

Besemah Language Translation Machine Model Based on Machine Learning with Recurrent Neural Network (RNN) Model Algorithm

Muhamad Andika^{1*)}, Yesi Novaria Kunang²⁾, Ilman Zuhri Yadi³⁾, Susan Dian Purnamasari⁴⁾

¹⁾²⁾³⁾⁴⁾Sistem Informasi, Sains Teknologi, Universitas Bina Darma

^{*)}Correspondence author: 191410165@student.binadarma.ac.id, Palembang, Indonesia

DOI: <https://doi.org/10.37012/jtik.v11i1.2614>

Abstract

Indonesia consists of various tribes with their respective regional languages, one of which is the Besemah tribe in South Sumatra Province with its language culture, namely the Besemah Language. Until now, the Besemah Language is still used by the Besemah tribe, but over time the number of Besemah Language speakers has decreased, plus most of the wider community does not know what the Besemah Language is. Machine Learning is a part of artificial intelligence that is often used to solve various problems. Machine Learning involves the use of computers and mathematical algorithms that use data to make predictions in the future. Machine translation is a tool that can convert one language to another. This study aims to collect datasets in the form of sentences and words from the Besemah Language, then create a Besemah Language translation machine to Indonesian and vice versa. In the research conducted, the approach used is Experimental Research in Machine Learning. Experimental research in machine learning for language translation is a research approach that involves designing and implementing a series of experiments to test and validate the performance of the language translation model. In this study, Neural Machine Translation (NMT) technology was applied with the Recurrent Neural Network (RNN) approach. The results of the study showed that the val_accuracy value for the Besemah-Indonesian translation was 0.8469 and for Indonesia-Besemah was 0.8492, in the translation trial conducted using the RNN model, 100 epochs, batch size 64, and validation split of 0.2.

Keywords: Neural Machine Translation, Besemah Language, Recurrent Neural Network

Abstrak

Indonesia terdiri dari berbagai suku dengan bahasa daerah masing-masing, salah satunya adalah suku Besemah yang berada di Provinsi Sumatera Selatan dengan budaya bahasanya, yaitu Bahasa Besemah. Hingga saat ini, Bahasa Besemah masih digunakan oleh suku Besemah, namun seiring berjalannya waktu jumlah penutur Bahasa Besemah semakin berkurang, ditambah lagi sebagian besar masyarakat luas tidak mengetahui apa itu Bahasa Besemah. Machine Learning adalah bagian dari kecerdasan buatan yang sering digunakan untuk menyelesaikan berbagai masalah. Machine Learning melibatkan penggunaan komputer dan algoritma matematika yang menggunakan data untuk membuat prediksi di masa depan. Penerjemahan mesin (machine translation) merupakan alat yang dapat mengalihkan satu bahasa ke bahasa lainnya. Penelitian ini bertujuan untuk mengumpulkan dataset berupa kalimat dan kata dari Bahasa Besemah, kemudian membuat mesin penerjemah Bahasa Besemah ke Bahasa Indonesia dan sebaliknya. Dalam penelitian yang dilakukan, pendekatan yang digunakan adalah Penelitian Eksperimen dalam Machine Learning. Penelitian eksperimen dalam machine learning untuk penerjemahan bahasa merupakan pendekatan penelitian yang melibatkan perancangan dan pelaksanaan serangkaian eksperimen untuk menguji dan memvalidasi kinerja model penerjemahan bahasa. Dalam penelitian ini diterapkan teknologi Neural Machine Translation (NMT) dengan pendekatan Recurrent Neural Network (RNN). Hasil penelitian menunjukkan bahwa nilai val_accuracy untuk penerjemahan Besemah-Indonesia adalah 0,8469 dan untuk Indonesia-Besemah adalah 0,8492, dalam uji coba

penerjemahan yang dilakukan menggunakan model RNN, 100 epoch, ukuran batch 64, dan validation split sebesar 0,2.

Kata Kunci: Neural Machine Translation, Bahasa Besemah, Recurrent Neural Network

PENDAHULUAN

Machine Learning adalah bagian dari kecerdasan buatan yang sering digunakan untuk menyelesaikan berbagai masalah. Machine Learning melibatkan penggunaan komputer dan algoritma matematika yang menggunakan data untuk membuat prediksi di masa depan. Proses ini melibatkan tahap pelatihan dan pengujian, serta terus berkembang melalui penelitian, termasuk di bidang penerjemahan bahasa (Roihan, Sunarya, & Rafika, 2020).

Machine translation merupakan bidang penelitian yang aktif, meskipun penelitian khusus mengenai penerjemahan dari bahasa Indonesia ke bahasa Besemah dengan menggunakan metode berbasis jaringan saraf tiruan masih jarang dilakukan (Setiawan & Sujaini, 2017). Neural Machine Translation (NMT) adalah istilah yang merujuk pada metode penerjemahan menggunakan jaringan saraf tiruan. Mesin Penerjemah bertugas untuk secara otomatis mengubah teks dari satu bahasa ke bahasa lain. Metode yang umum digunakan meliputi RNN (Recurrent Neural Network), CNN (Convolutional Neural Network) (Wismoyo & Kusumaningrum, 2018).

Artificial neural network yang dikenal sebagai Recurrent Neural Network (RNN) sangat cocok untuk mengenali pola dalam data yang disusun secara berurutan, misalnya dalam mengubah urutan kata dari bahasa Indonesia ke bahasa Besemah. Penelitian ini bertujuan untuk melestarikan bahasa daerah Besemah agar tetap hidup dan diteruskan dari generasi ke generasi. Pendekatan yang digunakan dalam penelitian ini adalah dengan memanfaatkan teknologi Neural Machine Translation (NMT). Pendekatan ini mengandalkan arsitektur Recurrent Neural Network (RNN) dalam proses penerjemahan (Aristyanto & Kurniawan, 2021).

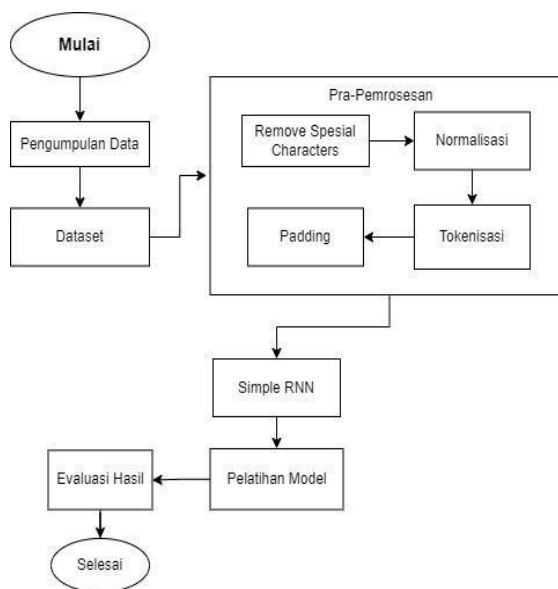
Besemah adalah nama sebuah suku yang telah lama berada di Indonesia. Suku Besemah memiliki bahasa daerah sendiri yang disebut **Bahasa Besemah**, sebagaimana suku-suku lainnya yang juga memiliki bahasa masing-masing. Hingga saat ini, Bahasa

Besemah masih digunakan oleh penuturnya sebagai alat komunikasi dan interaksi antar anggota masyarakat (Saputra, 2018).

Penelitian ini bertujuan untuk merancang atau membangun mesin penerjemah yang dapat menjadi solusi dalam meningkatkan pengetahuan, penggunaan, dan pelestarian bahasa daerah Besemah. Oleh karena itu, penulis mengangkat penelitian dengan judul "**Mesin Penerjemah Bahasa Besemah Berbasis Machine Learning dengan Algoritma Model Recurrent Neural Network (RNN)**".

METODE

Dalam penelitian yang dilakukan, pendekatan yang digunakan adalah **Penelitian Eksperimen dalam Machine Learning**. Penelitian eksperimen dalam machine learning untuk penerjemahan bahasa merupakan pendekatan penelitian yang melibatkan perancangan dan pelaksanaan serangkaian eksperimen untuk menguji dan memvalidasi kinerja model penerjemahan bahasa. Dalam konteks ini, eksperimen dilakukan dengan menyesuaikan berbagai parameter dan variabel dalam model, seperti arsitektur jaringan saraf, jumlah layers (lapisan), fungsi aktivasi, dan lain sebagainya. Alur penelitian yang dilakukan dapat dilihat melalui flowchart berikut:



Gambar 1. Flowchart

A. Flowchart

Metodologi penelitian untuk pengembangan mesin penerjemah dari Bahasa Besemah ke Bahasa Indonesia ditampilkan pada Gambar 1.

B. Pengumpulan Dataset

Pada tahap ini, pengumpulan data bahasa Besemah yang akan diterjemahkan dilakukan dengan memindai data sumber dari kamus **BESEMAH-INDONESIA-ENGLISH** (Mahdi, 2014).

C. Pembuatan Dataset

Pada tahap ini, sebuah dataset dibentuk dari data yang telah dikumpulkan sebelumnya. Dataset ini kemudian akan dibagi menjadi 2 bagian, yaitu dataset bahasa Indonesia dan dataset bahasa Besemah. Kedua dataset tersebut akan dibuat dalam bentuk file dengan format txt.

D. Pra Pemrosesan

- **Remove Special Characters:** Sebuah langkah untuk menghapus atau mengganti karakter yang tidak diinginkan dalam sebuah string. Karakter khusus adalah karakter yang berbeda dari huruf, angka, atau spasi, seperti tanda baca, simbol, atau karakter non-ASCII.
- **Normalisasi:** Proses ini mengubah teks ke format yang distandarisasi dengan mengubah teks menjadi huruf kecil untuk memudahkan pemrosesan. Tujuannya adalah untuk membuat teks yang bervariasi menjadi seragam, sehingga memudahkan pemrosesan selanjutnya dan meningkatkan konsistensi data
- **Tokenisasi:** Langkah ini memecah teks menjadi kata-kata terpisah. Pemisahan ini dilakukan dengan mempertimbangkan spasi di antara kata-kata atau dengan menerapkan aturan tertentu, seperti pemisahan berdasarkan tanda baca atau karakter khusus.
- **Padding:** Pada tahap ini, elemen tambahan ditambahkan ke data input untuk memastikan ukuran data seragam atau memenuhi persyaratan tertentu. Hal ini dilakukan dengan tujuan agar semua sampel dalam dataset memiliki dimensi yang sama, sehingga mempermudah pemrosesan batch secara efisien

E. Model RNN

Pada tahap ini, modifikasi dilakukan pada struktur model **RNN** dengan lapisan-lapisan **Recurrent Neural Network** yang terhubung secara berurutan. Ini adalah langkah untuk mempersiapkan data yang akan digunakan dalam proses pelatihan model **RNN**.

Rumus-rumus dari algoritma **RNN** meliputi hal-hal berikut:

$$a^{<t>} = a_{aa} a^{<t-1>} + W_{ax} x^{<t>} + b_a \text{ dan } y^{<t>} = g_2(W_{ya} a^{<t>} + b_y)$$

Wax, **Waa**, **Wya**, **ba**, dan **by** adalah faktor-faktor yang disesuaikan seiring waktu, dan **g1** serta **g2** adalah fungsi aktivasi. Dalam pelatihan jaringan saraf tiruan sederhana (**RNN**), metode **backpropagation through time (BPTT)** digunakan. **BPTT** menghitung bagaimana gradien dari fungsi kerugian (**L**) terkait dengan parameter model, menggunakan aturan rantai. Gradien ini kemudian digunakan untuk memperbarui parameter model dengan menggunakan algoritma optimasi seperti **stochastic gradient descent (SGD)**.

F. Pelatihan Model

Setelah memperoleh dataset dan model terbentuk, langkah selanjutnya adalah pelatihan model. Model yang telah terbentuk sebelumnya akan diberikan dataset yang telah disiapkan untuk melatih model. Tujuannya adalah agar model dapat memahami hubungan antara kalimat dalam bahasa Besemah dan bahasa Indonesia.

G. Evaluasi Hasil

Pada tahap evaluasi ini, dilakukan pengujian terhadap model dan perbaikan untuk mencapai hasil terbaik. Pada tahap ini, model dievaluasi menggunakan dataset uji untuk mengukur kinerja dan kualitas prediksi.

HASIL DAN PEMBAHASAN

A. Pengumpulan Data

Data yang dikumpulkan berasal dari pemindaian kamus **Besemah-Indonesia-Inggris** oleh dr. sutyono mahdi D.rs., M.Hum. Data yang dipindai disimpan dalam file Excel

untuk pemrosesan selanjutnya. Dalam file Excel tersebut, terdapat 5104 data yang akan digunakan nanti. Dari 5104 data tersebut, 3375 data berupa kata-kata dan 1729 data berupa kalimat.

5083	Ndaq ngape die	Mau apa dia
5084	Diq keruan li ku	Tidak tahu saya
5085	isan dimane nining	Dari mana nenek
5086	Ndaq tuape die	Ingin apa dia
5087	Nganuka sape die	Mengganggu siapa dia
5088	Njemugh tuape kabah	Apa yang engkau jemur
5089	Aku dang kurang sihat	Saya sekarang kurang sehat
5090	Kalu aku lah ghadu	Bila saya sudah sembuh
5091	kabah ka kukanceghi	engkau akan saya temani
5092	Die nde pacaq gale nga jeme di situ	Dia yang kenal semua dengan orang di sana
5093	Mertuaku akan hajatan besok malam	Mertuaku akan hajatan besok malam
5094	Die ka bekiaji	Dia akan naik haji
5095	Mamaq mbuat ghumah di dusun kamu	Paman membuat rumah di desamu
5096	Ghumah itu kandi anaq nde empai tunaq	Rumah itu untuk anaknya yang baru menikah
5097	Banyaq pisang di bawah ghumah	Di bawa rumah banyak pisan
5098	Pisang itu empai nebang petang tadi	Pisang itu baru diambil sore tadi
5099	Jeme ghumah itu lah agung sandi baghi	Orang itu sudah kaya sejak lama
5100	Anye cakagh duit tu masih katah nemane	Tetapi mencari uang masih sangat giat
5101	Masuqa gale ikan ni ke kulkas	Masukan semua ikan ini ke dalam kulkas
5102	Kalu dide pagi busuq gale	Kalau tidak besok busuk semuanya
5103	Die tu lah tue samegi nga aku	Dia itu sudah tua sama dengan saya
5104	Anye kinaqane masih muda	Tetapi kelihatannya masih muda

Gambar 2. Data Kamus yang dipindai disimpan dalam berkas Excel

B. Dataset

Pada tahap ini, dataset dipisahkan dari file Excel sebelumnya. Pemisahan dataset ini dibagi menjadi dua, yaitu dataset bahasa Indonesia dan dataset bahasa Basemah. Masing-masing dataset tersebut kemudian akan disimpan kembali dalam bentuk file **txt** dan akan digunakan dalam pemrosesan data pada machine learning di tahap selanjutnya.

basemah File Edit View Aku dang kurang sihat,Kalu aku lah ghadu,Kabah ka kukanceghi Aku ngajung bogy ke kantor camat,Die nde pacaq gale nga jeme di situ Bogy pelisi Bogy Tentra Bapae riye Dewi penyanyi Sapi Binatang Mujair ikan Hape telepon Ibu Yesi guru kami Kakangku penjaga toko roti Ugha itu jeme Lematang Bapang rengis Dusune sudut Tebat Libagh Anaq kerencal Kuwini ini masam Wali tu majel Kebaue bunguq Kudug tu landap ige Budao-budao lah besaq Deghian itu dangkal Potro kabah ni bekedal Durung tu nicingkik Ayiq nendidin Ratenye njegil Ruslan ngadut di sawah Bao masih tiduq di ghuma Itiq itu betelue saje setiap akap Adingku baliq sandi pasar mbeli gulei kande mekan Kundangu ka beghusiq ke ghuna akap pagi Kucing ini mbatak ikan sandi ghuma jeme Uneaq nga bea dang ngatem di parak ghuma Ln1, Col 1 87862 characters 100% Windows (CRU) UTF-8	indonesia File Edit View Saya sekarang kurang sehat, Bila saya sudah sembuh,engkau akan saya temani Saya meminta Bogy ke kantor camat, dia yang kenal semua dengan orang disana Bogy polisi Bogy Tentra Bapaknya Ketua RT Dewi Penyanyi Sapi Binatang Mujair Ikan Hape Telepon Ibu yesi guru kami Kakak saya penjaga toko roti Orang itu orang Lematang Ayahnya galek Kemungnya terpendil Kolannya luas Anaknya sehat Buah ku ini asan Pisau itu tumpul kerasanya gembuk sehal Perang itu terlalu tajam Anak-anak sudah besar Durian itu dagingnya tipis Motormu ini belepotan Burung itu sangat tinggi Air mendidih Melanya membuka Ruslan merokok di sawah Ayah masih tidur dirumah Itik itu bertelur terus setiap pagi Adikku pulang dari pasar membeli lauk untuk nakan Kawanku akan berkunjung kerumah besok pagi Kucing ini membawa ikan dari rumah orang Ln1, Col1 101124 characters 100% Windows (CRU) UTF-8
--	--

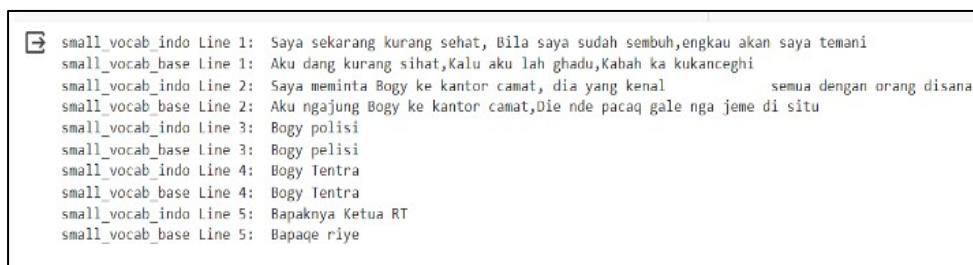
Gambar 3. Dataset Bahasa Basemah dan Dataset Bahasa Indonesia

C. Pra pemrosesan

Ada tahap ini, dataset yang telah dibuat akan menjalani proses **pre-processing** terlebih dahulu. Tahap pre-processing ini mencakup empat langkah, yaitu penghapusan karakter khusus, normalisasi, tokenisasi, dan padding.

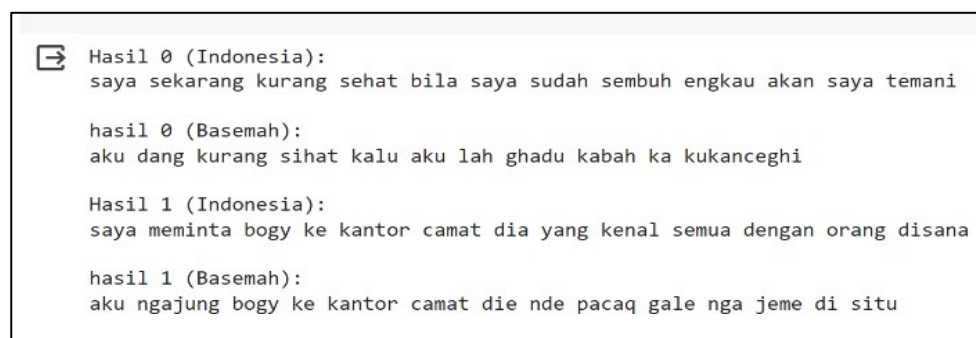
- Remove Special Characters

Pada tahap ini, teks akan dibersihkan dari karakter-karakter khusus. Karakter khusus ini meliputi tanda baca, simbol, atau karakter khusus lainnya yang kurang penting dan tidak diperlukan dalam pemrosesan data selanjutnya.



```
small_vocab_indo Line 1: Saya sekarang kurang sehat, Bila saya sudah sembuh,engkau akan saya temani
small_vocab_base Line 1: Aku dang kurang sihat,Kalu aku lah ghadu,Kabah ka kukanceghi
small_vocab_indo Line 2: Saya meminta Bogy ke kantor camat, dia yang kenal semua dengan orang disana
small_vocab_base Line 2: Aku ngajung Bogy ke kantor camat,Die nde pacaq gale nga jeme di situ
small_vocab_indo Line 3: Bogy polisi
small_vocab_base Line 3: Bogy pelisi
small_vocab_indo Line 4: Bogy Tentra
small_vocab_base Line 4: Bogy Tentra
small_vocab_indo Line 5: Bapaknya Ketua RT
small_vocab_base Line 5: Bapaqe riye
```

Gambar 4. Data Sebelum Menghapus Karakter Khusus



```
Hasil 0 (Indonesia):
saya sekarang kurang sehat bila saya sudah sembuh engkau akan saya temani

hasil 0 (Basemah):
aku dang kurang sihat kalu aku lah ghadu kabah ka kukanceghi

Hasil 1 (Indonesia):
saya meminta bogy ke kantor camat dia yang kenal semua dengan orang disana

hasil 1 (Basemah):
aku ngajung bogy ke kantor camat die nde pacaq gale nga jeme di situ
```

Gambar 5. Data Setelah Proses Penghapusan Karakter Khusus

Berdasarkan Gambar 4 dan Gambar 5, dapat dilihat bahwa terdapat perubahan pada teks setelah melewati proses **remove special characters**. Pada teks awal, masih terdapat tanda baca seperti "," yang terletak setelah kata-kata tertentu. Setelah melewati proses ini, tanda "," tersebut hilang atau dihapus.

- Normalisasi

Tahap ini berupa penyederhanaan teks dengan mengubah semua huruf menjadi huruf kecil. Bentuk normalisasi ini dilakukan agar huruf-huruf dalam teks menjadi seragam dan lebih sederhana, sehingga dapat mempermudah pemrosesan selanjutnya.

```
[Contoh kalimat bersih dari dataset Indo]:
Saya sekarang kurang sehat  Bila saya sudah sembuh engkau akan saya temani
Saya meminta Bogy ke kantor camat  dia yang kenal                semua dengan orang disana

[Contoh kalimat bersih dari dataset basem]:
Aku dang kurang sihat Kalu aku lah ghadu Kabah ka kukanceghi
Aku ngajung Bogy ke kantor camat Die nde pacaq gale nga jeme di situ
```

Gambar 6. Data Sebelum Proses Normalisasi

```
⇒ Hasil 0 (Indonesia):
saya sekarang kurang sehat bila saya sudah sembuh engkau akan saya temani

hasil 0 (Basemah):
aku dang kurang sihat kalu aku lah ghadu kabah ka kukanceghi

Hasil 1 (Indonesia):
saya meminta bogy ke kantor camat dia yang kenal semua dengan orang disana

hasil 1 (Basemah):
aku ngajung bogy ke kantor camat die nde pacaq gale nga jeme di situ
```

Gambar 7. Data Setelah Proses Normalisasi

Dari Gambar 6 dan Gambar 7, terlihat adanya perubahan yang terjadi pada teks bahasa Indonesia dan teks bahasa Besemah setelah melewati proses normalisasi, yaitu perubahan semua huruf menjadi huruf kecil dan penghapusan spasi berlebih.

- Tokenisasi

Proses ini dilakukan dengan mengubah data dari bentuk teks menjadi bentuk numerik dengan memberikan nilai token pada setiap kata dalam kalimat. Pemberian nilai token ini bertujuan untuk memecah teks menjadi beberapa unit kecil sehingga pemrosesan data akan lebih mudah dilakukan nantinya.


```
[x] Dataset 1 word index: {'tidak': 1, 'saya': 2, 'dia': 3, 'di': 4, 'itu': 5, 'yang': 6, 'ini': 7, 'sudah': 8, 'orang': 9, 'engkau': 10, 'kau': 11, 'kami': 12, 'aku': 13}
Dataset 2 word index: {'dia': 1, 'aku': 2, 'di': 3, 'itu': 4, 'kabar': 5, 'lah': 6, 'jane': 7, 'diga': 8, 'ni': 9, 'nga': 10, 'ka': 11, 'ndaq': 12, 'kari': 13, 'paq': 14}
Samples from Dataset 1:
Sequence 1 in Dataset 1
Input: saya sekarang kurang sehat bila saya sudah sembuh engkau akan saya tenani
Output: [2, 40, 110, 173, 35, 2, 0, 395, 10, 13, 2, 800]
Sequence 2 in Dataset 1
Input: saya meminta bogy ke kantor carat dia yang kenal semua dengan orang disana
Output: [2, 801, 802, 15, 250, 088, 3, 6, 344, 45, 17, 9, 1131]
Samples from Dataset 2:
Sequence 1 in Dataset 2
Input: aku dang kurang sihat kalau aku lah ghadu kabah ka kukancegihl
Output: [2, 24, 223, 338, 26, 2, 6, 330, 5, 11, 614]
Sequence 2 in Dataset 2
Input: aku ngajang bogy ke kantor carat dia nde paq gale nga jane di situ
```

Gambar 8. Hasil Tokenisasi

- Padding

Pada tahap ini, nilai token yang telah ditentukan pada proses tokenisasi sebelumnya akan disamakan melalui proses padding. Tujuan dari teknik padding adalah untuk menyesuaikan panjang data sehingga dapat dimasukkan ke dalam algoritma atau model dengan ukuran input yang tetap. Padding biasanya diterapkan pada data yang memiliki panjang bervariasi agar setiap contoh atau sampel memiliki ukuran yang seragam.

```

Sequence 1 in Dataset 1
Input: [ 2 40 110 173 35 2 8 395 10 13 2 800]
Output: [ 2 40 110 173 35 2 8 395 10 13 2 800 0 0 0 0 0 0]
Sequence 2 in Dataset 1
Input: [ 2 801 802 15 250 608 3 6 344 45 17 9 1131]
Output: [ 2 801 802 15 250 608 3 6 344 45 17 9 1131 0]
Sequence 1 in Dataset 2
Input: [ 2 24 223 338 26 2 6 339 5 11 614]
Output: [ 2 24 223 338 26 2 6 339 5 11 614 0 0 0 0 0 0]
Sequence 2 in Dataset 2
Input: [ 2 956 615 19 1858 431 1 18 58 35 10 7 3 96]
Output: [ 2 956 615 19 1858 431 1 18 58 35 10 7 3 96]

```

Gambar 9. Hasil Padding

Tujuan dari proses ini adalah untuk menyiapkan data dalam format yang sesuai agar dapat dimasukkan ke dalam model machine learning. Dengan menyamakan panjang setiap kalimat, model dapat memproses data dengan lebih efisien dan seragam. Proses padding ini diperlukan karena model machine learning membutuhkan input

dengan panjang yang konsisten agar dapat menjalankan proses pembelajaran dengan baik.

D. RNN Model

Tahap ini berupa pembangunan model **RNN** untuk percobaan penerjemahan bahasa Besemah. **Recurrent Neural Network (RNN)** adalah jenis arsitektur jaringan saraf yang dirancang khusus untuk menangani data berurutan, seperti teks atau deret waktu. Dalam konteks mesin penerjemahan bahasa, model RNN digunakan untuk memahami dan menerjemahkan urutan kata atau frasa dari satu bahasa ke bahasa lain. Berikut adalah kode yang digunakan untuk membuat model RNN.

```
def simple_rnn_model1(input_shape, output_sequence_length, indonesia_vocab_size, basemah_vocab_size):  
    learning_rate = 1e-3  
    input_seq = Input(input_shape[1:])  
    rnn = GRU(64, return_sequences = True)(input_seq)  
    logits = TimeDistributed(Dense(basemah_vocab_size))(rnn)  
    model = Model(input_seq, Activation('softmax')(logits))  
    model.compile(loss = sparse_categorical_crossentropy,  
                  optimizer = Adam(learning_rate),  
                  metrics = ['accuracy'])  
  
    return model  
    tests.test_simple_model(simple_rnn_model)  
    print(tmp_x.shape)  
  
tmp_x = pad(preproc_indonesia_sentences, max_basemah_sequence_length)  
tmp_x = tmp_x.reshape((-1, preproc_basemah_sentences.shape[-2], 1))  
  
# Train the neural network  
simple_rnn_model1 = simple_rnn_model1(  
    tmp_x.shape,  
    max_basemah_sequence_length,  
    indonesia_vocab_size,  
    basemah_vocab_size  
)  
print(simple_rnn_model1.summary())
```

Gambar 10. Kode Pembuatan Model RNN Indonesia – Besemah

```
def simple_rnn_model1b(input_shape, output_sequence_length, basemah_vocab_size, indonesia_vocab_size):
    learning_rate = 1e-3
    input_seq = Input(input_shape[1:])
    rnn = GRU(64, return_sequences = True)(input_seq)
    logits = TimeDistributed(Dense(indonesia_vocab_size))(rnn)
    model = Model(input_seq, Activation('softmax')(logits))
    model.compile(loss = sparse_categorical_crossentropy,
                  optimizer = Adam(learning_rate),
                  metrics = ['accuracy'])

    #print(model.summary())
    return model
    print(tmp_x2.shape)

# tests.test_simple_model(simple_rnn_model)
tmp_x2 = pad(preproc_basemah_sentences, max_indonesia_sequence_length)
tmp_x2 = tmp_x2.reshape((preproc_indonesia_sentences.shape[-2], -1, 1))

# Train the neural network
simple_rnn_model1b = simple_rnn_model1b(
    tmp_x2.shape,
    max_indonesia_sequence_length,
    basemah_vocab_size,
    indonesia_vocab_size
)
print(simple_rnn_model1b.summary())
```

Gambar 11. Kode Pembuatan Model RNN Indonesia – Besemah

Kode ini mengimplementasikan model Recurrent Neural Network (RNN) menggunakan lapisan Gated Recurrent Unit (GRU) untuk melakukan penerjemahan bahasa. Model ini menerima sekumpulan kalimat dalam bahasa Indonesia sebagai input dan bertujuan menghasilkan terjemahan dalam bahasa Besemah. Model terdiri dari lapisan GRU dengan 64 unit, diikuti oleh lapisan Time Distributed dan Dense yang bertugas menghasilkan output untuk setiap langkah waktu. Fungsi aktivasi softmax diterapkan untuk mengubah nilai output menjadi distribusi probabilitas. Model dilatih menggunakan fungsi kerugian sparse categorical crossentropy dan dioptimalkan dengan metode Adam pada dataset pelatihan. Selama pelatihan, model dijalankan selama 100 epoch dengan ukuran batch 64, dan validasi dilakukan pada 20% data. Setelah pelatihan, model diuji dengan memprediksi output untuk satu instance input dan hasilnya diubah menjadi teks menggunakan fungsi logits to text.

E. Pelatihan Model

Setelah model RNN dibuat, pada proses ini pelatihan model akan dilakukan agar model yang telah dibuat sebelumnya dapat menerjemahkan bahasa Besemah. Pelatihan model ini dilakukan dengan 100 epoch dan menggunakan ukuran batch 64.

```
Epoch 91/100
64/64 [=====] - 1s 12ms/step - loss: 0.9389 - accuracy: 0.8472 - val_loss: nan - val_accuracy: 0.9463
Epoch 92/100
64/64 [=====] - 1s 12ms/step - loss: 0.9369 - accuracy: 0.8472 - val_loss: nan - val_accuracy: 0.9463
Epoch 93/100
64/64 [=====] - 1s 12ms/step - loss: 0.9345 - accuracy: 0.8477 - val_loss: nan - val_accuracy: 0.9464
Epoch 94/100
64/64 [=====] - 1s 12ms/step - loss: 0.9323 - accuracy: 0.8475 - val_loss: nan - val_accuracy: 0.9465
Epoch 95/100
64/64 [=====] - 1s 13ms/step - loss: 0.9306 - accuracy: 0.8477 - val_loss: nan - val_accuracy: 0.9462
Epoch 96/100
64/64 [=====] - 1s 12ms/step - loss: 0.9284 - accuracy: 0.8478 - val_loss: nan - val_accuracy: 0.9465
Epoch 97/100
64/64 [=====] - 1s 12ms/step - loss: 0.9260 - accuracy: 0.8479 - val_loss: nan - val_accuracy: 0.9464
Epoch 98/100
64/64 [=====] - 1s 12ms/step - loss: 0.9244 - accuracy: 0.8478 - val_loss: nan - val_accuracy: 0.9463
Epoch 99/100
64/64 [=====] - 1s 12ms/step - loss: 0.9221 - accuracy: 0.8480 - val_loss: nan - val_accuracy: 0.9463
Epoch 100/100
64/64 [=====] - 1s 14ms/step - loss: 0.9199 - accuracy: 0.8479 - val_loss: nan - val_accuracy: 0.9463
1/1 [=====] - 0s 486ms/step
aku dang sihat sihat dang aku lah <PAD> jeme ke aku <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD>

[17] print(logits_to_text(simple_rnn_model1.predict(tmp_x[:1])[0], basemah_tokenizer))

1/1 [=====] - 0s 28ms/step
aku dang sihat sihat dang aku lah <PAD> jeme ke aku <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD>
```

Gambar 12. Hasil Pelatihan Model RNN Indonesia – Basemah

```
Epoch 91/100
64/64 [=====] - 1s 12ms/step - loss: 0.9962 - accuracy: 0.8415 - val_loss: nan - val_accuracy: 0.9281
Epoch 92/100
64/64 [=====] - 1s 12ms/step - loss: 0.9934 - accuracy: 0.8415 - val_loss: nan - val_accuracy: 0.9281
Epoch 93/100
64/64 [=====] - 1s 12ms/step - loss: 0.9912 - accuracy: 0.8416 - val_loss: nan - val_accuracy: 0.9282
Epoch 94/100
64/64 [=====] - 1s 12ms/step - loss: 0.9897 - accuracy: 0.8414 - val_loss: nan - val_accuracy: 0.9282
Epoch 95/100
64/64 [=====] - 1s 11ms/step - loss: 0.9870 - accuracy: 0.8418 - val_loss: nan - val_accuracy: 0.9282
Epoch 96/100
64/64 [=====] - 1s 13ms/step - loss: 0.9856 - accuracy: 0.8417 - val_loss: nan - val_accuracy: 0.9282
Epoch 97/100
64/64 [=====] - 1s 13ms/step - loss: 0.9826 - accuracy: 0.8420 - val_loss: nan - val_accuracy: 0.9282
Epoch 98/100
64/64 [=====] - 1s 13ms/step - loss: 0.9809 - accuracy: 0.8418 - val_loss: nan - val_accuracy: 0.9282
Epoch 99/100
64/64 [=====] - 1s 14ms/step - loss: 0.9784 - accuracy: 0.8421 - val_loss: nan - val_accuracy: 0.9282
Epoch 100/100
64/64 [=====] - 1s 11ms/step - loss: 0.9772 - accuracy: 0.8419 - val_loss: nan - val_accuracy: 0.9282
1/1 [=====] - 0s 295ms/step
saya tidak saya saya makan saya sudah <PAD> itu tidak <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD>

[21] print(logits_to_text(simple_rnn_model1b.predict(tmp_x2[:1])[0], indonesia_tokenizer))
# print(tmp_x2[:1])

1/1 [=====] - 0s 17ms/step
saya tidak saya saya makan saya sudah <PAD> itu tidak <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD>
```

Gambar 13. Hasil Pelatihan Model RNN Indonesia – Basemah

Dari hasil output yang diberikan, terlihat bahwa nilai **val_accuracy** yang tercatat selama pelatihan model **Besemah-Indonesia** adalah sekitar **0,8469** pada data pelatihan. Sedangkan akurasi model **Indonesia-Besemah** adalah **0,8492**.

KESIMPULAN DAN REKOMENDASI

Uji coba penerjemahan antara bahasa Besemah dan bahasa Indonesia menggunakan model **Recurrent Neural Network (RNN)** yang dikembangkan menghasilkan pencapaian yang cukup signifikan. Dalam serangkaian eksperimen dengan 100 epoch, ukuran batch 64, dan validation split 0,2, model berhasil mencapai tingkat **val_accuracy** sebesar **0,8469** untuk Besemah-Indonesia dan **0,8492** untuk Indonesia- Besemah. Meskipun demikian, hasil prediksi menunjukkan beberapa keterbatasan, terutama dalam mereplikasi kalimat acuan yang sesuai. Penggunaan token "<PAD>" yang berlebihan mengindikasikan kesulitan model dalam memahami struktur bahasa Besemah. Oleh karena itu, perlu dilakukan **peninjauan mendalam terhadap konfigurasi model, ukuran dataset, dan kompleksitas struktur bahasa** untuk meningkatkan kinerja dan akurasi model penerjemah yang telah dikembangkan.

REFERENSI

- Aristyanto, MY, & Kurniawan, R (2021). Pengembangan metode neural machine translation berdasarkan hyperparameter neural network. *Seminar Nasional Official ...*, prosiding.stis.ac.id, <https://prosiding.stis.ac.id/index.php/semnasoffstat/article/view/789>
- Canfora, C, & Ottmann, A (2020). Risks in neural machine translation. *Translation Spaces*, jbe-platform.com, <https://doi.org/10.1075/ts.00021.can>
- Dabre, R, Chu, C, & Kunchukuttan, A (2020). A survey of multilingual neural machine translation. *ACM Computing Surveys (CSUR)*, dl.acm.org, <https://doi.org/10.1145/3406095>
- Klein, G, Hernandez, F, Nguyen, V, & ... (2020). The OpenNMT neural machine translation toolkit: 2020 edition. ... *for Machine Translation ...*, aclanthology.org, <https://aclanthology.org/2020.amta-research.9.pdf>
- Koehn, P (2020). *Neural machine translation.*, Cambridge University Press

- Klimova, B, Pikhart, M, Benites, AD, Lehr, C, & ... (2023). Neural machine translation in foreign language teaching and learning: a systematic review. *Education and ...*, Springer, <https://doi.org/10.1007/s10639-022-11194-2>
- Mahdi, S (2014). *Kamus Bahasa Besemah-Indonesia-Inggris.*, Unpad Press
- Nature (2024). Scaling neural machine translation to 200 languages. *nature.com*, <https://www.nature.com/articles/s41586-024-07335-x>
- Roihan, A, Sunarya, PA, & Rafika, AS (2020). *Pemanfaatan Machine Learning dalam Berbagai Bidang: Review paper. IJCIT (Indonesian Journal on Computer and Information Technology)*, 5 (1), 75
- Saputra, H (2018). Upaya Pemertahanan Bahasa Daerah Besemah Sebagai Bagian Pelestarian Kearifan Lokal. *MEDAN MAKNA: Jurnal Ilmu Kebahasaan dan ...*, core.ac.uk, <https://core.ac.uk/download/pdf/288194986.pdf>
- Setiawan, A, Sujaini, H, & Negara, ABP (2017). Implementasi Optical Character Recognition (OCR) pada Mesin Penerjemah Bahasa Indonesia ke Bahasa Inggris. *JUSTIN (Jurnal Sistem dan Teknologi Informasi ...*
- Son, J, & Kim, B (2023). Translation performance from the user's perspective of large language models and neural machine translation systems. *Information*, mdpi.com, <https://www.mdpi.com/2078-2489/14/10/574>
- Stahlberg, F (2020). Neural machine translation: A review. *Journal of Artificial Intelligence Research*, jair.org, <https://www.jair.org/index.php/jair/article/view/12007>
- Wismoyo, PA, & Kusumaningrum, R (2018). *Mesin Penerjemah Bahasa Inggris-Indonesia Berbasis Jaringan Saraf Tiruan Dengan Mekanisme Attention Menggunakan*, Universitas Diponegoro
- Zhang, J, & Zong, C (2020). Neural machine translation: Challenges, progress and future. *Science China Technological Sciences*, Springer, <https://doi.org/10.1007/s11431-020-1632-x>