

BAB I

PENDAHULUAN

1.1 Latar Belakang

Bahasa Melayu Pontianak merupakan salah satu bahasa daerah yang ada di Provinsi Kalimantan Barat. Bahasa Melayu Pontianak banyak digunakan sebagai alat komunikasi oleh orang Melayu yang ada di Pontianak sebagai bahasa sehari-hari mereka. Berdasarkan hasil sensus penduduk yang dilakukan Badan Pusat Statistik pada 2010, persentase penggunaan bahasa Melayu yang digunakan oleh masyarakat Kalimantan Barat mencapai 20,45% (1.615.978 juta jiwa) dari total penduduk Kalimantan Barat (Na'im & Syaputra, 2010). Upaya pelestarian bahasa Melayu Pontianak agar tidak punah dan ditinggalkan karena pengaruh globalisasi tetap harus dilakukan. Salah satu upaya dengan penerapan teknologi *Text To Speech* (TTS).

Teknologi *Text To Speech* (TTS) adalah suatu sistem yang dapat melakukan konversi dari teks menjadi ucapan. Kualitas ucapan yang dihasilkan oleh sistem TTS dinilai dari ucapan yang jelas dan alami. Indikator ucapan yang jelas dan alami adalah nada, tekanan (keras-lemah), durasi, intonasi, dan jeda (kesenyapan). Pemberian jeda dalam kalimat sangat penting karena jeda yang terjadi dapat membentuk fase prosodi. Pemberian jeda dapat memperjelas informasi dari suatu kalimat yang ingin disampaikan sehingga mudah dipahami. Jeda ucapan didapat dari pemenggalan frasa ucapan. Frasa adalah satuan gramatik yang terdiri atas satu kata atau lebih (Abd Wahab Syahroni, Joan Santoso, Endang Setyati, 2017). Untuk mendapatkan frasa suatu kalimat dapat menggunakan metode *Shallow Parsing* dengan menstrukturkan ucapan berdasarkan aturan atau *rule grammar*. Metode *shallow parsing (chunking)* dapat menggunakan fungsi *RegexpParser* pada *Natural Language Toolkit* (NLTK) untuk memenggal kalimat menjadi frasa-frasa berdasarkan tipe *Part of Speech* (PoS) (Prathibha, R.J and Padma, M.C, 2017). *Grammar* atau aturan sederhana diperlukan untuk mendefinisikan struktur dari *chunk*. *Chunk* merepresentasikan penggalan kalimat yang terjadi ketika membaca semua kalimat (Abney, 1991). Berdasarkan hal ini, jeda dapat ditentukan menggunakan penggalan kalimat atau frasa dari metode *shallow parsing*.

Penelitian tentang *Shallow Parsing* pada bahasa Melayu Pontianak telah

dilakukan oleh M. Iqbal Kamiludin (2017), dimana di penelitian tersebut dibangun *rule grammar* pada metode *shallow parsing* untuk mengidentifikasi penggalan frasa jeda dalam kalimat bahasa Melayu Pontianak. Frasa jeda ucapan adalah frasa atau potongan yang terjadi dalam suatu kalimat dikarenakan adanya tanda baca atau tanda jeda. *Rule grammar* yang dibuat berdasarkan struktur kalimat dimana sistem akan memenggal kalimat menjadi potongan kalimat berdasarkan aturan S-P-O-K. Hasil pengujian yang didapatkan berupa total hasil perhitungan *F-measure* sebesar 0.64. Nilai *recall* dan *precision* untuk kalimat tunggal sebesar 0.78 dan 0.74 , sedangkan untuk kalimat majemuk sebesar 0.67 dan 0.57. Menurut M. Iqbal Kamiludin (2017), penggalan frasa jeda *shallow parsing* belum mampu menghasilkan nilai akurasi kecocokan pada frasa jeda penutur dengan nilai baik. Dari 168 kalimat yang ada, nilai kecocokan dengan jeda penutur sebesar 40.4% atau 68 kalimat. Hal ini dikarenakan penelitian sebelumnya menggunakan *rule* frasa struktur kalimat sehingga potongan frasa yang dihasilkan tidak mengacu kepada potongan frasa jeda yang ditandai penutur. Hasil penelitian masih tergolong rendah sehingga jika diimplementasikan ke sistem *text to speech* dapat memberikan makna yang berbeda dari penutur. Berdasarkan hal tersebut, diperlukan suatu *rule* baru untuk mengkategorikan frasa sesuai jeda penutur.

Tanda jeda dalam suatu kalimat digunakan untuk mempermudah penutur atau pendengar mendengar informasi dari kalimat yang disampaikan secara jelas. Jeda tidak hanya berdasarkan potongan struktur kalimat S-P-O-K seperti pada penelitian M. Iqbal Kamiludin (2017), tetapi bisa berdasarkan dari ucapan penutur langsung. Ada dua faktor yang mempengaruhi gaya jeda ucapan dalam berbicara yaitu keraguan pembicara dan pernafasan (Darjadowidjojo, 2005). Abney (1991) menjelaskan ketika membaca sebuah kalimat kita cenderung mengelompokkan kata menjadi frasa. Sehingga, jeda terjadi tidak hanya berdasarkan pengaruh aturan S-P-O-K, melainkan dapat dipengaruhi oleh penutur itu sendiri.

Berdasarkan penjelasan yang telah dipaparkan, maka dilakukan penelitian untuk mengembangkan *rule grammar* dan *rule* jeda untuk mengkategorikan frasa jeda ucapan pada kalimat bahasa Melayu Pontianak dengan menggunakan metode *Shallow Parsing* agar dapat meningkatkan nilai akurasi jeda pada ucapan kalimat bahasa Melayu Pontianak.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dikemukakan, maka permasalahan yang pada penelitian ini adalah bagaimana meningkatkan nilai akurasi prediksi jeda pada teks kalimat bahasa Melayu Pontianak menggunakan metode *shallow parsing* dengan mengembangkan *rule grammar* dan *rule* jeda berdasarkan frasa ucapan penutur.

1.3 Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk meningkatkan akurasi prediksi jeda pada kalimat bahasa Melayu Pontianak dengan metode *shallow parsing* berdasarkan pengembangan *rule grammar* dan *rule* jeda.

1.4 Pembatasan Masalah

Beberapa hal yang menjadi batasan dalam penelitian ini adalah sebagai berikut.

1. Data yang digunakan adalah kalimat bahasa Melayu Pontianak dari buku Sepok 1 karangan Pay Jarrot Sujarwo yang berjumlah 500 kalimat.
2. Data jeda berasal dari rekaman ucapan asli penutur ahli bahasa Melayu Pontianak.
3. Label *PoS* untuk penandaan kelas kata kalimat menggunakan *PoS Tagger* bahasa Melayu Pontianak yang dibuat pada penelitian ini.

1.5 Sistematika Penulisan Skripsi

Sistematika penulisan penelitian ini disusun untuk memberikan gambaran umum tentang penelitian yang dijalankan. Sistematika laporan tugas akhir ini disusun dalam 5 (lima) bab yang terdiri dari Bab I Pendahuluan, Bab II Tinjauan Pustaka dan Landasan Teori, Bab III Metodologi Penelitian, BAB IV Implementasi dan Hasil Penelitian, dan Bab V Penutup.

Bab I Pendahuluan menjelaskan secara umum mengenai latar belakang permasalahan, perumusan masalah, tujuan penelitian, pembatasan masalah, serta sistematika penulisan laporan tugas akhir.

Bab II Tinjauan Pustaka dan Landasan Teori membahas mengenai gambaran umum tentang penelitian yang didapat oleh peneliti sebelumnya serta perangkat lunak yang melandasi pembangunan sistem dan landasan teori yang berhubungan dalam proses analisis permasalahan penelitian yang akan dilakukan.

Bab III Metodologi Penelitian membahas mengenai data dan perangkat penelitian, metode yang akan digunakan pada penelitian, diagram alir penelitian, dan perancangan pengujian yang dilakukan pada penelitian.

Bab IV Hasil dan Analisis membahas mengenai hasil dari analisis yang dilakukan, hasil pengujian dan analisis pengujian yang mengarah pada suatu kesimpulan.

Bab V Penutup membahas mengenai kesimpulan yang sudah diperoleh dari hasil penulisan laporan tugas akhir dan saran untuk perbaikan, pengembangan atau kesempurnaan penelitian yang telah dilakukan.

BAB II

TINJAUAN PUSTAKA

2.1 Kajian Terkait

Beberapa penelitian terdahulu yang telah dilakukan tentang prediksi jeda salah satunya Claire Brierley dan Eric Atwell (2007), *University of Birmingham, UK* dalam jurnal yang berjudul *Corpus-Based Evaluation of Prosodic Phrase Break Prediction Using nltk_lite's Chunk Parser to Detect Prosodic Phrase Boundaries in the Aix-MARSEC Corpus of Spoken English* yang meneliti prediksi jeda berdasarkan *rule* untuk memprediksi jeda frasa prosodi. Frasa prosodi yang dimaksud adalah ucapan dari seseorang yang dipenggal menjadi penggalan-penggalan kalimat yang memiliki makna. Penelitian ini menggunakan bantuan *nltk_lite's regular expression chunk parser* untuk memprediksi jeda. Korpus yang digunakan adalah korpus dari *Aix-MARSEC Corpus of spoken English* yang berjumlah sekitar 1400 kata. Pengujian dilakukan dengan membandingkan jeda dari penutur dan jeda dari *rule*. Jeda dari *rule* dibagi menjadi dua yaitu jeda untuk inputan teks tanpa titik dan koma, dan jeda untuk inputan teks dengan titik dan koma. Hasil penelitian menunjukkan nilai 40.5% untuk jeda inputan teks tanpa titik dan koma, dan nilai 43.5% untuk jeda inputan teks dengan titik dan koma.

Jian-feng Li, Guo-ping Hu, Wan-ping Zhang dan Ren-hua Wang (2004), *iFly Speech Lab, University of Science and Technology of China, Hefei China*, dalam jurnal yang berjudul *Chinese Prosody Phrase Break Prediction Based on Maximum Entropy Model* menjelaskan tentang prediksi jeda dalam kalimat teks Mandarin menggunakan *maximum entropy based model*. Prediksi jeda pada penelitian ini dilakukan pada korpus bahasa Mandarin sebanyak 42.000 kalimat yang dibagi menjadi 10.000 korpus uji dan 32.000 korpus *train*. Penelitian ini menghasilkan nilai keakuratan sebesar 62.91% untuk model PoS dan 65.24% untuk model PoS dan *lexical*.

Adhitya Teguh Nugraha (2014), Universitas Tanjungpura, dalam skripsi yang berjudul *Prediksi Jeda dalam Ucapan Kalimat Bahasa Indonesia dengan Hidden Markov Model*, melakukan penelitian yang menghasilkan aplikasi untuk memprediksi jeda dalam ucapan kalimat berbahasa Indonesia dengan menggunakan *Hidden Markov Model (HMM)*. Korpus 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 (2010). Hasil dari penelitian ini yaitu nilai *recall* sebesar 0.132, nilai *precision* sebesar 0.364 dan *f-score* sebesar 0.194.

M.Iqbal Kamiludin (2017) , Universitas Tanjungpura, dalam skripsi yang berjudul *Prediksi Jeda Pada Ucapan Bahasa Melayu Pontianak dengan Menggunakan Metode Shallow Parsing* melakukan penelitian tentang prediksi frasa jeda ucapan kalimat Bahasa Melayu Pontianak dengan menggunakan penggalan *shallow parsing* yang dibentuk berdasarkan aturan *grammars* yang telah dibuat dan PoS *tagger*. Penggalan frasa yang terbentuk untuk memprediksi jeda berhubungan dengan struktur kalimat S-P-O-K. Teks kalimat yang digunakan berasal dari buku *Sepok Satu* dan *Sepok Duak* karangan Pay Jarot. Pengujian dilakukan terhadap 168 kalimat bahasa Melayu Pontianak yang terdiri atas 47 kalimat tunggal dan 121 kalimat majemuk dengan hasil perhitungan *F-measure* untuk kalimat tunggal 0.75 sedangkan *F-measure* untuk kalimat majemuk sebesar 0.61. Nilai akurasi kecocokan pada frasa jeda penutur juga belum dikategorikan bagus karena dari 168 kalimat, nilai kecocokan dengan jeda penutur sebesar 40.4% atau 68 kalimat.

Tabel 2.1 Perbandingan Studi Literatur

No.	Penulis	Metode yang Digunakan	Hasil Penelitian
1.	Claire Brierley dan Eric Atwell (2007), <i>Corpus-Based Evaluation of Prosodic Phrase Break Prediction Using nltk_lite's Chunk Parser to Detect Prosodic Phrase</i>	Prediksi jeda menggunakan <i>nltk_lite's regular expression chunk parser</i>	Prediksi jeda yang dihasilkan memiliki nilai kemunculan jeda yang sama dengan jeda penutur sebesar 40.5% untuk inputan teks tanpa titik dan koma, dan nilai 43.5% untuk inutan teks dengan titik dan koma.

No.	Penulis	Metode yang Digunakan	Hasil Penelitian
	<i>Boundaries in the Aix-MARSEC Corpus of Spoken English</i>		
2.	Jian-feng Li, Guo-ping Hu, Wan-ping Zhang dan Ren-hua Wang (2004), <i>Chinese Prosody Phrase Break Prediction Based on Maximum Entropy Model</i>	Prediksi jeda menggunakan metode <i>maximum entropy based model</i> untuk 2 model yaitu model PoS dan model PoS & <i>lexical</i>	Prediksi jeda yang dihasilkan memiliki nilai keakuratan sebesar 62.91% untuk model PoS dan 65.24% untuk model PoS dan <i>lexical</i> .
3.	Adhitya Teguh Nugraha (2014), Prediksi Jeda dalam Ucapan Kalimat Bahasa Indonesia dengan <i>Hidden Markov Model</i>	Menggunakan metode HMM untuk memprediksi jeda	Hasil dari penelitian adalah nilai <i>recall</i> sebesar 0,132, nilai <i>precision</i> sebesar 0,364 dan <i>f-score</i> sebesar 0,194
4.	M.Iqbal Kamiludin (2017), Prediksi Jeda Pada Ucapan Bahasa Melayu Pontianak Dengan	Menggunakan metode <i>Shallow Parsing</i>	Hasil penelitian adalah nilai <i>f-measure</i> sebesar 0.75 untuk uji kalimat tunggal dan 0,61 untuk uji kalimat majemuk.

No.	Penulis	Metode yang Digunakan	Hasil Penelitian
	Menggunakan Metode <i>Shallow Parsing</i>		

Penelitian ini melakukan penandaan jeda ucapan menggunakan metode *shallow parsing* pada teks kalimat Bahasa Melayu Pontianak dengan *rule grammar* dan *rule* jeda ucapan yang akan dikembangkan dalam penelitian ini untuk meningkatkan nilai akurasi jeda .

Tabel 2.2 Penelitian yang dilakukan

No.	Penulis	Judul	Keterangan
1.	Yulia Magdalena (2019), Universitas Tanjungpura	Prediksi Jeda pada Ucapan Kalimat Bahasa Melayu Pontianak Menggunakan Metode <i>Shallow Parsing</i> dengan Pengembangan <i>Rule Grammar</i> dan <i>Rule</i> Jeda	<ul style="list-style-type: none"> • Menggunakan metode <i>Shallow Parsing</i> • Menggunakan <i>rule grammars</i> dan <i>rule</i> jeda • Data kalimat berasal dari buku Sepok Satu berjumlah 500 kalimat.

2.2 *Part of Speech Tagging*

Part of Speech (PoS *Tagging*) , yang juga disebut sebagai pelabelan kelas kata, adalah suatu proses yang memberikan label kelas kata secara otomatis pada suatu kata dalam kalimat (Jurafsky, Daniel, dan James H.Martin, 2009). Kelas kata terdiri dari kata sifat, kata benda, kata kerja, kata keterangan, kata depan, kata kepemilikan, kata penghubung, dll. Suatu kata bisa diklasifikasikan ke dalam kelas kata yang berbeda (Kamiludin, 2017). *Part of Speech Tagging* merupakan bagian dari *Natural Language Processing* dalam menentukan kelas kata. Hasil penelitian *Part of Speech Tagging* pada dokumen dapat digunakan sebagai dasar penelitian dalam *Natural Language Processing* lainnya, seperti: *Language Generator*,

Information Retrieval, Text Summarization, Question and Answering, dan Machine Translation.

Pelabelan kata dapat dilakukan berbasis aturan (*rule-based*) dan probabilitas (*probability-based*) dari sebuah model yang dibangun. *Rule-based tagging* dilakukan dengan cara top-down, yaitu melakukan konsultasi dengan ahli linguistik untuk mendefinisikan aturan-aturan yang biasa digunakan manusia. Sedangkan *probability based tagging* dilakukan dengan cara *bottom-up*, yaitu menggunakan korpus sebagai *training* data untuk menentukan secara probabilistik tag yang terbaik untuk sebuah kata dalam sebuah konteks.

Berbagai set PoS telah digunakan dalam penelitian-penelitian bidang pemrosesan bahasa alami (PBA), diantaranya lewat PAN *Localization Project*, khusus untuk bahasa indonesia telah dikembangkan PoS khusus untuk terjemahan ke bahasa Inggris pada tahun 2009 (Adriani M, Riza H, 2008). Kelas kata tersebut dibangun berdasarkan Penn Treebank PoS *tag sets* (Marcus M.P, Maricinkiewicz M.A, Santoroni B, 1993) terdiri dari 29 PoS tag. PoS utama bahasa indonesia adalah kata kerja (*verb*), kata sifat (*adjective*), kata tugas (*function words*), kata keterangan(*adverb*), dan kata benda (*noun*). Pada penelitian Herry Sujaini (2014) digunakan 42 *tag* PoS untuk Bahasa Indonesia.

No	Tag	Description	Word Examples
1	OP	Opening parenthesis	{ { [
2	CP	Closing parenthesis	} }]
3	GM	Slash	/
4	;	Semicolon	;
5	:	Colon	:
6	"	Quotation	" "
7	.	Sentence terminator	. ? !
8	,	Comma	,
9	-	Dash	-
10	...	Ellipsis	...
11	JJ1	Adjectives 1	panjang, kuat, indah, besar
12	JJ2	Adjectives 2	genap, buntu, negatif
13	RB	Adverbs	sekedar, hampir, tidak
14	RB1	Adverbs 1	sangat, amat, cukup, paling
15	NN	Common Noun	mobil, air, negara
16	NNP	Proper nouns	tvri, jokowi, persib
17	NNG	Genitive nouns	bukunya, hatinya
18	VBI	Intransitive Verb	duduk, pergi, makan
19	VBI1	Intransitive Verb 1	dibuat, diambil
20	VBI2	Intransitive Verb 2	mengecil, menguat
21	VBT	Transitive Verb	membeli, memukul
22	IN	Preposition	di, ke, dari
23	MD	Modal	akan, harus
24	CC	Coor - conjunction	dan, atau, ketika, jika
25	DT	Determiner	ini, itu
26	UH	Interjections	wah, aduh, oi
27	CDO	Ordinal numerals	pertama, kedua
28	CDC	Collective numerals	berdua, bertiga
29	CDP	Primary numerals	1, 2, 3
30	CDP1	Primary numerals 1	satu, dua
31	CDP2	Primary numerals 2	puluh, ribu, juta
32	CDP3	Primary numerals 3	1990, 2001, 2013
33	CDI	Irregular numerals	beberapa
34	PRP	Personal pronoun	saya, kamu
35	WP	WH-pronouns	apa, siapa
36	PRN	Number pronouns	kedua-duanya
37	PRL+	Locative Proper nouns/pronouns	sini, situ, Jakarta, Bali
38	SYM	Symbols	@#\$%^&
39	RP	Particles	pun, kah
40	FW	Foreign words	foreign, Word
41	ART	Articles	sang, si, para
42	COP	Copula	adalah, bukan, merupakan

Gambar 2.1 Set PoS Herry Sujaini (2014)

No.	Category	Post-name	Postag
1	Noun	Countable Common Noun	NNC
2	Noun	Uncountable Common Noun	NNU
3	Noun	Genitive Common Noun	NNG
4	Noun	Proper Common Noun	NNP
5	Verb	Transitive	VBT
6	Verb	Intransitive	VBI
7	Verb	Modal	MD
8	Adjective	Adjective	JJ
9	Adverb	Adverb	RB
10	Wh-Adverb		WRB
11	Preposition		IN
12	Conjunction	Coordinate Conjunction	CC
13	Conjunction	Subordinate Conjunction	SC
14	Pronoun	Personal Pronoun	PRP
15	Pronoun	Wh-Pronoun	WP
16	Pronoun	Number Pronoun	PRN
17	Pronoun	Locative Pronoun	PRL
18	Interjection		UH
19	Punctuation		PUN
20	Symbol		SYM
21	Determiner		DT
22	Determiner	Wh-Determiner	WDT
23	Particle		RP
24	Cardinal Numeral	Primary Numeral	CDP
25	Cardinal Numeral	Ordinal Numeral	CDO
26	Cardinal Numeral	Irregular Numeral	CDI
27	Cardinal Numeral	Collective Numeral	CDC
28	Negation	Wh-Determiner	NEG
29	Foreign Word		FW

Gambar 2.2 Set PoS Adriani (2009)

Selain itu, pada penelitian Eka Rahayu Setyaningsih (2017) yang berjudul *Part-of-Speech Tagger* untuk bahasa Indonesia dengan menggunakan modifikasi *Bill*, digunakan Tagset (kelas kata) yang terdiri dari 8 bagian besar, yaitu kata benda (*noun*), kata sifat (*adjective*), kata keterangan (*adverb*), kata penghubung (*conjunction word*), kata ganti orang personal (*personal pronoun*), kata bilangan (*numeral*), dan lainnya (*other*) seperti pada gambar 2.3.

No	Tagset	Contoh	Simbol
NOUN			
1	Noun	Martabat	N
2	Singular Noun	Orang	NS
3	Plural Noun	Orang-orang	NP
4	Proper Common Noun	Jakarta, Raffles, Surabaya	NNP
5	Generative Common Noun	Keduanya, satunya	NNG
VERB			
1	Transitive Verb	Membeli, menyiram	VBT
2	Intransitive Verb	Pergi, pulang	VBI
PERSONAL PRONOUN			
1	Personal Pronoun	aku, kami, kamu, kalian	PRP
2	WH Pronoun	Siapa, apa	WDT
3	Locative Pronoun	Sana, sini, situ	PRL
ADJECTIVE			
1	Adjective	Mahal, besar, cantik	JJ
ADVERB			
1	Adverb	Paling, akan, sementara, sekarang	RB
2	WH Adverb	Bagaimana, mengapa	WPRB
NUMERAL			
1	Primary Numeral	Satu, dua	CDP
2	Ordinal Numeral	Kesatu, kedua	CDO
3	Irregular Numeral	Beberapa, segala, semua	CDI
4	Kata Bantu Bilangan	Biji, ekor, orang, buah	CDA
5	Collective Numeral	Ratusan, ribuan	CDC
CONJUNCTION			
1	Coordinate Conjunction	Dan, atau, tetapi	CC
2	Subordinate Conjunction	Jika, meskipun, yang oleh, ketika	SC
OTHER			
1	Determiner	Sesuatu, setiap, masing-masing	DT
2	Preposition	Ke, di, dari, pada	IN
3	Auxiliary (modal)	Akan, boleh, sudah, harus	MD
4	Auxiliary (negate)	Tidak, bukan, jangan, belum	NEG
5	Particle	Kah, lah, tah, pun	RP
6	Currency	Rp, \$	CR
7	Symbol	@/#/%^	SYM
8	Sentence Terminator	. ? !	ST
9	Comma	,	,
10	Ellipsis
11	Colon	:	:
12	Open Paranthesis	(OP
13	Close Paranthesis)	CP
14	Quotation	" "	QT
15	Dash	-	-
16	Interjection	Aduh, astaga, ah, hai	UH
17	Foreign Word	Music	FW
18	Article	Sri, Sang, Yang	AR
19	Unit Symbol	Kg, Gr, Cm	US

Gambar 2.3 Set PoS Eka Rahayu Setyaningsih (2017)

Iqbal Kamiluddin (2017) dalam penelitiannya mengenai prediksi jeda menggunakan *shallow parsing* menggunakan 35 tag PoS dari Alfian Farizki Wicaksono (2010) dan dilakukan secara manual sebanyak 168 kalimat Bahasa Melayu Pontianak. Tabel tag PoS Bahasa Melayu Pontianak seperti pada tabel 2.3 berikut:

Tabel 2.3 PoS Tag Bahasa Melayu Pontianak M.Iqbal Kamiluddin (2017)

No	PoS	Deskripsi	Contoh Kata
1	OP	<i>Parenthesis</i>	{[
2	CP	<i>Close Parenthesis</i>)]
3	GM	<i>Slash</i>	/
4	;	<i>Semicolon</i>	;
5	:	<i>Colon</i>	:
6	“	<i>Quotation</i>	“ ”
7	.	<i>Sentence Terminator</i>	. ! ?
8	,	<i>Comma</i>	,
9	-	<i>Dash</i>	-
10	...	<i>Ellipsis</i>	...
11	JJ	<i>Adjective</i>	Kaya, Manis
12	RB	<i>Adverb</i>	Nanti, Sementara
13	NN	<i>Common Noun</i>	Mobil, Sepatu, Roda
14	NNP	<i>Proper Noun</i>	Bekasi, Indonesia
15	NNG	<i>Genitive Noun</i>	Bukunya
16	VBI	<i>Intransitive Verb</i>	Pergi
17	VBT	<i>Transitive Verb</i>	Membeli
18	IN	<i>Preposition</i>	Di, Ke, Dari
19	MD	<i>Modal</i>	Bisa
20	CC	<i>Coor-Conjunction</i>	Dan, Atau, Tetapi
21	SC	<i>Subor-Conjunction</i>	Jika, Ketika
22	DT	<i>Determiner</i>	Para, Ini, Itu
23	UH	<i>Interjection</i>	Wah, Aduh, Oi
24	CDO	<i>Ordinal Numerals</i>	Pertama, Kedua
25	CDC	<i>Collective Numerals</i>	Bertiga
26	CDP	<i>Primary Numerals</i>	Satu, Dua
27	CDI	<i>Irregular Numerals</i>	Beberapa
28	PRP	<i>Personal Pronouns</i>	Saya, Kamu
29	WP	<i>WH-Pronouns</i>	Apa, Siapa, Dimana
30	PRN	<i>Number Pronouns</i>	Kedua-duanya, Satu-satunya
31	PRL	<i>Locative Pronouns</i>	Sini , Situ, Sana
32	NEG	<i>Negation</i>	Bukan, Tidak
33	SYM	<i>Symbols</i>	@#\$\$%^&
34	RP	<i>Particles</i>	Pun, Kah
35	FW	<i>Foreign</i>	Words

Penelitian ini membuat PoS bahasa Melayu Pontianak yang baru berdasarkan referensi tag PoS yang sudah dijelaskan untuk keperluan pengembangan *rule grammar* yang digunakan dalam proses *shallow parsing*.

2.3 *Shallow Parsing*

Shallow parsing adalah proses untuk mengelompokkan secara berurutan kata-kata ke dalam bentuk frasa dengan sebuah *chunker*, atau bisa disebut *chunks* (Ye, 2009). Proses *shallow parsing* akan memenggal kalimat menjadi frasa-frasa sesuai dengan aturan atau *rule grammar* yang telah disiapkan. *Shallow parsing* akan diproses menggunakan fungsi dari NLTK (*Natural Language Toolkit*) pada *python* yaitu *RegexpParser* dan aturan tata-bahasa NLTK *Tag Patterns* atau *Regular Expression*.

Contoh *shallow parsing* :

Presiden Jokowi telah meresmikan Jembatan Tayan tadi siang. Masukan kalimat lalu *tag* setiap kata melalui PoS tagger , maka fungsi *shallow parsing* yang akan memberikan tag untuk membentuk frasa:

- 0 (NP Presiden/NN (NP Jokowi/NNP))
- 1 (VP (KP telah/MD) (VP meresmikan/VBT))
- 2 (NP jembatan/NN Tayan/NNP)
- 3 (NP (KP tadi/RB) (NP siang/NN))

Output NP frasa untuk frasa Presiden Jokowi dapat dikategorikan sebagai subjek, frase VP untuk frasa telah meresmikan dikategorikan sebagai predikat, frasa NP untuk frasa jembatan Tayan dikategorikan sebagai objek, dan NP frasa dalam kalimat tadi siang sebagai deskripsi.

Jenis-jenis frasa *shallow parsing* yang telah dikembangkan dimaksudkan untuk membentuk frasa yang berhubungan dengan sintaksis kategori dan dapat digunakan untuk menginformasikan pembentukan frasa jeda ucapan kalimat. Jenis frasa yang dikembangkan berdasarkan penelitian M.Iqbal Kamiludin (2017) adalah TP (*Questioning Phrases*), BP (*Numeric Phrase*), KP (*Connection Phrase*), NP (*Noun Phrases*) dan VP (*Verb phrases*) yang merupakan set POS dari penelitian (Arry Akhmad Arman, 2013) seperti pada gambar 2.4.

Types of Phrases	POS Tag
Questioning Phrase (TP)	WP
Numeric Phrases (BP)	CDO, CDP, CDI, CDC
Connection Phrases (KP)	SC, CC, NEG, IN, MD, RB
Noun Phrases (NP)	NN, PRN, PRP, PRL, NNG, NNP, FW, RP, UH, JJ
Verb Phrases (VP)	VBL, VBT

Gambar 2.4 Set POS frasa *shallow parsing* (Arry Akhmad Arman, 2013)

2.4 *Natural Language Toolkit*

Meski 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 ataupun *pre-processed* yang digunakan dalam materi pengajaran NLP. NLTK adalah 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).

2.5 *Regular Expression*

Regular Expressions adalah konstruksi bahasa untuk mencocokkan sebuah string dengan sebuah pola. *Regular Expressions* memungkinkan kita untuk melakukan pencarian, substitusi, atau pemisahan string dalam kasus yang kompleks. *Regular Expressions* juga berguna dalam proses penguraian kata (*text parsing*). Berikut karakter *Regular Expression* yang digunakan dalam *shallow parsing* pada tabel 2.3.

Tabel 2.3 Makna karakter *Regular Expressions*

karakter	Makna karakter <i>Regular Expressions</i>
< >	Penentuan tag part-of-speech
?	Nol atau salah satu item sebelumnya
*	Nol atau lebih dari item sebelumnya.
+	Satu atau lebih dari item sebelumnya
	Mencocokkan satu item dengan yang lainnya

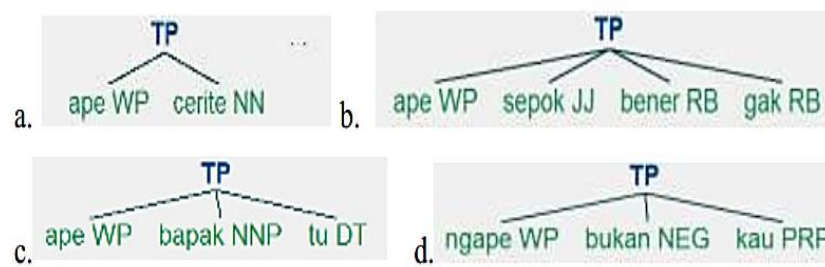
2.6 Rule Grammars

Rule Grammars adalah *rule* atau aturan yang akan digunakan sebagai pembentuk penggalan frasa dalam kalimat yang diolah. Pada penelitian yang dilakukan oleh M.Iqbal Kamiludin (2017), *rule* yang digunakan adalah *rule* dari penelitian Arry Akhmad Arman (2013) yang berjumlah 9 *rule* yang sebelumnya digunakan untuk proses penentuan frasa *chunking* Bahasa Indonesia. *Rule grammars* yang diteliti di penelitian tersebut dibentuk berdasarkan struktur kalimat Bahasa Melayu Pontianak dan berdasarkan data jeda dari rekaman penutur ahli Bahasa Melayu Pontianak.

Rule grammars terdiri dari nama *rule* yang diambil dari tipe frasa set PoS yang digunakan dan diikuti komponen utama pembentuk frasa dengan ketentuan kemunculan karakter dari tag PoS yang mengikuti komponen utama untuk digunakan dalam membentuk potongan frasa dan menggunakan bantuan *regular expression* untuk penguraian kata (*text parsing*). Berikut ini adalah contoh *rule grammars* yang dikembangkan oleh M.Iqbal Kamiludin(2017) di penelitian sebelumnya :

a. Rule 1 TP : {<WP>+ <NN|JJ|NNP|NNG|PRP|DT|RB|MD|NEG|RP>*

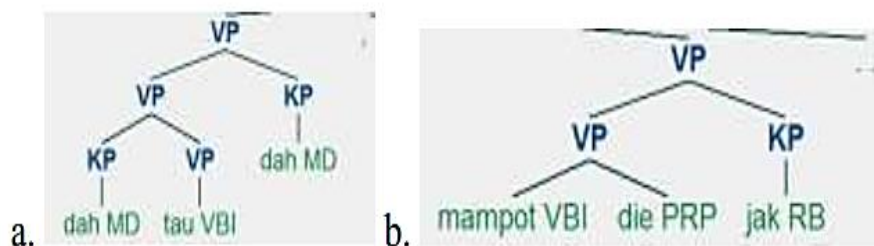
Dalam ketentuan *rule* TP ini komponen utama pembentuk *rule*-nya adalah tag PoS WP, dengan ketentuan <WP>+ mengharuskan ada minimal kemunculan satu karakter WP dalam *parsing* dengan diikuti salah satu *tag* PoS yang menghubungkannya dalam membangun frasa seperti <NN|JJ|NNP|NNG|PRP|DT|RB|MD|NEG|RP>* dengan kemunculan 0 atau N karakter. Contoh implementasi ketentuan *rule* ini dapat dilihat di gambar 2.5 :



Gambar 2.5 Contoh dari penerapan *rule* 1 dari Penelitian M.Iqbal Kamiluddin (2017)

b. *Rule* 9 VP : {<VP><KP>}

Frasa VP terbentuk jika terdapat kata yang mengandung kata hubung, depan dan kata keterangan dari frasa KP (*Connection phrases*) berada setelah frasa VP yang sudah terbentuk dari *rule* 5 yakni *rule* 5 VP : {<VBI|VBT>+ <PRP|DT>* |<PRP>* <VBI|VBT>+ |} dan saling terkoneksi membentuk satu-kesatuan frasa. Contoh implementasi ketentuan *rule* ini seperti pada gambar 2.6 :



Gambar 2.6 Contoh dari penerapan *rule* 9 dari penelitian M.Iqbal Kamiluddin (2017)

2.7 *Rule* Jeda

Rule jeda adalah *rule* atau aturan yang digunakan digunakan untuk memberi label jeda 1 (jeda singkat) dan jeda 2 (jeda panjang) pada pertemuan tiap potongan frasa hasil *shallow parsing*. *Rule* jeda dibentuk berdasarkan hasil analisa kejadian jeda yang paling sering terjadi antara dua penggalan frasa. Sebagai contoh :

- 1 (VP (KP telah/MD) (VP meresmikan/VBT))
- 2 (NP jembatan/NN Tayan/NNP) ,
- 3 (NP panjang/JJ itu/DT) ,

Jika dimasukkan dalam *rule* jeda yang berbunyi VP-NP = jeda 1, maka *output* kalimat yang dihasilkan menjadi “telah meresmikan/1 jembatan tayan”. Jika

dimasukkan dalam *rule* jeda yang berbunyi NP-NP = jeda 2, maka *output* kalimat yang dihasilkan menjadi “jembatan tayan/2 panjang itu”.

2.8 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. Untuk mengetahui jumlah pasti penutur bahasa ini memang sulit didapat, karena tidak ada data yang pasti mengenai jumlah penutur (Indrayana, 2016).

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 langsung Bang Jul ni ikut jadi duta wisata Kalbar. Maklom, Dinas Pariwisata di tempat kite ni tadak bise bekerja maksimal, jadi kalok ade orang – orang barat datang, bukan-e pegi ke pusat inpormasi turis, tapi pegi ke tempat – tempat macam Bang Jul ni.”

2.9 Frasa

Frasa adalah gabungan atau kesatuan kata yang terbentuk dari dua kelompok kata atau lebih yang memiliki satu makna gramatikal (makna yang berubah-ubah menyesuaikan dengan konteks).

Berdasarkan jenisnya, frasa dapat digolongkan menjadi frasa verbal, frasa nominal, frasa numerik, frasa ajektiva, dan frasa preposisional.

a. Frasa Verbal

Frasa verbal adalah frasa yang mempunyai inti kata kerja sebagai pengganti kedudukan kata kerja dalam kalimat.

Contoh pada kalimat : dua orang mahasiswa sedang membaca buku di perpustakaan.

Frasa membaca termasuk golongan frasa verbal.

b. Frasa Nominal

Frasa Nominal adalah frasa yang mempunyai inti kata benda sebagai pengganti

kedudukan kata benda dalam kalimat.

Contoh pada kalimat : Ia membeli baju baru yang cantik.

Frasa baju termasuk golongan frasa nominal.

c. Frasa Numerik

Frasa Numerik adalah frasa yang mempunyai inti berupa numerik atau bilangan.

Misalnya "dua buah buku baru" yang memiliki frasa dua buah.

d. Frasa Adjektiva

Frasa adjektiva adalah frasa yang mempunyai inti berupa kata sifat. Misalnya frasa hancur lebur dalam kalimat "Beberapa butir telur itu hancur lebur setelah terjatuh dan terlindas sepeda motor".

e. Frasa Preposisional

Frasa yang diawali oleh preposisi sebagai penanda dan diikuti oleh kata/frasa kategori nomina, verba, numerik atau Ket. sebagai penanda atau aksinya.

Contoh : di sebuah rumah.

Frasa di sebuah rumah terdiri atas preposisi di sebagai penanda, diikuti frasa nomina sebagai penanda.

2.10 Pengujian *Precision & Recall*

Imam Fahrur Rozi (2012) menjelaskan bahwa, *precision* adalah rasio jumlah dokumen relevan yang ditemukan dengan total jumlah dokumen yang ditemukan oleh sistem. *Recall* adalah rasio jumlah dokumen relevan yang ditemukan kembali dengan total jumlah dokumen dalam kumpulan dokumen yang dianggap relevan. *Precision* dapat dianggap sebagai ukuran ketepatan atau ketelitian, sedangkan *recall* adalah kesempurnaan. Persamaan untuk *precision* ditunjukkan pada persamaan (2.1) dan *recall* pada persamaan (2.2).

Tabel 2.4 Variabel Untuk Perhitungan *Precision* dan *Recall*

	Frasa Jeda Yang Relevan	Frasa Jeda Tidak Relevan
Ditemukan	Terambil (a)	Terambil (c)
Tidak ditemukan	Tidak Terambil (b)	Tidak Terambil (d)

$$\text{Precision} = \frac{\text{item yang relevan yang ditemukan}}{\text{item ditemukan}} = P(\text{relevan/ditemukan}) \quad (2.1)$$

$$\text{Recall} = \frac{\text{item yang relevan yang ditemukan}}{\text{semua item yang relevan}} = P(\text{ditemukan/relevan}) \quad (2.2)$$

Berdasarkan Tabel 2.4, persamaan (2.1) dan (2.2) dapat disederhanakan menjadi :

$$\text{Precision} = \frac{a}{a+c} \quad (2.3)$$

$$\text{Recall} = \frac{a}{a+b} \quad (2.4)$$

Pengujian *recall* dan *precision* digunakan untuk mengetahui nilai akurasi keberhasilan *rule grammars* yang dibuat dalam menghasilkan potongan frasa jeda. Pengujian ini juga digunakan sebagai nilai acuan untuk pengujian frasa jeda terhadap frasa jeda yang didapat dari penutur apakah bisa menghasilkan frasa jeda yang baik sesuai penutur Bahasa Melayu Pontianak.

2.11 Pengujian F-measure

F-measure (nilai F atau *FI-Score* atau *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. Nilai F dapat diartikan sebagai sebuah hasil rata-rata dari *precision* & *recall*, dimana sebuah nilai F mendapat hasil 1 sebaga hasil terbaik dan 0 sebagai nilai terburuk.

Rumus nilai F adalah sebagai berikut:

$$\text{F-measure} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \quad (2.5)$$

2.12 Pengujian Akurasi Jeda dalam Satu Kalimat

Pengujian ini dilakukan untuk membandingkan hasil kecocokan frasa jeda yang dibentuk sistem dengan frasa jeda kalimat asli dari penutur bahasa Melayu Pontianak. Pengujian dilakukan untuk mendapatkan nilai akurasi frasa jeda yang

terbentuk dari *shallow* parsing yang sama dengan frasa jeda penutur. Rumus yang digunakan untuk pengujian ini adalah :

$$\text{Akurasi frasa jeda dalam kalimat} = \frac{\text{jumlah kalimat yang sesuai}}{\text{Total kalimat yang diuji}} \times 100\% \quad (2.6)$$

BAB III

METODOLOGI PENELITIAN

3.1 Bahan Penelitian

Bahan penelitian berupa *file* suara berformat *.wav hasil rekaman dari teks yang dikumpulkan dari buku Sepok Satu karangan Pay Jarot Sujarwo. Kumpulan teks buku Sepok Satu selanjutnya diolah menjadi korpus teks paralel bahasa Melayu Pontianak dan Bahasa Indonesia yang akan digunakan untuk proses *shallow parsing* dan kalimat teks sudah ditandai jedanya oleh penutur Bahasa Melayu Pontianak.

3.2 Perangkat Penelitian

Perangkat penelitian yang digunakan berupa alat penelitian, perangkat keras, dan perangkat lunak. Adapun penjabaran perangkat penelitian yang dipergunakan adalah sebagai berikut:

3.2.1 Perangkat Keras

Perangkat keras yang digunakan dalam penelitian ini adalah:

- a. Lenovo Ideapad S410-p dengan spesifikasi *Intel Core i5-4200U*
- b. Harddisk 500 GB
- c. Kapasitas RAM 4 GB
- d. LCD Monitor 14 inchi
- e. Headphone

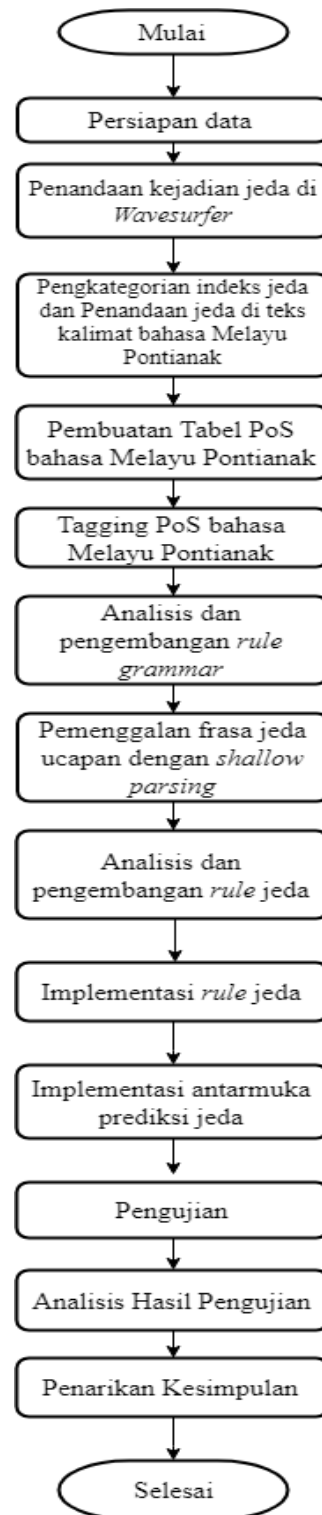
3.2.2 Perangkat Lunak

Perangkat lunak yang digunakan dalam penelitian ini adalah:

- a. Sistem Operasi Windows 10.0
- b. Python 3.6.12
- c. PHP 7.1.14
- d. NLTK untuk proses *Shallow parsing*.
- e. WaveSurfer sebagai apliasi untuk menandai kejadian jeda pada *file* rekaman suara.
- f. IPOSTagger sebagai *tools* yang digunakan untuk *tagging* PoS.
- g. Sublime Text 3 sebagai aplikasi untuk teks *editor*.

3.3 Metode Penelitian

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



Gambar 3.1 Diagram Alir Penelitian

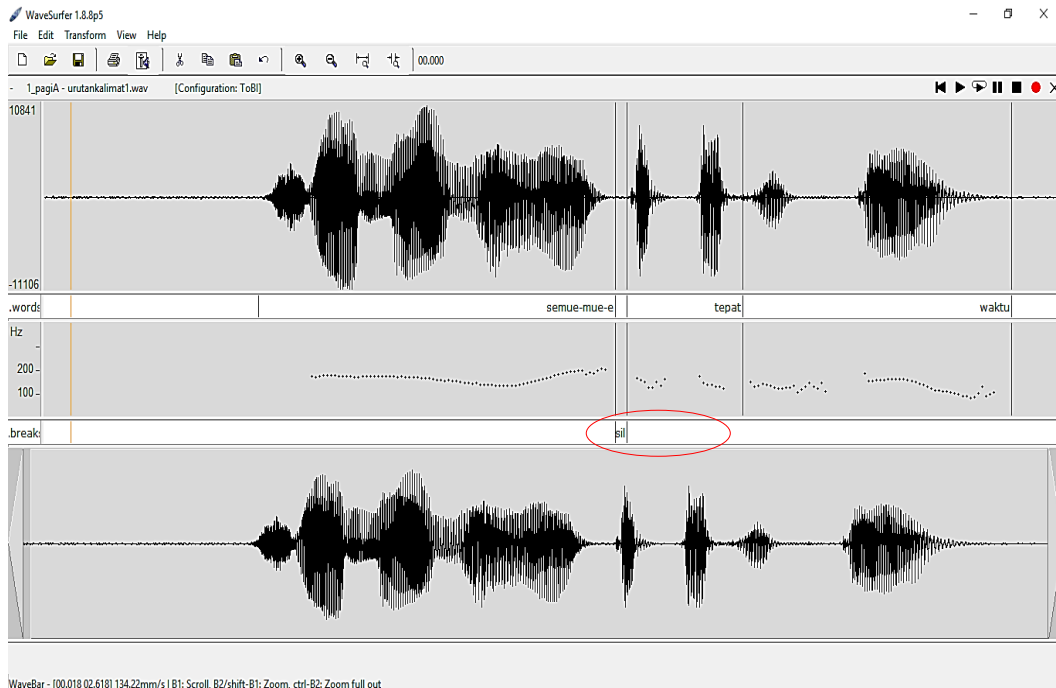
Seperti diagram alir penelitian yang terlihat pada gambar 3.1, terdapat beberapa tahapan yang dilakukan pada penelitian, diantaranya dimulai dari persiapan data berupa *file* teks dan *file* rekaman suara penutur bahasa Melayu Pontianak yang berjumlah 500 kalimat, penandaan kejadian jeda di *Wavesurfer*, pengkategorian indeks jeda dan penandaan jeda di teks kalimat bahasa Melayu Pontianak, pengembangan PoS bahasa Melayu Pontianak, melakukan *tagging* PoS bahasa Melayu Pontianak, analisis dan pengembangan *rule grammar*, pemenggalan frasa jeda dengan *shallow parsing*, analisis dan pengembangan *rule* jeda, implementasi *rule* jeda, melakukan pengujian, analisa hasil pengujian dan yang terakhir melakukan penarikan kesimpulan.

3.3.1 Persiapan Data

Persiapan data berupa teks kalimat bahasa Melayu Pontianak yang bersumber dari buku Sepok 1 karangan Pay Jarrot Sujarwo. Jumlah kalimat yang digunakan pada penelitian sebanyak 500 kalimat dan direkam oleh narasumber. Rekaman yang dihasilkan disimpan dalam *file* suara berformat .wav dengan resolusi 16bit dan *sampling rate* 44100 Hz. Narasumber merupakan penutur asli bahasa Melayu dialek Pontianak. Teks kalimat digunakan untuk penandaan PoS atau label kata dan rekaman suara ditandai jeda menggunakan bantuan *wavesurfer*.

3.3.2 Penandaan Kejadian Jeda di *Wavesurfer*

File suara yang telah disiapkan kemudian diolah menggunakan aplikasi WaveSurfer untuk ditandai fonem dan per kejadian jeda. Setiap kejadian jeda diberi tanda “sil”. “Sil” atau durasi jeda yang dihasilkan disimpan pada *file* dengan format *.*breaks*.



Gambar 3.2 Proses penandaan jeda/SIL di Wavesurfer

Pada gambar 3.2 yang merupakan proses penandaan jeda/SIL di Wavesurfer, dapat dilihat jeda ditandai dengan “|sil|” pada bagian “break”. Tanda “|” sebelum <sil> menandakan jeda awal, sedangkan tanda “|” setelah <sil> menandakan jeda akhir. Hasil dari “break” diolah untuk dijadikan durasi jeda pada indeks jeda.

3.3.3 Pengkategorian Indeks Jeda dan Penandaan Jeda di Teks Kalimat bahasa Melayu Pontianak

Setelah semua file suara ditandai, selanjutnya setiap kejadian jeda dikategorikan jeda menggunakan indeks jeda. Penelitian ini menggunakan 5 kategori indeks jeda, yaitu “0” untuk indeks yang menyatakan tidak ada jeda, “1” untuk indeks yang menyatakan terjadinya jeda singkat, “2” untuk indeks yang menyatakan terjadinya jeda panjang, “,” untuk indeks yang menyatakan tanda baca koma, dan “.” untuk indeks yang menyatakan akhir kalimat.

Tabel 3.1 Kategori Indeks Jeda

Indeks Jeda	Keterangan
0	Tidak ada jeda
1	Jeda yang terjadi tipis
2	Jeda yang terjadi tebal
,	Tanda baca koma (,)
.	Akhir kalimat

Indeks jeda pada tabel 3.1 , selanjutnya akan ditentukan lama durasi jeda yang terjadi. Perhitungan durasi jeda didapatkan dengan cara menghitung nilai *mean* atau rata-rata, modus, nilai minimum, dan nilai maksimum dari kejadian jeda di file *.breaks yang telah diurutkan dari nilai yang terkecil hingga terbesar dengan menggunakan bantuan *spreadsheet* (Nugraha, 2014).

Tabel 3.2 Tabel Proses Perhitungan Durasi Jeda

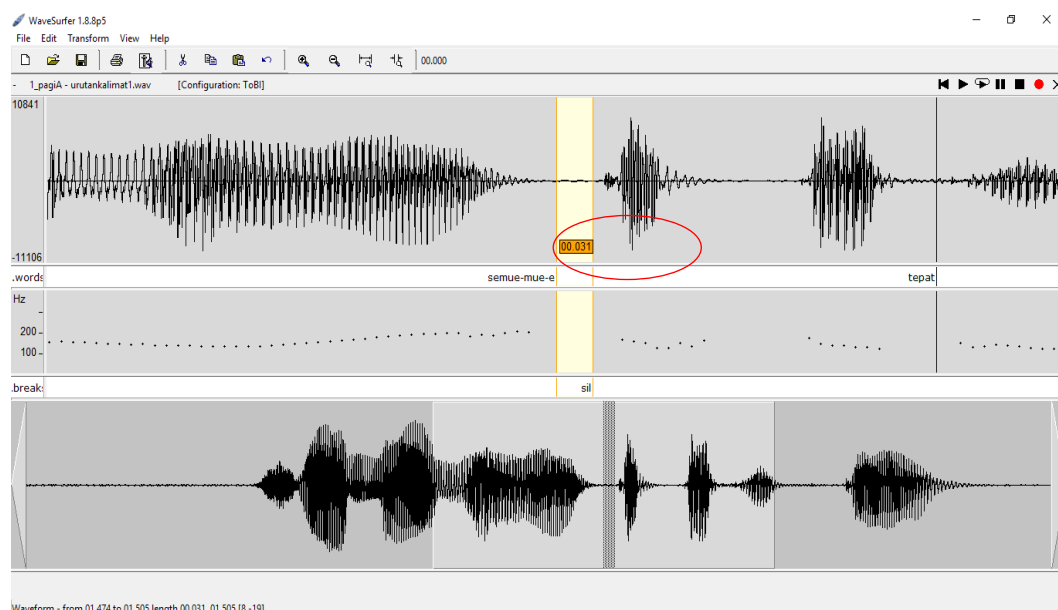
Jeda awal	Jeda Akhir	Durasi Jeda	Jumlah Kemunculan
...

Durasi jeda didapat dengan mengurangkan nilai jeda akhir dan jeda awal. Nilai di kolom durasi jeda dihitung untuk mendapatkan nilai MIN, MAX, Median dan Mean atau rata-rata. Sedangkan nilai di kolom jumlah kemunculan digunakan sebagai nilai modus. Nilai tersebut digunakan sebagai nilai durasi untuk indeks jeda sehingga hasilnya dapat dilihat pada tabel 3.3.

Tabel 3.3 Kategori Indeks Jeda dan Durasinya

Indeks Jeda	Keterangan	Durasi Jeda
0	Tidak ada jeda	0 - < nilai modus
1	Jeda yang terjadi tipis	nilai modus - <= nilai <i>mean</i>
2	Jeda yang terjadi tebal	> nilai <i>mean</i>
,	Tanda baca koma (,)	,
.	Akhir kalimat	.!?

Indeks jeda 0 memiliki durasi jeda dari nol detik sampai kurang dari nilai modus. Indeks jeda 1 atau jeda singkat memiliki durasi jeda dari nilai modus sampai kurang dari sama dengan nilai mean. Sedangkan untuk indeks jeda 2 atau jeda panjang memiliki durasi jeda lebih dari nilai *mean*. Kategori indeks jeda singkat dan panjang beserta durasi yang telah didapat kemudian digunakan untuk menandai kejadian jeda di korpus bahasa Melayu Pontianak. Penandaan jeda dilakukan dengan membuka kembali *wavesurfer* dan *file* rekaman suara yang sudah ditandai dan mengecek durasi jeda “sil” yang telah ditandai seperti pada gambar 3.3.

**Gambar 3.3** Proses pengecekan durasi jeda/SIL di Wavesurfer

Misalkan pada kalimat : “Semue-mue-e tepat waktu”. Pada proses penandaan “sil” di *wavesurfer* , lama durasi “sil” yang ditunjukkan di lingkaran merah pada gambar 3.3 sebesar 0.031 . Nilai ini dicek berdasarkan tabel hasil

perhitungan durasi indeks jeda di tabel 3.3 untuk menentukan nilai indeks jeda yang mana yang digunakan untuk penandaan jeda.

3.3.4 Pembuatan Tabel PoS bahasa Melayu Pontianak

Pada tahap ini digunakan set PoS yang baru yang dibentuk berdasarkan Set PoS Bahasa Melayu Pontianak. Pengelompokkan kelas kata dalam Bahasa Melayu dilakukan secara manual dengan merujuk pada referensi set PoS pada penelitian M.Iqbal Kamiludin (2017), dan beberapa referensi set PoS yang ada yakni dari Set Pos Setyaningsih (2017) dan Set PoS Adriani (2009). *Set* PoS baru dibuat dengan tujuan untuk memperjelas penandaan tipe kata dalam kalimat dan meningkatkan akurasi prediksi jeda. Set PoS ini digunakan pada proses pengembangan *rule grammar* yang digunakan pada proses *shallow parsing* pada tahap selanjutnya.

Pembuatan tabel PoS bahasa Melayu Pontianak dilakukan dengan cara menganalisa tabel PoS referensi yang ada dan mencocokkan label PoS yang sesuai dengan kelas kata yang digunakan pada bahasa Melayu Pontianak. Penambahan set PoS disesuaikan dengan fungsi kata dalam kalimat bahasa Melayu Pontianak. Misalkan, pada penelitian M.Iqbal Kamiludin, kata “si” belum memiliki label PoS yang tepat yaitu NN. Pada penelitian Setyaningsih (2017) , terdapat label PoS AR atau artikel untuk panggilan seperti “si”, “sang”, atau “Yang”, sehingga PoS AR diambil dan ditambahkan ke tabel PoS yang baru. Untuk kata keterangan atau *adverb*, pada penelitian sebelumnya hanya digunakan PoS RB menandai kata jarang, sementara, dan agak. Ketiga kata ini sebenarnya memiliki keterangan yang berbeda dan bisa dikelompokkan menjadi PoS baru untuk memperjelas tipe kelas kata agar pengembangan *rule grammar* lebih mudah. Ketiga kata dapat dikembangkan menjadi PoS RB atau kata keterangan umum untuk kata “sementara”. Kata “jarang” menandai keterangan waktu seseorang melakukan kegiatan sehingga dibuat PoS FRB atau kata keterangan untuk kata yang mendeskripsikan seberapa sering suatu kejadian terjadi atau dilakukan. Kata “agak” menunjukkan keterangan perkiraan berapa lama suatu kegiatan sehingga dibuat PoS DRB atau kata keterangan tingkatan. Hasil dari pembuatan tabel PoS adalah tabel set PoS baru yang digunakan untuk menandai kata di korpus teks bahasa Melayu Pontianak.

3.3.5 *Tagging PoS bahasa Melayu Pontianak*

Pada dasarnya, IPOSTagger menggunakan korpus Bahasa Indonesia. Maka dari itu, korpus yang ada harus diganti dengan korpus Bahasa Melayu Pontianak. Teks kalimat untuk korpus latih IPOSTagger yang berasal dari buku Sepok Satu 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 “UH” 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 “VBI” 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 seperti “kau”, “aku” , “kamek”. Hasil penandaan PoS menjadi kalimat yang berisi “kata/PoS” seperti berikut:

“Eh/UH ./, jangan/NEG banyak/JJ umong/VBI kau/PRP ./.”

3.3.5.1 *Pelatihan Korpus di IPOSTagger*

Setelah korpus latih selesai dibuat, selanjutnya dilakukan proses pelatihan korpus pada IPOSTagger untuk keperluan implementasi prediksi jeda. 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 *Analisis dan Pengembangan Rule Grammar Bahasa Melayu Pontianak*

Setelah melakukan *Tagging* PoS pada kalimat Bahasa Melayu Pontianak, selanjutnya kalimat Bahasa Melayu Pontianak dianalisis untuk mendapatkan *rule grammar* ucapan yang diperlukan dalam melakukan proses *shallow parsing*.

Pengembangan *rule grammar* dilakukan untuk mendapatkan frasa jeda ucapan. Frasa ucapan ini didapat dari penandaan jeda di *wavesurfer*. Ada beberapa *rule* yang diubah dan ditambah, berdasarkan pelabelan PoS yang ditambah dan analisis kejadian jeda.

Analisis untuk pembuatan *rule* dilakukan dengan membentuk *rule grammar* berdasarkan kejadian jeda di tiap frasa dalam kalimat yang sudah ditandai indeks jeda sesuai dengan kejadian jeda dari penutur. Frasa yang dipakai adalah frasa yang sudah dilabel dengan kelas kata sesuai dengan tabel kelas kata yang sudah dibuat sebelumnya. Frasa yang terbentuk nantinya berupa frasa ucapan sehingga *rule* yang dibuat berdasarkan analisa frasa ucapan dari penutur dan tidak berdasarkan struktur kalimat S-P-O-K-dan Pel. Dari deretan frasa dengan kelas kata ini dibuat *rule* dengan bantuan karakter *Regular Expression* untuk menentukan kemunculan karakter/kata dalam tiap frasa. Berikut karakter *Regular Expression* yang digunakan dalam *shallow parsing* pada tabel 3.4.

Tabel 3.4 Makna Karakter *Regular Expression*

Karakter	Makna Karakter <i>Regular Expression</i>
<>	Penentuan <i>tag part-of-speech</i>
?	Nol atau salah satu item sebelumnya
*	Nol atau lebih dari satu item sebelumnya
+	Satu atau lebih dari item sebelumnya
	Mencocokkan satu item dengan yang lainnya

Dengan memanfaatkan karakter *regular expression* akan dibentuk *rule* untuk membantu proses *shallow parsing* dalam memeriksa dan mencocokkan urutan kata dalam kalimat dalam membentuk potongan frasa jeda yang ingin dibentuk. *Shallow parsing* akan membentuk potongan frasa ke dalam tipe frasa.

3.3.6.1 Pengembangan Tipe Frasa

Jenis tipe frasa yang digunakan dalam penelitian ini diambil dari penelitian yang dilakukan oleh Arry Akhmad Arman (2013), dan ditambah dengan satu tipe frasa baru yakni AP (frasa keterangan). Tujuannya untuk memudahkan

pengembangan *rule grammar* agar dapat menghasilkan frasa jeda ucapan sesuai ucapan penutur.

Berikut tipe frasa yang digunakan pada penelitian ini pada tabel 3.5.

Tabel 3.5 Tabel Tipe Frasa

Tipe Frasa	PoS Tag
<i>Questioning Phrases</i> (TP)	WP, WPRB, WDT
<i>Numeric Phrases</i> (BP)	CDP, US, CDI, CDO, CDF, CDA, CDC, SYM, PRN
<i>Noun Phrases</i> (NP)	NNP, UH, FW, PRP, NNP, NNG, NNC, NNU, NN, JJ, RP
<i>Connection Phrases</i> (KP)	CON, RB, MD, NEG
<i>Verb Phrases</i> (VP)	VBK, VBT, VBR, VBI
<i>Adverb Phrases</i> (AP)	IN, DRB, PRL, FRB,

3.3.6.2 Pengembangan Rule Grammar

Pengembangan *rule grammar* dibuat dengan menganalisis teks kalimat bahasa Melayu Pontianak yang sudah diberi PoS dan kejadian jeda yang terjadi berdasarkan hasil *tagging* di *wavesurfer*. Referensi *rule grammar* dari penelitian sebelumnya yaitu penelitian M.Iqbal Kamiludin (2017) . *Rule grammar* ada yang ditambah dan diubah. Proses pembuatan *rule grammar* dengan melihat kejadian frasa jeda ucapan yang terbentuk berdasarkan indeks jeda. Misalkan pada kalimat :

“Semue-mue-e/1 tepat waktu.”

Pada kalimat “semue-mue-e tepat waktu”, jeda singkat muncul setelah kata “semue-mue-e” dan kata “tepat”. Selanjutnya *rule grammar* dibuat dengan bantuan label PoS , *regular expression*, dan tipe frasa yang dibentuk seperti pada tabel 3.3. Kalimat tersebut jika diberi label PoS menjadi:

“Semue-mue-e/PRN tepat/DRB waktu/NNU ./.”

Label kata PRN menunjukkan keterangan jumlah sehingga dimasukkan ke dalam frasa BP atau frasa numerik dan dibuat menjadi *rule* yang berbunyi :

BP : { <PRN>+ }

yang menunjukkan frasa BP mengandung kata yang memiliki label PoS PRN dan label PoS ini pasti menjadi label PoS BP karena ditandai dengan keterangan *regular expression* “+” yang memiliki makna harus muncul minimal satu atau lebih dari satu item sebelumnya.

Selanjutnya, kata “tepat waktu” yang muncul setelah jeda memiliki makna keterangan waktu sehingga dikelompokkan menjadi satu frasa dengan tipe AP (*adverb phrases*) dan dibuat mejadi *rule* yang berbunyi :

$$AP : \{ <DRB>+ <NNU>+ \}$$

yang menunjukkan frasa AP mengandung kata yang memiliki label PoS DRB dan label PoS NNU secara berurutan dan label PoS ini akan menjadi frasa AP karena ditandai dengan keterangan *regular expression* “+” yang memiliki makna harus muncul minimal satu atau lebih dari satu item sebelumnya.

Misalkan pada kalimat : “Orang pon/1 semaken banyak.”

Pada kalimat “ekonomi pon semaken betumboh”, jeda singkat muncul dan membagi kelaimat menjadi frasa “Ekonomi pon” dan :”semaken betumboh”. Selanjutnya *rule grammar* dibuat dengan bantuan label PoS , *regular expression*, dan tipe frasa yang dibentuk seperti pada tabel 3.3. Kalimat tersebut jika diberi label PoS menjadi:

“Orang/NNC pon/RP semaken/DRB banyak./.”

Pada frasa “orang pon”, menggunakan pos NNC dan RP. PoS NNC dikategorikan menjadi frasa NP seperti pada tabel 3.3 sehingga dibuat *rule* yang berbunyi :

$$NP : \{ <NNC>+ <RP>* \}$$

yang menunjukkan frasa NP mengandung kata yang memiliki label PoS NNC yang merupakan pembentuk utama frasa NP dan label PoS RP yang merupakan partikel atau kata penegas objek yang berada didepan kata partikel diberi tanda * karena dapat muncul minimal nol atau lebih dari satu item. Frasa NP dapat terbentuk jika tidak diikuti PoS RP , misalnya “Orang semaken banyak”.

Selanjutnyanya, kata “semaken banyak ” yang muncul setelah jeda memiliki makna keterangan sehingga dikelompokkan menjadi satu frasa dengan tipe AP (*adverb phrases*) seperti pada contoh sebelumnya dan dibuat mejadi *rule* yang berbunyi :

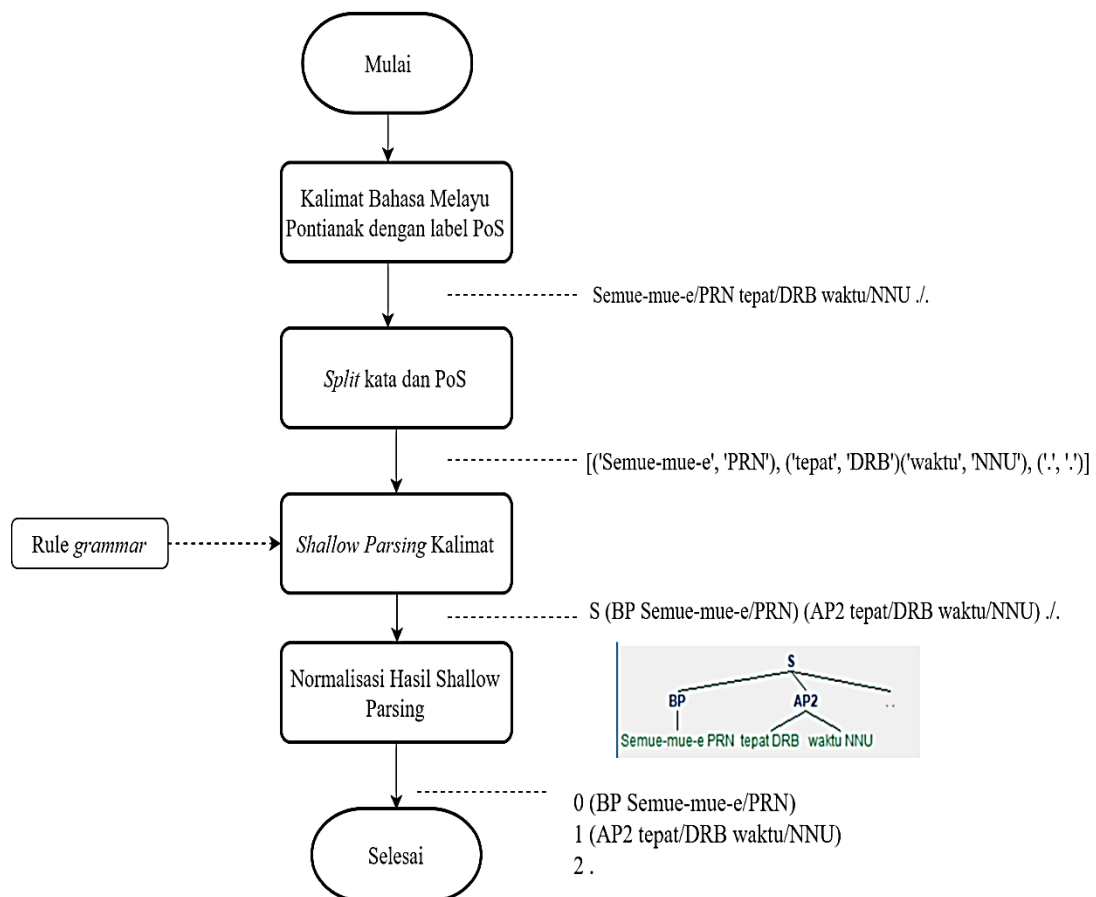
$$AP : \{ <DRB>+ <NNU|JJ>+ \}$$

yang menunjukkan frasa AP mengandung kata yang memiliki label PoS DRB dan label PoS NNU|JJ secara berurutan dan label PoS ini akan menjadi frasa AP karena ditandai dengan keterangan *regular expression* “+” yang memiliki makna harus muncul minimal satu atau

lebih dari satu item sebelumnya. <NNUJJ>+ berarti karakter NNU dan muncul bersamaan dengan JJ , atau karakter NNU dan JJ dapat berdiri sendiri.

3.3.7 Pemenggalan Frasa Jeda Ucapan dengan *Shallow Parsing*

Tahap yang selanjutnya dilakukan adalah melakukan pemenggalan frasa jeda menggunakan *rule grammar* yang telah dibuat. Cara pemenggalan frasa jeda, yakni dengan memproses kalimat inputan menjadi penggalan frasa sesuai *rule grammar* dengan bantuan fungsi *RegexParser* pada NLTK. Diagram alir proses pemenggalan frasa jeda ucapan dengan *shallow parsing* dapat dilihat pada gambar 3.4.



Gambar 3.4 Diagram Alir Pemenggalan Jeda dengan *Shallow Parsing*

Tahapan proses pemenggalan jeda dengan *shallow parsing* pada gambar 3.4 terdiri dari membaca file kalimat bahasa Melayu Pontianak, indentifikasi label PoS kata, *shallow parsing* kalimat, serta normalisasi hasil *shallow parsing*.

3.3.7.1 Kalimat bahasa Melayu Pontianak dengan Label PoS

Proses yang pertama kali dilakukan dalam pemenggalan frasa jeda dengan *shallow parsing* adalah membaca *file* teks berupa teks kalimat bahasa Melayu Pontianak yang sudah diberi label PoS dan disimpan dalam format *.txt.

3.3.7.2 Split Kata dan PoS

Selanjutnya, kalimat akan dibagi menjadi blok kata per kata dan dipisah berdasarkan label PoS. Tahap ini dilakukan untuk mempermudah pengecekan *rule grammar* berdasarkan label PoS.

3.3.7.3 Shallow Parsing Kalimat

Kalimat yang sudah dipisah per kata dan PoS , selanjutnya diperiksa berdasarkan *rule grammar* yang telah dibuat dan dipenggal menjadi penggalan frasa jeda ucapan.

3.3.7.4 Normalisasi Hasil Shallow Parsing

Output dari proses *shallow parsing* adalah penggalan frasa dari kalimat teks bahasa Melayu Pontianak sesuai *rule grammar* yang dibuat yang disimpan dalam file format *.txt dan ditampilkan per baris penggalan frasa dan tanda baca.

3.3.8 Analisis dan Pengembangan Rule Jeda

Tahapan selanjutnya adalah analisis penggalan frasa hasil *shallow parsing* untuk mendapatkan tipe jeda. Tipe jeda didapatkan dengan melihat kejadian jeda penutur dengan penggalan frasa hasil *shallow parsing*. Tipe jeda terdiri dari dua jeda singkat (/1) dan jeda panjang (/2).

index	KORPUS KATA-POS 500.txt	Maret\penjelasan	chunktes.t
1	0 (BP Semue-mue-e/PRN)		
2	1 (AP2 tepat/DRB waktu/NNU)		
3	2 .	→	Kalimat
4			
5	0 (VP Kau/PRP bikin/VBT)		Pertama
6	1 (NP janji/NN jam/NN (BP limak/CDP))		
7	2 ,		
8	3 (KP make/CON)		
9	4 (NP jam/NN (BP limak/CDP))		
10	5 (NP1 kau/PRP)	→	Kalimat Kedua
11	6 (KP haros/MD)		
12	7 (VP1 datang/VBI)		
13	8 .		
14			
15	0 (VP Ade/VBT kegiatan/NN jam/NN)		
16	1 (BP tujuh/CDP)		
17	2 ,		
18	3 (KP make/CON)		
19	4 (NP jam/NN (BP tujuh/CDP))		
20	5 (NP1 kegiatan/NN tu/DT)		
21	6 (KP haros/MD)	→	Kalimat Ketiga
22	7 (VP1 dimulai/VBI)		
23	8 .		

Gambar 3.5 Contoh Frasa Hasil *Shallow Parsing*

Frasa hasil *shallow parsing* kemudian ditandai kemunculan kejadian jeda yang terjadi antara penggalan frasa sesuai korpus jeda dari penutur. Pengelompokan kemunculan kejadian jeda dibagi menjadi dua yaitu jeda singkat dan jeda panjang. Misalkan pada kalimat asli jeda dari penutur :

1. Semue-mue-e/1 tepat waktu .
2. Kau bikin janji jam limak/2 , make/1 jam limak/1 kau haros datang .
3. Ade kegiatan jam tujuh/2 , make/1 jam tujuh/1 kegiatan tu haros dimulai .

Kemudian, frasa hasil *shallow parsing* seperti pada contoh gambar 3.5 dikategorikan menjadi dua yaitu frasa yang menghasilkan jeda singkat (1) dan jeda panjang(2).

Tabel 3.6 Kemunculan Jeda pada Frasa Hasil Shallow Parsing

Jeda 1	Jumlah Kemunculan	Jeda 2	Jumlah Kemunculan
BP-AP2	1	NP-,	1
KP-NP	2	BP-,	1
NP-NP1	2	KP-NP	1

Frasa BP dan AP2 pada kalimat pertama gambar 3.5, berpotongan pada jeda singkat (1) sehingga dimasukkan ke dalam kolom jeda 1 pada tabel 3.6 . Pada kalimat kedua gambar 3.5, frasa NP diikuti jeda panjang (2) jika bertemu tanda

koma, sehingga dimasukkan ke kolom jeda 2.

Kolom jumlah kemunculan untuk menghitung frekuensi kemunculan jeda berdasarkan tipe frasa. Misalnya frasa KP-NP dari kalimat pertama sampai ketiga, dipenggal berdasarkan jeda singkat sebanyak dua kali sehingga jumlah kemunculan frasa KP-NP menjadi dua. Penggalan frasa di kolom jeda 1 dan jeda 2 yang akan dijadikan *rule* jeda untuk proses prediksi jeda kalimat bahasa Melayu Pontianak.

Jika terdapat kalimat yang mengandung tipe frasa yang sama tetapi dipenggal berdasarkan jeda yang berbeda, misalnya berdasarkan data jeda dari penutur ada frasa KP-NP yang dipenggal berdasarkan jeda panjang sebanyak satu kali, maka yang diambil untuk dijadikan *rule* jeda adalah data penggalan frasa dengan jumlah kemunculan terbanyak.

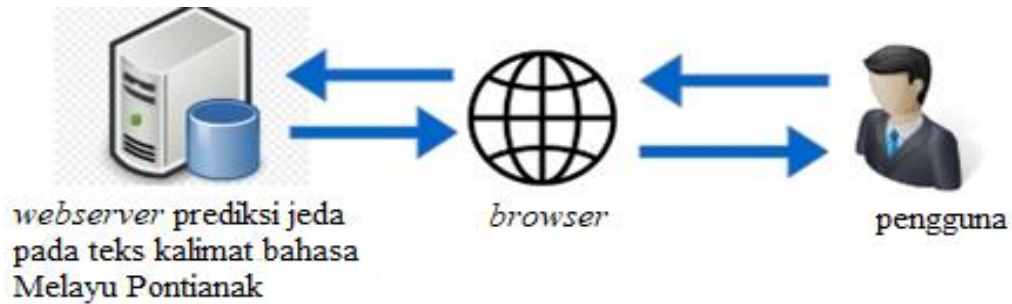
3.3.9 Implementasi *Rule* Jeda

Selanjutnya, indeks jeda ditandai berdasarkan *rule* jeda yang dibuat pada tahap sebelumnya. Pada proses ini, frasa hasil shallow parsing diperiksa menggunakan *rule* jeda. Jika frasa yang terbentuk sesuai dengan *rule* jeda untuk kejadian jeda singkat (1) , maka teks yang keluar akan diberi tanda “/1” untuk keterangan kemunculan jeda singkat. Jika frasa yang terbentuk sesuai dengan *rule* jeda untuk kejadian jeda panjang(2) , maka teks yang keluar akan diberi tanda “/2” untuk keterangan kemunculan jeda panjang. Hasil dari proses ini adalah teks kalimat bahasa Melayu Pontianak beserta tanda jedanya. Misalnya :

1. “Semue-mue-e/1 tepat waktu .”
2. “Kau bikin janji jam limak/2 , make/1 jam limak/1 kau haros datang .”

3.3.10 Implementasi Antarmuka Prediksi Jeda

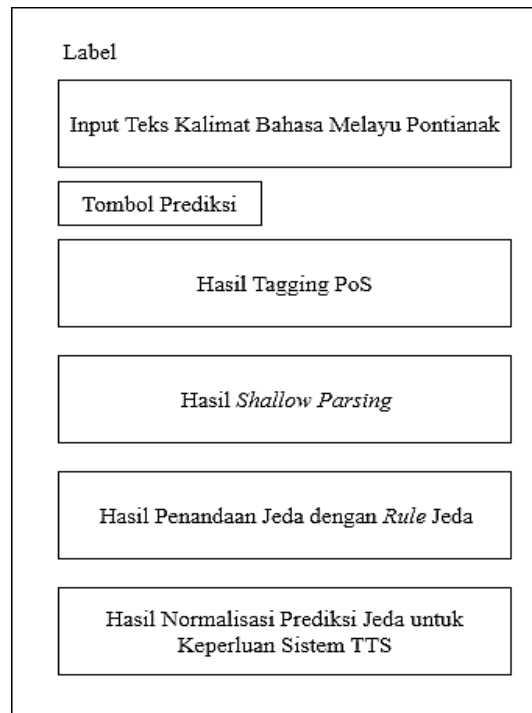
Prediksi jeda dimulai dengan memasukkan teks kalimat Bahasa Melayu Pontianak yang kemudian melewati beberapa proses hingga menghasilkan keluaran berupa teks kalimat Bahasa Melayu Pontianak beserta prediksi jedanya. Gambaran umum sistem prediksi jeda dapat dilihat dari rancangan arsitektur sistem untuk menjelaskan hubungan antar komponen yang terlibat dalam sistem yang dibangun. Arsitektur sistem yang dibuat dapat dilihat pada gambar 3.6.



Gambar 3.6 Arsitektur Prediksi Jeda

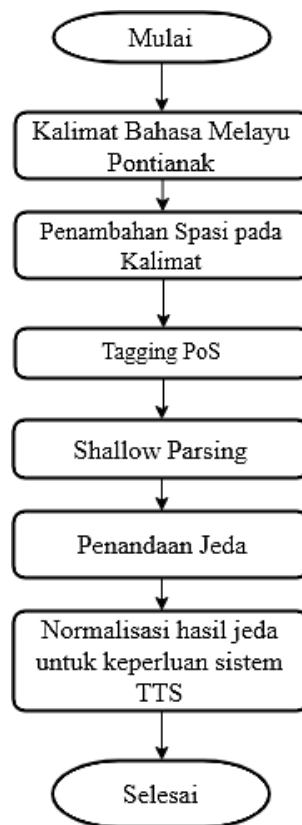
Sistem terdiri dari sebuah *webserver* dan pengguna. Pengguna merupakan orang awam atau *end-user* yang mengakses halaman *web* melalui *browser*. Sistem yang dibangun merupakan perantara untuk proses prediksi jeda yang telah dibangun sebelumnya dengan menggunakan bahasa Python. Sistem dapat menerima inputan berupa teks yang diolah melalui proses prediksi jeda yang telah dibuat. Pengguna mendapatkan *respond* berupa hasil prediksi jeda teks kalimat bahasa Melayu Pontianak yang telah diinputkan dalam bentuk teks kalimat.

Untuk halaman web dibuat rancangan antar muka sistem seperti pada gambar 3.7. Rancangan antarmuka sistem dibuat dengan tujuan agar pengguna dapat menggunakan sistem dengan baik.



Gambar 3.7 Rancangan antarmuka sistem

Antarmuka sistem prediksi jeda berupa halaman web yang berisi sebuah tombol untuk memulai proses prediksi jeda dan empat buah kolom untuk menampilkan hasil dari tiap proses prediksi jeda. Tiap proses mengikuti diagram alir proses sistem prediksi jeda yang dapat dilihat pada gambar 3.8.



Gambar 3.8 Diagram alir proses sistem prediksi jeda

Proses prediksi jeda dimulai dengan memasukkan teks kalimat Bahasa Melayu Pontianak yang akan diprediksi jedanya. Teks masukan tersebut kemudian diolah untuk mendapatkan penambahan spasi dan di *tagging* untuk mendapatkan PoS menggunakan bantuan IPOSTagger. Hasil *tagging* PoS selanjutnya diolah menggunakan *shallow parsing* untuk mendapatkan tipe frasa. Tipe frasa yang dihasilkan selanjutnya diberi penandaan indeks jeda. Kemudian hasil penandaan indeks jeda dinormalisasi menjadi *output* yang diperlukan untuk keperluan sistem TTS.

3.3.10.1 Penambahan Spasi pada Kalimat

Penambahan spasi pada kalimat merupakan proses menambahkan spasi baik sebelum dan setelah tanda baca koma dan tanda baca yang menandai akhir kalimat seperti tanda titik, tanda seru dan tanda tanya untuk mempermudah proses *tagging* PoS menggunakan IPOSTagger.

3.3.10.2 *Tagging PoS*

Tagging PoS adalah proses yang dilakukan untuk mendapatkan *tag PoS* atau kelas kata dari setiap kata yang ada pada kalimat masukan. Setelah kalimat mendapatkan penambahan spasi dan titik, kemudian ditandai untuk mendapatkan *tag PoS* dengan menggunakan aplikasi IPOSTagger yang telah dilatih sebelumnya dan hasilnya disimpan pada *file* dengan format *.hsl yang berisikan “kata/PoS”.

3.3.10.3 *Shallow Parsing*

Hasil dari *tagging PoS* kemudian diproses menggunakan metode *shallow parsing* untuk mendapatkan tipe frasa dan disimpan dalam file *.txt yang berisi hasil *chunk* dari kalimat bahasa Melayu Pontianak yang diproses.

3.3.10.4 *Penandaan Jeda*

Hasil dari *shallow parsing* kemudian diproses menggunakan *rule jeda* untuk ditandai kejadian indeks jeda singkat dan panjang. Hasil dari penandaan jeda disimpan dalam file *.txt yang berisi kalimat bahasa Melayu Pontianak yang telah ditandai jeda singkat (/1) dan jeda panjang (/2).

3.3.10.5 *Normalisasi Hasil Jeda untuk Keperluan Sistem TTS*

Hasil dari penandaan jeda kemudian dinormalisasi dengan mengubah tanda jeda singkat(/1) menjadi tanda “|” dan mengubah tanda jeda panjang(/2) menjadi tanda “||” untuk keperluan implementasi sistem *Text-to-Speech*.

3.3.11 *Perancangan Pengujian*

Pengujian dilakukan untuk mendapatkan nilai pengujian yang akan dianalisis menjadi kesimpulan penelitian. Pengujian dilakukan dengan membandingkan kalimat asli dengan kalimat hasil prediksi jeda dari hasil implementasi *rule jeda* menggunakan pengujian *Precision*, *Recall* dan *F-Score* dan pengujian akurasi kecocokan frasa jeda dalam satu kalimat. Total kalimat yang akan diuji adalah 500 kalimat bahasa Melayu Pontianak.

3.3.11.1 Pengujian Akurasi Kecocokan Frasa Jeda dalam Satu Kalimat Penuh

Pengujian ini dilakukan untuk membandingkan hasil kecocokan frasa jeda prediksi oleh sistem dengan frasa jeda kalimat asli dari penutur bahasa Melayu Pontianak. Pengujian dilakukan untuk mendapatkan nilai akurasi frasa jeda yang terbentuk yang sama dengan frasa jeda penutur. Rumus yang digunakan untuk pengujian ini adalah :

$$\text{Akurasi frasa jeda dalam kalimat} = \frac{\text{jumlah kalimat yang sesuai}}{\text{Total kalimat yang diuji}} \times 100\% \quad (3.1)$$

Tabel 3.7 Rencana Pengujian pada Frasa Jeda dalam Satu Kalimat

No.	Jeda Penutur	Jeda <i>Shallow Parsing</i>	Jeda 1+2		Jeda 2	
			Sesuai	Tidak Sesuai	Sesuai	Tidak Sesuai
			√	X	√	X
Total						

Jumlah kalimat yang diuji adalah sebanyak 500 kalimat dari frasa jeda penutur dan 500 kalimat dari frasa jeda hasil shallow parsing. Pengujian dilakukan sebanyak dua kali untuk membandingkan jeda 1+2 dan jeda 2 seperti pada tabel 3.7. Total dari kalimat yang sesuai dimasukkan ke dalam persamaan 3.1 untuk mendapatkan hasil akurasi frasa jeda dalam kalimat.

Pengujian akurasi kecocokan frasa jeda dilakukan dua kali yaitu :

3.3.11.1.1 Pengujian Akurasi Kecocokan Frasa Jeda dengan *Rule Grammar* dan *Rule Jeda Baru*

Pengujian ini dilakukan untuk melihat akurasi kecocokan frasa jeda dari hasil penerapan *rule grammar* dan *rule jeda* yang baru pada 500 kalimat yang sudah disiapkan dengan menggunakan rancangan pengujian seperti pada tabel 3.7. Pengujian dilakukan sebanyak dua kali untuk membandingkan jeda singkat dan panjang(1+2) dan jeda panjang (2).

3.3.11.1.2 Pengujian Akurasi Kecocokan Frasa Jeda dengan *Rule Grammar* dari Penelitian Sebelumnya (Kamiludin, 2017)

Pengujian ini dilakukan untuk melihat akurasi kecocokan frasa jeda dari hasil penerapan *rule grammar* yang lama terhadap 500 kalimat yang sudah disiapkan. Pengujian dilakukan sebanyak satu kali untuk menghitung jumlah kalimat yang sama di prediksi jeda singkat dan panjang(1+2). Pengujian jeda panjang tidak dilakukan karena pada penelitian sebelumnya, tidak dilakukan prediksi untuk jeda panjang, sehingga semua jeda dikategorikan sama. Hasil dari pengujian ini digunakan untuk melihat apakah *rule grammar* dan *rule* jeda yang baru bisa dikategorikan berhasil untuk meningkatkan akurasi prediksi jeda pada kalimat bahasa Melayu Pontianak.

3.3.11.2 Pengujian Recall, Precision dan F-Measure

Pengujian *Recall* dan *Precision* merupakan pengujian untuk mendapatkan informasi hasil pencarian dokumen yang relevan dengan dokumen asli yang ingin dibandingkan. *Precision* dapat dianggap sebagai ukuran ketepatan atau ketelitian, sedangkan *recall* adalah pengulangan kejadian yang sama dengan dokumen asli. Rancangan pengujian *precision* dan *recall* adalah seperti berikut :

3.3.11.2.1 Skenario Pengujian Jumlah Kalimat

Untuk melihat perbandingan hasil pengujian dibuat skenario jumlah kalimat pengujian dengan jumlah kalimat 100, 200, 300, 400, dan 500.

Tabel 3.8 Skenario Pengujian Terhadap 100 Kalimat dari 500 Kalimat Analisis

No.	Kalimat Korpus Uji	Relevan		Tidak Relevan	Precision $(\frac{a}{a+c})$	Recall $(\frac{a}{a+b})$	F- Measure $(2 \cdot \frac{p \cdot r}{p+r})$
		Teramb il (a)	Tidak Terambil (b)	Terambil(c)			
1.	0-100						
2.	101-200						
3.	201-300						
4.	301-400						

No.	Kalimat Korpus Uji	Relevan		Tidak Relevan	Precision $(\frac{a}{a+c})$	Recall $(\frac{a}{a+b})$	F- Measure $(2 \cdot \frac{p \cdot r}{p+r})$
		Teramb il (a)	Tidak Terambil (b)	Terambil(c)			
5.	401-500						

Tabel 3.9 Skenario Pengujian Terhadap 200 Kalimat dari 500 Kalimat Analisis

No.	Kalimat Korpus Uji	Relevan		Tidak Relevan	Precision $(\frac{a}{a+c})$	Recall $(\frac{a}{a+b})$	F- Measure $(2 \cdot \frac{p \cdot r}{p+r})$
		Teramb il (a)	Tidak Terambil (b)	Terambil(c)			
1.	0-200						
2.	0-100, 201-300						
3.	0-100, 301-400						
4.	0-100, 401-500						
5.	101-300						
6.	101-200, 301-400						
7.	101-200, 401-500						
8.	201-400						
9.	201-300, 401-500						
10.	301-500						

Tabel 3.10 Skenario Pengujian Terhadap 300 Kalimat dari 500 Kalimat Analisis

No.	Kalimat Korpus Uji	Relevan		Tidak Relevan	Precision $(\frac{a}{a+c})$	Recall $(\frac{a}{a+b})$	F- Measure $(2 \cdot \frac{p \cdot r}{p+r})$
		Teramb il (a)	Tidak Terambil (b)	Terambil(c)			
1.	0-300						
2.	0-200, 301-400						
3.	0-200, 401-500						
4.	0-100, 201-400						
5.	0-100, 201-300, 401-500						
6.	0-100, 301-500						
7.	101-400						
8.	101-300, 401-500						
9.	101-200, 301-500						
10.	201-500						

Tabel 3.11 Skenario Pengujian Terhadap 400 Kalimat dari 500 Kalimat Analisis

No.	Kalimat Korpus Uji	Relevan		Tidak Relevan	Precision $(\frac{a}{a+c})$	Recall $(\frac{a}{a+b})$	F-Measure $(2 \cdot \frac{p \cdot r}{p+r})$
		Terambil (a)	Tidak Terambil (b)	Terambil(c)			
1.	0-400						
2.	0-300, 401-500						
3.	0-200, 301-500						
4.	201-500						
5.	101-500						

Tabel 3.12 Skenario Pengujian Terhadap 500 Kalimat dari 500 Kalimat Analisis

No.	Kalimat Korpus Uji	Relevan		Tidak Relevan	Precision $(\frac{a}{a+c})$	Recall $(\frac{a}{a+b})$	F-Measure $(2 \cdot \frac{p \cdot r}{p+r})$
		Terambil (a)	Tidak Terambil (b)	Terambil(c)			
1.	0-500						

3.3.11.2.2 Rancangan Pengujian Penggalan Frasa Jeda dalam Satu Kalimat

Pengujian penggalan frasa jeda dalam satu kalimat digunakan untuk melihat persamaan penggalan frasa jeda dalam kalimat asli dan kalimat hasil prediksi *shallow parsing*. Skenario pengujian kemunculan jeda dapat dilihat pada tabel 3.13.

Tabel 3.13 Skenario Pengujian pada Frasa Jeda dalam Satu Kalimat Untuk Jeda 1 dan Jeda 2

No.	Frasa Penutur	Frasa <i>Shallow Parsing</i>	Relevan		Tidak Relevan	<i>Precision</i> n (a+c)	<i>Recall</i> l (a+b)
			Terambil	Tidak Terambil	Terambil		
1.	Contoh Kalimat	Contoh Kalimat	a	b	c	a+c	a+b
Total			a	b	c	a+c	a+b

Keterangan :

a = jumlah frasa jeda / jeda yang relevan ditemukan

b = jumlah frasa jeda / jeda yang relevan tidak ditemukan

Hasil dari total *precision* dan *recall* digunakan untuk menghitung nilai *f-measure* menggunakan rumus dari persamaan 2.5 pada bab II.

Contoh pengujian terhadap kalimat pada pengujian potongan frasa jeda 1 dapat dilihat pada tabel 3.14 sebagai berikut:

Tabel 3.14 Contoh Skenario Pengujian Potongan Frasa Jeda Singkat dalam Satu Kalimat

No.	Frasa Penutur	Frasa <i>Shallow Parsing</i>	Relevan		Tidak Relevan	<i>Precision</i> (a+c)	<i>Recall</i> (a+b)
			Terambil (a)	Tidak Terambil (b)	Terambil (c)		
1.	Kau bikin janji jam limak/2 , make/1 datanglah.	Kau bikin janji jam limak/2 , make/1 datanglah.	3	0	0	3	3
2.	Semue-mue-e/1 tepat/1 waktu	Semue-mue-e/1 tepat waktu	1	2	1	2	3
..
N - kali mat
Total			4	2	1	5	6

Pengujian penggalan frasa jeda singkat digunakan untuk melihat persamaan penggalan frasa jeda dari kalimat asli dan kalimat hasil prediksi *shallow parsing*. Pengujian penggalan frasa jeda singkat menggunakan perbandingan penggalan frasa dengan kejadian jeda singkat dan jeda panjang. Total kalimat yang diuji sebanyak 500 kalimat bahasa Melayu Pontianak dan 500 kalimat dari hasil implementasi *rule* jeda yang akan dipecah sesuai skenario pengujian jumlah kalimat pada bagian **3.3.10.2.1** tentang Skenario Pengujian Jumlah Kalimat.

Selanjutnya, contoh skenario pengujian potongan frasa jeda panjang dapat dilihat pada tabel 3.15.

Tabel 3.15 Contoh Skenario Pengujian Potongan Frasa Jeda Panjang dalam Satu Kalimat

No.	Frasa Penutur	Frasa <i>Shallow Parsing</i>	Relevan		Tidak Relevan	<i>Precision</i> (a+c)	<i>Recall</i> (a+b)
			Terambil (a)	Tidak Terambil (b)	Terambil (c)		
1.	Kau bikin janji jam limak/2 , make/1 datanglah.	Kau bikin janji jam limak/2 , make/1 datanglah.	2	0	0	2	2
2.	Semue-mue-e/1 tepat/1 waktu	Semue-mue-e/1 tepat waktu	1	0	0	1	1
..
N - kali mat
Total			3	0	0	3	3

Pengujian penggalan frasa jeda panjang digunakan untuk melihat persamaan penggalan frasa jeda dari kalimat asli dan kalimat hasil implementasi *rule* jeda. Pengujian penggalan frasa jeda panjang menggunakan perbandingan penggalan frasa dengan kejadian jeda panjang, dan tidak termasuk kejadian jeda singkat. Total kalimat yang diuji sebanyak 500 kalimat bahasa Melayu Pontianak dan 500 kalimat dari hasil implementasi *rule* jeda yang akan dipecah sesuai skenario pengujian jumlah kalimat pada bagian **3.3.11.2.1** tentang Skenario Pengujian Jumlah Kalimat.

3.3.12 Analisis Hasil Pengujian

Pada tahap ini, hasil pengujian berupa nilai akurasi jeda dalam satu kalimat penuh dan nilai *precision*, *recall*, dan *f-measure* dianalisis secara keseluruhan untuk mempermudah penarikan kesimpulan.

3.3.13 Penarikan Kesimpulan

Kesimpulan dirumuskan berdasarkan tahapan-tahapan yang telah dilakukan sebelumnya apakah metode *shallow parsing* menggunakan rule *grammar* dan *rule* jeda yang dikembangkan dalam penelitian ini dapat memprediksi jeda sesuai dengan yang diharapkan.

BAB IV

HASIL DAN ANALISIS

4.1 Hasil Penelitian

Hasil dari penelitian yang dilakukan adalah teks kalimat bahasa Melayu Pontianak yang sudah ditandai jeda singkat dan jeda panjang sesuai implementasi *rule* jeda yang didapat dari penggalan frasa *shallow parsing*.

4.1.1 Hasil Pengkategorian Indeks Jeda dan Penandaan Jeda di Korpus

Nilai "sil" dari *file **'breaks diolah dengan bantuan *spreadsheet* untuk menghitung nilai durasi tiap jeda seperti pada tabel 4.1.

Tabel 4.1 Tabel Perhitungan Durasi Jeda untuk Durasi Indeks Jeda

Jeda awal	Jeda Akhir	Durasi Jeda	Jumlah Kemunculan
0.6445475	0.6459904	0.001	1
5.3936384	5.3999506	0.006	1
1.4805433	1.4871555	0.007	1
3.2740529	3.2848776	0.011	3
4.4084134	4.4192347	0.011	
4.4432269	4.4543107	0.011	
1.8115121	1.8250113	0.013	3
1.986097	1.999496	0.013	
3.6817244	3.6950429	0.013	
7.8831156	7.8972757	0.014	1
2.7735575	2.7902864	0.017	2
3.8444939	3.8614356	0.017	
3.3583633	3.3777758	0.019	2
5.49	5.5094529	0.019	
6.7718082	6.7913141	0.02	
2.9380365	2.95885	0.021	1
4.0075	4.03	0.022	1
0.6475	0.67	0.023	1
...

Nilai durasi jeda digunakan untuk menghitung nilai min , max, median, dan mean dari nilai durasi jeda yang telah dijelaskan pada bagian 3.3.3 . Nilai modus didapat dari nilai yang paling besar di kolom jumlah kemunculan. Hasil perhitungan nilai min, max, median, mean dan modus dapat dilihat pada pada tabel 4.2.

Tabel 4.2 Hasil perhitungan durasi jeda penutur

MEAN	0.333
MEDIAN	0.177
MODUS	0.025
NILAI MIN	0.001
NILAI MAX	2.377

Berdasarkan perhitungan dari kejadian jeda yang terjadi, nilai rata-rata dari seluruh kejadian jeda adalah 0.333 detik. Nilai median atau nilai tengah dari rentang kejadian jeda adalah 0.177 detik. Nilai modus atau nilai yang paling sering muncul adalah 0.025 detik. Nilai terkecil dari kejadian jeda adalah 0.001 detik dan nilai terbesarnya adalah 2.377 detik. Hasil perhitungan dimasukkan ke dalam kolom durasi jeda pada tabel 4.3 untuk kategori indeks jeda.

Tabel 4.3 Kategori Indeks Jeda dengan durasi jeda

Indeks Jeda	Keterangan	Durasi Jeda
0	Tidak ada jeda	0 – < 0.025 s
1	Jeda yang terjadi tipis	0.025 – <= 0.33 s
2	Jeda yang terjadi tebal	> 0.33 s
,	Tanda baca koma (,)	,
.	Akhir kalimat	.!?

Nilai modus digunakan sebagai batas atas untuk lama indeks jeda 0 yang berarti lama kejadian jeda 0 adalah 0 sampai 0.025 detik. Untuk nilai *mean* digunakan sebagai batas atas untuk lama indeks jeda singkat (1) yang memiliki batas 0.025 – 0.333 detik. Untuk jeda panjang (2) menggunakan nilai lebih dari 0.333 detik.

Durasi jeda untuk indeks jeda dari tabel 4.2, kemudian digunakan untuk menandai kejadian jeda penutur di korpus teks bahasa Melayu Pontianak. Hasil penandaan kemunculan jeda sesuai kejadian jeda penutur dapat dilihat pada gambar 4.1.

- 1 Semue-mue-e/1 tepat waktu .
- 2 Kau bikin janji jam limak/2 , make/1 jam limak/1 kau haros datang .
- 3 Ade kegiatan jam tujuh/2 , make/1 jam tujuh/1 kegiatan tu haros dimulai .
- 4 Tadak macam tabiat kau tu/2 , kau bikin janji jam limak/2 , jam tujuh/1 kau baru datang .
- 5 Ade kegiatan jam tujuh/2 , jam sembilan baru dimulai .
- 6 Aoklah-aoklah/2 aku ngakulah/2 kalok pemerintah yang tak tepat waktu .
- 7 Ha/2 , ngaku gak/1 kau aher-e/2 , tak usahlah/1 kau nak nyangkal agek .

Gambar 4.1 Hasil Penandaan Jeda di Korpus

Penandaan angka satu dan dua disesuaikan dengan nilai durasi "sil" di *wavesurfer*. Jika nilai durasi "sil" bernilai 0.025 – 0.333 detik, maka "sil" yang terjadi setelah kata yang diucapkan diberi tanda "/1" yang menandakan kejadian jeda singkat. Jika nilai durasi "sil" bernilai batas lebih dari 0.333 detik, maka "sil" yang terjadi setelah kata yang diucapkan diberi tanda "/2" yang menandakan kejadian jeda panjang.

4.1.2 Hasil Pembuatan PoS Bahasa Melayu Pontianak

Hasil dari pembuatan PoS bahasa Melayu Pontianak yaitu tabel PoS dengan jumlah sebanyak 46 set PoS. Set PoS bahasa Melayu Pontianak dapat dilihat pada tabel 4.4.

Tabel 4.4 Tabel PoS Bahasa Melayu Pontianak

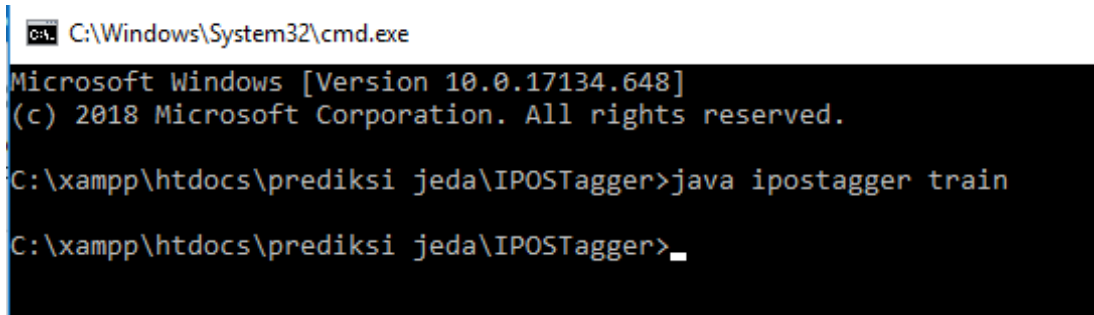
No	PoS	Deskripsi	Contoh Kata
1	VBR	Verba Reduplikasi	Jalan-jalan, poto-poto
2	VBK	Verba Berkonjugasi	Bersalam-salam, berputar-putar
3	VBT	Verba Transitif	Makai, nenggek, njajah
4	VBI	Verba Intransitif	Betanyak, balek, nuron
5	IN	Kata Depan	di, ke, dari, pada
6	UH	Kata Seru	Oi, woi, alamak
7	AR	Artikulus	Sang, si
8	RP	Partikel	pon, lah, jak
9	JJ	Kata Sifat	kaye, lawar, pandai, budoh
10	CON	Konjungsi	dan, kalok
11	OP	Open Parenthesis	({ [
12	CP	Close Parenthesis) }]
13	.	Sentence Terminator	! ? ...
14	.	Koma	,
15	:	Colon	::
16	SYM	Simbol	*%#&@
17	CR	Currency	Rp, \$
18	MD	Modal	nak, haros
19	NEG	Negation	bukan, jangan, tadak
20	SL	Slash	/
21	DS	Dash	-
22	QT	Quotation	" "
23	WP	WH-Pronoun	Ape, siape, berape
24	WDT	WH-Determiner	Ape, siape, barangsiape
25	DT	Determiner	ini, ni, tu, tu, tuh
26	FW	Foreign Word	wonderful, story
27	US	Unit Symbol	Gr, Kg, Cm
28	CDP	Primary Numeral	Satu, duak, tige
29	CDO	Ordinal Numeral	Kesatu, Keduak, ketige
30	CDI	Irregular Numeral	Beberape, segale, semue
31	CDF	Fraction Numeral	Setengah, seperempat
32	CDA	Kata Bantu Bilangan	biji, ekor, buah, orang
33	CDC	Collective Numeral	ratusan, ribuan, pulohan
34	RB	Adverb	paleng, sementara
35	WPRB	WH-Adverb	Cemane, ngape
36	FRB	Adverb of Frequency	jarang, sering, kadang-kadang
37	DRB	Adverb of Degree	agak, hamper, cukup

No	PoS	Deskripsi	Contoh Kata
38	TRB	Adverb of Time	udah, belom, dulok, sekarang
39	PRP	Personal Pronoun	aku, saye, kau, die
40	PRL	Locative Pronoun	sanak, sine, situk
41	PRN	Number Pronoun	satu-satunye, dua-duanye
42	NNP	Proper Noun	Eropa, Indonesia, Belanda
43	NNG	Genitive Common Noun	bukunye, rumahnye
44	NNC	Countable Common Noun	buku, rumah, karyawan
45	NNU	Uncountable Common Noun	aek, gula, nasi, ujan
46	NN	Common Noun	Martabat, janji

Dari 46 set PoS yang terbentuk pada tabel 4.4, ada 28 kelas kata yang sama dengan penelitian Alfa Farizki (2010) dan M.Iqbal Kamiludin (2017) yaitu VBT, VBI, IM, UH, RP, JJ, OP, CP, tanda baca (.), SYM, MD, NEG, QT, WP, DT, FW, CDP, CDO, CDI, CDC, RB, PRP, PRL, PRN, NNP, NNG dan NN. Selain itu ada 9 kelas kata yang merujuk pada penelitian Setyaningsih (2017) yaitu AR, CON, : , CR, WDT, US, CDF, CDA, dan WPRB. Kelas kata NNC dan NNU merujuk pada penelitian Adriani (2009). Penambahan kelas kata VBR dan VBK untuk membedakan kelas kata kerja yang mengalami pengulangan dan penambahan imbuhan. Penggantian nama untuk tanda “/” menjadi SL dan tanda “-“ menjadi DS untuk mempermudah penandaan kelas kata dan membedakan dengan *slash* atau “/” pemisah kata dan PoS.

4.1.3 Hasil *Tagging* PoS bahasa Melayu Pontianak

Selanjutnya, teks kalimat bahasa Melayu Pontianak ditandai menggunakan set PoS baru yang sudah dikembangkan. Penandaan atau *tagging* PoS bertujuan untuk mempermudah proses pengembangan *rule grammar* dan proses pemenggalan frasa menggunakan *shallow parsing*. Penandaan PoS dilakukan secara manual. Korpus yang telah diberi label PoS dapat dilihat pada gambar 4.2.



```

C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.17134.648]
(c) 2018 Microsoft Corporation. All rights reserved.

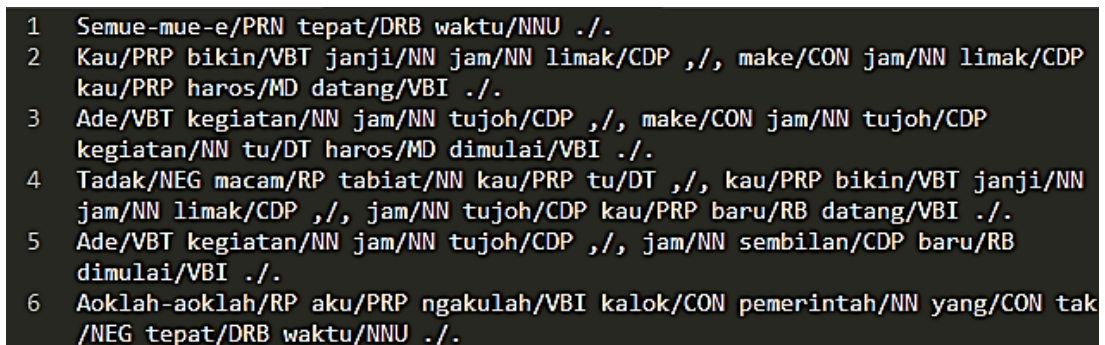
C:\xampp\htdocs\prediksi_jeda\IPOSTagger>java ipostagger train

C:\xampp\htdocs\prediksi_jeda\IPOSTagger>_

```

Gambar 4.2 Teks kalimat yang telah ditandai PoS

Tiap kata dalam kalimat bahasa Melayu Pontianak diberi tanda “/” atau garis miring dan diikuti label PoS dari tabel PoS di tabel 4.2 . Penandaan dilakukan secara manual berdasarkan makna dari kata tersebut ketika diucapkan dan dari Kamus Besar Bahasa Indonesia (KBBI). Hasil dari penandaan PoS ini yang digunakan sebagai korpus latih yang disimpan di dalam *subfolder* “*resource*” pada *folder* IPOSTagger dengan ekstensi .crp. Kemudian proses pelatihan dilakukan melalui *command prompt* pada *folder* IPOSTagger seperti yang terlihat pada gambar 4.3.



```

1 Semue-mue-e/PRN tepat/DRB waktu/NNU ./
2 Kau/PRP bikin/VBT janji/NN jam/NN limak/CDP ,/, make/CON jam/NN limak/CDP
  kau/PRP haros/MD datang/VBI ./
3 Ade/VBT kegiatan/NN jam/NN tujuh/CDP ,/, make/CON jam/NN tujuh/CDP
  kegiatan/NN tu/DT haros/MD dimulai/VBI ./
4 Tadak/NEG macam/RP tabiat/NN kau/PRP tu/DT ,/, kau/PRP bikin/VBT janji/NN
  jam/NN limak/CDP ,/, jam/NN tujuh/CDP kau/PRP baru/RB datang/VBI ./
5 Ade/VBT kegiatan/NN jam/NN tujuh/CDP ,/, jam/NN sembilan/CDP baru/RB
  dimulai/VBI ./
6 Aoklah-aoklah/RP aku/PRP ngakulah/VBI kalok/CON pemerintah/NN yang/CON tak
  /NEG tepat/DRB waktu/NNU ./

```

Gambar 4.3 Proses Pelatihan IPOSTagger

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

4.1.4 Hasil Pengembangan *Rule Grammar* Bahasa Melayu Pontianak

Teks kalimat bahasa Melayu Pontianak yang sudah ditandai label PoS dianalisis kejadian jeda sesuai jeda penutur yang terjadi pada setiap penggalan frasa sehingga dihasilkan *rule grammar*. *Rule grammar* yang dihasilkan digunakan untuk proses pemenggalan frasa dengan metode *shallow parsing* untuk mendapatkan frasa

jeda. Total *rule grammar* yang berhasil dikembangkan sebanyak 19 *rule*. Penjabaran *rule grammar* dibuat sebagai berikut.

1. **Rule 1 BP :** { <PRN|CDO|CDP|CDI|CDC|CDA|CDF>+ <DT>* | <SYM> }

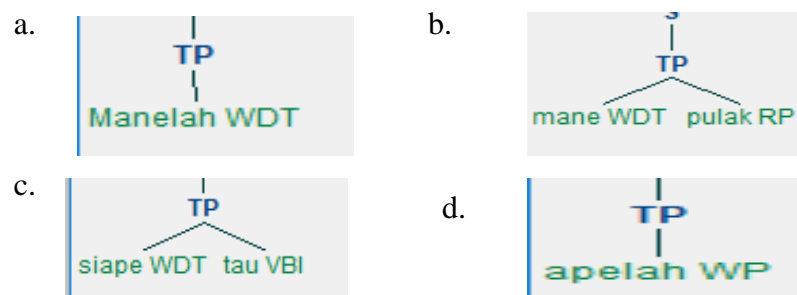
<PRN|CDO|CDP|CDI|CDC|CDA|CDF>+ <DT>* mengharuskan ada minimal satu karakter dari PRN|CDO|CDP|CDI|CDC|CDA|CDF untuk kategori frasa BP dan dapat diikuti oleh karakter DT seperti pada gambar 4.4 nomor a. Untuk <SYM> dikhususkan untuk kategori huruf yang berbentuk simbol seperti % (persen) seperti pada gambar 4.4 no b.



Gambar 4.4 Contoh dari penerapan *rule 1*

2. **Rule 2 TP :** { <RP|RB>* <WDT|WP>+ <RP|JJ>* <VBI>? }

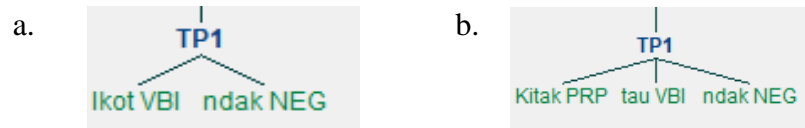
<RP|RB>* <WDT|WP>+ <RP|JJ>* <VBI>? dipakai untuk frasa yang terdiri dari karakter <WDT|WP>+ yang harus muncul minimal satu karakter dan diikuti <RP|JJ>* dengan kemunculan minimal nol atau N karakter. Karakter VBI dapat muncul dengan syarat kemunculan nol atau satu karakter. Contoh *rule 2* dapat dilihat pada gambar 4.5



Gambar 4.5 Contoh dari penerapan *rule 2*

3. **Rule 3 TP1 :** { <PRP>* <VBT|VBI>+ <NEG>+ | <WPRB>+ <RP>* }

3.1. <PRP>* <VBT|VBI>+ <NEG>+ dipakai untuk frasa yang terdiri dari karakter VBT|VBI dan NEG yang dapat diawali dengan pilihan PRP yang muncul minimal nol atau N karakter seperti pada gambar 4.6



Gambar 4.6 Contoh dari penerapan *rule 3* bagian 1

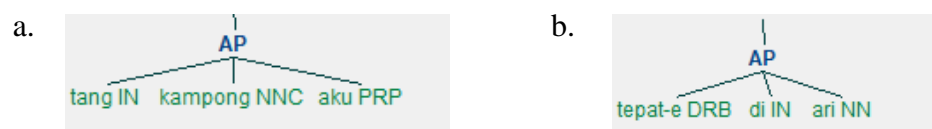
3.2. <WPRB>+ <RP>* dipakai untuk frasa yang terdiri dari karakter WPRB sebagai pembentuk utama frasa TP yang harus muncul minimal satu karakter dan dapat diikuti karakter RP yang muncul minimal nol atau N karakter seperti pada gambar 4.7.



Gambar 4.7 Contoh dari penerapan *rule 3* bagian 2

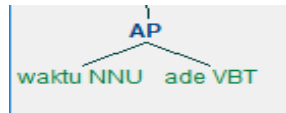
4. **Rule 4 AP :** { <DRB|RP>* <IN>+
<RB|VBR|NN|DT|NNP|NNC|PRL|RP|NNU|NNG|PRP>+ |
<NNU>+ <VBT>+ }

4.1. <DRB|RP>* <IN>+ <RB|VBR|NN|DT|NNP|NNC|PRL|RP|NNU|NNG|PRP>+ dikhususkan untuk frasa yang terdiri dari karakter IN dan <RB|VBR|NN|DT|NNP|NNC|PRL|RP|NNU|NNG|PRP>+ yang muncul minimal satu kali. Karakter DRB|RP dapat muncul minimal nol atau N karakter di awal frasa seperti pada gambar 4.8.



Gambar 4.8 Contoh dari penerapan *rule 4* bagian 1

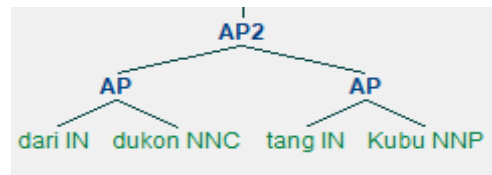
4.2. <NNU>+ <VBT>+ diperuntukkan untuk frasa ucapan yang mengandung karakter NNU dan VBT yang muncul minimal satu karakter. Penerapannya dapat dilihat pada gambar 4.9.



Gambar 4.9 Contoh dari penerapan *rule 4* bagian 2

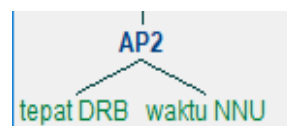
5. Rule 5 AP2 : {<AP> <AP> | <DRB>+ <NNU>* | <FRB>+ <RB>* }

5.1. <AP> <AP> dipakai untuk frasa yang terdiri dari gabungan dua frasa AP dari *rule* keempat. Contoh *rule 5* bagian satu dapat dilihat pada gambar 4.10.



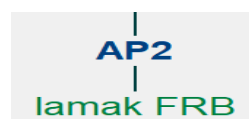
Gambar 4.10 Contoh dari penerapan *rule 5* bagian 1

5.2. <DRB>+ <NNU>* dipakai untuk frasa yang terdiri dari karakter DRB sebagai pembentuk utama frasa dan dapat diakhiri karakter NNU yang muncul minimal nol atau N karakter. Contoh *rule 5* bagian dua dapat dilihat pada gambar 4.11.



Gambar 4.11 Contoh dari penerapan *rule 5* bagian 2

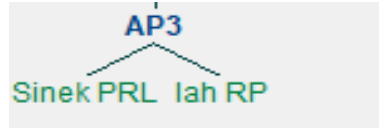
5.3. <FRB>+ <RB>* dipakai untuk frasa yang terdiri dari karakter FRB sebagai pembentuk frasa ucapan dan dapat diakhiri PoS RB yang muncul minimal nol atau N karakter. Contoh *rule 5* bagian tiga dapat dilihat pada gambar 4.12.



Gambar 4.12 Contoh dari penerapan *rule 5* bagian 3

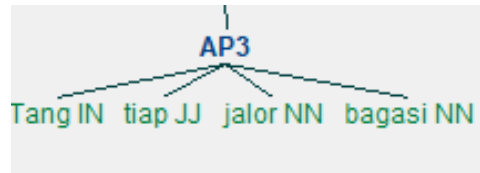
6. **Rule 6 AP3 :** { <PRL>+ <DT|RP>* | <IN>+ <JJ>+ <NN>+ }

6.1. <PRL>+ <DT|RP>* dipakai untuk frasa yang terdiri dari karakter PRL minimal satu karakter dan karakter DT|RP yang muncul minimal nol atau N karakter. Contoh *rule 6* bagian satu dapat dilihat pada gambar 4.13.



Gambar 4.13 Contoh dari penerapan *rule 6* bagian 1

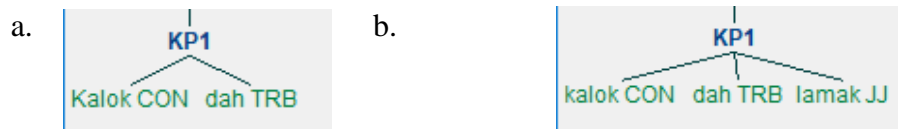
6.2. <IN>+ <JJ>+ <NN> dipakai untuk frasa yang terdiri dari karakter IN , JJ , dan NN yang muncul minimal satu karakter secara berurutan. Contoh *rule 6* bagian dua dapat dilihat pada gambar 4.14.



Gambar 4.14 Contoh dari penerapan *rule 6* bagian 2

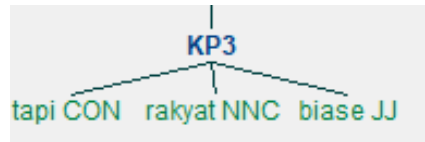
7. **Rule 7 KP1 :** { <CON>+ <TRB>+ <PRP>* <JJ>* | <CON>+ <JJ|RP>+ <NNU|NN>* | <JJ>* <VBT>? <TRB>+ <JJ|VBT|NNC|NN>* }

7.1. <CON>+ <TRB>+ <PRP>* <JJ>* dipakai untuk frasa yang terdiri dari CON , TRB dan dapat diakhiri PRP dan JJ yang muncul minimal nol atau N karakter. Contoh *rule 7* bagian satu dapat dilihat pada gambar 4.15.



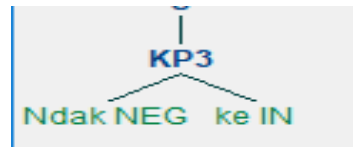
Gambar 4.15 Contoh dari penerapan *rule 7* bagian 1

7.2. <CON>+ <JJ|RP>+ <NNU|NN>* digunakan untuk membantu frasa ucapan yang terdiri atas CON dan JJ|RP dan diikuti karakter NNU|NN yang muncul minimal nol atau N karakter. Contoh *rule 7* bagian dua dapat dilihat pada gambar 4.16.



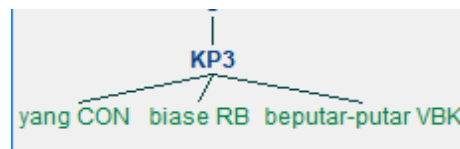
Gambar 4.19 Contoh dari penerapan *rule* 8 bagian 2

8.3. <NEG>+ <IN>+ dipakai untuk frasa yang terdiri dari karakter NEG dan IN secara berurutan. Contoh *rule* 8 bagian tiga dapat dilihat pada gambar 4.20.



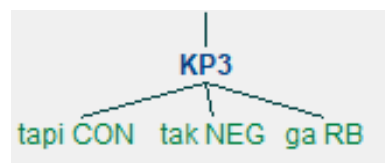
Gambar 4.20 Contoh dari penerapan *rule* 8 bagian 3

8.4. <CON>+ <RB|VBK>+ <VBI>* dipakai untuk frasa yang terdiri dari karakter CON dan RB|VBK dan dapat diakhiri karakter VBI minimal nol atau N karakter. Contoh *rule* 8 bagian empat dapat dilihat pada gambar 4.21.



Gambar 4.21 Contoh dari penerapan *rule* 8 bagian 4

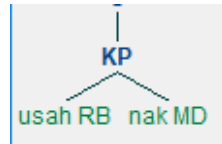
8.5. <CON>+ <NEG>+ <RB>+ dipakai untuk frasa yang terdiri dari karakter CON, NEG dan RB. Contoh *rule* 8 bagian lima dapat dilihat pada gambar 4.22.



Gambar 4.22 Contoh dari penerapan *rule* 8 bagian 5

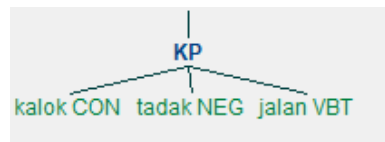
9. **Rule 9 KP :** { <RB>+ <MD|NNG>+ | <CON>+ <NEG>+ <VBT>+ | <CON>+ <DT>+ <NNU|NNC>+ | <CON|MD>? }

9.1. <RB>+ <MD|NNG>+ dipakai untuk frasa yang terdiri dari karakter RB dan MD|NNG sebagai pembentuk utama frasa. Contoh *rule 9* bagian satu dapat dilihat pada gambar 4.23.



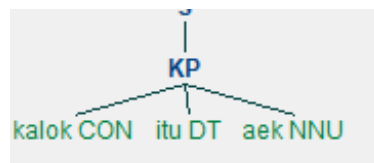
Gambar 4.23 Contoh dari penerapan *rule 9* bagian 1

9.2. <CON>+ <NEG>+ <VBT>+ dipakai untuk frasa yang terdiri dari karakter CON, NEG, dan VBT. Contoh *rule 9* bagian dua dapat dilihat pada gambar 4.24.



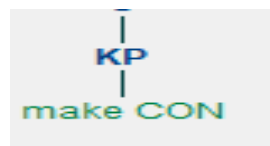
Gambar 4.24 Contoh dari penerapan *rule 9* bagian 2

9.3. <CON>+ <DT>+ <NNU|NNC>+ dipakai untuk frasa yang terdiri dari karakter CON, DT, dan NNU|NNC . Contoh *rule 9* bagian tiga dapat dilihat pada gambar 4.25.



Gambar 4.25 Contoh dari penerapan *rule 9* bagian 3

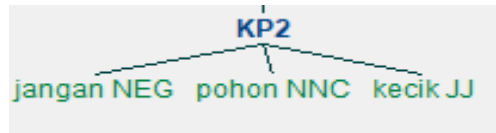
9.4. <CON|MD>? dipakai untuk frasa ucapan yang terdiri dari karakter CON dan MD yang muncul minimal nol atau satu karakter. Contoh *rule 9* bagian empat dapat dilihat pada gambar 4.26.



Gambar 4.26 Contoh dari penerapan *rule 9* bagian 4

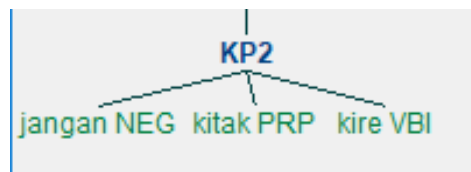
10. **Rule 10 KP2** : { <NEG>+ <NNC>+ <JJ>+ | <NEG>+ <PRP>+ <VBI>* | <RB>* <NEG>+ <RP|JJ|UH||RB|VBT>* }

- 10.1. <NEG>+ <NNC>+ <JJ>+ dipakai untuk frasa yang terdiri dari NEG, NNC dan JJ. Contoh *rule* 10 bagian satu dapat dilihat pada gambar 4.27.



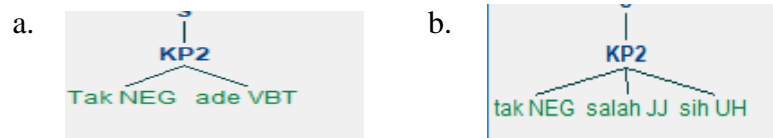
Gambar 4.27 Contoh dari penerapan *rule* 10 bagian 1

- 10.2. <NEG>+ <PRP>+ <VBI>* dipakai untuk frasa yang terdiri dari karakter NEG dan PRP dan dapat diikuti VBI yang muncul minimal nol atau N karakter. Contoh *rule* 10 bagian dua dapat dilihat pada gambar 4.28.



Gambar 4.28 Contoh dari penerapan *rule* 10 bagian 2

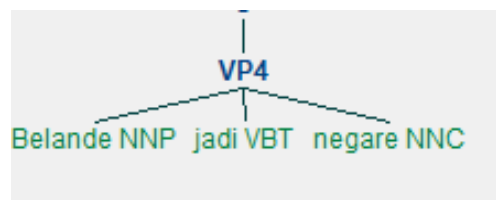
- 10.3. <RB>* <NEG>+ <RP|JJ|UH||RB|VBT>* dipakai untuk frasa yang terdiri dari karakter NEG sebagai pembentuk utama frasa dan diawali RB yang muncul minimal nol atau N karakter dan diakhiri RP|JJ|UH||RB|VBT yang muncul minimal nol atau N karakter seperti pada gambar 4.29.



Gambar 4.29 Contoh dari penerapan *rule* 10 bagian 3

11. **Rule 11 VP4** : <NNP>+ <VBT>+ <NNC>*

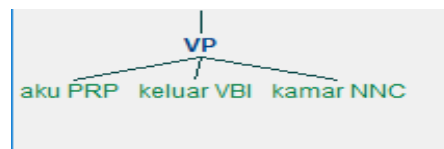
<NNP>+ <VBT>+ <NNC>* dipakai untuk frasa yang terdiri dari NNP dan VBT. NNC dapat muncul minimal nol atau N karakter. Contoh *rule* 11 dapat dilihat pada gambar 4.30.



Gambar 4.30 Contoh dari penerapan *rule 11*

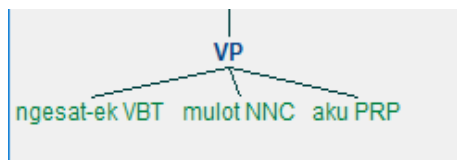
12. **Rule 12 VP :** {<RP>? <PRP>+ <MD|RB>* <VBI|VBT>+ <NNC>*|
 <VBT>+ <RB|RP|NNC|PRP|NN|JJ>* }

- 12.1. <RP>? <PRP>+ <MD|RB>* <VBI|VBT>+ <NNC>* dipakai untuk frasa ucapan yang terdiri dari frasa PRP dan VBI|VBT sebagai pembentuk utama frasa seperti pada gambar 4.31.



Gambar 4.31 Contoh dari penerapan *rule 12* bagian 1

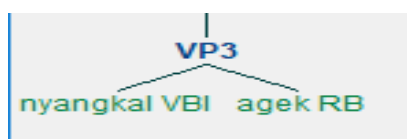
- 12.2. <VBT>+ <RB|RP|NNC|PRP|NN|JJ>* dipakai untuk frasa ucapan yang terdiri dari karakter VBT sebagai pembentuk frasa dan RB|RP|NNC|PRP|NN|JJ yang dapat muncul minimal nol atau N karakter. Contoh *rule 12* bagian dua dapat dilihat pada gambar 4.32.



Gambar 4.32 Contoh dari penerapan *rule 12* bagian 2

13. **Rule 13 VP3 :** { <VBI>+ <RB>+ }

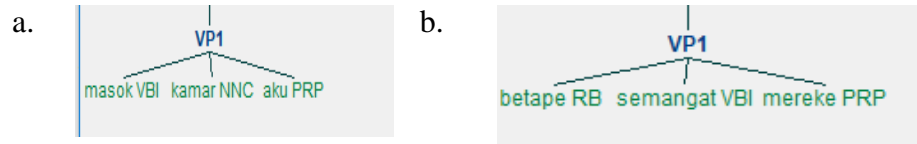
<VBI>+ <RB>+ diperuntukkan untuk frasa ucapan yang mengandung karakter VBI dan RB secara berurutan seperti pada gambar 4.33.



Gambar 4.33 Contoh dari penerapan *rule 13*

14. Rule 14 VP1 : { <RB>* <VBI>+ <PRP|RP|NNC|DT>* <BP>* }

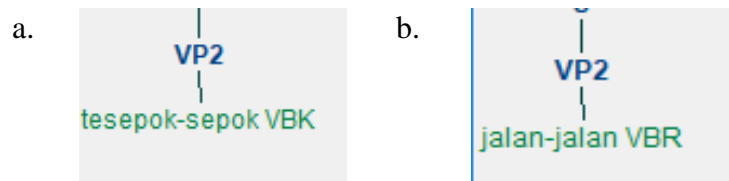
<RB>* <VBI>+ <PRP|RP|NNC|DT>* <BP>* dipakai untuk frasa yang terdiri dari karakter VBI sebagai pembentuk utama frasa dan dapat diawali karakter RB dan diakhiri karakter PRP|RP|NNC|DT dan frasa BP . Contoh rule 14 dapat dilihat pada gambar 4.34.



Gambar 4.34 Contoh dari penerapan rule 14

15. Rule 15 VP2 : { <VBK|VBR>+ }

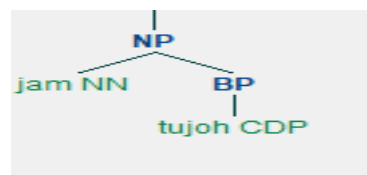
<VBK|VBR>+ dipakai untuk frasa ucapan yang terdiri dari karakter VBK atau VBR atau keduanya seperti pada gambar 4.35.



Gambar 4.35 Contoh dari penerapan rule 15

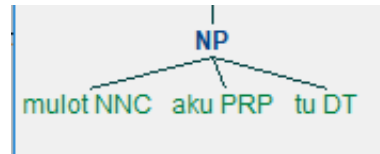
16. Rule 16 NP : { <NN>+ <BP> | <NNC|JJ>+ <PRP>+ <DT>* | <JJ>+ <RB>+ | <BP> <NN>+ | <NNC>+ <NNP>+ <DT>+ }

16.1. <NN>+ <BP> dipakai untuk NP yang terdiri atas NN dan frasa BP seperti pada gambar 4.36.



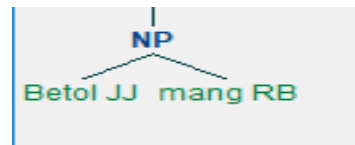
Gambar 4.36 Contoh dari penerapan rule 16 bagian 1

16.2. <NNC|JJ>+ <PRP>+ <DT>* dipakai untuk frasa ucapan yang terdiri dari frasa VP dan IN. Frasa VP diakhir kalimat bisa muncul minimal 0 atau 1 karakter. Contoh penerapan rule 16 bagian dua dapat dilihat pada gambar 4.37.



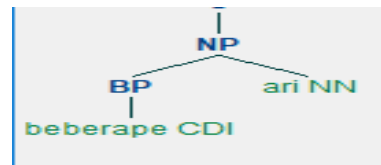
Gambar 4.37 Contoh dari penerapan *rule* 16 bagian 2

- 16.3.** $\langle JJ \rangle + \langle RB \rangle +$ dipakai untuk frasa ucapan yang terdiri dari JJ dan RB seperti pada gambar 4.38.



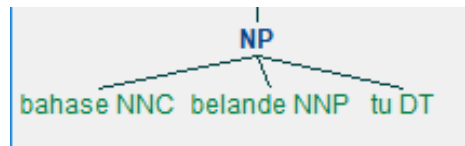
Gambar 4.38 Contoh dari penerapan *rule* 16 bagian 3

- 16.4.** $\langle BP \rangle \langle NN \rangle +$ dipakai untuk frasa ucapan yang terdiri dari BP dan NN seperti pada gambar 4.39.



Gambar 4.39 Contoh dari penerapan *rule* 16 bagian 4

- 16.5.** $\langle NNC \rangle + \langle NNP \rangle + \langle DT \rangle +$ dipakai untuk frasa ucapan yang terdiri dari NNC, NNP dan DT seperti pada gambar 4.40.

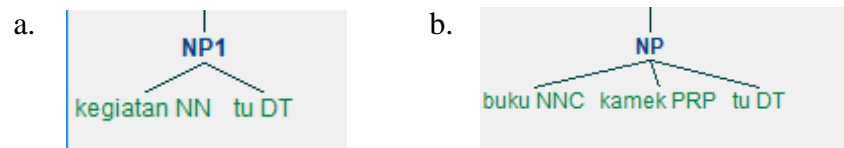


Gambar 4.40 Contoh dari penerapan *rule* 16 bagian 5

- 17. Rule 17 NP1:** { $\langle NNC \rangle^* \langle NN|PRP|NNU \rangle + \langle DT \rangle +$ | $\langle NNC \rangle + \langle JJ \rangle +$
 $\langle DT \rangle^*$ | $\langle JJ \rangle + \langle DT \rangle + \langle RP \rangle +$ | $\langle NNC \rangle + \langle BP \rangle$
 $\langle NNC \rangle +$ | $\langle NNU \rangle + \langle NN \rangle +$ | $\langle NN|RP \rangle^* \langle PRP \rangle +$
 $\langle RB|RP \rangle^*$ }

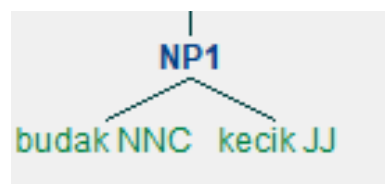
- 17.1.** $\langle NNC \rangle^* \langle NN|PRP|NNU \rangle + \langle DT \rangle +$ dipakai untuk frasa ucapan yang terdiri dari $NN|PRP|NNU$ dan DT. Karakter NNC dapat muncul minimal nol atau N karakter.

Contoh *rule* 17 bagian satu dapat dilihat pada gambar 4.41



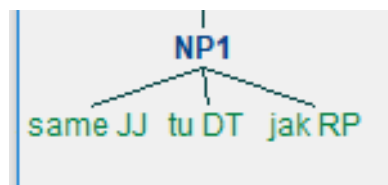
Gambar 4.41 Contoh dari penerapan *rule* 17 bagian 1

17.2. $\langle \text{NNC} \rangle + \langle \text{JJ} \rangle + \langle \text{DT} \rangle^*$ dipakai untuk frasa yang terdiri dari karakter NNC dan JJ. DT dapat muncul minimal nol atau N karakter seperti pada gambar 4.42.



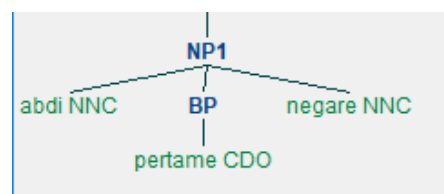
Gambar 4.42 Contoh dari penerapan *rule* 17 bagian 2

17.3. $\langle \text{JJ} \rangle + \langle \text{DT} \rangle + \langle \text{RP} \rangle$ dipakai untuk frasa yang terdiri dari karakter JJ, DT, dan RP. Contoh *rule* 17 bagian tiga dapat dilihat pada gambar 4.43.



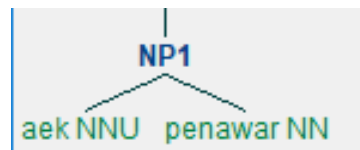
Gambar 4.43 Contoh dari penerapan *rule* 17 bagian 3

17.4. $\langle \text{NNC} \rangle + \langle \text{BP} \rangle \langle \text{NNC} \rangle +$ dipakai untuk frasa yang terdiri dari karakter NNC, BP dan NNC. Contoh *rule* 17 bagian empat dapat dilihat pada gambar 4.44.



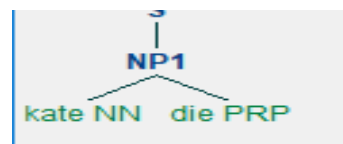
Gambar 4.44 Contoh dari penerapan *rule* 17 bagian 4

- 17.5.** $\langle \text{NNU} \rangle + \langle \text{NN} \rangle +$ dipakai untuk frasa yang terdiri dari karakter NNU dan NN. Contoh penerapan dari *rule* 17 bagian lima dapat dilihat pada gambar 4.45.



Gambar 4.45 Contoh dari penerapan *rule* 17 bagian 5

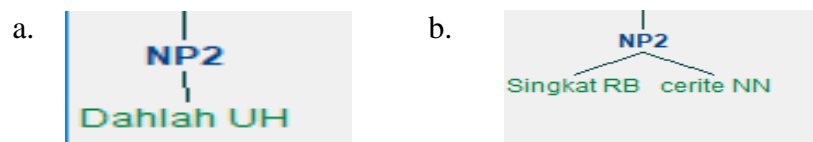
- 17.6.** $\langle \text{NN|RP} \rangle^* \langle \text{PRP} \rangle + \langle \text{RB|RP} \rangle^*$ dipakai untuk frasa yang terdiri dari karakter PRP. Karakter NN|RP dapat muncul minimal nol atau N karakter diawal. Karakter RB|RP dapat muncul minimal nol atau N karakter di akhir. Contoh penerapan dari *rule* 17 bagian enam dapat dilihat pada gambar 4.46.



Gambar 4.46 Contoh dari penerapan *rule* 17 bagian 6

- 18.** *Rule* 18 NP2 : $\{ \langle \text{RP|UH|RB} \rangle + \langle \text{NNP|DT|NN|JJ|NNC} \rangle^* \langle \text{RP} \rangle^* | \langle \text{NNC} \rangle + \langle \text{DT} \rangle + \langle \text{RP} \rangle^* | \langle \text{NNG|NN} \rangle^* \langle \text{DT|NNP|FW} \rangle + \langle \text{NNC|NNG|RB|RP} \rangle^* | \langle \text{NNG} \rangle + \langle \text{JJ} \rangle^* \}$

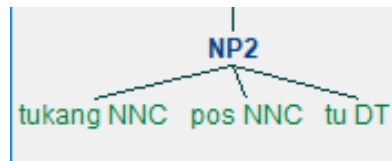
- 18.1.** $\langle \text{RP|UH|RB} \rangle + \langle \text{NNP|DT|NN|JJ|NNC} \rangle^* \langle \text{RP} \rangle^*$ dipakai untuk frasa ucapan yang terdiri dari RP|UH|RB sebagai pembentuk utama frasa dan dapat diikuti NNP|DT|NN|JJ|NNC dan RP yang muncul minimal nol atau N karakter. Contoh penerapan *rule* 18 bagian satu dapat dilihat pada gambar 4.47.



Gambar 4.47 Contoh dari penerapan *rule* 18 bagian 1

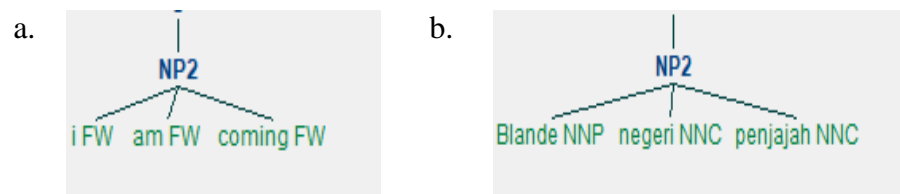
- 18.2.** $\langle \text{NNC} \rangle + \langle \text{DT} \rangle + \langle \text{RP} \rangle^*$ dipakai untuk frasa ucapan yang terdiri dari karakter NNC|DT. Karakter RP dapat muncul minimal 0 atau N

karakter. Contoh penerapan *rule* 18 bagian dua dapat dilihat pada gambar 4.48.



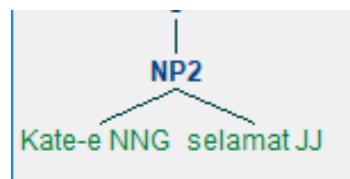
Gambar 4.48 Contoh dari penerapan *rule* 18 bagian 2

18.3. $\langle \text{NNG} | \text{NN} \rangle^* \langle \text{DT} | \text{NNP} | \text{FW} \rangle^+ \langle \text{NNC} | \text{NNG} | \text{RB} | \text{RP} \rangle^*$ dipakai untuk frasa ucapan yang terdiri dari karakter $\text{DT} | \text{NNP} | \text{FW}$. NNG dan NN dapat muncul diawal minimal 0 atau N karakter. Karakter $\text{NNC} | \text{NNG} | \text{RB} | \text{RP}$ dapat muncul di akhir minimal 0 atau N karakter. Contoh dari penerapan *rule* 18 bagian tiga dapat dilihat pada gambar 4.49.



Gambar 4.49 Contoh dari penerapan *rule* 18 bagian 3

18.4. $\langle \text{NNG} \rangle^+ \langle \text{JJ} \rangle^?$ dipakai untuk frasa ucapan yang terdiri dari NNG dan JJ yang dapat muncul nol atau satu karakter saja. Contoh penerapan dari *rule* 18 bagian empat dapat dilihat pada gambar 4.50.

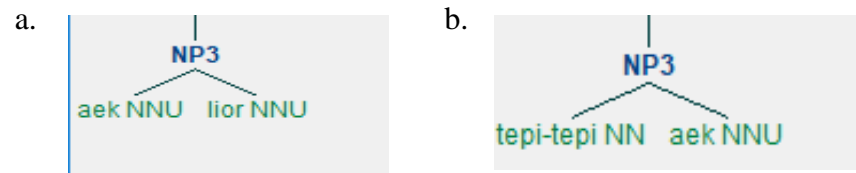


Gambar 4.50 Contoh dari penerapan *rule* 18 bagian 4

19. Rule 19 NP3 : $\{ \langle \text{AR} \rangle^* \langle \text{JJ} | \text{NN} | \text{NNU} \rangle^+ \langle \text{BP} | \text{NNC} | \text{RP} \rangle^? \langle \text{NNC} | \text{NNU} \rangle^* | \langle \text{NNC} \rangle^+ \langle \text{DS} | \text{NNC} | \text{NNU} | \text{NN} \rangle^* \}$

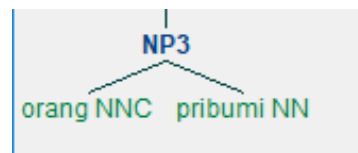
19.1. $\langle \text{AR} \rangle^* \langle \text{JJ} | \text{NN} | \text{NNU} \rangle^+ \langle \text{BP} | \text{NNC} | \text{RP} \rangle^? \langle \text{NNC} | \text{NNU} \rangle^*$ dipakai untuk frasa ucapan yang terdiri dari $\text{JJ} | \text{NN} | \text{NNU}$ sebagai pembentuk frasa yang muncul minimal 1 karakter. $\text{BP} | \text{NNC} | \text{RP}$ dapat muncul minimal nol atau satu karakter. Karakter $\text{NNC} | \text{NNU}$ dapat muncul minimal nol atau N

karakter. Karakter AR dapat muncul diawal dengan kemunculan minimal nol atau N karakter . Contoh penerapan *rule 19* bagian satu dapat dilihat pada gambar 4.51.



Gambar 4.51 Contoh dari penerapan *rule 19* bagian 1

19.2. $\langle \text{NNC} \rangle^+ \langle \text{DS} | \text{NNC} | \text{NNU} | \text{NN} \rangle^*$ dipakai untuk frasa ucapan yang terdiri dari karakter NNC sebagai pembentuk frasa. Karakter $\text{DS} | \text{NNC} | \text{NNU} | \text{NN}$ muncul minimal nol atau N karakter. Contoh penerapan *rule 19* bagian dua dapat dilihat pada gambar 4.52.



Gambar 4.52 Contoh dari penerapan *rule 19* bagian 2

4.1.5 Implementasi Metode Shallow Parsing

Shallow Parsing digunakan sebagai metode untuk memenggal kalimat dengan bantuan fungsi *RegexpParser* pada NLTK untuk menghasilkan penggalan frasa ucapan. Hasil penelitian dari *shallow parsing* yaitu penggalan frasa ucapan berdasarkan *rule grammar* baru yang sudah di jelaskan pada bagian 4.1.4.

Proses *shallow parsing* dilakukan seperti yang telah dijelaskan pada bagian 3.3.7 yang dimulai dari membaca inputan kalimat bahasa Melayu Pontianak.



Gambar 4.53 Teks kalimat bahasa Melayu Pontianak untuk *Shallow Parsing*

Inputan untuk proses *shallow parsing* berupa file korpus bahasa Melayu Pontianak yang sudah diberi label PoS dan disimpan dalam format *.txt seperti pada gambar 4.53.

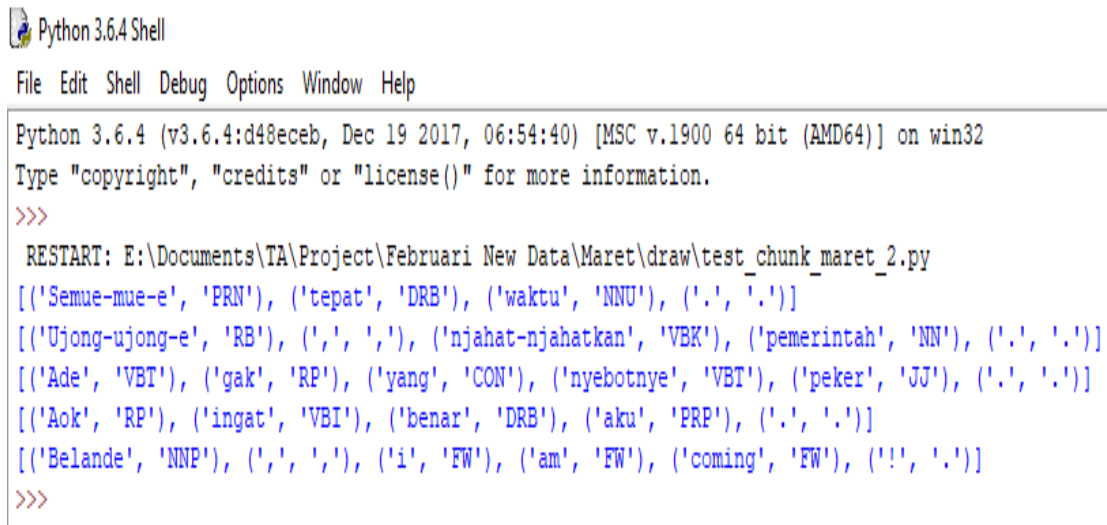
4.1.5.1 *Split Kata dan PoS*

Pada bagian ini, kalimat dibagi menjadi per kata dan PoS untuk mengidentifikasi label PoS tiap kata sehingga bisa diperiksa menggunakan *rule grammar*.

Kode program 4.1 Indentifikasi label PoS Kata

1. `f0 = open("KORPUS KATA-POS 500.txt", "r")`
2. `open('kal_prediksi_jeda.txt', 'w').close()`
3. `open('chunk.txt', 'w').close()`
4. `i = 0`
5. `for line in f0:`
6. `kal = line.strip()`
7. `kalimat = [nltk.tag.str2tuple(t) for t in kal.split()]`

Pada kode program 4.1 baris nomor 7, dengan menggunakan bantuan NLTK, kalimat dipisah menjadi per kata dan PoS. Hasil pemisahan kata dan PoS dapat dilihat pada gambar 4.54



```

Python 3.6.4 Shell
File Edit Shell Debug Options Window Help

Python 3.6.4 (v3.6.4:d48eceb, Dec 19 2017, 06:54:40) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.

>>>
RESTART: E:\Documents\TA\Project\Februari New Data\Maret\draw\test_chunk_maret_2.py
[('Semue-mue-e', 'PRN'), ('tepat', 'DRB'), ('waktu', 'NNU'), ('.', '.')]
[('Ujong-ujong-e', 'RB'), ('', ''), ('njahat-njahatkan', 'VBK'), ('pemerintah', 'NN'), ('.', '.')]
[('Ade', 'VBT'), ('gak', 'RP'), ('yang', 'CON'), ('nyebotnye', 'VBT'), ('peker', 'JJ'), ('.', '.')]
[('Aok', 'RP'), ('ingat', 'VBI'), ('benar', 'DRB'), ('aku', 'PRP'), ('.', '.')]
[('Belande', 'NNP'), ('', ''), ('i', 'FW'), ('am', 'FW'), ('coming', 'FW'), ('!', '.')]
>>>

```

Gambar 4.54 Identifikasi label PoS kata

Hasil dari pemisahan kata dan PoS pada gambar 5.4 selanjutnya diproses pada tahap selanjutnya.

4.1.5.2 *Shallow Parsing Kalimat*

Tiap kata dalam kalimat inputan kemudian diperiksa menggunakan *rule grammar* berdasarkan tipe PoS kata tersebut dan dikelompokkan ke tipe frasa berdasarkan *rule grammar*. *Rule grammar* yang telah dibuat dapat dilihat pada kode program 4.2.

Kode program 4.2 *Rule grammar*

1. grammar = r"""
2. BP : { <PRN|CDO|CDP|CDI|CDC|CDA|CDF>+ <DT>* | <SYM> }
3. TP : { <RP|RB>* <WDT|WP>+ <RP|JJ>* <VBI>? }
4. TP1 : { <PRP>* <VBT|VBI>+ <NEG>+ | <WPRB>+ <RP>* }
5. AP : { <DRB|RP>* <IN>+ <RB|VBR|NN|DT|NNP|NNC|PRL|RP|NNU|NNG|PRP>+ | <NNU>+ <VBT>+ }
6. AP2 : { <AP> <AP> | <DRB>+ <NNU>* | <FRB>+ <RB>* }
7. AP3 : { <PRL>+ <DT|RP>* | <IN>+ <JJ>+ <NN>+ }
8. KP1 : { <CON>+ <TRB>+ <PRP>* <JJ>* | <CON>+ <JJ|RP>+ <NNU|NN>* | <JJ>* <VBT>? <TRB>+ <JJ|VBT|NNC|NN>* }
9. KP3 : { <CON>+ <PRP>+ <RP>+ | <CON>+ <NNC>+ <JJ>+ | <NEG>+ <IN>+ | <CON>+ <RB|VBK>+ <VBI>* | <CON>+ <NEG>+ <RB>+ }
10. KP : { <RB>+ <MD|NNG>+ | <CON>+ <NEG>+ <VBT>+ | <CON>+ <DT>+ <NNU|NNC>+ | <CON|MD>? }
11. KP2 : { <NEG>+ <NNC>+ <JJ>+ | <NEG>+ <PRP>+ <VBI>* | <RB>* <NEG>+ }

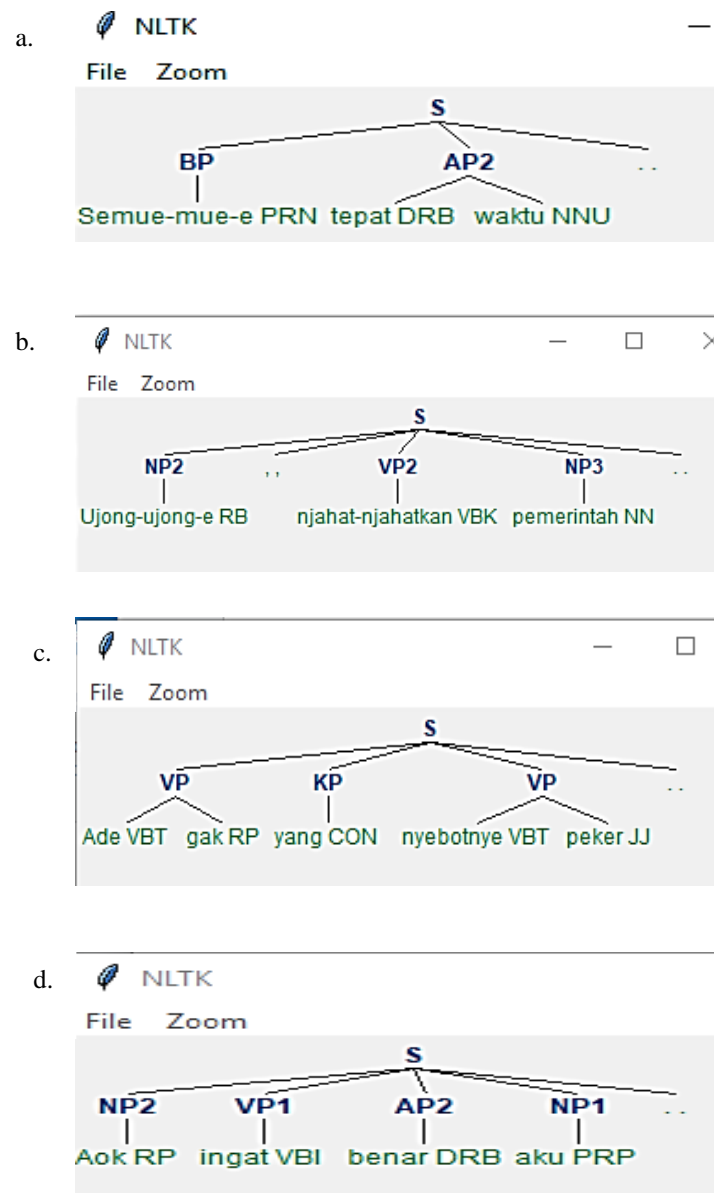
- <RP|JJ|UH|RB|VBT>* }
12. VP4 : { <NNP>+ <VBT>+ <NNC>* }
 13. VP : { <RP>? <PRP>+ <MD|RB>* <VBI|VBT>+ <NNC>* | <VBT>+ <RB|RP|NNC|PRP|NN|JJ>* }
 14. VP3 : { <VBI>+ <RB>+ }
 15. VP1 : { <RB>* <VBI>+ <PRP|RP|NNC|DT>* <BP>* }
 16. VP2 : { <VBK|VBR>+ }
 17. NP : { <NN>+ <BP> | <NNC|JJ>+ <PRP>+ <DT>* | <JJ>+ <RB>+ | <BP> <NN>+ | <NNC>+ <NNP>+ <DT>+ }
 18. NP1 : { <NNC>* <NN|PRP|NNU>+ <DT>+ | <NNC>+ <JJ>+ <DT>* | <JJ>+ <DT>+ <RP>+ | <NNC>+ <BP> <NNC>+ | <NNU>+ <NN>+ | <NN|RP>* <PRP>+ <RB|RP>* }
 19. NP2 : { <RP|UH|RB>+ <NNP|DT|NN|JJ|NNC>* <RP>* | <NNC>+ <DT>+ <RP>* | <NNG|NN>* <DT|NNP|FW>+ <NNC|NNG|RB|RP>* | <NNG>+ <JJ>? }
 20. NP3 : { <AR>* <JJ|NN|NNU>+ <BP|NNC|RP>? <NNC|NNU>* | <NNC>+ <DS|NNC|NNU|NN>* }
 21. ""

Kalimat kemudian di-*parsing* atau dipecah menjadi penggalan frasa berdasarkan *rule grammar* pada kode program 4.2 menggunakan bantuan *Regexparser* yang dapat dilihat pada kode program 4.3

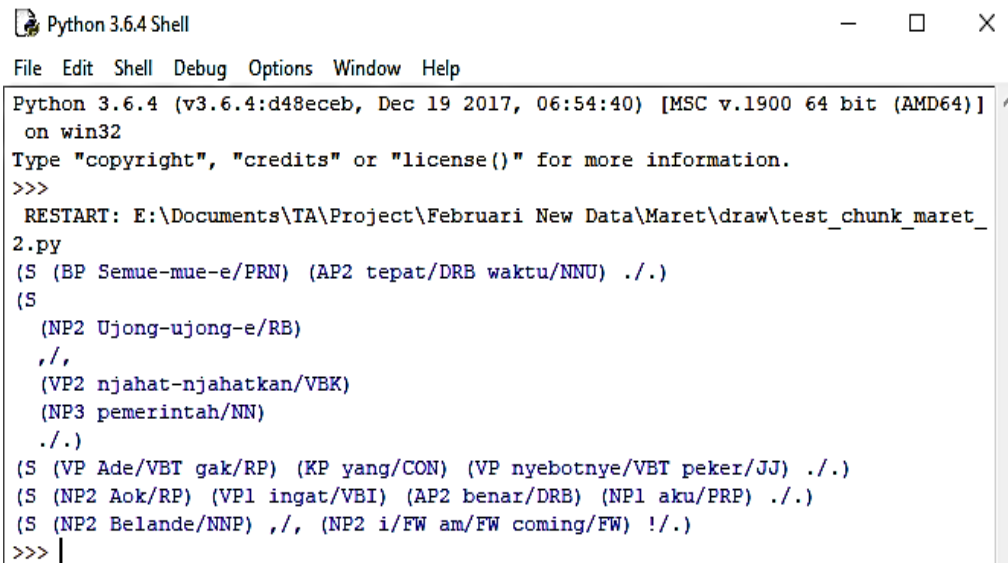
Kode program 4.3 *Parsing* Kalimat

1. cp = nltk.RegexpParser(grammar)
2. result = cp.parse(kalimat)

Berdasarkan kode program 4.3 baris 2, kalimat di-*parse* menggunakan fungsi *RegexpParser* pada NLTK berdasarkan *grammar* yang sudah dibuat di kode program 4.2. Hasil dari *parsing* berupa pecahan kalimat yang berupa *parsing tree*. *Parsing tree* merupakan representasi penggalan *parsing* yang saling berhubungan. Hasil *parsing* dapat dilihat pada gambar 4.55 dan 4.56



Gambar 4.55 Parsing tree dari shallow parsing



```
Python 3.6.4 (v3.6.4:d48eceb, Dec 19 2017, 06:54:40) [MSC v.1900 64 bit (AMD64)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>>
RESTART: E:\Documents\TA\Project\Februari New Data\Maret\draw\test_chunk_maret_
2.py
(S (BP Semue-mue-e/PRN) (AP2 tepat/DRB waktu/NNU) ./.)
(S
  (NP2 Ujong-ujong-e/RB)
  ./,
  (VP2 njahat-njahatkan/VBK)
  (NP3 pemerintah/NN)
  ./.)
(S (VP Ade/VBT gak/RP) (KP yang/CON) (VP nyebotnye/VBT peker/JJ) ./.)
(S (NP2 Aok/RP) (VP1 ingat/VBI) (AP2 benar/DRB) (NP1 aku/PRP) ./.)
(S (NP2 Belande/NNP) ./, (NP2 i/FW am/FW coming/FW) !/.)
>>> |
```

Gambar 4.56 Hasil *parsing* berdasarkan *grammar*

Hasil *parsing* yang ditunjukkan di gambar 4.56 merupakan frasa jeda ucapan hasil dari *shallow parsing*.

4.1.5.3 Normalisasi Hasil *Shallow Parsing*

Hasil *parsing* disimpan dalam file format *.txt dan merupakan penggalan frasa dari kalimat teks bahasa Melayu Pontianak sesuai *rule grammar* yang dibuat. Hasil dapat dilihat pada gambar 4.57.

```

bruari New Data\Maret\draw\chunk.txt - Sublime Text (UNREGISTERED)
File Goto Tools Project Preferences Help
chunk.txt x
1 0 (BP Semue-mue-e/PRN)
2 1 (AP2 tepat/DRB waktu/NNU)
3 2 .
4
5 0 (NP2 Ujong-ujong-e/RB)
6 1 ,
7 2 (VP2 njahat-njahatkan/VBK)
8 3 (NP3 pemerintah/NN)
9 4 .
10
11 0 (VP Ade/VBT gak/RP)
12 1 (KP yang/CON)
13 2 (VP nyebotnye/VBT peker/JJ)
14 3 .
15
16 0 (NP2 Aok/RP)
17 1 (VP1 ingat/VBI)
18 2 (AP2 benar/DRB)
19 3 (NP1 aku/PRP)
20 4 .
21
22 0 (NP2 Belande/NNP)
23 1 ,
24 2 (NP2 i/FW am/FW coming/FW)
25 3 !

```

Gambar 4.57 Output shallow parsing

Hasil dari *output* shallow parsing berupa penggalan frasa dalam satu kalimat yang ditandai dari angka nol untuk tiap kalimat. Hasil *shallow parsing* diproses pada tahap imlementasi *rule* jeda.

4.1.6 Hasil Pengembangan *Rule* Jeda

Untuk meningkatkan nilai akurasi jeda, dikembangkan *rule* jeda untuk penandaan jeda singkat dan jeda panjang di teks kalimat bahasa Melayu Pontianak. Pada penelitian sebelumnya oleh M.Iqbal Kamiludin, tiap penggalan frasa merupakan jeda karena *rule grammar* yang dibuat berdasarkan aturan S-P-O-K. Pada penelitian ini, tiap penggalan frasa tidak harus menghasilkan jeda berdasarkan analisis kejadian jeda yang telah dilakukan. Sehingga dikembangkan *rule* jeda untuk mencocokkan penggalan frasa yang tepat yang perlu ditandai jeda. Hasil dari pengembangan *rule* jeda berupa 76 *rule* untuk penandaan jeda singkat dan 33 *rule* untuk penandaan jeda panjang.

Berikut adalah *rule* jeda yang akan dipakai dalam untuk prediksi jeda

ucapan bahasa Melayu Pontianak :

a. **RULE 1 =** ["BP-AP2", "KP-NP", "NP-VP", "NP-NP1", "VP1-NP1", "KP2-VP", "AP-AP", "KP-KP", "VP-AP", "AP-KP", "BP-AP", "VP-TP", "VP-KP1", "NP2-VP1", "VP-KP2", "VP1-AP", "VP2-NP", "NP2-AP", "NP1-AP2", "KP1-KP1", "NP3-AP", "VP2-VP", "TP-VP1", "AP3-VP1", "NP2-KP", "NP-AP", "NP2-AP2", "VP-KP2", "KP2-KP1", "KP-KP2", "KP2-NP1", "AP-NP1", "NP2-NP3", "NP2-VP2", "KP-BP", "KP-NP1", "KP1-VP", "NP2-TP", "NP2-NP2", "AP3-KP", "KP1-NP1", "AP-VP", "NP2-KP2", "VP3-VP1", "NP2-NP1", "BP-KP1", "KP1-KP", "NP3-NP1", "BP-NP3", "NP-KP2", "KP3-VP", "NP2-BP", "VP3-AP", "KP-VP2", "VP2-NP3", "NP3-KP2", "TP-NP3", "NP2-VP", "KP-AP", "AP-AP2", "NP3-NP3", "BP-VP", "VP3-VP", "VP2-AP", "KP1-KP2", "NP1-AP3", "VP1-VP1", "TP1-VP", "AP3-NP", "KP-TP1", "TP1-TP1", "KP-AP2", "VP1-TP1", "NP1-NP1", "NP2-NP", "NP3-TP1",

merupakan *rule* yang digunakan untuk menandai kejadian jeda singkat atau jeda dengan indeks jeda 1.

b. **RULE 2 =** ["NP-", "VP-", "NP1-", "KP-", "BP-", "VP1-KP", "NP2-", "KP2-", "NP3-", "NP2-KP1", "BP-NP1", "AP2-KP3", "NP-KP1", "TP-", "AP-", "VP1-", "NP-KP", "AP2-", "KP1-", "KP1-NP3", "VP1-TP", "VP2-", "KP3-KP", "KP3-NP2", "VP-AP2", "AP-KP3", "KP3-", "VP-KP3", "AP3-", "TP1-AP", "NP1-VP4", "TP1-", "NP-VP4"]

merupakan *rule* yang dipakai untuk menandai kejadian jeda panjang atau jeda dengan indeks jeda 2.

4.1.7 Implementasi *Rule* Jeda

Indeks Jeda diberikan pada potongan kalimat sesuai dengan *rule* jeda yang telah dijelaskan di bagian 4.1.6.

```

rule_1 = ["BP-AP2", "KP-NP", "NP-VP", "NP-NP1", "VP1-NP1", "KP2-VP",
"AP-AP", "KP-KP", "VP-AP", "AP-KP", "BP-AP", "VP-TP", "VP-KP1",
"NP2-VP1", "VP-KP2", "VP1-AP", "VP2-NP", "NP2-AP", "NP1-AP2",
"KP1-KP1", "NP3-AP", "VP2-VP", "TP-VP1", "AP3-VP1", "NP2-KP",
"NP-AP", "NP2-AP2", "VP-KP2", "KP2-KP1", "KP-KP2", "KP2-NP1",
"AP-NP1", "NP2-NP3", "NP2-VP2", "KP-BP", "KP-NP1", "KP1-VP",
"NP2-TP", "NP2-NP2", "AP3-KP", "KP1-NP1", "AP-VP", "NP2-KP2",
"VP3-VP1", "NP2-NP1", "BP-KP1", "KP1-KP", "NP3-NP1", "BP-NP3",
"NP-KP2", "KP3-VP", "NP2-BP", "VP3-AP", "KP-VP2", "VP2-NP3",
"NP3-KP2", "TP-NP3", "NP2-VP", "KP-AP", "AP-AP2", "NP3-NP3",
"BP-VP", "VP3-VP", "VP2-AP", "KP1-KP2", "NP1-AP3", "VP1-VP1",
"TP1-VP", "AP3-NP", "KP-TP1", "TP1-TP1", "KP-AP2", "VP1-TP1",
"NP1-NP1", "NP2-NP",
"NP3-TP1"]
rule_2 = ["NP-", "VP-", "NP1-", "KP-", "BP-", "VP1-KP", "NP2-",
"KP2-", "NP3-", "NP2-KP1", "BP-NP1", "AP2-KP3", "NP-KP1",
"TP-", "AP-", "VP1-", "NP-KP", "AP2-", "KP1-", "KP1-NP3",
"VP1-TP", "VP2-", "KP3-KP", "KP3-NP2", "VP-AP2", "AP-KP3",
"KP3-", "VP-KP3", "AP3-", "TP1-AP", "NP1-VP4", "TP1-", "NP-VP4"]

```

Gambar 4.58 Rule jeda

Setelah itu, penggalan frasa yang didapat dari proses *shallow parsing* selanjutnya diproses untuk menghilangkan label PoS di tiap kata dalam penggalan frasa. Kemudian, penggalan frasa digabung menjadi satu kalimat penuh dengan indeks jeda yang sudah ditandai sesuai *rule* jeda pada gambar 4.58 berdasarkan kode program 4.4.

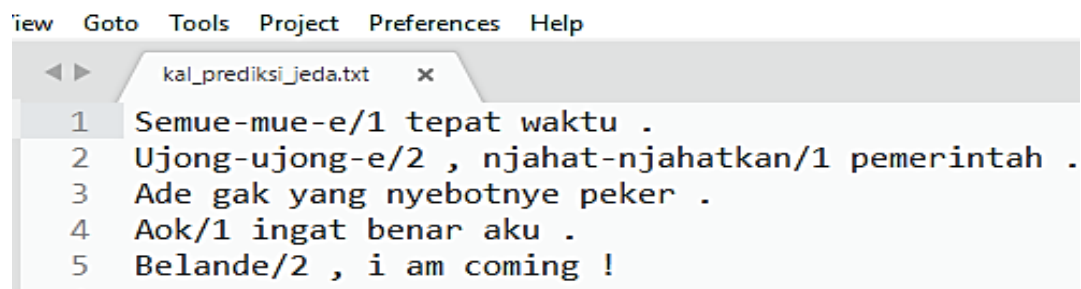
Kode program 4.4 Penandaan jeda berdasarkan *rule* jeda

```

1. for i, x in enumerate(result):
2.     n = i+1
3.
4.     if(checkTree(x) and len(result) > n):
5.         if(checkTree(result[n]) and (x.label()+"-"+result[n].label() in rule_1)):
6.             tmp_txt.append(getText(x)+"/1")
7.
8.         elif((not checkTree(result[n])) and x.label()+"-"+result[n][0] in rule_2):
9.             tmp_txt.append(getText(x)+"/2")
10.        elif (checkTree(result[n]) and (x.label()+"-"+result[n].label() in rule_2)):
11.            tmp_txt.append(getText(x)+"/2")
12.        else:
13.            if(not checkTree(x)):
14.                tmp_txt.append(x[0])
15.        else:
16.            tmp_txt.append(getText(x))
17.        else:
18.            if(not checkTree(x)):
19.                tmp_txt.append(x[0])
20.        else:
21.            tmp_txt.append(getText(x))
22.    hasil = " ".join(tmp_txt)
23.    i = i + 1

```

Hasil penandaan indeks jeda berupa teks kalimat dengan jeda singkat dan jeda panjang yang disimpan dalam file format *.txt.



Gambar 4.59 Output penandaan indeks jeda

Pemberikan indeks jeda dibuat untuk memudahkan perbandingan kalimat dari penutur dan kalimat dari hasil *rule grammar* dan *rule jeda*. Perbandingan jeda dapat dilihat pada tabel 4.5.


Tabel 4.5 Perbandingan jeda penutur dan jeda *shallow parsing*

No.	Jeda Penutur	Jeda <i>Shallow Parsing</i>
1.	Aoklah-aoklah/2 aku ngakulah/2 kalok pemerintah yang tak tepat waktu .	Aoklah-aoklah/1 aku ngakulah/1 kalok pemerintah yang tak tepat waktu .
2.	Dah panjang lebar aku beleter/1 lalu apelah/1 kate die ? .	Dah panjang lebar aku beleter/1 lalu apelah/1 kate die ? .
3.	Sekarang/1 udah/1 ku tengok /1 ngan mate kepala aku sorang .	sekarang udah ku tengok/1 ngan mate kepala aku sorang .

Berdasarkan tabel 4.5, penggalan frasa pada kalimat nomor 1 dan nomor 2 sesuai. Potongan frasa kalimat nomor 3 tidak sesuai, karena pada jeda penutur, kata “sekarang” dan “udah” dipisah, sedangkan berdasarkan *rule grammar* yang dimasukkan dalam *shallow parsing*, sekarang dan udah menjadi 1 frasa. Ini dikarenakan berdasarkan analisis pengembangan *rule*, kata keterangan seperti “sekarang” dan “udah” yang tergolong dalam label TRB banyak diucapkan menjadi satu frasa ucapan.

4.1.8 Hasil Implementasi Antarmuka Prediksi Jeda

Prediksi jeda ditampilkan dalam halaman *web* untuk memudahkan pengguna melakukan prediksi jeda. Antarmuka dari prediksi jeda dapat dilihat pada gambar 4.60.

 **PREDIKSI JEDA DALAM UCAPAN KALIMAT BAHASA MELAYU PONTIANAK**

- Masukkan kalimat dibawah ini

Semue-mue-e tepat waktu.
 Kau bikin janji jam limak, make jam limak kau harus datang.
 Ade kegiatan jam tujuh, make jam tujuh kegiatan tu harus dimulai.
 Tadak macam tabiat kau tu, kau bikin janji jam limak, jam tujuh kau baru datang.
 Ade kegiatan jam tujuh, jam sembilan baru dimulai.
 Tibe-tibe tengah ngasek kamek begaye macam yeye.

PREDIKSI
- Hasil Tagging PoS

Semue-mue-e/PRN tepat/DRB waktu/NNU ./.
 Kau/PRP bikin/VBT janji/NN jam/NN limak/CDP ./, make/CON jam/NN limak/CDP kau/PRP harus/MD datang/VBI ./.
 Ade/VBT kegiatan/NN jam/NN tujuh/CDP ./, make/CON jam/NN tujuh/CDP kegiatan/NN tu/DT harus/MD dimulai/VBI ./.
 Tadak/NEG macam/RP tabiat/NN kau/PRP tu/DT ./, kau/PRP bikin/VBT janji/NN jam/NN limak/CDP ./, jam/NN
 tujuh/CDP kau/PRP baru/RB datang/VBI ./.
 Ade/VBT kegiatan/NN jam/NN tujuh/CDP ./, jam/NN sembilan/CDP baru/RB dimulai/VBI ./.
 Tibe-tibe/RB tengah/TRB ngasek/II kamek/PRP begaye/VBI macam/RP yeye/II ./.
- Hasil Chunk

0 (BP Semue-mue-e/PRN)
 1 (AP2 tepat/DRB waktu/NNU)
 2 .
 0 (VP Kau/PRP bikin/VBT)
 1 (NP janji/NN jam/NN (BP limak/CDP))
 2 ,
 3 (KP make/CON)
- Hasil Prediksi Jeda

Semue-mue-e/1 tepat waktu .
 Kau bikin janji jam limak/2 , make/1 jam limak/1 kau harus datang .
 Ade kegiatan jam tujuh/2 , make/1 jam tujuh/1 kegiatan tu harus dimulai .
 Tadak macam/1 tabiat kau tu/2 , kau bikin janji jam limak/2 , jam tujuh/1 kau baru datang .
 Ade kegiatan jam tujuh/2 , jam sembilan baru dimulai .
 Tibe-tibe/2 tengah ngasek/1 kamek begaye macam yeye .
- Hasil Prediksi Jeda untuk Keperluan Implementasi TTS

Semue-mue-e| tepat waktu .
 Kau bikin janji jam limak || make| jam limak| kau harus datang .
 Ade kegiatan jam tujuh || make| jam tujuh| kegiatan tu harus dimulai .
 Tadak macam| tabiat kau tu || kau bikin janji jam limak || jam tujuh| kau baru datang .
 Ade kegiatan jam tujuh || jam sembilan baru dimulai .
 Tibe-tibe || tengah ngasek| kamek begaye macam yeye .

Gambar 4.60 Antarmuka Sistem Prediksi Jeda

Pada gambar 4.60, dapat dilihat terdapat lima buah kolom dan sebuah tombol pada halaman *web* prediksi jeda. Kolom pertama berfungsi sebagai tempat

untuk memasukkan teks kalimat bahasa Melayu Pontianak yang ingin diprediksi jeda. Tombol prediksi berfungsi untuk memulai proses prediksi jeda. Kolom kedua berfungsi untuk menampilkan hasil dari proses *tagging* PoS. Kolom ketiga berfungsi untuk menampilkan hasil dari proses *shallow parsing*. Kolom keempat berfungsi untuk menampilkan hasil dari proses penandaan jeda berdasarkan *rule* jeda. Kolom terakhir berfungsi untuk menampilkan hasil akhir dari prediksi jeda berupa teks kalimat yang dapat digunakan untuk keperluan implementasi *text-to-speech*.

Proses prediksi jeda dimulai dari memasukkan teks kalimat bahasa Melayu Pontianak kemudian menekan tombol Prediksi. Selanjutnya tiap proses menghasilkan *output* seperti yang dijelaskan pada poin berikut ini.

4.1.8.1 Penambahan Spasi pada Kalimat

Kalimat masukan diolah untuk dilakukan penambahan spasi pada penulisan tanda koma (,), tanda titik(.), tanda seru (!) dan tanda tanya(?). Penambahan spasi dilakukan untuk kalimat yang diketik tanpa menambahkan spasi pada tanda baca, misalnya : “Pergi lah sana.” diubah menjadi “Pergi lah sana .”. Hal ini diperlukan untuk keperluan *tagging* PoS pada IPOSTagger yang mengenali tanda baca sebagai label PoS juga. Penambahan spasi pada pada kalimat dapat dilihat pada kode program 4.5.

Kode program 4.5 Penambahan spasi pada kalimat

```

1.     def simpan_file(kalimat):
2.         tokens = [',', '!', '?']
3.
4.         for token in tokens:
5.
6.             kalimat = kalimat.replace(token, " "+token+" ")
7.             kalimat = re.sub(r"\s+", " ", kalimat)
8.             kalimat = re.sub(r"\s$", "", kalimat)
9.             simpan_file = open('IPOSTagger\kalimatinput.tag', 'w')
10.            simpan_file.write(kalimat)
11.            simpan_file.close()

```

Setelah penambahan spasi dilakukan pada baris dua sampai delapan, pada baris sembilan, hasil proses ini kemudian disimpan di file “kalimatinput.tag” di folder IPOSTagger untuk diproses menggunakan IPOSTagger.

Pada gambar 4.61, hasil *tagging* PoS ditampilkan di kolom kedua. Hasil ini yang dipakai di proses *shallow parsing* pada tahap selanjutnya.

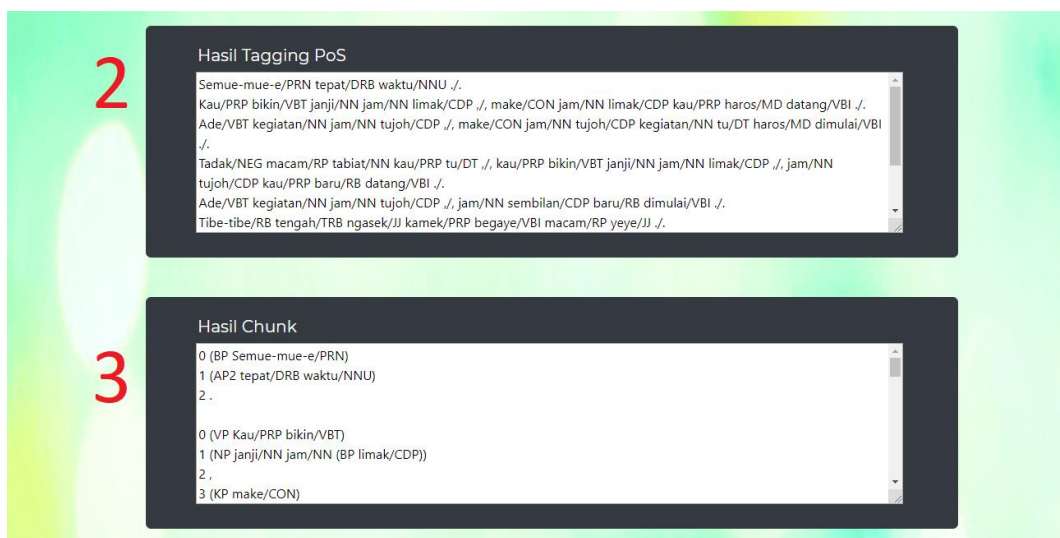
4.1.8.3 Shallow Parsing

Hasil dari *tagging* PoS kemudian diproses menggunakan metode *shallow parsing*. Proses *shallow parsing* memanggil bantuan file kode program *python* yang telah dirancang untuk menjalankan proses *chunking* kalimat.

Kode program 4.7 Menjalankan proses *chunking*

```
shell_exec("python prediksi_jeda.py");
```

Kode program ini sesuai dengan penjelasan pada bagian 4.1.5 tentang Implementasi *Shallow Parsing*. Hasil dari proses ini ditampilkan seperti pada gambar dibawah ini.



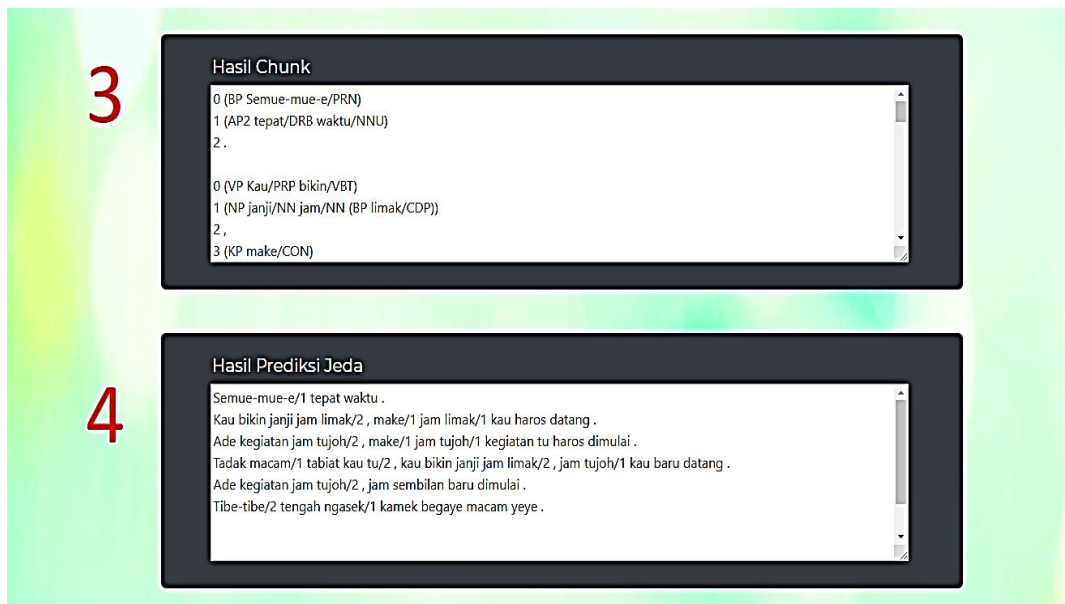
Gambar 4.62 Tampilan output proses shallow parsing

Hasil dari *tagging* PoS di kolom kedua pada gambar 4.62 diolah menggunakan proses *shallow parsing* menghasilkan *chunk* atau penggalan kalimat berupa tipe frasa sesuai *rule grammar* yang telah dibuat. Hasil *chunk* ditampilkan di kolom ketiga.

4.1.8.4 Penandaan Jeda

Hasil *chunk* kemudian diproses menggunakan *rule* jeda sesuai penjelasan pada bagian 4.1.7 tentang Implementasi *Rule* Jeda. Pada tahap ini, hasil *chunk* diperiksa menggunakan *rule* jeda sehingga hasil akhir proses berupa teks kalimat

bahasa Melayu Pontianak dengan penandaan jeda singkat dan jeda panjang seperti pada gambar 4.63 dibawah ini.



Gambar 4.63 Tampilan *output* proses penandaan jeda

Hasil *chunk* di kolom ketiga di proses untuk ditandai jeda singkat dan jeda panjang sehingga hasil akhir berupa teks kalimat bahasa Melayu Pontianak dengan penandaan jeda singkat dan jeda panjang seperti pada kolom nomor empat.

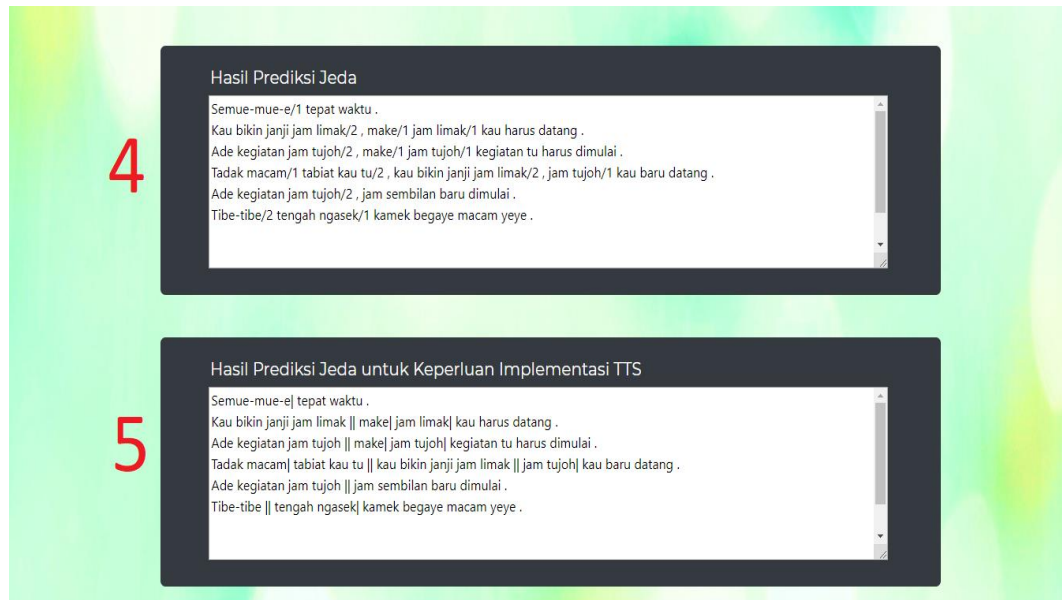
4.1.8.5 Normalisasi Hasil Jeda untuk Keperluan Sistem TTS

Pada tahap ini dilakukan normalisasi hasil prediksi jeda untuk mengubah tanda jeda singkat(/1) menjadi tanda “|” dan mengubah tanda jeda panjang (/2) menjadi tanda “||” untuk keperluan implementasi sistem *Text-to-Speech*.

Kode program 4.8 Normalisasi Hasil Jeda

1. `with open('kal_prediksi_jeda.txt', 'r') as file :`
2. `filedata = file.read()`
3. `filedata = filedata.replace('/1', '|')`
4. `filedata = filedata.replace('/2', '||')`
5. `filedata = filedata.replace('/2', '||')`
6. `with open('kal_prediksi_tts.txt', 'w+') as file:`
7. `file.write(filedata)`

Kode program 4.8 dipakai untuk normalisasi hasil jeda sehingga hasil akhir dari prediksi jeda berupa teks kalimat bahasa Melayu Pontianak dengan tanda baca | dan || untuk jeda. Hasil akhir dapat dilihat pada gambar 4.64 dibawah ini.



Gambar 4.64 Tampilan *output* hasil akhir prediksi jeda

Hasil prediksi jeda pada kolom keempat kemudian dinormalisasi menjadi hasil akhir prediksi jeda yang digunakan untuk keperluan implementasi sistem TTS seperti pada kolom kelima.

4.2 Hasil Pengujian

4.2.1 Hasil Pengujian Kecocokan Frasa Jeda dalam Satu Kalimat Penuh

Pengujian kecocokan frasa jeda dilakukan untuk melihat persamaan kemunculan kejadian jeda pada kalimat asli korpus yang sudah ditandai kejadian jeda sesuai ucapan penutur dan kalimat hasil prediksi dari *shallow parsing*. Total kalimat yang diuji sebanyak 500 kalimat dari kalimat penutur dan 500 kalimat hasil *shallow parsing*. Ada dua pengujian yang dilakukan yaitu pengujian menggunakan *rule grammar* dan *rule jeda* yang baru yang dibandingkan dengan pengujian *rule grammar* dari penelitian sebelumnya (Kamiludin, 2017). Tabel pengujian yang dipakai mengacu pada tabel 3.7 pada Bab 3 untuk rencana pengujian pada frasa jeda ucapan dalam satu kalimat. Hasil perhitungan tiap pengujian dimasukkan ke dalam persamaan 3.1 pada Bab 3 :

$$\text{Akurasi frasa jeda dalam kalimat} = \frac{\text{jumlah kalimat yang sesuai}}{\text{Total kalimat yang diuji}} \times 100\%$$

4.2.1.1 Hasil Pengujian Akurasi Kecocokan Frasa Jeda dengan *Rule Grammar* dan *Rule Jeda Baru*

Hasil pengujian akurasi kecocokan frasa jeda dengan *rule grammar* dan *rule jeda* yang baru dapat dilihat pada tabel 4.6.

Tabel 4.6 Perbandingan jeda penutur dan jeda shallow parsing

Tipe Pengujian	Jumlah Kalimat	Akurasi (%)
Kemunculan jeda singkat dan panjang	500	33.6%
Kemunculan jeda panjang	500	72.8%

Dari hasil pengujian akurasi kecocokan jeda antara jeda dari penutur dan jeda dari *shallow parsing* didapatkan nilai akurasi dari total 500 kalimat dengan akurasi pada kemunculan jeda singkat dan panjang sebesar 33.6% atau sebanyak 168 kalimat. Kemunculan jeda panjang memiliki akurasi sebesar 72.8% atau sebanyak 364 kalimat. Hasil lengkap pengujian dapat dilihat pada Lampiran A.

4.2.1.1.1 Analisa Hasil Pengujian Kecocokan Frasa Jeda dalam Satu Kalimat Penuh dengan *Rule Grammar* dan *Rule Jeda yang Baru*

Hasil pengujian menyatakan bahwa nilai akurasi terbaik pada tipe pengujian kemunculan jeda panjang. Tingginya nilai akurasi kemunculan jeda panjang dikarenakan menurut hasil analisa jeda penutur, pola jeda penutur ketika berbicara cenderung di posisi atau frasa yang sama dari 500 kalimat. Sedangkan nilai pengujian kemunculan jeda singkat dan panjang menghasilkan nilai akurasi yang rendah, karena pola lama berhentinya penutur untuk jeda singkat bervariasi dari 500 kalimat sehingga *rule grammar* dan *rule jeda* yang dikembangkan belum dapat menangkap jeda singkat dengan baik.

4.2.1.2 Pengujian Akurasi Kecocokan Frasa Jeda dengan *Rule Grammar* dari Penelitian Sebelumnya (Kamiludin, 2017)

Hasil pengujian akurasi kecocokan frasa jeda dengan *rule grammar* dan *rule* jeda yang baru dapat dilihat pada tabel 4.7.

Tabel 4.7 Perbandingan jeda penutur dengan jeda prediksi *rule* penelitian sebelumnya

Tipe Pengujian	Jumlah Kalimat	Akurasi (%)
Kemunculan jeda singkat dan panjang	500	10.6%

Dari hasil pengujian akurasi kecocokan jeda antara jeda dari penutur dan jeda dari *shallow parsing* dengan *rule grammar* dari penelitian sebelumnya didapatkan nilai akurasi dari total 500 kalimat dengan akurasi pada kemunculan jeda singkat dan panjang sebesar 10.6% atau sebanyak 53 kalimat. Hasil lengkap pengujian dapat dilihat pada Lampiran A.

4.2.1.2.1 Analisa Hasil Pengujian Kecocokan Frasa Jeda dalam Satu Kalimat Penuh dengan *Rule Grammar* Penelitian Sebelumnya (Kamiludin, 2017)

Hasil pengujian menyatakan bahwa nilai akurasi sebesar 10.6% pada pengujian jeda singkat dan jeda panjang. Pengujian jeda panjang tidak dilakukan karena pada penelitian sebelumnya, tidak dilakukan prediksi untuk jeda panjang, sehingga semua jeda dikategorikan sama. Rendahnya nilai akurasi sudah pernah dijelaskan di penelitian sebelumnya (Kamiludin, 2017) yang dikarenakan *rule grammar* yang dibuat berdasarkan pola struktur kalimat S-P-O-K. Hal ini yang menyebabkan *rule grammar* belum bisa membentuk frasa jeda yang sama dengan frasa jeda penutur.

4.2.2 Hasil Pengujian *Precision* , *Recall*, dan *F-Measure* Terhadap Penggalan Frasa

Pengujian *precision*, *recall*, dan *F-Measure* dilakukan dengan membandingkan frasa jeda penutur dan frasa jeda hasil implementasi *rule* jeda .

Pengujian *precision*, *recall* dan *F-Measure* pada penggalan frasa dilakukan untuk melihat persamaan penggalan frasa kalimat terhadap jeda pada kalimat asli korpus yang sudah ditandai kejadian jeda sesuai ucapan penutur dan kalimat hasil prediksi dari *shallow parsing*. Pengujian terhadap penggalan frasa dilakukan sebanyak dua kali, yaitu pengujian terhadap penggalan frasa berdasarkan jeda singkat dan penggalan frasa terhadap jeda panjang.

4.2.2.1 Hasil Pengujian Penggalan Frasa Terhadap Jeda Singkat dan Panjang

Pengujian penggalan frasa terhadap jeda singkat dibagi menjadi lima pengujian yaitu membandingkan kalimat sebanyak 100 kalimat, 200 kalimat, 300 kalimat, 400 kalimat, dan 500 kalimat. Tabel yang dipakai untuk pengujian adalah tabel 3.8-3.12 yang telah dijelaskan di bagian 3.3.10.2.1 tentang Skenario Pengujian Jumlah Kalimat. Tabel rangkuman hasil pengujian dapat dilihat pada tabel 4.8. Untuk hasil lengkap pengujian dapat dilihat di Lampiran C.

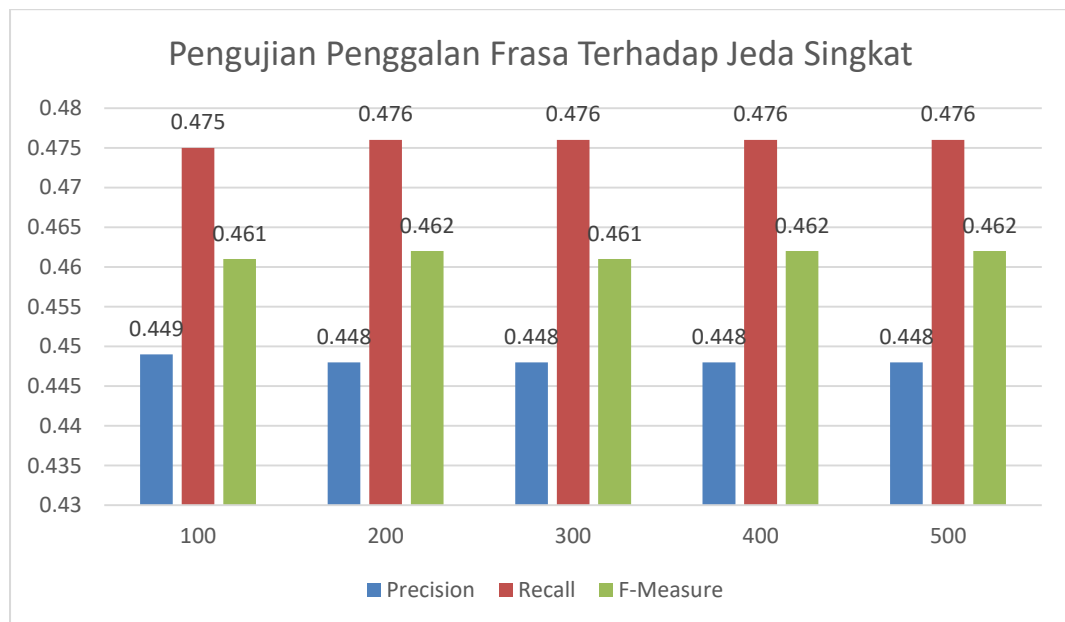
Tabel 4.8 Rangkuman Nilai Pengujian Penggalan Frasa Terhadap Jeda Singkat

No.	Kalimat Korpus Uji	Precision $(\frac{a}{a+c})$	Recall $(\frac{a}{a+b})$	F-Measure $(2 \cdot \frac{p \cdot r}{p+r})$
1.	100	0.449	0.475	0.461
2.	200	0.448	0.476	0.462
3.	300	0.448	0.476	0.461
4.	400	0.448	0.476	0.462
5.	500	0.448	0.477	0.462

Dari hasil pengujian *precision*, *recall*, dan *F-Measure* penggalan frasa terhadap jeda singkat dari 100, 200, 300, 400 dan 500 kalimat didapatkan nilai F-Measure pada 100 kalimat dan 300 kalimat sebesar 0.461, sedangkan pada 200, 300, dan 500 kalimat memiliki nilai sebesar 0.462. Nilai *precision* pada pengujian 100 kalimat sebesar 0.449 dan pada pengujian 200-500 kalimat sebesar 0.448. Pengujian *recall* menghasilkan nilai 0.475 untuk pengujian 100 kalimat, nilai 0.476 untuk pengujian 200-400 kalimat dan nilai 0.477 untuk pengujian 500 kalimat.

4.2.2.1.1 Analisis Hasil Pengujian Penggalan Frasa Terhadap Jeda Singkat

Hasil pengujian menyatakan bahwa nilai akurasi memiliki rata-rata yang hampir sama untuk pengujian 100-500 kalimat dengan kisaran nilai 0.46. Tampilan hasil pengujian penggalan frasa terhadap jeda singkat dapat dilihat pada gambar grafik 4.65 berikut.



Gambar 4.65 Grafik Hasil Pengujian Penggalan Frasa terhadap Jeda Singkat dan Panjang

Nilai pada pengujian yang berada dikisaran 0.4 disebabkan jeda singkat prediksi *shallow parsing* banyak yang tidak akurat dilihat dari nilai *precision* yang rendah. Banyak frasa tidak relevan atau frasa jeda yang belum benar terbentuk. Frasa jeda yang salah ini dikarenakan *rule grammar* membentuk frasa sesuai tipe pos yang muncul di kalimat. Seperti yang telah dijelaskan di poin 4.2.1.1.1, jeda singkat memiliki pola jeda yang bervariasi dari penutur yang menyebabkan kemunculan jeda menjadi tidak sama.

4.2.2.2 Hasil Pengujian Penggalan Frasa Terhadap Jeda Panjang

Pengujian penggalan frasa terhadap jeda panjang dibagi menjadi lima pengujian yaitu membandingkan kalimat sebanyak 100 kalimat, 200 kalimat, 300 kalimat, 400 kalimat, dan 500 kalimat. Tabel yang dipakai untuk pengujian adalah

tabel 3.8-3.12 yang telah dijelaskan di bagian 3.3.10.2.1 tentang Skenario Pengujian Jumlah Kalimat. Tabel rangkuman hasil pengujian dapat dilihat pada tabel 4.9. Untuk hasil lengkap pengujian dapat dilihat di Lampiran D.

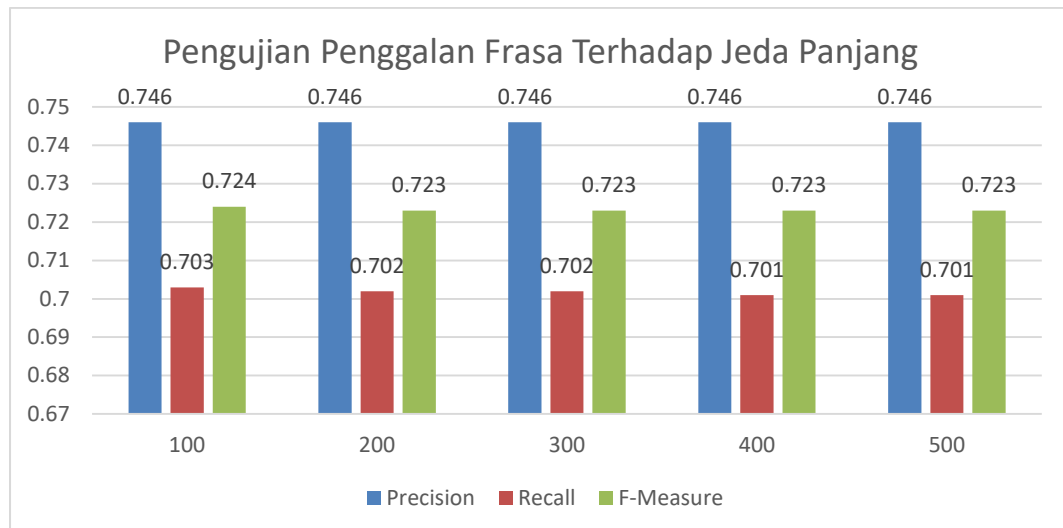
Tabel 4.9 Rangkuman Nilai Pengujian Potongan Frasa Terhadap Jeda Panjang

No.	Kalimat Korpus Uji	Precision $(\frac{a}{a+c})$	Recall $(\frac{a}{a+b})$	F-Measure $(2 \cdot \frac{p \cdot r}{p+r})$
1.	100	0.746	0.703	0.724
2.	200	0.746	0.702	0.723
3.	300	0.746	0.702	0.723
4.	400	0.746	0.701	0.723
5.	500	0.746	0.701	0.723

Dari hasil pengujian *precision*, *recall*, dan *F-Measure* penggalan frasa terhadap jeda panjang dari 100, 200, 300, 400 dan 500 kalimat didapatkan nilai F-Measure pada 100 kalimat sebesar 0.724, dan pada 200 -500 kalimat 0.723. Pengujian *precision* memiliki nilai sebesar 0.724. Untuk pengujian *recall*, pada 100 kalimat memiliki nilai sebesar 0.703, pada 200 dan 300 kalimat sebesar 0.702, dan pada 400 dan 500 kalimat sebesar 0.701.

4.2.2.2.1 Analisis Hasil Pengujian Penggalan Frasa Terhadap Jeda Panjang

Hasil pengujian menyatakan bahwa nilai akurasi memiliki rata-rata yang hampir sama dengan kisaran nilai 0.72. Tampilan hasil pengujian penggalan frasa terhadap jeda panjang dapat dilihat pada gambar grafik 4.66 berikut.



Gambar 4.66 Grafik Hasil Pengujian Penggalan Frasa Terhadap Jeda Panjang

Nilai pengujian *precision* menunjukkan angka yang sama di nilai 0.746 yang berarti *rule grammars* dan *rule jeda* berhasil memprediksi penggalan frasa yang tepat untuk semua kalimat baik dari 100 kalimat sampai 500 kalimat. Nilai *recall* lebih bervariasi, karena masih ada penggalan frasa yang tidak sesuai dengan frasa jeda penutur karena *rule* yang belum cocok. Nilai *f-measure* memiliki nilai yang hampir sama dan tergolong baik yakni 0.72. Prediksi jeda panjang memiliki nilai yang lebih baik dan tinggi, karena berdasarkan jeda penutur, letak jeda panjang cenderung memiliki pola berhenti di suatu frasa yang sama, sehingga *rule grammar* dan *rule jeda* yang dibuat dapat memprediksi jeda dengan baik.

4.3 Analisa Hasil Keseluruhan Pengujian

Berdasarkan hasil pengujian akurasi kecocokan frasa jeda dalam satu kalimat, nilai akurasi prediksi jeda telah meningkat. Perbandingan nilai akurasi dapat dilihat dimana hasil penerapan *rule grammar* yang lama menunjukkan nilai akurasi sebesar 10.6% atau sebanyak 53 kalimat dari 500 kalimat untuk prediksi jeda pada 500 kalimat uji yang dipakai di penelitian ini (dapat dilihat di lampiran B). Sedangkan nilai akurasi yang didapat berdasarkan penerapan *rule grammar* yang baru dan *rule jeda* menunjukkan nilai sebesar 33.6% untuk jeda singkat dan jeda panjang atau sebanyak 168 kalimat dari 500 kalimat uji. Untuk jeda panjang memiliki nilai akurasi sebesar 70.2% atau sebanyak 364 kalimat dari 500 kalimat

uji. Nilai akurasi meningkat sebesar 23% dari 53 kalimat menjadi 168 kalimat dilihat dari pengujian jeda singkat dan panjang.

Berdasarkan hasil pengujian dengan menghitung nilai *precision*, *recall* dan *F-measure* dari kemunculan jeda dan penggalan frasa jeda, analisis yang dapat disimpulkan yakni nilai akurasi prediksi jeda *shallow parsing* untuk jeda singkat dan jeda panjang berbeda. Berdasarkan tabel 4.6 – 4.8, nilai akurasi terbaik dihasilkan pada prediksi jeda panjang. Hal ini dikarenakan, jeda panjang lebih mudah diterima daripada jeda singkat penutur sehingga *rule grammar* dan *rule* jeda yang terbentuk bisa mendapatkan nilai yang sama dengan jeda penutur. Akurasi yang rendah pada jeda singkat, dikarenakan *rule grammars* untuk *shallow parsing* yang dirancang masih belum sempurna untuk menghasilkan potongan frasa jeda. Hal ini juga disebabkan pemberian kelas kata yang belum sempurna pada kalimat sehingga *rule grammars* yang dibuat belum bisa memotong frasa secara akurat sesuai frasa penutur.

Tabel 4.10 Perbandingan Hasil Frasa Jeda

No.	Jeda dari Penutur	Jeda <i>Shallow Parsing</i>
1.	kame/PRP ni/DT jaim/VBI/1 tang/IN atas/NN kapal/NNC	kame/PRP ni/DT jaim/VBI/1 tang/IN atas/NN kapal/NNC
2.	Naekan/VBT ke/IN atas/NN kapal/NNC klotok/NNC.	Naekan/VBT/1 ke/IN atas/NN kapal/NNC klotok/NNC.

Berdasarkan tabel 4.10, dapat dilihat perbedaan potongan jeda penutur dan *shallow parsing*. Pada kalimat pertama, setelah kata dengan label PoS kata kerja (VBI) terjadi jeda singkat sebelum diikuti kata depan “tang” yang memiliki label PoS IN. *Rule grammars* dan *rule* jeda berhasil memprediksi hasil yang sama dengan frasa jeda dari penutur. Sedangkan pada kalimat kedua, pada frasa penutur, untuk kata kerja VBT dan IN tidak terjadi jeda, sehingga hasil prediksi menjadi tidak sesuai.

Rule grammar dan *rule* jeda yang telah dibuat pada penelitian ini dikhususkan untuk memprediksi jeda pada kalimat bahasa Melayu Pontianak. *Rule* yang telah dibentuk tidak dapat digunakan untuk memprediksi jeda pada kalimat bahasa lain.

Berdasarkan pengujian pada 15 kalimat bahasa Indonesia yang diterjemahkan dari bahasa Melayu Pontianak pada lampiran E, hanya ada lima kalimat yang prediksi jedanya sama sedangkan 10 kalimat lain tidak dapat diprediksi dengan baik. Kesalahan prediksi ini disebabkan oleh tag PoS yang tidak cocok pada kalimat bahasa Indonesia, dimana tag PoS yang dipakai adalah tag PoS bahasa Melayu Pontianak. *Rule grammar* yang dibentuk berdasarkan frasa jeda ucapan bahasa Melayu Pontianak, sehingga *rule grammar* juga tidak dapat memenggal kalimat selain bahasa Melayu Pontianak dengan baik. Struktur kalimat yang beda juga menjadi salah satu faktor *rule* tidak dapat digunakan dengan baik. Seperti pada contoh kalimat lampiran E nomor 12 :

“Rele mereke ngancorkan beberapa bangunan tang tengah kote untuk dijadikan tempat parkir.”

Pada kalimat bahasa lain, misalnya bahasa Indonesia, kalimat cenderung diikuti oleh subjek “mereka” terlebih dahulu, sehingga jika diterjemahkan ke dalam bahasa Indonesia menjadi :

“Mereka rela menghancurkan beberapa bangunan di tengah kota untuk dijadikan tempat parkir.”.

Hal ini membuat letak PoS berubah dan *rule* yang dibuat tidak bisa menghasilkan jeda yang diinginkan.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil analisis dan pengujian terhadap prediksi frasa jeda ucapan kalimat bahasa Melayu Pontianak dengan menggunakan *shallow parsing* dengan pengembangan *rule grammar*, maka dapat ditarik kesimpulan sebagai berikut:

1. Prediksi jeda ucapan dengan metode *shallow parsing* yang sudah dikembangkan *rule grammar* dan *rule* jeda sesuai ucapan penutur memiliki akurasi untuk kecocokan jeda dalam satu kalimat sebesar 33.6% untuk jeda singkat dan panjang, atau sebanyak 168 kalimat dari 500 kalimat . Nilai akurasi sebesar 70.2% untuk jeda panjang atau sebanyak 364 kalimat dari 500 kalimat. Nilai akurasi meningkat sebesar 23% jika dibandingkan dengan prediksi *rule* berdasarkan struktur kalimat yang telah dilakukan pada penelitian sebelumnya dimana hasil akurasi yang didapatkan ketika diuji di 500 kalimat yang sama sebesar 10.6% atau sebanyak 53 kalimat dari 500 kalimat. Hasil lengkap perbandingan pengujian dapat dilihat di Lampiran A dan Lampiran B.
2. Prediksi jeda ucapan dengan metode *shallow parsing* yang sudah dikembangkan *rule grammar* dan *rule* jeda memiliki nilai akurasi yang baik yaitu nilai akurasi jeda untuk jeda panjang sebesar 72.8% . Jeda panjang memiliki nilai *precision* dan *recall* yang cukup tinggi yaitu sebesar 0.74 dan 0.70 dan nilai *f-measure* sebesar 0.723.

5.2 Saran

Adapun beberapa hal yang perlu ditambahkan dalam pengembangan penentuan frasa jeda ucapan dengan metode *shallow parsing* ini adalah sebagai berikut :

1. Perlu adanya pengembangan PoS yang lebih spesifik lagi untuk pelabelan kelas kata dalam kalimat supaya pengembangan *rule grammars* dan *rule* jeda yang dihasilkan jadi lebih akurat dan dapat menyesuaikan jeda penutur untuk prediksi jeda singkat dan dapat diimplementasikan untuk memprediksi bahasa lain selain bahasa Melayu Pontianak.
2. Perlu penambahan sumber *file* suara dari sumber pembicara yang berbeda untuk memperkaya korpus indeks jeda.
3. Perlu pengembangan sistem *text to speech* bahasa Melayu Pontianak dengan memanfaatkan informasi jeda yang dihasilkan *shallow parsing* dalam penelitian ini untuk pembangkitan suara oleh sistem TTS agar lebih alami.
4. Perlu dilanjutkan penelitian untuk menyederhanakan *rule* dengan mengkombinasikan *rule chunking* dan *rule* jeda.