

Perbandingan Penggunaan Metode Part-of-speech pada Bahasa Indonesia Menggunakan Metode N-Gram, TnT, dan Classifier Based

Krisna Fernando¹⁾, Yulius Denny Prabowo²⁾

Informatika, Fakultas Industri Kreatif, Institut Teknologi dan Bisnis Kalbis
Jalan Pulomas Selatan Kav. 22, Jakarta 13210

¹⁾ Email: krisnandofernando@gmail.com

²⁾ Email: yulius.prabowo@kalbis.ac.id

Abstract: *The lack of a corpus in Indonesian makes text processing in Indonesian have little interest in processing the text, therefore the aim of doing POS-Tagging is to do word labeling that takes into account the essence of the sentence based on the words contained in the sentence into a per-word tagset, the tagging process uses the N-gram, TnT, and Classifier Based methods. The three methods are used because the three methods have similar work methods and work by looking at the traces of learning and working on the previous method. The method that produces the optimal tagger of the three methods is the Classifier Based method because this method is a combination of the three methods, so that the tagger results issued are more optimal than the other two methods.*

Keywords: *Part-Of-Speech Tagging, N-Gram Tagger, TnT Tagger, Classifier-based Tagger.*

Abstrak: *Kurangnya korpus dalam bahasa Indonesia membuat pengolahan teks pada bahasa Indonesia memiliki peminat yang sedikit untuk dilakukan pengolahan terhadap teks, oleh hal itu tujuan dari melakukan POS-Tagging adalah melakukan pelabelan kata yang memperhitungkan inti dari kalimat berdasarkan kata yang terkandung dalam kalimat menjadi tagset per-kata, proses melakukan tagging menggunakan metode N-gram, TnT, dan Classifier Based. Ketiga metode tersebut digunakan dikarenakan ketiga metode memiliki cara pengerjaan yang mirip dan bekerja dengan cara melihat jejak pembelajaran dan pengerjaan metode sebelumnya. Metode yang menghasilkan tagger optimal dari ketiga metode tersebut adalah metode Classifier Based dikarenakan metode ini merupakan gabungan dari ketiga metode tersebut, sehingga hasil tagger yang dikeluarkan lebih optimal dibandingkan dengan dua metode lainnya.*

Kata kunci: *Part-Of-Speech Tagging, N-Gram Tagger, TnT Tagger, Classifier-based Tagger.*

I. PENDAHULUAN

Perkembangan pada pengolahan bahasa alami di Indonesia membuat korpus bahasa Indonesia menjadi sumber yang penting sebagai data untuk melakukan pengolahan dan data penelitian, korpus sendiri merupakan kumpulan dokumen teks yang dijadikan sebagai dataset standar bagi para peneliti pengolahan bahasa alami tersebut[1].

Bahasa memang berfungsi sebagai alat refleksi dan ekspresi terhadap nilai budaya masyarakat, melalui bahasa pula manusia memperbarui proses berpikir dan secara simultan teknologi berperan besar memperbaiki kualitas dan kuantitas pengembangan bahasa[1].

Untuk kasus di Indonesia sumber daya data penelitian berbahasa Indonesia masih terbatas, hal ini diutarakan oleh Ketua Jurusan Teknik Informatika

Fakultas Teknologi Industri Universitas Trisakti Anung B. Ariwibowo[1]. Akan tetapi literature penulisan dapat ditemukan dalam beberapa platform daring seperti media sosial ataupun portal berita.

Dengan banyaknya pengguna *internet* di Indonesia yang renta usia penggunaannya mulai dari usia 16 tahun hingga usia 64 tahun dengan masing-masing jenis perangkat seperti *mobile phone* sebesar 96%, *smartphone* sebesar 94%, *non-smartphone mobile phone* sebesar 21%, laptop dan komputer sebesar 66%, tablet sebesar 23%, konsol gim sebesar 16%, dan *virtual reality device* sebesar 5,1%[2]. Akan tetapi data menunjukkan bahwa masyarakat Indonesia memiliki presentase kepemilikan ponsel sebesar 125% lebih besar atau sebanyak 338.2 juta koneksi *internet* melalui *mobile phone* dibandingkan dengan jumlah penduduknya[2].

Untuk literature review pada jurnal yang berjudul “*Investigating Bi-LSTM and CRF with POS Tag Embedding for Indonesian Named Entity Tagger*” tahun 2018 oleh Devin Hoesen dan Ayu Purwarianti, metode yang digunakan merupakan Bi-directional long short term memory dan Conditional Random Field dengan menggunakan data teks sebanyak 8400 kalimat sebagai arsitekturnya untuk membangun named entity. Pada jurnal yang berjudul “On Part of Speech Tagger for Indonesian Language” oleh R.Sandra Yuwana, Asri R.Yuliani dan Hilman F.Pardede menggunakan metode unigram, hidden markov model, TnT, Brills. Naïve Bayes dan Maximum Entropy.

Dari banyaknya data diatas banyaknya kalimat komentar yang terdapat pada media sosial ataupun tulisan informasi berita yang terdapat pada portal berita daring dapat dijadikan media pembelajaran untuk machine

learning pada pengolahan teks yaitu Natural Language Processing dengan menjadikan kalimat-kalimat tersebut sebagai data teks yang akan dijadikan corpus bahasa Indonesia dan dapat digunakan untuk pengolahan teks pada machine learning.

II. METODE PENELITIAN

A. Part-of-Speech Tag

POS-Tagging merupakan suatu cara pengkategorian kelas kata, seperti kata benda, kata kerja, kata sifat, dan lain- lain[6]. *POS Tagger* merupakan sebuah aplikasi yang mampu melakukan proses anotasi *part-of-speech* tag untuk setiap kata didalam dokumen secara otomatis[3].

Dikarenakan tag secara umum juga dapat diaplikasikan pada tanda baca, maka dalam tagging, tanda baca seperti tanda titik, tanda koma, dan lain-lain perlu dipisahkan dari kata-kata[4]. oleh sebab itu proses tokenisasi biasanyadilakukan sebelum POS-Tagging dan beberapa preprocessing dilakukan untuk pemisahan koma, tanda petik, dan lain-lain dari kata serta dilakukan disambiguitas pada tanda baca penanda akhir seperti tanda titik dan tanda tanya agar dapat dibedakan dari tanda yang digunakan untuk singkatan[4].

B. Natural Language Processing

NLP (Natural Language Processing) adalah bidang ilmu komputer dan teknik yang telah dikembangkan dari studi bahasa dan linguistik komputasi dalam bidang Artificial Intelligence[5]. Tujuan dari NLP adalah untuk merancang dan membangun aplikasi yang memfasilitasi interaksi manusia dengan mesin dan perangkat lain melalui penggunaan

bahasa alami, beberapa bidang utama NLP yaitu[5]:

1. *QAS(Question Answering System)*

Membuat komputer/gadget dapat menjawab pertanyaan pengguna seputar pertanyaan sehari-hari seperti pertanyaan mengenai tempat makan yang direkomendasikan oleh komputer berdasarkan jam tutup tempat makan tersebut dengan cara berbicara dengan komputer/gadget pengguna dan disesuaikan dengan bahasa alami pengguna seperti bahasa Inggris, bahasa Cina, ataupun bahasa Spanyol[5].

2. *Summarization*

Mencakup aplikasi yang dapat mengambil koleksi dokumen atau email dan menghasilkan ringkasan yang koheren mengenai konten tersebut. Program tersebut juga bertujuan untuk menyediakan snap “elevator summaries” dari dokumen dan mungkin bahkan mengubahnya menjadi presentasi slide[6].

3. *Machine Translation*

Merupakan bagian penting dalam NLP, yang bertujuan untuk menerjemahkan bahasa alami dari Negara yang berbeda seperti penerjemahan bahasa Inggris ke bahasa Indonesia dan bahkan terdapat aplikasi translasi yang dapat melakukan translasi secara real time[6].

4. *Speech Recognition*

Speech Recognition adalah salah satu masalah yang paling sulit dalam NLP, telah ada kemajuan besar dalam model bangunan yang dapat digunakan pada ponsel atau komputer untuk mengali pengucapan bahasa alami menjadi pertanyaan dan perintah[6].

5. *Document Classification*

Document Classification merupakan salah satu daerah pengetahuan yang

paling sukses dalam NLP, yang bertugas untuk mengidentifikasi kategori dokumen harus ditempatkan, seperti penyaringan spam, klasifikasi berita, dan review film, .

C. *Tokenisasi*

Tokenisasi merupakan proses pemisahan teks menjadi potongan-potongan yang disebut sebagai token untuk kemudian dianalisa[7]. Kata, angka, symbol, tanda baca, dan entitas penting lainnya dapat dianggap sebagai token.

Kata selanjutnya dapat disebut dengan kumpulan token. Token sendiri terdiri dari bermacam jenis, seperti berbentuk *words, lines, sentences, atau paragraph*[7].

D. *N-Gram*

N-Gram merupakan kumpulan kata dengan masing-masing memiliki n-kata, sebagai contoh ukuran 1 disebut dengan unigram, ukuran 2 disebut sebagai bigram, ukuran 3 disebut dengan trigram, dan seterusnya[8]. Penerapan n-gram dapat dilihat dengan contoh kalimat “pemilihan kepala daerah Jakarta tahun ini tidak begitu ramai dibandingkan tahun sebelumnya”[8].

Tabel 2.1 Penjelasan N-Gram

Unigram	<i>Pemilihan, kepala, daerah, Jakarta, tahun, ini, tidak, begitu, ramai, dibandingkan, dengan, tahun, sebelumnya</i> [8].
Bigram	<i>Pemilihan kepala, kepala daerah, daerah Jakarta, Jakarta tahun, tahun ini, ini tidak, tidak begitu, begitu ramai, ramai dibandingkan, dibandingkan dengan, dengan tahun, tahun sebelumnya</i> [8].
Trigram	<i>Pemilihan kepala daerah, kepala daerah Jakarta, daerah Jakarta tahun, Jakarta tahun ini, tahun ini tidak, ini tidak begitu, tidak begitu ramai, begitu ramai dibandingkan, ramai dibandingkan dengan, dibandingkan dengan tahun, dengan tahun sebelumnya</i> [8].

E. Classifier Based POS-Tagger

Classifier Based POS-Tagger menggunakan klasifikasi untuk melakukan part-of-speech[26]. Fitur diekstraksi dari kata-kata dan kemudian diteruskan ke klasifikasi internal[26]. Classifier Based POS-Tagger adalah subclass dari Classifier Based Tagger untuk mengimplementasikan fitur detektor yang menggabungkan banyak teknik dan penandaan sebelumnya menjadi satu set fitur tunggal[9].

Detektor fitur menemukan beberapa sufiks panjang, melakukan pencocokan regular ekspresi, dan melihat sejarah unigram, bigram, dan trigram untuk menghasilkan set fitur lengkap yang optimal untuk setiap kata, set fitur yang dihasilkan digunakan untuk melatih classifier internal, dan digunakan untuk mengklasifikasikan kata-kata menjadi part-of-speech tags[9].

F. Python

Python merupakan bahasa pemrograman yang pertama kali diciptakan oleh seseorang bernama Guido Van Rossum di Belanda pada tahun 1990[10]. Python merupakan bahasa pemrograman yang sangat baik digunakan untuk aplikasi big data, data mining, dan machine learning[11]. Python merupakan bahasa pemrograman tingkat tinggi karena kedekatan kodenya mirip dengan bahasa manusia dan dijalankan menggunakan sistem interpreted, hal tersebut disebabkan program python yang dijalankan baris per baris tanpa memerlukan compile[12]. Beberapa aplikasi yang dibuat dari bahasa pemrograman python adalah facebook, instagram, mesin pencarian google, spotify, dan lainnya[13].

G. Library Python

Untuk memudahkan objek penelitian dalam melakukan pengembangan aplikasi, maka peneliti menggunakan beberapa library python.

1. NLTK

Natural Language Toolkit atau disingkat NLTK, adalah library python untuk bekerja dengan pemodelan teks. NLTK menyediakan alat yang baik mempersiapkan teks sebelum digunakan pada machine learning atau algoritma deep learning[14].

2. Tkinter

Tkinter (Tk Interface) merupakan suatu pustaka GUI (Graphic User Interface) yang terdapat pada bahasa pemrograman python dan bekerja berdasarkan toolkit yang terdapat pada python itu sendiri[15]. Setiap GUI toolkit menyediakan widget berupa objek user interface seperti button, scrollbar, listbox, checkbutton, radiobutton, label text dan lainnya[15].

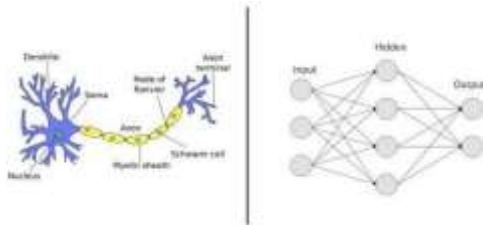
F. Neural Network

Neural network merupakan salah satu cabang ilmu dari ilmu soft computing[16]. Neural Network merupakan sebuah model yang terinspirasi dari cara kerja otak manusia dan dikenalkan pertama kali pada tahun 1943 oleh Warren McCulloch dan Walter Pitts[16]. Neural Network telah dikembangkan dan memiliki fungsi yang beragam[16], dan mempunyai beberapa fungsi antara lain[16]:

1. Pengklasifikasian pola.
2. Memetakan pola yang didapat dari input kedalam pola baru pada output.
3. Penyimpanan pola yang akan dipanggil kembali.
4. Memetakan pola-pola sejenis.
5. Optimasi permasalahan dan prediksi.

Neural Network merupakan model yang terinspirasi dari otak manusia, tiap neuron pada otak manusia saling

terhubung dan menyampaikan informasi[17].

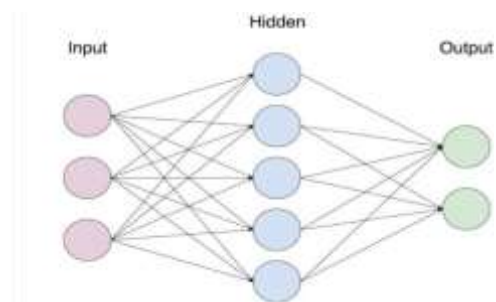


Gambar 1 Cara Kerja Neural Network

Seperti pada cara kerja otak manusia neural network bekerja mirip dengan kerja otak manusia yaitu impuls yang diterima oleh dendrite dialirkan ke inti sel untuk diproses, dendrite kemudian melakukan operasi dot dengan menggunakan sebuah weight, kemudian menjumlahkan setiap input yang ada dan menambahkan dengan bias[18]. Hasil dari operasi dimasukan kedalam fungsi aktifasi yang akhirnya menghasilkan output dari neuron tersebut[19].

Fungsi aktifasi pada neural network berfungsi untuk menentukan neuron tersebut aktif atau tidak berdasarkan hasil operasi dot yang dilakukan sebelumnya. Fungsi ini nantinya akan memproduksi output untuk diteruskan ke neuron lain. Secara umum fungsi aktifasi dibagi menjadi dua jenis yaitu fungsi linear dan fungsi non-linear. Fungsi linear dapat dikatakan sebagai fungsi standar dari sebuah neuron, jika fungsi ini digunakan pada sebuah neuron maka keluaran dari neuron tersebut merupakan weighted sum + bias[20].

Neural Network memiliki arsitektur. Pada arsitektur neural network terdapat 3 layer[20]. Layer tersebut adalah input layer, hidden layer, dan output layer[20]. Input layer merupakan layer pertama yang berfungsi sebagai input, hidden layer merupakan layer yang berada diantara input layer dan output layer. Output layer merupakan layer terakhir yang menghasilkan output dari model.



Gambar 2. 1Layer pada Neural Network

G. Machine Learning

Machine Learning merupakan sebuah metode yang merupakan cabang dari kecerdasan buatan yang membuat sebuah program dapat belajar secara mandiri melalui data yang diberikan[21]. Menciptakan sebuah program yang dapat belajar memerlukan data. Proses pembelajaran sebuah komputer hampir mirip dengan manusia yaitu memerlukan contoh. Beberapa contoh dari penerapan machine learning yaitu computer vision, pemrosesan bahasa alami (Natural Language Processing), program diagnosa medis[22]. Machine Learning dibagi menjadi beberapa jenis pembelajaran yaitu supervised learning, unsupervised learning, dan reinforcement learning[21].

H. Metode Inkremental

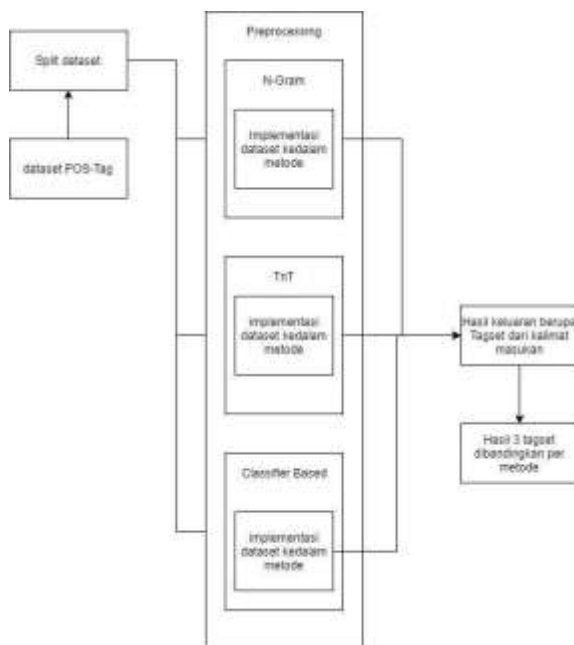
Model inkremental merupakan sebuah model dalam pengembangan suatu perangkat lunak. Metode ini membagi setiap kebutuhan pada software menjadi beberapa fungsi atau bagian yang dikerjakan secara bertahap[23].

Model inkremental memiliki beberapa tahapan, yaitu[23]:

1. Requirement adalah proses menentukan kebutuhan dan analisis mengenai kebutuhan tersebut.
2. Spesification adalah proses menentukan spesifikasi menggunakan analisis dari kebutuhan sebagai acuan.

3. Architecture Design adalah perancangan software yang digunakan untuk perancangan sistem pembangunan setiap bagian pada tahap berikutnya.
4. Code merupakan tahap pembuatan kode.
5. Test merupakan tahapan melakukan testing pada model.

I. Kerangka Pemikiran



Gambar 3 Kerangka Pemikiran

Penelitian ini diharapkan dapat menghasilkan aplikasi yang dapat mengeluarkan tag atau label dari kata teks menggunakan machine learning dan library NLTK. Pada pengembangan aplikasi ini membutuhkan data teks sebagai data utama. Pengguna memasukan teks yang ingin dicari dan aplikasi dapat memproses kata yang terdapat pada teks, dengan menggunakan proses machine learning diharapkan aplikasi ini mampu menghasilkan hasil keluaran berupa penggalan tag atau label dari teks.

Pada awal penelitian, objek penelitian melakukan beberapa observasi yang mendukung tujuan dan faktor

penting dari penelitian ini. Observasi dilakukan dengan mencari informasi dengan teori-teori yang mendukung penelitian. Teori-teori tersebut berasal dari buku dan jurnal penelitian sebelumnya. Observasi dilakukan untuk mengetahui masalah yang menyebabkan penelitian ini dilakukan.

Pada penelitian ini, objek penelitian menggunakan metode inkremental. Dengan metode inkremental, peneliti dapat membuat aplikasi dari tahap inkremental awal hingga tahap inkremental akhir dengan menambah modul secara bertahap. Metode inkremental digunakan, dibagi menjadi dua modul, setiap modul terdiri dari empat tahap, yaitu analisis, desain, implementasi, dan pengujian.

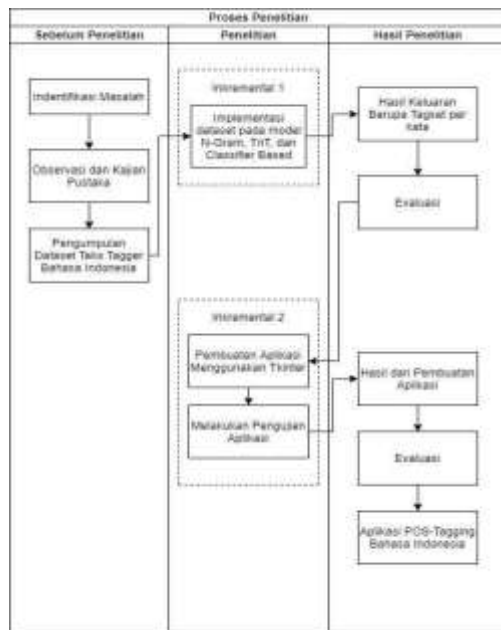
Tahap analisis adalah tahap yang dilakukan untuk mencari hal yang dibutuhkan, menentukan tujuan, mendefinisikan proses dan merancang konsep. Tahap desain adalah tahap yang dilakukan untuk merancang proses, mengurutkan proses, menjelaskan proses secara detail. Tahap implementasi adalah tahap yang dilakukan untuk menjelaskan secara teknis dari setiap proses dan hal yang dibutuhkan. Tahap pengujian adalah tahap yang dilakukan untuk menguji kesesuaian hasil implementasi dengan tujuan.

J. Proses Penelitian

Pada penelitian ini diperlukan dataset yang merupakan data teks yang berisikan penggalan kata Bahasa Indonesia yang sudah labeling dengan arti dari kata-kata tersebut. Penelitian Ini menggunakan metode rule-based dan metode machine learning, metode rule-based merupakan metode yang menerapkan kamus sebagai acuan dari implementasi penelitian, yang terdiri dari nama negara, kota, perusahaan ataupun nama-nama sejenis, sedangkan metode machine learning merupakan metode yang digunakan untuk melakukan

klasifikasi pada teks dengan cara mempelajari pola pada model sebelumnya yang terlebih dahulu di-train.

Pengambilan data dilakukan dengan cara observasi dalam laman internet (github) dan mengunduh data tersebut untuk dijadikan dataset, data yang diunduh merupakan file dengan format tsv dan txt.



Gambar 3. 1 Proses Penelitian dengan model inkremental

K. Preprocessing Data

Data yang didapatkan merupakan dataset POS-Tagging yang diaplikasikan kedalam bahasa Indonesia. Kumpulan tagset kata yang digunakan adalah 'cc', 'cd', 'od', 'dt', 'fw', 'in', 'jj', 'md', 'neg', 'nn', 'nnp', 'nnd', 'pr', 'prp', 'rb', 'rp', 'sc', 'sym', 'uh', 'vb', 'wh', 'x', dan 'z'[24], istilah kata tersebut disebut dengan tagset.

Data yang digunakan merupakan dataset Indonesian_Manually_Tagged_Corpus yang sudah terlebih dahulu dilakukan tagging oleh objek penelitian sebelumnya dan data ini sering digunakan oleh banyak objek peneliti

dalam melakukan penelitiannya untuk kasus POS-Tagging bahasa Indonesia.

Kera	NN
untuk	SC
amankan	VB
pesta olahraga	NN
Pemerintah	MNP
kota	NMP
delhi	NMP
mengerahkan	VB
monyet	NN
untuk	SC
mengusir	VB
monyet-monyet	NN
lain	JJ
yang	SC
berbedan	VB
lebih	RB
kecil	JJ
dari	IN
arena	NN
Pesta Olahraga	MNP
Persamanakuran	MNP
z	

Gambar 3. 2 Data Teks yang sudah diberikan tag

Pada Gambar 3.2 dataset berupa data dengan format tab-separated file (.tsv), yang mengartikan setiap baris berisi token dengan part-of-speech tag dari token tersebut yang terpisahkan oleh satu karakter tab (\t) membuat antar kalimat dipisahkan oleh satu baris kosong[22].

L. Proses Pengembangan Aplikasi

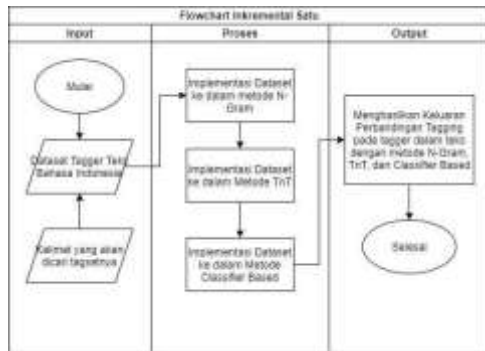
Pada penelitian ini tahap pengembangan aplikasi dibagi menjadi dua tahapan pengembangan yaitu pengembangan inkremental satu dan pengembangan inkremental dua. Inkremental satu merupakan tahap pembuatan model klasifikasi dalam penelitian, sedangkan inkremental satu merupakan tahap pembuatan tampilan antar muka dari aplikasi hingga fungsi-fungsi yang ada pada tampilan tersebut.

M. Inkremental Satu

1. Analisa

Tahap inkremental satu merupakan tahap pengerjaan membentuk aplikasi, menggunakan dataset sebagai tahap implementasi yang akan dilakukan untuk setiap metode yang digunakan, objek penelitian menggunakan metode N-Gram, TnT, dan ClassifierBased.

Kalimat yang akan digunakan menentukan hasil tagger yang akan dikeluarkan dan kalimat tersebut digunakan pada setiap metode untuk melihat perbandingan tagger pada kata dari kalimat yang akan dikeluarkan oleh setiap metode.



Gambar 3.4 Flowchart Inkremental Satu

Objek penelitian menggunakan peralatan pendukung proses pengerjaan aplikasi, perangkat yang objek peneliti gunakan akan disebutkan pada tabel 3.2.

Tabel 3. 1 Perangkat Keras Pembuatan Aplikasi

Nama Perangkat Keras	Spesifikasi	Perangkat Lunak
Laptop Lenovo Ideapad 310	Processor :2.3 Ghz-2.4Ghz, Intel Core i5 Memory : 12 GB Graphic : Intel® HD Graphics 520 / GeForce GT920MX Storage : 1 TB	Anaconda Spyder Python 3.7

Pengembangan aplikasi dibutuhkan perangkat keras dan lunak untuk menunjang proses pengembangan aplikasi, perangkat keras berupa laptop yang digunakan untuk mendapatkan data berupa dataset dan sebagai tempat mengolah dataset tersebut. Laptop yang digunakan memiliki perangkat lunak seperti pada tabel 3.1. Serta bahasa pemrograman yang digunakan untuk membangun aplikasi adalah bahasa

pemrograman python versi 3.7 yang diimplementasikan dengan menggunakan perangkat lunak Spyder.

Pada tahap inkremental satu peneliti melakukan observasi atau pencarian literasi penelitian mengenai penelitian POS-Tagging yang dilakukan oleh peneliti sebelumnya dan mencari dataset yang dapat peneliti gunakan sebagai bahan penelitian, data yang digunakan berupa data teks.

Pada objek penelitian dilakukan implementasi sebanyak tiga kali yaitu dengan menggunakan metode n-gram tagging, metode TnT, dan metode Classifier Based tagging. Tiga metode tersebut dilakukan untuk melihat perbedaan tagging yang dilakukan dengan menggunakan modul pada metode-metode tersebut.

Tagging dengan metode N-Gram data diproses dengan metode n-gram untuk melihat kata sambungan dari data teks yang digunakan, unigram digunakan pada data teks untuk token yang terdiri dari satu kata, bigram, digunakan pada data teks untuk token yang terdiri dari dua kata, sedangkan trigram digunakan pada data teks untuk token yang terdiri dari tiga kata.

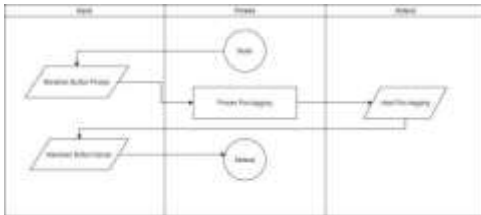
Tagging dengan metode TnT Data diproses dengan metode TnT untuk melihat kata sambungan dari teks yang digunakan, dan lebih spesifik digunakan pada teks yang dicocokkan untuk tiga kata.

Tagging dengan metode Classifier Based merupakan metode yang melakukan beberapa pencocokan regular ekspresi dengan memperhatikan sejarah metode unigram, bigram, dan trigram.

N. Inkremental Dua

Pada Inkremental dua merupakan pembuatan tampilan antarmuka serta fungsi aplikasi POS-Tagging berbahasa Indonesia. Pembuatan aplikasi menggunakan library

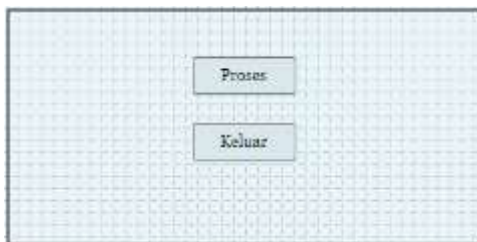
Tkinter. Pada pembuatan tampilan antar muka aplikasi dengan memanfaatkan library Tkinter. Aplikasi ini digunakan untuk membuat desain untuk tampilan antarmuka aplikasi. Keluaran yang dihasilkan dari aplikasi ini merupakan file '.py'.



Gambar 3.9 Flowchart Aplikasi

Pada Gambar 3.9 tahap penggunaan aplikasi, pengguna dapat menekan button proses untuk melakukan proses POS-Tagging dengan fungsi yang sudah ditambahkan dari proses pengembangan inkremental satu. Hasil keluaran akan berupa teks. Akan tetapi input yang dilakukan hanya melalui input manual pada coding, untuk keluar dari aplikasi pengguna dapat menekan button keluar, dan berikut tampilan mockup aplikasi POS-Tagging

2. Desain



Gambar 3.10 Tampilan Mockup Aplikasi



Gambar 3.11 Tampilan Mockup Hasil Keluaran Aplikasi

3. Implementasi

Implementasi dilakukan untuk membuat Graphic User Interface (GUI) dengan menggunakan bahasa pemrograman python, sehingga perancangan dapat dilakukan dengan menggunakan library khusus untuk membuat tampilan pengguna.

4. Pengujian

Tabel 3 Tabel Uji BlackBox inkremental dua

Nama Proses	Skenario	Kode	Harapan
Melakukan proses dengan kata yang dimasukan oleh pengguna.	Memasukan data berupa teks untuk proses dan ditampilkan oleh POS-Tagger.	Memproses data teks untuk dilakukan tagging dan menampilkan hasil.	Dapat menghasilkan keluaran berupa tagger berdasarkan kata dari kalimat yang dimasukkan.

Pada Tabel 3.3 menjelaskan seluruh proses pengembangan inkremental dua yang diuji dengan Blackbox, dengan memperhatikan skenario yang telah ditentukan, dengan menggunakan kode yang akan digunakan serta uji Blackbox ini akan dibandingkan dengan harapan objek penelitian.

III. HASIL DAN PEMBAHASAN

1. Inkremental Satu

Pada tahap inkremental satu dilakukan pengujian dari fungsi yang dibuat menggunakan bahasa pemrograman python, implementasinya sebagai berikut.

A. Implementasi

Pada Tahap Implementasi akan ditampilkan hasil keluaran pada setiap

tahap implementasi. Penjelasan kode akan dilakukan dengan cara membahas setiap fungsi-fungsi yang akan digunakan dalam proses pembuatan aplikasi, implementasi dilakukan untuk menghasilkan keluaran sesuai dengan metode-metode yang digunakan, berikut beberapa tahapannya:

- a) Membaca dataset
- b) Memeriksa tagging pada kata
- c) Memeriksa kata yang memiliki lebih dari satu tag
- d) Tag yang sering digunakan pada kata
- e) Banyaknya frekuensi kata dengan tagging NN

B. Hasil

Pada inkremental satu Pada inkremental satu, tahap implementasi dilakukan dengan menjalankan perintah koding pada setiap model tagging yang terdiri dari model N-Gram, model TnT, dan model Classifier Based. Berikut merupakan koding dan penjelasan hasil keluaran model tagging.

Ketiga model tagging tersebut dijalankan satu per-satu sebelum menjalankan koding hasil. Pada koding hasil peneliti menggunakan kalimat "Budi merupakan mahasiswa di Universitas-Jakarta angkatan-2016, dia menjadi salah satu mahasiswa terbaik di Jakarta". Kalimat tersebut digunakan pada setiap model untuk melihat perbedaan tagging yang diterapkan oleh ketiga model tersebut.

C. Model N-Gram

model N-Gram digunakan dengan memperhatikan token-token pada kalimat yang dipenggal menjadi kata-kata, seperti unigram untuk token dengan satu kata, bigram untuk token dengan dua kata, dan trigram untuk token dengan tiga

kata. dan model ini memeriksa kata mulai dari bigram sampai ke trigram.

model N-Gram hasil keluaran pada kalimat yang dimasukkan menghasilkan tagging berupa (Budi, NNP), (Merupakan, VB), (mahasiswa, NN), (di, IN), (Universitas-Jakarta, NN), (angkatan-2016, NN), ('', Z), (dia, PRP), (menjadi, VB), (salah, JJ), (satu, CD), (mahasiswa, NN), (terbaik, JJ), (di, IN), (Jakarta, NNP).

D. Model TnT

model TnT pada umumnya mirip dengan model N-Gram, perbedaannya terletak pada pemenggalan kata yang dipakai pada model ini, yaitu model ini hanya memperhatikan trigram untuk token pada kata, sehingga kalimat dibaca dalam model ini per tiga kata untuk menghasilkan tagging pada kata.

model TnT hasil keluaran pada kalimat yang dimasukkan menghasilkan tagging berupa (Budi, NNP), (Merupakan, VB), (mahasiswa, NN), (di, IN), (Universitas-Jakarta, NN), (angkatan-2016, NN), ('', Z), (dia, PRP), (menjadi, VB), (salah, JJ), (satu, CD), (mahasiswa, NN), (terbaik, JJ), (di, IN), (Jakarta, NNP).

E. Model Classifier Based

Model Classifier Based digunakan sebagai metode terakhir dalam penelitian dikarenakan metode Classifier Based bekerja dengan memperhatikan jejak pengerjaan model N-Gram dan model TnT.

model Classifier Based hasil keluaran pada kalimat yang dimasukkan menghasilkan tagging berupa (Budi, NNP), (Merupakan, VB), (mahasiswa, NN), (di, IN), (Universitas-Jakarta, NNP), (angkatan-2016, CD), ('', Z), (dia, PRP), (menjadi, VB), (salah, JJ), (satu, CD), (mahasiswa, NN), (terbaik, JJ), (di, IN), (Jakarta, NNP).

F. Evaluasi

Pada penggunaan fungsi untuk membaca dataset pada setiap metode perlu dijalankan satu per-satu sebelum koding terakhir untuk menjalankan proses POS-Tagging dapat dilakukan, dikarenakan setiap fungsi modul memiliki peranan masing-masing dalam melakukan tagging pada dataset teks yang digunakan.

Pada implementasi dari ketiga model dapat ditemukan perbedaan tagging antara model N-Gram dan model TnT dengan model Classifier Based yang terletak pada tagging kalimat “Universitas-Jakarta” dan “angkatan-2016”, perbedaan yang terdapat bahwa dalam model N-Gram dan model TnT kata “Universitas-Jakarta” dan “angkatan-2016” memiliki tagging “NN” dan “NN”, sedangkan dalam model Classifier Based kata “Universitas-Jakarta” dan “angkatan-2016” memiliki tagging “NNP” dan “CD”

Menjelaskan bahwa pada tagging model N-Gram dan model TnT kata “Universitas- Jakarta” dan “angkatan-2016” di tagging menjadi kata benda, sedangkan pada model Classifier Based kata “Universitas-Jakarta” dan “angkatan-2016” di tagging menjadi kata sebuah tempat dan kata numeralia kardinal atau kata yang didefinisikan untuk pertanyaan “berapa?”.

Metode Classifier Based menjadi metode yang optimal dalam menghasilkan tagging kata pada kalimat dibandingkan dengan kedua metode lainnya yaitu metode N-Gram dan metode TnT.

5. Inkremental Dua

Pada tahap inkremental dua akan dilakukan pembuatan graphic user interface menggunakan library tkinter, implementasinya sebagai berikut:

A. Implementasi

Pada tahap implementasi akan dijelaskan kode-kode pembuatan graphic user interface. Penjelasan kode akan dilakukan dengan cara membahas setiap fungsi yang akan digunakan didalam proses pembuatan aplikasi, implementasi pada inkremental dua dilakukan untuk membuat fungsi yang menghasilkan model, beberapa tahapannya yaitu:

1. Define Proses Program
2. Pembuatan User Interface

B. Hasil

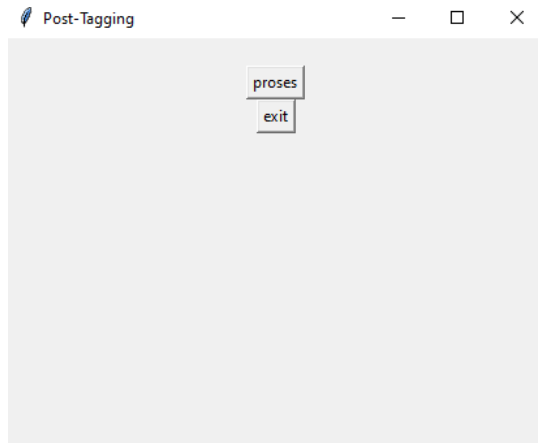
Tabel 4. 1 Hasil pengujian blackbox inkremental dua

<i>Nama Proses</i>	<i>Skenario</i>	<i>Kode</i>	<i>Haran</i>	<i>Hasil</i>
<i>Melakukan proses sing data dengan kata yang dimasukan oleh pengguna.</i>	<i>Memasukan data berupa teks untuk di proses dan ditampilkan oleh POS-Tagger.</i>	<i>Memproses data teks untuk dilakukan tagging dan menampilkan hasil.</i>	<i>Dapat menghasilkan tagging yang optimal pada setiap kata yang dima</i>	<i>Valid</i>

Pada Tabel 4.1 pengujian blackbox pada inkremental dua dilakukan untuk melakukan pengujian tagging teks berdasarkan teks yang dimasukan oleh pengguna, proses pengujian dengan metode Blackbox berhasil dicapai berdasarkan skenario yang diharapkan dengan menggunakan kode untuk melakukan proses tersebut.

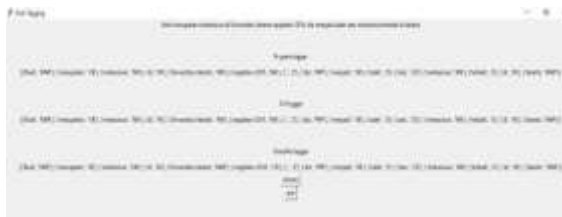
Pada tahap implementasi akan dijelaskan tampilan user dan hasil keluaran dari aplikasi berupa

perbandingan tagging tiga metode akan ditampilkan pada Gambar 4.20.



Gambar 4.20 Tampilan Aplikasi

Pada Gambar 4.20 menunjukkan tampilan pengguna ketika koding dijalankan dengan ditampilkan dua button ditampilmnya.



Gambar 4.21 Tampilan Ketika tombol proses ditekan

Tampilan Aplikasi setelah tombol proses ditekan dan tampilan yang diberikan berupa hasil keluaran POS-Tagging dengan tiga model yang digunakan, ditampilkan seperti Gambar 4.21.



Gambar 4.22 Tampilan untuk pemeriksaan tagging

Tampilan aplikasi setelah tombol proses ditekan dan tampilan yang diberikan berupa hasil keluaran dari

pemeriksaan tagging yang sudah diproses, ditampilkan seperti Gambar 4.22.

IV. SIMPULAN

Setelah dari hasil pembuatan aplikasi POS-Tagging teks bahasa Indonesia dapat ditarik simpulan sebagai berikut:

1. Objek Penelitian dilakukan dengan menggunakan tiga metode yaitu metode N-Gram, metode TnT, dan Metode Classifier Based
2. Penelitian ini menghasilkan hasil keluaran berupa perbandingan tagger antara ketiga model yang digunakan dan hanya berfokus pada perbandingan tagging tersebut
3. Proses melakukan tagging dipengaruhi oleh kalimat yang dimasukan oleh pengguna
4. Metode yang menghasilkan hasil tagger optimal pada dataset bahasa Indonesia adalah metode Classifier Based dikarenakan metode ini melihat proses dari dua metode lainnya yaitu N-Gram dan TnT.

DAFTAR RUJUKAN

- [1] D. Sulistyonni, "Masyarakat Linguistik Komputasional inginkan korpus bahasa Indonesia", Antara News, 2019. [Daring]. Tersedia: <https://www.antaraneews.com/berita/962732/masyarakat-linguistik-komputasional-inginkan-korpus-bahasa-indonesia#mobile-src>. [Akses: 09-Aug-2020].
- [2] A. Haryanto, "Riset: Ada 175,2 Juta Pengguna Internet di Indonesia", detiknet, 2020. [Daring]. Tersedia: <https://inet.detik.com/cyberlife/d-4907674/riset-ada-1752-juta-pengguna-internet-di-indonesia>. [Akses: 07- Mar- 2020].

- [3] F. Rashel, A. Luthfi, A. Dinakaramani and R. Manurung, "POS Tag - Tagger Site", Bahasa.cs.ui.ac.id, 2020. [Daring]. Tersedia: <http://bahasa.cs.ui.ac.id/postag/tagger>. [Akses: 10- Mar- 2002].
- [4] D. Hartono, "Part of Speech Tagging", School of Computer Science, 2020. [Daring]. Tersedia: <https://socs.binus.ac.id/2019/12/31/part-of-speech-tagging/>. [Akses: 10- Mar- 2020].
- [5] Bird, S., Klein, E. and Loper, E., 2009. Natural Language Processing With Python. Beijing: O'Reilly.
- [6] Pustejovsky, J., 2013. Natural Language Annotation - For Machine Learning. [Place of publication not identified]: Shroff Publishers & Distr
- [7] Setyo Nugroho, K., 2019. Dasar Text Preprocessing Dengan Python. [Daring] Medium. Tersedia at: <https://medium.com/@ksnugroho/dasar-text-preprocessing-dengan-python-a4fa52608ffe> [Akses 7 March 2020].
- [8] S. Rostianingsih, S. Andrian Sugianto and Liliana, Repository.petra.ac.id, 2020. [Daring]. Tersedia: http://repository.petra.ac.id/16621/1/Publikasi1_01043_1332.pdf. [Akses: 13- Apr- 2020].
- [9] W. Candra Indhiarta, Eprints.ums.ac.id, 2020. [Daring]. Tersedia: <http://eprints.ums.ac.id/55077/1/NASKAH%20PUBLI KASI.pdf>. [Akses: 13- Apr- 2020].
- [10] "Sejarah Python" 2020. [Daring]. Tersedia: <https://www.pythonindo.com/sejarah-python/>. [Akses: 14- May- 2020].
- [11] "Tutorial Belajar Bahasa Pemrograman Python Untuk Pemula," Duniaikom, [Daring]. Tersedia: <https://www.duniaikom.com/tutorial-belajar-bahasa-pemrograman-python-untuk-pemula/>. [Akses 24- May- 2020].
- [13].N. Zakiah, "Gak Nyangka, 8 App Populer Ini Dibuat dengan Bahasa Pemrograman Phyton," 23 October 2019. [Daring]. Tersedia: <https://www.idntimes.com/tech/trend/nena-zakiah-1/aplikasi-yang-dibuat-dengan-bahasa-pemrograman-phyton/6>. [Akses 24- May- 2020].
- [14]. Repository.unpas.ac.id, 2020. [Daring]. Tersedia: http://repository.unpas.ac.id/28589/9/_III%20%20BA B%20III%20EXPLORASI.pdf. [Akses: 24- May- 2020].
- [15]R.T. Tedja, "SOFTWARE DEVELOPMENT MODEL: INKREMENTAL MODEL," Bina Nusantara, 2 July 2019. [Daring]. Tersedia: <https://sis.binus.ac.id/2019/07/02/software-development-model-inkremental-model/>. [Akses 7 June 2020].
- [16]D. Suhartono, "Dasar Pemahaman Neural Network," Bina Nusantara, [Daring]. Tersedia: <https://socs.binus.ac.id/2012/07/26/konsep-neural-network/>. [Akses 10 June 2020].
- [17]S. Sena, "Pengenalan Deep Learning Part 1 : Neural Network," 28 October 2017. [Daring]. Tersedia: <https://medium.com/@samuelsena/pengenalan-deep-learning-8fbb7d8028ac>. [Akses 10 June 2020].
- [18]S. Sharma, "Activation Functions in Neural Networks," 6 September 2017. [Daring]. Tersedia: <https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6>. [Akses 10 June 2020].
- [19]"Pengenalan terhadap Machine Learning," Makers Institute, 8 February 2018. [Daring]. Tersedia: <https://medium.com/@makersinstitute/pengenal-an-terhadap-machine-learning-9011fe71d1e4>. [Akses 3 June 2020].
- [20]M. Bigsmile, "Mengenal Teknologi Machine Learning (Pembelajaran Mesin)," 17 May 2016. [Daring]. Tersedia: <https://www.codepolitan.com/mengena-teknologi-machine-learning-pembelajaran-mesin>. [Akses 3 June 2020].
- [21]S. Arora, "Supervised vs Unsupervised vs Reinforcement," 29 January 2020. [Daring]. Tersedia: <https://www.aitude.com/supervised-vs-unsupervised-vs-reinforcement/>. [Akses 3 June 2020].
- [22]P. kurniawati, "Pengujian Sistem", Medium, 2018. [Daring]. Tersedia: <https://medium.com/skyshidigital/pengujian-sistem-52940ee98c77>. [Akses: 24- Jun- 2020].