**PERBANDINGAN TIPE METODE POS TAGGER TERHADAP NILAI AKURASI UNTUK BAHASA MELAYU PONTIANAK**

**SKRIPSI**

Jurusan Informatika

Program Studi Sarjana Informatika

Oleh:

**ADE SUMOKO**

NIM D1042131030



**FAKULTAS TEKNIK**

**UNIVERSITAS TANJUNGPURA**

**PONTIANAK**

**2020**

Halaman Pernyataan

Yang bertanda tangan di bawah ini:

Nama : Ade Sumoko

NIM : D1042131030

Menyatakan bahwa dalam skripsi yang berjudul “Perbandingan Tipe Metode PoS Tagger Terhadap Nilai Akurasi Untuk Bahasa Melayu Pontianak” tidak terdapat karya yang pernah diajukan untuk memperoleh gelar sarjana di suatu perguruan tinggi manapun. Sepanjang pengetahuan Saya, tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam Daftar Pustaka.

Demikian pernyataan ini dibuat dengan sebenar-benarnya. Saya sanggup menerima konsekuensi akademis dan hukum di kemudian hari apabila pernyataan yang dibuat ini tidak benar.

Pontianak, Desember 2020

ADE SUMOKO

NIM D1042131030

KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN

UNIVERSITAS TANJUNGPURA

**FAKULTAS TEKNIK**

Jalan Prof. Dr. H. Hadari Nawawi Pontianak 78124

Telepon (0561) 740186 Faximili (0561) 740186

Email : [ft@untan.ac.id](mailto:ft@untan.ac.id) Website : [http://teknik.untan.ac.id](http://teknik.untan.ac.id/)



**HALAMAN PENGESAHAN**

“PERBANDINGAN TIPE METODE POS TAGGER TERHADAP NILAI AKURASI UNTUK BAHASA MELAYU PONTIANAK”

Jurusan Informatika

Program Studi Sarjana Informatika

Oleh : “Ade Sumoko”

NIM. D1042131030

Telah dipertahankan didepan Penguji Skripsi pada tanggal 24 Desember 2020 dalam sidang secara daring (*online*) dan diterima sebagai salah satu persyaratan untuk memperoleh gelar sarjana.

Susunan Penguji Skripsi :

|  |  |  |
| --- | --- | --- |
| Dosen Pembimbing Utama | : Dr. Arif Bijaksana Putra Negara, S.T., M.T., | (NIP. 197208081998021002 ) |
| Dosen Pembimbing Kedua | : Helen Sasty Pratiwi, ST, M.Eng | (NIP. 198601172012122004) |
| Dosen Penguji Utama | : Tursina, S.T., M.Cs. | (NIP. 197801152002122003) |
| Dosen Penguji Kedua | : Morteza Muthahhari, S.Kom., M.T.I. | (NIP. 198607092019031008) |
|  |  |  |

Wakil Dekan Bidang Akademik,

Dr.-Ing. Ir. Slamet Widodo, M.T.,IPM.  
NIP. 196712231992031002

Pontianak, Desember 2020

Dekan,

Dr.rer.nat. Ir. R. M. Rustamaji, M.T., IPU

NIP. 196801161994031003

**HALAMAN PERSEMBAHAN**

Skripsi ini saya persembahkan untuk kedua orang tua dan kakak-kakak saya yang tiada hentinya memberikan dukungan selama menjalani kuliah.

Terima kasih saya ucapkan kepada rekan-rekan mahasiswa Informatika Universitas Tanjungpura angkatan 2013 yang telah saling mendukung selama perkuliahan, saat skripsi, dan di luar bidang akademik.

Serta seluruh kerabat, dosen, staff TU, dan pihak yang telah membantu, yang tidak dapat disebutkan satu per satu. Semoga Tuhan senantiasa memberikan balasan kebaikan.

Kata Pengantar

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa, karena berkat rahmat-Nya, penelitian dan penulisan skripsi yang berjudul “Perbandingan Tipe Metode PoS Tagger Terhadap Nilai Akurasi Untuk Bahasa Melayu Pontianak” ini dapat diselesaikan sebagai syarat untuk mengambil gelar Sarjana Komputer (S.Kom.).

Melalui penelitian ini, ucapan terimakasih saya sampaikan kepada pihak-pihak yang telah membantu saya baik berupa materil maupun dukungan moril yaitu kepada Bapak **Dr. Arif Bijaksana Putra Negara, S.T., M.T.,** selaku Dosen Pembimbing Pertama sekaligus Dosen Pembimbing Akademik, yang telah memberikan banyak bimbingan, saran dan masukan yang membangun dalam melakukan perancangan dan pembuatan skripsi ini, dan kepada Ibu **Helen Sasty Pratiwi, ST, M.Eng** selaku Dosen Pembimbing Kedua yang telah memberikan banyak bimbingan, membantu dalam menyelesaikan penelitian ini.

Kepada Dosen Penguji Pertama Ibu **Tursina, S.T., M.Cs.** dan Dosen Penguji Pendamping Bapak **Morteza Muthahhari, S.Kom., M.T.I.** yang telah memberikan saran serta masukan untuk perbaikan skripsi ini. Tentunya saya sadari terdapat banyak kekurangan dalam penyusunan skripsi ini, oleh karena itu kritik dan saran saya butuhkan. Semoga skripsi ini bermanfaat bagi perkembangan ilmu pengetahuan.

Pontianak, Desember 2020

Penulis,

ADE SUMOKO

NIM. D1042131030

Abstrak­­

Metode PoS Tagger adalah metode yang digunakan sebagai acuan untuk melakukan tagging pos secara langsung sesuai dengan data uji yang dan data latih yang telah dilakukan untuk mendeteksi seberapa akurat metode pos tagger tersebut dalam melakukan tagging, yang dimana hal ini dilakukan pengecekan dengan data tagging secara manual untuk melihat kearutan dari metode tersebut. Data kalimat teks korpus yang digunakan sebanyak 1500 kalimat bahasa Melayu Pontianak yang dimana kalimat teks digunakan dalam penelitian ini sebagai kalimat latih maupun kalimat uji. Penelitian ini melakukan pengembangan korpus yang awalnya berjumlah 500 kalimat menjadi 1500 kalimat, dan perbandingan tag PoS yang menggunakan data tag PoS dari penelitian lainnya untuk mendapatkan tag PoS yang baru, kemudian hasil dari analisa penelitian ini adalah mendapatkan nilai akurasi dari setiap metode PoS tagger.

Kata Kunci : PoS Tagger, Bahasa Melayu Pontianak, Metode PoS Tagger, Nilai Akurasi.

Abstract­­

PoS Tagger method is a method used as a reference to do tagging the post directly in accordance with the test data and training data that has been done to detect how accurate the tagger post method is in tagging, which is done checking with tagging data manually to see the wood of the method. Corpus text sentence data used as many as 1500 sentences Pontianak Melayu’s language where the sentence text is used in this study as a sentence training and test sentences. This research carried out the development of corpus which initially amounted to 500 sentences to 1500 sentences, and comparison of PoS tags that use PoS tag data from other studies to get new PoS tags, then the result of this research analysis is to get the accuracy value of each PoS tagger method

Keywords: PoS Tagger, Pontianak Melayu Language, PoS Tagger Method, accuracy value.

vii

Daftar Isi

[Halaman Pernyataan ii](#_Toc60606521)

[Halaman Persetujuan iii](#_Toc60606522)

[Halaman Persembahan iv](#_Toc60606523)

[Kata Pengantar v](#_Toc60606524)

[Abstrak vi](#_Toc60606525)

[Abstract vii](#_Toc60606526)

[Daftar Isi viii](#_Toc60606527)

[Daftar Tabel xi](#_Toc60606528)

[Daftar Gambar xii](#_Toc60606529)

[Daftar Kode Program xiii](#_Toc60606530)

[Daftar Grafik xiv](#_Toc60606531)

[Daftar Lampiran xv](#_Toc60606532)

[BAB I Pendahuluan 1](#_Toc60606533)

[1.1 Latar Belakang 1](#_Toc60606534)

[1.2 Rumusan Masalah 2](#_Toc60606535)

[1.3 Tujuan Penelitian 2](#_Toc60606536)

[1.4 Pembatasan Masalah 2](#_Toc60606537)

[1.5 Sistematika Penulisan 3](#_Toc60606538)

[BAB II Tinjauan Pustaka 4](#_Toc60606539)

[2.1 Penelitian Terkait 4](#_Toc60606540)

[2.2 Part Of Speech Tagging 6](#_Toc60606541)

[2.3 Metode PoS Tagger 17](#_Toc60606542)

[2.4 Python 17](#_Toc60606543)

[2.5 *Natural Language Toolkit* 18](#_Toc60606544)

[2.6 Bahasa Melayu Pontianak 19](#_Toc60606545)

[2.7 Pengujian Teks kalimat metode PoS Tagger 19](#_Toc60606546)

[2.8 Pengujian *Accuracy, Precision, Recall,* dan *F-measure* 20](#_Toc60606547)

[2.9 Pengujian K-*fold Cross Validation* 20](#_Toc60606548)

[BAB III Metodologi Penelitian 22](#_Toc60606549)

[3.1 Bahan Penelitian 22](#_Toc60606550)

[3.2 Alat Bantu Penelitian 22](#_Toc60606551)

[3.2.1 Perangkat Keras 22](#_Toc60606552)

[3.2.2 Perangkat Lunak (*Software*) 22](#_Toc60606553)

[3.3 Metode Penelitian 22](#_Toc60606554)

[3.3.1 Persiapan Data 24](#_Toc60606555)

[3.3.2 Pengumpulan Tag PoS 24](#_Toc60606556)

[3.3.3 Pengembangan Tag PoS 24](#_Toc60606557)

[3.3.4 Pengembangan Korpus 25](#_Toc60606558)

[3.3.5 *Tagging* Korpus 25](#_Toc60606559)

[3.3.6 Pelatihan Korpus 25](#_Toc60606560)

[3.3.7 Perancangan Pengujian 26](#_Toc60606561)

[3.3.8 Analisis Hasil Pengujian 28](#_Toc60606562)

[3.3.9 Kesimpulan Hasil Pengujian 28](#_Toc60606563)

[BAB IV Hasil dan pengujian 29](#_Toc60606564)

[4.1 Hasil Penelitian 29](#_Toc60606565)

[4.1.1 Pembuatan Korpus untuk *Tagging* PoS 30](#_Toc60606566)

[4.1.2 Hasil *Tagging* PoS Metode HMM untuk Bahasa Melayu Pontianak 30](#_Toc60606567)

[4.1.3 Hasil *Tagging* PoS Metode Brigram untuk Bahasa Melayu Pontianak 30](#_Toc60606568)

[4.1.4 Hasil *Tagging* PoS Metode Perceptron untuk Bahasa Melayu Pontianak 32](#_Toc60606569)

[4.1.5 Hasil *Tagging* PoS Metode Trigram untuk Bahasa Melayu Pontianak 33](#_Toc60606570)

[4.1.6 Hasil *Tagging* PoS Metode Unigram untuk Bahasa Melayu Pontianak 35](#_Toc60606571)

[4.2 Hasil Pengujian 36](#_Toc60606572)

[4.2.1 Pengujian *Accuracy, Precision, Recall* dan F*-measure* 36](#_Toc60606573)

[4.2.2 Pengujian K-*fold Cross Validation* 38](#_Toc60606574)

[4.3 Analisa Hasil Keseluruhan Pengujian 39](#_Toc60606575)

[BAB V Kesimpulan dan Saran 41](#_Toc60606576)

[5.1 Kesimpulan 41](#_Toc60606577)

[5.2 Saran 41](#_Toc60606578)

[Daftar Pustaka 42](#_Toc60606579)

Daftar Tabel

[**Tabel 2 1** Kajian Penelitian Terkait 5](#_Toc60606915)

[**Tabel 3. 1** Rencana Tabel Hasil Nilai Pengujian Accuracy, Precision, Recall, dan F-Measure 27](#_Toc60606212)

[**Tabel 3. 2** Tabel Pengujian 5 Fold Cross Validation 28](#_Toc60606213)

[**Tabel 3. 3** Rencana Tabel Hasil Perhitungan Accuracy, Precision, Recall, dan F-measure Pengujian K-Fold Cross Validation 28](#_Toc60606214)

[**Tabel 4. 1** *Set* PoS Bahasa Melayu Pontianak 29](#_Toc60604740)

[**Tabel 4. 2** Tabel Hasil Pengujian *Accuracy, Precision, Recall* dan F-*measure* 36](#_Toc60604741)

[**Tabel 4. 3** Tabel Hasil Pengujian K-*fold Cross Validation* 38](#_Toc60604742)

Daftar GAMBAR

[**Gambar 2.1** *Set* PoS Arawinda Dinakaramani (2014) 7](#_Toc60604755)

[**Gambar 2. 2** Set PoS Pisceldo(2009) Tabel 1 8](#_Toc60604756)

[**Gambar 2. 3** Set PoS Pisceldo(2009) Tabel 2 9](#_Toc60604757)

[**Gambar 2. 4** Set PoS Alfan Farizki Wicaksono (2010) 10](#_Toc60604758)

[**Gambar 2. 5** Set PoS Adriani (2009) 11](#_Toc60604759)

[**Gambar 2. 6** Set PoS Setyaningsih (2017) 12](#_Toc60604760)

[**Gambar 2. 7** *Set* PoS Purwarianti (2010) 13](#_Toc60604761)

[**Gambar 2. 8** *Set* PoS Achmad Fatchuttaman Abka (2017) 14](#_Toc60604762)

[**Gambar 2. 9** Set PoS Muliono (2012) 15](#_Toc60604763)

[**Gambar 2. 10** Set PoS Septina Dian Larasati(2011) 16](#_Toc60604764)

[**Gambar 2. 11** Ilustrasi Pembuatan K-fold Cross Validation set 21](#_Toc60604765)

[**Gambar 3. 1** Diagram Alir Penelitian 23](#_Toc60604766)

[**Gambar 4. 1** Korpus Latih Proses Tagging PoS 30](#_Toc60132360)

[**Gambar 4. 2** Proses Pelatihan IPOSTagger 30](#_Toc60132361)

[**Gambar 4. 3** Hasil Tagging Metode Bigram 32](#_Toc60132362)

[**Gambar 4. 4** Hasil Tagging Metode Perceptron 33](#_Toc60132363)

[**Gambar 4. 5** Hasil Tagging Metode Trigram 34](#_Toc60132364)

[**Gambar 4. 6** Hasil Tagging Metode Unigram 36](#_Toc60132365)

Daftar kode program

[**Kode Program 3. 1** Pengujian Metode PoS Tagger 26](#_Toc60604790)

[**Kode Program 4. 1** Melatih Data Uji Metode Bigram 31](#_Toc60604796)

[**Kode Program 4. 2** Melatih Data Uji Metode Perceptron 32](#_Toc60604797)

[**Kode Program 4. 3** Melatih Data Uji Metode Trigram 33](#_Toc60604798)

[**Kode Program 4. 4** Melatih Data Uji Metode Unigram 35](#_Toc60604799)

Daftar Grafik

[**Grafik 4. 1** Hasil Pengujian Accuracy, Precision, Recall, dan F-measure 38](#_Toc60604812)

[**Grafik 4. 2** Hasil Pengujian K-fold Cross Validation 39](#_Toc60604813)

Daftar lampiran

[**Lampiran A** A-1](#_Toc60607587)

[**Lampiran B** B-1](#_Toc60607588)

[**Lampiran C** C-1](#_Toc60607589)

[**Lampiran D** D-1](#_Toc60607590)

[**Lampiran E** E-1](#_Toc60607591)

# BAB I Pendahuluan

## Latar Belakang

Bahasa merupakan salah satu alat komunikasi yang dipakai oleh manusia dalam kehidupan sehari-hari. Kapasitas kemahiran bahasa yang baik bisa memudahkan untuk berinteraksi dengan orang lain. Bahasa Melayu Pontiaak adalah bahasa yang dituturkan oleh masyarakat Pontianak dalam kehidupan sehari-hari. Hasil dari Sensus Penduduk pada tahun 2010 memperlihatkan kalau dari beraneka bahasa wilayah yang terdapat d Indonesia, persentase pemakaian bahasa melayu yang digunakan oleh masyarakat Kalimantan Barat (Na'im, 2010). Pelestarian kebudayaan bahasa daerah sangat dibutuhkan untuk menghindari ancaman dari globalisasi yang berakibat pada berkurangnya penutur dalam memakai bahasa Melayu Pontianak. Salah satunya dengan menggunakan teknologi saat ini yakni *Text To Speech* (TTS).

Teknologi *Text To Speech* (TTS) merupakan suatu sistem yang dapat mengkonversi teks dalam format suatu bahasa menjadi ucapan sesuai dengan teks dalam bahasa yang digunakan. TTS sendiri dapat diimplementasikan ke dalam suatu bentuk aplikasi telekomunikasi yang dapat mempermudah pekerjaan manusia. Selain itu TTS juga dapat membantu orang-orang dengan kebutuhan khusus seperti tunanetra ataupun tunawicara.

POS (*Part Of Speech*) merupakan kelas kata yang dipahami oleh mesin untuk menentukan jenis dari kata-kata yang terdapat pada kalimat. PoS tag adalah suatu proses yang memberikan label kelas kata secara otomatis atau manual pada setiap kata yang ada pada suatu teks atau dokumen. Sehingga untuk pembuatan PoS Tag baru maka akan menggunakan Tag PoS dari beberapa penelitian sebelumnya.

Tag PoS pada penelitian sebelumnya adalah sebagai berikut : Tag PoS Arawinda Dinakaramani(2014), Tag Pos (Pisceldo, Adriani, & Manurung, 2009), Tag PoS (Wicaksono & Purwarianti, 2010), Tag PoS (Larasati, Kuboň, & Zeman, 2011), Tag PoS (Purwarianti, 2010), Tag PoS (Arry & Arif, 2013), Tag PoS (Adriani, 2009), Tag PoS (Setyaningsih, 2017), Tag PoS Muliono (2012) yang akan dibandingkan dan diuji untuk menghasilkan sebuah tipe PoS yang baru, kemudian hasil dari perbandingan tersebut akan menjadi landasan Tag Pos dalam penelitian ini.

Pada penelitian sebelumnya terdapat PoS Tagger yang digunakan rata-rata berjumlah diatas 25, sehingga hal ini akan memberikan beban pada saat proses perhitungan dan tingkat kecepatan juga akan berpengaruh dikarenakan menghitung dengan jumlah PoS Tagger yang lebih banyak. Oleh sebab itu, dalam penelitian ini, dilakukan perbandingan antara beberapa PoS Tagger kemudian dianalisa sehingga menghasilkan PoS Tagger baru. Setelah itu maka akan dilakukan perbandingan tipe metode PoS Tagger.

Pengumpulan korpus, pengumpulan ini mengggunakan korpus dari penelitian sebelumnya yaitu penelitian (Muniyati, 2019), (Magdalena, 2019), dan penambahan dari buku Sepok Tiga karya Pay Jarot Sujarwo. Pengumpulan korpus ini kemudian akan dilakukan perbaikan pada tanda baca, ejaan, atau kalimat-kalimat yang dinilai terlalu pendek atau panjang untuk pengujian. Setelah ini akan dilakukan tagging PoS pada korpus menggunakan tipe PoS yang telah dianalisa sebelumnya, setelah proses ini dilakukan akan dilakukan pengujian nilai akurasi metode PoS tagger, lalu di analisa untuk setelah dilakukan pengujian, kemudian akan ditarik kesimpulan dari hasil analisa yang telah dilakukan.

## Rumusan Masalah

Perumusan masalah dalam penelitian ini berdasarkan hal-hal yang telah disampaikan pada latar belakang adalah pengembangan tipe PoS yang baru dan membandingkan tipe metode Pos Tagger untuk mencari nilai akurasi yang terbaik dari tipe metode Pos Tagger.

## Tujuan Penelitian

Adapun tujuan dari penelitian yang dilakukan adalah untuk membandingkan tipe metode Pos Tagger untuk mencari nilai akurasi yang terbaik dari tipe metode Pos Tagger.

0

## Pembatasan Masalah

Pembatasan masalah pada penelitian ini, sebagai berikut :

1. Menggunakan Korpus dari penelitian sebelumnya oleh (Muniyati, 2019), (Magdalena, 2019), dan penambahan dari buku Sepok Tiga karya Pay Jarot Sujarwo.
2. Total Kalimat yang digunakan sebanyak 1500 kalimat.
3. Bahasa Pemrograman Python.
4. Menggunakan Tools NLTK.

## 1.5 Sistematika Penulisan

Adapun sistematika dalam penulisan tugas akhir ini tersusun atas lima bab, yang terdiri dari BAB I Pendahuluan, BAB II Tinjauan Pustaka, BAB III Metodologi Penelitian dan Perancangan Sistem, BAB IV Hasil Perancangan dan Analisis Sistem, serta BAB V Penutup.

**BAB I Pendahuluan** adalah bab pertama dalam penelitian, pada bab ini berisi latar belakang, rumusan masalah, tujuan penelitian, pembatasan masalah, sistematika penulisan.

**BAB II Tinjauan Pustaka** adalah bab yang berisi tentang tinjauan pustaka yang membahas mengenai pengertian dan informasi terhadap penelitian yang akan dilakukan.

**BAB III Metodologi Penelitian** adalah bab yang berisi tentang metodologi penelitian, alat bantu penelitian, serta metode penelitian.

**BAB IV Hasil dan Analisis** adalah bab yang berisi hasil dan pengujian yang telah dilakukan serta analisa hasil pengujian.

**BAB V Penutup** adalah bab yang berisi kesimpulan dari penelitian yang telah dilakukan dan saran atau rekomendasi untuk perbaikan, pengembangan atau kesempurnaan dan kelengkapan penelitian yang telah dilakukan.

# BAB II Tinjauan Pustaka

## 2.1 Penelitian Terkait

Beberapa penelitian terkait yang berhubungan dengan proses *tagging* PoS menggunakan metode PoS Tagger yang berhubungan dengan penelitian ini.

Evi Fahtiyah Muniyati dengan judul skripsi *“Prediksi Jeda Dalam Ucapan Kalimat Bahasa Melayu Pontianak Menggunakan Hidden Markov Model Berbasis Part Of Speech”* dengan jeda ucapan bigram nilai akurasi frasa jeda 1+2 = 30.8% dan akurasi jeda 2 = 80% dan dengan jeda ucapan trigram nilai akurasi jeda 1+2 = 30.8% dan akurasi jeda 2 = 81.4%.

Yulia Magdalena dengan judul skripsi *“Prediksi Jeda pada Ucapan Kalimat Bahasa Melayu Pontianak Menggunakan Metode Shallow Parsing dengan Pengembangan Rule Grammar dan Rule Jeda”* dengan akurasi kecocokan pada satu kalimat sebesar 33.6% dan nilai akurasi 70,2% jeda panjang atau sebanyak 364 kalimat dari 500 kalimat.

M. Iqbal Kamiludin dengan judul skripsi *“Prediksi Jeda pada Ucapan Bahasa Melayu Pontianak dengan Menggunakan Metode Shallow Parsing”* dengan nilai recall dan precision untuk kalimat tunggal sebesar 0.78 dan 0.74, sedangkan untuk kalimat majemuk sebesar 0.67 dan 0.57. Dari 168 kalimat yang ada, nilai kecocokan dengan jeda penutur sebesar 40.4% atau 68 kalimat.

Adhitya Teguh Nugraha, Teknik Informatika di Universitas Tanjungpura dalam skripsi yang berjudul ”*Prediksi Jeda dalam Ucapan Kalimat Bahasa Indonesia dengan Hidden Markov Model”* menjelaskan bahwa tujuan penelitian tersebut adalah memprediksi terjadinya jeda pada ucapan kalimat berbahasa Indonesia dengan menggunakan *Hidden Markov Model* (HMM). Penelitian ini menghasilkan aplikasi untuk memprediksi jeda dalam ucapan kalimat berbahasa Indonesia dan menggunakan Delphi untuk membuat antarmuka sistem. Korpus yang digunakan berasal dari berita yang disiarkan oleh Lembaga Penyiaran Publik TVRI Kalimantan Barat. Penelitian ini menggunakan 35 *tag* PoS seperti yang digunakan pada penelitian milik Wicaksono dan Purwarianti (2010). Hasil dari penelitian ini yaitu nilai *recall* sebesar 0.132, nilai *precision* sebesar 0.364 dan *f-score* sebesar 0.194.

Dari penelitian sebelumnya dapat dilihat dari nilai akurasi dan kecocokan nilai akurasi yang paling tinggi didapatkan dari penelitian oleh Evi Fahtiyah Muniyati(2019) dengan Tag PoS yang berjumlah 45, dalam hal ini dapat dikembangkan lagi dengan mengurangi jumlah PoS tag pada penelitian sebelumnya.

**Tabel 2 1** Kajian Penelitian Terkait

| No. | Penulis | Metode | Hasil |
| --- | --- | --- | --- |
| 1 | Evi Fahtiyah Muniyati, Prediksi Jeda Dalam Ucapan Kalimat Bahasa Melayu Pontianak Menggunakan Hidden Markov Model Berbasis Part Of Speech, 2019. | * Menggunakan *tools* HMM yaitu IPOSTagger. | * jeda ucapan bigram nilai akurasi frasa jeda 1+2 = 30.8% dan akurasi jeda 2 = 80% * jeda ucapan trigram nilai akurasi jeda 1+2 = 30.8% dan akurasi jeda 2 = 81.4% |
| 2 | Yulia Magdalena, Prediksi Jeda pada Ucapan Kalimat Bahasa Melayu Pontianak Menggunakan Metode Shallow Parsing dengan Pengembangan Rule Grammar dan Rule Jeda, 2019. | * Metode yang digunakan *Shallow Parsing* * Menggunakan pengembangan *Rule Grammar*  dan *Rule Jeda* | * akurasi kecocokan pada satu kalimat sebesar 33.6% dan nilai akurasi 70,2% jeda panjang atau sebanyak 364 kalimat dari 500 kalimat. |
| 3 | M. Iqbal Kamiludin, Prediksi Jeda pada Ucapan Bahasa Melayu Pontianak dengan Menggunakan Metode *Shallow Parsing*, 2017. | * Metode yang digunakan *Shallow Parsing*. * Menggunakan *rule grammar* untuk memprediksi jeda. | * Nilai *recall* sebesar 0.78, *precision* sebesar 0.74 dan *f-measure* sebesar 0.75 untuk kalimat tunggal. * Nilai *recall* sebesar 0.67, *precision* sebesar 0.57 dan *f-measure* sebesar 0.61 untuk kalimat majemuk. |
| 4 | Adhitya Teguh Nugraha, Prediksi Jeda dalam Ucapan Kalimat Bahasa Indonesia dengan *Hidden Markov Model*, 2014. | * Menggunakan *tools* HMM yaitu IPOSTagger. * GUI aplikasi menggunakan Delphi. | * Nilai *recall* sebesar 0.132, *precision* sebesar 0.364 dan *f-score* sebesar 0.194. |

Penelitian ini dilakukan untuk mengetahui nilai akurasi PoS Tag yang telah dikembangkan terhadap metode PoS Tagger pada Bahasa Melayu Pontianak.

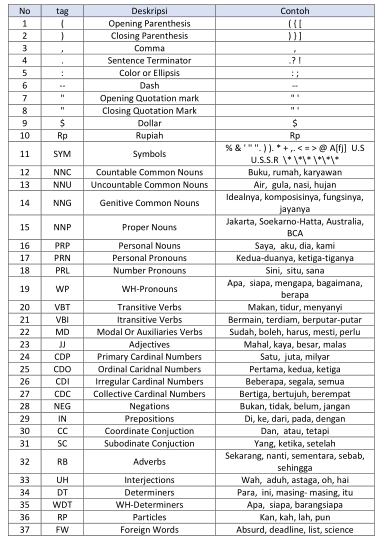
## 2.2 Part Of Speech Tagging

*Part of speech* (PoS) *tagging* adalah proses menentukan kelas kata untuk setiap kata dalam sebuah kalimat. *Part of speech* atau kelas kata terdiri dari kata kerja, kata sifat, kata benda, kata keterangan, dan lain-lain. Beberapa penggunaan PoS-*tagging* adalah untuk menghapus perbedaan yang tidak relevan, menghapus ambiguitas, membantu *stemming,* dan membantu pencarian kata benda (Christianti, Pragantha, & Victor, 2016).

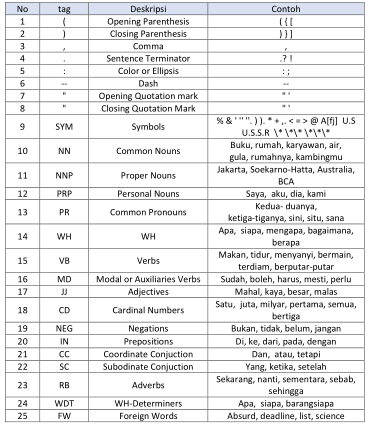
Hingga saat ini sudah ada beberapa *set* PoS berbahasa Indonesia yang dihasilkan dari penelitian sebelumnya, diantaranya yaitu tipe PoS Arawinda Dinakaramani(2014), tipe Pos Pisceldo(2009), tipe PoS Alfan Farizki Wicaksono(2010), tipe PoS Septina Dian Larasati(2011), tipe PoS Ayu Purwarianti(2010), tipe PoS Achmad Fatchuttaman Abka(2017), tipe PoS Andriani(2009), tipe PoS Setyaningsih(2017), tipe PoS Muliono(2012). Kesepuluh penelitian tersebut dijadikan sebagai acuan untuk pengembangan *set* PoS pada penelitian ini.



Gambar 2.1 *Set* PoS Arawinda Dinakaramani (2014)



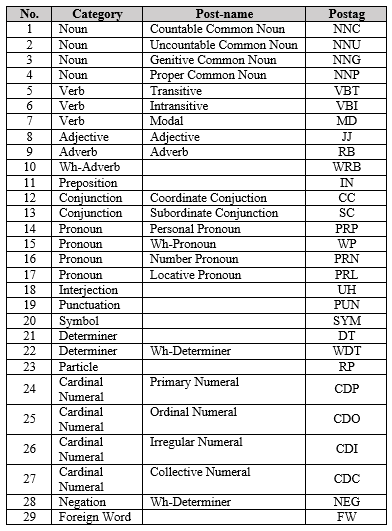
**Gambar 2. 2** Set PoS Pisceldo(2009) Tabel 1



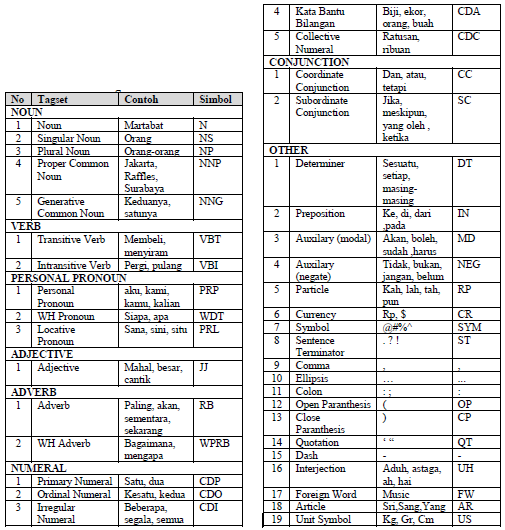
**Gambar 2. 3** Set PoS Pisceldo(2009) Tabel 2



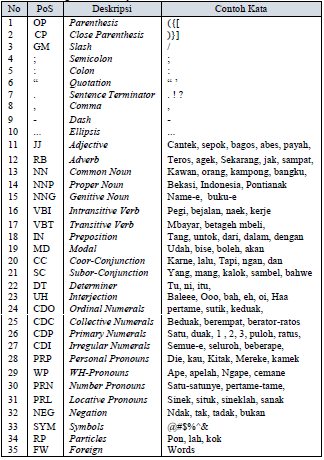
**Gambar 2. 4** Set PoS Alfan Farizki Wicaksono (2010)



**Gambar 2. 5** Set PoS Adriani (2009)



**Gambar 2. 6** Set PoS Setyaningsih (2017)



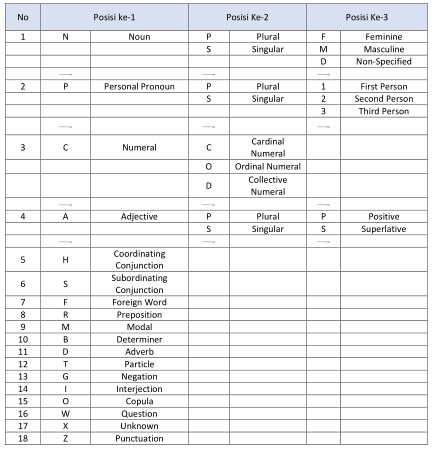
Gambar 2. 7 *Set* PoS Purwarianti (2010)



Gambar 2. 8 *Set* PoS Achmad Fatchuttaman Abka (2017)



**Gambar 2. 9** Set PoS Muliono (2012)



**Gambar 2. 10** Set PoS Septina Dian Larasati(2011)

Berdasarkan gambar *tagging* PoS tersebut maka penelitian ini akan membuat Tagging PoS Bahasa Melayu Pontianak yang baru berdasarkan referensi Tag PoS yang sudah dijelaskan untuk keperluan perhitungan nilai akurasi metode PoS Tagger.

## 2.3 Metode PoS Tagger

Metode PoS Tagger yang digunakan pada penelitian ini adalah, Bigram, HMM, Perceptron, Trigram, Unigram. Penjelasan mengenai metode tagger adalah sebagai berikut :

Bigram

Brigram tagger adalah pada dasarnya bekerja dengan cara yang sama seperti unigram, perbedaannya adalah pada bigram tagger mempertimbangkan konteks kapan menetapkan tag ke kata, sehingga jika dijelaskan lebih rinci bigram adalah urutan dua eleman yang berdekatan dari untaian token yang biasanya berupa huruf, suku kata, atau kata n = 2.

HMM (Hidden Markov Model)

HMM dikembangkan dari *Markov Chain* yaitu keadaan yang akan datang dari suatu *sequence* tidak ditentukan dari keadaan saat ini, tapi juga perpindahan dari suatu *state* ke *state* *sequence* yang lain.

Perceptron

Perceptron tagger mengimplementasikan penandaan PoS (*Part Of Speech*) menggunakan rata-rata, struktur algoritma perceptron.

Trigram

Trigram tagger adalah sebuah tagger yang menggunakan dua kata sebagai konteksnya untuk menentukan Tag PoS (*Part Of Speech*), kata n = 3.

Unigram

Unigram tagger adalah sebuah tagger yang hanya menggunakan satu kata sebagai konteksnya untuk menentukan Tag PoS (*Part Of Speech*), sederhananya tagger ini berbasis konteks yang konteksnya adalah satu kata, yaitu Unigram, kata n = 1.

## 2.4 Python

Python merupakan bahasa pemrograman dengan tujuan umum yang dikembangkan secara khusus untuk membuat source code mudah dibaca. Python juga memiliki library yang lengkap sehingga memungkinkan *programmer* untuk membuat aplikasi yang mutakhir dengan menggunakan *source code* yang tampak sederhana.

Beberapa keunggulan Python apabila dibandingkan dengan bahasa pemrograman lain adalah :

* Memiliki kepustakaan yang luas, dalam distribusi Python telah disediakan modul-modul `siap pakai’ untuk berbagai keperlua.
* Memiliki tata bahasa yang jernih dan mudah dipelajari.
* Memiliki aturan layout kode sumber yang memudahkan pengecekan, pembacaan kembalidan penulisan ulang kode sumber.
* Berorientasi obyek.
* Memiliki sistem pengelolaan memori otomatis (*garbage collection*, seperti java) Modullar, mudah dikembangkan dengan menciptakan modul-modul baru; modul-modul tersebut dapat dibangun dengan bahasa Python maupun C/C++.

Salah satu kegunaan Bahasa Pemrograman Python yaitu dapat mendukung aktivitas pemrosesan Bahasa alami, dengan menggukan modul dari *Natural Language Toolkit* yang merupakan salah satu koponen yang disediakan oleh Bahasa pemrograman python, khususnya dalam pengolahan digital Bahasa alami. Dalam penelitian ini python digunakan sebagai Bahasa pemrograman dalam proses menentukaan frasa jeda ucapan pada kalimat dengan menggunakan fungsi *shallow* parsing yang merupakan salah satu tugas dari *Natural Language Toolkit* (NLTK).

## 2.5 *Natural Language Toolkit*

Walaupun python memiliki kemampuan untuk melakukan tugas-tugas NLP (*Natural Language Processing*) secara dasar, namun tidak cukup efisien dalam melakukan tugas-tugas standar yang terdapat dalam NLP, maka dari itu digunakan modul NLTK (*Natural Language Toolkit)*. Modul NLTK menyediakan berbagai fungsi dan *wrapper,* serta *corpora* standar baik itu mentah atau *pre-processed* yang digunakan dalam materi pengajaran NLP. NLTK sendiri merupakan sebuah *platform* yang dirilis oleh Steven Bird dan Edward Loper pada tahun 2001 yang digunakan untuk membangun program analisis teks. NLTK menyediakan antarmuka yang mudah digunakan untuk lebih dari 50 sumber korpora dan leksikal dan seperangkat pustaka (*library*) pemrosesan teks untuk klasifikasi (*classification*), tokenisasi (*tokenization*), penumpukan (*stemming*), penandaan (*tagging*), penguraian (*parsing*), penalaran semantik (*semantic reasoning*), pembungkus (*wrapper*) pustaka NLP dan forum diskusi aktif (Lestari, 2017).

## 2.6 Bahasa Melayu Pontianak

Bahasa Melayu Pontianak merupakan satu di antara banyak bahasa yang terdapat di Provinsi Kalimantan Barat. Bahasa ini dituturkan oleh orang Melayu yang ada di kota Pontianak. Penggunaan bahasa Melayu Pontianak juga sering disingkat dari kata aslinya pada pembicaraan sehari-hari.

Contoh bahasa Melayu Pontianak yang dikutip dari buku sepok tige karangan Pay Jarot Sujarwo seperti berikut:

“…*Karne lukisan – lukisan Bang Jul ni hamper rate nyritekan tentang Kalbar, secare tak langsong Bang Jul ni ikut jadi duta wisata Kalbar. Maklom, Dinas Pariwisata di tempat kite ni tadak bise bekerje maksimal, jadi kalok ade orang – orang barat datang, bukan-e pegi ke pusat inpormasi turis, tapi pegi ke tempat – tempat macam Bang Jul ni.”*

## 2.7 Pengujian Teks kalimat metode PoS Tagger

Pengujian teks kalimat yang lakukan antara kalimat *tagging* manual yaitu korpus yang telah dikembangkan kemudian dilakukan *tagging* terlebih dahulu dengan cara manual yaitu sebagai contoh :

“Semue-mue-e tepat waktu.”

Kalimat diatas adalah contoh kalimat dari korpus yang telah dikembangakn sehingga berjumlah 1500 kalimat, kemudian dilakukan tagging dengan menambahkan “/” yaitu garis miring dan label kelas kata yaitu tag PoS, sehingga menjadi “Semue-mue-e/PRP tepat/RB waktu/NN ./.” hal ini perlu dilakukan agar mesin mengetahui apa jenis kata yang digunakan atau kelas kata yang terdapat pada kalimat tersebut, hal ini dilakukan sebanyak 1500 kalimat semuanya dilakukan *tagging* secara manual, kemudian akan dilakukan perbandingan antara *tagging* manual tersebut dengan *tagging* metode PoS Tagger.

## 2.8 Pengujian *Accuracy, Precision, Recall,* dan *F-measure*

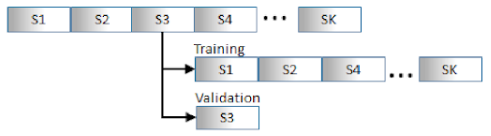
*Precision* dan *recall* menjadi ukuran yang paling sering digunakan untuk mengevaluasi suatu sistem. *Precision* adalah rasio jumlah dokumen relevan yang ditemukan dengan total jumlah dokumen yang ditemukan oleh sistem (Rozi, Pramono, & Dahlan, 2012) Nilai *precision* tertinggi adalah 1, yang berarti seluruh dokumen yang ditemukan adalah relevan (Fitri, 2013). *Precision* adalah tingkat ketepatan antara informasi yang diminta oleh pengguna degan jawaban yang diberikan oleh sistem. *Recall* adalah rasio jumlah dokumen tidak relevan yang ditemukembalikan terhadap jumlah seluruh dokumen relevan dalam koleksi (Nugraha, 2014). Nilai *recall* tertinggi adalah 1, yang berarti seluruh dokumen dalam koleksi berhasil ditemukan. (Rozi, Pramono, & Dahlan, 2012) menjelaskan bahwa, precision adalah rasio jumlah dokumen relevan yang ditemukan dengan total jumlah dokumen yang ditemukan oleh sistem. Recall adalah tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi. Precision dapat dianggap sebagai ukuran ketepatan atau ketelitian, sedangkan recall adalah kesempurnaan. *Accuracy* dapat didefinisikan sebagai tingkat kedekatan antara nilai nilai prediksi dengan nilai aktual.

*F-measure* (nilai F atau F1-Score ) merupakan sebuah nilai dari keakuratan sebuah tes. Nilai F menggunakan precision & recall dari tes untuk menghitung nilainya, dengan precision yang menyatakan jumlah hasil benar dibandingkan dengan jumlah hasil yang ditemubalikkan dan recall yang menyatakan jumlah hasil benar dibandingkan dengan jumlah hasil yang harus ditemubalikkan.

Pengujian *accuracy, precision, recall*, dan *f-measure* digunakan untuk mengetahui nilai akurasi keberhasilan PoS Tag terhadap beberapa metode PoS Tagger.

## 2.9 Pengujian K-*fold Cross Validation*

K-*fold* merupakan salah satu metode cross validation yang popular dengan menghitung data sebanyak K dan juga menjalankan iterasi sebanyak K juga. Membuat set data K training-validation dengan menentukan salah satu golongan sebagai validation set dan mengabungkan seluruh kelompok yang tersisa ke dalam training set seperti yang digambarkan pada gambar 2.11.



**Gambar 2. 11** Ilustrasi Pembuatan K-fold Cross Validation set

# BAB III Metodologi Penelitian

* 1. **Bahan Penelitian**

Bahan penelitian berupa Korpus dari penelitian sebelumnya oleh (Muniyati, 2019), (Magdalena, 2019), dan penambahan dari buku Sepok Tiga karya Pay Jarot Sujarwo selanjutnya diolah menjadi korpus teks bahasa Melayu Pontianak.

* 1. **Alat Bantu Penelitian**

Perangkat penelitian yang digunakan dalam penelitian ini berupa perangkat lunak dan perangkat keras. Adapun penjabaran perangkat penelitian yang digunakan adalah sebagai berikut :

* + 1. **Perangkat Keras**Perangkat keras yang digunakan dalam penelitian ini adalah :

1. Dell Inspiron 14 3442 dengan spesifikasi *Intel Celeron* 2957U, SSD 128GB, Kapasitas RAM 4GB, LCD Monitor 14 inch.
   * 1. **Perangkat Lunak (*Software*)**

Perangkat lunak yang digunakan dalam penelitian ini adalah :

1. Sistem Operasi Windows 10
2. Python dan Phycharm Community
3. NLTK (*Natural Language Toolkit*)
4. IPOSTagger sebagai *tools* yang digunakan untuk metode HMM
5. Sublime Text 3 sebagai aplikasi untuk teks *editor.*
   1. **Metode Penelitian**

Metode Penelitian yang akan dilakukan dijelaskan pada diagram alir penelitian pada penelitian gambar 3.1 berikut :



**Gambar 3. 1** Diagram Alir Penelitian

Seperti diagram alir penelitian yang terlihat pada gambar 3.1, terdapat beberapa tahapan yang dilakukan pada penelitian, diantaranya pengumpulan tag pos, membandingkan tag pos, pengembangan pos, pembuatan korpus tagging pos, pelatihan korpus, ujicoba korpus, pengujian, analisa hasil pengujian, dan kesimpulan.

* + 1. **Persiapan Data**

Persiapan data berupa teks kalimat bahasa Melayu Pontianak yang bersumber dari buku Sepok 3 karangan Pay Jarrot Sujarwo dan korpus dari penelitian sebelumnya yang berjumlah 500 kalimat. Jumlah kalimat yang digunakan pada penelitian sebanyak 1500 kalimat. Teks kalimat digunakan untuk penandaan PoS atau label kata.

* + 1. **Pengumpulan Tag PoS**

Pengumpulan tag PoS dari penelitian sebelumnya, penelitian proses pengumpulan tipe PoS ini terdari dari 10 penelitian sebelumnya, yaitu : tipe PoS Arawinda Dinakaramani(2014), tipe Pos Pisceldo(2009), tipe PoS Alfan Farizki Wicaksono(2010), tipe PoS Septina Dian Larasati(2011), tipe PoS Ayu Purwarianti(2010), tipe PoS Achmad Fatchuttaman Abka(2017), tipe PoS Andriani(2009), tipe PoS Setyaningsih(2017), tipe PoS Muliono(2012).

**3.3.3 Pengembangan Tag PoS**

Pada tahap ini digunakan set PoS yang baru dibentuk berdasarkan Set PoS Bahasa Melayu Pontianak. Pengelompokkan kelas kata dalam Bahasa Melayu dilakukan secara manual dengan merujuk pada referensi tipe PoS Arawinda Dinakaramani(2014), tipe Pos Pisceldo(2009), tipe PoS Alfan Farizki Wicaksono(2010), tipe PoS Septina Dian Larasati(2011), tipe PoS Ayu Purwarianti(2010), tipe PoS Achmad Fatchuttaman Abka(2017), tipe PoS Andriani(2009), tipe PoS Setyaningsih(2017), tipe PoS Muliono(2012). Set PoS baru dibuat dengan tujuan untuk mereduksi jumlah tag PoS, sehingga untuk mendapatkan tag PoS yang baru maka dilakukan pengembangan terhadap ke-10 penelitian sebelumnya dan dilakukan pencarian dan perbandingan terhadap fungsi atau arti yang sama, sehingga dapat menghasilkan tag PoS yang baru. Penambahan set PoS disesuaikan dengan fungsi kata dalam kalimat bahasa Melayu Pontianak. Hasil dari pembuatan tabel PoS adalah tabel set PoS baru yang digunakan untuk menandai kata korpus teks bahasa Melayu Pontianak.

* + 1. **Pengembangan Korpus**

Peningkatan jumlah korpus dari penelitian sebelumnya yang hanya berjumlah 500 kalimat kemudian ditambah dengan kalimat baru dari buku sepok tiga sehingga mendapatkan korpus yang dapat mewakili keseluruhan dari tag PoS yang sudah dibuat , sehingga korpus menjadi lebih maksimal untuk pengujian.

* + 1. ***Tagging* Korpus**

Teks kalimat untuk korpus latih yang berasal dari penelitian sebelumnya diberikan tag PoS sesuai dengan kelas katanya secara manual sesuai set PoS yang telah dikembangkan. Penandaan PoS berdasarkan makna dari kata tersebut ketika diucapkan dan dari Kamus Besar Bahasa Indonesia (KBBI). Korpus latih tersebut berisikan “kata/PoS”.

Misalkan pada korpus terdapat kalimat :

“Eh , jangan banyak umong kau .”

Setiap kata akan dilihat dan dikelompokkan berdasarkan label PoS. Misalkan kata “Eh” menunjukkan panggilan pada kalimat dengan kata seru sehingga ditandai dengan PoS “RP” untuk kata seru. Kata “jangan” menunjukkan kata negatif sehingga ditandai dengan PoS “NEG” untuk kata negatif. Kata “banyak” menunjukkan kata sifat sehingga ditandai dengan PoS “JJ” untuk kata sifat. Kata “ngumong” menunjukkan kegiatan berbicara sehingga ditandai dengan PoS “VB” untuk kata kerja intrasitif yang memerlukan objek dibelakang kata kerja. Kata “kau”

merupakan objek yang berupa kata ganti orang sehingga ditandai dengan PoS “PRP” yang dikhususkan untuk kata ganti orang seperi “kau”, “aku” , “kamek”. Hasil penandaan PoS menjadi kalimat yang berisi “kata/PoS” seperti berikut:

“ Eh/RP ,/, jangan/NEG banyak/JJ umong/VB kau/PRP ./. ”

* + 1. **Pelatihan Korpus**

Pelatihan korpus dilakukan menjadi dua bagian yaitu dengan menggunakan aplikasi IPOSTagger untuk metode *HMM*, dan menggunakan aplikasi *PyCharm Community* untuk pelatihan metode *Bigram*, *Trigram*, *Perceptron*, dan *Unigram*.

##### 3.3.6.1 Pelatihan Korpus di IPOSTagger

Setelah korpus latih selesai dibuat, selanjutnya dilakukan proses pelatihan korpus pada IPOSTagger. Pada proses pelatihan korpus untuk tagging PoS, korpus latih yang berisi “kata/PoS” disimpan dalam folder IPOSTagger dengan ekstensi \*.crp. Kemudian proses pelatihan dilakukan melalui command prompt pada folder IPOSTagger.

##### 3.3.6.2 Pelatihan Korpus Bigram, Trigram, Perceptron, Unigram

Setelah dilakukan pelatihan korpus pada IPOSTagger yaitu pelatihan metode *HMM*, maka dilakukan pelatihan pada metode *PoS Tagger* lainnya. Pada pelatihan korpus untuk tagging PoS, korpus latih yang berisi “kata/PoS” disimpan pada folder masing-masing metode dengan nama file uji500.txt. kemudian proses pelatihan dilakukan melalui aplikasi *phycharm community*.

* + 1. **Perancangan Pengujian**

Pengujian dilakukan untuk mendapatkan nilai pengujian yang akan dianalisis menjadi kesimpulan penelitian. Pengujian dilakukan dengan membandingkan kalimat uji dan kalimat latih kemudian dilakukan pengujian *Accuracy, Precision, Recal,* dan *F-Score*.

##### 3.3.7.1 Skenario Pengujian *Accuracy*, *Recall*, *Precision*, dan *F-Measure*

Pada pengujian ini, dilakukan perhitungan dari setiap korpus uji dari setiap metode yang telah dilatih, dan kemudian pengujian dijalan degan menjalan aplikasi phycharm community dengan kode program yang telah dibuat dan dijalankan untuk mendapatkan hasil dari nilai setiap pengujian. Kode Program untuk setiap pengujian adalah sebagai berikut :

**Kode Program 3. 1** Pengujian Metode PoS Tagger

corp\_manualasli = **"saya/PRP itu/DT belajar/VBT pemrograman/NN ./."**corp\_otomatisasli = **"saya/NN itu/SC belajar/VBT pemrograman/NN ./."  
  
import** nltk  
cm = nltk.ConfusionMatrix(corp\_manual.split(), corp\_otomatis.split())  
**from** nltk.metrics **import** \*  
  
a = accuracy(corp\_manual.split(), corp\_otomatis.split())  
reference\_set = set(corp\_manual.split())  
test\_set = set(corp\_otomatis.split())  
  
p = precision(reference\_set, test\_set)  
  
r = recall(reference\_set, test\_set)  
  
f = f\_measure(reference\_set, test\_set)  
  
print(**"confusion\_matrix"**)  
print(cm)  
print(**"\naccuracy\_score"**)  
print(a)  
print(**"\nprecision\_score"**)  
print(p)  
print(**"\nrecall\_score"**)  
print(r)  
print(**"\nf1\_score"**)  
print(f)

Setelah dilakukan pengujian menggunakan kode program pada gambar 3.2 hasil yang di hasilkan akan di tulis pada tabel 3.1, hasil ini lah yang menjadi hasil dari pengujian metode PoS Tagger ini.

**Tabel 3. 1** Rencana Tabel Hasil Nilai Pengujian Accuracy, Precision, Recall, dan F-Measure

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Metode PoS Tagger | Accuracy | Precision | Recall | F-measure |
| Bigram |  |  |  |  |
| HMM |  |  |  |  |
| Perceptron |  |  |  |  |
| Trigram |  |  |  |  |
| Unigram |  |  |  |  |

##### 3.3.7.2 Skenario Pengujian *K-fold Cross Validation*

Pengujian *K-fold cross validation* dilakukan untuk menguji secara keseluruhan dari total kalimat yang digunakan yaitu sebanyak 1500 kalimat. Pengujian terdiri dari lima *fold* dengan pembagian 20% sebagai corpus uji dan 80% sebagai corpus latih. Skenario pengujian *K-fold cross validation* dapat dilihat lebih jelas pada tabel 3.2.

**Tabel 3. 2** Tabel Pengujian 5 Fold Cross Validation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Korpus | Accuracy | Precision | Recall | F-Measure |
| 1-300 |  |  |  |  |
| 301-600 |  |  |  |  |
| 601-900 |  |  |  |  |
| 901-1200 |  |  |  |  |
| 1200-1500 |  |  |  |  |
| Rata-rata |  |  |  |  |

Pada skenario tabel pengujian yaitu tabel 3.2 akan dimasukkan pada lampiran A, B, C, D, dan E. Hasil perhitungan rata-rata dari nilai *accuracy, precision, recall,* dan *F-measure* dari kelima fold yang didapatkan kemudian dimasukkan ke dalam tabel seperti pada tabel 3.3.

**Tabel 3. 3** Rencana Tabel Hasil Perhitungan Accuracy, Precision, Recall, dan F-measure Pengujian K-Fold Cross Validation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Metode PoS Tagger | Accuracy | Precision | Recall | F-measure |
| Bigram |  |  |  |  |
| HMM |  |  |  |  |
| Perceptron |  |  |  |  |
| Trigram |  |  |  |  |
| Unigram |  |  |  |  |

* + 1. **Analisis Hasil Pengujian**

Pada tahap ini, hasil pengujian sistem dianalisis dan dipresentasikan ke dalam grafik untuk mempermudah penarikan kesimpulan.

* + 1. **Kesimpulan Hasil Pengujian**

Kesimpulan dibuat berdasarkan tahapan-tahapan yang telah dilakukan sebelumnya dengan melihat nilai rata-rata pengujian dari setiap metode PoS Tagger.

# BAB IV Hasil dan pengujian

## Hasil Penelitian

Hasil pengembangan Tag PoS bahasa Melayu Pontianak yaitu tabel PoS dengan jumlah sebanyak 22 set PoS. Set PoS bahasa Melayu Pontianak dapat dilihat pada tabel 4.1.

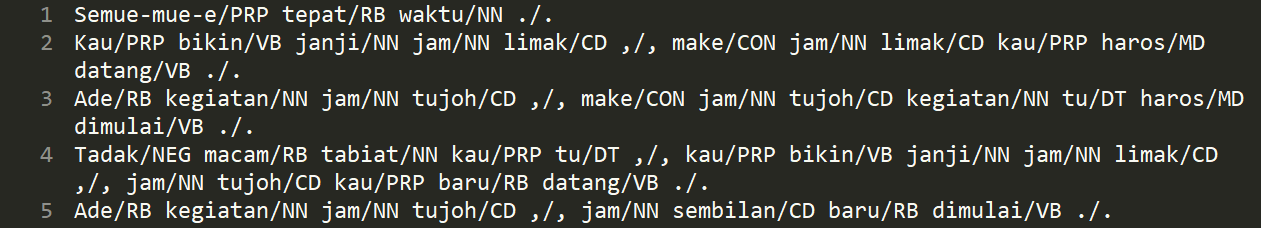
**Tabel 4. 1** *Set* PoS Bahasa Melayu Pontianak

|  |  |  |  |
| --- | --- | --- | --- |
| No | tag | Deskripsi | Contoh |
| 1 | NN | nouns | kata benda |
| 2 | NNP | proper nouns | nama, nnama benda atau nama orang, |
| 3 | PRP | Personal Nouns | Saya, aku, dia, kami, orang |
| 4 | VB | Verbs | Makan, tidur, menyanyi, bermain, terdiam, berputar-putar |
| 5 | JJ | adjective | kata sifat |
| 6 | RB | adverb | kata keterangan |
| 7 | CD | Cardinal Number | ratusan, ribuan, number, satu, dua, kesatu, ketiga, angka bilangan Rp, $, Kedua- duanya, ketiga-tiganya, sini, situ, sana |
| 8 | CDA | Kata Bantu Bilangan | Biji, Ekor, Orang, Buah orang, ton, helai, lembar |
| 9 | CDI | Irregular Numeral | Beberapa, Segala, Semua, BANYAK |
| 10 | CON | Conjunction | Dan, Atau, Tetapi, sejak, jika, seandainya, supaya, meski, seolah-olah, sebab, maka, tanpa, dengan, bahwa, yang, lebih ... daripada ..., semoga |
| 11 | DT | Determiners | Para, ini, masing- masing, itu, sang Sri, Sang, Yang, ini, itu, sini, situ |
| 12 | IN | Preposition | ke, di, dari, pada, tentang |
| 13 | MD | Auxiliary(modal) | akan, boleh, sudah, harus, mesti, Bisa |
| 14 | RP | Particle | Kah, lah, tah, pun, Aduh, astaga, ah, hai, oi |
| 15 | WH | WH-Determiners | Apa, siapa, barangsiapa,Apa, siapa, mengapa, bagaimana, berapa |
| 16 | NEG | Negations | Bukan, tidak, belum, jangan |
| 17 | FW | Foreign Word | Music, tidak ketahui |
| 18 | SYM | Symbol | @#%^ , …, : ; , ( ) {}[]\ <> - + = ` ~ Kg, Gr, Cm , ..., ?, . |
| 19 | . | TITIK | . |
| 20 | , | KOMA | , |
| 21 | ? | Tanda Tanya | ? |
| 22 | ! | Tanda seru | ! |

Set Pos ini didapatkan dari hasil perbandingan yang telah dilakukan dengan membandingkan 10 tipe pos dari penelitian lainnya sehingga menghasilkan set pos pada tabel 4.1 yang berjumlah 22.

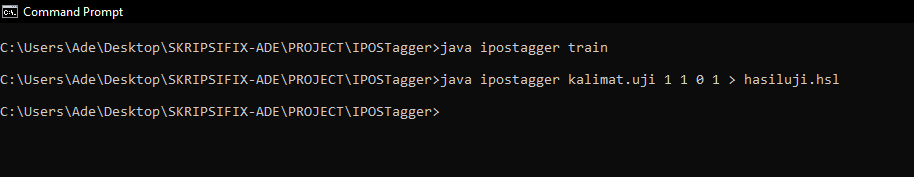
### Pembuatan Korpus untuk *Tagging* PoS

Korpus Sebanyak 1500 teks kalimat yang digunakan sebagai korpus latih yang berasal dari buku Sepok Satu ditandai kelas katanya secara manual. Korpus latih untuk *tagging* PoS yang berisi “kata/PoS” dapat dilihat pada gambar 4.1.

  
**Gambar 4. 1** Korpus Latih Proses Tagging PoS

### Hasil *Tagging* PoS Metode HMM untuk Bahasa Melayu Pontianak

Selanjutnya, teks kalimat bahasa Melayu Pontianak menggunakan set PoS baru yang sudah dikembangkan. Penandaan atau *tagging* PoS bertujuan untuk mencari nilai akurasi dari metode hmm. Korpus yang telah diberi label PoS dapat dilihat pada gambar 4.1



**Gambar 4. 2** Proses Pelatihan IPOSTagger

Tiap kata dalam kalimat teks bahasa Melayu Pontianak diberi tanda”/” atau garis miring dan diikuti label PoS, penandaan dilakukan secara manual yang telah dilakukan seperti pada gambar 4.1 kemudian disimpan pada subfolder “*resource*” pada folder IPOSTagger dengan extensi .crp. Kemudian proses pelatihan dilakukan melalui *command prompt* pada folder IPOSTagger seperti pada gambar 4.2.

Proses pelatihan korpus pada IPOSTagger menggunakan perintah “java ipostagger train” pada command prompt untuk melatih IPOSTagger sehingga bisa digunakan men-*tagging* kalimat bahasa Melayu Pontianak.

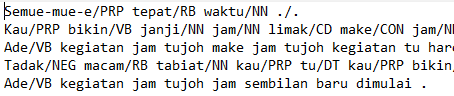
### Hasil *Tagging* PoS Metode Brigram untuk Bahasa Melayu Pontianak

Teks kalimat bahasa Melayu Pontianak kemudian dilakukan pengujian terhadap metode lainnya yaitu metode *bigram*, kalimat yang digunakan untuk latih dan pengujian sama seperti pada pengujian metode hmm, namun yang berbeda pada pengujian ini adalah proses melakukan pengujiannya yang sebelumnya menggunakan command prompt, pada kali ini menggunakan aplikasi *phycharm community*, kode program untuk menjalan proses ini, adalah sebagai berikut :

**Kode Program 4. 1** Melatih Data Uji Metode Bigram

**import** nltk  
**import** bigram  
**import** time  
startTime = time.time()  
  
*#kalimat uji*kalimatuji=[]  
f1 = open(**"uji500.txt"**, **"r"**)  
**for** baris **in** f1 :  
 kaluji = baris.strip()  
 kalimatuji.append(kaluji.split())  
 *#print(kalimatuji, len(kalimatuji))*train\_data=[]  
f0 = open(**"manual500.txt"**, **"r"**) *#corpustrain.txt***for** baris **in** f0 :  
 kal = baris.strip()  
 kalimat =[nltk.tag.str2tuple(t) **for** t **in** kal.split()]  
 *#print(kalimat)* train\_data.append(kalimat)  
  
*#print(train\_data)*databigram=[]  
bigram\_tagger = nltk.BigramTagger(train\_data)  
**for** jumkal **in** range(len(kalimatuji)):  
 *#print("Unigram =", unigram\_tagger.tag(kalimatuji[jumkal]))* databigram.append([nltk.tag.tuple2str(x) **for** x **in** bigram\_tagger.tag(kalimatuji[jumkal])])  
  
  
x=0  
**for** x **in** range(0, len(databigram)):  
 y=0  
 txtOut = **""  
 for** y **in** range(0, len(databigram[x])):  
 txtOut=txtOut + databigram[x][y]+ **' '** print(txtOut [:-1])  
**"""oleh clik jawaban dulu lalu proses durasi --> stop time"""**print(**f"waktu eksekusi = {**time.time()-startTime**}"**)

Setelah menjalan kode program ini pada aplikasi *phycharm community* maka kita akan mendapat hasil yang seperti seperti pada gambar 4.1 namun isinya akan berbeda karena setiap metode memberikan hasil yang berbeda dan hasilnya adalah sebagai berikut :



**Gambar 4. 3** Hasil Tagging Metode Bigram

Seperti yang dapat dilihat terdapat beberapa perbedaan antara gambar 4.1 dan gambar 4.3 ini, data ini lah yang akan digunakan sebagai bahan untuk pengujian nilai akurasi dari setiap metode PoS Tagger.

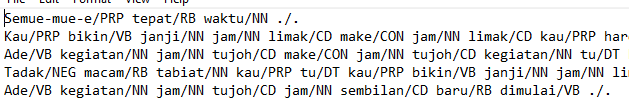
### Hasil *Tagging* PoS Metode Perceptron untuk Bahasa Melayu Pontianak

Metode *Perceptron* untuk menjalan proses pengujian ini sama halnya dengan metode bigram, tetapi tetap hasil keluaran dari metode ini tentunya akan berbeda juga dengan *hmm*, dan *bigram*. Dibawah ini adalah kode program untuk proses *tagging* metode *perceptron*.

**Kode Program 4. 2** Melatih Data Uji Metode *Perceptron*

**import** perceptron  
**from** nltk.tag.perceptron **import** PerceptronTagger  
**import** time  
startTime = time.time()  
  
*#kalimat uji*kalimatuji=[]  
f1 = open(**"uji500.txt"**, **"r"**)  
**for** baris **in** f1 :  
 kaluji = baris.strip()  
 kalimatuji.append(kaluji.split())  
 *#print(kalimatuji, len(kalimatuji))*train\_data=[]  
f0 = open(**"manual500.txt"**, **"r"**) *#corpustrain.txt***for** baris **in** f0 :  
 kal = baris.strip()  
 kalimat =[nltk.tag.str2tuple(t) **for** t **in** kal.split()]  
 *#print(kalimat)* train\_data.append(kalimat)  
  
  
*#print(train\_data)*dataperception=[]  
tagger = PerceptronTagger(load=**True**)  
tagger.train(train\_data)  
**for** jumkal **in** range(len(kalimatuji)):  
 *#print("perception =", tagger.tag(kalimatuji[jumkal]))  
 # dataperception.append(tagger.tag(kalimatuji[jumkal]))* dataperception.append([nltk.tag.tuple2str(x) **for** x **in** tagger.tag(kalimatuji[jumkal])])  
  
*#cetak data*x=0  
**for** x **in** range(0, len(dataperception)):  
 y=0  
 txtOut = **""  
 for** y **in** range(0, len(dataperception[x])):  
 txtOut=txtOut + dataperception[x][y]+ **' '** print(txtOut [:-1])  
**"""oleh clik jawaban dulu lalu proses durasi --> stop time"""**print(**f"waktu eksekusi = {**time.time()-startTime**}"**)

Setelah dilakukan kode program seperti diatas maka hasilnya akan seperti pada gambar 4.4 yang dimana hasil *tagging* dari metode perceptron ini tentunya akan berbeda dengan hasil dari metode *hmm, bigram*.



**Gambar 4. 4** Hasil Tagging Metode Perceptron

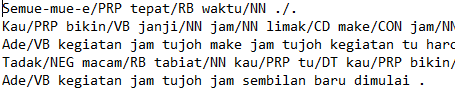
### Hasil *Tagging* PoS Metode Trigram untuk Bahasa Melayu Pontianak

Pada metode trigram selanjutnya juga sama lakukan pengetikan kode program pada aplikasi *phycharm community* kemudian jalankan dan akan menghasilkan hasil yang dapat dilihat pada gambar 4.5.

**Kode Program 4. 3** Melatih Data Uji Metode Trigram

**import** nltk  
**import** trigram  
**import** time  
startTime = time.time()  
  
*#kalimat uji*kalimatuji=[]  
f1 = open(**"uji500.txt"**, **"r"**)  
**for** baris **in** f1 :  
 kaluji = baris.strip()  
 kalimatuji.append(kaluji.split())  
 *#print(kalimatuji, len(kalimatuji))*train\_data=[]  
f0 = open(**"manual500.txt"**, **"r"**) *#corpustrain.txt***for** baris **in** f0 :  
 kal = baris.strip()  
 kalimat =[nltk.tag.str2tuple(t) **for** t **in** kal.split()]  
 *#print(kalimat)* train\_data.append(kalimat)  
  
*#print(train\_data)*datatrigram=[]  
trigram\_tagger = nltk.TrigramTagger(train\_data)  
**for** jumkal **in** range(len(kalimatuji)):  
 *#print("Unigram =", unigram\_tagger.tag(kalimatuji[jumkal]))* datatrigram.append([nltk.tag.tuple2str(x) **for** x **in** trigram\_tagger.tag(kalimatuji[jumkal])])  
  
  
x=0  
**for** x **in** range(0, len(datatrigram)):  
 y=0  
 txtOut = **""  
 for** y **in** range(0, len(datatrigram[x])):  
 txtOut=txtOut + datatrigram[x][y]+ **' '** print(txtOut [:-1])  
  
  
*#for kalimat in datatrigram:  
# print(kalimat)  
#print(datatrigram[0])***"""oleh clik jawaban dulu lalu proses durasi --> stop time"""**print(**f"waktu eksekusi = {**time.time()-startTime**}"**)

Hasil dari metode trigram ini tentunya dapat kita lihat pada gambar 4.5 yang dimana pada kalimat “Ade/VB kegiatan jam tujoh jam Sembilan baru dimulai .” pada teks kalimat ini dengan metode sebelumnya terdapat tagging namun pada kalimat ini tidak, dikarenakan setiap metode termasuk trigram memberikan nilai tagging yang manyatakan bahwa kalimat ini tidak ada nilai tagging, walaupun berdasarkan data latih yang sama dengan metode sebelumnya.



**Gambar 4. 5** Hasil Tagging Metode Trigram

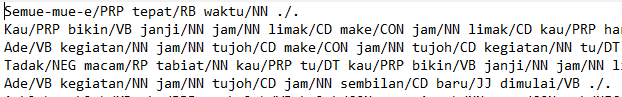
### Hasil *Tagging* PoS Metode Unigram untuk Bahasa Melayu Pontianak

Metode terakhir yang digunakan pada penelitian ini adalah metode unigram. Terdapat perbedaan dari kode program dan hasil yang dihasilkan dari metode ini, dibawah ini adalah kode program untuk proses tagging metode unigram.

**Kode Program 4. 4** Melatih Data Uji Metode Unigram

**import** nltk  
**import** unigram  
**import** time  
startTime = time.time()  
  
*#kalimat uji*kalimatuji=[]  
f1 = open(**"uji500.txt"**, **"r"**)  
**for** baris **in** f1 :  
 kaluji = baris.strip()  
 kalimatuji.append(kaluji.split())  
 *#print(kalimatuji, len(kalimatuji))*train\_data=[]  
f0 = open(**"manual500.txt"**, **"r"**) *#corpustrain.txt***for** baris **in** f0 :  
 kal = baris.strip()  
 kalimat =[nltk.tag.str2tuple(t) **for** t **in** kal.split()]  
 *#print(kalimat)* train\_data.append(kalimat)  
  
  
*#print(train\_data)*dataunigram=[]  
unigram\_tagger = nltk.UnigramTagger(train\_data)  
**for** jumkal **in** range(len(kalimatuji)):  
 *#print("Unigram =", unigram\_tagger.tag(kalimatuji[jumkal]))* dataunigram.append([nltk.tag.tuple2str(x) **for** x **in** unigram\_tagger.tag(kalimatuji[jumkal])])  
  
x=0  
**for** x **in** range(0, len(dataunigram)):  
 y=0  
 txtOut = **""  
 for** y **in** range(0, len(dataunigram[x])):  
 txtOut=txtOut + dataunigram[x][y]+ **' '** print(txtOut [:-1])  
  
  
*#for kalimat in dataunigram:  
# print(kalimat)  
#print(dataunigram[0])***"""oleh clik jawaban dulu lalu proses durasi --> stop time"""**print(**f"waktu eksekusi = {**time.time()-startTime**}"**)

Setelah dilakukan proses tagging maka hasil yang didapatkan ada pada gambar 4.6 pada gambar tersebut memiliki perbedaan kelas kata antara satu dan lainnya, pada metode trigram pada kalimat “ade kegiatan jam tujoh jam Sembilan baru dimulai.” hanya terdapat beberapa kata saja yang terdapat tagging, namun pada metode unigram ini terdapat tagging bagi setiap kata dari kalimat yang contoh tersebut.



**Gambar 4. 6** Hasil Tagging Metode Unigram

* 1. **Hasil Pengujian**

Total kalimat yang digunakan pada pengujian yaitu sebesar 1500 kalimat. Pengujian dilakukan dengan pengujian *accuracy, precision, recall,* dan F*­*-*measure* dan pengujian K-*fold cross validation.*

### Pengujian Accuracy, Precision, Recall dan F-measure

#### 4.2.1.1 Hasil Pengujian Accuracy, Precision, Recall dan F-measure

Pengujian *accuracy, precision, recall,* dan *F-measure* dilakukan dengan menggunakan dua buah korpus yang telah dilatih sebelumnya. Setiap metode mempunyai korpus latih yang sama dan korpus uji yang sama tetapi memiliki hasil berbeda, karena setiap metode melakukan hasil *tagging* pada korpus uji dengan mendapatkan data dari korpus latih.

Setelah mendapatkan data tersebut dilakukan pengujian dengan cara menggunakan kode program pada gambar 3.1 dan menggunakan tabel 3.1 untuk melakukan pendataan dari setiap nilai akurasi yang dihasilkan dari setiap metode.

**Tabel 4. 2** Tabel Hasil Pengujian *Accuracy, Precision, Recall* dan F-*measure*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Metode PoS Tagger** | **Accuracy Score** | **Precision Score** | **Recall Score** | **F-Measure Score** |
| Bigram | 0.950 | 0.921 | 0.962 | 0.941 |
| HMM | 0.975 | 0.984 | 0.973 | 0.978 |
| **Metode PoS Tagger** | **Accuracy Score** | **Precision Score** | **Recall Score** | **F-Measure Score** |
| Perceptron | 0.982 | 0.963 | 0.973 | 0.968 |
| Trigram | 0.964 | 0.936 | 0.980 | 0.957 |
| Unigram | 0.981 | 0.997 | 0.965 | 0.981 |

Dari tabel 4.2 dapat dilihat dari tabel diatas untuk nilai *accuracy score* yang tertinggi adalah Perceptron, pada metode ini memiliki nilai yang lebih rendah pada pengujian *precision, recall,* dan *f-measure*, sedangkan untuk nilai *precision* tertinggi yaitu dari metode Unigram, metode ini juga memiliki nilai *accuracy* yang hanya berbeda 0,001 dari perceptron, sedangkan untuk nilai *recall* yang tertinggi adalah metode trigram yang memiliki nilai sebesar 0,980, dan pada pengujian terakhir yaitu *f-measure* adalah metode unigram yaitu sebesar 0,981.

#### 4.2.1.2 Analisis Hasil Pengujian *Accuracy, Precision, Recall* dan F*-measure*

Pada grafik gambar 4.1 hasil pengujian nilai *accuracy* dengan perhitungan kalimat 500 kalimat uji dan 500 kalimat latih yang memberikan nilai yang cukup tinggi dengan nilai dari setiap metode PoS Tagger berjumlah diatas 0.9 dengan pengujian *accuracy* tertinggi adalah perceptron dengan nilai sebesar 0,982, untuk pengujian *precision* tertinggi adalah unigram dengan nilai sebesar 0,997, untuk pengujian *recall* tertinggi adalah trigram dengan nilai 0,980 dan untuk pengujian *f-measure* tertinggi adalah unigram dengan nilai sebesar 0,981 .

**Grafik 4. 1** Hasil Pengujian Accuracy, Precision, Recall, dan F-measure

### Pengujian K-fold Cross Validation

#### 4.2.2.1 Hasil Pengujian K-*fold Cross Validation*

Pengujian K-fold cross validation menggunakan lima fold dengan pembagian 20% korpus uji dan 80% korpus latih dari total 1500 kalimat. Skenario pengujian K-*fold cross validation* seperti yang terlihat pada tabel 3.3. Hasil rincian perhitungan metode PoS tagger dengan pengujian *accuracy, precision, recall,* dan F-*measure* dapat dilihat lebih jelas pada lampiran A. Hasil perhitungan pada tabel 3.4 dapat dilihat pada tabel 4.3.

**Tabel 4. 3** Tabel Hasil Pengujian K-*fold Cross Validation*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Metode PoS Tagger | Accuracy | Precision | Recall | F-measure |
| Bigram | 0.197 | 0.211 | 0.227 | 0.219 |
| HMM | 0.969 | 0.972 | 0.961 | 0.967 |
| Perceptron | 0.859 | 0.736 | 0.761 | 0.748 |
| Trigram | 0.124 | 0.155 | 0.159 | 0.157 |
| Unigram | 0.770 | 0.539 | 0.523 | 0.529 |

Tabel 4.3 menunjukkan bhawa metode PoS Tagger bigram dan trigram untuk pengujian *accuracy* memberikan nilai yang cukup rendah dengan nilai besaran untuk akurasi dibawah 0.2, untuk pengujian *precision* bigram memberikan nilai dengan besaran 0.211, sedangkan untuk trigram sebesar 0.155, untuk pengujian *recall* metode bigram memberikan nilai sebesar 0.227, sedangkan untuk trigram sebesar 0.159, dan untuk pengujian *f-measure*, bigram memberikan nilai sebesar 0.219, dan trigram memberikan nilai 0.157. Sedangkan untuk metode hmm, perceptron, dan unigram memberikan nilai yang cukup dengan rata-rata nilai pada setiap pengujian berada diatas 0,5.

#### 4.2.2.2 Analisis Hasil Pengujian K-*fold Cross Validation*

Berdasarkan hasil pengujian K-*fold cross validation* yang telah dilakukan, nilai *accuracy, precision, recall,* dan *F-measure* tertinggi adalah metode HMM, pada metode ini dapat dilihat jelas pada gambar grafik 4.2. Pada pengujian dengan metode HMM memberikan nilai akurasi pada pengujian *accuracy* sebesar 0.969, *precision* sebesar 0.972, recall sebesar 0.961, dan *f-measure* sebesar 0.967, nilai tersebut adalah hasil dari rata-rata perhitungan yang dilakukan dengan melakukan pengujian sesusai dengan pada tabel 3.3 dengan perhitungan nilai yang diambil adalah hasil dari rata-rata setiap pengujian yang dilakukan.

**Grafik 4. 2** Hasil Pengujian K-fold Cross Validation

* 1. **Analisa Hasil Keseluruhan Pengujian**

Berdassarkan hasil dari keseluruhan pengujian yang telah dilakukan, nilai *accuracy, precision, recall,* dan *F-measure* yang didapatkan menunjukan bahwa untuk nilai *accuracy* metode perceptron memberikan nilai tertingi pada pengujian *accuracy*, sedangkan untuk pengujian *precision* metode unigram memberikan nilai tertinggi, sedangkan untuk pengujian *recall* nilai tertinggi adalah metode trigram, dan untuk pengujian *f-measure* nilai tertinggi adalah metode unigram. Hal ini dapat terjadi karena setiap metode dapat memprediksi setiap tagging yang telah dilatih dan dapat memprediksi antara perbedaan kelas kata dengan pengujian ini berbasiskan data dari 500 kalimat uji dan 500 kalimat latih yang sama.

Kemudian pengujian dilakukan dengan pengujian K-*fold cross validation* untuk melakukan pengujian terhadap 1500 kalimat yang dimana pada pengujian ini dilakukan sebanyak 5 fold yang terlampir pada lampiran A, B, C, D, E sehingga didapatkan nilai rata-rata dan kemudian dijabarkan kembali kedalam tabel 4.3 yang dimana pada tabel ini hasil latih dari tipe metode pos tagger dilihat kemampuan setiap tipe metode pos tagger tersebut dalam melakukan pengujian apakah dapat menghasilkan atau memilah kelas kata sesuai dengan data yang dilatih dan diuji, pengujian yang dilakukan berbasiskan 1500 kalimat yang dibagi menjadi 1200 kalimat latih, dan 300 kalimat uji, dari hasil tabel 4.3 dapat dilihat juga perbandingan setiap metode dari pengujian *accuracy, precision, recall,* dan F*-measure* memberikan nilai yang cukup baik bagi metode hmm dengan nilai-nilai yang dihasilkan memberikan nilai yang tinggi dibandingkan metode lainnya, sehingga dalam metode hmm memberikan analisa yang dimana pada metode ini dapat mengklasifikasikan kelas kata pada tiap kata yang diuji dan memberikan nilai yang lebih baik daripada metode lainnya.

# BAB V Kesimpulan dan Saran

## Kesimpulan

Berdasarkan hasil analisis dan pengujian terhadap nilai akurasi metode PoS Tagger untuk bahasa Melayu Pontianak, maka dapat ditarik kesimpulan sebagi berikut :

1. Hasil dari perbandingan tag PoS yang telah dilakukan menghasilkan tagging pos baru yang berumlah 22.
2. Hasil pengembangan korpus menghasilkan kalimat sebesar 1500 kalimat.
3. Dari hasil pengujian dengan menggunakan data kalimat uji dan kalimat latih sebesar 500 kalimat yang sama, didapatkan nilai *accuracy, precision, recall,* dan *F-measure*. Nilai *accuracy* dengan nilai yang paling tinggi dari tipe metode PoS Tagger adalah metode perceptron dengan nilai sebesar 0.982. Nilai *Precision* dengan nilai paling tinggi dari tipe metode PoS Tagger adalah metode unigram dengan nilai sebesar 0.997. Nilai *Recall* dengan nilai paling tinggi dari tipe metode PoS Tagger adalah metode trigram dengan nilai sebesar 0.980. Nilai *F-measure* dengan nilai paling tinggi dari tipe metode PoS Tagger adalah metode unigram dengan nilai sebesar 0.981.
4. Hasil dari pengujian *K-fold cross validation* dilakukan dengan pengujian terhadap 1500 kalimat korpus dan nilai tertinggi dari pengujian ini adalah metode HMM dengan nilai *accuracy* sebesar 0.969, nilai *precision* sebesar 0.972, *recall* sebesar 0.961, dan *F-measure* sebesar 0.967.

## Saran

Adapun beberapa hal yang perlu ditambahkan dalam pengembangan penelitian ini adalah sebagai berikut:

1. Dapat dilakukan pemanfaatan terhadap penelitian ini dengan menggunakan tag PoS dari penelitian yang telah dilakukan sehingga dapat mengembangkan sistem *text to speech* (TTS)Bahasa Melayu Pontianak.
2. Perlu dilakukan penambahan jumlah korpus dan dilakukan penelitian untuk mendapatkan nilai akurasi jeda terhadap kalimat berdasarkan tag PoS dari penelitian ini.

Daftar Pustaka

Abd Wahab Syahroni, Joan Santoso, Endang Setyati. (2017, Agustus 10). Pendekatan Rule Handmade untuk Menentukan Klausa Bahasa Indonesia. *Konferensi Nasional Sistem & Informatika 2017*.

Abney, S. P. (1991). *Parsing By Chunks, In Robert Berwick, Steven Abney, and Carol Tenny (eds)*. Kluwer Academic Publishers.

Adriani, M. 2009. *Developing Postag for Bahasa Indonesia*. PAN Localization Project*.* Universitas Indonesia.

Arry Akhmad Arman, A. B. (2013). *Syntatic Phrase Chunking for Indonesian Language*. Bandung: Institut Teknologi Bandung.

Christianti M, V.; Pragantha, J.; dan Victor. 2016. *Part-of-Speech Tagging untuk Bahasa Indonesia Menggunakan Stanford POS-Tagging.* Jakarta: Universitas Tarumanegara.

Fitriawati, Lia Suci. 2020. *Implementasi Text To Speech Pada Website Menggunakan Metode Shallow Parsing*. Pontianak: Universitas Tanjungpura.

Fitri, M. 2013. *Perancangan Sistem Temu Balik Informasi dengan Metode Pembobotan Kombinasi TF-IDF untuk Pencarian Dokumen Berbahasa Indonesia.* Pontianak: Universitas Tanjungpura.

Handel, V. R. 2008. *Uniform Observability of Hidden Markov Models and Filter Stability for Unstable Signals.* New Jersey: Universitas Princeton.

Kamiludin, M. I. 2017. *Prediksi Jeda pada Ucapan Bahasa Melayu Pontianak dengan Menggunakan Metode Shallow Parsing*. Pontianak: Universitas Tanjungpura.

Larasati, S. D.; Kuboň, V.; dan Zeman, D. 2011. *Indonesian Morphology Tool (MorphInd): Towards an Indonesian Corpus*. Springer CCIS proceedings of the Workshop on Systems and Frameworks for Computational Morphology*.* Zurich.

Muljono, Askarya Qaulan Syadida, De Rosal Ignatius Moses Setiadi, and Andik Setyono. 2017 ”Sphinx4 for Indonesian Continuous Speech Recognition System.” *Proceedings – 2017 International Seminar on Application for Technology of Information and Communication: Empowering Technology for a Better Human Life, Isemantic 2017* 2018-Januari(October):264-67.

Muniyati, Evi F. 2019. *Prediksi Jeda Dalam Ucapan Kalimat Bahasa Melayu Pontianak Menggunakan Hidden Markov Model Berbasis Part Of Speech*. Pontianak: Universitas Tanjungpura.

Magdalena, Y. 2019. *Prediksi Jeda pada Ucapan Kalimat Bahasa Melayu Pontianak Menggunakan Metode Shallow Parsing dengan Pengembangan Rule Grammar dan Rule Jeda.* Pontianak, Indonesia

Na'im, A. dan Syaputra, H. 2010. *Hasil Sensus Penduduk 2010: Kewarganegaraan, Suku Bangsa, Agama dan Bahasa Sehari-Hari Penduduk Indonesia.* Jakarta: Badan Pusat Statistik.

Nugraha, A. T. 2014. *Prediksi Jeda dalam Ucapan Kalimat Bahasa Indonesia dengan Hidden Markov Model.* Pontianak: Universitas Tanjungpura.

Pisceldo, F.; Adriani, M; dan Manurung, R. 2009. *Probabilistic Part Of Speech Tagging for Bahasa Indonesia*. Third International Wokshop on Malay and Indonesian Language Engineering*.* Singapore.

Prasetyo M. E. B. 2010. *Teori Dasar Hidden Markov Model*. Bandung: Institut Teknologi Bandung.

Rashel, F., Luthfi, A., Dinakaramani, A., & Manurung, R. (2014). *Building an Indonesian Rule-Based Part-of-Speech Tagger.* Depok: Universitas Indonesia.

Setyaningsih, E. R. 2017. *Part of Speech Tagger untuk Bahasa Indonesia dengan Menggunakan Modifikasi Brill*. Surabaya: Sekolah Tinggi Teknik Surabaya.

Purwarianti, A. (2010). *HMM Based Part-of-Speech Tagger for Bahasa Indonesia.* Bandung: Institut Teknologi Bandung.

Wibisono, Y. 2008. *Penggunaan Hidden Markov Model untuk Kompresi Kalimat.* Bandung: Institut Teknologi Bandung.

Wicaksono, A. F. dan Purwarianti, A. 2010. *HMM Based Part-of-Speech Tagger for Bahasa Indonesia*.Proceedings of 4th International MALINDO (Malay and Indonesian Language) Workshop*.*

1. LAMPIRAN A

**HASIL PERHITUNGAN K-*FOLD CROSS VALIDATION* METODE BIGRAM**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Bigram | | | | | |
| Korpus Latih | Korpus Uji | Accuracy | Precision | Recall | F-measure |
| 301+1500 | 1-300 | 0,156 | 0,201 | 0,217 | 0,209 |
| 1-300+601-1500 | 301-600 | 0,16 | 0,199 | 0,213 | 0,206 |
| 1-600+901-1500 | 601-900 | 0,195 | 0,207 | 0,219 | 0,213 |
| 1-900+1201+1500 | 901-1200 | 0,251 | 0,234 | 0,251 | 0,242 |
| 1-1200 | 1201-1500 | 0,215 | 0,217 | 0,236 | 0,227 |
| Rata-rata | | 0,197 | 0,212 | 0,227 | 0,219 |

1. LAMPIRAN B

**HASIL PERHITUNGAN K-*FOLD CROSS VALIDATION* METODE HMM**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| HMM | | | | | |
| Korpus Latih | Korpus Uji | Accuracy | Precision | Recall | F-measure |
| 301+1500 | 1-300 | 0.972 | 0.985 | 0.974 | 0.979 |
| 1-300+601-1500 | 301-600 | 0.976 | 0.973 | 0.960 | 0.967 |
| 1-600+901-1500 | 601-900 | 0.966 | 0.968 | 0.960 | 0.964 |
| 1-900+1201+1500 | 901-1200 | 0.966 | 0.978 | 0.960 | 0.969 |
| 1-1200 | 1201-1500 | 0.965 | 0.959 | 0.951 | 0.955 |
| Rata-rata | | 0.969 | 0.972 | 0.961 | 0.967 |

1. LAMPIRAN C

**HASIL PERHITUNGAN K-FOLD CROSS VALIDATION METODE PERCEPTRON**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Perceptron | | | | | |
| Korpus Latih | Korpus Uji | Accuracy | Precision | Recall | F-measure |
| 301+1500 | 1-300 | 0.818 | 0.653 | 0.693 | 0.673 |
| 1-300+601-1500 | 301-600 | 0.982 | 0.963 | 0.973 | 0.968 |
| 1-600+901-1500 | 601-900 | 0.839 | 0.699 | 0.729 | 0.713 |
| 1-900+1201+1500 | 901-1200 | 0.829 | 0.693 | 0.708 | 0.700 |
| 1-1200 | 1201-1500 | 0.829 | 0.671 | 0.701 | 0.686 |
| Rata-rata | | 0.859 | 0.736 | 0.761 | 0.748 |

1. LAMPIRAN D

**HASIL PERHITUNGAN K-*FOLD CROSS VALIDATION* METODE TRIGRAM**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Trigram | | | | | |
| Korpus Latih | Korpus Uji | Accuracy | Precision | Recall | F-measure |
| 301+1500 | 1-300 | 0.096 | 0.142 | 0.145 | 0.144 |
| 1-300+601-1500 | 301-600 | 0.120 | 0.157 | 0.163 | 0.160 |
| 1-600+901-1500 | 601-900 | 0.133 | 0.163 | 0.167 | 0.165 |
| 1-900+1201+1500 | 901-1200 | 0.140 | 0.159 | 0.161 | 0.160 |
| 1-1200 | 1201-1500 | 0.128 | 0.153 | 0.157 | 0.155 |
| Rata-rata | | 0.124 | 0.155 | 0.159 | 0.157 |

1. LAMPIRAN E

**HASIL PERHITUNGAN K-*FOLD CROSS VALIDATION* METODE UNIGRAM**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Unigram | | | | | |
| Korpus Latih | Korpus Uji | Accuracy | Precision | Recall | F-measure |
| 301+1500 | 1-300 | 0.768 | 0.539 | 0.521 | 0.530 |
| 1-300+601-1500 | 301-600 | 0.768 | 0.546 | 0.529 | 0.537 |
| 1-600+901-1500 | 601-900 | 0.778 | 0.541 | 0.517 | 0.529 |
| 1-900+1201+1500 | 901-1200 | 0.770 | 0.547 | 0.547 | 0.534 |
| 1-1200 | 1201-1500 | 0.768 | 0.522 | 0.503 | 0.513 |
| Rata-rata | | 0.770 | 0.539 | 0.523 | 0.529 |