

# ANALISIS SENTIMEN PADA TWITTER TERHADAP PROGRAM KARTU PRA KERJA DENGAN RECURRENT NEURAL NETWORK

Rosit Sanusi<sup>1</sup>, Femi Dwi Astuti<sup>2</sup>, dan Indra Yatini Buryadi<sup>3</sup>

<sup>1-3</sup>Informatika, STMIK AKAKOM

Email: sanuusirosyid@gmail.com<sup>1</sup>, femi@akakom.ac.id<sup>2</sup>, indrayatini@akakom.ac.id<sup>3</sup>

## Abstrak

Twitter menjadi salah satu media sosial dengan jumlah pengguna aktif paling banyak di Indonesia. Dengan berlakunya program kartu prakerja sejak pendaftaran gelombang pertama hingga sekarang, banyak pengguna twitter di Indonesia yang menyampaikan pendapat dan gagasan mengenai program kartu prakerja melalui twitter. Oleh karena itu penelitian ini mencoba untuk menganalisa tweet berbahasa Indonesia yang membicarakan mengenai program kartu prakerja yang ditandai dengan kata kunci prakerja dalam tweet tersebut. Analisis sentimen dilakukan dengan menggunakan metode Recurrent Neural Network (RNN) dengan Long Short Term Memory (LSTM). Dalam penelitian ini data yang digunakan di crawling menggunakan bantuan Twitter API yang diambil pada periode bulan April 2020 sampai Januari 2021 sebanyak 4122 tweet. Penelitian menghasilkan sebuah sistem yang mampu melakukan klasifikasi sentimen (positif, netral dan negatif) terhadap sebuah tweet. Tingkat akurasi dari proses training yang didapat sebesar 95,66% serta tingkat akurasi dari proses testing sebesar 64,48%. Beberapa kendala dalam proses analisis sentimen adalah data untuk pembuatan model tidak seimbang sehingga menyebabkan overfitting.

**Kata Kunci:** Analisis Sentimen, Deep Learning, RNN, LSTM, API

## Abstract

Twitter is one of social media with the most active users in Indonesia. With the implementation of the pre-employment card (Kartu Prakerja) program ever since the first batch of registration until recently, many twitter users in Indonesia have expressed their opinions and ideas about the pre-employment card program via twitter. Therefore, this study aims to analyze Indonesian-language tweets talking about the pre-employment card program identified by the "prakerja" or pre employment keyword within their tweet. Sentiment analysis was performed using Recurrent Neural Network (RNN) method with Long Short Term Memory (LSTM). A total of 4122 tweets data acquired from April 2020 to January 2021 was crawled using Twitter API. The research resulted in a system capable of classifying sentiments (positive, neutral and negative) on a tweet. The accuracy level of the training process obtained is 95.66% and the accuracy level of the testing process is 64.48%. Some of the obstacles in the sentiment analysis process are that the data for modeling is not balanced, causing overfitting.

**KeyWords :** Sentiment Analysis, Deep Learning, RNN, LSTM, API

## I. PENDAHULUAN

Sosial Media merupakan salah satu media yang paling banyak digunakan orang Indonesia. Hal ini dibuktikan dengan riset yang telah dilakukan oleh Hootsuite: *Indonesian Digital Report 2020* pada bulan Januari 2020 menunjukkan sebanyak 160 juta dari 272,1 juta penduduk Indonesia merupakan pengguna aktif sosial media. Dari hasil riset yang sama salah satu platform yang masuk jajaran 5 teratas platform paling sering digunakan adalah *Twitter* dengan persentase 56%.

Mengacu dari website resmi kartu prakerja, Program Kartu Prakerja adalah program pengembangan kompetensi kerja dan kewirausahaan yang ditujukan untuk pencari kerja, pekerja/buruh yang terkena pemutusan hubungan kerja, dan/atau pekerja/buruh yang membutuhkan peningkatan kompetensi, termasuk pelaku usaha mikro dan kecil. Walaupun program ini merupakan salah satu program kerja pemerintah yang telah dimulai sejak bulan April 2020 lalu akan tetapi masih ada saja masalah yang menyertainya seperti insentif terlambat cair, membludaknya calon peserta yang menyebabkan tidak bisa nya akses website untuk melakukan pendaftaran [1], salah sasaran, dll.

Dengan muncul nya masalah – masalah tersebut, maka dibuat lah sebuah analisis sentimen mengenai bagaimana pandangan masyarakat terhadap program kartu prakerja di platform media sosial *Twitter*. Dipilihnya *Twitter* sebab banyak digunakan masyarakat Indonesia [2] yang menggunakan nya serta latar belakang pengguna nya yang beragam.

Digunakannya metode *Recurrent Neural Network* dengan *Long Short Term Memory* untuk melakukan analisis sentimen pada tweet dengan topik program kartu prakerja didasarkan pada hasil penelitian yang telah dilakukan mengenai Analisis Sentimen Tempat Wisata Di Yogyakarta. Menggunakan Metode *Recurrent Neural Network* Dengan *Long Short Term Memory* yang menghasilkan tingkat akurasi pelatihan sebesar 99,31% [4]. Hasil penelitian lain yang dilakukan mengenai Analisis Sentimen Konten Sosial Media Instagram STMIK Akakom Yogyakarta yang menghasilkan tingkat akurasi pengujian sebesar 65% dan tingkat akurasi penerapan sebesar 79,46% [4]. Dari hasil penelitian ini lah diketahui bahwa metode ini mempunyai tingkat akurasi yang memuaskan.

II. METODE

A. Analisis Sentimen

Analisis sentimen atau biasa disebut *opinion mining* merupakan bidang studi yang menganalisa pendapat manusia, sentimen, evaluasi, penilaian, *attitude*, dan emosi terhadap entitas tertentu misalnya produk, layanan, organisasi, individu, isu, event, dan topik [5]. Pada Analisis sentimen terdiri dari tiga level analisis meliputi level dokumen, level kalimat serta level entitas dan aspek.

B. Twitter

Twitter merupakan salah satu sosial media dengan jumlah pengguna paling banyak diseluruh dunia [2]. Twitter termasuk kedalam *micro-blogging* yaitu sistem yang mampu untuk mengirim dan menerima *post* pendek atau biasa disebut *tweet* yang mempunyai panjang maksimal 140 karakter.

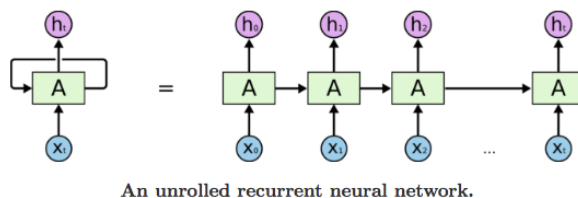
Cara kerja dari twitter sendiri adalah pengguna mengikuti pengguna lain, jika kita mengikuti seseorang maka kita dapat melihat *tweet* orang tersebut di laman twitter kita atau *timeline*. Pengguna bisa memilih untuk mengikuti orang atau organisasi dengan berbagai latar belakang. Selain itu pengguna bisa menuliskan *tweet* kita sendiri atau melakukan *posting* Kembali *tweet* orang lain (*retweet*).

C. Program Kartu Pra Kerja

Menurut Peraturan Presiden (PERPRES) Republik Indonesia Nomor 36 Tahun 2020 Tentang Pengembangan Kompetensi Kerja Melalui Program Kartu Prakerja pasal 1 ayat 1 yaitu “Program Kartu Prakerja adalah program pengembangan kompetensi kerja yang ditujukan untuk pencari kerja, pekerja/buruh yang terkena pemutusan hubungan kerja, dan/atau pekerja/ buruh yang membutuhkan peningkatan kompetensi”. PERPRES ini ditetapkan pada tanggal 26 Februari 2020 dan mulai tanggal 28 Februari 2020.

D. Recurrent Neural Network (RNN)

Recurrent Neural Network atau RNN adalah keluarga dari *neural network* yang digunakan untuk memproses data. Berbeda dengan *Convolutional Neural Network (CNN)* yang digunakan untuk pemrosesan data seperti gambar, RNN sering digunakan untuk memproses nilai yang bersifat sekuensial  $x_1, \dots, \dots, x_n$  [6], struktur RNN dapat dilihat pada Gambar1.

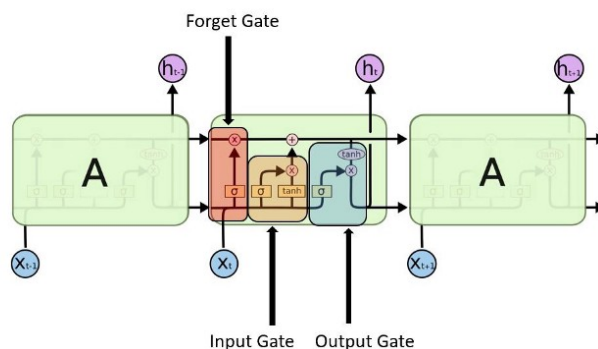


Gambar 1: Struktur RNN

Gambar 1  $x_t$  merupakan masukan dan  $h_t$  merupakan keluaran, sederhananya bentuk konseptual dari RNN (gambar sebelah kiri) terlihat seperti 1 jaringan dengan pengulangan. Pengulangan ini membuat informasi dapat dilewatkan dari satu langkah pada jaringan ke selanjutnya. Jadi RNN dapat dijabarkan sebagai banyak salinan dari jaringan yang sama (gambar sebelah kanan).

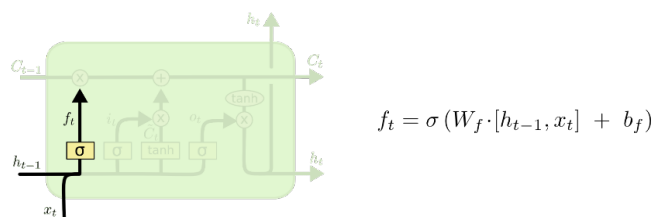
E. Long Short Term Memory

LSTM atau *Long Short Term Memory* merupakan bentuk spesial dari RNN yang dapat melakukan pembelajaran pada dependensi jangka panjang (*long-term dependencies*). Model ini diperkenalkan oleh Hochreiter dan Schmidhuber pada tahun 1997 [6].



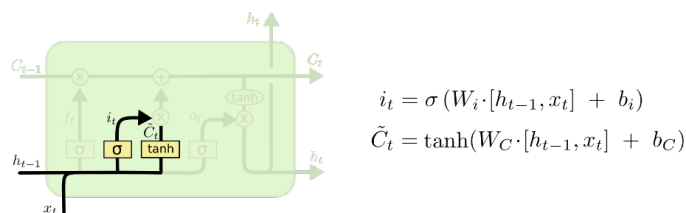
Gambar 2: Struktur LSTM

Pada Gambar 2 dapat dilihat struktur dari LSTM. Seluruh RNN mempunyai bentuk rangkaian jaringan syaraf yang berulang. Begitu pula dengan LSTM namun disini LSTM mempunyai tambahan fitur pada gerbang sel yaitu *forget gate*, *input gate*, dan *output gate*. Langkah pertama dalam LSTM adalah menentukan informasi yang akan dibuang atau dilupakan dalam sel, hal ini merupakan kewenangan dari *Forget Gate* yang mempunyai fungsi untuk membuat keputusan informasi apa yang akan dibuang dari sel. Disini layer akan memperhatikan  $h_{t-1}$  dan  $x_t$  yang nantinya akan menghasilkan keluaran 0 atau 1. Jika keluaran 0 menunjukkan bahwa informasi akan dilupakan sedangkan keluaran 1 menunjukkan bahwa informasi tidak akan dilupakan seperti yang tampak pada Gambar 3.



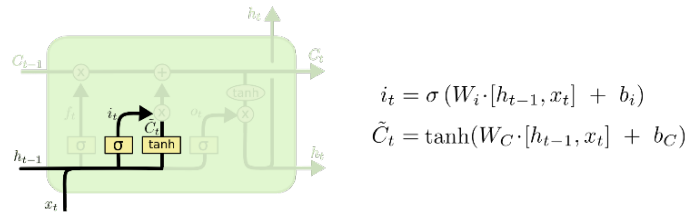
Gambar 3: Struktur Forget Gate

Langkah selanjutnya adalah menentukan informasi baru yang akan disimpan dalam sel. Langkah ini terdiri dari bagian, pertama layer sigmoid disebut "*input gate layer*" yang akan menentukan nilai mana yang akan di perbaharui. Selanjutnya sebuah *layer tanh* yang akan membuat vektor sebagai nilai kandidat yang baru ( $\tilde{C}_t$ ), yang dapat ditambahkan kedalam *state*. Berikutnya kedua layer ini akan dikombinasikan untuk memperbaharui *state* seperti pada Gambar 4.



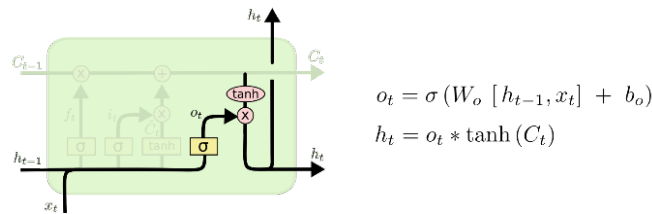
Gambar 4: Struktur Input Gate

Berikutnya, *state* lama akan diperbaharui ( $C_{t-1}$ ) menjadi *state* baru  $C_t$ . dimana *state* lama akan dikalikan dengan  $f_t$  dengan mengabaikan informasi yang telah dilupakan sebelumnya. Lalu, keluaran dari  $i_t$  ditambahkan dengan, sehingga akan memperbaharui *state* dengan nilai baru ( $C_t$ ) menurut jaringan saraf relevan seperti pada Gambar 5.



Gambar 5: Menghitung State Baru ( $C_t$ )

Langkah terakhir adalah menentukan keluarannya. Pertama *layer sigmoid* akan dijalankan untuk bagian dari sel yang akan dikeluarkan. Kemudian sel tersebut akan dilewatkan melalui *layer tanh* (untuk mendorong nilai antara -1 dan 1) dan mengalikannya dengan *output* dari *sigmoid gate* seperti Gambar 6.

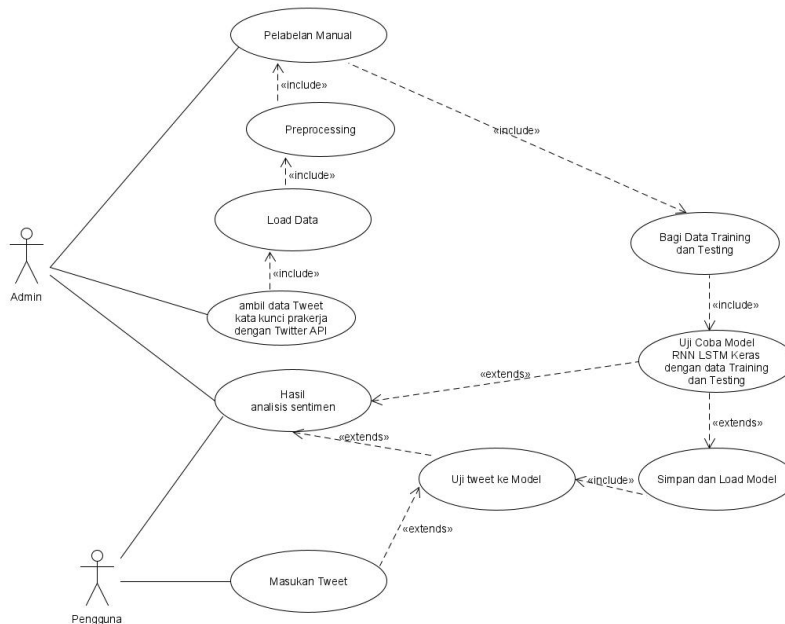


Gambar 6: Struktur Output Gate

### III. HASIL DAN PEMBAHASAN

#### A. Use Case Diagram

Use case diagram pada penelitian ini dapat dilihat pada Gambar 7. Pada Gambar 7 dapat dilihat bahwa sistem dibuat untuk 2 user yaitu admin dan pengguna umum.



Gambar 7: Use Case Diagram

#### B. Pengumpulan Data

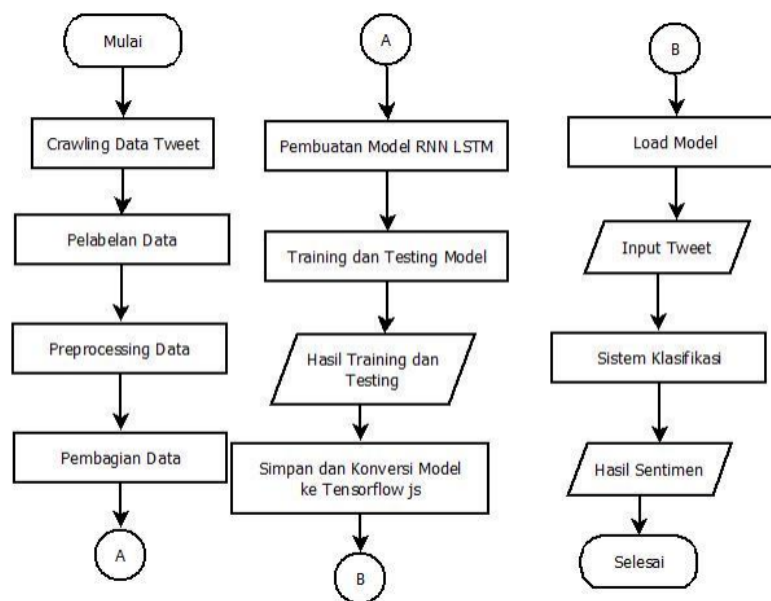
Proses pengumpulan data dilakukan dengan langkah sebagai berikut :

- 1) Membuat akun *twitter developers* untuk mendapat *API key*, *API secret*, *access token* dan *access secret*.
- 2) Masuk ke *google colab* dengan alamat <https://colab.research.google.com/>.

- 3) Memasang *library searchtweet* dengan perintah `!pip install searchtweets`.
- 4) Membuat *file config* yang berisi *credential* berupa *account\_type*, *endpoint*, dan *bearer\_token* dan mengupload ke *Google colab*.
- 5) Membuat *rule* pencarian *query* tiap bulan dengan kata kunci prakerja tanpa *hyperlink* dan berbahasa Indonesia.
- 6) Membuat batas maksimal pencarian *query* dan menginputkan *rule*.
- 7) Melakukan *request* ke *Twitter API* menggunakan *rule* dan batas yang telah didefinisikan dan hasilnya disimpan kedalam *list*.
- 8) *List response* digabung menjadi 1 *Dataframe*.
- 9) *Dataframe* dicek apakah terdapat didalamnya data yang duplikat jika ada maka data tersebut akan di hapus.
- 10) *Dataframe* terakhir inilah yang akan disimpan kedalam format *xlsx* untuk selanjutnya didownload untuk dilakukan pelabelan.

### C. Alur Penelitian

Gambaran alur proses sistem bekerja dapat dilihat pada Gambar 8.



Gambar 8: Alur Penelitian

### D. Implementasi Sistem

Tahap implementasi merupakan tahap penerapan dari rancangan dan pengujian agar didapat hasil penelitian yang baik. Berikut tahapan implementasi yang dilakukan.

#### 1) Crawling Data Tweet

Pada tahap ini dilakukan pengumpulan data *tweet* dengan menggunakan *library searchtweets*. *Library* menggunakan *twitter API* untuk mengakses *tweet* pengguna. Langkah yang dilakukan yaitu :

- a. Membuat developer account
- b. Membuat project dan app
- c. Proses crawling data

#### 2) Pelabelan Data

Ada 3 kelas yang digunakan sebagai berikut :

- a. Positif : *tweet* yang mengandung informasi mendukung, tidak menyangkal, tidak mengandung kata sangkalan (tidak, bukan).
- b. Netral : *tweet* yang bukan merupakan opini atau tidak berpihak (bukan positif atau negatif).
- c. Negatif : *tweet* yang mengandung kata sangkalan, menyangkal, tidak mendukung atau kurang baik.

Proses ini dilakukan guna menentukan kelas awal sebagai acuan pada saat dilakukan pelatihan model. Contoh pelabelan data dapat dilihat pada Tabel I.

Tabel I: Pelabelan Data

Teks	Label
@hrdbacot Min, tmn saya ikut prakerja. Dia gak ikut pelatihan online. Tapi insentifnya bisa cair. Apa emang bisa gitu?	0
@wahyu_lestarii @ganjarpranowo Bingung enyong arep ra mudik tapi duit ntek, mudik ra ulih aku kon piye jal lagi njajal gawe kartu prakerja kie koe dingendi ? Aku nang mburi pasar tambun	0
@helmitology Selamat berpuasa buat para pejuang kartu prakerja yg tak kunjung lolos	1
@UviaVia Sejauh ini bagis kak, pengajarnya memang yg kompeten dan pelaku bahkan trainee, terus ad pilihan paket juga tinggal kategori keahliannya mau yg mana, apa lg kl dapet prakerja, lumayan tu di bayarin semua paket nya	1
@KRMTRoySuryo2 Rombongan tol langit biasa omdo?buat apa kartu prakerja ,langsung ajah kasih kerjaan di tiap bidang keahlian masing”, cuma mau mengeluarkan anggaran yg tdk jls ,	2
Temen2ku banyak yang nganggur pak @jokowi , ikutan test prakerja gagal mulu ceunah, kayaknya kita belum butuh TKA deh pak .. #TolakTKAChina	2

3) Praprocessing Data

Proses ini bertujuan untuk mengolah data mentah agar siap digunakan seperti menghilangkan tanda baca, konversi huruf kecil, *stemming* dan menghilangkan *stopword*. Pada proses menggunakan beberapa *library* seperti re untuk dan Sastrawi untuk menghilangkan *stopword* dan melakukan *stemming*.

4) Pembagian Data

data dibagi menjadi data *training* dan data *testing* dengan proporsi 80% dari total data akan menjadi data *training* dan 20% nya menjadi data *testing*.

5) Pembuatan Model

Pada model ini ada beberapa layer yang digunakan dari *library keras.layers*, sebagai berikut

- a) *Embedding, layer* ini digunakan untuk mengkonstruksi sekuensial kata. Disini didefinisikan parameter berupa panjang *input* vektor, panjang *output* vektor, dan panjang input maksimal sekuensi kata.
- b) *BatchNormalization, layer* ini akan melakukan normalisasi terhadap *input*.
- c) *LSTM, layer* ini merupakan inti dari penelitian ini yang didefinisikan dengan 128 unit.
- d) *Dense, layer* ini digunakan untuk merepresentasikan *neuron* dalam model. Ada *layer 2 dense* yang digunakan disini. *Layer dense* pertama punya 32 unit dengan *activation function* relu dan *layer dense* terakhir punya 3 unit dengan *activation function* softmax karena merepresentasikan jumlah kelas.
- e) *Dropout, layer* ini berfungsi untuk mengeluarkan Sebagian *neuron* selama proses pelatihan dengan tujuan *overfitting* dapat dicegah selama pelatihan berlangsung.

6) Pelatihan Model

pelatihan dari model menggunakan fungsi *fit()* yang berisi *X\_train* (teks data *training*), *y\_train* (label data *training*) dan menggunakan *validation* data *X\_test* dan *y\_test*. Dalam model ini digunakan juga *callback* yang berguna untuk mengawasi *val\_loss* dari *training step* dan akan menghentikan secara paksa proses pelatihan jika *val\_loss* tidak mengalami penurunan yang signifikan walaupun nanti belum mencapai epoch ke 50.

7) Pengetesan Model

Ditahap ini sebanyak 505 data diujikan kepada model pelatihan, dari hasil pengujian nanti akan diketahui tingkat akurasi dari model yakni seberapa besar loss dan akurasi nya.

8) *Import Model*

Pada langkah ini model yang telah dilatih di import menggunakan *library tensorflow.js*, dengan maksud menggunakan *library* ini karena nanti sistem klasifikasi nya dibuat menggunakan *javascript*.

IV. PEMBAHASAN

A. *Crawling Data*

Hasil *crawling* yang didapat disimpan kedalam file dengan ekstensi *xlsx*. Didalam file ini berisi atribut dan nilai hasil *crawling*, berikut struktur atribut hasil *crawling* :

- 1) *Screename* : nama pengguna *twitter*
- 2) *Type* : janis *tweet* (*tweet/retweet*)
- 3) *Text* : konten teks yang ditulis dan dipost pengguna

B. Hasil Praprocessing

Dari proses ini akan dihasilkan informasi berupa data terstruktur, dimana data telah dihilangkan noise dan siap digunakan untuk proses pelatihan. Praprocessing dapat dilihat pada Tabel II. Contoh tweet : Teks : @sarahputriandi Mungkin jika kasusnya saldo masih ada 500k dan sudah 30hari otomatis kan expired yah, nanti bakalan di Orp in lagi saldonya. tapi baru asumsi aku saja ya karena merujuk ke FAQ di prakerja

Tabel II: Hasil Praprocessing

Proses	Text
Menghapus Username dan Hashtag	Mungkin jika kasusnya saldo masih ada 500k dan sudah 30hari otomatis kan expired yah, nanti bakalan di Orp in lagi saldonya. tapi baru asumsi aku saja ya karena merujuk ke FAQ di prakerja
Menghapus Tanda Baca	Mungkin jika kasusnya saldo masih ada 500k dan sudah 30hari otomatis kan expired yah nanti bakalan di Orp in lagi saldonya tapi baru asumsi aku saja ya karena merujuk ke FAQ di prakerja
Menghapus Angka	Mungkin jika kasusnya saldo masih ada k dan sudah hari otomatis kan expired yah nanti bakalan di rp in lagi saldonya tapi baru asumsi aku saja ya karena merujuk ke FAQ di prakerja
Mengubah Ke Huruf Kecil	mungkin jika kasusnya saldo masih ada k dan sudah hari otomatis kan expired yah nanti bakalan di rp in lagi saldonya tapi baru asumsi aku saja ya karena merujuk ke faq di prakerja
Menghapus Stopword	kasusnya saldo otomatis expired yah rp in saldonya asumsi merujuk faq prakerja
Melakukan Stemming	kasus saldo otomatis expired yah rp in saldo asumsi rujuk faq prakerja

C. Model

1) Pembuatan Model

Model Summary yang merupakan ringkasan model yang berisi informasi mengenai layer, ukuran output layer, bobot tiap layer, dan total parameter dari model dapat dilihat pada Gambar 9.

```

Model: "sequential"
Layer (type)                Output Shape                Param #
-----
embedding (Embedding)      (None, 50, 16)             128000
batch_normalization (Batch Normalization) (None, 50, 16)             64
lstm (LSTM)                 (None, 128)                74240
dense (Dense)               (None, 32)                 4128
dropout (Dropout)          (None, 32)                 0
dense_1 (Dense)            (None, 3)                  99
-----
Total params: 206,531
Trainable params: 206,499
Non-trainable params: 32
    
```

Gambar 9: Model Summary

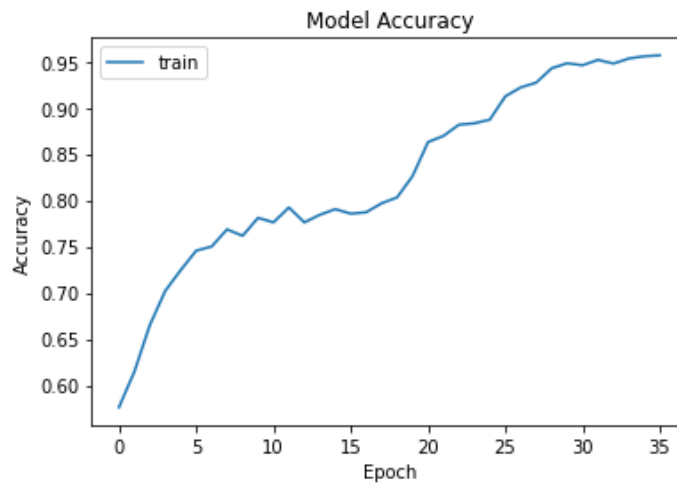
2) Gambar 10 merupakan *training step* dari model.fit(), yang mana didapat hasil akurasi dan *loss training* serta *validation loss* dan akurasinya. Proses pelatihan ini berhenti di epoch ke 36 dengan waktu pelatihan tiap epoch antara 4 – 9 detik. Dari hasil pelatihan didapat tingkat akurasi training sebesar 0.9566 dan loss sebesar 0.1999 dari 35 epoch. kemudian untuk tingkat akurasi validasi sebesar 0.6448 dengan loss sebesar 1.5922 dari 36 epoch.

```

Epoch 1/50
52/52 - 8s - loss: 1.3388 - accuracy: 0.5763 - val_loss: 1.2544 -
val_accuracy: 0.6145
Epoch 2/50
52/52 - 5s - loss: 1.0599 - accuracy: 0.6148 - val_loss: 1.0490 -
val_accuracy: 0.6145
. . . . .
Epoch 34/50
52/52 - 4s - loss: 0.2238 - accuracy: 0.9533 - val_loss: 1.6761 -
val_accuracy: 0.6412
Epoch 35/50
52/52 - 4s - loss: 0.2025 - accuracy: 0.9557 - val_loss: 1.8621 -
val_accuracy: 0.6206
Epoch 36/50
52/52 - 4s - loss: 0.1999 - accuracy: 0.9566 - val_loss: 1.5922 -
val_accuracy: 0.6448
    
```

Gambar 10: Training Steps

3) Gambar 11 adalah grafik yang menggambarkan laju akurasi dari proses pelatihan model. Dari grafik diatas dapat diketahui bahwa tingkat akurasi bisa dikatakan tinggi sebab grafik menunjukan sampai pada titik mendekati 100%. Grafik ini dibuat menggunakan *library matplotlib.pyplot*.



Gambar 11: Grafik loss

*D. Hasil Testing Model*

```

Evaluate on test data
26/26 - 0s - loss: 1.5922 - accuracy: 0.6448
test loss, test acc: [1.5922175645828247, 0.6448484659194946
    
```

Gambar 12: Hasil Testing Model

Gambar 12 merupakan hasil dari pengetesan model. Dengan menggunakan fungsi `evaluate()` untuk mengevaluasi model, data yang digunakan disini adalah 20% dari total dataset yakni sebanyak 825 data. Dari hasil pengetesan didapat tingkat loss 159% dengan tingkat akurasi sebesar 64.48%.

*E. Import Model*

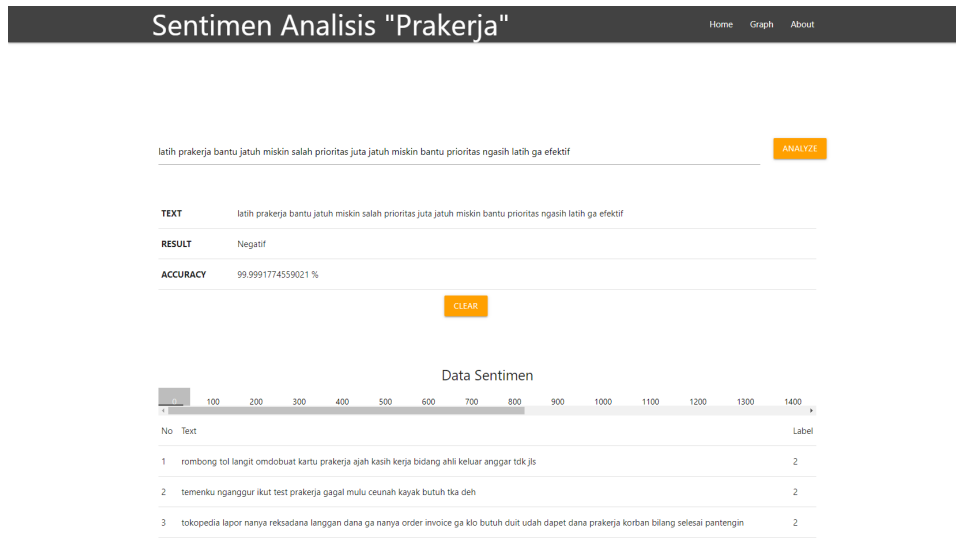
Setelah proses *training model* selesai langkah selanjutnya model perlu disimpan sebelum digunakan untuk proses deployment. Model sendiri merupakan sebuah file yang telah dilatih untuk mengenali tipe dari suatu pola data. Hal ini dilakukan agar Ketika sistem nantinya diberi kan data baru ia dapat mengenali 55 tipe dari pola tersebut dan membuat prediksi terhadap data tersebut. Selain untuk digunakan sebagai deployment model disimpan juga berguna agar pengguna ketika ingin menggunakan kembali model sehingga tidak perlu melakukan pelatihan ulang. Berikut model yang digunakan dalam penelitian ini :

- a) `model.h5` : file ini berisi arsitektur dari model, bobot nilai, informasi `compile()` dari model. Dalam penelitian model ini diperlukan sebelum di konversi ke `tensorflow.js`.
- b) `model.json` : hasil konversi dari `model.h5` yang berisi grafik aliran data dan bobot manifest.
- c) `group1-shard1of1.bin` : pasangan `model.json` hasil konversi yang berisi kumpulan bobot biner file.

Proses konversi tadi dilakukan menggunakan *library tensorflowjs*.



F. Pembahasan



Gambar 13: Halaman Home

Gambar 13 adalah halaman home, dihalaman ini terdapat *form input* untuk memasukan *tweet* yang ingin diklasifikasi. Ketika pengguna mensubmit dengan menekan *button analyze* sistem akan melakukan prediksi sentimen dengan menampilkan teks, hasil analisa, dan tingkat akurasi analisa.

G. Pengujian

Tabel III berisi data teks, label, dan hasil pengujian. Pengujian dilakukan dengan mengisikan teks satu persatu kedalam form input halaman home kemudian dari hasil Analisa sistem dibandingkan dengan label yang telah dideklarasikan diawal sehingga didapat hasil dibawah.

Dari hasil pengujian diketahui dari 10 data teks yang diuji ada 3 hasil prediksi yang salah. Sehingga dapat dihitung prediksi :

$$Akurasi = \frac{Jumlah\ prediksi\ benar}{Jumlah\ total\ presdiksi} \tag{1}$$

Persamaan (1) merupakan rumus menghitung akurasi jadi  $7/10 = 0.7$  adalah tingkat akurasi dari model klasifikasi.

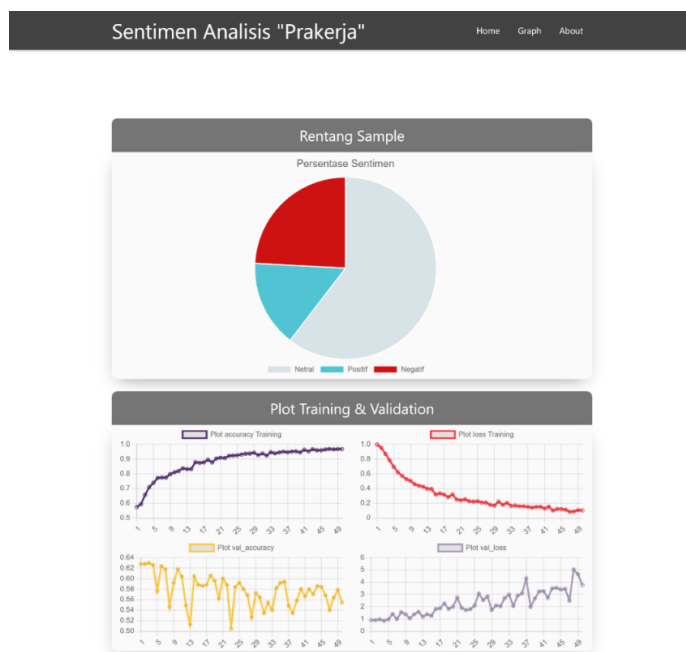
Tabel III: Caption

Teks	label	test
@sbsinews Banyak bantu banget emang dr pelatihan prakerja, terutama buat yg kena PHK kek tetangga ku dia ngambil kelas pembuatan masker di bukalahap trus dia realisasiin ilmunya buat cuan jd rame tuh pesenan maskernya bagus2 jg design warna nya	1	1
Pas kemarin ada RUU prakerja dengan tujuan memajukan negara protesMoyaikatanya menyengsarakan buruh. Sekarang, sosoan donasi ke negara lain. Bro liat bro utang indoneaia segitu. Gimana kalo kita patungan aja lunasin utang negara? Ga ngerti lagi aku tu hadoh.	2	2
Hallo Bencooleners! Bagi kalian yang telah lolos seleksi prakerja dan sedang mencari pelatihan yang dapat membantu mengembangkan kedai kopinya agar semakin laku, kelas "Strategi Pemasaran Usaha Kopi" dari Bencoolen Coffee sudah dibuka!	0	2
kebijakan yg diambil tdk terencana dan terukur. mudik dilarang,menghabiskan anggaran utk penyekatan,tempat wisata dibuka,bansos tdk tepat sasaran,prog prakerja tdk terarah,mau ekonomi brgerak tp THR buruh dan ASN dikurangi. pertumbuhan jalan dikit,pertumbuhan kendaraan naik trus.	2	2
Internet udah bisa di akses semua orang dan dimana aja, ada juga prakerja yang bisa bikin lu belajar dari dasar skill marketing atau hal" yang kalian minat Ilmu bisa bikin kalian kaya kalo kalian tau cara eksekusinya, pelajari dan peraktekan. Jangan cuma teori dan iming" mimpi -	1	0
Belajar daring tuh susah bgt. Apalagi aku belajar sesuatu yg baru yg blm pernah aku ketahui sebelumnya. Trs pengajarnya juga kurang kompeten aku rasa. Karena gak punya jadwal pembelajaran yg tetap . Mending" ikut prakerja deh, setidaknya ada jadwal tetapnya.	1	0
Kementerian Keuangan mencatat realisasi anggaran kartu prakerja mencapai Rp 9,85 triliun yang disalurkan kepada 2,77 juta peserta hingga 30 April 2021.	0	0
Berawal dari ia di PHK dari perusahaannya, Mas Bayu mengikuti program kartu prakerja. Beliau pun mengikuti pelatihan prakerja by Bencoolen Coffee. Dan akhirnya memutuskan untuk bermitra dengan Bencoolen Coffee.	0	0
Kartu korupsi bernama prakerja Tertawa berguling di lantaiTertawa berguling di lantaiJangan terkecoh sama tai meski disepuj emas Tertawa berguling di lantaiTertawa berguling di lantai	2	2
Program kartu prakerja ini memang bagus dan banyak manfaatnya ya, apalagi juga dapet insentif yang bisa digunakan untuk buka usaha habis ikut pelatihan di Bukalahap.	1	1

Gambar 14 merupakan visualisasi – visualisasi antara lain :

- a) Diagram Lingkaran : menunjuk kan rentang dari data crawling yang digunakan yaitu netral, positif, dan negatif.
- b) Diagram Garis : menunjuk kan pergerakan nilai akurasi dan loss baik itu untuk training dan testing selama proses training step berjalan.

Grafik disini dibuat dengan bantuan library visualisasi javascript Chart.js.



Gambar 14: Halaman Graf

V. SIMPULAN

Berdasarkan pembahasan yang sudah dipaparkan sebelumnya dapat disimpulkan bahwa data sampel yang digunakan termasuk dataset kurang seimbang dimana dari 4122 dataset, 2460 termasuk label netral, 689 termasuk label positif dan 973 termasuk

label negatif. Hasil proses training menghasilkan akurasi sebesar 0,9566 (95,66%) dengan tingkat loss 0,1999, hasil ini diperoleh dari 36 epoch. Hasil testing menunjukkan masih belum memuaskan karena akurasi masih diangka 0,6448 (64,48%) dengan loss sebesar 1,5922. Hasil training dan testing diketahui bahwa model termasuk dalam kategori overfitting.

#### PUSTAKA

- [1] CNN Indonesia, "Calon Peserta Membludak, Situs Kartu Prakerja Sulit Diakses", <https://www.cnnindonesia.com/ekonomi/20200918115006-92-548007/calon-peserta-membludak-situs-kartu-prakerja-sulit-diakses>
- [2] Kemp, Simon, "Digital 2020: Indonesia", <https://datareportal.com/reports/digital-2020-indonesia>, 9/21/2020 11:20:41 AM, 2020.
- [3] Harnantyo, Oni, "Analisis Sentimen Tempat Wisata Di Yogyakarta Menggunakan Metode Recurrent Neural Network Dengan Long Short Term Memory", 2019.
- [4] Aryadani, Dede, "Analisis Sentimen Konten Sosial Media Instagram STMIK Akakom Yogyakarta", 2020.
- [5] Liu, Bing, "Sentiment Analysis and Opinion Mining", Morgan & Claypool Publishers, 2012.
- [6] Olah, Cristhoper, "Understanding LSTM Networks" , <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>, 9/30/2020 12:43:53 PM, 2015.