukkan akan ada kelompok yang berbeda-beda terbentuk dalam jaringan Gojek. Setiap kelompok yang terbentuk dapat menjadi komunitas yang berbeda sehingga butuh spesifikasi lebih terhadap produk di setiap komunitas.

Properti jaringan keempat yaitu diameter dengan nilai yaitu 4, hal ini menunjukkan bahwa pada jaringan Gojek memiliki diameter yang kecil sehingga akan memudahkan *node* untuk saling berkomunikasi karena jaraknya yang pendek.

Properti jaringan Kelima yaitu *average degree* dengan memiliki nilai 5.04, hal ini menunjukkan bahwa penyebaran informasi pada jaringan Gojek akan semakin cepat dan mudah karena memiliki derajat rata-rata dari jumlah *link* yang menghubungkan *node* satu dengan *node* lain *link*.

Properti jaringan keenam yaitu *average path length* dengan memiliki nilai yaitu 3.407, hal ini menunjukkan bahwa jaringan Gojek memiliki kecepatan informasi antar *actor* yang cepat. Properti jaringan ketujuh yaitu *clustering coefficient* dengan memiliki nilai *clustering coefficient* yaitu 0, hal ini berarti dalam jaringan Gojek tidak ada *node* berhubungan dengan *node* lain di sekitar.

Perbandingan Properti Jaringan Grab dan Gojek

Masing-masing perhitungan properti jaringan pada kedua *brand* jasa transportasi *online* yaitu Grab dan Gojek, diperoleh nilai pada setiap properti jaringan. Perbandingan nilai properti jaringan dari masing masing *brand* jasa transportasi *online* Grab dan Gojek dapat dilihat pada Tabel 3.

Properti Size

Properti jaringan yang pertama adalah *size*. *Size* dikatakan aktif apabila terdapat banyak *node* dalam suatu jaringan sehingga banyak *node* yang berinteraksi. Semakin tinggi *node* maka semakin banyak aktor yang terlibat dalam jaringan sosial. Hal ini dapat diartikan bahwa banyak aktor yang *aware* terhadap keberadaan jasa transportasi *online*. Pada properti jaringan *size* terbagi menjadi nilai *node* dan nilai *edge*.

Nilai *node* yang tertinggi adalah Grab yaitu 1446 *nodes* yang menunjukkan terdapat 1446 aktor yang membicarakan di media sosial Twitter sedangkan gojek hanya memiliki 1073 *nodes*. Sedangkan *edge* menunjukkan interaksi yang terjadi antar aktor. Semakin tinggi nilai *edge* akan menunjukkan bahwa banyak percakapan yang membicarakan kedua *brand* jasa transportasi *online* di media sosial Twitter. Grab memiliki nilai *edge* tertinggi yaitu sebanyak 3895 *edges*, sedangkan gojek hanya memiliki 2704 *edges*.

Tabel 3. Perbandingan Properti Jaringan Grab dan Gojek

No	Properti Jaringan	Grab	Gojek	Peringkat
1	<i>Size</i>	Nodes: 1446 Edges: 3895	Nodes: 1073 Edges: 2074	1. Grab 2. Gojek
2	<i>Density</i>	0.0004	0.0005	1. Gojek 2. Grab
3	<i>Modularity</i>	0.216	0.226	1. Grab 2. Gojek
4	<i>Diameter</i>	4	4	1. Grab 2. Gojek
5	<i>Average Degree</i>	5.387	5.04	1. Grab 2. Gojek
6	<i>Average Path Length</i>	3.374	3.407	1. Grab 2. Gojek
7	<i>Clustering Coefficient</i>	0	0	1. Grab 2. Gojek

Properti Density

Properti jaringan yang kedua adalah *density*, dimana semakin padat jaringan akan menghasilkan nilai *density* yang lebih besar. Semakin besar nilai *density* dari suatu jaringan maka semakin baik karena aktor-aktor yang ada di dalam jaringan tersebut lebih banyak terhubung. Dari kedua *brand* jasa transportasi *online* tersebut yang memiliki nilai *density* yang paling besar adalah Gojek dengan nilai 0.005 dibandingkan Grab yang memiliki nilai *density* 0.004.

Properti Modularity

Properti jaringan selanjutnya adalah *modularity*. *Modularity* merupakan pengukuran untuk mendeteksi banyaknya komunitas/group/genk yang ada di dalam suatu graf. Semakin besar nilai *modularity* maka semakin jelas kelompok yang terbentuk. Setiap kelompok yang terbentuk dapat diasumsikan sebagai komunitas yang

berbeda sehingga butuh spesifikasi lebih terhadap produk di setiap komunitas. Dari kedua *brand* jasa transportasi *online* tersebut yang mendapatkan peringkat pertama untuk nilai *modularity* adalah Grab dengan nilai 0.216 dan Gojek yang memiliki nilai *modularity* 0.226.

Properti Diameter

Diameter merupakan jarak maksimal antar *nodes*. Semakin kecil diameter, maka akan semakin cepat informasi beredar dalam jaringan sosial tersebut. Grab dan Gojek memiliki nilai diameter yang sama yaitu 4, yang berarti memiliki kecepatan yang sama dalam penyebaran informasi.

Properti Average Degree

Average degree menunjukkan jumlah rata-rata hubungan yang dimiliki aktor dalam jaringan sosial. Semakin tinggi nilai *average degree* semakin baik dikarenakan setiap aktor dalam jaringan tersebut memiliki banyak

hubungan sehingga akan memperluas penyebaran informasi dengan cepat dan mudah. Dari kedua *brand* jasa transportasi *online* tersebut yang mendapatkan peringkat pertama untuk nilai *average degree* adalah Grab dengan nilai 5.387 dan Gojek yang memiliki nilai *average degree* 5.04.

Properti Average Path Length

Average path length adalah jarak rata-rata antar *node*. Semakin sedikit aktor yang dilewati maka semakin baik karena berarti jaringan sosial tersebut memiliki hubungan yang kuat. Maka untuk nilai *average path length* grab mendapat peringkat pertama dengan nilai 3.374 sedangkan gojek yang memiliki nilai 3.407.

Properti Clustering Coefficient

Clustering coefficient menunjukkan bagaimana suatu *node* berhubungan dengan *node* lain di sekitarnya. Apabila aktor dalam jaringan saling mengenal dan terhubung satu sama lain, berarti informasi yang disebarkan akan lebih cepat diketahui. Grab maupun gojek sama-sama memiliki nilai *clustering coefficient* 0, yang berarti dalam jaringan tersebut tidak ada akun yang saling mengenal satu sama lain.

Strategi Meningkatkan Jaringan pada Transportasi Online

Strategi yang harus dilakukan Grab dan Gojek dari hasil perbandingan peringkat kedua *brand* jasa transportasi *online* ber-

dasarkan properti jaringannya dengan mempertahankan dan meningkatkan peringkatnya berdasarkan tinggi rendahnya nilai dari masing-masing properti pada jaringan *brand* jasa transportasi *online* tersebut. *Brand* jasa transportasi *online* baik grab dan gojek harus lebih aktif dalam mengirim *tweet* dengan konten yang kekinian, dan menarik perhatian pengguna media sosial Twitter terkait masing-masing *brand*, sehingga akan menimbulkan banyak interaksi untuk meningkatkan nilai dari properti jaringan *size*. Selain itu akan terbentuk hubungan yang lebih banyak dan padat didalam jaringan tersebut.

Mengadakan *event* atau acara tertentu yang mengharuskan syarat peserta untuk mengajak orang lain dengan mewajibkan membuat *tweet* ajakin dan *mention* untuk mengajak bergabung ke dalam event tersebut untuk meningkatkan nilai dari properti jaringan *density*.

Perusahaan dapat melakukan kampanye dengan topik tertentu agar kelompok-kelompok yang terbentuk dalam jaringan semakin kecil dan penyebaran informasi antar aktor pada kelompok di dalam jaringan bisa semakin cepat untuk meningkatkan nilai properti *modularity* dan *clustering coefficient*.

Perusahaan dapat melakukan kerja sama dengan akun populer seperti selebgram atau berkolaborasi dengan perusahaan lain untuk saling mengikuti satu sama lain. Sehingga jarak yang terbentuk akan semakin pendek untuk meningkatkan nilai properti jaringan *diameter* dan *average path length*.

Hal ini akan memudahkan antar *node* saling berkomunikasi.

Menggunakan hashtag ‘#’ dalam penyebaran informasi melalui tweet pada Twitter. Sehingga penyebaran informasi akan lebih mudah diketahui dan cepat untuk meningkatkan properti jaringan *average degree*.

KESIMPULAN DAN SARAN

Berdasarkan hasil analisis terhadap kedua *brand* jasa transportasi *online* yaitu Grab dan Gojek, maka dapat disimpulkan bahwa dari hasil perhitungan dan analisis berdasarkan properti jaringan yaitu, *size*, *density*, *modularity*, *diameter*, *average degree*, *average path length*, dan *clustering coefficient*, didapat nilai untuk masing-masing jaringan *brand* jasa transportasi *online*.

Setelah dilakukan perbandingan, didapat peringkat berdasarkan properti jaringan, yaitu Grab unggul pada 4 properti jaringan seperti *size*, *modularity*, *average degree*, dan *average path length*. Sedangkan Gojek hanya unggul pada 1 properti jaringan yaitu *density*. Properti lainnya yaitu *diameter*, dan *clustering coefficient* jaringan Grab dan jaringan Gojek memiliki nilai yang sama. Perbandingan nilai yang didapat dari masing-masing properti jaringan kedua *brand* jasa transportasi *online* Grab dan Gojek mendapatkan hasil yang sama dengan penentuan peringkat *brand* menggunakan Top Brand

Award, yaitu Grab menempati peringkat 1 dan Gojek menempati peringkat 2.

Strategi yang bisa dilakukan oleh kedua *brand* jasa transportasi *online* yaitu Grab dan Gojek yang ingin mempertahankan atau meningkatkan peringkatnya berdasarkan properti jaringan yaitu mereka harus membuat konten yang menarik agar dapat menarik perhatian pengguna sehingga secara otomatis akan menyebabkan banyak interaksi didalamnya dan juga *brand* jasa transportasi *online* harus bekerja sama dengan akun populer untuk saling *follow* sehingga jaraknya akan semakin pendek dan penyebaran informasinya lebih cepat dan mudah. Hasil analisis dalam menentukan peringkat dengan menggunakan metode *social network analysis* pada kedua jasa transportasi *online* yaitu Grab dan gojek dapat dijadikan sebagai alternatif penentuan peringkat Top Brand dengan membandingkan dari setiap properti jaringan pada media sosial Twitter. Dua metode dalam menentukan peringkat *brand* dapat digunakan pada kondisi yang berbeda. Apabila perusahaan ingin menentukan peringkat *brand* dengan hasil yang lebih dalam dan detail serta memiliki waktu dan biaya yang cukup, maka perusahaan dapat menggunakan metode konvensional dengan cara menyebarkan kuesioner dan melakukan wawancara. Sedangkan apabila perusahaan ingin menentukan peringkat dengan waktu yang lebih efisien atau menginginkan hasil cepat dan *real time analytic* dan biaya yang rendah, maka perusahaan dapat menggunakan

metode *social network analysis*, dengan mengambil data berdasarkan kata kunci yang akan digunakan pada jejaring sosial Twitter.

Penelitian ke depan, perhitungan properti jaringan dapat dilakukan lebih dari satu. Selain itu yang dipergunakan tidak hanya dengan *software* Gephy. Hal ini agar dapat menjadi pembanding dan dapat mengukur keakuratan dari perhitungan properti jaringan. Selanjutnya penelitian selanjutnya dapat meneliti objek yang berbeda dengan menggunakan metode yang sama sebagai sarana memperkaya sumber pustaka dan pengetahuan yang dibutuhkan dalam penelitian terkait untuk metode *Social Network Analysis*.

DAFTAR PUSTAKA

- [1] A. P. Maharani Maulydia, "Comparison of User Experience on GoJek and Grab Mobile Apps (A study on PT Gojek and PT Grab Indonesia Consumers in DKI Jakarta)", Universitas Padjajaran, Bandung, 2017.
- [2] "Top Brand Award," [Online]. Available: <https://www.topbrandaward.com>. [Accessed 2 Maret 2019].
- [3] A. A. Nurshafa, "Analisis Peringkat Brand Pada Jejaring Sosial Percakapan Menggunakan Metode Social Network Analysis (Studi Kasus Brand Alfamart Dan Indomaret Pada Media Sosial Twitter Indonesia)", Universitas Telkom, Bandung, 2016.
- [4] V. N. A. Aini, "Analisis Pada Peringkat Top Brand Menggunakan Jejaring Sosial Percakapan Dengan Social Network Analysis (Studi Kasus Pada Smartphone Samsung, Blackberry, Nokia, Iphone Di Indonesia)", Universitas Telkom, Bandung, 2016.
- [5] A. E. P. Umaimah Wahid, "Upaya Peningkatkan Brand Awareness PT. Go-Jek Indonesia Melalui Aktivitas Marketing Public Relations", *Jurnal Komunikasi.*, vol. 9, no. 1, p. 31 – 43, Juli 2017.
- [6] K. S. Rifaldi, "Pengaruh Kualitas Pelayanan Transportasi Online Gojek Terhadap Kepuasan Pelanggan Pada Mahasiswa/I Administrasi Niaga Politeknik Negeri Jakarta", Politeknik Negeri Jakarta, Jakarta, 2016.
- [7] [Online]. Available: id.beritasatu.com. [Accessed 2019].
- [8] P. R. S. B. V. D. H. David Westerman, "Social media as information source: Recency of updates and credibility of information", *Journal of computer-mediated communication*, vol. 19, no. 2, pp. 171-183, 2014.
- [9] [Online]. Available: <https://support.twitter.com>. [Accessed 2019].
- [10] H. L. P. P. C. E. Suh B, " Want to be retweeted? large scale analytics on factors impacting retweet in twitter

- network", *IEEE Second International Conference on Social Computing*, pp. 177-184, 2010.
- [11] J. Scott, "Social Network Analysis," *Sociology*, vol. 22, no. 1, pp. 109-127, 1988.
- [12] A. Technologies, "Social Network Analysis Software - Cultural Domain Analysis Software", Analytic Technologies, P.O. Box 910359, Lexington, KY 40513 USA.
- [13] S. Y. David Knoke, "Social Network Analysis", SAGE Publications, Inc , University of Minnesota, Twin Cities Song Yang University of Arkansas , 2008.
- [14] G. Cheliotis, "Social Network Analysis (SNA) including a tutorial on concepts and methods", National University of Singapore, 2010.
- [15] A. J. P. V. M. O'malley, "The analysis of social networks" , *Health services and outcomes research methodology* 8, vol. 4 , pp. 222-269, 2008.
- [16] A.-L. Barabási, "The origin of bursts and heavy tails in human dynamics", *Nature*, vol. 435, pp. 207-211, 2005.
- [17] M. P. Albert-László Barabási, *Network Science*, Cambridge University, 2016.
- [18] M. R. Robert A Hanneman, "Introduction to social network methods", University of California, Riverside , 2005.
- [19] M. E. J. Newman, "Modularity and community structure in networks", in *Proceedings of the National Academy of Sciences*, USA, 2006.
- [20] M. E. J. Newman, "Network data" , Paul Dirac Collegiate Professor of Physics Department of Physics and Center for the Study of Complex Systems University of Michigan, 2013. [Online]. Available: <http://www-personal.umich.edu/~mejn/netdata/>. [Accessed March 2019].
- [21] G. Csardi, "inside-R" , Harvard Statistics Department, [Online]. Available: <http://www.inside-r.org/packages/cran/igraph/docs/average.path.length>. [Accessed 2019].

APLIKASI KOREKSI KESALAHAN PENULISANKATA DALAMBahasa INGGRIS DENGAN MENGGUNAKAN ALGORITMA RABIN- KARP

¹Sendy Agustian, ²Kenny, ³Kristien Margi Suryaningrum
^{1,2,3}Fakultas Teknologi dan Desain Universitas Bunda Mulia
^{1,2,3}Jl. Lodan Raya No. 2 Ancol, Jakarta Utara 14430
^{1,2,3}{agustianosendy, kenny220699, kristienmargi}@gmail.com,

Abstrak

Mengetik merupakan suatu kebiasaan untuk melakukan suatu kepentingan berupa tugas ataupun keperluan lainnya. Sementara itu, pastinya akan ada saat dimana ada sebuah kesalahan dalam pengetikan (typo). Jika ada suatu typo terkadang akan membuat kata-kata itu menjadi ambigu, tidak benar ataupun menjadi acak. Untuk penulisan karya ilmiah atau skripsi, kesalahan pada penulisan kata mungkin sudah menjadi hal yang biasa, tetapi untuk melakukan koreksi secara manual akan memakan waktu yang lama. Maka untuk menyelesaikan masalah yang terjadi salah satunya dapat dibantu oleh bantuan program, dengan membuat suatu aplikasi yang dapat digunakan oleh semua orang sebagai alat bantu untuk memeriksa kesalahan dalam pengetikan dengan menggunakan algoritma Rabin-Karp. Cara kerja algoritma Rabin-Karp adalah dengan melakukan pencocokan string berdasarkan masing-masing nilai hash pada teks dan pattern. Pattern didapat dengan melakukan pencarian kata yang tersimpan di dalam database berdasarkan kata terdekat dari typo yang ditemukan. Berdasarkan hasil perhitungan yang telah dilakukan secara manual dan program, diperoleh hasil bahwa kata yang merupakan kata singkatan akan menghasilkan hasil dan nilai yang berbeda daripada kata yang bukan merupakan kata singkatan. Hasil untuk kata kedua dari kata singkatan akan menghasilkan nilai yang berbeda dikarenakan kata tersebut menghasilkan kata yang berbeda atau tidak sama dengan yang seharusnya.

Kata Kunci: *Algoritma Rabin-Karp, Hash, koreksi, String Matching*

Abstract

Typing is a habit to do something important such as homework or other tasks. Meanwhile, there will be definitely an error occurred while in typing (typo). If there is a typo it will make the words become ambiguous, incorrect or random. For scientific papers or thesis, errors in writing maybe have become common, but to do correction manually will take a long time .To overcome the problem one of the way can be assisted by program, by making an application that can be used by everyone as a tool to check errors in typing using Rabin-Karp algorithm. The way the Rabin-Karp algorithm works is by matching strings based on each hash value in the text and pattern. Pattern can be found by searcng word listed in database based on the closest word from the typo that was found. Based on the test results of calculation that have been done by manually and program, the results are obtained that the word which is an abbreviation word will make different results from words that are not an abbreviation word. The results for the second word of the abbreviation will make a different value because the word produce a different word or not the same as it should.

Keywords: *Correction, Hash, Rabin-Karp Algorithm, String Matching*

PENDAHULUAN

Mengetik merupakan suatu kebiasaan untuk melakukan suatu kepentingan berupa tugas seperti laporan ilmiah atau skripsi ataupun keperluan lainnya seperti *chatting* atau berkomentar dalam media sosial. Terutama bagi kaum muda sudah sewajarnya jika setiap hari kita melakukan pengetikan baik itu melalui *smartphone* atau komputer.

Sementara itu, pastinya akan ada saat dimana ada sebuah kesalahan dalam pengetikan (*typo*). Jika ada suatu *typo* terkadang akan membuat kata-kata itu menjadi ambigu, tidak benar ataupun menjadi acak. Kesalahan pada penulisan kata mungkin sudah menjadi hal yang biasa untuk penulisan karya ilmiah atau skripsi, tetapi untuk melakukan koreksi secara *manual* akan memakan waktu yang lama.

Aplikasi koreksi kesalahan kata dapat membantu untuk menemukan letak kesalahan dan memberi sugesti dengan menggunakan algoritma *Rabin-Karp*. Algoritma ini merupakan salah satu algoritma *string matching* dengan melakukan pencocokan *string* berdasarkan masing-masing nilai *hash* pada teks dan *pattern*.

Penelitian mengenai aplikasi koreksi kesalahan berbasis pada tulisan berbahasa Indonesia untuk meningkatkan kualitas penulisan karya ilmiah menggunakan metode *N-Gram*. Aplikasi koreksi dari penelitian ini selain melakukan pengecekan kata yang sesuai dengan EyD juga dapat melakukan

pengecekan penggunaan tanda baca yang tidak sesuai dalam sebuah kalimat yang ada pada dokumen yang diuji. Dari proses pengecekan penggunaan tanda baca, sistem dapat melakukan koreksi secara otomatis terhadap kalimat-kalimat yang tidak sesuai tanda bacanya. Hasil dari penelitian tersebut adalah sebuah aplikasi koreksi yang dapat mendeteksi kesalahan-kesalahan yang terjadi pada dokumen-dokumen Bahasa Indonesia dan dapat melakukan perbaikan secara otomatis terhadap kata dan kalimat yang tidak sesuai dengan EyD [1].

Penelitian selanjutnya menggunakan metode *Levenshtein Distance* dan *N-Gram* untuk identifikasi kesalahan penulisan kata (*Typographical Error*) pada dokumen berbahasa Indonesia. Berdasarkan hasil pengujian dan analisis pada identifikasi *typographical error* pada dokumen bahasa Indonesia menggunakan metode *N-gram* dan *Levenshtein Distance* dapat diambil kesimpulan bahwa metode pendekatan *Dictionary Lookup* pada proses identifikasi *typographical error* pada dokumen bahasa Indonesia dapat diterapkan dengan baik untuk mencari kata *typographical error* dalam dokumen data input. Untuk menentukan kandidat kata, metode *Levenshtein Distance* dapat menghasilkan kandidat kata yang sesuai dengan nilai aktual yang diharapkan *user*. Namun untuk kata *typographical error* tertentu, jumlah kandidat kata yang ditampilkan dalam sistem terlalu banyak. Hasil presisi dan *recall* pada penelitian ini memiliki

nilai yang beragam pada setiap skenario pengujian. Nilai presisi terbaik yang dihasilkan sistem sebesar 0.97 pada skenario pengujian *typographicalerror* jenis *insertion*. Sementara itu, nilai *recall* terbaik yang dihasilkan sistem sebesar 1 pada skenario pengujian *typographicalerror* jenis *substitution* [2]. Penelitian lain mengenai implementasi algoritma *Rabin-Karp* pada pendeteksian pengulangan kata kerja dalam bahasa Inggris. Hasil dari penelitian tersebut yaitu suatu aplikasi yang dirancang telah mampu memberikan solusi dengan mengimplementasikan algoritma *Rabin Karp StringMatching* untuk melakukan pencarian *string* dan dapat menambahkan kosa kata yang baru pada aplikasi[3].

Algoritma *Rabin-Karp* merupakan salah satu algoritma *String Matching* dengan membandingkan nilai *hash*. Terdapat rumus untuk mencari nilai *hash* yang ditunjukkan pada persamaan (1) [4].

$$H = (c_1 * b^{k-1} + c_2 * b^{k-2} + c_3 * b^{k-3} + \dots + c_n * b^k) \quad (1)$$

H = nilai *hash*
 c = nilai *ascii* karakter
 k = banyak karakter
 b = basis (bilangan prima)

Text mining merupakan proses pencarian pola atau penggalian informasi dari data teks untuk menghasilkan informasi baru. Tujuan dari *text*

mining adalah menemukan informasi yang penting dari teks dengan meng-ubah teks menjadi data yang dapat digunakan untuk analisis yang lebih lanjut.

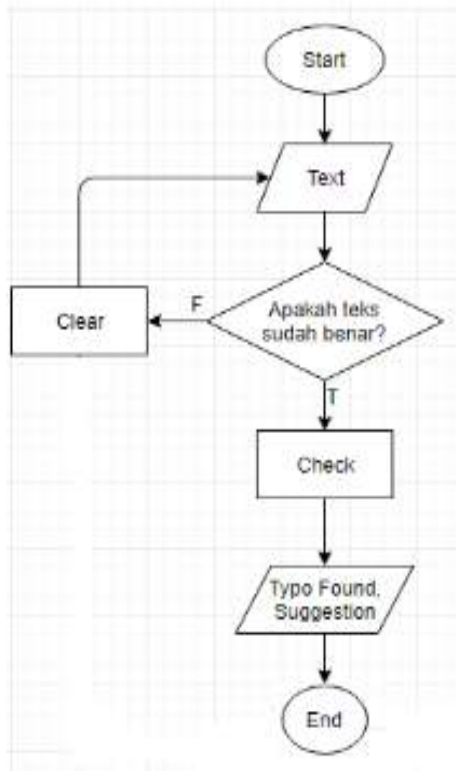
Tahapan awal yang dilakukan pada *text mining* yaitu *preprocessing* diantaranya *case folding*, *tokenizing*, *filtering*, dan *stemming* [5]

METODE PENELITIAN

Perancangan Proses

Cara kerja sistem yang ditunjukkan pada Gambar 1 adalah *user* menjalankan aplikasi koreksi kesalahan kata, *user* memasukkan teks yang ingin diperiksa, jika *user* salah memasukkan teks, maka tekan *clear* untuk menghapus teks yang sudah dimasukkan, tetapi jika teks yang dimasukkan *user* sudah benar, maka tekan *check* untuk memeriksa apakah ada kesalahan pada pengetikkan yang dilakukan *user*.

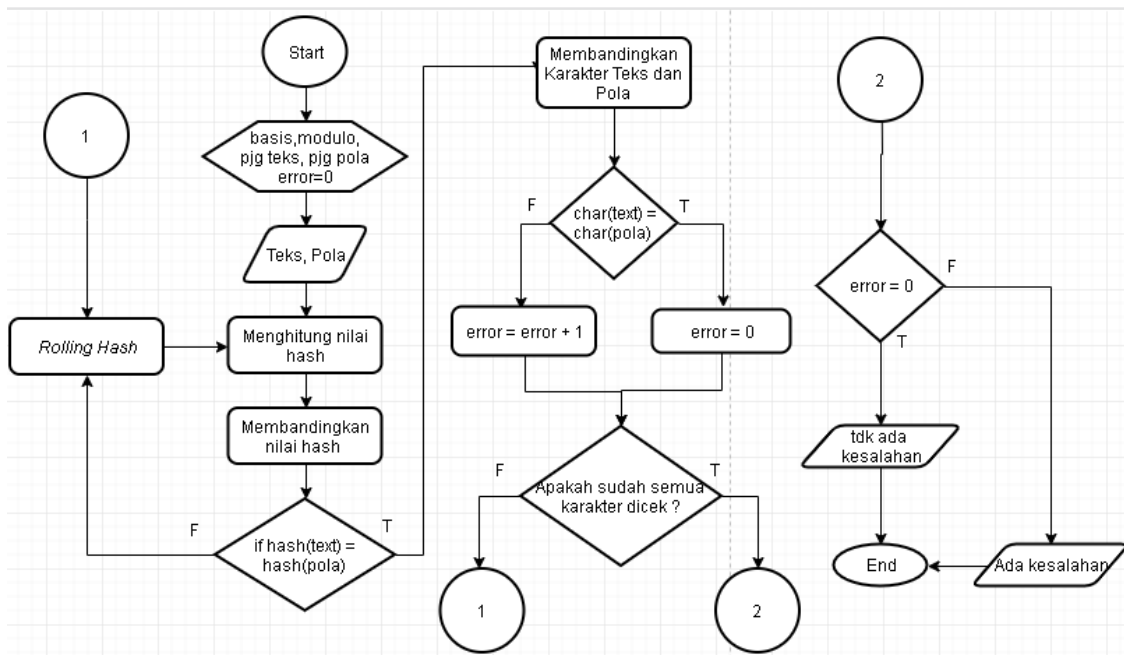
Hasil yang dikeluarkan oleh sistem jika menemukan *typo* adalah memberikan *output* berupa *typo found* dan *suggestion*. Jika *user* ingin melakukan pemeriksaan ulang atau pemeriksaan terhadap teks lain, maka *user* harus menekan tombol *clear*, tetapi jika *user* tidak ingin melakukan pemeriksaan ulang atau pemeriksaan terhadap teks lain, maka *user* dapat menekan tombol *exit* untuk keluar dari aplikasi.



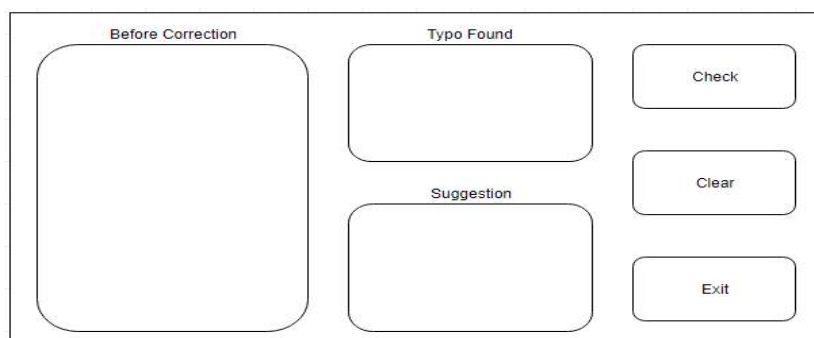
Gambar 1. Flowchart Aplikasi

Cara kerja dari algoritma Rabin-Karp di tunjukkan pada Gambar 2. yaitu: (1) Inisialisasi basis, modulo, p₁g teks, p₁g pola, error. (2) Melakukan inputan teks dan juga pola, pola didapatkan dengan pencarian kata dari database yang mirip dengan inputan teks (3) Teks akan dipotong menjadi beberapa substring berdasarkan panjang maksimal dari pola dan akan di cari nilai hash dari teks dan pola dan akan dilakukan perbandingan. (4) Jika sudah didapatkan nilai hash yang sama maka akan dilakukan perbandingan lagi

terhadap substring teks dan pola. Jika karakter tidak sama maka error akan bertambah satu, jika sama maka error sama dengan 0. (5) Lalu akan dilakukan dengan pengecekan semua karakter, jika false maka akan menuju on-page connector nomor satu, dan akan melakukan rolling hash. Jika true maka akan menuju on-page connector nomor dua. (6) Setelah itu akan dilakukan pengecekan terhadap error jika tidak sama dengan 0 maka akan dicetak ada kesalahan, jika sama dengan 0 maka akan dicetak tidak ada kesalahan



Gambar 2. Flowchart Algoritma Rabin-Karp

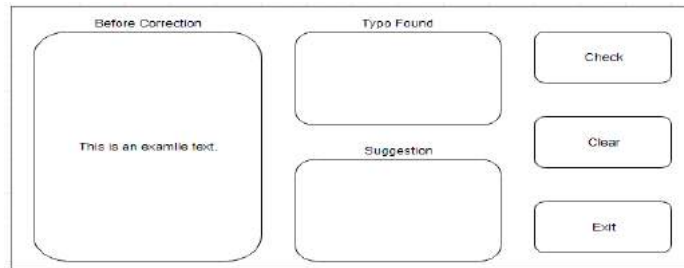


Gambar 3. Perancangan Tampilan Koreksi Kata

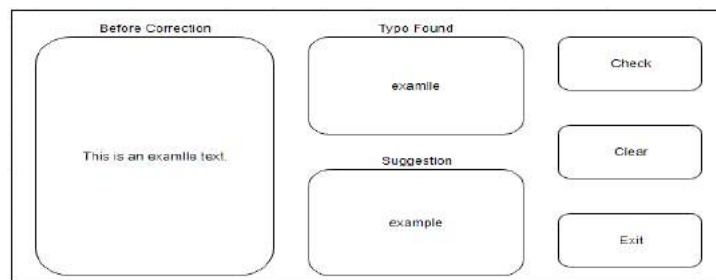
Perancangan tampilan

Berikut merupakan penjelasan yang di tunjukkan pada Gambar 3 (1) *Before Correction*: merupakan tempat dimana *user* menuliskan teks (2) *Typo Found*: merupakan *output* jika terdapat *typo* pada teks yang dituliskan oleh *user* (3) *Suggestion* merupakan *output* yang diberikan sistem untuk memberikan saran untuk mengganti kata yang salah. (4) *Check* merupakan tombol untuk

user memeriksa teks yang dituliskannya. (5) *Clear* merupakan tombol untuk menghapus teks yang dituliskan *user* jika salah memasukkan teks, dan *Exit* merupakan tombol untuk keluar dari aplikasi jika *user* sudah tidak mau menggunakan aplikasi. Perancangan tampilan input di tunjukkan pada Gambar 4. *User* dapat memasukkan sebuah teks yang nanti nya akan di cek pada bagian input teks *Before Correction*.



Gambar 4. Perancangan Tampilan *Input*



Gambar 5. Perancangan Tampilan *Output*

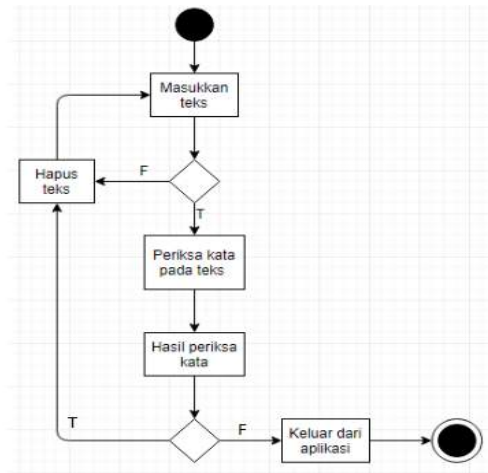
Perancangan tampilan output di tunjukkan pada Gambar 5 merupakan hasil setelah tombol *Check* ditekan, Sistem akan menampilkan letak kesalahan kata pada bagian *Typo Found*, dan akan menampilkan saran pada bagian *Suggestion*.

Perancangan Sistem

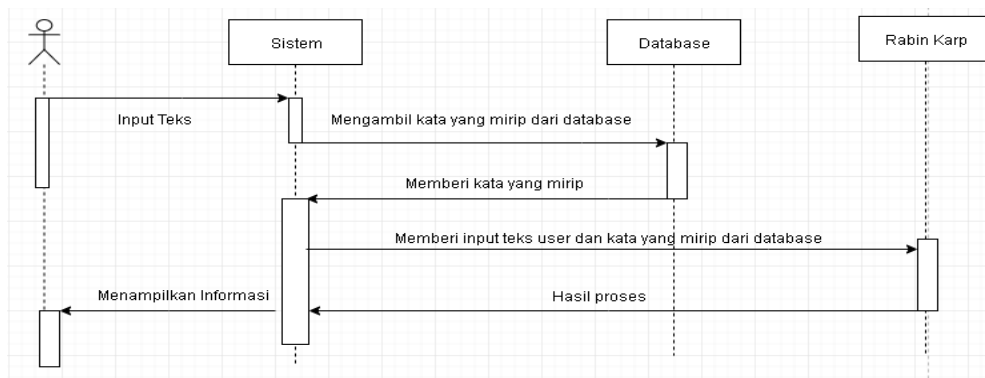
Activity diagram tunjukkan pada Gambar 6. *User* memasukkan teks, *user* dapat menghapus teks jika salah, jika teks sudah benar, sistem akan memeriksa kata yang terdapat pada teks. Sistem akan memberikan hasil periksa kata, jika terdapat kesalahan maka sistem akan memberitahukan letak kesalahan dan saran untuk menggantikan kata yang salah tersebut. Jika *user* ingin mem-

eriksa teks yang lain, maka *user* harus menghapus teks untuk memasukkan teks yang lain, tetapi jika *user* sudah tidak mau memeriksa teks yang lain, maka *user* dapat keluar dari aplikasi.

Sequence diagram di tunjukkan pada Gambar 7. *User* memasukkan teks, kemudian algoritma *Rabin-Karp* melakukan pengambilan kata pada *database* yang ada pada sistem. Sistem melakukan pencocokan kata dengan algoritma, kemudian algoritma memberikan kata kerja yang sama pada sistem. Sistem menampilkan letak kesalahan kata sekaligus saran kata pengganti kepada *user*, kemudian *user* dapat memilih kata pengganti sesuai dengan yang disarankan sistem.



Gambar 6. Activity diagram



Gambar 7. Sequence diagram

Algoritma

Terdapat fungsi algoritma *Rabin-Karp* di tunjukkan pada Gambar 8. Algoritma *Rabin-Karp* digunakan untuk mencocokkan nilai *hash* dari teks dan pola. Cara kerja dari algoritma *Rabin-Karp* yaitu mencari nilai *hash* suatu pola terlebih dahulu setelah itu

mencari *N-Gram* dengan cara mengurangi panjang karakter dari teks dan pola. Lalu mencocokkan nilai *hash*, jika nilai sama maka akan di bandingkan lagi terhadap karakter pola dan teks tersebut, jika tidak

sama maka akan mengulangi *looping* sampai selesai, jika karakter sama maka proses akan memberi tahu letak posisi indeks yang sudah di-temukan. Fungsi *hash* dapat ditunjukkan pada Gambar 9.

Fungsi *hash* digunakan untuk mencari nilai *hash* dari suatu kata, dari mencari panjang maksimal suatu *array* dan maksimal kata yang terdapat pada *array* tersebut dan dilakukan perhitungan terhadap masing-masing karakter dengan mencari nilai ASCII dan diproses dengan rumus *hash*.

```

Function RabinKarp(input pat: string[1..m], txt: string[1..n])
h_pat <- FungsiHash(pat[1..m])
for i <- 0 to n-m do
  h_txt <- FungsiHash(txt[i..i+m-1])
  if h_pat = h_txt then
    for j <- 0 to m do
      if txt[0].charAt(i+j) != pat[0].charAt(j) then
        break
      Endif
    Endfor
    if j = m then
      write("Pattern found at index")
      write(i)
    Endif
  Endif
Endfor

```

Gambar 8. Algoritma Rabin-Karp

```

Function FungsiHash(input txt: string[1..m])
for i <- 0 to maxarray do
  for j <- 0 to maxchar do
    maxchar <- txt[i].length()
    nilai_ascii <- txt[0].charAt(j)
    h_temp <- nilai_ascii * Math.pow(q, maxchar + (j-1)) + h_temp
  Endfor
Endfor
h_temp <- h_temp % modulo
return h_temp

```

Gambar 9. Fungsi Proses Nilai Hash

HASIL DAN PEMBAHASAN

Terdapat contoh kalimat yang dapat diuraikan sesuai dengan masing-masing kata yang terdapat pada kalimat tersebut untuk membandingkan perhitungan nilai hash secara manual dan program.

Contoh kalimat :

“Finally my exam is over, tomorrow we’ll go to some places I’ve never been before, with my family. Yes, it’s holiday. I’m going to sleep

earlier tonight!” Setelah disusun ke dalam Tabel sesuai dengan masing-masing kata maka didapat hasil seperti pada Tabel 1.

Perhitungan Secara Mnual

Perhitungan nilai hash diawali dari indeks pertama sampai indeks terakhir. Dengan nilai basis 3 dan q 107. Berikut merupakan contoh perhitungan hash daritek “finally”. Setelah dilakukan perhitungan sampai indeks akhir secara manual maka didapat hasil yang ditunjukkan pada Tabel 2.

Perhitungan dengan Program

Setelah dilakukan perhitungan sampai indeks akhir dengan bantuan program maka

didapat hasil seperti kata dan nilai *hash* disampingnya yang dapat di tunjukkan pada Gambar 10.

Tabel 1. Daftar Kata-kata

Kata - kata	
<i>Finally</i>	<i>Before</i>
<i>My</i>	<i>With</i>
<i>Exam</i>	<i>My</i>
<i>Is</i>	<i>With</i>
<i>Over</i>	<i>My</i>
<i>Tomorrow</i>	<i>Family</i>
<i>We</i>	<i>Yes</i>
<i>Will</i>	<i>It</i>
<i>Go</i>	<i>Is</i>
<i>To</i>	<i>Holiday</i>
<i>Some</i>	<i>I</i>
<i>Places</i>	<i>Am</i>
<i>I</i>	<i>Going</i>
<i>Have</i>	<i>To</i>
<i>Never</i>	<i>Sleep</i>
<i>Been</i>	<i>Earlier</i>
	<i>Tonight</i>

Tabel 2. Nilai Hash Kata

Kata	Hash	Kata	hash
<i>Finally</i>	41	<i>Before</i>	25
<i>My</i>	20	<i>With</i>	9
<i>Exam</i>	34	<i>My</i>	20
<i>Is</i>	2	<i>Family</i>	61
<i>Over</i>	89	<i>Yes</i>	9
<i>Tomorrow</i>	38	<i>It</i>	3
<i>We</i>	30	<i>Is</i>	2
<i>Will</i>	96	<i>Holiday</i>	17
<i>Go</i>	99	<i>I</i>	105
<i>To</i>	31	<i>Am</i>	79
<i>Some</i>	38	<i>Going</i>	92
<i>Places</i>	88	<i>To</i>	31
<i>I</i>	105	<i>Sleep</i>	73
<i>Have</i>	70	<i>Earlier</i>	74
<i>Never</i>	62	<i>Tonight</i>	89
<i>Been</i>	9		

```

finally 41.0
my 20.0   before 25.0
exam 34.0   with 9.0
is 2.0     my 20.0
over 89.0   family 61.0
tomorrow 38.0 yes 9.0
we 30.0    it 3.0
ll 4.0     s 8.0
go 99.0    holiday 17.0
to 31.0    i 105.0
some 38.0  m 2.0
places 88.0 going 92.0
i 105.0    to 31.0
ve 27.0    sleep 73.0
never 62.0 earlier 74.0
been 9.0   tonight 89.0
    
```

Gambar 10. Perhitungan Hash dengan Program

Tabel 3. Analisa Hasil

Kata asal	Kata Hasil	Manual	Program
<i>We'll</i>	<i>We ll</i>	30 & 96	30 & 4
<i>I've</i>	<i>I ve</i>	105 & 70	105 & 27
<i>It's</i>	<i>It s</i>	3 & 2	3 & 8
<i>I'm</i>	<i>I m</i>	105 & 79	105 & 2

Analisis Hasil

Setelah menghitung nilai *hash* secara *manual* dan program terdapat beberapa hasil yang tidak sama yang dapat ditunjukkan pada Tabel 3.

$$\begin{aligned}
 \text{hash}(\textit{finally}) &= ((102 * 3^6) + (105 * 3^5) + (110 * 3^4) + (97 * 3^3) \\
 &\quad + (108 * 3^2) + (108 * 3^1) + (121 * 3^0)) \textit{mod}_{107} \\
 &= (74,358) + (25,515) + (8,910) \\
 &\quad + (2,619) + (972) + (324) + (121) \\
 &= 112,819 \textit{mod}_{107} \\
 &= 41
 \end{aligned}$$

Terdapat beberapa nilai *hash* yang tidak sama berdasarkan analisa hasil Tabel 3 dapat di ketahui jika nilai *hash* dari kata asal dan kata hasil berbeda karena kata tersebut merupakan kata singkatan seperti “*we'll*” yang merupakan kata dari “*we will*” dan lain sebagainya.

KESIMPULAN DAN SARAN

Berdasarkan hasil perhitungan yang telah dilakukan secara *manual* dan program, diperoleh hasil bahwa kata yang merupakan kata singkatan akan menghasilkan hasil dan

nilai yang berbeda daripada kata yang bukan merupakan kata singkatan. Hasil untuk kata kedua dari kata singkatan akan menghasilkan nilai yang berbeda dikarenakan kata tersebut menghasilkan kata yang berbeda atau tidak sama dengan yang seharusnya.

Adapun beberapa saran yang dapat disampaikan untuk dapat mengembangkan penelitian ini yaitu dapat dikembangkan dengan menerapkan *tenses* bahasa Inggris dan sistem aplikasi yang dirancang dapat memberikan *interface* yang lebih sederhana dan menarik.

DAFTAR PUSTAKA

- [1] Andri, S. Ariana dan M. Andriani, “Aplikasi koreksi kesalahan berbasis pada tulisan berbahasa Indonesia untuk meningkatkan kualitas penulisan karya ilmiah”, Dalam Prosiding Seminar Nasional Aplikasi Sains & Teknologi (SNAST), 2014, hal. A.169 – A.172.
- [2] A. I. Fahma, I. Cholissodin, dan R. S. Perdana. “Identifikasi kesalahan penulisan kata (*Typographical Error*) pada dokumen berbahasa Indonesia menggunakan metode *N-gram* dan *Levenshtein Distance*”, *Jurnal Pengembangan Teknologi Informasi*

- dan Ilmu Komputer*, vol. 2, no.1, hal. 53 – 62, 2017.
- [3] H. P. F. A. Sinaga, “Implementasi algoritma Rabin KARP pada pendeteksian pengulangan kata kerja dalam penulisan bahasa Inggris”, Skripsi Sarjana, Universitas Sumatera Utara, Medan, 2018.
- [4] Herryance, Handrizal, dan S. D. Faradilla, “Analisis algoritma Rabin-Karp pada kamus umum berbasis Android”, *Jurnal Riset Sistem Informasi dan Teknik Informatika*, vol. 2, no.1, hal. 64 – 74, 2017.
- [5] D. A. Putra dan H. Sujaini, “Implementasi algoritma Rabin-Karp untuk membantu pendeteksian plagiat pada karya ilmiah”, *Jurnal Sistem dan Teknologi Informasi*, vol. 1, no. 1, hal. 1 – 9, 2015.

PREDIKSI CURAH HUJAN DI JAKARTA BERBASIS ALGORITMA LEVENBERG MARQUARDT

¹Gema Indah Merdekawati, ²Ismail

^{1,2}Fakultas Teknologi Industri Universitas Gunadarma
Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat
²ismail_muchsin@staff.gunadarma.ac.id,

Abstrak

Perubahan pola curah hujan yang tidak menentu sangat berpengaruh terhadap berbagai aspek kehidupan terutama di kota Jakarta dimana segala aktivitas penting berada di dalamnya, sehingga perlu dilakukan prediksi curah hujan agar tidak mengganggu aktifitas penting dan harus segera dilaksanakan. Penelitian ini akan melakukan prediksi curah hujan di Jakarta berbasis algoritma levenberg marquardt menggunakan data curah hujan harian mulai dari 1 Mei 2016 – 30 April 2018 dari stasiun meteorology Kemayoran. Penelitian ini terdiri dari beberapa tahap, yakni pengolahan data curah hujan harian, normalisasi data curah hujan, konversi data curah hujan yang telah di normalisasi, proses training jaringan, dan hasil akurasi prediksi. Prediksi curah hujan di Jakarta menggunakan pola 23-14-1 dan dipengaruhi oleh 1 parameter yaitu hidden layer. Proses training terhadap jaringan dilakukan sebanyak 5 kali dengan merubah nilai parameternya untuk memperoleh hasil berupa akurasi terbaik, sehingga algoritma levenberg marquardt dapat digunakan pada prediksi curah hujan dimasa yang akan datang. Hasil yang diperoleh pada penelitian ini adalah algoritma levenberg marquardt dapat dilakukan untuk memprediksi curah hujan di kota Jakarta dengan nilai akurasi sebesar 96%, hal tersebut menunjukkan bahwa algoritma levenberg marquardt pada penelitian ini memiliki performance yang baik dan dapat digunakan untuk prediksi curah hujan kota Jakarta dimasa yang akan datang.

Kata Kunci: *Prediksi, curah hujan, Neural Network, Levenberg Marquardt*

Abstract

Unpredictable changes in rainfall patterns are very influential on various aspects of life, especially in Jakarta where most all important activities are in it, so it is necessary to predict rainfall in order not to disturb the activities. This research predicted rainfall in Jakarta by implementing Levenberg Marquardt algorithm and used daily rainfall data May 1, 2016 - April 30, 2018 from Kemayoran meteorological station. This research consisted of daily rainfall data processing, rainfall data normalization, normalized rainfall data conversion, network training processing, and prediction accuracy results. Prediction of rainfall in Jakarta used 23-14-1 pattern and influenced by 1 parameter, which is hidden layer. The networks training process was done 5 times by changing the parameter values in order to get the best accuracy results, so Levenberg Marquardt algorithm can be used to predict rainfall in the future. The results of this research indicated that Levenberg Marquardt algorithm could be used to predict rainfall in Jakarta with accuracy value up to 96%, and it also indicated that Levenberg Marquardt algorithm in this research had good performance and could be used to predict rainfall in Jakarta in the future.

Keywords: *Prediction, rainfall, Neural Network, Levenberg Marquardt.*

PENDAHULUAN

Indonesia memiliki tiga jenis iklim yang selalu mengalami perubahan. Perubahan iklim dapat diamati dari perubahan pola curah hujan rata-rata di beberapa wilayah [1]. Perubahan pola curah hujan yang tidak menentu mengakibatkan perubahan cuaca yang semakin memburuk dan berdampak pada beberapa sektor [2]. Salah satu lembaga yang mencatat perubahan iklim di Indonesia adalah Badan Meteorologi Klimatologi dan Geofisika (BMKG). Badan Meteorologi Klimatologi dan Geofisika (BMKG) merupakan lembaga pemerintahan non departemen yang menyediakan informasi penting, salah satunya adalah memprediksi curah hujan [3]. Dalam melakukan prediksi terhadap curah hujan selama ini BMKG menggunakan metode subyektif, yaitu prediksi yang dibuat berdasarkan pertimbangan atau penilaian prakirawan [4]. Pengamatan curah hujan yang dilakukan oleh BMKG dilakukan pada pagi dan sore hari dan dicatat pada kartu ujian. Beberapa penelitian dilakukan dalam melakukan prediksi hujan dilakukan peneliti terdahulu. Penelitian [5] melakukan prediksi curah hujan di wilayah Makassar menggunakan metode *wavelet-neural network* mencapai tingkat akurasi sebesar 63.83% pada tahun 2010. Indrabayu et al menggunakan parameter temperatur, kelembaban, kecepatan angin, tekanan udara, dan curah hujan dalam melakukan prediksinya. Penelitian [6] me-

lakukan prediksi cuaca berbasis logika *fuzzy* untuk rekomendasi penerbangan di bandar udara Raja Haji Fisabilillah dengan prediksi hujan menggunakan tiga masukkan yaitu suhu udara, kecepatan angin, dan kelembaban udara dengan *output* curah hujan. Nilai akurasi prediksi hujan adalah 61.73%. Penelitian [7] melakukan analisis algoritma prediksi curah hujan menggunakan algoritma *competitive neural network* dengan menggunakan beberapa skenario berupa bobot dan akurasi yang dapat digunakan untuk melakukan prediksi yang menghasilkan akurasi rata-rata sebesar 82.915%. Penelitian [8] melakukan prediksi curah hujan di kota

Medan menggunakan metode *back-propagation neural network* berdasarkan banyaknya hidden layer mendapatkan nilai akurasi tertinggi sebesar 43.27%. Penelitian [9] melakukan implementasi *back-propagation neural network* dalam prakiraan cuaca di daerah Bali Selatan dengan melakukan penambahan jumlah *neuron* pada lapisan tersembunyi yang mencapai tingkat akurasi prakiraan cuaca terbaik sebesar 51.6129%. Pada penelitian ini, peneliti akan mengimplementasikan penggunaan algoritma *levenberg marquardt* menggunakan data curah hujan periode 2016-2018 untuk melakukan prediksi hujan di Jakarta. Peneliti juga akan menghitung akurasi algoritma dalam menentukan prediksi curah hujan. Hasil penelitian ini diharapkan dapat meningkatkan hasil akurasi pada prediksi curah

hujan di Jakarta berbasis kecerdasan buatan (*Artificial Intelligence*).

METODE PENELITIAN

Pada metode penelitian terdiri atas beberapa tahapan proses. Tahapan awal dimulai dari mencari data curah hujan harian kota Jakarta dari 1 Mei 2016 – 30 April 2018. Data curah hujan ini digunakan sebagai parameter tunggal, dimana data yang digunakan disajikan dalam bentuk tabel yang dibagi menjadi empat kolom yaitu tanggal, bulan, tahun dan curah hujan per hari dalam satuan mm yang disimpan dalam format (.dot) txt. Kemudian pada tahap preprocessing, data curah hujan tersebut di normalisasi dengan mengasumsikan setiap bulan terdiri dari 31 hari yang dilanjutkan dengan melakukan konversi struktur data. Data yang telah dikonversi menjadi baris kemudian dilakukan *training neural network* yang membagi data menjadi 3 yaitu *data training*, *data validation*, dan *data testing*.

Data Curah Hujan

Pada penelitian ini, data curah hujan di dapat dari *website* BMKG, www.dataonline.bmkg.go.id pada kota Jakarta melalui stasiun cuaca Kemayoran. Data curah hujan (mm) per hari kota Jakarta yang digunakan dari 1 Mei 2016 – 30 April 2018 di mana data yang akan digunakan sebanyak 24 bulan dengan jumlah data yang diambil sebanyak 731 *record*. Data curah hujan yang diperoleh

terdiri dari beberapa parameter seperti suhu minimum (°C), suhu maksimum (°C), suhu rata-rata (°C), kelembapan rata-rata (%), curah hujan (mm), lama penyinaran (jam), kecepatan angin rata-rata (knot), arah angin terbanyak (deg), kecepatan angin terbesar (knot), dan arah angin saat kecepatan maksimum (deg), tetapi pada penelitian ini hanya menggunakan parameter curah hujan (mm) saja untuk menyesuaikan dengan penelitian yang akan dilakukan.

Data curah hujan yang digunakan pada penelitian ini diolah sesuai dengan kebutuhan penelitian dengan membagi menjadi empat kolom yaitu tanggal, bulan, tahun, dan curah hujan per hari dalam satuan mm. Contoh data curah hujan dapat dilihat pada Tabel 1. Data curah hujan pada Tabel 1 memiliki nilai curah hujan mm/day yang bervariasi.

Pada tanggal 4 bulan 5 terdapat nilai curah hujan sebesar 16.6 yang mana nilai tersebut menunjukkan bahwa pada hari tersebut hujan ringan karena kategori hujan ringan berada pada interval 5 mm sampai 20 mm per hari, sedangkan pada tanggal 10 bulan 5 terdapat nilai curah hujan sebesar 9999 dimana nilai tersebut bukan menunjukkan pada kategori hujan tetapi menunjukkan bahwa data intensitas curah hujan pada tanggal tersebut tidak ada. Tabel 2 merupakan kategori intensitas curah hujan yang digunakan pada penelitian sebagai acuan penentuan cuaca harian.

Tabel 1 Data Curah Hujan 1 Mei – 20 Mei 2016

<i>Date</i>	<i>Month</i>	<i>Year</i>	<i>Rainfall (nun/day)</i>
1	5	2015	0
2	5	2015	0
3	5	2015	0
4	5	2015	16.6
5	5	2015	9999
6	5	2015	0
7	5	2015	0
8	5	2015	0
9	5	2015	0
10	5	2015	9999

Tabel 2. Kategori Hujan

Intensitas Curah Hujan Harian (mm)	Keterangan
Curah Hujan < 5 mm	Cerah
Curah Hujan ≥ 5 mm	Hujan

Preprocessing

Pada tahap *preprocessing* data curah hujan yang sudah diolah dilanjutkan untuk proses normalisasi data. Hasil dari data yang telah di normalisasi kemudian akan digunakan untuk tahap berikutnya dalam penelitian yaitu tahap konversi dan *training* data. Data yang di dapat dari *websitewww.dataonline.bmkg.go.id* masih memiliki data yang rancu seperti terdapat nilai curah hujan per hari dalam satuan mm sebesar 8888 atau 9999. Data 8888 berarti data tidak terukur dan nilai 9999 berarti tidak ada data, data tersebut diganti menjadi 0.

Selain data diatas, terdapat pula perbedaan hari dalam setiap bulannya dalam satu tahun. Seperti bulan Januari yang memiliki 31 hari, bulan Februari 28 hari, bulan Juni 30 hari, dan terdapat bulan kabisat yang memiliki 29 hari pada bulan Februari 2016. Hari yang paling banyak dalam satu bulan adalah 31 hari sehingga dilakukan normalisasi

pada bulan yang memiliki 28, 29, dan 30 hari. Setiap bulan tersebut di-asumsikan menjadi 31 hari dimana hari yang di tambah diberikan nilai curah hujan (mm) 0 atau hari cerah.

Perhitungan Prediksi

Pada tahap ini, disiapkan variabel untuk *input* data dan target data lalu dilakukan setting untuk *hidden layer* sebanyak 14.

Dilakukan pembagian data untuk data *training*, data *validation*, dan data *testing* sebanyak 80%, 10%, dan 10%, kemudian dilakukan setting algoritma yang akan digunakan yaitu algoritma *levenberg marquardt* dengan perhitungan dengan langkah sebagai berikut :

1) Pembentukan Data Set Neural Network.

Pembentukan data set *neural network* merupakan tahapan sebelum dilakukannya *training*. Dalam pembentukan data set *neural*

network ini menggunakan data yang telah dinormalisasi sebanyak 744 *record* setelah data sebelumnya sebanyak 731 *record*.

Data masih dalam bentuk baris yang berisi bulan, tanggal, tahun, dan curah hujan perhari kemudian dikonversi menjadi baris yang merepresentasikan bulan dan 31 kolom hari yang berisi nilai curah hujan per hari dalam satuan mm.

2) Pembentukan *Training Set*, *Validation Set*, dan *Testing Set*.

Sebelum pembentukan *training set*, *validation set*, dan *testing set* dilakukan penentuan jumlah *input layer*, *hidden layer*, dan *output layer* yang akan digunakan pada penelitian ini.

Penentuan jumlah *input layer* dan *hidden layer* yang tepat dapat menghasilkan pengenalan pola yang baik. Seperti pada Gambar 1, arsitektur jaringan syaraf tiruan yang digunakan pada penelitian ini adalah arsitektur multi *layernetwork* dimana terdapat *input layer*, *hidden layer*, dan *output layer* pada prosesnya.

Terdapat 23 unit pada *input layer* dimana semua unit pada *layer* ini harus melalui *hidden layer* yang memiliki 14 unit untuk mengirimkan nilai sebelum akhirnya sampai pada unit *output*. Banyaknya unit pada lapisan *input* telah ditentukan sebelumnya

yang mana 23 unit tersebut adalah bulan ke 1 sampai bulan ke 23 yang akan diolah untuk mendapatkan hasil prediksi nantinya, sedangkan unit-unit pada *hidden layer* yang akan merespon *input layer* dan akan meneruskan kepada *output layer*. Pada *neural network* biasanya digunakan *training set* untuk membentuk sebuah model *classifier* yang nantinya akan digunakan untuk memprediksi data. *Testing set* digunakan untuk mengukur keberhasilan klasifikasi.

Data *testing set* harus berbeda dengan data *training set* agar memiliki kemampuan generalisasi yang baik dalam klasifikasi data. *Validation set* digunakan untuk mencari parameter (*hidden layer*) yang paling baik untuk sebuah algoritma klasifikasi sekaligus mencegah terjadinya *over fitting* (banyak *error* acak). Untuk itu pada tahap ini data dibagi menjadi tiga bagian, yaitu data untuk *training set*, *validation set*, dan *testing set* dengan perintah seperti berikut :

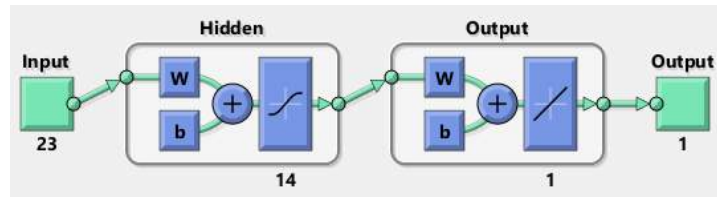
```
net.divideParam.trainRatio = 22/24;  
net.divideParam.valRatio = 1/24;  
net.divideParam.testRatio = 1/24;
```

Keterangan :

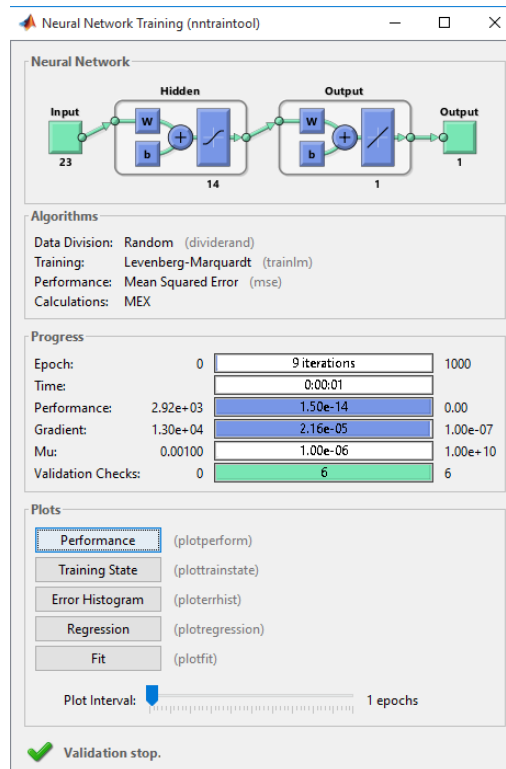
Data untuk *training set* sebesar 22/24 dari jumlah data

Data untuk *validation set* sebesar 1/24 dari jumlah data

Data untuk *testing set* sebesar 1/24 dari jumlah data



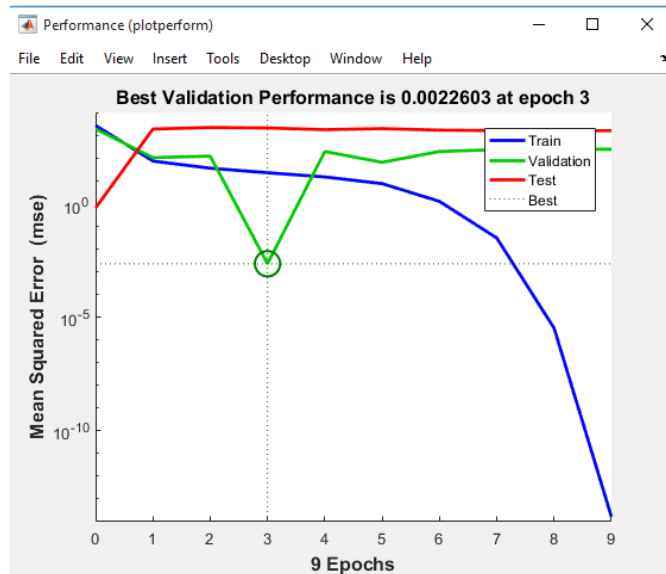
Gambar 1. Arsitektur Multi *Layer Network* Pada Prediksi Curah Hujan Kota Jakarta



Gambar 2. Jendela *Ntraintool*

Training atau pelatihan yang digunakan pada penelitian ini adalah *trainlm* (*levenberg marquardt*) seperti dapat dilihat Gambar 2. *Trainlm* adalah metode yang tepat untuk merancang jaringan syaraf tiruan, parameter yang perlu di set untuk metode ini ialah jumlah unit *hidden layer*-nya. Proses *training* dengan parameter tersebut menghasilkan tampilan proses jaringan terdiri dari arsitektur

neural network, algoritma, *progress*, dan plot yang muncul pada jendela *ntraintool* seperti Gambar 2. Proses *training* menggunakan *trainlm* (*levenberg marquardt*), nilai *epoch* 1000, memperoleh nilai *training mse* sebesar 1.5×10^{-14} , dengan waktu 1 detik, dan disajikan pula dalam bentuk grafik/*plot*.



Gambar 3. Grafik Performance

Parameter Perhitungan Performance atau Akurasi

Pada penelitian ini, Peneliti menggunakan tiga parameter pengukuran antara lain *recall*, *precision* dan *accuracy* seperti dapat dilihat pada persamaan 1, persamaan 2, dan persamaan 3.

$$precision = \frac{a}{x} \times 100\% \quad (1)$$

$$Recall = \frac{a}{y} \times 100\% \quad (2)$$

$$Accuracy = \frac{a+b}{z} \times 100\% \quad (3)$$

Dimana :

a = Data cerah yang dipisahkan dengan benar

b = Data cerah yang bukan cerah

x = Jumlah data cerah yang dipisahkan

y = Data cerah sebenarnya

z = Jumlah data total

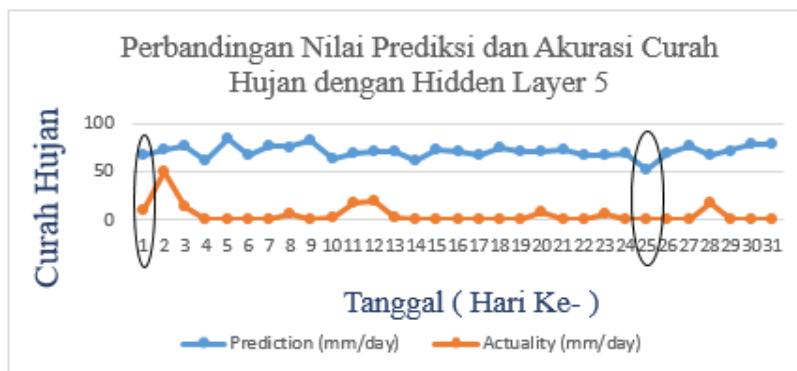
Hasil *training* untuk prediksi curah hujan kota Jakarta menggunakan algoritma *levenberg marquardt* seperti pada Gambar 3

yaitu tampilan grafik yang menunjukkan nilai mse untuk *performance* terbaik pada proses *validation*. Pada grafik tersebut terlihat bahwa mse yang dihasilkan adalah 0,0022603 dan nilai mse *validation* tersebut diperoleh pada *epoch* ke 3.

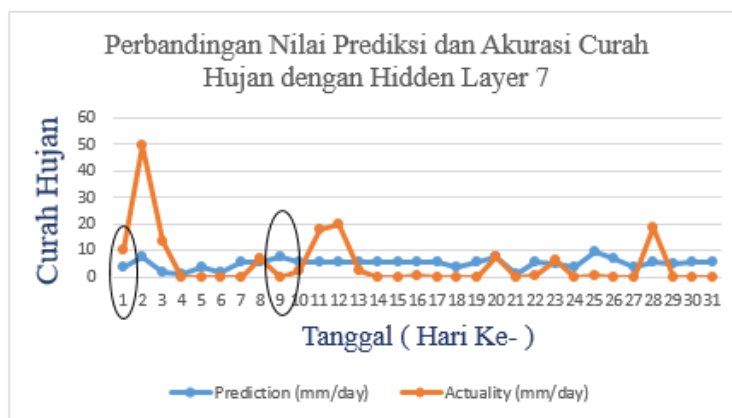
HASIL DAN PEMBAHASAN

Pada penelitian ini, dilakukan beberapa kaliproses *training* dengan melakukan perubahan pada parameter *hidden layer* sampai mendapatkan nilai akurasi yang optimal. Sebagai contoh, pada proses *training* pertama, digunakan nilai parameter *hidden layer* sebanyak 5 untuk dilakukan pengecekan apakah dengan nilai parameter tersebut dapat mencapai nilai akurasi yang optimal.

Pada Gambar 4 dapat diketahui perbandingan data hasil prediksi dan aktualnya yang diperoleh pada proses *training* pertama dengan nilai parameter tersebut.



Gambar 4. Grafik Perbandingan Hasil Prediksi dengan 5 *HiddenLayer*



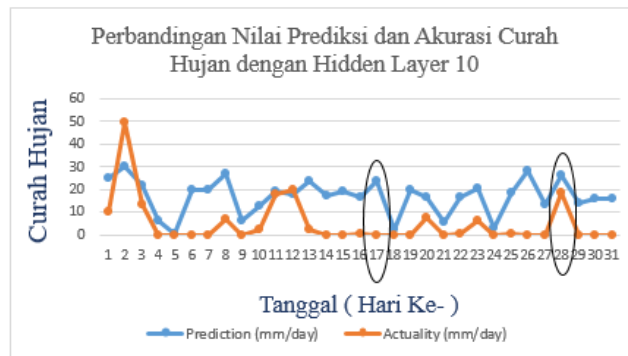
Gambar 5. Grafik Perbandingan Hasil Prediksi dengan 7 *Hidden Layer*

Sebagai contoh, pada hari ke-4 terdapat nilai prediksi sebesar 61.1848 yang masuk dalam kategori hujan dan nilai aktual sebesar 0 yang masuk dalam kategori cerah. Kemudian pada hari ke-28 terdapat nilai prediksi sebesar 67.8756 yang masuk dalam kategori hujan dan nilai aktualnya sebesar 18.6 yang juga masuk pada kategori hujan. Pada proses *training* kedua, digunakan nilai parameter *hidden layer* sebanyak 7 untuk dilakukan pengecekan apakah dengan nilai parameter tersebut dapat mencapai nilai akurasi yang optimal.

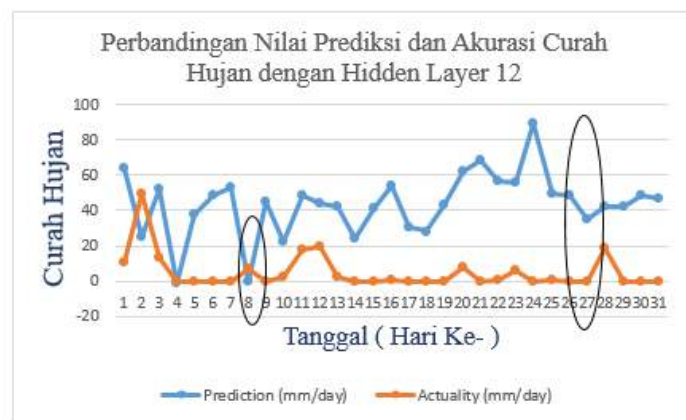
Pada Gambar 5 dapat diketahui data hasil prediksi dan aktualnya yang diperoleh

pada proses *training* kedua dengan nilai parameter tersebut. Sebagai contoh, pada hari ke-1 terdapat nilai prediksi sebesar 3.4328 yang masuk dalam kategori cerah dan nilai aktual sebesar 10.2 yang masuk dalam kategori hujan. Kemudian pada hari ke-12 terdapat nilai prediksi sebesar 5.2681 yang masuk dalam kategori hujan dan nilai aktualnya sebesar 19.5 yang juga masuk pada kategori hujan.

Pada proses *training* ketiga, digunakan nilai parameter *hiddenlayer* sebanyak 10 untuk dilakukan pengecekan apakah dengan nilai parameter tersebut dapat mencapai nilai akurasi yang optimal.



Gambar 6. Grafik Perbandingan Hasil Prediksi dengan 10 *Hidden Layer*



Gambar 7. Grafik Perbandingan Hasil Prediksi dengan 12 *Hidden Layer*

Pada Gambar 6 dapat diketahui data hasil prediksi dan aktualnya yang diperoleh pada proses *training* ketiga dengan nilai parameter tersebut.

Sebagai contoh, pada pada hari ke-17 terdapat nilai prediksi sebesar 23.8926 yang masuk dalam kategori hujan dan nilai aktualnya sebesar 0 yang masuk pada kategori cerah. Kemudian pada hari ke-28 terdapat nilai prediksi sebesar 26.4932 yang masuk dalam kategori hujan dan nilai aktual sebesar 18.6 yang juga masuk dalam kategori hujan. Pada proses *training* keempat, digunakan nilai

parameter *hidde nlayer* sebanyak 12 untuk dilakukan pengecekan apakah dengan nilai parameter tersebut dapat mencapai nilai akurasi yang optimal.

Pada Gambar 7 dapat diketahui data hasil prediksi dan aktualnya yang diperoleh pada proses *training* keempat dengan nilai parameter tersebut.

Sebagai contoh, pada baris atau hari ke-8 terdapat nilai prediksi sebesar -0.5249 yang masuk dalam kategori cerah dan nilai aktual sebesar 7.1 yang masuk dalam kategori hujan. Kemudian pada hari ke-27 terdapat nilai

prediksi sebesar 34.9541 yang masuk dalam kategori hujan dan nilai aktual sebesar 0 yang masuk dalam kategori cerah Pada proses *training* kelima, digunakan nilai parameter *hidden layer* sebanyak 14 untuk dilakukan pengecekan apakah dengan nilai parameter tersebut dapat mencapai nilai akurasi yang optimal.

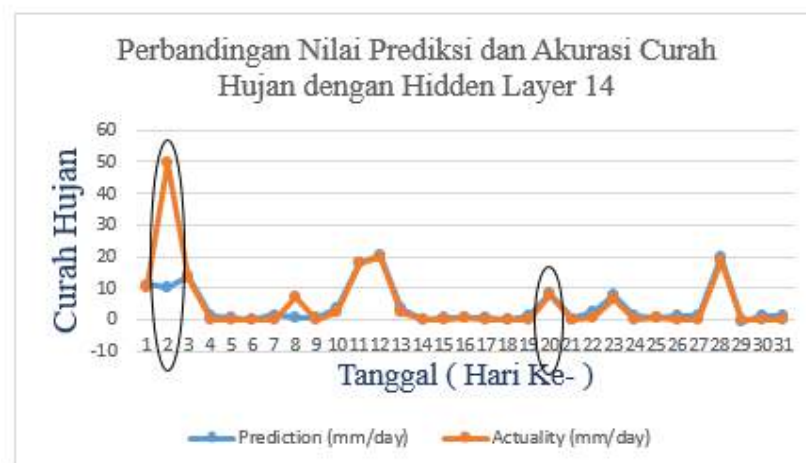
Pada Gambar 8 dapat diketahui data hasil prediksi dan aktualnya yang diperoleh pada proses *training* kelima dengan nilai parameter tersebut.

Sebagai contoh, pada baris atau hari ke-2 terdapat nilai prediksi sebesar 10.3069 yang masuk dalam kategori hujan dan nilai aktual sebesar 49.6 yang juga masuk dalam kategori hujan. Kemudian pada hari ke-20 terdapat nilai prediksi sebesar 8.2788 yang masuk dalam kategori hujan dan nilai aktual sebesar 7.5 yang juga masuk dalam kategori hujan.

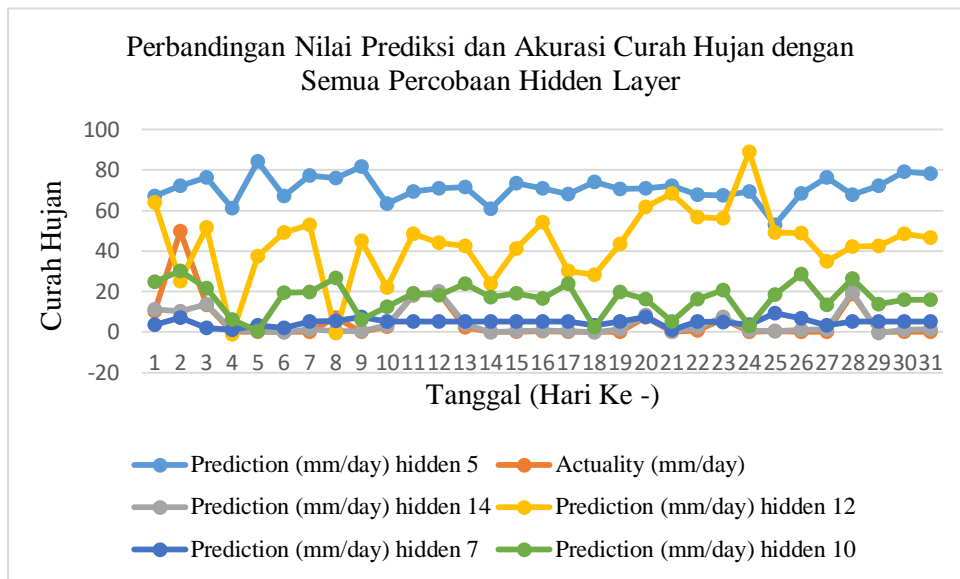
Hasil Prediksi Curah Hujan

Setelah melakukan percobaan *training* sebanyak 5 kali, hasil prediksi curah hujan di Jakarta berbasis algoritma *levenberg marquardt* dapat dilihat pada Gambar 9 dimana hasil yang paling mendekati nilai aktual adalah ketika *hidden layer* bernilai 14.

Sebagai contoh, grafik garis dengan warna merah menunjukkan data aktual dengan grafik garis berwarna hijau menunjukkan data dengan *hidden layer* 14 membentuk grafik garis yang menimpa menandakan bahwa *hidden layer* 14 adalah nilai yang cocok untuk digunakan pada penelitian ini. Setelah muncul data prediksi selama 31 hari, tepatnya data tersebut adalah data pada bulan april maka akan muncul pula hasil pembagian jumlah hari dengan kategori cerah dan hujan dapat dilihat pada Tabel 2.



Gambar 8. Grafik Perbandingan Hasil Prediksi dengan 14 *Hidden Layer*



Gambar 9. Grafik Perbandingan Seluruh Percobaan *Hidden Layer*

Tabel 3. Hasil Pembagian Jumlah Hari

	Sunny	Rain
Prediction	23	8
Actuality	22	9

Pada Tabel 3 terdiri dari 2 baris, yaitu baris *prediction* dan baris *actuality*. Baris tersebut memiliki nilai jumlah hari yang terbagi pada 2 kolom, yaitu kolom sunny dan kolom rain. Tabel tersebut menunjukkan bahwa terdapat 23 hari cerah dan 8 hari hujan pada hasil prediksi, sedangkan 22 hari cerah dan 9 hari hujan pada data asli. Pembagian kategori jumlah hari cerah (*sunny*) dan hari perhitungan akurasi berdasarkan tiga parameter, yaitu *precision*, *recall*, dan *accuracy* sesuai dengan rumus yang telah dijelaskan sebelumnya.

Pada data aktual terdapat 22 hari cerah dan 9 hari hujan. Saat dilakukan *training* terdapat hasil prediksi 23 hari cerah dan 8 hari

hujan (*rain*) berdasar pada nilai intensitas curah hujan pada hari tersebut. Kategori cerah adalah nilai curah hujan < 5 mm per hari dan kategori hujan dengan nilai curah hujan ≥ 5 mm per hari.

Hasil Perhitungan Akurasi

Setelah proses *training* dilakukan dan keluarnya hasil data prediksi, maka dilakukan hujan, kemudian dilakukan pengecekan kembali oleh peneliti, ternyata terdapat 22 hari cerah, 8 hari hujan, dan 1 hari lainnya. Maka dapat dilakukan perhitungan sebagai berikut :

$$Precision = \frac{22}{23} \times 100\% = 95\%$$

$$Recall = \frac{22}{22} \times 100\% = 100\%$$

$$Accuracy = \frac{22 + 8}{31} \times 100\% = 96\%$$

Precision dihitung berdasarkan persentase perbandingan antara cerah yang dipisah benar dengan jumlah cerah yang dipisah. Pada parameter *recall*, merupakan persentase perbandingan cerah yang dipisah benar dengan cerah sebenarnya. Parameter *Accuracy* merupakan persentase perbandingan antara hasil penjumlahan cerah yang dipisah benar dan hujan yang dipisah benar dengan jumlah keseluruhan hari.

KESIMPULAN DAN SARAN

Beberapa kesimpulan yang dapat diambil pada penelitian ini antara lain: Normalisasi data curah hujan berhasil dilakukan pada sejumlah 731 *record* data curah hujan. *Training* data berbasis algoritma *levenberg marquardt* berhasil dilakukan pada data yang telah dinormalisasi sejumlah 744 *record* data dengan menggunakan metode *trainlm*. Pelatihan berhasil dilakukan terhadap data curah hujan yang sudah dinormalisasi menghasilkan jumlah prediksi curah hujan selama 31 hari. Prediksi curah hujan di Jakarta menggunakan algoritma *Levenberg-Marquadt* menghasilkan nilai *precision* 95%, *recall* 100% dan *accuracy* 96%, dimana untuk perhitungan akurasi prediksi >95% diasumsikan memiliki prediksi yang baik.

Pengembangan lebih lanjut dapat dilakukan dengan meningkatkan *performance* pada jaringan. Hal yang dapat dilakukan adalah dengan memperbanyak data training, mengubah nilai parameter yang dalam metode ini adalah perubahan nilai *hidden layer* dan melakukan proses pelatihan menggunakan machine learning lain seperti *Support Vector Machine* (SVM) dan *Bayesian Classifier*.

DAFTAR PUSTAKA

- [1] N. Ritha dan R. Wardoyo, "Implementasi *neural fuzzy inference system* dan algoritma pelatihan *Levenberg-Marquardt* untuk prediksi curah hujan" *Indonesian Journal of Computing and Cybernetics Systems*, vol. 10, no. 2, hal. 125 – 136, 2016.
- [2] M. I. Hastuti, J. A. I. Paski, dan F. Fatkhuroyan, "Improving numerical weather prediction of rainfall events using radar data assimilation", *Indonesian Journal of Geography*, vol. 51, no. 3, hal. 273 – 284, 2019.
- [3] I. Gustari, T. W. Hadi, S. Hadi, dan F. Renggono, "Akurasi prediksi curah hujan harian operasional di Jabodetabek: Perbandingan dengan Model WRF" *Jurnal Meteorologi dan Geofisika*, vol. 13, no. 2, hal. 119 – 130, 2012.
- [4] Indrabayu, N. Harun, M. S. Pallu, dan A. Achmad, "Prediksi curah hujan di wilayah Makassar dengan menggunakan metode *wavelet neural network*", *Jurnal*

- Ilmiah "Elektrikal Enjiniring" UNHAS*, vol. 9, no. 2, hal. 50 – 59, 2011.
- [5] N. E. P. Sari dan E. Sukirman, "Prediksi cuaca berbasis logika *fuzzy* untuk rekomendasi penerbangan di Bandar Udara Raja Haji Fisabilillah", Skripsi Sarjana, Universitas Gunadarma, Depok, 2011.
- [6] M. Feridiansyah, F. Nhita, dan Adiwijaya, "Analisis algoritma prediksi curah hujan menggunakan algoritma *competitive neural network*", Dalam *Eproceedings of Engineering*, 2015, vol. 2, no. 3.
- [7] Y. Andrian dan E. Ningsih, "Prediksi curah hujan di Kota Medan menggunakan metode *backpropagation neural network*", Dalam Prosiding Seminar Nasional Informatika, 2014, hal. 184 – 189.
- [8] I M. D. U. Putra, G. K. Gandhiadi, dan L. P. I. Harini, "Implementasi *backpropagation neural network* dalam prakiraan cuaca di Daerah Bali Selatan", *E-Jurnal Matematika* vol. 5, no. 4, hal. 126 – 132 , 2016.

PUBLIC SENTIMENT ANALYSIS OF PASAR LAMA TANGERANG USING K-NEAREST NEIGHBOR METHOD AND PROGRAMMING LANGUAGE R

¹Hustinawaty, ²Rama Al Azis Dwiputra, ³Tavipia Rumambi

^{1,2,3}Department of Information Systems, Faculty of Computer Science and Information Technology, Universitas Gunadarma

Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat

^{1,3}{hustina, tavipia } @staff.gunadarma.ac.id, ²ramaalazis@yahoo.co.id

Abstract

Pasar Lama Tangerang is a tourist attraction in the city of Tangerang. With the development of current technology, the public can provide an overview of how the facilities and services are provided by expressing opinions on the internet. However, it is difficult to distinguish which opinions belong to positive or negative opinions. Sentiment analysis is needed to overcome this problem. The stage in sentiment analysis starts with collecting data first, then the data is processed. Furthermore, the data that has been propagated is given a sentiment classification using the K-Nearest Neighbor (KNN) algorithm. Then the classification results obtained an accuracy of 83% with a value of $k = 1$ of 120 data divided by 92 positive and 28 negative comments. Sentiment analysis is made using the R and Rstudio programming languages as supporting software

Keywords: Classification, KNN, opinion, Pasar Lama, sentiment

INTRODUCTION

In line with the rapid development of current information technology, the utilization of technological advances can be seen from the ease with which all people in society search for information that is spread freely on the internet. Only by entering keywords will get the desired information, a series of information about it will appear very quickly and easily. Pasar Lama Tangerang is a tourist attraction in the city of Tangerang. With the amount of information that is spread free of charge on media information, especially those on the internet such as the web, blogs, and social media, most people can assess how the Pasar Lama Tangerang in terms of facilities

and services provided by providing comments on media websites that can be done freely and easily. Information written by the public is sometimes not only in the form of opinion, but it is not uncommon to reveal facts about something or only express emotions.

A comment written by the public about a matter is an opinion about the evaluation of a service quality and in general that opinion is about satisfaction or actually expresses the disappointment of consumers from a quality of service that is being enjoyed by consumers. However, it will be difficult to determine which opinion is classified as positive or negative opinion. The number of positive and negative opinions will determine the decision of the new consumer will take in

visiting the Pasar Lama Tangerang, to solve this problem can be done by making a sentiment analysis. One site that can be utilized in this analysis is a review on Google. The reason to use Google, is because almost everyone uses Google as a search engine. Google has a feature that can track the whereabouts of each place visited by Google Maps location and provide feedback services for users in the form of ratings and comments from each user.

This analysis uses the K-Nearest Neighbor (K-NN) method. There are several advantages of the K-NN method, namely the K-Nearest Neighbor classification algorithm which is proven to achieve good accuracy results and in accordance with the calculations applied in an application [1].

The performance of K-NN as a classification algorithm is quite good shown by several studies that use it. The K-Nearest Neighbor algorithm is very commonly used to categorize texts [2]. This is known because the algorithm is easy and efficient for text classification. Not only easy and efficient, the nature of the K-Nearest Neighbor algorithm is self-learning, where the algorithm can study existing data structures and categorize itself.

RESEARCH METHODOLOGY

Sentiment analysis starts with reading a dataset in the form of comment data that has been made previously in Microsoft Excel totaling 120 data, after reading the dataset

then the preprocessing stage that consists of the case of folding, tokenizing, stopword removal and stemming.

From the preprocessing stage will be presented in the histogram and wordcloud visualization. Then the preprocessed dataset is divided into two data: training data with 102 data and testing data with 18 data. Furthermore, at the classification stage, calculations will be performed based on the K-Nearest Neighbor method to display the results of the classification of comments and system accuracy.

RESULTS AND DISCUSSION

In this research, there will be testing and analysis of the results of the analysis of public sentiments towards the Tangerang Old Market using the K-Nearest Neighbor method. The data tested were 18 data with a value of $k = 1$. Confusion matrix calculation in RStudio is calculated based on the classification of actual and predictive sentiments, and will then be read back by the system forming predictive and actual data such as Table 1. Classes that are predicted manually are displayed at the top of the table, while classes that are read by the system are displayed on the left, so that the table can be calculated for accuracy [3].

1. Calculate *Sensitivity*

$$Sensitivity = \frac{TP}{P} = \frac{13}{16} = 0,8125$$

2. Calculate *Specificity*

$$Specificity = \frac{TN}{N} = \frac{2}{2} = 1$$

3. Calculate *Precision*

$$Precision_{Positive} = \frac{TP}{TP+FP} = \frac{13}{13+0} = 1$$

$$Precision_{Negative} = \frac{TN}{TN+FN} = \frac{2}{2+3} = 0,4$$

4. Calculate *Accuracy*

$$Accuracy = Sensitivity \frac{P}{(P+N)} + Specificity \frac{N}{(P+N)}$$

$$Accuracy = 0,8125 \frac{16}{(16+2)} + 1 \frac{2}{(16+2)} = 0,722 + 0,11 = 0,832$$

After doing the classification, the values of accuracy, sensitivity, specificity,

and precision. Comparison of review data contained in Table 1 as many as 18 reviews, consists of 3 negative testing data and 15 positive testing data. While those that have been classified by the system, as many as 16 reviews are positive sentiments and 2 reviews are negative sentiments. Based on accuracy, the results of the review classification obtained from the sentiment analysis system using K-Nearest Neighbor with k = 1 value of 83% with 100% positive precision and 40% negative precision.

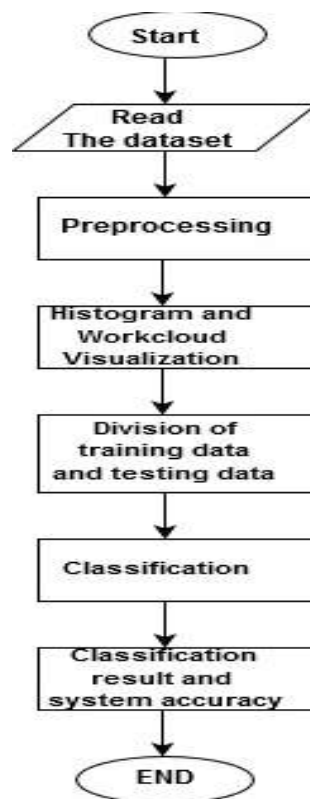


Figure 1 Sentiment Analysis

Table 1. Confusion Matrix

Amount of testing data : 18	Based on Actual Analysis of Negative Sentiments	Based on Actual Analysis of Positive Sentiments	Amount of Predicted Sentiments
Based on Negative Sentiment Prediction Analysis	TN (True Negative) = 2	FP (False Positive) = 0	2
Based on Positive Sentiment Prediction Analysis	FN (False Negative) = 3	TP (True Positive) = 13	16
Amount of Actual Sentiments	5	13	18

CONCLUSION AND SUGGESTION

This study analyzes people's sentiments towards the Pasar Lama Tangerang because it is to find out how the public's response to the Pasar Lama Tangerang in the facilities and services provided is in the form of positive and negative opinions. The sentiment analysis process begins with reading a dataset that has been made as much as 120 data, preprocessing to process raw data into data that is ready to be classified, histogram and wordcloud visualization, training data distribution and testing data in the form of 102 training data and 18 data testing, making data models , classification using the K-Nearest Neighbor algorithm, and finally the results of classification and system accuracy using the

Confusion Matrix produces an accuracy of 83% with 100% positive precision and 40% negative precision. From

120 data generated with 92 positive comments and 28 negative , with the many positive comments can show that the Pasar Lama Tangerang is worth visiting as a tourist attraction of the city of Tangerang for the people of the city of Tangerang and outside the city of Tangerang as consumers.

BIBLIOGRAPHY

- [1] R. R. Sani, J. Zeniarza, and A. Luthfiarta, "Pengembangan aplikasi penentuan tema tugas akhir berdasarkan data abstrak menggunakan algoritma *K-Nearest Neighbor*", In Prosiding Seminar Nasional Multi Disiplin Ilmu & Call For Papers Unisbank (Sendi_U) ke-2, 2016, pp. 103 - 111.
- [2] Y. Samuel, R. Delima, and A. Rachmat, "Implementasi metode *K-Nearest Neighbor* dengan *decision rule*

- untuk klasifikasi subtopik berita”, *Jurnal Informatika*, vol. 10, no. 1, pp. 1 – 15, 2014.
- [3] I. Menarianti, “Klasifikasi *data mining* dalam menentukan pemberian kredit bagi nasabah koperasi”, *Jurnal Ilmiah Teknosains*, vol. 1 No. 1, pp. 36 – 45, 2015.
- [4] R. R. Sani, J. Zeniarza, and A. Luthfiarta, “Penerapan algoritma *K-Nearest Neighbor* pada *information retrieval* dalam penentuan topik referensi tugas akhir”, *Journal of Applied Intelligent System*, vol. 1, no. 2, pp. 123 – 133, 2016.
- [5] Mustakim and G. Oktaviani, “Algoritma *K-Nearest Neighbor Classification* sebagai sistem prediksi predikat prestasi mahasiswa”, *Jurnal Sains, Teknologi dan Industri*, vol. 13, no. 2, pp. 195–202, 2016.
- [6] F. Agus, H. R. Hatta, and Mahyudin, “Pengklasifikasian dokumen berbahasa Arab menggunakan *K-Nearest Neighbor*”, *Jurnal SIFO Mikroskil*, vol. 18, no. 1, pp. 43–56, 2017.

IMPLEMENTASI METODE *EUCLIDEAN DISTANCE* UNTUK EKSTRAKSI FITUR JARAK PADA CITRA *SKELETON*

¹Miftahul Jannah, ²Nurul Humaira

^{1,2}Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Gunadarma

Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat

¹miftah@staff.gunadarma.ac.id, ²nurulhumaira44@yahoo.com

Abstrak

Gait adalah cara atau sikap berjalan kaki seseorang. Tiap orang memiliki cara berjalan yang berbeda, sehingga gerak jalan seseorang sulit untuk disembunyikan ataupun direkayasa. Analisis gait adalah ilmu pengetahuan yang mempelajari tentang kemampuan atau cara bergerak manusia. Dalam bidang kedokteran, analisis gait digunakan untuk menentukan penanganan dan terapi bagi pasien rehabilitasi medik. Dalam penelitian ini digunakan fitur jarak pada citra skeleton. Ekstraksi fitur jarak pada citra skeleton menggunakan metode euclidean distance terbagi dalam beberapa tahapan, dimulai dengan mengambil citra skeleton, konversi citra RGB menjadi citra Biner, proses menemukan titik koordinat dari titik akhir dan titik percabangan, dan ekstraksi fitur pada skeleton. Metode yang digunakan menghasilkan persentase tingkat keberhasilan sebesar 87.84%.

Kata Kunci : ekstraksi jarak, skeleton, matlab

Abstract

Gait is a way of walking people. Each person has a different way of walking so gait is difficult to hide or engineer. Gait analysis is the study of human capabilities or motion. the use of gait analysis in the medical field is used for early treatment and therapy of medical rehabilitation patients. In this research, we will use the distance feature on the skeleton image. Distance feature extraction on skeleton images using euclidean distance method is done in several stages, starting with taking skeleton images, converting RGB images into binary images, the process of finding coordinates of end points and branching points, and feature extraction on skeleton. The method used produces a percentage of success rate of 87.84%.

Keywords: distance extraction, skeleton, matlab

PENDAHULUAN

Sistem keamanan biometrik adalah sistem keamanan yang menggabungkan sebuah sistem *hardware*, *software* dan *database* untuk mengenali seseorang secara unik melalui satu atau lebih karakteristik fisiologis atau perilaku yang dimiliki manusia

seperti gaya berjalan, bentuk wajah, suara, iris mata, sidik jari, dan lain-lain [1].

Berjalan merupakan gerakan yang memungkinkan seseorang untuk berpindah dari satu tempat ke tempat lain dengan menggerakkan kaki maju ke depan dalam posisi yang benar secara bergantian [2]. Saat seseorang berjalan terjadi gerakan berulang-

ulang atau siklus yang terkoordinasi membentuk suatu gaya berjalan (*Gait*).

Gaya berjalan fokus pada cara atau sikap berjalan seseorang [3]. Tiap orang memiliki cara berjalan yang berbeda, sehingga gerak jalan seseorang sulit untuk disembunyikan ataupun direkayasa. Analisis *gait* adalah ilmu pengetahuan yang mempelajari tentang kemampuan atau cara manusia bergerak. Dalam bidang kedokteran, analisis *gait* digunakan untuk menentukan penanganan dan terapi bagi pasien rehabilitasi medik.

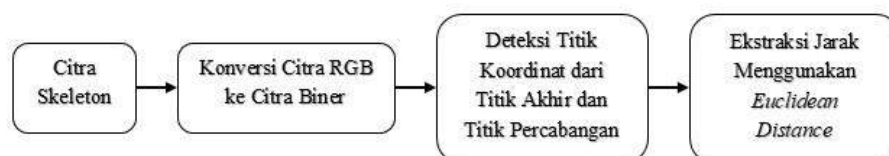
Sejumlah penelitian terkait dengan analisis *gait* telah banyak dilakukan. Hustinawaty dkk [4] dalam penelitiannya melakukan beberapa tahapan untuk pembentukan *skeleton* yaitu proses akuisisi, *background subtraction*, *filtering*, *thresholding* dan pembentukan siluet sehingga menghasilkan citra biner dengan metode *most prominent ridge line* [4]. Ahmed [5] dalam penelitiannya mengekstraksi fitur dengan menggabungkan fitur statis yang didasarkan pada panjang tulang dan fitur dinamis yang diekstraksi dari ketinggian tangan, bahu dan kaki dari permukaan tanah untuk meningkatkan tingkat pengenalan. Citra *skeleton* yang diperoleh dari penggunaan sensor *Microsoft Kinect* digunakan untuk pengenalan *gait*. Penelitian yang dilakukan oleh Sinha dkk [6]

menggunakan sensor *Microsoft Kinect* dan melakukan ekstraksi fitur tubuh bagian atas, tubuh bagian bawah dan titik sentroid untuk analisis *gait*.

Berdasarkan beberapa penelitian yang telah dilakukan, maka dalam penelitian ini yang akan dilakukan adalah mengekstraksi fitur jarak pada citra *skeleton* manusia menggunakan metode *euclidean distance* dan mengimplementasikannya pada Matlab 7.14.0(R2012a). Jarak yang diekstraksi adalah jarak dari tangan ke poros dada, jarak kepala ke poros dada, jarak kaki depan ke kaki belakang, jarak sudut tungkai depan ke sudut tungkai belakang. Adapun metode *euclidean distance* digunakan dikarenakan metode ini merupakan salah satu metode yang cukup baik untuk mengukur jarak (*distance*). Pemanfaatan hasil ekstraksi *skeleton* ini dapat digunakan untuk membantu pihak-pihak yang membutuhkan teknologi ekstraksi fitur *skeleton*.

METODE PENELITIAN

Ekstraksi jarak pada citra *skeleton* dilakukan untuk mencari jarak dari titik yang ada pada *skeleton*. Tahapan ekstraksi jarak pada citra *skeleton* tubuh manusia ditampilkan pada Gambar 1.



Gambar 1. Tahapan Penelitian

Konversi Citra Skeleton

Tahapan diawali dengan mengambil citra *skeleton* manusia untuk dikonversi dari citra RGB (berwarna) diubah menjadi citra biner yang hanya memiliki nilai 0 dan 1. Konversi ini menghasilkan warna latar belakang bernilai 0 dan citra *skeleton* bernilai 1 sehingga mempertajam bentuk citra *skeleton* yang digunakan sehingga memudahkan mendeteksi titik koordinat dari titik akhir dan titik percabangan. Citra *skeleton* yang diperoleh dari proses konversi selanjutnya dicari titik koordinat dari titik akhir dan titik percabangan, mengelompokkan koordinat titik percabangan dan titik akhir untuk menjadi nilai masukan dalam tahap ekstraksi jarak menggunakan *Euclidean Distance*.

Deteksi Titik Koordinat Citra Skeleton

Setelah citra *skeleton* berubah nilai pikselnya mejadi citra biner tahapan selanjutnya mendeteksi titik koordinat titik akhir dan titik percabangan. Koordinat titik akhir yang akan di deteksi adalah ujung tangan, ujung kepala, kaki depan, dan kaki belakang, sudut tungkai depan ke sudut tungkai belakang sedangkan koordinat titik percabangan yang akan dideteksi adalah titik poros dada.

Ekstraksi Fitur Jarak

Titik koordinat titik akhir dan titik percabangan yang telah diperoleh dikelompokkan dan menjadi nilai masukan dalam

persamaan *Euclidean Distance* (1). *Euclidean distance* merupakan generalisasi dari teorema *pythagoras*[7]. Pada persamaan (1), x_1, x_2 merupakan oordinat sumbu x dari sebuah titik dan y_1, y_2 merupakan oordinat sumbu y dari sebuah titik .

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (1)$$

Pada penelitian ini jarak dikelompokkan yaitu jarak dari tangan ke poros dada, jarak kepala ke poros dada, jarak kaki depan ke kaki belakang, jarak sudut tungkai depan ke sudut tungkai belakang.

HASIL DAN PEMBAHASAN

Ekstraksi fitur jarak citra *skeleton* ini menggunakan 20 atlet dengan masing-masing atlet memiliki 32 citra *skeleton*. Contoh gambaran hasil ekstraksi jarak pada citra *skeleton* untuk atlet ke-1 dan atlet ke-2 dapat dilihat pada Tabel 1. Hasil ekstraksi dikelompokkan menjadi 4 kolom yaitu jarak tangan ke poros dada (J1), jarak kepala ke poros dada (J2), jarak kaki depan ke kaki belakang (J3), dan jarak sudut tungkai depan ke sudut tungkai belakang (J4) Tabel 1 menunjukkan bahwa pada pengujian yang dilakukan pada atlet ke-1 dan atlet ke-2 diperoleh persentase keberhasilan program dalam ekstraksi jarak. Gerakan yang menghasilkan jarak yang bernilai 0 yang terdapat pada atlet ke-1 dan atlet ke-2 menunjukkan bahwa proses ekstraksi gagal.

Persamaan digunakan untuk menghitung persentase (2).

$$\text{Persentase} = \frac{(\text{Jumlah fitur} * \text{jumlah gerakan}) - \text{jumlah data yang bernilai 0}}{\text{Jumlah fitur} * \text{jumlah gerakan}} * 100\% \quad (2)$$

Berdasarkan persamaan (2), Atlet ke-1 diperoleh persentase sebesar 92,96% dan Atlet ke-2 diperoleh persentase sebesar 96,09%.

Tabel 1. Contoh Hasil Uji Ekstraksi Jarak *Skeleton*

Atlet ke-1					Atlet ke-2				
Gerakan ke-	J1	J2	J3	J4	Gerakan ke-	J1	J2	J3	J4
1	125	247	132	76	1	99	139	127	0
2	116	206	184	182	2	153	195	191	162
3	131	203	240	171	3	110	101	238	178
4	134	194	311	183	4	149	135	271	187
5	135	190	320	185	5	99	87	304	198
6	159	235	326	168	6	157	145	304	198
7	126	200	277	142	7	102	87	311	177
8	118	221	256	110	8	107	92	295	159
9	110	161	237	137	9	118	126	288	132
10	94	199	192	112	10	113	143	215	124
11	84	227	157	0	11	110	210	157	106
12	106	189	0	0	12	104	188	104	0
13	115	168	123	0	13	124	95	159	0
14	127	171	178	158	14	147	149	199	144
15	121	153	241	159	15	162	199	263	163
16	128	111	311	173	16	173	164	303	171
17	124	112	328	165	17	142	121	311	189
18	104	155	255	138	18	133	138	311	190
19	93	156	259	141	19	139	103	305	178
20	78	170	245	106	20	132	99	303	161
21	90	198	195	122	21	99	140	287	147
22	84	205	140	0	22	86	187	258	169
23	115	183	0	0	23	72	186	198	126
24	123	177	0	0	24	78	166	136	97
25	131	163	141	122	25	87	139	94	0
26	144	157	191	167	26	125	186	122	0
27	136	126	265	151	27	143	133	183	157
28	141	132	255	152	28	80	68	239	166
29	145	136	287	135	29	135	129	279	169
30	135	134	273	129	30	129	130	302	172
31	132	141	270	130	31	136	142	295	158
32	135	154	226	127	32	138	144	287	146

Tabel 2. Persentase Tingkat Keberhasilan

Atlet ke-	Hasil Persentase	Atlet ke-	Hasil Persentase
1	92.96%	11	95.31%
2	96.09%	12	85.93%
3	96.09%	13	84.37%
4	94.53%	14	84.37%
5	89.84%	15	83.59%
6	97.65%	16	83.59%
7	87.50%	17	78.12%
8	88.28%	18	75.78%
9	92.18%	19	83.59%
10	89.06%	20	78.12%

Berdasarkan ekstraksi fitur jarak citra *skeleton* menggunakan 20 atlet dengan masing-masing atlet memiliki 32 citra *skeleton* memberikan tingkat keberhasilan uji coba yang dilakukan dengan hasil seperti ditampilkan pada Tabel 2.

Pada Tabel 2. terlihat bahwa hasil persentase yang diperoleh mulai dari atlet ke-1 sampai atlet ke-20 memiliki hasil persentase diatas 75%. Berdasarkan Tabel tersebut, maka rata-rata persentase dari 20 atlet dapat dihitung menggunakan persamaan (3).

$$\begin{aligned} \text{Total Persentase} \\ = \frac{\text{Total persentase seluruh atlet}}{\text{total atlet}} * 100\% \end{aligned} \quad (3)$$

Berdasarkan persamaan (3), maka total persentase yang diperoleh dari hasil persentase 20 atlet sebesar 87,84%.

KESIMPULAN DAN SARAN

Aplikasi ekstraksi fitur jarak citra *skeleton* pada 20 atlet berhasil dilakukan dengan menggunakan *software* Matlab 7.14.0(R2012a). Proses dilakukan dengan mengubah citra *skeleton* manusia yang masih berupa citra RGB menjadi citra biner, mencari titik koordinat akhir dan percabangan dari citra *skeleton* manusia, mengelompokkan titik koordinat percabangan dan akhir dan dimasukkan ke dalam rumus *Euclidean Distance*. Secara keseluruhan aplikasi eks-

traksi jarak *skeleton* ini dapat mendeteksi jarak dari citra *skeleton* yang ingin dicari, walaupun citra *skeleton* yang ada belum sempurna dan masih ada titik *skeleton* yang tidak terdeteksi.

Pada hasil uji coba citra *skeleton* 20 atlet, diperoleh persentase tingkat keberhasilan sebesar 87.84%. Persentase ini didapat karena terdapat beberapa *skeleton* yang tidak sempurna sehingga titik akhir ataupun titik percabangannya tidak terdeteksi.

Pengembangan penelitian selanjutnya antara lain menggunakan citra *skeleton* yang lebih baik hasilnya, mendeteksi titik yang ingin dicari dengan lebih tepat dengan menggunakan metode ekstraksi fitur lain.

DAFTAR PUSTAKA

- [1] I. M. Alsaadi, "Physiological biometric authentication systems, advantages, disadvantages and future development: A review", *International Journal of Scientific & Technology Research*, vol. 4, no. 12, hal. 285 – 289, 2015.
- [2] J. Carpentier, M. Benallegue, dan J.-P. Laumond "On the centre of mass motion in human walking", *International Journal of Automation and Computing*, vol. 14, no. 5, hal. 542 – 551, 2017.
- [3] W. Tao, T. Liu, R. Zheng, dan H. Feng, "Gait analysis using wearable

- sensors”, *Sensors*, vol. 12, no. 2, hal. 2255 – 2283, 2012.
- [4] Hustinawaty, J. S. Puspito, dan O. Siahaan, Metode most prominent ridge line pada pengukuran rangka atlet jalan cepat”, Dalam *Proceedings Konferensi Nasional Sistem Informasi 2014*, 2014, hal. 195 – 200.
- [5] M. H. Ahmed, “Kinect-based human gait recognition using static and dynamic features”, *International Journal of Computer Science and Information Security*, vol. 14, no. 12, 2016.
- [6] A. Sinha, K. Chakravarty, dan B. Bhowmick, “Person identification using skeleton information from kinect”, Dalam *Proceedings The Sixth International Conference on Advances in Computer-Human Interactions*, 2013, hal.101-108.
- [7] Weisstein, Eric W. "Distance", From MathWorld--A Wolfram Web. [Daring]. Tersedia: <http://mathworld.wolfram.com/Distance.html>. [Diakses: 24 Januari 2018].

IMPLEMENTASI LEXICON BASED DAN NAIVE BAYES PADA ANALISIS SENTIMEN PENGGUNA TWITTER TOPIK PEMILIHAN PRESIDEN 2019

¹Gusti Nur Aulia, ²Eka Patriya

^{1,2}Fakultas Teknologi Industri Universitas Gunadarma
Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat

²ekapatriya@staff.gunadarma.ac.id,

Abstrak

Pilpres saat ini cukup menyita perhatian, karena berbagai rumor yang beredar. Masyarakat juga menjadi sasaran elit politik, dimana suara mereka merupakan penentu keberlangsungan arah politik untuk lima tahun kedepan. Opini-opini positif, netral maupun negatif dapat menimbulkan ancaman munculnya berita bohong (hoax). Salah satu sarana yang digunakan masyarakat dalam mengekspresikan pilihan politiknya adalah melalui media sosial salah satunya twitter. Data seperti opini publik dapat diolah menjadi sebuah informasi yang bermanfaat, salah satunya melalui analisis sentimen. Pada penelitian ini, akan dilakukan analisis sentimen pada Twitter tentang pemilihan presiden 2019. Tahapan analisis sentimen pada penelitian ini terdiri dari akuisisi data, pre-processing, klasifikasi data, evaluasi data dan visualisasi data. Preprocessing dilakukan dengan case folding, normalisasi data, filtering, ubah kata baku, stopword dan stemming. Penelitian ini melakukan 2 metode yaitu dengan metode Lexicon Based dan Naive Bayes Classifier. Hasil akhir dari analisis kemudian dihitung nilai akurasi menggunakan confusion matrix dan di visualisasikan menggunakan web server. Penentuan sentimen prediksi dilakukan menggunakan metode Lexicon Based dan Labelisasi dengan perhitungan secara manual. Data latih dan data uji akan digunakan dalam proses pelatihan dan pengujian menggunakan Naive Bayes Classifier. Hasil klasifikasi yang dilakukan oleh metode Naive Bayes Classifier disebut sentimen aktual. Perhitungan tingkat keakurasian antara sentimen prediksi terhadap sentimen aktual menggunakan pengujian confusion matrix. Hasil yang didapatkan adalah tingkat akurasi antara sentimen prediksi dan sentimen aktual dengan Lexicon Based sebesar 64,49% pada data uji dan pada data latih sebanyak 94,2% serta dengan menggunakan Labelisasi dan Naive Bayes Classifier sebesar 86,53% pada data uji dan data latih sebesar 94,08%. Hasil penelitian ini diharapkan dapat membantu melakukan riset atas opini masyarakat pada Twitter mengenai Pilpres 2019 yang mengandung sentimen positif, negatif atau netral.

Kata Kunci: Classifier, Lexicon Based, prediksi, Naive Bayes, Twitter

Abstract

In this study, researchers will conduct a sentiment analysis on Twitter about the election presidential 2019. The stages of sentiment analysis in this study consist of data acquisition, pre-processing, data processing or classification, data evaluation and data visualization. The pre-processing processes are case folding, data normalization, filtering, changing standard words, stopword and stemming. This study conducted 2 methods, namely the Lexicon Based method and the Naive Bayes Classifier. The final result of the analysis is then calculated the accuracy value using a confusion matrix and visualized using a web server. Determination of initial sentiments or commonly referred to as predictive sentiments is done using the Lexicon Based method and Labeling with manual calculations. After determining the initial sentiment, the data is divided into training data and test data. In this study the number of training data is 845 data (data to 1-845) and for test data amounted to 245 data (data to 846-1090). Training data and test data will be used in the training and testing process using Naive Bayes Classification.

The results of the classification carried out by the Naive Bayes Classification method are called actual sentiments. After the prediction of sentiment is determined at the beginning of the process and the actual sentiment has been obtained, then the level of accuracy is calculated between the predicted sentiment to the actual sentiment using the confusion matrix test. The results obtained are the level of accuracy between predicted sentiments and actual sentiments with Lexicon Based of 64,49% in the test data and in training data as much as 94,2% and by using Labeling and Naive Bayes Classification of 86,53% in the test data and data training of 94,08%.

Keywords : Classifier, Lexicon Based, prediction, Naive Bayes, Twitter

PENDAHULUAN

Indonesia adalah salah satu negara yang menganut sistem demokrasi, sebagai negara yang menganut sistem demokrasi, penting bagi warga negara Indonesia memiliki sebuah proses untuk memilih orang yang dapat mengisi jabatan-jabatan politik tertentu. Proses tersebut kita kenal sebagai Pemilu atau Pemilihan Umum. Di Indonesia sendiri diketahui bahwa pemilu diselenggarakan secara periodik yaitu selama lima tahun sekali. Pada tanggal 17 April 2019 yang lalu telah diselenggarakan kembali pemilu, salah satunya adalah pemilihan presiden (Pilpres). Pilpres saat ini cukup menyita perhatian, karena berbagai rumor beredar saling menjatuhkan antar kedua pasangan calon presiden (paslon). Masyarakat juga menjadi sasaran elit politik, dimana suara mereka merupakan penentu keberlangsungan arah politik untuk lima tahun kedepan. Opini-opini positif, netral maupun negatif seperti SARA, Hak Asasi Manusia (HAM) dan ekonomi dapat menimbulkan ancaman munculnya berita bohong (*hoax*). Salah satu sarana yang digunakan masyarakat dalam mengekspresikan pilihan politiknya adalah melalui media

sosial salah satunya twitter. Twitter merupakan jejaring sosial yang populer di kalangan pengguna internet saat ini, karena menyediakan banyak fitur yang menarik untuk digunakan oleh pengguna seperti berita, tweet antar sesama pengguna dan lain sebagainya. Data seperti opini publik dapat dikumpulkan dan diolah menjadi sebuah informasi yang bermanfaat, salah satunya melalui analisis sentimen [1]. Analisis sentimen diimplementasikan untuk mengklasifikasikan data [2] kedalam klasifikasi yang bersifat positif, negatif maupun netral.

Penelitian terkait mengenai analisis sentimen pada Twitter dilakukan peneliti terdahulu. Penelitian dilakukan [3] dalam melakukan analisis sentimen Twitter menggunakan metode *Lexicon Based* dan *Double Propagation*. Kombinasi *Lexicon Based* dan *Double Propagation* mampu menghasilkan 7 parameter analisis sentimen yaitu sangat positif, positif, agak positif, netral, agak negatif, negatif dan sangat negatif. Penelitian [4] membuat aplikasi klasifikasi opini yang menerapkan pendekatan *Naive Bayes* untuk mengklasifikasikan kata-kata dan difokuskan pada *tweets* dalam Bahasa Indonesia. Aplikasi ini kemudian diterapkan untuk mengklasifika-

sikan opini publik pada Twitter terkait layanan pemerintah terhadap masyarakat, berdasarkan sentimen positif, negatif atau netral. Data latih diperoleh melalui aplikasi platform KNIME *Analytic* dan sumber teks diperoleh dari akun Twitter Dinas pemerintah Kota Bandung. Penelitian [5] penelitian ini mencoba menganalisis persepsi masyarakat kedalam kelas sentimen menggunakan metode *Lexicon Based* dengan *SentiWordNet*. *Dataset* yang digunakan adalah *tweets* mengenai kenaikan harga rokok dalam Bahasa Indonesia berjumlah 350 buah. Data diklasifikasikan sesuai *SentiWordNet* pada tiap-tiap kata dalam kalimat. Untuk kata yang memiliki lebih dari satu arti maka *synset* dipilih berdasarkan metode *First Sense* dari *SentiWordNet* yang muncul paling populer. Peneliti (Antinasari, Perdana, & Fauzi, 2017), pada penelitian ini digunakan kamus kata tidak baku dan normalisasi *Levenshtein Distance* untuk memperbaiki kata yang tidak baku menjadi kata baku dengan pengklasifikasian *Naïve Bayes*. Peneliti [6] melakukan penelitian analisis sentimen, untuk *preprocessing* data menggunakan *tokenisasi*, *cleansing* dan *filtering*, untuk menentukan *class* sentimen dengan metode *Lexicon Based*. Untuk proses klasifikasinya menggunakan metode *Naïve Bayes Classifier (NBC)* dan *Support Vector Machine (SVM)*. Peneliti [7] melakukan penelitian yang terdahulu tentang ujaran kebencian. Metode yang digunakan dalam mengolah data dokumen tersebut

adalah *Backpropagation Neural Network* dengan pembaruan fitur menggunakan *Lexicon Based Features* yang dikombinasikan dengan *Bag of Words*.

Pada penelitian ini, peneliti akan mengimplementasikan penggunaan metode *Lexicon Based* dan *Naïve Bayes Classifier* pada penentuan klasifikasi sentimen positif, negatif maupun netral dalam analisis sentimen mengenai pemilihan presiden 2019. Hasil klasifikasi tersebut kemudian akan di hitung akurasi menggunakan *confusion matrix*. Hasil penelitian ini diharapkan dapat bermanfaat untuk membantu melakukan riset atas opini masyarakat pada Twitter mengenai Pilpres 2019 yang mengandung sentimen positif, negatif atau netral.

METODE PENELITIAN

Metode penelitian yang digunakan pada penelitian ini terdiri atas beberapa tahap proses. Tahap awal merupakan akuisisi data yang diperoleh dari pengambilan data berupa *tweet* dari Twitter. Tahap *pre-processing*, yaitu terdiri dari *case folding*, normalisasi data *tweet*, *filtering*, mengubah kata baku, *stopword* serta *stemming*. Setelah dilakukan *stemming*, tahap selanjutnya merupakan *processing*. *Processing* dilakukan dengan klasifikasi *Lexicon Based*, yaitu klasifikasikan data kedalam bentuk positif, negatif atau netral. Kemudian, *processing* dengan klasifikasi *Naïve Bayes* yang terdiri dari tahap

pelatihan dan pengujian data. Setelah itu, dilakukan evaluasi data dari hasil klasifikasi tersebut. Tahap akhir dari proses ini adalah visualisasi hasil data menggunakan *package shiny* pada RStudio. Akuisisi data (*tweets*) menggunakan API Search Twitter. Pada bahasa pemrograman R tersedia *package* untuk mengambil *tweets*. Tahap akuisisi terdiri dari proses pembuatan koneksi dengan API Search Twitter dan proses pengambilan datanya. Untuk mengambil data penelitian dalam penelitian ini *tweets* sesi R harus dihubungkan dengan API Search Twitter.

Preprocessing Data Twitter

Preprocessing data bertujuan untuk mentransformasi data ke dalam suatu format agar bisa lebih mudah dipahami. Tahap ini merupakan tahap penting dalam analisis sentimen. Pada tahap ini, tweet mengalami kapitalisasi dan pembersihan dari komponen @, RT, url, serta komponen lainnya. Data yang telah diambil dari Twitter tentunya perlu diekstraksi guna menghilangkan variabel yang tidak berguna. Merujuk pada penelitian sebelumnya yang dilakukan oleh [8] pada proses ini dilakukan *case folding*, normalisasi data *tweet*, *filtering*, mengubah kata baku, *stopword* dan *stemming* dengan penjelasan sebagai berikut :

1) *Case folding* atau kapitalisasi merupakan tahap pengubahan huruf pada data tweets menjadi huruf kecil. Hanya huruf 'a' sampai dengan 'z' yang diterima. Karakter selain huruf dihilangkan dan dianggap sebagai

delimiter. Perubahan dilakukan dengan memeriksa ukuran setiap karakter dari awal sampai akhir karakter. Jika ditemukan karakter yang menggunakan huruf kapital, maka huruf tersebut akan diubah menjadi huruf kecil

2) Normalisasi Data Tweet. Data tweets yang didapat dari Twitter seringkali mengandung komponen yang tidak diperlukan seperti delimiter sehingga perlunya penghapusan komponen. Dalam R, penghapusan atau pernormalisasian dapat menggunakan fungsi pada *package tokenizers* dan *package textclean*. Proses pernormalisasian dilakukan dengan menghapus komponen emoji dan html pada data tweets, komponen ini tidak dibutuhkan dalam proses sentimen. Sintaks yang digunakan dalam R adalah :

```
tweets <- tweets %>%  
  replace_emoji(.) %>%  
  replace_html(.)
```

3) *Filtering*. *Filtering* adalah tahap menghilangkan tweets yang duplikat atau ganda menjadi satu. Terkadang, data yang diambil pada Twitter dapat berupa data duplikat atau ganda karena adanya *retweet* atau *repost tweet*. Maka dari itu dengan tahapan *filtering* ini, dapat menjadikan data tweets tidak ada yang ganda. Pada penelitian ini awalnya data yang di dapat dari tahapan akuisisi data adalah sebanyak 5818 tweets, namun setelah di *filtering* data menjadi sebanyak 1090 tweets.

4) Mengubah kata baku. Mengubah kata-kata pada tweets yang tidak baku atau kata-kata cakapan (*slang*) menjadi kata-kata yang baku sesuai dengan Kamus Besar Bahasa Indonesia. Tujuan proses ini sama dengan proses *stemming*.

5) *Stopword*. Pada tahap ini, kumpulan data tweets yang telah melewati tahap ubah kata baku akan melalui tahap menghapus kata-kata yang tidak perlu (*stopword*). Setiap kata pada data tweets akan diperiksa. Jika terdapat kata sambung, kata depan, kata ganti atau kata yang tidak ada hubungannya dalam analisis sentimen, maka kata tersebut akan dihilangkan. Langkah-langkah membuang kata-kata yang tidak perlu adalah sebagai berikut: a) Hasil proses ubah kata baku akan disbandingkan dengan daftar *database* katakata yang tidak perlu b) Dilakukan pengecekan apakah terdapat kata sama dengan daftar atau tidak. c) Jika terdapat kata yang sama pada daftar *database*, maka kata yang sama tersebut akan dihilangkan.

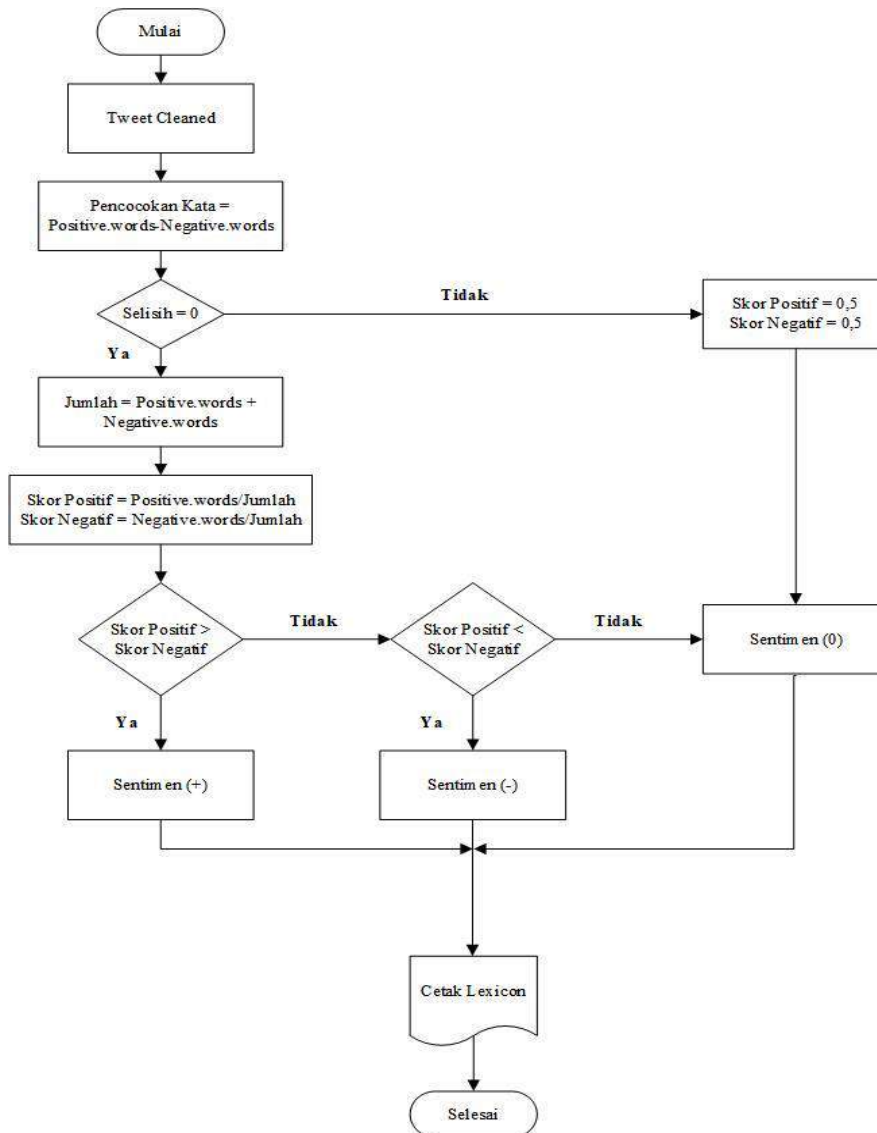
6) *Stemming*. *Stemming* adalah proses mengubah kata menjadi bentuk kata dasarnya dengan cara menghilangkan imbuhan-imbuhan pada kata dalam suatu dokumen. Algoritma *stemming* yang digunakan dalam penelitian ini adalah Algoritma Nazief dan Adriani.

Klasifikasi Data dengan Lexicon Based

Klasifikasi sentimen dengan *Lexical Based* adalah klasifikasi berdasarkan kata positif, kata negatif ataupun netral yang ada pada tweets yang telah dibersihkan.

Klasifikasi ini telah dicocokkan dengan kata-kata yang terdapat dalam kamus *Lexicon* Bahasa Indonesia. Jika tweets memiliki kata positif, maka akan digolongkan pada sentimen tweet positif. Jika tweets memiliki kata negatif, maka akan digolongkan pada sentimen tweet negatif. Namun pada kasus lain jika kedua kata ini bernilai sama, maka digolongkan dalam tweet netral. Bagan alur dari klasifikasi ini dapat dilihat pada Gambar 1. Klasifikasi menggunakan metode *Lexicon Based* dari data tweets sebanyak 1091 menghasilkan sentimen positif sebanyak 484, sentimen negatif sebanyak 392 dan sentimen netral sebanyak 214.

Berikut akan ditampilkan proses klasifikasi kalimat menggunakan metode *Lexicon Based*. Jumlah data yang ditampilkan sebanyak 17 data, baris disebelah kanan merupakan hasil sentimen yang dibagi kedalam sentimen positif, negatif atau netral berdasarkan kamus *Lexicon* yang tersedia.



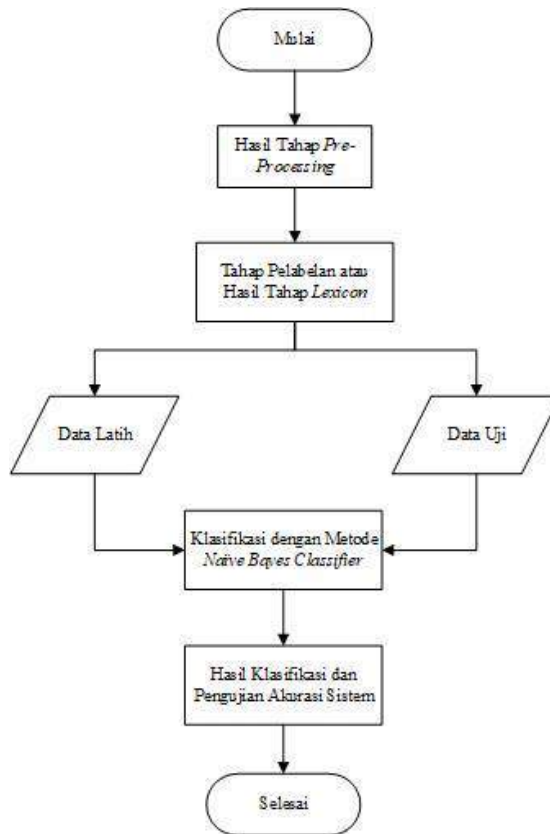
Gambar 1. Alur *Lexicon Based*

Klasifikasi Data dengan *Naive Bayes Classifier*

Tahap Selanjutnya adalah tahap klasifikasi, yaitu tahap pengklasifikasian untuk menentukan apakah data yang akan diuji termasuk kedalam sentimen positif atau negatif. Pada tahap ini digunakan sebuah metode yaitu *Naive Bayes Classifier* dengan

menggunakan pelabelan terlebih dahulu. Metode ini terdiri dari 2 proses, yaitu proses pelatihan dan proses pengujian.

Gambar 2 merupakan bagan alur dari klasifikasi. Adapun tahapan klasifikasi *Naive Bayes Classifier* dengan menggunakan pelabelan.



Gambar 2. Alur *Naive Bayes Classifier*

Tabel 1. Kriteria Label Teks Tweets

Label	Kriteria
Positif	Berisi kata-kata “menang”, “jujur”, “dukung”, “selamat”
Negatif	Berisi kata-kata “curang”, “tolak”, “bodoh”

Proses Pelabelan pada tahapan klasifikasi *Naive Bayes Classifier* adalah sebagai berikut:

1. Penentuan Kategori Sentimen Teks Tweets

Pada tahap ini teks tweets dibaca satu per satu kemudian ditentukan kriteria kategorinya berdasarkan kata-kata tertentu (kata kunci) yang muncul pada teks tweets. Kriteria label pada penelitian ini ditunjukkan oleh Tabel 1.

2. Labelisasi Tweets

Setelah menganalisis, penelitian dilanjutkan ke tahap labelisasi. Pada tahap labelisasi tweets diberikan 1 dari 2 label (“positif” atau “negatif”). Labelisasi dilakukan secara manual dengan cara menuliskan labelnya pada kolom tambahan yang dibuat pada berkas CSV hasil dari data *pre-processing* yang sudah di *stemming*.

Kedua label yang digunakan dalam penelitian ini dijelaskan sebagai berikut: (1) Label positif, dilabelkan pada teks tweets dengan

kriteria berupa dukungan maupun pembelaan. (2) Label negatif, dilabelkan pada teks tweets yang mengandung kata-kata ejekan, sindiran maupun hinaan.

3. Hasil Labelisasi Tweets

Adapun hasil pelabelan dai 1090 data tweets menghasilkan 807 teks tweets terlabeli positif dan 283 teks tweets terklasifikasi negatif. Beberapa contoh teks tweets beserta labelnya dituangkan ke dalam Tabel 2.

4. Transformasi

Setelah selesai pelabelan teks tweets, teks tersebut harus diubah menjadi bentuk yang dapat digunakan oleh komputer (atau lebih tepatnya *machine learning* dalam hal ini). Perubahan bentuk tersebut dikenal sebagai vektorisasi. Dalam lingkup *text mining*, vektorisasi adalah proses membuat vektor dengan nilainya berupa angka-angka kemunculan term (yaitu kata unik) dalam dokumen [9]. Untuk melakukan vektorisasi maka teks tweets harus dipisah menjadi kata per kata. Pemisahan teks tweets menjadi kata-kata disebut dengan tokenisasi. Tokenisasi adalah proses pemecahan dokumen menjadi komponen-komponen individual [10]. Dalam

hal ini komponen individual tersebut yaitu token adalah kata dari tiap teks tweets. Tujuan tokenisasi adalah agar dapat digunakan oleh model *machine learning* (Hari, 2015). Dalam penelitian ini token-token dibuat dalam bentuk *document term matrix*. Untuk membuat *document term matrix* digunakan perintah berikut:

```
dtm_tweets<-
DocumentTermMatrix(VCorpus(VectorSource(tweets$Teks)))
```

Keterangan :

- a. DocumentTermMatrix() adalah fungsi untuk membuat document term matrix dari sebuah corpora, yang disediakan oleh *package tm*.
- b. VCorpus (VectorSource(tweets\$teks)) adalah parameter berupa fungsi untuk membuat corpora volatil. Pada penelitian ini hanya variabel teks yang dijadikan corpora, sehingga dituliskan teks tweets\$teks sebagai vektor sumber.
- c. VectorSource() adalah fungsi untuk membuat sebuah vektor sumber.

Tabel 2. Contoh Teks Tweets yang Sudah Dilabeli

Label	Teks Tweets
Positif	selamat menang pilpres moga pimpin amanah rakyat indonesia
Negatif	tidak gratis bohong curang pilpres bentar lagi laknat dunia turun

5. Implementasi Algoritma Pengklasifikasi *Naïve Bayes*

Setelah selesai melakukan tahap transformasi, analisis dilanjutkan ke tahap Implementasi Algoritma Pengklasifikasi *Naïve Bayes*. Pada tahap *processing* diimplementasikan algoritma pengklasifikasi *Naïve Bayes*.

Adapun tahap *processing* pada penelitian ini terdiri dari 3 proses, yaitu 1) Membuat Set Data Latih dan Uji. Untuk membuat set data latih maupun set data uji, objek `dtm_tweets` terlebih dahulu “dibagi” menjadi 2 buah objek menggunakan 2 perintah berikut:

```
dtm_tweets_latih <- dtm_tweets[1:845, ]  
dtm_tweets_uji <- dtm_tweets[846:1090, ]
```

Keterangan:

[1:845,] dan [846:1090,] menunjukkan nomor baris teks tweets dalam objek `dtm_tweets`. Dengan menulis 2 perintah di atas, terbentuklah 2 buah objek matriks berisi data latih sebanyak 845 dan data uji sebanyak 245.

2) Melakukan Latih pada Set Data Latih

Setelah set data latih dan set data uji dibuat, maka dilanjutkan ke proses latih. Pada proses latih diimplementasikan algoritma pengklasifikasi *Naïve Bayes*. Algoritma pengklasifikasi *Naïve Bayes* akan menggunakan kemunculan tiap *term* (kata unik) dalam teks tweets untuk menghitung

pembuatan set data latih dan set data uji, melakukan data latih (membangun model pengklasifikasi menggunakan algoritma *Naïve Bayes*), dan diakhiri dengan klasifikasi (mengimplementasikan model pengklasifikasi pada set data uji teks tweets). Berikut adalah proses-proses tahapan implementasi algoritma pengklasifikasi *Naïve Bayes*:

probabilitas klasifikasi teks tweets. Dengan kata lain, proses latih akan menghasilkan model pengklasifikasi teks tweets untuk penelitian ini.

Proses latih pada algoritma pengklasifikasi *Naïve Bayes* terdiri dari 2 buah perhitungan, yaitu perhitungan untuk mencari probabilitas kemunculan label (kategori sentimen) dan perhitungan untuk mencari probabilitas kemunculan tiap *term* untuk tiap klasifikasi. Proses latih dapat dituliskan dengan 2 persamaan berikut:

$$P(\text{Label}_i) = \frac{\text{Jumlah Label}_i}{\text{Panjang Data Latih}} \quad (1)$$

$$P(\text{Term}_n | \text{Label}_i) = \frac{\text{Frekuensi Kemunculan Term}_n \text{ pada Teks dengan Label}_i}{\text{Jumlah Label}_i} \quad 2)$$

3) Melakukan Klasifikasi

Setelah model pengklasifikasi dibuat, maka proses pengklasifikasian dapat dilakukan. Berdasarkan teorema *Naïve Bayes*, probabilitas kategori teks tweets dituliskan pada persamaan 3.

$$P(V_{MAP}) = \frac{P(\text{term}_1, \text{term}_2, \text{term}_3, \dots, \text{term}_n | \text{label}_i) P(\text{label}_i)}{P(\text{term}_1, \text{term}_2, \text{term}_3, \dots, \text{term}_n)} \quad (3)$$

Nilai $P(\text{term}_1, \text{term}_2, \text{term}_3, \dots, \text{term}_n)$ konstan untuk semua label_i , sehingga persamaan di atas dapat disederhanakan menjadi persamaan 4.

$$V_{MAP} = \prod_{x=1}^n P(\text{term}_x | \text{label}_i) \frac{1}{Z} P(\text{label}_i) \quad (4)$$

Keterangan:

a. $\prod_{x=1}^n P(\text{term}_x | \text{label}_i)$ adalah hasil perkalian sekuensial dari nilai probabilitas *term* pertama yang muncul pada label pertama

hingga nilai probabilitas *term* ke-n yang muncul pada label terakhir.

b. Z adalah faktor penyekala yang mengubah probabilitas menjadi persamaan di atas. Nilai Z dihitung secara otomatis oleh algoritma.

HASIL DAN PEMBAHASAN

Visualisasi data dalam bentuk histogram hasil analisis sentimen menggunakan lexicon based dan naïve bayes classifier dapat dilihat pada Gambar 3. Gambar visualisasi data dalam bentuk *wordcloud* dapat dilihat pada Gambar 4.



Gambar 3. Tampilan Halaman Visualisasi Histogram



Gambar 4. Tampilan Halaman Visualisasi *Wordcloud*

Pengujian Sistem dengan *Confusion Matrix*

Pengujian akurasi klasifikasi dilakukan untuk mengetahui tingkat akurasi klasifikasi data *tweets* yang dilakukan secara manual menggunakan Labelisasi dan dengan metode *Lexicon Based* dengan klasifikasi data *tweets* yang dilakukan oleh sistem dengan menggunakan metode *Naïve Bayes Classifier*. Pengujian dilakukan dengan menggunakan *confusion matrix* yaitu sebuah matrik dari prediksi yang akan dibandingkan dengan kelas yang asli dari data inputan. Pengujian dilakukan menggunakan 245 data uji yang sudah diberi label. Data uji yang telah diklasifikasikan secara manual akan dibandingkan dengan hasil klasifikasi yang dilakukan oleh

sistem menggunakan metode *Naïve Bayes Classifier*. Hasil pengujian akurasi klasifikasi data *tweets* dapat dilihat pada Tabel 3 dan 4.

$$\text{Akurasi} = (\text{TNegatif} + \text{TNetral} + \text{TPositif}) / \text{Data Uji}$$

$$\text{Akurasi} = (78 + 26 + 54) / 245$$

$$\text{Akurasi} = 158 / 245 = 0,6449 * 100\% = 64,49\%$$

Data pengujian akurasi yang digunakan pada Tabel 3 dari 845 data latih dan 245 data uji, sistem berhasil mengklasifikasi 158 data dengan tepat dan 87 data dengan keliru. Berdasarkan pengujian akurasi, didapatkan hasil akurasi klasifikasi data *tweets* dari sistem analisis sentimen dengan menggunakan metode *Lexicon Based* dan diuji dengan sistem *Naïve Bayes Classifier* sebesar 64,49%.

Tabel 3. Tabel *Confusion Matrix Lexicon* dengan Sistem

Prediksi Sistem	<i>Lexicon</i>		
	Negatif	Netral	Positif
Negatif	78	21	8
Netral	20	26	18
Positif	11	9	54

Tabel 4. Tabel *Confusion Matrix Labelisasi* dengan Sistem

Jumlah Data Uji: 245	Sentimen Hasil Analisis Negatif	Sentimen Hasil Analisis Positif
Sentimen Asli Negatif	(TrueNegatif) 25	(FalsePositif) 14
Sentimen Asli Positif	(FalseNegatif) 19	(TruePositif) 187

1. Akurasi

Akurasi = (TNegatif + TPositif) / Data Uji

$$\text{Akurasi} = (25+187) / 245$$

$$\text{Akurasi} = 212 / 245 = 0,8653 * 100\% = 86,53\%$$

2. Kesalahan Sistem

Kesalahan Sistem = 1 – Akurasi

$$\text{Kesalahan Sistem} = 1 - 0,8653 = 0,1347 * 100\% = 13,47\%$$

3. Presisi

Presisi Positif = (TP) / (TP + FP)

$$\text{Presisi Positif} = 187 / (187+14)$$

$$\text{Presisi Positif} = 187 / 201 = 0,9303 * 100\% = 93,03\%$$

Presisi Negatif = (TN) / (TN + FN)

$$\text{Presisi Negatif} = 25 / (25+19)$$

$$\text{Presisi Negatif} = 25 / 44 = 0,5681 * 100\% = 56,81\%$$

4. Recall/Sensitivity

Recall = (TP) / (TP+FN)

$$\text{Recall} = 187 / (187+19)$$

$$\text{Recall} = 187 / 206 = 0,9077 * 100\% = 90,77\%$$

5. Specificity

Specificity = (TN) / (TN+FP)

$$\text{Specificity} = 25 / (25+14)$$

$$\text{Specificity} = 25 / 39 = 0,6410 * 100\% = 64,10\%$$

Data pengujian akurasi yang digunakan pada Tabel 4 dari 845 data latih dan 245 data uji, sistem berhasil mengklasifikasi 212 data dengan tepat dan 33 data dengan keliru. Berdasarkan pengujian akurasi, didapatkan hasil akurasi klasifikasi data *tweets* dari

sistem analisis sentimen dengan menggunakan Labelisasi dan diuji dengan sistem *Naïve Bayes Classifier* sebesar 86,65% dengan kesalahan sistem sebesar 13,47%, presisi positif sebesar 93,03% serta presisi negatif sebesar 56,81%, *recall* sebesar 90,77% dan *specificity* sebesar 64,10%. Kesimpulan yang diperoleh dari pengujian akurasi menggunakan *confusion matrix* adalah bahwa analisis sentimen menggunakan Labelisasi dan *Naïve Bayes Classifier* dapat digunakan sebagai metode pengklasifikasian pada analisis sentimen karena tingkat akurasi yang lebih besar dibandingkan dengan menggunakan metode *Lexicon Based*.

KESIMPULAN DAN SARAN

Hasil klasifikasi menggunakan metode *Lexicon Based* menghasilkan klasifikasi positif sejumlah 484, negatif 392 dan netral 214 dari data *tweets* sebanyak 1090. Pada metode ini klasifikasi terbanyak adalah pada sentimen positif. Klasifikasi menggunakan metode *Naïve Bayes Classifier* dan pelabelan manual dengan data latih sebanyak 845 dan data uji sebanyak 245, menghasilkan klasifikasi sentimen positif sebanyak 206 dan negatif sebanyak 39. Pada metode ini hasil klasifikasi terbanyak adalah pada sentiment positif. Pada penelitian analisis sentimen ini didapatkan hasil akurasi yang diuji oleh *Confusion Matrix*, klasifikasi data *tweets* dari sistem analisis sentimen dengan menggunakan metode *Lexicon Based* sebesar

64,49% pada data *testing* dan pada data *training* sebanyak 94,2% serta dengan menggunakan Labelisasi dan *Naive Bayes Classifier* sebesar 86,53% pada data *testing* dan data *training* sebesar 94,08%. Maka berdasarkan hasil analisis sentimen pada penelitian ini dapat disimpulkan bahwa opini masyarakat melalui Twitter mengenai Pilpres 2019 menggunakan 2 metode yaitu *Lexicon Based* dan *Naive Bayes Classifier*, hasil yang mendapatkan tingkat akurasi paling tinggi yaitu dengan Labelisasi *Naive Bayes Classifier* dan dari kedua metode tersebut hasil dari klasifikasi sentimen memiliki kecenderungan yang sama yaitu sentimen positif.

Pengembangan lebih lanjut dapat dilakukan seperti data yang ditarik dari Twitter dalam jumlah yang besar, menggunakan algoritma yang berbeda, selain melakukan *classification* juga melakukan *clustering* untuk pemisahan data dalam jumlah besar, serta adanya fitur *real-time* pada visualisasi data dan dapat diakses kapan saja. Pada pengembangan lebih lanjut dari sistem yang sudah dibuat, diharapkan aplikasi ini dapat dikembangkan dengan menggunakan metode lain yang lebih baik dan dengan mengikuti perkembangan teknologi informasi.

DAFTAR PUSTAKA

- [1] M. Adriani, J. Asian, B. Nazief, S. M. Tahaghoghi, dan H. E. Williams, “Stemming Indonesian: A confix-stripping approach”, *ACM Transactions on Asian Language Information Processing (TALIP)*, vol. 6, no. 4, hal. 1 – 33, 2007.
- [2] X. Ding, B. Liu, dan P. S. Yu, “A holistic lexicon-based approach to opinion mining”, Dalam Proceedings of the 2008 International Conference on Web Search and Data Mining, 2008.
- [3] G. A. Buntoro, T. B. Adji, dan A. E. Permanasari “,Sentiment analysis Twitter dengan kombinasi *lexicon based* dan *double propagation*,” dalam *CITEE 2014*, 2014, hal. 39 – 43.
- [4] Falahah dan D. W. A. Nur, “Pengembangan aplikasi *sentiment analysis* menggunakan metode Naive Bayes (Studi kasus sentiment analysis dari media Twitter)”, Dalam Seminar Nasional Sistem Informasi Indonesia, 2015 Hal. 335 – 340.
- [5] I. Kusumawati, “Analisa sentimen menggunakan *lexicon based* kenaikan harga rokok pada media sosial Twitter,” Skripsi Sarjana, Universitas Muhammadiyah Surakarta, Surakarta, 2017.
- [6] G. A. Buntoro, “Analisis sentimen calon Gubernur DKI Jakarta 2017 di Twitter”, *INTEGER: Journal of Information Technology*, vol. 2, no. 1, hal. 32 – 41, 2017.
- [7] M. M. Munir, M. A. Fauzi, dan R. S. Perdana, “Implementasi metode

- backpropagation neural network berbasis lexicon based features dan bag of words untuk identifikasi ujaran kebencian pada Twitter,” *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 2, no. 10, hal. 3182 – 3191, 2018.
- [8] S. K. Ravindran dan V. Garg, Vikram, *Mastering Social Media Mining with R*. Packt Publishing Ltd. UK, 2015.
- [9] W. Baugh, 2013. <https://stackoverflow.com/a/17054702>.
- [10] B. Lantz, *Machine Learning With R*. Birmingham: Packt Publishing Ltd, 2015.

APLIKASI VIRTUAL REALITY TAMAN WISATA BAWAH LAUT SEA WORLD ANCOL BERBASIS ANDROID

¹Deviliana, ²Nurma Nugraha

^{1,2} Fakultas Teknologi Industri Universitas Gunadarma,
Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat

¹ deviliana1295@gmail.com, ² nurma@staff.gunadarma.ac.id

Abstrak

Aplikasi ini memberikan gambaran tempat, suasana serta biota laut yang ada di dalam Sea World Ancol dengan cara yang berbeda yaitu dengan menggunakan teknologi Virtual Reality. Sehingga dengan aplikasi ini masyarakat dapat mengetahui dan mengenal berbagai biota laut apa saja yang ada di Sea World Ancol. Metode penelitian dilakukan dengan menggunakan metode SDLC (Software Development Life Cycle), dengan beberapa tahapan dalam melakukan penelitian seperti pengumpulan data, perancangan, implementasi, dan uji coba. Tahapan uji coba menggunakan metode blackbox testing dilakukan menggunakan sebuah ponsel android OPPO F1f. Hasil uji coba didapatkan bahwa semua fungsi pada button berjalan sesuai dengan rancangan aplikasi. Hasil dari uji Coba User (User Acceptance Testing) dapat disimpulkan aplikasi sudah sangat baik dalam berbagai aspek yang dijadikan pertanyaan dalam kuisisioner. Spesifikasi minimum untuk menjalankan aplikasi virtual reality taman wisata bawah laut sea world Ancol adalah smartphone yang memiliki OS Android v5.0 Lollipop dan mempunyai RAM minimal 3 GB. Semakin tinggi spesifikasi smartphone yang digunakan semakin baik performa aplikasi yang dihasilkan. Aplikasi virtual reality taman wisata bawah laut sea world Ancol dibuat menggunakan Unity 5.5.2f1.

Kata Kunci: Blender, Sea World, Virtual Reality

Abstract

This application provides an overview of the place, atmosphere and marine life in the Sea World Ancol in a different way by using Virtual Reality technology. So that with this application the public can know and recognize any kind of marine life in the Sea World Ancol. The research method was carried out using the SDLC (Software Development Life Cycle) method, with several stages in conducting research such as data collection, design, implementation, and testing. Stages of testing using the blackbox testing method are performed using an OPPO F1f android phone. The test results obtained that all functions on the button run in accordance with the application design. The results of the User Acceptance Testing can be concluded that the application has been very good in various aspects used as questions in the questionnaire. The minimum specifications for running application is a smartphone that has an Android OS v5.0 Lollipop and has a minimum of 3GB RAM. The higher the specifications of the smartphone that is used the better the performance of the resulting application. The virtual sea park underwater sea world Ancol application was created using Unity 5.5.2f1.

Keywords: Blender, Sea World, Virtual Reality

PENDAHULUAN

Sea World Indonesia resmi dibuka untuk umum pada tanggal 3 Juni 1982.

Akuarium laut terbesar di Asia Tenggara itu menampung berbagai biota laut dari seluruh nusantara. Sea World Ancol senantiasa akan menghadirkan edukasi terkini mengenai dunia

air pada umumnya dan kelautan khususnya dengan berbagai biota yang dihadirkan maupun inovasi dalam tampilannya sehingga menjadi yang terdepan di Indonesia. *Sea World Ancol* merupakan sebuah kawasan yang lengkap untuk memberikan hiburan dan pendidikan khususnya dunia laut. Dalam operasionalnya *Sea World Ancol* mengemban 3 misi besar yaitu: Pendidikan, Konservasi dan Hiburan [1]. Melalui misi ini *Sea World Ancol* menempatkan diri sebagai tempat hiburan berkualitas. *Sea World Ancol* terletak di kawasan rekreasi terpadu Taman Impian Jaya Ancol, Jakarta Utara. Hal ini menyebabkan orang-orang yang berada di luar kota maupun wilayah akan membutuhkan banyak biaya akomodasi dan transportasi untuk mengunjungi taman wisata *Sea World Ancol*.

Perkembangan teknologi yang sedang *tren* saat ini salah satunya yaitu *Virtual Reality* (VR). Teknologi ini dapat meningkatkan imajinasi yang dimiliki pengguna untuk merancang dan menggambarannya ke dalam bentuk yang nyata. Teknologi *Virtual Reality* membuat pengguna seolah-olah berinteraksi dengan suatu lingkungan yang telah disimulasikan. Teknologi *Virtual Reality* sudah mulai diterapkan dalam berbagai bidang diantaranya dalam bidang hiburan dan pendidikan. Dalam dunia pendidikan dan dalam dunia hiburan, teknologi *Virtual Reality* menjadi salah satu terobosan baru dalam kegiatan belajar sekaligus menjadi

suatu hiburan tersendiri bagi pengguna aplikasi dengan teknologi *Virtual Reality*.

Telah banyak penelitian yang membahas aplikasi *virtual reality*, seperti penelitian yang dilakukan oleh R. Ilham, A. Ina, dan Fauziah [2] yang berjudul Implementasi *Virtual Reality* Pada Tur *Virtual* Monumen Nasional Menggunakan Unity 3D Algoritma Greedy Berbasis Android. Pada penelitian tersebut Tur *Virtual* yang dibuat yaitu *Virtual* yang bertujuan untuk memperkenalkan objek wisata di Monumen Nasional, pemain dapat menelusuri apa saja yang ada di sekitar dan di dalam Monumen Nasional. Setiap *virtual* objek memiliki penjelasan tentang sebuah objek wisata dan bangunan 3D. Model 3D dibuat dengan menggunakan aplikasi Blender 3D Modeling, tampilan dalam aplikasi yang dibuat dengan Unity 3D Engine.

Penelitian tentang aplikasi *virtual reality* lainnya yaitu Toba Museum “Museum Berbasis *Virtual Reality* Untuk Mempromosikan Kebudayaan Sumatera Utara”. Toba Museum merupakan aplikasi pengenalan kebudayaan Sumatera Utara berbasis VR dengan objek 3D. Aplikasi tersebut menggambarkan sebuah museum dan didalamnya terdapat pakaian adat, senjata tradisional serta benda-benda yang menggambarkan kebudayaan dari Sumatera Utara. Seluruh objek/asset pada aplikasi ini dibuat menggunakan Blender dan kemudian diolah kembali menggunakan Unity. Dalam pembuatan

aplikasi Toba Museum dilakukan identifikasi asset, *modelling* asset, *export* asset ke unity, pembuatan VR pada Unity, dan *finishing*.

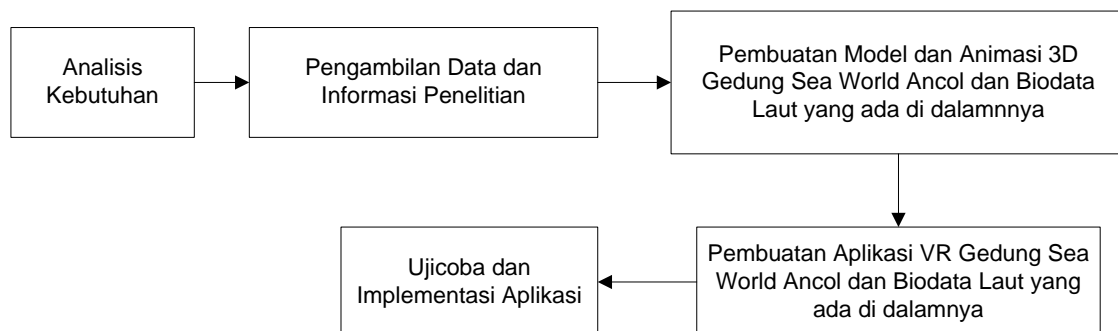
Penelitian aplikasi virtual reality taman wisata bawah laut sea world ancol berbasis android bertujuan membuat aplikasi *Virtual Reality* Taman Wisata Bawah Laut *Sea World* Ancol Berbasis Android. Aplikasi ini diharapkan dapat memberikan gambaran tempat, suasana serta biota laut yang ada di dalam *Sea World* Ancol secara menarik, edukatif dan ekonomis.

METODE PENELITIAN

Aplikasi ini dapat menampilkan visualisasi objek animasi 3D yang terdapat di dalam *Sea World* Ancol menggunakan aplikasi *Virtual Reality*. Metode penelitian dalam pembuatan aplikasi *Virtual Reality* Taman Wisata Bawah Laut *Sea World* Ancol

berbasis Android ini meliputi dua hal utama yaitu pemodelan dan visualisasi dalam bentuk *virtual reality*. *Virtual Reality Sea World* Ancol ini dapat berjalan pada sistem operasi Android yang memiliki sensor *gyroscope*. Gambar 1 memperlihatkan langkah-langkah penelitian.

Berdasarkan Gambar 1 langkah pertama adalah melakukan analisis kebutuhan, yaitu menganalisa perangkat keras dan perangkat lunak yang dibutuhkan dalam pembuatan aplikasi ini. Langkah selanjutnya adalah pengambilan data dan informasi. Data dan informasi didapat dengan cara mendalami pengetahuan tentang biota laut, teknologi *Virtual Reality* dan penggunaan *software* yang digunakan dalam pembuatan aplikasi yaitu *Blender 3D* dan *Unity 3D*. Hal tersebut dari buku-buku dan juga materi dari internet, serta berkunjung langsung ke *Sea World Ancol*.

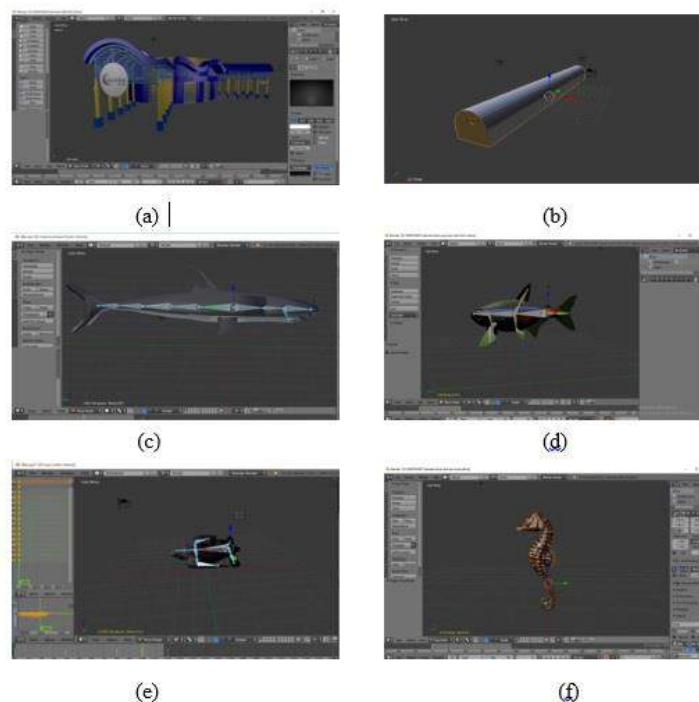


Gambar 1. Langkah-langkah Penelitian

Selain itu, pada langkah ini dilakukan pula cara membuat *scene* dan pembuatan *script* yang dibutuhkan dalam pembuatan aplikasi *Virtual Reality*. Jumlah *scene* yang digunakan pada aplikasi Taman Wisata Bawah Laut *Sea World* Ancol ini berjumlah 8 *scene*, yaitu *menu*, *start*, *how to*, *credit*, *quit*, dalam gedung, *toilet*, dan *main aquarium*. Pada aplikasi ini terdapat *camera player*, animasi pergerakan biota laut, dan sepuluh button yang menggunakan *script* yaitu button *start*, *how to*, *credit*, *entrance*, *menu*, *toilet*, *main aquarium*, *out*, *exit*. Selanjutnya dilakukan pembuatan *user interface* aplikasi dan tahap kompilasi program sehingga dapat di instalasi pada perangkat sistem operasi Android yang memiliki sensor *gyroscop*.

Setelah melakukan tahap pembuatan,

tahap selanjutnya yaitu uji coba pada perangkat Android. Uji coba pada tahapan ini menggunakan metode *blackbox testing* dan *user acceptance testing* yang dilakukan menggunakan sebuah ponsel android. *Black Box Testing* merupakan pengujian yang berfokus pada pengetesan spesifikasi fungsional dari aplikasi. Pengujian ini dilakukan dengan tujuan menemukan kesalahan pada aplikasi yang telah selesai dikembangkan seperti kesalahan interface, kesalahan fungsi-fungsi yang diterapkan, kesalahan dalam struktur data atau akses database, dan kesalahan performa. Sedangkan *user acceptance testing* dilakukan untuk mengetahui tanggapan responden (*user*) terhadap aplikasi *virtual reality* taman wisata bawah laut *Sea World* Ancol.



Gambar 2. (a). Objek Gedung *Sea World*, (b). Objek Aquarium Utama, (c). Objek Ikan Hiu Sirip Hitam, (d). Objek Ikan Hias *Puntius*, (e). Objek Penyu, (f). Objek Kuda Laut

HASIL DAN PEMBAHASAN

Model 3D Sea World Ancol

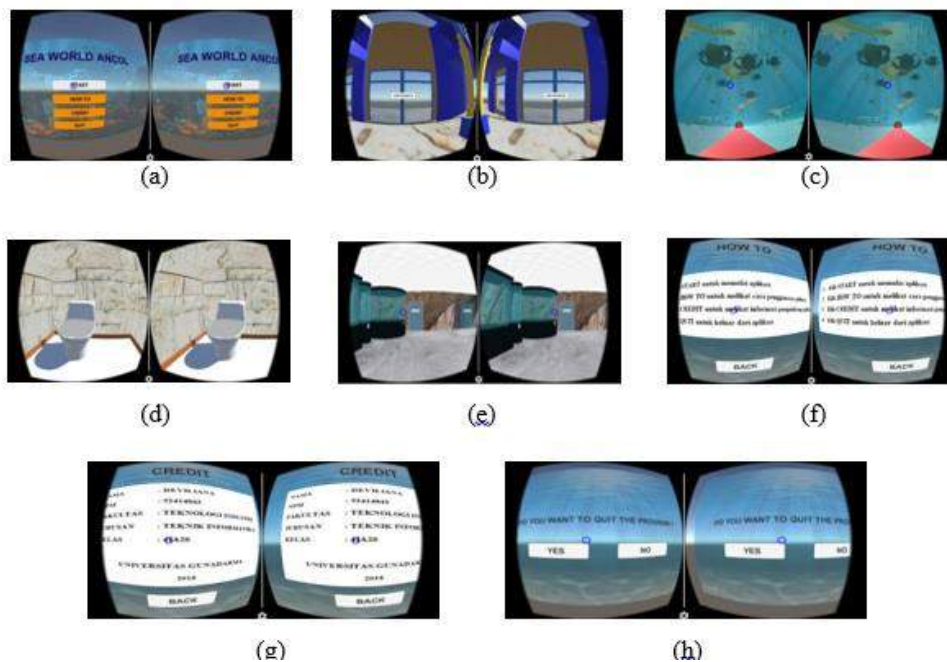
Model 3D yang dihasilkan pada penelitian ini yaitu gedung *Sea World*, akuarium utama, akuarium setengah lingkaran, akuarium silinder, akuarium persegi panjang, ikan hiu sirip hitam, penyu, ikan pari manta, ikan kepe-kepe, ikan arwana merah, ikan *green tang*, ikan hiu martil, ikan hias puntius, ikan buntal, dan kuda laut. Pada Gambar 2 diperlihatkan beberapa model 3D yang telah dibuat pada penelitian ini.

Virtual Reality Sea World Ancol

Pada penelitian ini dibuat beberapa *scene* yang dibutuhkan dalam pembuatan aplikasi *Virtual Reality*. Jumlah *scene* yang digunakan pada aplikasi Taman Wisata

Bawah Laut *Sea World Ancol* ini berjumlah 8 *scene*, yaitu *MENU*, *START*, *HOWTO*, *CREDIT*, *QUIT*, *DALAM GEDUNG*, *TOILET*, dan *MAIN AQUARIUM*. Gambar 3 memperlihatkan 8 *scene* pada aplikasi Taman Wisata Bawah Laut *Sea World Ancol*. Uji coba yang dilakukan pada penelitian ini menggunakan metode *blackbox testing* dan *User Acceptance Testing* yang dilakukan menggunakan sebuah ponsel android.

Uji coba berdasarkan hardware ini dilakukan pada OPPO F1f, dengan spesifikasi OS Android, ROM 16 GB, RAM 3 GB, processor Snapdragon 616, ukuran layar 5 inchi, memiliki sensor *Gyroscope*, dan memiliki *Bluetooth*. *Black Box Testing* merupakan pengujian yang berfokus pada pengetesan spesifikasi fungsional dari aplikasi.



Gambar 3. (a). Hasil Pembuatan *Scene Main Menu*, (b). Hasil Pembuatan *Scene Start*, (c). Hasil Pembuatan *Scene Main Aquarium*, (d). Hasil Pembuatan *Toilet*, (e). Hasil Pembuatan *Scene Dalam Gedung Sea World*, (f). Tampilan *Scene How To*, (g). Tampilan *Scene Credit*, (h). Tampilan *Scene Quit*

Tujuan dari test ini adalah menemukan kesalahan pada aplikasi yang telah selesai dikembangkan seperti kesalahan interface, fungsi-fungsi yang diterapkan, struktur data atau akses database, dan performa. Hasil yang diperoleh dari uji coba *black box testing*, aplikasi ini sudah berjalan dengan baik dan semua fungsi pada *button* berjalan sesuai dengan rancangan aplikasi.

Untuk mengetahui tanggapan responden terhadap aplikasi *virtual reality* taman wisata bawah laut *Sea World Ancol*, dilakukan pengujian dengan memberikan 10 pertanyaan kepada 10 responden secara acak dengan rentang umur 17 sampai 25 tahun. Metode yang digunakan untuk melakukan kuisisioner ini ialah metode skala *likert*. Dari data pernyataan pertama sampai kesepuluh dapat disimpulkan bahwa aplikasi *Virtual Reality* Taman Wisata Bawah Laut *Sea World Ancol* berbasis Android sudah sangat baik dalam berbagai aspek yang dijadikan pertanyaan dalam kuisisioner.

KESIMPULAN DAN SARAN

Aplikasi *Virtual Reality* Taman Wisata Bawah Laut *Sea World Ancol* Berbasis Android berhasil dibuat dengan menggunakan *software* utama Unity 5.5.2f1. Simulasi 3D berbasis *Virtual Reality* yang dihasilkan memberikan kesan seolah-olah pengguna berada di *Sea World Ancol*. Aplikasi *Virtual Reality* Taman Wisata Bawah Laut *Sea World Ancol* hanya dapat dijalankan pada

smartphone android dengan aspek sensor *gyroscope* dan aspek *bluetooth*. Aplikasi ini hanya dapat dinikmati dengan menggunakan kacamata *VR CardBoard* dan *bluetooth controller*. Berdasarkan hasil pengujian dan evaluasi dengan metode *Black Box Testing*, aplikasi *Virtual Reality* Taman Wisata Bawah Laut *Sea World Ancol* berjalan dengan sangat baik dan *output* yang ditampilkan pada aplikasi sesuai dengan rancangan yang telah dibuat. Pada pengujian menggunakan metode UAT (*User Acceptance Test*) memberikan hasil analisa setiap pertanyaan pada kuisisioner memiliki presentase dengan *range* 84% hingga 100%. Dari hasil evaluasi yang dilakukan pada lima *smartphone* android yang memiliki sensor *gyroscope* dan teknologi *bluetooth*, performa yang dimiliki masing-masing ponsel berbeda-beda. Spesifikasi minimum untuk menjalankan aplikasi *Virtual Reality* Taman Wisata Bawah Laut *Sea World Ancol* Berbasis Android adalah ponsel yang memiliki OS Android v5.0 Lollipop dan mempunyai RAM minimal 3 GB. Semakin tinggi spesifikasi *smartphone* yang digunakan, maka semakin baik performa aplikasi yang dihasilkan.

Penelitian ini telah menghasilkan sebuah Aplikasi *Virtual Reality* Taman Wisata Bawah Laut *Sea World Ancol* Berbasis Android ini diharapkan untuk dikembangkan lagi. Objek biota laut yang akan disajikan ditingkatkan lebih banyak lagi dan penambahan objek pelengkap seperti rumput laut, koral, kolam sentuh biota serta

menambahkan audio pada aplikasi sehingga aplikasi akan menjadi lebih menarik dan interaktif.

DAFTAR PUSTAKA

- [1] [Daring]. Tersedia: <https://www.ancol.com/destinasi/sea-world-ancol>. [Diakses 28 Juni 2018].
- [2] R. Ilham, A. Ina, dan Fauziah, “Implementasi *virtual reality* pada tur virtual monumen nasional menggunakan *unity 3D* algoritma greedy berbasis android“, *Journal of Information Technology and Computer Science*, vol. 3, no. 2, hal. 75 – 80, 2018.
- [3] D. Banjarnahor, M. F. Ginting, I. Widyaningrum, H. Hidayatin dan A. A. Gozali, “Museum berbasis virtual reality untuk mempromosikan kebudayaan Sumatera Utara”, Dalam e-Proceeding of Applied Science, 2016, vol. 2, no. hal. 733-740.
- [4] G. C. Burdea dan P. Coiffet, *Virtual reality technology*. New Jersey: John Wiley & Sons, 2003.
- [5] G. Swain, *Object-oriented analysis and design through unified modeling language*. New Delhi: University Science Press, 2010.
- [6] Alex Okita. *Learning C sharp programming with unity 3D*. CRC Press, 2015.
- [7] J. Tidwell, *Designing interfaces*. Sebastopol: O'Reilly Media, 2006.