

ANALISIS SENTIMEN PEMANFAATAN OBROLAN GRUP TELEGRAM BERBAGI INFORMASI LOWONGAN KERJA MENGGUNAKAN METODE NAÏVE BAYES CLASSIFIER

Nova Andre Saputra, Jessica Alexandra, Indra Budi Trisno

Program Studi Teknik Informatika S1, Fakultas Teknik, Universitas Widya Kartika Surabaya

Jalan Sutorejo Prima Utara II No. 1, Surabaya, Indonesia

novaandre.saputraadit@gmail.com

ABSTRAK

Di era digital, seluruh informasi dapat dibagikan secara online, begitu pula dengan informasi berbagi lowongan pekerjaan. Pembagian informasi lowongan pekerjaan secara online dapat dilakukan di beberapa platform terkenal, salah satunya adalah aplikasi Telegram. Aplikasi Telegram banyak dimanfaatkan untuk informasi lowongan pekerjaan karena penggunaannya yang mudah, selain itu Telegram mendukung fasilitas pembuatan grup yang mampu menampung banyak anggota dan bersifat *open public*, sehingga dapat diakses dengan bebas oleh seluruh pengguna Telegram. Dalam hal ini, kami merancang sistem yang dapat melakukan pemantauan percakapan pada grup Telegram berbagi lowongan pekerjaan. Pemantauan dilakukan agar tidak terdapat pesan spam atau menyimpang pada obrolan grup. Perancangan sistem analisis sentimen dibuat menggunakan *text mining*, serta metode Naïve Bayes Classifier. Perancangan sistem diimplementasikan menjadi bot Telegram yang dapat diakses oleh publik. Dengan adanya bot Telegram yang melakukan pemantauan, maka tidak ada lagi pesan spam yang mengganggu anggota lain, sehingga tujuan utama dibuatnya grup, yaitu untuk berbagi informasi lowongan kerja dapat berjalan dengan maksimal.

Kata kunci: Telegram, analisis sentimen, text mining, Naïve Bayes Classifier.

1. PENDAHULUAN

Kebutuhan pekerja setiap tahun terus mengalami kenaikan seiring dengan pertumbuhan ekonomi di suatu daerah. Lapangan pekerjaan yang sedang meningkat tersebut membuka peluang bagi pelajar atau masyarakat umum untuk mencari pekerjaan yang sesuai dengan keahlian atau kemampuan yang mereka miliki. Disisi lain, perekrut mencari calon pekerja yang sesuai dengan standar keahlian perusahaan yang dibutuhkan. Grup media sosial Telegram merupakan salah satu dari media sosial yang sering digunakan untuk berbagi informasi lowongan pekerjaan dan/atau mencari pekerjaan. Hal ini terjadi sebab grup Telegram memiliki kapasitas yang cukup besar untuk mengumpulkan pengguna dalam satu grup.

Akses informasi grup Telegram sangat mudah, bahkan ada yang bersifat *public group*, sehingga perekrut dan calon pekerja mudah untuk melakukan komunikasi ataupun tukar pendapat di dalam grup yang sama. Selain membicarakan atau berbagi lowongan pekerjaan, grup tersebut juga dapat difungsikan untuk berdiskusi mengenai pengalaman bekerja disuatu perusahaan, tips & trik, berbagi riwayat hidup/CV, serta pertanyaan-pertanyaan yang diajukan perekrut ketika melamar suatu pekerjaan tertentu.

Informasi lowongan pekerjaan yang dibagikan di grup Telegram biasanya sudah dilengkapi dengan posisi, batas waktu pengiriman lamaran, keterangan gaji, lokasi, orang yang bisa dihubungi, persyaratan keahlian, jam kerja, penghasilan, keuntungan bergabung, tugas dan tanggung jawab pada

pekerjaan, serta waktu atau tanggal dimulainya masa kerja. Namun, informasi-informasi yang disebarakan tidak dapat terjamin keasliannya karena kurangnya pengawasan dari pengelola grup yang tidak bisa memantau ruang obrolan secara terus menerus pada grup Telegram yang ia miliki.

Sistem analisis sentimen mengambil peran yang efektif dan efisien dalam mengawasi setiap obrolan yang dikirim oleh seluruh pengguna di grup Telegram. Adanya sistem analisis sentimen akan meminimalisir informasi yang tidak sesuai atau tidak pantas ada di dalam grup, sehingga hanya informasi yang penting dan terpercaya yang akan muncul pada ruang obrolan. Dengan adanya sistem analisis sentimen, pengelola grup dapat lebih ringan dalam memantau ruang obrolan tanpa khawatir adanya anggota grup yang merasa terganggu atau bahkan mengalami kerugian akibat informasi yang tidak terpercaya. Selain itu, pengelola grup telegram dapat melihat anggota mana yang sudah lama menyebarkan informasi tidak sesuai dengan tujuan grup dibuat dan bisa menindaklanjuti pengguna tersebut dengan tepat.

2. TINJAUAN PUSTAKA

2.1. Telegram

Telegram merupakan sebuah aplikasi messenger yang berbasis cloud. Aplikasi ini dapat digunakan pada smartphone maupun laptop. Pada smartphone, Telegram dapat digunakan pada sistem operasi iPhone, iPad, Android, dan Windows Phone. Pada laptop, Telegram dapat digunakan pada sistem operasi PC, Mac, Linux, macOS, serta melalui aplikasi web browser [1]. Telegram sangat cocok dimanfaatkan untuk melakukan

pembagian informasi kepada banyak orang, karena Telegram mampu menampung banyak anggota dalam satu grup. Selain itu, grup Telegram juga dapat dibuka untuk publik sehingga dapat diakses oleh seluruh pengguna Telegram.

2.2. Bot Telegram

Sebuah layanan yang disediakan oleh Telegram, yang memiliki fungsi sesuai dengan permintaan atau perintah dari Bot Telegram memiliki tampilan berbasis dan berbasis *open source*, yaitu dapat diakses oleh siapa saja, yang disediakan oleh Telegram Messenger LLP untuk membangun bot Telegram bagi para pengembang. Beberapa kegunaan bot telegram yaitu: sebagai penghubung antara aplikasi layanan lainnya, menerima pembayaran dari pengguna Telegram, sebagai perangkat elektronik seperti pengingat, dan dapat digunakan untuk layanan sosial [2]. Mudah-mudahan akses untuk membuat bot Telegram meningkatkan banyaknya pengembang bot yang terus memberikan fitur dan variasi baru.

2.3. Python

Bahasa pemrograman tingkat tinggi yang mampu melakukan eksekusi berbagai instruksi secara langsung (interpretatif) dengan menggunakan metode pemrograman berorientasi objek, serta menggunakan semantik dinamis untuk memberikan tingkat keterbacaan *syntax* [3]. Selain itu, Python juga sering digunakan untuk pemrograman *machine learning* karena banyaknya *library* yang tersedia, contohnya Scikit Learn, Tensorflow, PyTorch, dan OpenCV.

2.4. Text Mining

Proses untuk memperoleh informasi berkualitas tinggi dari teks. Informasi berkualitas tinggi biasanya didapatkan dengan memperhatikan pola dan tren, serta mempelajari pola statistik. Pada proses *teks mining* terdapat pembobotan kata yang bertujuan untuk memberikan nilai/bobot pada term yang terdapat pada suatu dokumen [4].

2.5. Naïve Bayes Classifier

Algoritma yang dapat menerima input dalam berbagai bentuk dan memiliki kecepatan dalam memproses suatu data. Setiap kali data baru masuk, algoritma akan mengevaluasi kemungkinan untuk setiap kelas yang ada. Hasil akhir ditentukan oleh nilai tertinggi, sehingga algoritma ini dianggap efektif dalam menentukan kemungkinan hasil [5].

2.6. Crawling Data

Proses pengumpulan berbagai macam informasi yang ada di internet [6]. Crawling data sangatlah diperlukan untuk kebutuhan data statistik yang nantinya digunakan sebagai bahan analisis sistem yang akan dibuat. Data yang dikumpulkan

dapat bersumber dari database, situs web, dokumen, dan lain-lain.

2.7. Pre-processing

Proses mengubah data ke dalam format yang lebih sederhana, efektif, dan sesuai dengan kebutuhan pengguna. Parameter yang dapat digunakan sebagai acuan adalah hasil yang lebih tepat, waktu perhitungan yang lebih singkat, serta data yang lebih ringkas, namun tanpa mengubah informasi yang terdapat di dalamnya [7].

2.8. Pembobotan Kata

Suatu mekanisme untuk memberikan skor/nilai terhadap frekuensi munculnya sebuah kata dalam dokumen teks. [8] Pembobotan kata berfungsi untuk mengelompokkan dokumen teks menjadi satu cluster, sehingga terjadi peningkatan relevansi antar dokumen satu dengan lainnya. Terdapat banyak algoritma yang dapat digunakan untuk melakukan pembobotan kata, antara lain algoritma TF, Idf, RF, TR-IDF, TF. RF, dan WIDF.

2.9. Splitting Data

Splitting data adalah proses membagi dataset menjadi dua bagian, yaitu data latih dan data uji dengan proporsi yang telah ditentukan. Data latih akan digunakan untuk melatih model dalam membuat prediksi pada *machine learning*, sedangkan data uji akan digunakan untuk mengevaluasi kinerja model yang telah dibuat. Dengan melakukan *splitting data*, penelitian dalam membuat prediksi akan menjadi lebih mudah, karena data yang digunakan akan menjadi statis. Sehingga pada saat evaluasi hasil dengan data uji, hasilnya tidak akan berubah-ubah [9].

2.10. Predictive Modelling

Suatu proses pembuatan, pengujian, dan validasi model yang dilakukan untuk memprediksi probabilitas hasil terbaik. Pemilihan model didasarkan pada hasil pengujian, validasi, dan evaluasi dengan menerapkan teori deteksi untuk memperkirakan kemungkinan hasil dari sejumlah data input yang telah ditentukan [10].

2.11. Webhook

Link *URL* yang ditambahkan agar data yang dikirim dapat langsung diterima pada waktu sama dengan link *URL* yang telah ditentukan [11]. Terdapat beberapa cara untuk menggunakan webhook, antara lain *push* (informasi dikirimkan secara searah), *pipe* (penerima dapat melakukan tindakan lebih lanjut dari pengiriman informasi awal), dan *plugin* (pengiriman informasi dilakukan dua arah sesuai hak akses di plugin).

2.12. Klasifikasi Sentimen

Salah satu cabang dari *text mining* yang memiliki fungsi untuk mencari tahu polaritas dari sentimen positif, negatif, maupun netral [12]. Klasifikasi sentimen dapat membantu pemrograman dalam *machine learning* agar

sistem dapat mengelompokan, memahami, serta membaca tren data dari label atau data yang ada.

3. METODE PENELITIAN

3.1. Tahap Pengambilan Data Awal

Peneliti melakukan ekspor data pesan grup pada salah satu grup *chat* telegram. Data penelitian tersebut didapat dari grup telegram berjudul "Lowongan kerja IT" pada tanggal 6 April 2016 sampai dengan 2 April 2023. Sebanyak 120.000 data *chat* (data awal) yang digunakan pada penelitian ini akan digunakan sebagai data latih (data *training*) dan juga data uji (data *testing*).

3.2. Tahap *Labeling* Data

Data pesan grup yang telah diambil akan dilakukan pengelompokan mana yang termasuk ke dalam kata *spamming* dan mana tidak. Pesan yang termasuk mengandung kata *spamming* akan ditandai dengan label '1', sedangkan selain pesan tersebut akan dilabeli dengan '0'.

3.3. Tahap Data *Preprocessing*

Langkah ini bertujuan untuk membersihkan data mentah dari hal-hal yang membuat data sukar atau tidak penting untuk diolah pada tahap selanjutnya. Tahap ini terdiri dari 4 langkah yakni:

- Penghapusan data kosong
Data yang berupa spasi atau data baris kosong akan dihapus dari data awal/mentah agar tidak membebani kinerja dari model akhir.
- Penghapusan *mention*, email, dan seluruh simbol selain huruf dan hastag (*Tokenization*)

Mention merupakan fitur yang sering digunakan dalam mengirimkan pesan di grup Telegram agar pesan dapat tertuju ke satu/beberapa anggota grup secara spesifik. Hal tersebut harus dihilangkan dari data mentah karena data tersebut terlalu unik, sehingga model akan memiliki kecenderungan untuk menghindari *mention* tersebut padahal fitur tersebut membantu untuk mempermudah membaca pesan, bukan penanda dari terduga *spamming*.

nampak attribute dari suatu pesan yang dikirimkan seperti diantaranya *type*, link, hashtag, email, *text*, bold, http, *mention*, *send*. *Attribute* dalam bentuk JSON/struktur pesan bawaan ini harus dihapus karena data tersebut tidak akan dipertimbangkan untuk diproses dan akan mengganggu tahap proses nantinya.

- d. Penghapusan kata tidak penting (*Filtering*)
Penghapusan kata pendek merupakan salah satu cara untuk menentukan mana teks yang dianggap penting dan mana yang tidak penting. Penghapusan kata dilandaskan pada panjang kata, jika kata tidak lebih dari 3 maka kata akan terhapus dari data mentah.
- e. Pengubahan menjadi kata asal (*Stemming*)
Semua kata dalam baris data mentah akan dilakukan pengkonversian oleh *Stemmer* sehingga data mentah akan mengandung kata asal dari kata yang diproses.
- f. Penghapusan kata umum yang tidak memiliki makna (*Stopwords*)

Dalam suatu tata bahasa tentunya ada kata – kata penghubung. Kata penghubung adalah untuk menghubungkan atau menggabungkan antar-klausa, antar-kalimat dan antar-paragraf. Karena kata penghubung tidak digunakan untuk menjelaskan maksud dari inti pesan, maka kata penghubung dapat dikecualikan dari data mentah. Ada 758 kata yang menjadi dasar pengecualian.

3.4. Visualisasi Data Hasil Preprocessing

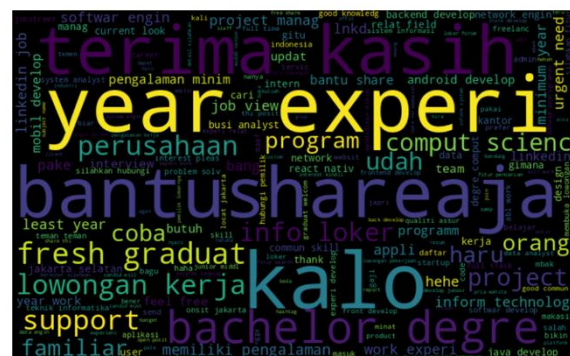
Hasil dari tahap preprocessing dapat dilakukan menggunakan library python bernama wordcloud. Library ini dapat menyusun kata – kata yang diinginkan dengan mempertimbangkan ukuran kata berdasarkan banyaknya kata yang berada pada data tersebut. Berikut hasil pengelompokan data yang dianggap sebagai spam atau data berlabel ‘1’ dari data mentah:

Gambar 1. menunjukkan kata-kata yang dianggap tidak masuk ke dalam kata-kata yang diduga pesan *spamming*. Kata *spamming* yang dimaksud adalah satu kata atau gabungan kata yang sering digunakan oleh *spammer* untuk menyebarkan informasi diluar tujuan dibuatnya grup *sharing* lowongan kerja.

Tabel 1. Proses *Tokenization*

Teks sebelum proses	Teks sesudah proses
["Privy membuka ",{"type":"bold","text":"Interns hip Frontend Engineer WFO/Onsite in Jogja"}],"\n\nUntuk apply bisa melalui : ",{"type":"link","text":"http://bi t.ly/PrivyIDCareer"}]	Privy membuka type bold text Internship Frontend Engineer WFO Onsite in Jogja Untuk apply bisa melalui type link text http bit ly PrivyIDCareer

- c. Penghapusan *attribute chat* dari data telegram
Saat melakukan export data, telegram akan memberikan struktur dari chat sehingga akan



Gambar 1. Wordcloud kata non-spamming

Dan Gambar 2. menunjukan data yang diduga masuk ke dalam pesan grup yang bersifat *spamming*.



Gambar 2. Wordcloud kata *spamming*

3.5. Pembuatan *Modelling*

Data yang sudah bersih akan masuk ke tahap analisis. Dimana data mentah akan dibagi menjadi 2 bagian yakni data training dan data testing. Ukuran dari pembagian data mentah adalah 80% data training (96000 pesan) dan 20% data testing (24000 pesan). Dengan bantuan dari library NLTK, model dapat dibuat secara instan dengan memasukkan data training ke dalam fungsi *MultinomialNB().fit()* pada library *sklearn.naive_bayes*.

3.6. Integrasi Webhook

Telegram menyediakan fitur developer berupa webhook yang berguna untuk mengirim pesan yang dikirimkan oleh pengguna telegram dari suatu obrolan/grup. Webhook memungkinkan para developer atau pengembang suatu aplikasi untuk menerima setiap pesan yang dikirimkan oleh anggota grup. Sehingga pesan dapat ditangkap oleh server dan diprediksi oleh model yang sudah di-export dari proses modeling sebelumnya. Hasil prediksi akan menentukan apakah pesan layak untuk masuk ke dalam grup atau tidak. Jika tidak layak untuk masuk ke dalam grup, bot Telegram akan mengirimkan notifikasi bahwa chat tersebut tidak pantas dan akan menghapus pesan tersebut dari grup Telegram.

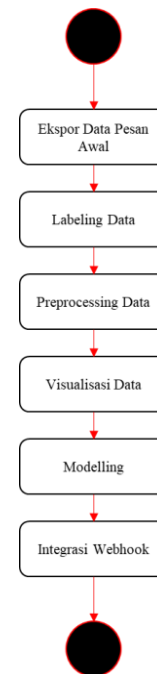
4. HASIL DAN PEMBAHASAN

4.1. Implementasi Algoritma

Implementasi algoritma penelitian melewati proses tahapan pada gambar 3.

Pada tahap *preprocessing* peneliti melakukan pembersihan data crawling menjadi data yang lebih sederhana dan memindahkan seluruh hasil dari *crawling data* sebelumnya menjadi data tabel. Selanjutnya, pada tahap *labelling*, peneliti memberikan label secara manual pada data yang telah didapatkan. Label “0” diberikan untuk pesan yang sesuai dengan topik obrolan lowongan pekerjaan, sebaliknya label “1” diberikan untuk pesan yang tidak sesuai. Lalu, pada tahap *splitting data*, peneliti membagi data yang telah di beri label

menjadi 2 bagian yaitu data latih sebanyak 80% dan data uji sebanyak 20%. Pada tahap *modelling*, peneliti melakukan pengujian dan evaluasi pada hasil dari sistem yang telah dibuat. Terakhir, pada tahap integrasi *webhook* peneliti melakukan pemasangan sistem yang telah dibuat ke aplikasi Telegram.



Gambar 3. Tahapan Pemrosesan Data

4.2. Pengujian Tingkat Akurasi Metode Klasifikasi Naïve Bayes

Pengujian tingkat akurasi menggunakan metode Naïve Bayes ini dilakukan menggunakan *library* Python NLTK (*Natural Language Toolkit*) untuk memperoleh tingkat akurasi dari sistem yang telah dirancang.

Beberapa perhitungan yang dilakukan peneliti adalah *precision* prediksi positif, *precision* prediksi negatif, *recall* true positif, *recall* true negatif, dan akurasi sistem yang telah dirancang.

Berikut hasil perhitungan yang didapatkan oleh peneliti.

Tabel 2. Tabel Perhitungan (*Confusion Matrix*)

	<i>True Positif</i>	<i>True Negatif</i>	<i>Class Precision</i>
<i>Pred Positif</i>	20093	1432	93,34%
<i>Pred Negatif</i>	715	1760	70,11%
<i>Class Recall</i>	96,56%	55,13%	

Precision prediksi positif :

$$\begin{aligned} & \frac{TP}{(TP + FP)} \times 100\% \\ &= \frac{20093}{(20093 + 1432)} \times 100\% \\ &= 93.34\% \end{aligned}$$

Precision prediksi negatif:

$$\frac{TN}{(TN + FN)} \times 100\%$$

$$= \frac{1760}{(1760 + 2475)} \times 100\%$$

$$= 70.11\%$$

Recall true positif

$$\frac{TP}{(TP + FN)} \times 100\%$$

$$= \frac{20093}{(20093 + 715)} \times 100\%$$

$$= 96.56\%$$

Recall true negatif

$$\frac{TN}{(TN + FT)} \times 100\%$$

$$= \frac{1760}{(1760 + 1432)} \times 100\%$$

$$= 55.13\%$$

Akurasi sistem

$$\frac{(TP+TN)}{(TP+TN+FP+FN)} \times 100\%$$

$$= \frac{(20093+1760)}{(20093+1760+1432+715)} \times 100\%$$

$$= 91.05\%$$

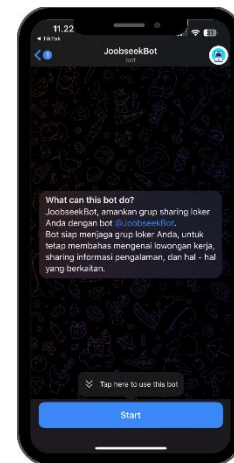
4.3. Hasil Analisis

Berdasarkan hasil pengujian tingkat akurasi dengan metode klasifikasi Naïve Bayes yang dilakukan oleh peneliti, didapatkan hasil akurasi sebesar 91.05%. Setelah melakukan pengujian, peneliti melakukan integrasi *webhook* pada aplikasi telegram, berikut desain dan implementasi yang dihasilkan oleh peneliti.



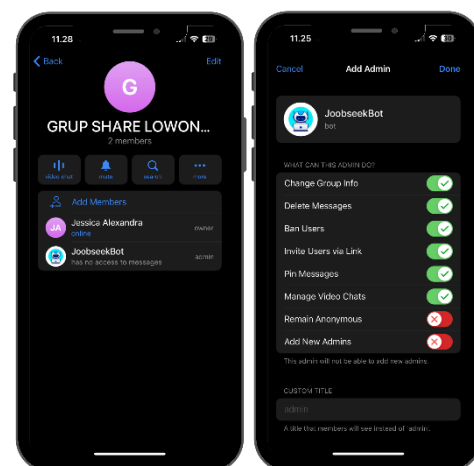
Gambar 3. Tampilan profil jooobseekbot

Penerapan sistem analisis yang dibuat menghasilkan desain bot seperti pada Gambar 3. Terdapat *profile picture* dan keterangan tentang bot.



Gambar 4. Tampilan roomchat jooobseekbot

Pada Gambar 2, terlihat tampilan *roomchat* dari Jooobseekbot. Informasi pertama yang muncul pada *roomchat* merupakan keterangan sebelumnya yang ditulis pada profil bot.



Gambar 5. Mengaktifkan jooobseekbot dalam grup

Untuk mengaktifkan Joobseekbot di dalam grup, pertama-tama pengguna perlu mengundang bot untuk masuk ke dalam ruang obrolan grup. Setelah itu, pengguna perlu mengatur Joobseekbot sebagai admin. Jika sudah berhasil, maka bot akan otomatis aktif dan dapat melakukan pemantauan pada grup tersebut.



Gambar 6. Contoh pesan diterima

Ketika pesan yang dikirimkan oleh pengguna tidak terdeteksi spam, maka pesan akan diterima dan akan muncul secara normal ke seluruh anggota.



Gambar 7. Contoh pesan ditolak

Ketika pesan yang dikirimkan pengguna terdeteksi spam, maka pesan akan ditolak oleh bot. Penolakan pesan akan ditindaklanjuti dengan adanya penghapusan pesan dan pesan peringatan dari Joobseekbot. Peringatan dari Joobseekbot akan otomatis terhapus setelah 8 detik ditampilkan.

5. KESIMPULAN DAN SARAN

Berdasarkan hasil dari pengujian yang telah dilakukan oleh peneliti, maka didapatkan beberapa kesimpulan pada penelitian ini, yaitu penggunaan metode Naïve Bayes pada Analisis Sentimen Pemanfaatan Obrolan Grup Telegram Berbagi

Informasi Lowongan Kerja didapatkan akurasi sebesar 91.05%. Tingkat akurasi sangat dipengaruhi oleh banyaknya sampel data dan proses labelling yang dilakukan oleh peneliti. Semakin banyak data yang diambil serta semakin detail proses labelling, maka semakin tinggi tingkat akurasi yang dihasilkan. Pada penelitian yang telah dilakukan ini masih terdapat beberapa kekurangan, sehingga terdapat saran yang diajukan peneliti untuk pengembangan selanjutnya, yaitu diharapkan adanya peningkatan jumlah data latihan agar hasil akurasi meningkat dan sistem berjalan dengan lebih baik, dapat memahami kata-kata yang bermakna ganda, hingga membaca adanya salah penulisan ataupun ejaan.

DAFTAR PUSTAKA

- [1] K. Qamar, and S. Riyadi, "Efektivitas *Blended Learning* Menggunakan Aplikasi Telegram", *Jurnal Ilmu Tarbiyah.*, pp.5, Vol 6, No 1, January 2017.
- [2] R. Parlika, H. Khairono, H. A. Kusuma, D. A. Setyawan, "Pemanfaatan Bot Telegram Sebagai E-Learning Ujian Berbasis File" *Jurnal Informatika Polinema*, pp.67, vol. 7, no. 4, August 2021.
- [3] M. K. Rahmadhika, and A. M. Thantawi, "Rancang Bangun Aplikasi *Face Recognition* Pada Pendekatan CRM Menggunakan *Open CV* dan Algoritma *Haarcascade*" *Jurnal IKRA-ITH Informatika.*, pp.112, Vol 5, No 1, March.2021.
- [4] Deolika, K. Kusri, E. T. Luthfi, "Analisis Pembobotan Kata Pada Klasifikasi *Text Mining*" *Jurnal Teknologi Informasi.*, pp.179, Vol 3, No 2, 2019.
- [5] M. R. Handoko, and N. Neneng, "Sistem Pakar Diagnosa Penyakit Selama Kehamilan Menggunakan Metode *Naive Bayes* Berbasis Web" *Jurnal Teknologi dan Sistem Informasi.*, pp.50, Vol 2, No 1, 2021.
- [6] I. Ramadhan, and H. Sastramihardja, "Pemanfaatan *Web Crawler* Dalam Mengumpulkan Informasi Melalui Internet" *Konferensi Nasional Sistem Informasi.*, pp.548, 8-9 March.2018.
- [7] Saifullah, M. Zalis, Zakaria, and R. W. Sembiring, "Analisa Terhadap Perbandingan Algoritma *Decision Tree* dengan Algoritma *Random Tree* untuk *Pre-Processing Data*" *Jurnal Sains Komputer & Informatika.*, pp.182, Vol 1, No 2, September.2017.
- [8] N. M. S. Hadna, P. I. Santosa, and W. W. Winarno, "Studi Literatur Tentang Perbandingan Metode untuk Proses Analisis Sentimen di Twitter" *Seminar Nasional Teknologi Informasi dan Komunikasi.*, 18-19 March.2016
- [9] V. A. P. Putri, A. B. Prasetyo, and D. Eridani, "Perbandingan Kinerja Algoritma *Naive Bayes* dan *K-Nearest Neighbor (KNN)* Untuk Prediksi Harga Rumah" *Jurnal Ilmiah Teknik Elektro.*, pp. 164, October.2022.
- [10] I. Iskandar, "Implementasi *Predictive Modelling* Dalam Memprediksi Besarnya Penggunaan Listrik

- Rumah Tangga” *Jurnal Riset Komputer.*, pp. 622, Vol V. No 6, December.2018.
- [11] T. Istiana, R. I. A., G. S. B Dharmawan, and B. Prakoso, “Pengembangan Sistem Siseminasi Prakiraan Cuaca Menggunakan Aplikasi Bot Telegram dengan Metode Webhook” *Elektron Jurnal Ilmiah.*, pp.42, Vol XII, No 1, Juny.2020.
- [12] S. Suryono, E. Utami, and E. T. Luthfi, “Klasifikasi Sentimen Pada Twitter dengan *Naive Bayes Classifier*” *Jurnal Ilmiah Bidang Teknologi, Angkasa.*, pp.89, Vol X, No 1, May.2018.