



TUGAS AKHIR - KS 141501

**RANCANG BANGUN EKSTRAKSI EKSPRESI KATA KERJA  
PADA ULASAN PELANGGAN DENGAN *TEXT CHUNKING*  
UNTUK MEMAPARKAN PENGALAMAN PENGGUNAAN  
PRODUK**

***DEVELOPMENT OF VERB EXPRESSIONS EXTRACTION  
FROM CONSUMER REVIEWS USING TEXT CHUNKING  
TO EXPOSE PRODUCT'S USER EXPERIENCE***

ALMIRA FIANA DHARA H  
NRP 5213 100 184

Dosen Pembimbing :  
Rully Agus Hendrawan, S.Kom., M.Eng.

JURUSAN SISTEM INFORMASI  
Fakultas Teknologi Informasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017





**ITS**  
Institut  
Teknologi  
Sepuluh Nopember

**TUGAS AKHIR - KS 141501**

**RANCANG BANGUN EKSTRAKSI EKSPRESI KATA KERJA  
PADA ULASAN PELANGGAN DENGAN *TEXT CHUNKING*  
UNTUK MEMAPARKAN PENGALAMAN PENGGUNAAN  
PRODUK**

ALMIRA FIANA DHARA H  
NRP 5213 100 184

Dosen Pembimbing :  
Rully Agus Hendrawan, S.Kom., M.Eng.

JURUSAN SISTEM INFORMASI  
Fakultas Teknologi Informasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017



**ITS**  
Institut  
Teknologi  
Sepuluh Nopember

**FINAL PROJECT - KS 141501**

***DEVELOPMENT OF VERB EXPRESSIONS EXTRACTION  
FROM CONSUMER REVIEWS USING TEXT CHUNKING  
TO EXPOSE PRODUCT'S USER EXPERIENCE***

**ALMIRA FIANA DHARA H  
NRP 5213 100 184**

**SUPERVISOR:  
Rully Agus Hendrawan, S.Kom., M.Eng.**

**DEPARTMENT OF INFORMATION SYSTEMS  
Faculty of Information Technology  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017**

## LEMBAR PENGESAHAN

### RANCANG BANGUN EKSTRAKSI EKSPRESI KATA KERJA PADA ULASAN PELANGGAN DENGAN *TEXT CHUNKING* UNTUK MEMAPARKAN PENGALAMAN PENGUNAAN PRODUK

#### TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat  
Memperoleh Gelar Sarjana Komputer  
pada

Jurusan Sistem Informasi  
Fakultas Teknologi Informasi  
Institut Teknologi Sepuluh Nopember

Oleh:

**ALMIRA FIANA DHARA H**  
**NRP. 5213 100 184**

Surabaya, 13 Juni 2017

**KETUA JURUSAN SISTEM INFORMASI**



**Dr. Ir. Aris Tjahyanto, M.Kom**  
**NIP.19650310 199102 1 001**

## LEMBAR PERSETUJUAN

### RANCANG BANGUN EKSTRAKSI EKSPRESI KATA KERJA PADA ULASAN PELANGGAN DENGAN *TEXT CHUNKING* UNTUK MEMAPARKAN PENGALAMAN PENGUNAAN PRODUK

#### TUGAS AKHIR

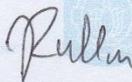
Disusun Untuk Memenuhi Salah Satu Syarat  
Memperoleh Gelar Sarjana Komputer  
pada

Jurusan Sistem Informasi  
Fakultas Teknologi Informasi  
Institut Teknologi Sepuluh Nopember  
Oleh :

**ALMIRA FIANA DHARA HARAHAP**  
**NRP. 5213 100 184**

Disetujui Tim Penguji : Tanggal Ujian : 05 Juli 2017  
Periode Wisuda : September 2017

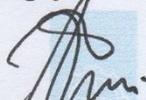
**Rully Agus Hendrawan, S.Kom., M.Eng**

  
(Pembimbing I)

**Mahendrawati ER, ST, M.Sc, Ph.D**

  
(Penguji I)

**Erma Suryani, ST, MT, Ph.D**

  
(Penguji II)

# **RANCANG BANGUN EKSTRAKSI EKSPRESI KATA KERJA PADA ULASAN PELANGGAN DENGAN *TEXT CHUNKING* UNTUK MEMAPARKAN PENGALAMAN PENGUNAAN PRODUK**

**Nama Mahasiswa** : Almira Fiana Dhara H  
**NRP** : 5213 100 184  
**Jurusan** : Sistem Informasi FTIf-ITS  
**Pembimbing 1** : Rully Agus Hendrawan, S.Kom., M.Eng.

## **ABSTRAK**

*Saat ini, hampir seluruh situs e-commerce memiliki fitur ulasan pelanggan mengenai produk atau jasa yang telah dibeli dan dapat dibaca oleh seluruh pengguna. Seiring berjalannya waktu, ulasan pelanggan tersebut terus bertambah semakin banyak, terutama pada produk dan jasa yang populer. Sehingga, e-commerce menambahkan beberapa fitur seperti rating, ulasan positif dan netral populer, dan beberapa fitur lainnya untuk memudahkan pengguna. Tetapi, hal yang luput diperhatikan oleh e-commerce adalah e-commerce tidak menonjolkan pengalaman seperti apa yang didapat pengguna saat menggunakan produk atau jasa. Padahal, pengalaman tersebut merupakan informasi penting bagi pengguna agar mengetahui kegunaan dan manfaat dari produk dan jasa tersebut. Banyaknya ulasan pengguna juga membuat pelanggan sulit memahami pengalaman penggunaan produk yang dirasakan karena membaca banyak ulasan tersebut cukup melelahkan.*

*Oleh karena itu, diperlukan fitur yang dapat menampilkan pengalaman penggunaan produk secara ringkas sehingga pelanggan dapat mengetahui secara umum kegunaan dan manfaat dari produk tersebut. Pada penelitian ini, pengalaman penggunaan produk terbatas pada aktivitas yang dilakukan pengguna ketika memakai produk tersebut. Metode yang akan digunakan adalah membuat prototipe dengan Text Chunking untuk*

*mengekstrak informasi mengenai pengalaman penggunaan produk dari ulasan pelanggan. Luaran yang diharapkan dari penelitian ini adalah perangkat lunak yang menampilkan daftar kata kerja yang sering muncul pada ulasan sebagai representasi aktivitas yang dilakukan pengguna ketika memakai produk.*

*Pada penelitian ini ditemukan bahwa ekstraksi ekspresi kata kerja menggunakan text chunking masih kurang efektif dikarenakan kalimat yang dihasilkan masih kurang sesuai dengan harapan setelah melakukan percobaan dengan menggunakan 3, 4 dan 5 chunks pada ekspresi frase kata kerja yang dihasilkan. Selain itu, tata bahasa dari kalimat yang dihasilkan juga masih kurang tepat dibandingkan dengan ulasan pelanggan yang sesungguhnya. Permasalahan tersebut dikarenakan dataset yang digunakan memiliki ejaan kata dan tata bahasa yang kurang tepat sehingga diharapkan penelitian selanjutnya dapat membuat model yang dapat melakukan chunking yang memiliki akurasi lebih tinggi dan memilih dataset yang tepat.*

***Kata Kunci : Online Reviews, CRM, Text Chunking***

**DEVELOPMENT OF VERB EXPRESSIONS EXTRACTION  
FROM CONSUMER REVIEWS USING TEXT CHUNKING  
TO EXPOSE PRODUCT'S USER EXPERIENCE**

**Student Name** : Almira Fiana Dhara H  
**NRP** : 5213 100 184  
**Department** : Sistem Informasi FTIf-ITS  
**Supervisor 1** : Rully Agus Hendrawan, S.Kom., M.Eng

**ABSTRACT**

*Nowadays, almost all of e-commerce sites has a feature that shows user reviews about a product or service that has been bought and can be read by all of the e-commerce's users. As time goes by, those user reviews keep adding up, especially for products and services that are popular. Because of that, most of e-commerces nowadays have decided to add more features like rating, popular positive reviews and popular neutral reviews, and more features to increase to make their applications become more convinience for users. But, most e-commerce sites have not pay attention to the kind of experiences that the users feel when they use the product or service. When actually, the users' experiences when using the products or services are one of the most important information for all users to make them know the usabilities and benefits from those products or seVICES. The large amount of user reviews are also another reason that makes users have difficulties to understand the products' user experience that the users feel because reading all the reviews is a very exhausting thing to do.*

*Because of all those reasons, we need a feature that can shows products' users experiences briefly so users know generally usabilities and benefits from the product or service. In this research, products' users experience is limited to activities that users do when they use the product. The method that will be used in this research is to make a prototype with Text Chunking to*

*extract information about product users' experiences from user reviews. The output that is expected from this research is to make a software that can show a list of verb expressions that often show up in the product reviews as a representation from the activities which the users do when they use the product.*

*This research has found that verb expressions extraction using text chunking is still not effective enough because of the sentences which are generated from the model is still not suitable with the purpose of this research after we have done three experiments using 3, 4 and 5 chunks which have generated those verb expressions. Another reason is the generated sentences are still not suitable enough from the grammatical perspective compared to the real user reviews. One of the biggest reason is because the dataset that has been used has incorrect grammar and orthography so we hope that the next research can make a better chunker model and choose a suitable dataset.*

***Kata Kunci : Online Reviews, CRM, Text Chunking***

## **KATA PENGANTAR**

Puji dan syukur penulis tuturkan ke hadirat Allah SWT, Tuhan Semesta Alam yang telah memberikan kekuatan dan hidayah-Nya kepada penulis sehingga penulis mendapatkan kelancaran dalam menyelesaikan tugas akhir dengan judul:

### **RANCANG BANGUN EKSTRAKSI EKSPRESI KATA KERJA PADA ULASAN PELANGGAN DENGAN *TEXT CHUNKING* UNTUK MEMAPARKAN PENGALAMAN PENGUNAAN PRODUK**

yang merupakan salah satu syarat kelulusan pada Jurusan Sistem Informasi, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember Surabaya.

Terima kasih penulis sampaikan kepada pihak-pihak yang telah mendukung, memberikan saran, motivasi, semangat, dan bantuan baik berupa materiil maupun moril demi tercapainya tujuan pembuatan tugas akhir ini. Tugas akhir ini tidak akan pernah terwujud tanpa bantuan dan dukungan dari berbagai pihak yang sudah melauangkan waktu, tenaga dan pikirannya. Secara khusus penulis akan menyampaikan ucapan terima kasih yang sebanyak-banyaknya kepada:

- 1) Orang tua dan adik penulis, Bapak Irfan Surya Harahap, SH, Ibu Selfina, SH, MKn, Raissa Maharani Harahap dan Adinda Kamila Harahap yang telah memberikan motivasi, semangat, keyakinan, kasih sayang, kesabaran dalam menghadapi keluh kesah dan menjadi luapan kekesalan penulis, serta doa sehingga penulis mampu menyelesaikan pendidikan S1 ini dengan baik.
- 2) Bapak Rully Agus Hendrawan, S.Kom., M.Eng selaku dosen pembimbing yang telah memberikan bimbingan dan ilmu sehingga penulis dapat menyelesaikan Tugas Akhir ini.

- 3) Ibu Feby Artwodini, S.Kom, MT. selaku dosen wali penulis selama menempuh pendidikan di Jurusan Sistem Informasi yang telah memberikan pengalaman serta nasehat kepada penulis selama ini.
- 4) Ibu Mahendrawati ER, ST, M.Sc, Ph.D dan Ibu Amna Shifia Nisafani, S.Kom, M.Sc. selaku dosen penguji yang telah memberikan kritik, saran, dan masukan yang berharga sehingga dapat menyempurnakan Tugas Akhir ini.
- 5) Seluruh dosen pengajar beserta staf dan karyawan di Jurusan Sistem Informasi, Fakultas Teknologi Informasi ITS Surabaya yang telah memberikan ilmu dan pengalaman yang berharga kepada penulis selama ini.
- 6) Teruntuk sahabat penulis yang telah berada bersama penulis sejak kecil, yang telah melewati berbagai momen bersama, yang tanpa mereka penulis bukanlah orang yang sama seperti penulis sekarang, Dyah Indri Lestari dan Cut Vadea Aqila.
- 7) Sahabat-sahabat penulis yang setia menemani dan membantu penulis ketika penulis membutuhkan semenjak masa Sekolah Menengah Pertama, Nadia Setianingsih, Amalia Azrina dan Devi Fadhullahi Alaika.
- 8) Sahabat penulis yang selalu ada, menemani dan selalu mendukung penulis baik di keadaan senang maupun sulit, Nadya Chandra, Ashma Hanifah, Hanum Fitriani, Faisal Setia Putra, Kevin Setiawan dan Robbigh Faubendri dalam menjalani kehidupan kampus hingga dapat menyelesaikan tugas akhir ini.
- 9) Teman-teman seperjuangan di Laboratorium Sistem Enterprise dan Beltrans yang selalu memberikan dukungan moral bagi penulis untuk menyelesaikan Tugas Akhir ini.
- 10) Rekan penulis, Muhammad Alvin Darari yang tidak pernah lelah mendukung penulis untuk terus giat mengerjakan tugas

akhir, membantu jika terdapat kesulitan, serta menjadi teman untuk berbagi di saat senang maupun susah.

- 11) Serta semua pihak yang telah membantu dalam pengerjaan Tugas Akhir ini yang belum mampu penulis sebutkan diatas.

Terima kasih atas segala bantuan, dukungan, serta doa yang telah diberikan. Penulis menyadari bahwa tugas akhir ini masih belum sempurna dan memiliki banyak kekurangan di dalamnya. Oleh karena itu, penulis juga memohon maaf atas segala kesalahan penulis buat dalam buku tugas akhir ini. Penulis membuka pintu selebar-lebarnya bagi pihak yang ingin memberikan kritik maupun saran, serta penelitian selanjutnya yang ingin menyempurnakan karya dari tugas akhir ini. Semoga buku tugas akhir ini bermanfaat bagi seluruh pembaca.

Surabaya, Januari 2017

*Halaman ini sengaja dikosongkan*

## DAFTAR ISI

ABSTRAK.....	v
ABSTRACT.....	vii
KATA PENGANTAR .....	ix
DAFTAR ISI.....	xiii
DAFTAR GAMBAR .....	xvi
DAFTAR KODE.....	xviii
DAFTAR TABEL.....	xx
BAB I PENDAHULUAN .....	23
1.1. Latar Belakang.....	23
1.2. Perumusan Masalah .....	25
1.3. Batasan Masalah .....	25
1.4. Tujuan Tugas Akhir .....	25
1.5. Manfaat Tugas Akhir .....	26
1.6. Relevansi Tugas Akhir.....	26
BAB II TINJAUAN PUSTAKA .....	29
1.7. Penelitian Sebelumnya.....	29
1.7.1. Ulasan Pelanggan Online .....	32
1.7.2. E-Commerce .....	33
1.7.3. Ekstraksi Ekspresi Kata Kerja .....	35
1.7.4. Part Of Speech Tagging (POS Tagging).....	37
1.7.5. Usability Testing .....	41
1.7.6. Apache OpenNLP .....	41
BAB III METODOLOGI .....	43
2.1. Tahapan .....	45
2.1.1. Studi Literatur .....	45
2.1.2. Preproses Data.....	46
2.1.3. Pengembangan Prototipe .....	46
2.1.4. Hasil dan Pembahasan.....	47
BAB IV PERANCANGAN .....	49
3.1. Perancangan Pembuatan Prototipe .....	49
3.1.1. Perancangan Pengambilan Data .....	49
3.1.2. Perancangan Preproses Data.....	59
3.1.3. Perancangan Pengembangan Prototipe .....	65
3.2. Perancangan Validasi Prototipe .....	76
BAB V IMPLEMENTASI .....	79

4.1.	Perangkat Penelitian.....	79
4.2.	Ekstraksi Data Ulasan Pelanggan.....	80
4.3.	Implementasi Praproses Data.....	87
4.3.1.	Memuat Data Teks.....	88
4.3.2.	Melakukan Segmentasi Kalimat.....	90
4.3.3.	Melakukan Tokenisasi.....	92
4.4.	Implementasi Pengembangan Prototipe.....	94
4.4.1.	Melakukan Labeling dengan POS Tagging....	94
4.4.2.	Implementasi Pemodelan Chunk Parser.....	99
4.4.3.	Implementasi Validasi Pemodelan Chunk Parser.....	102
4.4.4.	Implementasi Sorting Data.....	108
4.5.	Implementasi Usability Testing.....	136
<b>BAB VI HASIL DAN PEMBAHASAN.....</b>		<b>139</b>
5.1.	Hasil dan Pembahasan Ekstraksi Data Ulasan Pelanggan.....	139
5.2.	Hasil dan Pembahasan Praproses Data.....	140
5.2.1.	Muatan Data Teks.....	140
5.2.2.	Muatan Data Ulasan Pelanggan.....	141
5.2.3.	Hasil dan Pembahasan Segmentasi Kalimat	141
5.2.4.	Hasil dan Pembahasan Tokenisasi.....	142
5.3.	Hasil Pengembangan Prototipe.....	143
5.3.1.	Hasil dan Pembahasan Labeling dengan POS Tagging.....	143
5.3.2.	Hasil dan Pembahasan Validasi Model.....	154
5.3.3.	Hasil dan Pembahasan Pemodelan Chunk Parser pada Aplikasi Plant vs Zombies.....	156
5.3.4.	Hasil dan Pembahasan Pemodelan Chunk Parser pada Aplikasi Instagram.....	163
5.3.5.	Perbandingan Hasil Pada Aplikasi Plant vs Zombies dengan Instagram.....	168
5.4.	Hasil dan Pembahasan Validasi Prototipe.....	171
<b>BAB VII KESIMPULAN DAN SARAN.....</b>		<b>175</b>
6.1.	Kesimpulan.....	175
6.2.	Saran.....	178
<b>DAFTAR PUSTAKA.....</b>		<b>179</b>
<b>BIODATA PENULIS.....</b>		<b>183</b>

A.	LAMPIRAN 1 : SAMPEL HASIL SCRAPING ULASAN DATA PELANGGAN .....	184
	LAMPIRAN 2 : CONSENT FORM .....	185
	LAMPIRAN 3: SYSTEM USABILITY SCALE FORM....	187

## DAFTAR GAMBAR

Gambar 2.1 Relevansi Dasar Teori.....	32
Gambar 2.2 Contoh Ulasan pada Aplikasi Whatsapp (Sumber : <a href="https://play.google.com/store/apps/details?id=com.whatsapp">https://play.google.com/store/apps/details?id=com.whatsapp</a> ) .....	<b>Error! Bookmark not defined.</b>
Gambar 2.3 Konsep Text Chunking [16].....	39
Gambar 3.1 Metodologi Penelitian.....	44
Gambar 4.1 Alur Pengambilan Data oleh Natanael Yabes Wirawan.....	50
Gambar 4.2 Alur Pengambilan Data Penelitian.....	51
Gambar 4.3 Alur Segmentasi Kalimat.....	60
Gambar 4.4 Alur Tokenisasi.....	62
Gambar 4.5 Alur Pemodelan Chunk Parser.....	69
Gambar 4.6 Alur Melakukan Chunking.....	72
Gambar 4.7 Pseudocode untuk Melakukan Chunking.....	73
Gambar 4.8 Alur Sorting Data.....	73
Gambar 5.1 Tampilan Aplikasi Google Play Scrapper.....	80
Gambar 5.2 Contoh Hasil Scraping Data (1).....	87
Gambar 5.3 Contoh Hasil Scraping Data (2).....	87
Gambar 5.4 Data Training dan Testing untuk Uji Validasi .	103
Gambar 5.5 Contoh Data yang Terdapat pada CoNLL-2000 untuk Melakukan Uji Validasi.....	104
Gambar 5.6 Tampilan Awal Uji Validasi Model.....	104
Gambar 5.7 Fitur yang Tersedia Terkait dengan Chunker ...	105
Gambar 5.8 Parameter yang Dimasukkan untuk Melakukan Training pada Model.....	105
Gambar 5.9 Hasil Dari Training Terhadap Pemodelan Chunk Parser.....	105
Gambar 5.10 Hasil dari loglikelihood Pada Setiap Iterasi ...	106
Gambar 5.11 Model yang Dihasilkan dari Proses Training .	107
Gambar 5.12 Parameter yang Dimasukkan Untuk Melakukan Validasi.....	107
Gambar 5.13 Rata-Rata Evaluasi Entitas Sampel dari Model .....	107
Gambar 5.14 Hasil Validasi Model Secara Keseluruhan Oleh OpenNLP.....	108

Gambar 6.1 Konsep Treebank Pada Linguistik Bahasa Inggris [10].....	153
Gambar 6.2 Hasil Perhitungan Usability Testing .....	172

## DAFTAR KODE

Kode 4.1 Kode untuk Melakukan Segmentasi Kalimat .....	61
Kode 4.2 Contoh Kode untuk Tokenisasi .....	65
Kode 4.3 Contoh Kode Tokenisasi pada Pemodelan Chunker Parser.....	67
Kode 4.4 Contoh Kode untuk POS Tagging.....	67
Kode 4.5 Contoh Kode untuk Melakukan Load .....	68
Kode 4.6 Contoh Kode untuk Menginisiasi Class ChunkerME .....	68
Kode 4.7 Contoh Kode untuk Menampilkan Hasil Chunks ...	69
Kode 4.8 Contoh Kode untuk Melakukan Validasi Terhadap Model .....	71
Kode 4.9 Contoh Hasil Evaluasi terhadap Validasi Terhadap Model.....	71
Kode 4.10 Contoh Kode untuk Menulis Data dalam Bentuk CSV.....	75
Kode 5.1 URL untuk mengakses data .....	81
Kode 5.2 Contoh Format Data yang Memuat Data Ulasan Pelanggan .....	82
Kode 5.3 Melakukan Koneksi ke Database .....	82
Kode 5.4 Membaca Data dalam Format JSON.....	83
Kode 5.5 Membaca Data Bentuk JSON .....	84
Kode 5.6 Memasukkan Data .....	86
Kode 5.7 Fungsi getTextFromDB Untuk Pemuatan Data Teks .....	88
Kode 5.8 Pemanggilan Fungsi getTextFromDB pada Main .....	89
Kode 5.9 Kode yang Digunakan untuk Segmentasi Kalimat.	91
Kode 5.10 Fungsi tokenizing pada Main Class.....	93
Kode 5.11 Labeling Token .....	95
Kode 5.12 Menyimpan Hasil Labeling ke Array .....	95
Kode 5.13 Pemanggilan Model Chunk Parser .....	100
Kode 5.14 Memasukkan Hasil Chunk ke dalam Arraylist ...	100
Kode 5.15 Membuat Chunk untuk Ekspresi Kata Kerja (1)	101
Kode 5.16 Membuat Chunk untuk Ekspresi Kata Kerja (2)	102

Kode 5.17 Inisiasi HashMap dan ArrayList untuk Menampung Kalimat.....	109
Kode 5.18 Hashmap Untuk Melakukan Sorting .....	109
Kode 5.19 Menghitung Nomor yang Telah Dibuat HashMap .....	110
Kode 5.20 Melakukan Sorting Terhadap Data .....	110
Kode 5.21 Mengurutkan Kalimat .....	111
Kode 5.22 Menyimpan Hasil Sorting ke File Baru.....	111

## DAFTAR TABEL

Tabel 1.1 Relevansi Tugas Akhir .....	27
Tabel 2.1 Hasil Kajian Penelitian Sebelumnya.....	29
Tabel 2.2 Confusion Matrix .....	40
Tabel 4.1 Atribut Data.....	51
Tabel 4.2 Data Dictionary dari Tabel Application.....	53
Tabel 4.3 Data Dictionary untuk Tabel Review .....	55
Tabel 4.4 Sampel Data Mentah (1).....	57
Tabel 4.5 Sampel Data Mentah (2).....	58
Tabel 4.6 Label Berdasarkan Penn Treebank Set .....	62
Tabel 4.7 Daftar Method dan Parameter yang Digunakan .....	69
Tabel 4.8 Constructor pada Class HashMap.....	75
Tabel 5.1 Sampel Hasil Pemuatan Data.....	89
Tabel 5.2 Hasil Segmentasi Kalimat .....	91
Tabel 5.3 Hasil Tokenisasi pada Beberapa Ulasan Pelanggan .....	93
Tabel 5.4 Sampel Hasil Labeling Kata .....	95
Tabel 5.5 Hasil Percobaan dengan 3 Chunks.....	112
Tabel 5.6 Kesalahan pada Hasil Ekstraksi Ekspresi Kata Kerja yang Dihasilkan.....	115
Tabel 5.7 Hasil Percobaan dengan 4 Chunks.....	115
Tabel 5.8 Kesalahan pada Kalimat yang Dihasilkan .....	119
Tabel 5.9 Sampel Hasil Percobaan dengan 5 Chunks .....	119
Tabel 5.10 Kesalahan pada Kalimat yang Dihasilkan.....	124
Tabel 5.11 Hasil Percobaan dengan 3 Chunks pada Instagram .....	124
Tabel 5.12 Kesalahan pada Kalimat yang Dihasilkan.....	127
Tabel 5.13 Hasil Percobaan dengan 4 Chunks pada Instagram .....	128
Tabel 5.14 Kesalahan pada Kalimat yang Dihasilkan.....	132
Tabel 5.15 Hasil Percobaan dengan 5 Chunks pada Instagram .....	132
Tabel 5.16 Kesalahan pada Hasil Chunking .....	136
Tabel 5.17 Kuisisoner Usability Testing.....	137
Tabel 6.1 Sampel Data Ulasan Pelanggan pada Aplikasi Plant vs Zombies 2 .....	139

Tabel 6.2 Sampel Hasil Segmentasi Kalimat.....	141
Tabel 6.3 Sampel Hasil Tokenisasi .....	142
Tabel 6.4 Pengecekan Kesesuaian Hasil Labeling dengan Model POS Tagging.....	143
Tabel 6.5 Hasil Validasi Model.....	154
Tabel 6.6 Struktur dari Sebuah Chunk .....	156
Tabel 6.7 Penggabungan Label POS Tagging Menjadi Chunk .....	157
Tabel 6.8 Sampe 10 teratas Hasil Chunking dan Sorting dengan 3 chunks .....	159
Tabel 6.9 Sampel 10 Teratas Hasil Chunking dan Sorting dengan 4 chunks.....	160
Tabel 6.10 Hasil Chunking dan Sorting dengan 5 chunks ...	162
Tabel 6.11 Hasil Chunking dan Sorting dengan 3 chunks pada Instagram .....	164
Tabel 6.12 Hasil Chunking dan Sorting dengan 4 chunks pada Instagram .....	165
Tabel 6.13 Hasil Chunking dan Sorting dengan 5 chunks pada Instagram .....	167
Tabel 6.14 Perbandingan Ekspresi Kata Kerja yang Dihasilkan .....	169
Tabel 6.15 Justifikasi System Usability Scale Terhadap Kuisioner.....	172
Tabel 7.1 Aktivitas yang Sering Dibahas Pada Aplikasi .....	177

*Halaman ini sengaja dikosongkan*

# **BAB I**

## **PENDAHULUAN**

Bagian ini akan menjelaskan tahap pengidentifikasian permasalahan yang diangkat pada Tugas Akhir, dimulai dari latar belakang permasalahan, perumusan permasalahan, batasan masalah, tujuan serta manfaat dari pengerjaan Tugas Akhir tersebut.

### **1.1. Latar Belakang**

E-commerce yang beredar pada saat ini jumlahnya cukup banyak, dan masing-masing memiliki proses bisnis dan fitur yang berbeda yang dapat digunakan untuk memudahkan calon pengguna untuk membuat keputusan dalam pembelian produk selain hanya dengan memberikan deskripsi mengenai spesifikasi produk. Salah satunya adalah adanya fitur ulasan pelanggan. Ulasan dari pelanggan tersebut telah menjadi salah satu sumber terpenting bagi pelanggan lainnya dalam membuat keputusan untuk membeli suatu produk maupun jasa, dimana sebelumnya sumber utama bagi pelanggan dalam pembelian suatu produk hanya pada iklan komersial maupun informasi produk yang disediakan oleh vendor [1].

Ulasan pelanggan biasanya memuat banyak informasi mengenai produk tersebut, mulai dari deskripsi produk dan kesesuaian produk dengan spesifikasi yang ditawarkan, pendapat pelanggan mengenai produk tersebut, hingga pengalaman pengguna saat menggunakan produk tersebut. Banyak calon pengguna mencari pendapat pelanggan mengenai produk tersebut yang tercermin dalam ulasan pelanggan. Pihak e-commerce juga telah menyadari hal tersebut, maka dapat dilihat perkembangan fitur ulasan pengguna sekarang, selain menampilkan ulasan pelanggan, dapat dilihat juga sentimen pengguna yang tercermin dalam *rating*. Selain itu, banyak e-commerce yang menyediakan fitur ulasan pelanggan juga dapat dilihat dari perspektif *helpfulness*, positif maupun netral.

Namun, selain sentimen pelanggan, salah satu hal yang paling sering dicari calon pengguna adalah pengalaman pelanggan lain tentang bagaimana memanfaatkan produk tersebut dalam aktivitasnya. Aktivitas tersebut tercermin dalam pengalaman pelanggan tersebut pada saat menggunakan produk tersebut. Saat ini, untuk memahami hal tersebut membutuhkan usaha yang cukup besar karena calon pengguna harus menyimpulkan sendiri pengalaman penggunaan produk tersebut dengan membaca ulasan pelanggan yang jumlahnya cukup banyak. Sehingga, waktu yang dibutuhkan calon pengguna untuk memutuskan pembelian suatu produk lebih lama sehingga calon pengguna menunda merasakan kebermanfaatannya dari produk tersebut. Hal tersebut juga merupakan hal yang baik bagi pemilik e-commerce karena penjualan menjadi tertunda. Sehingga, butuh cara yang lebih efisien untuk konsumen agar dapat mendapatkan informasi tersebut dengan lebih mudah.

Oleh karena itu, penelitian ini akan berfokus pada pengembangan sebuah *prototype* untuk melakukan ekstraksi pengalaman pengguna dalam menggunakan suatu produk berdasarkan ulasan yang telah ditulis pengguna. Dasar teori yang digunakan untuk melakukan ekstraksi ini adalah dengan *Natural Language Processing* yaitu cabang penelitian untuk menganalisis sekumpulan teks yang didasari oleh sekumpulan teori dan teknologi. Model yang dapat dilakukan untuk mengolah informasi tersebut bermacam-macam, dan pada penelitian ini metode yang akan dikembangkan adalah model statistik *Chunking* berdasarkan algoritma *Maximum Entropy* dengan *library* yang terdapat pada OpenNLP. Ulasan akan diekstrak menjadi kata kerja yang mencerminkan pengalaman pengguna saat menggunakan produk. Selanjutnya, informasi yang berhasil diekstrak tersebut akan diurutkan mulai dari yang terbanyak muncul hingga yang paling sedikit muncul.

Penelitian ini diharapkan dapat membantu pengguna untuk mencari produk yang sesuai berdasarkan dengan pengalaman penggunaan produk tersebut. Selain itu, penelitian ini juga dapat membantu pengguna untuk mendapatkan informasi lebih cepat karena tidak perlu membaca banyak ulasan pelanggan untuk

memahami fitur produk secara keseluruhan. Sehingga, pelanggan dapat dengan mudah untuk mengidentifikasi produk yang dibutuhkan berdasarkan pengalaman pengguna lain pada saat menggunakan produk tersebut.

## 1.2. Perumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan pada bagian sebelumnya, maka rumusan masalah yang akan diselesaikan pada Tugas Akhir ini adalah sebagai berikut :

1. Bagaimana struktur penyimpanan data mengenai ulasan produk dari situs e-commerce?
2. Bagaimana ulasan pelanggan akan diolah selanjutnya sehingga menghasilkan ekstraksi ekspresi kata kerja yang sesuai?
3. Aktivitas apa sajakah yang sering dibahas dari pengalaman penggunaan produk tersebut berdasarkan ulasan yang diberikan oleh pelanggan?

## 1.3. Batasan Masalah

Batasan masalah yang dapat diperhatikan dalam penyelesaian Tugas Akhir ini adalah sebagai berikut :

1. Data ulasan produk yang diberikan oleh pengguna merupakan ulasan produk dari satu produk tertentu saja.
2. Sumber data yang diperoleh pada studi kasus ini berasal dari situs e-commerce Google PlayStore.
3. Penelitian ini dikembangkan dengan menggunakan *library* Apache OpenNLP untuk melakukan ekstraksi informasi terhadap data ulasan produk oleh pelanggan.

## 1.4. Tujuan Tugas Akhir

Tujuan dari penelitian Tugas Akhir ini adalah untuk membuat sebuah *prototype* aplikasi CLI menggunakan Java SDK yang dapat menampilkan hasil dari ekstraksi informasi berupa aktivitas-aktivitas yang dilakukan ketika menggunakan sebuah produk.

### **1.5. Manfaat Tugas Akhir**

Tugas Akhir ini diharapkan dapat memberikan manfaat yaitu :

#### **Bagi pengguna e-commerce**

1. Mampu memberikan informasi mengenai aktivitas yang dapat dilakukan ketika menggunakan produk berdasarkan pengalaman penggunaan produk tersebut berdasarkan ulasan pengguna
2. Mampu memberikan sudut pandang baru bagi pengguna yang dapat membantu pengambilan keputusan dalam pembelian suatu produk berdasarkan pengalaman penggunaan produk tersebut. Misalnya, pengguna yang sedang mencari tas punggung untuk keperluan kantor mengetahui bahwa tas tersebut juga dapat digunakan untuk mendaki gunung ketika melihat halaman produk tas tersebut. Aktivitas tersebut didapat dari pengguna yang telah menggunakan produk tersebut sebelumnya sehingga informasi yang diberikan lebih akurat.

#### **Bagi pemilik e-commerce**

1. Mampu memberikan tambahan fitur yang dapat membantu pelanggan e-commerce untuk menampilkan informasi mengenai aktivitas yang dilakukan pada saat penggunaan produk.
2. Memperkaya informasi yang diberikan oleh e-commerce terkait dengan produk atau jasa sehingga dapat memberikan nilai tambah bagi e-commerce tersebut.

### **1.6. Relevansi Tugas Akhir**

Relevansi Tugas Akhir ditujukan untuk menyelaraskan pemahaman dan pengetahuan yang telah diperoleh penulis dengan topik penelitian yang dibahas pada Tugas Akhir ini serta keterkaitan penelitian Tugas Akhir ini dengan *roadmap*

penelitian yang terdapat pada Laboratorium Sistem Enterprise Jurusan Sistem Informasi ITS. Mata kuliah yang berkaitan dengan Tugas Akhir ini telah dipetakan pada Tabel 1.

**Tabel 1.1 Relevansi Tugas Akhir**

<b>No.</b>	<b>Rumusan Masalah</b>	<b>Mata Kuliah</b>
1.	Bagaimana metode pengambilan dan penyimpanan data mengenai ulasan produk dari situs e-commerce?	Manajemen Basis Data, Integrasi Aplikasi Korporasi
2.	Aktivitas apa sajakah yang sering dibahas dari pengalaman penggunaan produk tersebut berdasarkan ulasan yang diberikan oleh pelanggan?	Sistem Cerdas, Sistem Pendukung Keputusan
3.	Bagaimana metode pembuatan <i>prototype</i> berupa aplikasi yang dapat menampilkan aktivitas yang sering dibahas tersebut secara keseluruhan?	Algoritma dan Pemograman, Pemograman Berorientasi Objek, Sistem Cerdas

*Halaman ini sengaja dikosongkan*

## BAB II TINJAUAN PUSTAKA

Bagian ini akan memberikan penjelasan mengenai penelitian maupun studi literatur sebelumnya yang berkaitan dan dijadikan sebagai acuan selama pengerjaan tugas akhir, serta landasan teori yang berkaitan dengan tugas akhir yang dapat membantu pemahaman selama pengerjaan tugas akhir ini.

### 2.1. Penelitian Sebelumnya

Selama pengerjaan tugas akhir ini, terdapat beberapa penelitian yang telah dilakukan sebelumnya yang dapat dijadikan sebagai bahan kajian maupun referensi untuk studi literatur. Penelitian tersebut lalu dikaji untuk dilihat dari gambaran umum, tujuan hasil, dan keterkaitannya dengan penelitian tugas akhir ini. Hasil dari kajian tersebut dapat kita lihat pada tabel berikut ini

**Tabel 2.1 Hasil Kajian Penelitian Sebelumnya**

Judul	Extracting Action and Event Semantics from Web Text
Nama, Tahun	Avirup Sil, Fei Huang, Alexander Yates, 2010
Gambaran umum penelitian	Penelitian ini membuat suatu sistem bernama PREPOST, yang diharapkan mampu menyelesaikan permasalahan dalam mengekstraksi kondisi dan pengaruh dari sebab-akibat adanya sebuah aktivitas tersebut yang terdapat dalam sebuah teks di <i>Web</i> . Aktivitas yang dimaksud pada penelitian ini adalah sebuah fenomena yang terjadi pada waktu dan tempat tertentu. Sistem PREPOST ini telah mampu mendemonstrasikan ekstraksi entitas dari hubungan kondisi sebelum dan sesudah dari sebuah aktivitas dengan tingkat akurasi yang tinggi. [2]

Keterkaitan penelitian	Penelitian ini menggunakan metode <i>machine learning</i> yaitu SVM ( <i>support vector machine</i> ) untuk melakukan ekstraksi terhadap aktivitas dan hubungan sebab-akibat dari aktivitas tersebut berdasarkan <i>library</i> dari FrameNet. Penelitian ini memberikan sudut pandang baru pada proses pengerjaan tugas akhir ini terkait metodologi untuk melakukan ekstraksi aktivitas pada teks <i>Web</i> yang juga dapat diterapkan pada teks ulasan pelanggan.
------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Judul	Extracting Verb Expressions Implying Negative Opinions
Nama, Tahun	Huayi Li, Arjun Mukherjee, Jianfeng Si, Bing Liu
Gambaran umum penelitian	Penelitian ini membahas mengenai ekstraksi kata kerja yang digunakan untuk mendeteksi pendapat negatif pada ulasan pelanggan. Penelitian-penelitian sebelumnya mengenai analisis sentimen yang telah ada membahas mengenai ekstraksi kata benda maupun kata sifat. Sedangkan, kata kerja mencerminkan aktivitas yang sebenarnya juga mengandung sentimen yang digunakan untuk mengekspresikan keluhan dari pelanggan. Penelitian ini menggunakan metode <i>chunking</i> untuk mengekstrak aktivitas yang terdapat pada ulasan pelanggan dengan model <i>machine learning</i> yaitu Markov Networks untuk mendeteksi sentimen yang terdapat pada aktivitas yang telah diekstrak sebelumnya. Hasil dari penelitian ini menunjukkan bahwa model yang digunakan telah mampu menemukan

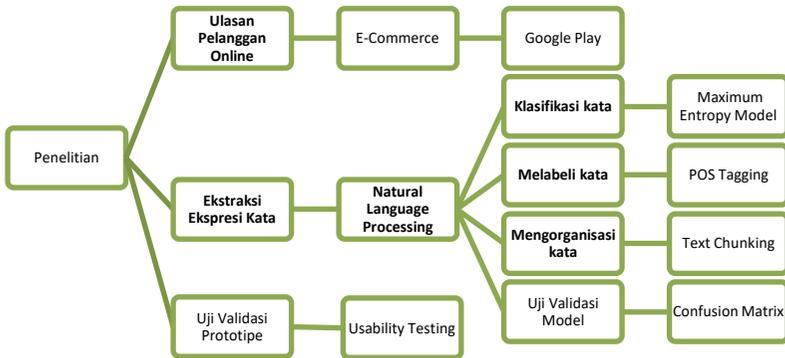
	aktivitas yang mencerminkan sentimen negatif secara efektif. [3]
Keterkaitan penelitian	Dasar dari penelitian ini menggunakan metode <i>chunking</i> dengan metode yang dikembangkan menggunakan <i>machine learning</i> yaitu Markov Networks. Metode <i>chunking</i> tersebut akan digunakan dalam pengembangan metode untuk melakukan ekstraksi aktivitas terhadap ulasan pelanggan, dengan menggunakan <i>library</i> yang dibahas pada penelitian ini yaitu OpenNLP.

Judul	Event Extraction as Dependency Parsing
Nama, Tahun	David McClosky, Mihai Surdeanu, dan Christopher D. Manning
Gambaran umum penelitian	Kebanyakan pendekatan untuk melakukan ekstraksi <i>event</i> hanya menggunakan model-model yang mengekstrak setiap <i>event</i> dan argumen secara independen dan mengabaikan struktur pokok dari <i>event</i> tersebut. Sehingga, seringkali keterkaitan dari satu <i>event</i> dengan <i>event</i> lain terabaikan. Penelitian ini membahas salah satu metode untuk melakukan ekstraksi <i>event</i> , yaitu menggunakan teknik <i>Parsing</i> sebagai dasar dari metode ini dengan algoritma <i>Treebank</i> untuk menentukan hubungan antar kata sehingga relasi antara satu <i>event</i> dengan <i>event</i> lainnya dapat terlihat dengan jelas. Fokus dari penelitian ini adalah untuk mendapatkan struktur dari <i>event</i> dengan beberapa argumen. [4]
Keterkaitan penelitian	Penelitian ini menggunakan <i>Parsing</i> sebagai metode dasar dalam melakukan ekstraksi <i>event</i> dengan menggabungkan

	<p>teknik <i>Parsing</i> tersebut dengan algoritma <i>Treebank</i> yang digunakan pada salah satu <i>library</i> yang telah dikembangkan sebelumnya, yaitu BioNLP. Metode ini diberi nama <i>Depedency Parsing</i>, dan memberikan alternatif lain untuk penelitian ini dalam melakukan ekstraksi eksperimen produk.</p>
--	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## 2.1 Dasar Teori

Bagian ini akan membahas teori dan konsep yang berkaitan dengan penelitian tugas akhir. Relevansi dari dasar teori yang dibahas pada penelitian ini dapat dilihat pada bagan berikut ini :



Gambar 2.1 Relevansi Dasar Teori

### 2.1.1. Ulasan Pelanggan Online

Ulasan pelanggan online merupakan salah satu fitur yang disediakan untuk menampilkan pendapat pelanggan mengenai suatu produk atau jasa yang telah digunakan dan ditulis oleh pelanggan tersebut sendiri. Ulasan pelanggan merupakan salah

satu faktor penting yang digunakan pelanggan dalam pengambilan keputusan pembelian suatu produk atau jasa. Sebanyak 88% pelanggan membaca ulasan pelanggan pada tahun 2014 yang semakin meningkat dari tahun ke tahun [5]. Saat ini, ulasan pelanggan merupakan salah satu karakteristik pada situs jual-beli (*e-commerce*) di dunia maya. Ulasan pelanggan dari situs *e-commerce* tersebut yang akan digunakan sebagai data untuk diteliti lebih lanjut pada penelitian tugas akhir ini, lebih spesifik kepada situs *e-commerce* Google Play, yang akan dijelaskan lebih lanjut pada bagian berikut.

### 2.1.2. E-Commerce

E-commerce adalah suatu istilah yang biasa digunakan untuk mendeskripsikan bisnis dengan jenis apapun, maupun transaksi komersial tertentu yang melibatkan transfer informasi melalui media elektronik apapun, terutama internet [6]. Istilah *e-commerce* pertama kali dikenalkan pada tahun 1960-an melalui sebuah *Electronic Data Interchange* (EDI) pada *Value-added Networks* (VANs). Media ini terus berkembang seiring dengan meningkatnya avaiabilitas dari akses internet di seluruh dunia dan juga penjual online populer pada tahun 1990an dan awal tahun 2000an. Sebagai contoh, Amazon, mulai beroperasi pada bidang penjualan buku pada tahun 1995. Selain itu, Ebay, yang dapat membuat pengguna melakukan transaksi jual-beli ke sesama pelanggan, memperkenalkan sistem lelang tersebut pada tahun 1995 dan sangat marak digunakan pada tahun 1997 [7]. E-commerce terus berkembang dari tahun ke tahun, sehingga sekarang terdapat banyak jenis *e-commerce* yang muncul di pasaran dengan melayani jenis bisnis tertentu yang terbagi atas beberapa kategori. Kategori pertama adalah *Business-to-Business* (B2B), yaitu *e-commerce* yang melayani transaksi jual-beli antar perusahaan. Kategori kedua adalah *Business-to-Consumer* (B2C), meliputi transaksi jual-beli antara bisnis yang telah berkembang dengan *end customers*. Kategori ini dapat dikatakan sebagai kategori yang paling berkembang dari seluruh *e-commerce* yang ada dikarenakan banyaknya *end customers*

yang ada. Salah satu contohnya adalah Google Play, yang merupakan studi kasus dari penelitian tugas akhir ini [8]. Selain kedua kategori tersebut, terdapat beberapa kategori lain, yaitu *Consumer-to-Consumer* (C2C), *Consumer-to-Business* (C2B), *Business-to-Administration* (B2A), dan *Consumer-to-Administration* (C2A) [8].

### **2.1.2.1. Google Play**

Google Play adalah sebuah layanan distribusi digital meliputi pasar media digital yang dioperasikan dan dikembangkan oleh Google. Google Play dibuat sebagai *application store* resmi untuk sistem operasi berbasis Android, yang membuat pengguna dapat menjelajahi dan mengunduh aplikasi yang dikembangkan dengan Android SDK (*Software Development Kit*) dan diluncurkan melalui Google [9].

Pada penelitian ini, Google Play digunakan sebagai studi kasus yang akan diteliti lebih lanjut. Penelitian ini akan mengambil seluruh ulasan pelanggan dari satu produk tertentu yang akan diambil untuk selanjutnya dilakukan analisis lebih lanjut. Karena Google Play merupakan *application store* yang menjual aplikasi virtual bagi pengguna sistem operasi Android, maka ulasan pelanggan banyak membahas aktivitas yang dapat dilakukan saat menggunakan produk. Contoh dari ulasan pelanggan yang terdapat pada Google Play adalah sebagai berikut :

Helpfulness ▾



**Prashant SONAWANE**  
18 February 2017  
★★★★★

I have been using this app for last 3 years...it works gud..but this time...i liked tht u hve been made a option of video calling tool! But the quality of it is not good..! I listen my own voice while talking..as well the voice of other person is also not clear. So i request u too improve tht. Otherwise..its all fine 🙄...I liked the new update made in emoji's☺):



**Geoffrey Ahn** 17 February 2017  
★★★★★

Messenger number one for me. No problems so far. Hopefully we can use it for making phone calls soon! :) About the blue ticks, those who want their privacy, dont use Whatsapp if you feel intimidated. If you dont like the fact others read but do not respond fast enough, I would say you got wrong friends then...



**Rajesh Goyal** 18 February 2017  
★★★★★

Whatsapp is indeed a wonderful app but i need a feature in next update that is:pls add hidden chat feature. Another thing is that there a bug in new update .If we erase an emoji in alphabet keyboard then instead of getting erased new emojis appear. pls fix this soon .otherwise a great app and would give it 5 stars when the requested feature is added. 😊



**panchal samir** 17 February 2017  
★★★★★

app really delivers smooth and high reliability and UI is really simple and perfect. Still there are some bugs in app. sometimes my message doesnt send when i type message and minimise it instantly. it shows my that you may have new messages serval times like 5-6 times in a couple of minutes but when i open app. there are no messages. so i request to fix this..

**Gambar 2.2 Contoh Ulasan pada Aplikasi Whatsapp (Sumber : <https://play.google.com/store/apps/details?id=com.whatsapp>)**

Berdasarkan gambar diatas, dapat dilihat bahwa banyak ulasan pelanggan menjelaskan mengenai aktivitas yang dilakukan pada saat penggunaan produk yang masih harus dibaca satu per satu oleh calon pengguna produk tersebut untuk melihat apakah aplikasi tersebut dapat bekerja dengan baik. Oleh karena itu, penelitian ini menggunakan Google Play sebagai studi kasus dikarenakan relevansi kebutuhan Google Play dengan adanya penelitian ini cukup besar.

### 2.1.3. Ekstraksi Ekspresi Kata Kerja

Pada penelitian ini, ekspresi kata kerja yang dimaksud adalah sebuah urutan frase yang berhubungan secara sintaksis yang meliputi satu atau lebih kata dan identifikasi dari batasan setiap

frase tersebut. Ekspresi kata kerja tersebut akan diekstrak dari kalimat ulasan pelanggan dengan cara mengidentifikasi batasan dari setiap frase ekspresi kata kerja tersebut yang akan dimodelkan sebagai urutan pelabelan. Untuk mendeteksi batasan tersebut, salah satu metode yang dapat dilakukan adalah *text chunking*, yang akan dijelaskan lebih lanjut [3].

Sebuah *chunker* akan mengurai sebuah kalimat menjadi urutan frase yang saling berhubungan satu sama lainnya, dan pada penelitian ini fokus utama dari frase adalah kata kerja yang berkaitan dengan frase lainnya. Sebagai contoh sederhana, sebuah ekspresi kata kerja yang akan diekstrak pada kalimat pelanggan adalah sebagai berikut [3] :

[**NP** *Windows and Linux*] [**VP** *do not respond*] [**PP** *to*] [**NP** *its scroll button*]

Dapat dilihat bahwa informasi utama yang ingin diekstrak adalah kata “*do not respond*” yang merupakan kata kerja. Namun, jika hanya kata kerja tersebut yang diekstrak, maka akan menghasilkan data yang kurang informatif, sehingga, frase di kiri dan kanan kata kerja tersebut akan diekstrak juga, untuk menghasilkan data yang lebih informatif. Metode untuk melakukan ekstraksi ekspresi kata kerja tersebut akan dijelaskan lebih lanjut pada bagian berikut.

### 2.1.3.1. Natural Language Processing

*Natural Language Processing* (NLP) adalah pendekatan terkomputerisasi yang digunakan untuk menganalisis sekumpulan teks yang didasari oleh sekumpulan teori dan teknologi. Saat ini, NLP telah menjadi salah satu area penelitian yang bertujuan untuk mengumpulkan ilmu pengetahuan mengenai bagaimana manusia dapat memahami dan menggunakan bahasa. Sehingga, teknologi sesuai dapat dikembangkan untuk memahami dan memanipulasi bahasa tersebut untuk menyelesaikan berbagai macam permasalahan yang diinginkan [10].

Tujuan utama dari NLP adalah untuk mencapai pemrosesan bahasa seperti manusia pada umumnya dengan menggunakan mesin [11]. Saat ini, NLP terus dikembangkan untuk mencapai

tujuan utamanya tersebut dengan menggunakan berbagai macam pendekatan, model dan algoritma. Pada penelitian ini, pendekatan yang digunakan adalah pendekatan dengan menggunakan algoritma *statistical machine learning*. Algoritma ini akan mengambil masukan berupa *features* yang didapatkan dari data tertentu .

### 2.1.3.2. Maximum Entropy Model

*Maximum Entropy Model* merupakan suatu model probabilistik statistika yang seringkali digunakan untuk memprediksi *Part-of-Speech (POS) Tagging*, yang merupakan metode dasar untuk melakukan *text mining* yang merupakan bagian dari NLP pada suatu *corpus*. Tingkat akurasi dari *Maximum Entropy Model* ini mencapai 96.6% dan telah menjadi salah satu model yang paling populer digunakan dalam penelitian berbasis NLP [12]. *Maximum Entropy Model* ini telah digunakan untuk *language modeling*, *machine translation*, *prepositional phrase attachment* dan *word morphology*.

Model probabilitas tersebut didefinisikan sebagai  $H \times T$ , dimana  $H$  merupakan sekumpulan kata yang mungkin dan konteks *tag*, sedangkan  $T$  adalah sekumpulan *tag* yang diperbolehkan. *Maximum Entropy Model* pada penelitian ini digunakan oleh OpenNLP sebagai dasar pembuatan model untuk melakukan klasifikasi kata yang terdapat pada ulasan pelanggan. Model probabilitas dari *Maximum Entropy* dapat dilihat sebagai berikut [12] :

$$p(h, t) = \pi \mu \prod_{j=1}^k \alpha_j^{f_j(h,t)}$$

Dengan keterangan :

$\Pi$  = konstan yang telah dinormalisasi

$\{\mu, \alpha_1, \dots, \alpha_k\}$  = parameter positif dari model

$\{f_1, \dots, f_k\}$  = *features* dari model

### 2.1.4. Part Of Speech Tagging (POS Tagging)

*Part Of Speech Tagging* adalah suatu metode untuk memberi label sesuai dengan bahasa yang digunakan pada setiap kata di

suatu kalimat didasari dengan bagian dari bahasa yang benar. *Part Of Speech* sering dijadikan sebagai langkah awal untuk melakukan berbagai jenis NLP. Meskipun pada awalnya POS Tagging diterapkan pada Bahasa Inggris, namun telah banyak penelitian menggunakan POS Tagging pada berbagai macam bahasa lainnya. POS Tagging memberikan label pada setiap kata tersebut yang disebut dengan *tagging* [13].

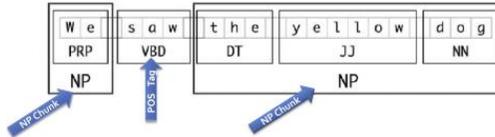
*Tagging* tersebut dilakukan dengan cara memberikan deskripsi berupa label tersebut sebagai deskriptor atau *tag*, setelah dilakukan tokenisasi. Label yang diberikan pada saat melakukan *tagging* tersebut dapat disesuaikan pada konsep bahasa yang digunakan secara natural. Secara umum, bahasa natural dalam setiap bahasa memiliki Kata Benda, Kata Kerja, Kata Sifat dan Kata keterangan. POS Tagging akan memberikan label pada setiap kata tersebut sesuai dengan konsep konstruksi kalimat tersebut [14].

Konsep POS Tagging ini digunakan untuk memberi label pada setiap kata sesuai dengan prinsip struktur kalimat pada penelitian ini. Karena, untuk melakukan proses *chunking* dibutuhkan pemberian label pada setiap kata yang setelah melakukan tokenisasi dan POS Tagging adalah metode yang paling sederhana untuk memberikan label sesuai dengan konsep bahasa yang digunakan, yaitu Bahasa Inggris. POS Tagging telah terdapat di dalam satu *library* yang disediakan oleh Apache OpenNLP.

#### **2.1.4.1. Text Chunking**

*Text Chunking*, atau biasa disebut juga dengan *Shallow Parsing*, adalah metode yang digunakan untuk membagi setiap kalimat menjadi beberapa *segment* yang tidak saling tumpang tindih [15]. *Chunking* merupakan salah satu alternatif metode yang berguna selain melakukan *full parsing*. Pada mulanya, *text chunking* digunakan sebagai tahap yang berguna sebelum melakukan *parsing*. Penelitian banyak menggunakan metode ini untuk mengekstrak kata benda yang berada pada level rendah dengan menggunakan berbagai jenis model statistika tertentu. Namun, metode ini terus dikembangkan sehingga *chunking*

dapat digunakan untuk mengekstrak informasi dengan level yang lebih tinggi. Secara umum, *Chunking* berguna untuk menemukan frase benda, kelompok kata kerja, frase lainnya [16]. Secara umum, konsep *Chunking* dapat dilihat pada gambar berikut :



**Gambar 2.3 Konsep Text Chunking [16]**

*Chunking* melakukan ekstraksi informasi dengan memecah setiap kalimat menjadi beberapa bagian yang biasa disebut *phrasal chunks*. *Phrasal chunks* tersebut lalu dapat disimpan dan diambil bagian yang diinginkan, pada penelitian ini dapat digunakan untuk mengekstrak kata kerja. *Chunking* mengekstrak informasi yang lebih akurat dari *POS Tagging* karena *POS Tagging* hanya memecah kalimat menjadi kata per kata yang diberi label dan mengabaikan keterkaitan kata tersebut dengan kata lainnya. Dengan melakukan *chunking*, kata yang saling terkait masih berada dalam satu *phrasal chunk* yang sama sehingga dapat memberikan informasi yang lebih jelas dibandingkan dengan *POS Tagging* [15].

#### **2.1.4.2. Performance Measure dan Confusion Matrix**

Pada pengembangan suatu model, salah satu faktor penting yang harus diperhatikan adalah ketepatan model tersebut dalam melakukan pendeteksian dan prediksi. Performa dari suatu model dalam melakukan hal tersebut dapat dilakukan dengan metode penilaian secara kuantitas. Secara umum, pada NLP, metode penilaian yang dilakukan adalah *Precision*, *Recall*, dan *F-Measure*. Metode penilaian ini didasari pada konsep *Confusion Matrix*.

*Confusion Matrix* adalah suatu metode analisis yang berguna untuk menangkap hal apapun pada saat melakukan evaluasi dengan menunjukkan hal yang lebih detail sebagai dasar dalam perhitungan performa lainnya. *Confusion Matrix* dilakukan dengan menghitung frekuensi pada setiap kemungkinannya yang dihasilkan pada saat model melakukan prediksi [17]. Untuk melakukan prediksi dengan fitur biner, yaitu dengan dua level, maka dihasilkan 4 luaran saat model membuat prediksi :

- **True Positive (TP)**

Jika nilai pada data testing positif dan diprediksi positif.

- **True Negative (TN)**

Jika nilai pada data testing negatif dan diprediksi negatif.

- **False Positive (FP)**

Jika nilai pada data testing negatif dan diprediksi positif.

- **False Negative (FN)**

Jika nilai pada data testing positif dan diprediksi negatif.

Bentuk tabel penyajian confusion matrix dapat dilihat pada tabel ini.

**Tabel 2.2 Confusion Matrix**

	$p'$ (Predicted)	$n'$ (Predicted)
$p$ (Actual)	True Positive	False Negative
$n$ (Actual)	False Positive	True Negative

**a. Precision**

Precision merupakan nilai ketepatan deteksi atau prediksi dengan benar. Sehingga, perhitungan precision dilakukan

dengan membandingkan jumlah deteksi yang benar dengan jumlah seluruh deteksi.

$$\frac{\textit{True Positive}}{\textit{False Positive} + \textit{True Positive}}$$

### b. Recall

Recall merupakan nilai perbandingan ketepatan prediksi benar dengan jumlah seluruh deteksi yang seharusnya benar.

$$\frac{\textit{True Positive}}{\textit{False Negative} + \textit{True Positive}}$$

### c. F-Measure

F-Measure adalah perhitungan yang digunakan untuk menggabungkan nilai Precision dan Recall. Ia akan menunjukkan nilai keseimbangan dari Precision dan Recall dengan cara:

$$\frac{2 \times (\textit{Precision} \times \textit{Recall})}{\textit{Precision} + \textit{Recall}}$$

#### 2.1.5. Usability Testing

*Usability testing* adalah suatu metode yang dapat digunakan untuk mengevaluasi suatu sistem dari segi efektivitas dan kemudahan sistem tersebut untuk dioperasikan oleh pengguna. *Usability testing* ini terdiri dari lima komponen utama, yaitu *learnability*, *efficiency*, *memorability*, *errors* dan *satisfaction* [18]. Pada penelitian ini *usability testing* digunakan untuk melakukan uji validasi terhadap prototipe yang telah dikembangkan.

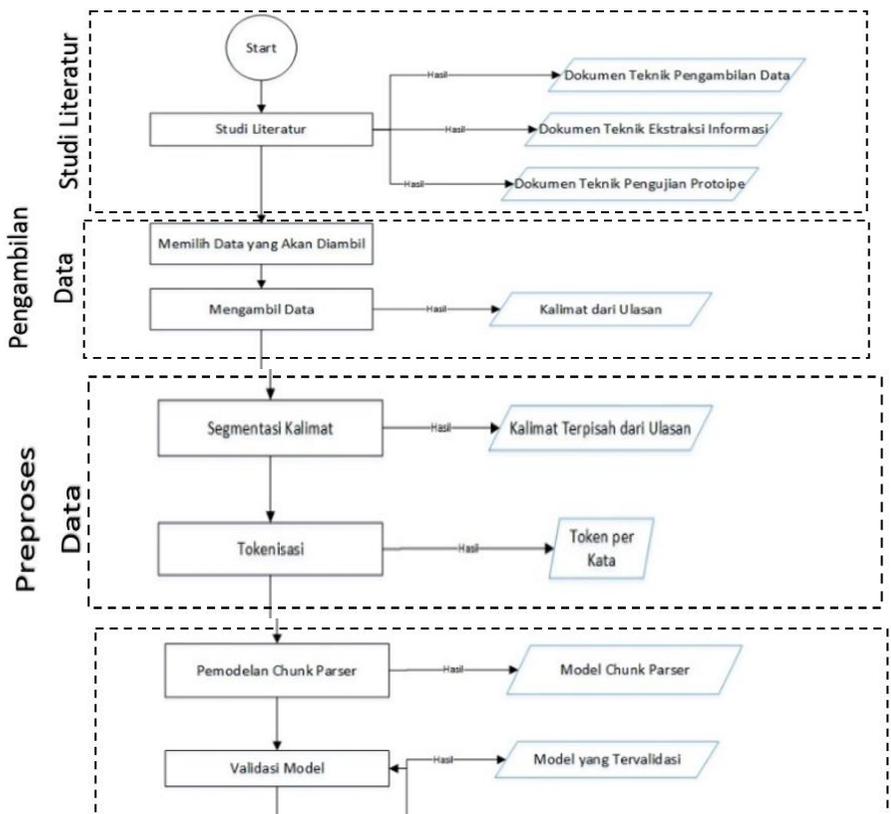
#### 2.1.6. Apache OpenNLP

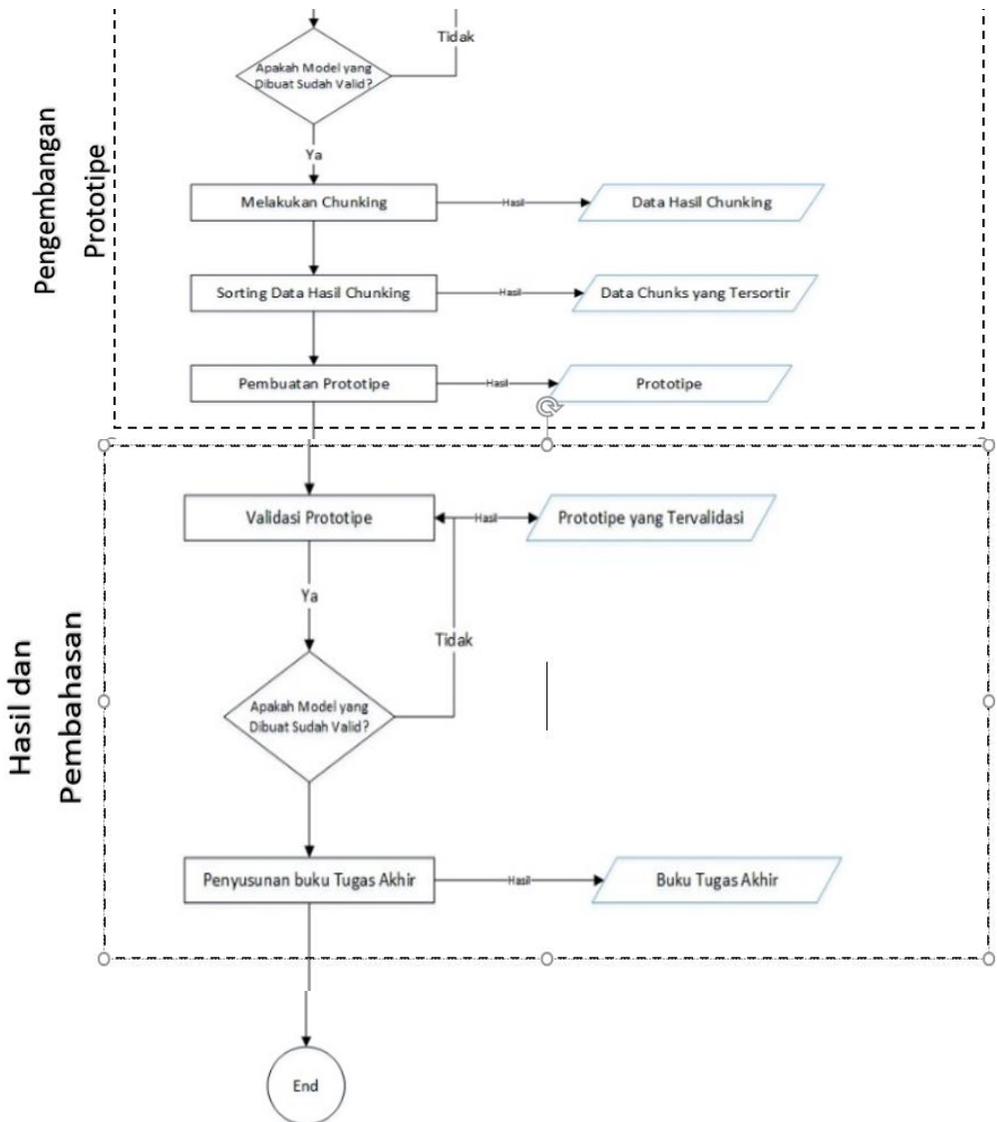
Apache OpenNLP adalah sebuah *library* yang berbasis *machine learning* yang dapat digunakan untuk melakukan pemrosesan

teks dengan basis *natural language processing*. Apache OpenNLP mendukung beberapa teknik untuk melakukan NLP seperti tokenisasi, segmentasi kalimat, *Part-of-Speech Tagging*, *Named Entity Extraction*, *Chunking*, *Parsing* dan *Coreference Resolution*. Teknik-teknik ini biasanya dibutuhkan untuk membangun sistem pemrosesan teks yang lebih mumpuni. OpenNLP menggunakan *Maximum Entropy* dan *Perceptron* sebagai basis model dari *machine learning* yang digunakan pada *library* tersebut.

### BAB III METODOLOGI

Bagian ini akan menjelaskan metodologi dan alur pengerjaan tugas akhir. Gambaran metodologi ini akan berguna sebagai pedoman selama pengerjaan tugas akhir yang dapat dilihat pada gambar berikut :





**Gambar 3.1** Metodologi Penelitian

### 3.1. Tahapan

Penjelasan setiap tahapan dari metodologi adalah sebagai berikut :

#### 3.1.1. Studi Literatur

Tahap awal dari pengerjaan Tugas Akhir ini adalah dengan mengkaji studi literatur. Studi literatur berguna untuk menentukan topik dan metode yang sesuai. Studi literatur dilakukan dengan cara menganalisa latar belakang permasalahan yang ditemukan saat menggunakan e-commerce. Selanjutnya, dilakukan penentuan rumusan masalah yang diangkat. Pada tahap ini dilakukan juga penentuan batasan masalah, tujuan, serta manfaat penentuan topik penelitian. Untuk mendukung latar belakang dari permasalahan serta rumusan permasalahan yang sesuai dengan topik yang diangkat, peninjauan pustaka dilakukan dimulai dari studi penelitian sebelumnya yang terkait dengan ekstraksi informasi dengan penggunaan *Text Chunking* dan dasar teori mengenai *Natural Language Processing*, *Text Chunking*, dan juga algoritma yang digunakan seperti *Maximum Entropy Model*, *Performance Measure* dan *Confusion Matrix*. Selain itu, sebagai referensi pustaka, dilakukan juga studi penelitian terkait studi kasus dari penelitian ini, yaitu e-commerce dan Amazon. Literatur yang digunakan sebagai studi penelitian ini berasal dari buku, jurnal ilmiah nasional maupun internasional, makalah penelitian, e-book dan bahan-bahan lain yang dapat dijadikan acuan resmi selama pengerjaan tugas akhir ini.

#### 3.1.1 Pengumpulan Data

Setelah melakukan studi literatur, selanjutnya akan dilakukan pengumpulan data. Tahapan ini akan mengambil seluruh data yang terdapat pada Google PlayStore dengan menggunakan metode *scraping*. Data yang diambil utamanya mengenai ulasan produk tersebut. Hasil data dari *crawling* ini selanjutnya akan dimasukkan ke dalam sebuah basis data untuk dilakukan *text mining* terhadap ulasan tersebut. Proses ini telah dilakukan

sebelumnya oleh pada Tugas Akhir yang dilakukan oleh Natanael Yabes Wirawan [19]. Sehingga, pada tahapan ini dilakukan pemilihan data yang telah disimpan pada basis data tersebut, mengambil data mengenai ulasan pelanggan yang dibutuhkan untuk melakukan penelitian lebih lanjut. Selanjutnya, ulasan pelanggan tersebut akan disimpan ke dalam basis data baru untuk memudahkan analisis yang dilakukan pada tahapan selanjutnya.

### **3.1.2. Preproses Data**

Pada tahapan ini, ada beberapa langkah yang harus dilakukan. Tahapan awal adalah dengan melakukan segmentasi kalimat. Segmentasi kalimat dilakukan dengan memisahkan setiap kalimat yang terdapat pada setiap ulasan pelanggan. Hal ini dilakukan untuk mempermudah proses pada tahapan-tahapan selanjutnya. Setelah dilakukan pemisahan ulasan pelanggan menjadi kalimat-kalimat terpisah, maka tahapan selanjutnya yang dapat dilakukan adalah tokenisasi. Tokenisasi adalah sebuah metode untuk mengubah kalimat yang telah dilakukan segmentasi tersebut menjadi potongan-potongan kata pada setiap baris. Setiap kata pada baris tersebut akan menjadi satu token dan akan disimpan dalam format file .tok. Batasan yang memisahkan setiap token adalah *whitespace* dan tanda baca tertentu sekaligus secara otomatis membuang tanda baca tersebut [20].

### **3.1.3. Pengembangan Prototipe**

Setelah dilakukan tokenisasi, maka tahapan yang dilakukan selanjutnya adalah dengan melakukan perancangan terhadap prototipe. Langkah pertama yang dilakukan adalah dengan membuat sebuah model chunk parser berdasarkan *library* yang telah disediakan oleh Apache OpenNLP. Model chunk parser akan melakukan labeling terhadap seluruh token yang ada terlebih dahulu untuk memudahkan proses *chunking* pada ulasan pelanggan tersebut dengan menggunakan konsep *POS Tagging*.

Setelah dilakukan pemodelan untuk melakukan *chunking*, maka langkah selanjutnya adalah melakukan uji validasi terhadap model yang telah dibuat tersebut dengan metode *performance measure* dan *confusion matrix*, yaitu dengan menghitung nilai *precision*, *recall* dan *f-measure* dari model tersebut. Jika uji validasi dari model menunjukkan nilai yang tinggi, maka model telah siap digunakan untuk tahap selanjutnya. Jika tidak, maka akan dilakukan perbaikan terhadap model yang telah dibuat tersebut sampai menunjukkan tingkat validasi yang diinginkan. Setelah labeling dan pembuatan model chunk parser dilakukan, maka selanjutnya proses *chunking* terhadap token-token ulasan pelanggan tersebut untuk menemukan aktivitas-aktivitas yang terdapat pada ulasan pelanggan. Proses *chunking* ini akan menghasilkan data dalam bentuk *chunks* yang berisi kumpulan ekspresi kata kerja yang telah diekstrak dari ulasan pelanggan. Setelah hasil didapatkan data hasil proses *chunking*, maka selanjutnya akan dilakukan *sorting*, yaitu pengurutan data agar data-data tersebut lebih mudah dipahami oleh pengguna. Data *chunks* tersebut akan diurutkan berdasarkan frekuensi munculnya ekspresi kata kerja, mulai dari yang terbanyak muncul hingga yang terkecil, sehingga pengguna nantinya dapat memahami dengan mudah aktivitas yang paling sering dilakukan dengan produk atau jasa tersebut.

Setelah hasil dari data *chunks* yang diurutkan tersebut telah dilakukan, maka tahapan selanjutnya yang dilakukan adalah dengan membuat prototipe yang dapat menampilkan hasil dari ekstraksi teks tersebut. Data akan ditampilkan menggunakan CLI yang menunjukkan hasil dari ekstraksi ekspresi kata kerja yang dapat mencerminkan aktivitas penggunaan produk secara terurut, dari yang terbanyak muncul hingga paling sedikit.

#### **3.1.4. Hasil dan Pembahasan**

Setelah prototipe dibuat, maka tahapan selanjutnya adalah melakukan uji validasi terhadap prototipe yang digunakan dengan metode *usability testing*, dengan melakukan survei terhadap calon pengguna dari fitur ini. Jika prototipe telah dapat berjalan dengan baik dan survei yang dilakukan mendapatkan

hasil yang telah diinginkan, maka dapat diambil kesimpulan terhadap penelitian ini.

Tahapan selanjutnya adalah penulisan Buku Tugas Akhir sebagai dokumentasi dari penelitian ini. penulisan buku dilakukan bersamaan dengan seluruh tahapan penelitian yang dilakukan. Luaran yang dihasilkan adalah Buku Tugas Akhir yang diharapkan dapat bermanfaat bagi berbagai pihak dan menjadi referensi untuk penelitian yang berkaitan dengan bidang yang sama selanjutnya.

## **BAB IV PERANCANGAN**

Bagian ini akan memberikan gambaran mengenai perancangan terhadap sistem yang akan dibangun sebelum melakukan tahapan implementasi pada penelitian tugas akhir. Perancangan pada bab ini meliputi subyek dan obyek penelitian, pemilihan subyek dan obyek penelitian, dan bagaimana tahapan penelitian akan dilakukan agar tetap sesuai dengan alur pengerjaan ketika akan melakukan tahapan implementasi.

### **4.1. Perancangan Pembuatan Prototipe**

Bagian ini akan menjelaskan perancangan pembuatan prototipe sebagai acuan pada saat akan melakukan implementasi pada penelitian ini sehingga terjadinya kesalahan pada proses implementasi dapat diminimalisir.

#### **4.1.1. Perancangan Pengambilan Data**

Pada bagian ini akan dijelaskan mengenai perancangan pengambilan data yang bertujuan sebagai panduan ketika akan melakukan implementasi pada penelitian ini.

##### **4.1.1.1. Memilih Data yang Akan Diambil**

Data adalah salah satu entitas terpenting untuk melakukan penelitian ekstraksi ekspresi kata kerja ini. Data yang digunakan pada penelitian ini merupakan dataset berupa teks dari salah satu situs E-Commerce, yaitu Google Play. Data yang diambil dari Google Play dan digunakan pada penelitian ini adalah data mengenai ulasan pengguna pada produk-produk yang dijual pada situs tersebut. Alur pengambilan data yang dilakukan secara umum dapat dilihat pada gambar berikut :



**Gambar 4.1 Alur Pengambilan Data oleh Natanael Yabes Wirawan**

Agar dapat menggunakan data tersebut, maka diperlukan pengambilan data. Proses pengambilan data tersebut telah dilakukan oleh Natanael Yabes Wirawan (Sistem Informasi ITS 13) [19]. Pengambilan data dilakukan secara *real time* dengan data yang diambil merupakan data yang berhubungan dengan produk yang terdapat pada Google Play seperti nama produk, jenis produk, tanggal produk dirilis, dan data lainnya yang berkaitan dengan produk-produk yang terdapat pada Google Play. Teknik yang digunakan adalah *scraping* menggunakan bahasa pemrograman PHP dan JSON yang disimpan ke dalam database MySQL, yang lalu di *upload* pada sebuah website agar dapat digunakan untuk keperluan penelitian Laboratorium Sistem Enterprise.

Data yang dibutuhkan oleh penelitian ini hanyalah data mengenai ulasan pelanggan, sehingga data yang akan diambil hanyalah data yang berkaitan dengan pelanggan dan ulasan pelanggan yang selanjutnya akan disimpan ke dalam database MySQL untuk digunakan pada penelitian ini. Data yang berkaitan dengan ulasan pelanggan terhadap aplikasi yang terdapat pada aplikasi dalam Google Play terdapat pada tabel Reviews pada database dengan atribut yang dapat dilihat pada tabel berikut :

Nama Atribut	Tipe Data	Keterangan
<b>Id (PK)</b>	varchar	ID ulasan pelanggan Google Play
<b>app_id</b>	int	ID dari aplikasi Google Play
<b>userName</b>	varchar	<i>Username</i> dari pelanggan
<b>userImage</b>	varchar	URL dari foto profil pelanggan
<b>date</b>	varchar	Tanggal pelanggan memberikan ulasan
<b>url</b>	varchar	URL dari ulasan pelanggan
<b>score</b>	int	Nilai dari pelanggan untuk aplikasi
<b>title</b>	varchar	Judul dari ulasan pelanggan
<b>text</b>	text	Isi dari ulasan pelanggan

Tabel 4.1 Atribut Data

Dari proses pengambilan data tersebut, terdapat sebanyak 121 halaman yang berhasil dilakukan *scraping* dan terdapat sebanyak 4840 ulasan pelanggan yang dapat dikumpulkan. Tabel berikut menunjukkan sampe data yang tersimpan dalam database MySQL. Proses pengambilan data untuk penelitian ini dapat dilihat pada gambar berikut :



Gambar 4.2 Alur Pengambilan Data Penelitian

#### 4.1.1.2. Data Dictionary

*Data Dictionary* merupakan tabel yang berisi struktur skema dan definisi variabel dari kolom yang akan diisi oleh data dengan berbagai tipe tertentu. Menurut Oracle, *Data dictionary*

juga dapat disebut sebagai sebuah struktur data dengan tampilan seperti basis data pada umumnya, namun berisi metadata *repository* bagi sebuah basis data, yang hanya dapat dibaca (*read-only*) dan tidak dapat diubah secara langsung [21]. Biasanya, di dalam sebuah *data dictionary* terdapat beberapa hal berikut :

- Definisi dari semua objek skema yang terdapat di dalam sebuah database (tabel, view, indeks, cluster, sinonim, urutan, prosedur, dll)
- Berapa banyak alokasi data yang digunakan oleh objek skema tersebut
- Nilai default untuk setiap kolom
- Nama dari pengguna
- Keuntungan dan peran dari setiap pengguna yang telah diperbolehkan
- Informasi yang telah diaudit, seperti siapa yang dapat mengakses atau memperbaharui objek skema tersebut
- Dan informasi umum mengenai basis data lainnya

*Data dictionary* biasanya mempunyai tiga pengguna utama, yaitu :

- Administrator yang dapat mengakses *data dictionary* tersebut untuk mengetahui informasi mengenai pengguna, objek skema dan struktur penyimpanan dari basis data
- Administrator yang dapat memodifikasi *data dictionary* setiap kali ada DDL (*Data Definition Language*) yang ditambahkan
- Pengguna lainnya yang dapat menggunakan *data dictionary* sebagai referensi yang dapat dibaca untuk mengetahui informasi apapun mengenai basis data

Pembuatan *data dictionary* berguna untuk mengetahui jenis tipe apa yang akan digunakan pada setiap kolom data pada basis data yang digunakan dan merupakan penjelasan lebih rinci dari ERD (*Entity Relation Diagram*) dari sebuah basis data. Pada

penelitian kali ini, akan ada dua tabel yang akan digunakan yaitu tabel *Application* sebagai tabel *primary* dari basis data dan tabel *review* yang merupakan tabel yang berisi data ulasan pelanggan yang beberapa kolom pada tabelnya berhubungan dengan tabel *Application*. *Data Dictionary* pada tabel *Application* dapat dilihat pada tabel berikut :

**Tabel 4.2 Data Dictionary dari Tabel *Application***

Table	Column	Xty pe	Data Type	Le n	Null	Description
<b>Applic ations</b>	id	PK, FK	int	11	N	Primary key of application records
<b>Applic ations</b>	title		varchar	100	N	Name of the application
<b>Applic ations</b>	summary		varchar	100	N	outline/abstract/tag of the application
<b>Applic ations</b>	icon		varchar	500	N	Image or symbol for the application
<b>Applic ations</b>	price		tinyint	1	N	no charge for the application
<b>Applic ations</b>	minInstalls		int	11	N	minimal installs of the application
<b>Applic ations</b>	maxInstalls		int	11	N	maximal installs of the application
<b>Applic ations</b>	score		decimal	10, 2	N	score for the application
<b>Applic ations</b>	reviews		int	11	N	amount reviews for the application

<b>Applications</b>	developer		varchar	42	N	Name of the application developer
<b>Applications</b>	developerEmail		varchar	45	N	email of the developer
<b>Applications</b>	developerWebsite		varchar	45	N	website of the developer
<b>Applications</b>	update		varchar	45	N	Last date for the application updated
<b>Applications</b>	genre		varchar	45	N	genre for the application
<b>Applications</b>	genreID		varchar	45	N	
<b>Applications</b>	description		text		N	description of the application
<b>Applications</b>	descriptionHTML		text		N	description of the application HTML version
<b>Applications</b>	offersIAP		tinyint	1	N	
<b>Applications</b>	adSupported		tinyint	1	N	information about the app have an ads
<b>Applications</b>	androidVersionText		varchar	45	N	
<b>Applications</b>	androidVersion		varchar	10	N	version of the application
<b>Applications</b>	contentRating		varchar	10	N	people category for the app content rating
<b>Applications</b>	video		varchar	100	N	

<b>Applications</b>	recentChanges		varchar	45	N	Last change description of the application
<b>Applications</b>	preregister		tinyint	1	N	
<b>Applications</b>	url		varchar	100	N	Link for get the application in googleplay
<b>Applications</b>	appID		varchar	100	N	

Sedangkan *data dictionary* untuk tabel *review* dapat dilihat pada tabel berikut :

**Tabel 4.3 Data Dictionary untuk Tabel Review**

Table	Column	Data Type	Len	Null	Keterangan
<b>Reviews</b>	Id (PK)	varchar	500	N	ID ulasan pelanggan Google Play
<b>Reviews</b>	app_id	int	100	N	ID dari aplikasi Google Play
<b>Reviews</b>	userName	varchar	45	N	<i>Username</i> dari pelanggan
<b>Reviews</b>	userImage	varchar	500	N	URL dari foto profil pelanggan
<b>Reviews</b>	date	varchar	50	N	Tanggal pelanggan memberikan ulasan
<b>Reviews</b>	url	varchar	200	N	URL dari ulasan pelanggan

<b>Reviews</b>	score	int	11	N	Nilai dari pelanggan untuk aplikasi
----------------	-------	-----	----	---	-------------------------------------

#### 4.1.1.3. Mengambil Data

Tahap selanjutnya yang dilakukan setelah memilih data yang akan digunakan adalah mengambil data. Untuk mengambil data yang sesuai, dilakukan pemilihan atribut yang mengandung data paling sesuai untuk diproses selanjutnya. Pemilihan atribut dilakukan untuk memilih dan menyeleksi atribut data mana saja yang akan digunakan selama penelitian ini berlangsung. Atribut yang dipilih adalah atribut yang mengandung data paling sesuai untuk penelitian ini.

Pada proses pengambilan data ini juga dilakukan *regular expression* yang bertujuan untuk menghilangkan seluruh tanda baca seperti titik, koma, tanda kurung, tanda tanya, tanda seru dan berbagai jenis tanda baca lainnya. Berbagai jenis tanda baca ini dihilangkan agar proses *chunking* dapat berjalan lebih lancar dan menghasilkan *chunks* yang lebih akurat karena dengan hilangnya tanda baca tersebut, *chunks* hanya akan melakukan *chunks* terhadap kata-kata berlabel, tidak terganggu dengan tanda baca tersebut.

Berdasarkan dataset Google Play yang diperoleh dari proses pengambilan data, atribut yang akan digunakan untuk pengembangan penelitian ini adalah atribut *text*. Atribut *text* dipilih karena atribut ini menunjukkan isi teks berupa ulasan dari pengguna yang dibutuhkan untuk penelitian ini. sehingga, atribut data inilah yang akan diolah dan dilakukan analisis lebih lanjut. Sebagai contoh, berikut adalah sampel data mentah yang berhasil diambil dari aplikasi Facebook yang dapat dilihat pada tabel berikut :

Tabel 4.4 Sampel Data Mentah (1)

id	app_id	userName	userImage	date	
<b>gp:AO qpTO E-2- 2tMPu LNoV 9MkE Kg9C H8ok0 M8gm</b>	com.fa cebook .katana	Claire Ward	https://lh5.g oogleuserc ontent.com/ - GYceyTIC 2r8	April 2017	19,
<b>gp:A OqpT OE- 5UaF _FgW</b>	com.fa cebook .katana	dan Green	https://lh5.g oogleuserc ontent.com/ -7swzG6- RWds/	April 2017	17,
<b>gp:A OqpT OE- 8kMj u4Kd mvM EWJg 8aa5s</b>	com.fa cebook .katana		https://lh5.g oogleuserc ontent.com/ -- uALvvp9O Lc	April 2017	19,
<b>gp:A OqpT OE- 8kMj u4Kd mvM EWJg 8aa5s</b>	com.fa cebook .katana		https://lh5.g oogleuserc ontent.com/ -- uALvvp9O Lc	April 2017	19,
<b>gp:A OqpT OE- Qxfm he4Z</b>	com.fa cebook .katana	Kelly Liesse	https://lh3.g oogleuserc ontent.com/ -	April 2017	17,

DM2q YAG	JvkFpjFcN kE
-------------	-----------------

Tabel 4.5 Sampel Data Mentah (2)

url	score	title	text
<a href="https://play.google.com/store/apps/details?">https:// play.g oogle.c om/sto re/app s/detail s?</a>	1		:( really poor, why if I open a post am I taken back to a new feed!!!! Can't get thru my posts, missing loads, please bring back a basic feed that is time related!!!!!!!!!! Stop messing fb up!!!!!! Seriously going you have to stop using if this rubbish continues, messing with my head, surely nobody needs this @#\$\$*
<a href="https://play.google.com/store/apps/details?id=co">https:// /play. google .com/s tore/a pps/de tails?i d=co</a>	3		I like all FB features apart from the chat option going from the side menu, and on a few occasions FB freezes, and why is there always a Facebook update nealy every week
<a href="https://play.google.com/store/apps/details?">https:// /play. google .com/s tore/a pps/de tails?</a>	1		Took to long yo install after slesfy waiting to update. Really does the left hand know what the rightt hand is doing
<a href="https://play.google.com/store/apps/details?">https:// /play. google .com/s tore/a pps/de tails?</a>	1		Took to long yo install after slesfy waiting to

google .com/s tore/a pps/de tails?		update. Really does the left hand know what the right hand is doing
https:// /play. google .com/s tore/a pps/de tails?	3	Not good that you can't see the chat side button and when your favourites were online
22726 87298 78_10 15388 43128 39879		8/1/2015 21:16

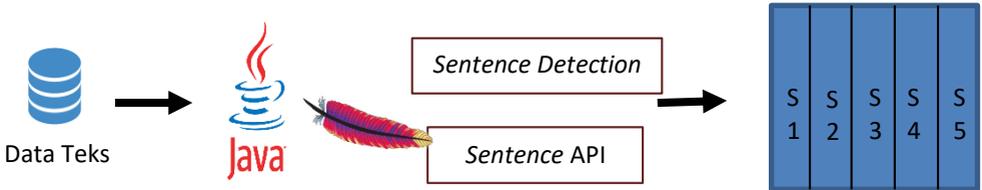
#### 4.1.2. Perancangan Preproses Data

Pada bagian ini akan dijelaskan mengenai perancangan preproses data yang bertujuan sebagai panduan ketika akan melakukan implementasi pada penelitian ini.

##### 4.1.2.1. Segmentasi Kalimat

Tahap awal untuk melakukan preproses data adalah segmentasi kalimat. Segmentasi kalimat bertujuan untuk memecah beberapa kalimat yang terdapat dalam satu paragraf menjadi kalimat per kalimat. Hal ini dilakukan dengan tujuan untuk mempermudah proses tokenisasi yang akan dilakukan selanjutnya. Pada penelitian yang terkait dengan *Natural Language Processing*, proses ini dikenal dengan *Sentence Boundary Disambiguation* (SBD) atau secara sederhana dapat

disebut dengan pemecahan kalimat. Proses segmentasi kalimat ini lebih jelasnya dapat dilihat pada gambar berikut :



**Gambar 4.3 Alur Segmentasi Kalimat**

Untuk melakukan segmentasi kalimat, penelitian ini menggunakan salah satu model yang terdapat dalam sebuah *library* yang telah disediakan oleh Apache OpenNLP. Model yang digunakan tersebut disebut dengan *Sentence Detector* dengan nama file *en-sent.bin*. Model ini telah dilatih untuk mendeteksi kalimat yang terdapat pada teks mentah, sehingga sesuai untuk diterapkan pada data teks mentah yang akan diolah pada penelitian ini.

Sebagai uji coba sebelum melakukan implementasi, maka data mentah ulasan pelanggan yang telah diambil sebelumnya dapat disimpan ke dalam sebuah file dengan format *.txt*. File tersebut disimpan ke dalam sebuah folder *package* dimana segmentasi kalimat akan dilakukan. Lalu, dilakukan segmentasi kalimat akan dilakukan dengan *pre-trained* model yang telah disediakan oleh OpenNLP untuk tokenisasi, yaitu *en-token.bin* dengan melakukan *load* terhadap model tersebut ke program.

*Method* utama yang digunakan untuk melakukan segmentasi kalimat ini adalah *sentDetect()* pada *class SentenceDetectorME* yang terdapat pada model tersebut yang digunakan untuk mendeteksi kalimat-kalimat yang terdapat pada teks mentah. *Method* ini dapat menerima variabel *String* sebagai parameternya. Untuk lebih jelasnya, kode yang dapat dibuat ke dalam sebuah program untuk melakukan segmentasi kalimat ini dapat dilihat sebagai berikut :

```

import java.io.FileInputStream;
import java.io.InputStream;
import opennlp.tools.sentdetect.SentenceDetectorME;
import opennlp.tools.sentdetect.SentenceModel;
import opennlp.tools.util.Span;

public class SentencePosDetection {
    public static void main(String args[]) throws
Exception {
        String paragraph = "Selamat Datang di
percoobaan Segmentasi Kalimat. "
            + "Penelitian ini berkaitan dengan Lab SE";

        //Melakukan Load terhadap model
        InputStream inputStream = new
FileInputStream("C:/OpenNLP_models/en-sent.bin");
        SentenceModel model = new
SentenceModel(inputStream);

        //Menginisiasi Class Sentence Detector
        SentenceDetectorME detector = new
SentenceDetectorME(model);

        //Mendeteksi posisi kalimat pada paragraf
        Span spans[] =
detector.sentPosDetect(paragraph);

        //Mencetak kalimat
        for (Span span : spans)
            System.out.println(span);
    }
}

```

#### **Kode 4.1 Kode untuk Melakukan Segmentasi Kalimat**

##### **4.1.2.2. Tokenisasi**

Tahap selanjutnya adalah melakukan tokenisasi, yaitu proses pemecahan kalimat per kalimat menjadi kata per kata. Proses ini dilakukan dengan tujuan untuk mempermudah proses

*labeling* kata pada tahap selanjutnya. Untuk lebih jelasnya, proses tokenisasi tersebut dapat dilihat pada gambar berikut :



**Gambar 4.4 Alur Tokenisasi**

Tahap yang perlu dilakukan untuk proses tokenisasi ini adalah dengan memisahkan kalimat menjadi kata per kata. Setelah memisahkan kalimat menjadi kata, maka *tools* untuk melakukan tokenisasi yang telah disediakan oleh Apache OpenNLP tersebut secara langsung akan melakukan *labeling* pada setiap kata. *Labeling* adalah proses memberikan label pada setiap kata yang ada sesuai dengan jenis kata tersebut. Apache OpenNLP menggunakan *Penn Treebank Set* sebagai panduan untuk melakukan label pada setiap jenis kata yang ada pada model *Tokenizing*-nya. Label pada setiap berdasarkan *Penn Treebank Set* dapat dilihat pada tabel berikut :

**Tabel 4.6 Label Berdasarkan Penn Treebank Set**

Nomor	Label	Deskripsi
1	CC	Coordinating Conjunction
2	CD	Cardinal Number
3	DT	Determiner
4	EX	Existential <i>there</i>
5	FW	Foreign Word
6	IN	Preposition or subordinating conjunction
7	JJ	Adjective
8	JJR	Adjective, superlative
9	JJS	Adjective, superlative

10	LS	List item marker
11	MD	Modal
12	NN	Noun, singular or mass
13	NNS	Noun, Plural
14	NNP	Proper Noun, singular
15	NNPS	Proper noun, plural
16	PDT	Predeterminer
17	POD	Possessive Ending
18	PRP	Personal Pronoun
19	PRP\$	Possessive Pronoun
20	RB	Adverb
21	RBR	Adverb, comparative
22	RBS	Adverb, superlative
23	RP	Particle
24	SYM	Symbol
25	TO	<i>To</i>
26	UH	Interjection
27	VB	Verb, base form
28	VBD	Verb, past tense
29	VBG	Ver, gerund or present participle
30	VCN	Verb, past participle
31	VBP	Verb, non-3rd person singular present
32	VBZ	Verb,, 3rd person singular present
33	WDT	Wh-determiner
34	WP	Wh-pronoun
35	WP\$	Possessive we-pronoun
36	WRB	Wh- adverb

Sebagai contoh, berikut adalah sebuah kalimat sebelum dilakukan tokenisasi :

Pierre Vinken , 61 years old , will join the board as a nonexecutive director Nov. 29 .  
Mr. Vinken is chairman of Elsevier N.V. , the Dutch publishing group .

Hasil tokenisasi dari kalimat tersebut adalah :

```
Pierre_NNP Vinken_NNP , , 61_CD years_NNS
old_JJ , , will_MD join_VB the_DT board_NN
as_IN a_DT nonexecutive_JJ director_NN
Nov._NNP 29_CD . .
Mr._NNP Vinken_NNP is_VBZ chairman_NN of_IN
Elsevier_NNP N.V._NNP , , the_DT Dutch_NNP
publishing_VBG group_NN
```

Untuk melakukan tokenisasi pada penelitian ini, model yang digunakan adalah *pre-trained* model yang telah disediakan oleh Apache OpenNLP. *Pre-trained* model adalah model yang telah dilakukan *training* sebelumnya sehingga telah siap untuk digunakan. Karena bahasa yang digunakan pada ulasan pelanggan pada Google Play merupakan Bahasa Inggris, yaitu bahasa yang sama digunakan oleh OpenNLP, maka tidak dibutuhkan *training* model baru. Sehingga, *pre-trained* model dari OpenNLP ini dapat digunakan secara langsung.

Sebagai uji coba untuk melakukan tokenisasi, maka data hasil melakukan segmentasi kalimat dapat disimpan terlebih dahulu ke dalam sebuah file dalam bentuk .txt. File tersebut lalu disimpan ke folder yang sesuai untuk melakukan tokenisasi. Selanjutnya, dapat dibuat sebuah program yang berfungsi untuk membaca teks tersebut. Lalu, dilakukan tokenisasi menggunakan *pre-trained* model yang telah disediakan oleh OpenNLP untuk tokenisasi, yaitu en-token.bin dengan melakukan *load* terhadap model tersebut ke program. Hasil dari tokenisasi tersebut dapat ditampilkan dan disimpan dalam bentuk Arraylist. Untuk lebih jelasnya, proses tokenisasi tersebut dapat dilihat sebagai berikut :

```

File coba = new File("C:\\Users\\Almira
F\\Documents\\NetBeansProjects\\NyobaNLP\\coba.txt");
BufferedReader read = new BufferedReader(new
FileReader(file));
InputStream is = new FileInputStream("C:\\Users\\Almira
F\\Documents\\NetBeansProjects\\NyobaNLP\\en-token.bin");
String tes = read.readLine();
TokenizerModel model1 = new TokenizerModel(is);

Tokenizer tokenizer = new TokenizerME(model1);

String tokens[] = tokenizer.tokenize(""+tes);
System.out.println("Contoh Tokenisasi : ");
for (String a : tokens)
    System.out.println(a);
is.close();
System.out.println("-----");

```

#### Kode 4.2 Contoh Kode untuk Tokenisasi

### 4.1.3. Perancangan Pengembangan Prototipe

Pada bagian ini akan menjelaskan mengenai hal-hal apa saja yang perlu dilakukan setelah preproses data dilakukan pada saat akan melakukan implemantasi nantinya. Untuk melakukan pengembangan prototipe, implementasinya akan menggunakan *tools* komputasi secara otomatis dnegan bahasa pemograman Java. Terdapat lima tahapan dalam pengembangan prototipe, yaitu Pemodelan Chunk Parser, Validasi Model, Melakukan Chunking, Sortding Data Hasil Chunking, dan Pembuatan Prototipe.

#### 4.1.3.1. Perancangan Pemodelan Chunk Parser

Konsep utama dari *chunking* adalah menggabungkan token-token dengan jenis kata yang sama, seperti kata benda dan kata kerja sehingga akan didapatkan sebuah bentuk kata yang tidak ambigu dan dapat diproses dengan lebih mudah menjadi sebuah informasi. Meskipun begitu, sebuah *chunk* juga dapat berisi satu kata (Contoh : *she*). Konsep *tag* yang terdapat pada *chunking* terbagi menjadi dua atau tiga bagian, yaitu :

Bagian pertama :

- B : menandai dimulainya sebuah *chunk*
- I : menandai kelanjutan dari sebuah *chunk*
- E : menandai akhir dari sebuah *chunk*.

Bagian kedua :

- NP : *chunk* berupa kata benda
- VP : *chunk* berupa kata kerja.

Bagian ketiga, yang hanya berlaku untuk kata benda, yaitu :

- Singular
- Plural

Berdasarkan konsep tersebut, maka aturan pola yang diterapkan pada sebuah *chunk* dapat dilihat sebagai berikut :

She put the big knives on the table.

Maka, *chunk* yang dapat dihasilkan dari kalimat tersebut adalah sebagai berikut :

She/**B-NP-singular**|**E-NP-singular**  
 put/B-VP  
 the/**B-NP-plural**  
 big/**I-NP-plural**  
 knives/**E-NP-plural**  
 on/B-PP  
 the/**B-NP-singular**  
 table/**E-NP-singular**  
 ./O

Proses pemodelan Chunk Parser ini dilakukan untuk membuat model yang sesuai pada penelitian. Dalam hal ini, model yang akan dibuat menggunakan *library* dari Apache OpenNLP yang bernama *chunking*. *Chunking* telah mempunyai *pre-trained* model, sama halnya seperti model yang digunakan pada tahap tokenisasi sebelumnya. Tahapan utama yang dapat dilakukan untuk melakukan *chunking* yaitu :

- Melakukan tokenisasi  
Melakukan tokenisasi dapat dengan menggunakan method **tokenize()** dari class **whitespaceTokenizer** yang secara umum dapat dilakukan dengan kode berikut :

```
String contoh = "Selamat Datang";
WhitespaceTokenizer whitespaceTokenizer =
WhitespaceTokenizer.INSTANCE;
String[] tokens =
whitespaceTokenizer.tokenize(sentence);
```

#### **Kode 4.3 Contoh Kode Tokenisasi pada Pemodelan Chunker Parser**

- Melakukan POS Tagging terhadap setiap token  
Setelah melakukan tokenisasi, maka dapat dilakukan tagging pada setiap token yang telah dihasilkan dari kalimat dengan menggunakan method **tag()** dari class **POSTaggerME**. Contoh kode yang dapat digunakan untuk melakukan POS Tagging adalah :

```
//memasukkan model untuk POS Tagging
File file = new File("C:/OpenNLP_models/en-
pos-maxent.bin");
POSModel model = new
POSModelLoader().load(file);
//Memanggil model untuk POS Tagging
POSTaggerME tagger = new POSTaggerME(model);
//Melabeli setiap tokenString[] tags =
tagger.tag(tokens);
```

#### **Kode 4.4 Contoh Kode untuk POS Tagging**

- Melakukan load terhadap library **en-chunker.bin** yang dapat digunakan dengan class **ChunkerModel** Model untuk melakukan *chunking* pada sebuah kalimat direpresentasikan oleh sebuah class yang bernama **ChunkerModel**, yang terdapat dalam *package* OpenNLP Chunker.  
Untuk memasukkan model tersebut, maka dapat dibuat sebuah objek **InputStream** dari model tersebut dan

menginisiasi class **ChunkerModel** seperti pada contoh berikut :

```
InputStream inputStream = new
FileInputStream("C:/OpenNLP_models/en-
chunker.bin");
ChunkerModel chunkerModel = new
ChunkerModel(inputStream);
```

**Kode 4.5 Contoh Kode untuk Melakukan Load**

- Menginisiasi class **ChunkerME**

Class **ChunkerME** yang terdapat pada *package* **OpenNLP Chunker** tersebut terdiri dari berbagai *methods* untuk melakukan *chunk* terhadap kalimat. Untuk menginisiasi class tersebut dan memanggil model yang telah di load sebelumnya dapat dilakukan seperti berikut :

```
//Memanggil model
ChunkerME chunkerME = new
ChunkerME(chunkerModel);
```

**Kode 4.6 Contoh Kode untuk Menginisiasi Class  
ChunkerME**

- Melakukan *chunk* terhadap kalimat dengan menggunakan method **chunk()** yang terdapat pada class tersebut.  
Method **chunk()** dari class **ChunkerME** digunakan untuk melakukan *chunk* terhadap kalimat teks mentah yang terdapat pada data. Hasil dari *chunks* dapat ditampilkan dengan menggunakan *array* dari token dan tag yang telah dibuat sebelumnya sebagai parameter yang dapat dilakukan seperti berikut :

```
//menampilkan hasil chunks
String result[] = chunkerME.chunk(tokens,
tags);
```

#### Kode 4.7 Contoh Kode untuk Menampilkan Hasil Chunks

Lebih jelasnya, tahapan pemodelan Chunk Parser dapat dilihat pada gambar berikut :



**Gambar 4.5 Alur Pemodelan Chunk Parser**

Hal selanjutnya yang dapat dilakukan adalah memilih *method* yang akan digunakan untuk membuat aplikasi yang sesuai dalam penyelesaian permasalahan pada penelitian ini serta parameter yang dibutuhkan untuk membuat aplikasi. Daftar *method* serta parameter yang terdapat pada *library* Chunk Parser dan akan digunakan pada pembuatan pemodelan *Chunk Parser* yang sesuai dapat dilihat pada tabel berikut :

**Tabel 4.7 Daftar Method dan Parameter yang Digunakan**

<i>Method</i>	<i>Parameter</i>	<i>Hasil</i>
<b>chunk</b>	toks, tags	Daftar dari token yang telah mempunyai <i>chunk tags</i> pada satu <i>sequence</i>
<b>chunk[]</b>	toks, tags	Daftar dari token yang telah mempunyai <i>chunk tags</i> pada satu <i>sequence</i> yang dimasukkan ke

		dalam sebuah <i>array</i>
<b>chunkAsSpans</b>	toks, tags	Sebuah <i>array</i> yang berisi rentang <i>chunk tags</i> pada token dalam satu <i>sequence</i>

Setelah memilih *methods* dan parameter yang sesuai untuk pemodelan *chunk parser* tersebut, maka hal selanjutnya yang dapat dilakukan adalah membuat pemodelan *chunk parser* yang sesuai. Bahasa pemrograman yang digunakan adalah Java 8.

#### 4.1.3.2. Perancangan Validasi Model

Langkah selanjutnya yang dapat dilakukan adalah melakukan validasi terhadap model yang telah dibuat, telah memenuhi akurasi yang dibutuhkan atau tidak. Untuk melakukan pengukuran terhadap validasi, Apache OpenNLP telah menyediakan *method* yang dapat digunakan untuk mengukur validasi tersebut, yaitu Precision, Recall dan F-Measure. Hasil dari proses yang diharapkan untuk masing-masing parameter tersebut agar model dapat digunakan untuk tahap selanjutnya adalah :

- Precision : 0.80 (80%)
- Recall : 0.80 (80%)
- F-Measure : 0.70 (70%)

Untuk melakukan perancangan terhadap validasi model, maka dibutuhkan data *training* dan data *testing* untuk menguji apakah model tersebut telah valid. Data *training* dan *testing* untuk melakukan uji validasi terhadap model tersebut diambil berdasarkan data dari CoNLL2000, yang berisi data untuk melakukan pelatihan dan validasi untuk pemodelan *chunking*. Data tersebut terdiri dari tiga kolom yang dipisahkan oleh spasi. Setiap kata diletakkan pada garis pemisah dan ada baris kosong

setelah setiap kalimat, seperti yang terdapat pada contoh sebelumnya. Data ini diambil dari *WallStreet Journal Corpus* (WSJ Corpus) [22]. Hampir keseluruhan dari tipe *chunk* tersebut memiliki dua tipe *tag*, yaitu I-CHUNK untuk kata yang terkait satu sama lain dalam *chunk* dan B-CHUNK untuk kata pertama dari *chunk* tersebut.

Untuk melakukan uji validasi model dengan menggunakan Apache OpenNLP, maka langkah awal yang dapat dilakukan adalah membuka file OpenNLP pada Command Prompt. Selanjutnya, masukkan perintah berikut :

```
$ opennlp ChunkerEvaluator -model en-
chunker.bin -data en-chunker.eval -encoding
UTF-8
```

#### **Kode 4.8 Contoh Kode untuk Melakukan Validasi Terhadap Model**

Setelah itu, akan keluar hasil evaluasi dari model tersebut yang terdiri dari tiga parameter pengukuran, yaitu Precision, Recall dan F-Measure. Contoh hasil dari evaluasi yang dilakukan Apache OpenNLP terhadap model adalah :

```
Precision: 0.9255923572240226
Recall: 0.9220610430991112
F-Measure: 0.9238233255623465
```

#### **Kode 4.9 Contoh Hasil Evaluasi terhadap Validasi Terhadap Model**

Jika setelah diuji validasi model menghasilkan nilai tersebut, maka model telah siap untuk digunakan pada data yang tersedia. Jika model tidak menghasilkan nilai yang diharapkan tersebut, maka akan dilakukan rekonstruksi ulang terhadap model hingga model tersebut mencapai hasil yang sesuai dengan nilai yang diharapkan.

### **4.1.3.3. Melakukan Chunking**

Setelah dilakukan pengujian terhadap model, maka akan dihasilkan model yang akurasi telah sesuai. Maka langkah

selanjutnya yang dapat dilakukan adalah melakukan proses *chunking* terhadap data yang telah tersedia.



**Gambar 4.6 Alur Melakukan Chunking**

Gambar tersebut menunjukkan alur melakukan *chunking* pada penelitian ini. proses *chunking* tersebut dilakukan di *platform* Netbeans dan akan menghasilkan data *chunks* dari ulasan pelanggan.

Untuk melakukan proses *chunking* ini, telah terdapat *pseudocode* yaitu contoh alur kode yang akan diimplementasikan pada tahap selanjutnya yang berasal dari penelitian sebelumnya yang dijadikan acuan untuk penelitian ini [3]. *Pseudocode* yang telah dibuat tersebut dapat dilihat pada gambar berikut :

**Algorithm 1 Extracting Verb Expressions From a Sentence**

Input: a sentence *sent*, the maximum number words in any verb expression *K*

Output: a list of verb expressions *VE* represented by their starting and ending positions

```

1: chunks ← Chunker.chunk(sent)
2: VE ← ∅
3: for each chunki ∈ chunks s.t. i ∈ [1, chunks.length] do
4:   if chunki.label == VP and chunki contains verbs other than be then
5:     // extraction begins from VP with be verbs excluded
6:     chunkCnt ← 1
7:     start ← chunki.start
8:     end ← chunki.end
9:     // add optional NP to the left of chunki
10:    if i > 1 and chunki-1.label == NP then
11:      start ← chunki-1.start
12:      chunkCnt ← chunkCnt + 1
13:    end if
14:    // add optional PP, PRT, ADJP, ADVP, NP to the right
15:    for j = i + 1 to chunks.length do
16:      if chunkCnt == K then // the maximum number of tokens in verb expressions is reached
17:        break
18:      end if
19:      if chunkj.label ∈ {PP, PRT, ADJP, ADVP, NP} then
20:        end ← chunkj.end
21:        chunkCnt ← chunkCnt + 1
22:      end if
23:    end for
24:    VE ← VE ∪ < start, end >
25:  end if
26: end for

```

Gambar 4.7 Pseudocode untuk Melakukan Chunking

#### 4.1.3.4. Melakukan Sorting Data

Agar informasi mengenai *chunks* tersebut dapat dipahami dengan lebih baik, maka langkah selanjutnya yang dapat dilakukan adalah dengan melakukan *sorting* terhadap *chunks* yang dihasilkan. Proses melakukan *sorting* dapat dilihat pada gambar berikut :



Gambar 4.8 Alur Sorting Data

*Sorting* berfungsi untuk mengurutkan data. Data akan diurutkan berdasarkan *chunk* yang muncul paling banyak hingga paling sedikit, sehingga pengguna dapat dengan mudah mengetahui aktivitas apa yang paling sering dilakukan dengan aplikasi tersebut, sesuai dengan tujuan dari penelitian ini. *Sorting* data dilakukan dengan menggunakan Java 8 dari data hasil teks *chunking*. Hasil *sorting* data yang dilakukan akan dimasukkan ke dalam sebuah basis data sebagai hasil akhir dari penelitian.

Metode yang digunakan untuk melakukan *sorting* terhadap data tersebut adalah dengan menggunakan *class* yang terdapat pada Java yang disebut dengan *HashMap*.

*HashMap* merupakan *implementation class* dari *Map*. *Class* ini menggunakan sebuah *method hashtable* untuk mengimplementasikan *Map* tersebut. Dengan *method* tersebut, waktu eksekusi dari operasi sederhana seperti *get()* dan *put()* dapat menjadi konstan meskipun diaplikasikan terhadap data yang besar jumlahnya [23]. Oleh karena itu, *HashMap* sesuai untuk diimplementasikan pada penelitian ini karena dapat digunakan untuk *sorting* data pada penelitian ini yang berjumlah cukup besar.

*Map* merupakan sebuah *interface* yang mempunyai fungsi untuk memetakan nilai dengan sebuah *key* yang bersifat unik. *Hashmap* juga sangat bermanfaat sebagai *memory record management*, dimana dengan mengaplikasikan *HashMap* maka dapat mengurangi *space* yang digunakan karena setiap *record* akan disimpan dalam *Map* tersebut. Berikut merupakan daftar *constructor* yang akan digunakan :

Tabel 4.8 *Constructor* pada *Class HashMap*

<i>Constructor</i>	<i>Parameter</i>	<i>Hasil</i>
<b>HashMap()</b>	-	Membuat HashMap sederhana
<b>HashMap()</b>	Map m	Menginisialisasi HashMap dengan menggunakan element yang terdapat pada sebuah Map dengan variabel m
<b>HashMap</b>	int capacity	Menginisialisasi kapasitas dari HashMap yang diberikan sebuah nilai integer dengan kapasitas tertentu
<b>HashMap()</b>	int capacity, float fillRatio	Menginisialisasi kapasitas dan rasio dari HashMap dengan menggunakan argumennya tersebut. [24]

Hasil pada sorting data tersebut akan disimpan ke dalam sebuah dokumen dengan format .csv (*Comma Separated Value*) yang akan dibuat dengan menggunakan *library* OpenCSV. Dengan *library* tersebut dapat dilakukan penulisan ke file baru maupun membaca sebuah file dalam format data yang sama, yaitu .csv. Sistem akan menuliskan seluruh data yang telah berhasil disorting ke dalam sebuah file baru agar dapat dilakukan analisis dengan lebih mudah. Contoh penulisan data ke file csv dengan *library* ini :

```
CSVWriter writer = new CSVWriter(new
FileWriter("yourfile.csv"), '\t');
//masukkan data dalam bentuk array (ubah data terlebih ke
bentuk array)
    String[] entries = "first#second#third".split("#");
    writer.writeNext(entries);
    writer.close();
```

#### **Kode 4.10 Contoh Kode untuk Menulis Data dalam Bentuk CSV**

## 4.2. Perancangan Validasi Prototipe

Untuk melakukan validasi terhadap prototipe, maka dilakukan *usability testing* terhadap prototipe yang telah dibuat. *Usability testing* adalah sebuah metode yang digunakan untuk menguji validitas dari perspektif kegunaan aplikasi tersebut. *Usability testing* melakukan pendekatan terhadap calon pengguna dari sistem yang telah dibuat tersebut dengan menguji coba prototipe yang telah tersedia ke pengguna.

Untuk melakukan *usability testing* tersebut, maka akan dibuat kuisisioner yang bertujuan untuk menilai apakah aplikasi telah sesuai dengan kebutuhan pengguna dan telah menyelesaikan permasalahan yang ada. Dari hasil dari *usability testing* tersebut, akan didapatkan kesimpulan apakah prototipe tersebut telah berhasil menyelesaikan permasalahan yang ada dan apakah telah dapat memenuhi kebutuhan pengguna.

*Usability Testing* tersebut akan dilakukan dengan menggunakan suatu kuisisioner dengan mengadopsi *System Usability Scale* (SUS) yang merupakan sebuah alat yang dibuat oleh John Brooke pada tahun 1986 dan digunakan untuk mengevaluasi berbagai macam produk dan layanan.

Berdasarkan *System Usability Scale* tersebut, terdapat 10 item kuisisioner dengan masing-masing 5 respon terhadap setiap item yang diukur, mulai dari Sangat Tidak Setuju hingga Sangat Setuju untuk melakukan *usability testing* pada penelitian ini. Sepuluh item tersebut adalah sebagai berikut :

- Saya merasa bahwa saya informasi mengenai ulasan pelanggan yang terdapat pada sistem ini berguna
- Saya berpikir sistem ini sangat rumit dan tidak perlu
- Saya berpikir bahwa sistem ini mudah digunakan
- Saya berpikir saya akan membutuhkan bantuan seseorang untuk dapat menggunakan sistem ini
- Saya merasa bahwa informasi yang terdapat dalam sistem ini memberikan pengetahuan yang lebih terkait dengan ulasan pelanggan
- Saya berpikir terlalu banyak inkonsistensi terkait dengan informasi yang ditampilkan dari sistem ini

- Saya merasa banyak orang akan memahami informasi yang dimuat dalam sistem ini secara efektif
- Saya merasa informasi yang dimuat dalam sistem ini sulit untuk dipahami.
- Saya merasa sangat percaya diri dalam menggunakan sistem ini
- Saya perlu belajar banyak hal sebelum dapat menggunakan sistem ini

Setelah membuat pertanyaan tersebut, untuk menghitung apakah sistem yang telah dikembangkan ini telah sesuai, maka terdapat beberapa aturan, yaitu :

- Untuk pertanyaan pada nomor ganjil, kurangi 1 dari setiap jawaban responden
- Untuk pertanyaan pada nomor genap, kurangi nilai yang didapat dari responden dengan 5 pada setiap jawaban responden
- Skala dirubah menjadi 0-4 (4 menjadi respon yang paling positif)
- Seluruh tanggapan dari pengguna dikonversi dan dijumlahkan dan dikalikan dengan 2,5. Dengan dilakukannya hal tersebut, maka nilai dari responden akan berubah menjadi rentang nilai yang mungkin dari 0 sampai 100.

*Halaman ini sengaja dikosongkan*

## **BAB V IMPLEMENTASI**

Bab ini berisi tentang proses implementasi dari perancangan yang telah dilakukan sebelumnya. Implementasi terhadap model yang akan membahas mengenai perangkat penelitian apa yang digunakan, penerapan ekstraksi data ulasan pelanggan yang diambil dari basis data, penerapan pra proses data, penerapan validasi model, penerapan model untuk melakukan *chunking* dan *sorting* data yang dilakukan dengan menggunakan bahasa pemrograman Java, dan penerapan validasi prototipe menggunakan metode *usability testing*.

### **5.1. Perangkat Penelitian**

Dalam pelaksanaan penelitian rancang bangun ekstraksi ekspresi kata kerja pada ulasan pelanggan yang berguna untuk memaparkan pengalaman penggunaan produk dengan menggunakan *text chunking*, dibutuhkan perangkat-perangkat yang dapat mendukung proses yang ada pada setiap tahapan penelitian. Adapun perangkat-perangkat yang dibutuhkan meliputi perangkat keras dan perangkat lunak dengan spesifikasi sebagai berikut :

#### **Perangkat keras**

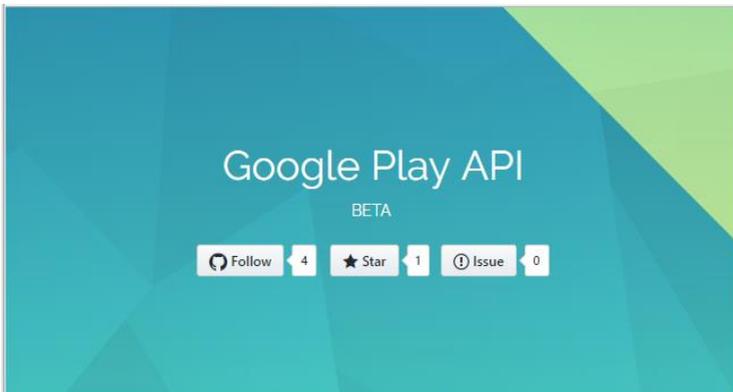
- CPU : Intel ® Core™ i5-2400 CPU @3.10 GHz, RAM 8.00 GB, Harddisk 160 GB SDD
- Monitor : 14 inch 1440x900
- Keyboard dan Mouse

#### **Perangkat Lunak**

- Sistem Operasi : Windows 8 Pro
- Bahasa Pemrograman : Java
- *Tools* : Netbeans IDE 8.2  
Ms. Excel 2016  
Notepad  
Ms. Word 2016

## 5.2. Ekstraksi Data Ulasan Pelanggan

Data mengenai ulasan pelanggan didapatkan dari sebuah aplikasi yang telah dibuat sebelumnya, yang dapat memuat data-data mengenai pelanggan, termasuk ulasan pelanggan dari Google Play dalam bentuk file JSON. Format file JSON tersebut akan lebih memudahkan penelitian ini untuk melakukan *scraping*. Setelah itu, data tersebut akan dimasukkan ke dalam sebuah basis data yang telah dibuat pada penelitian sebelumnya. Aplikasi yang digunakan tersebut memiliki tampilan seperti berikut ini :



Gambar 5.1 Tampilan Aplikasi Google Play Scraper

Untuk mendapatkan ulasan pelanggan dalam satu aplikasi yang ditampilkan pada Google Play, maka parameter yang dapat dimasukkan agar dapat mengakses data tersebut pada aplikasi ini adalah :

- appID : ID aplikasi dari Google Play (mengambil bagian parameter ?id pada URL aplikasi tersebut)
- lang : merepresentasikan bahasa yang digunakan pada ulasan pelanggan

- `sort` : digunakan untuk menampilkan ulasan pelanggan dengan urutan tertentu, yaitu urutan berdasarkan terbaru, rating, dan *helpfulness*
- `page` : menampilkan ulasan pelanggan hingga jumlah halaman tertentu. Setiap halaman mengandung maksimal 40 ulasan, dengan batasan halaman yang dapat dimuat adalah 121 halaman.

Pada penelitian ini, aplikasi yang akan dicoba melakukan *scraping* adalah aplikasi permainan Plant vs Zombies 2. Untuk mengakses data tersebut, maka URL yang dimasukkan adalah :

[https://gooplayapi.herokuapp.com/reviews/com.ea.game.pvz2\\_row/en/helpfulness](https://gooplayapi.herokuapp.com/reviews/com.ea.game.pvz2_row/en/helpfulness)

#### Kode 5.1 URL untuk mengakses data

Maka, data akan ditampilkan dalam bentuk JSON file, seperti berikut ini :

```
{
  • id: "gp:A0qpTOHVN15D0ZLqWpFA-79vXpwbpFDdQ5yLxcI3KUH1DX19DVZxxZUzSoIyUZjQDxVeVezpRbFDzVsM4GE78IQ",
  • userName: "Tim Lacey",
  • userImage: "https://lh6.googleusercontent.com/-nV1nn9tcmxA/AAAAAAAAAI/w96-h96-p/photo.jpg",
  • date: "May 6, 2017",
  • url: "https://play.google.com/store/apps/details?id=com.ea.game.pvz2\_row&reviewId=Z3A6QU9xcFRP\_SFZOMTVEMFpMcVdwZkEtNz12WHB3YnBGRGRRNXlMeGNJM0tVSGxEWDE5RFZaeFpVe1NvSX1VWmpRRHhWZVZ1enBSYkZE1ZzTTRHRTc4SVE",
```

```

• score: 5,
• title: "",
• text: "This is my favourite game ever! just
one thing; you know the speed up button?
Well lots of people want a slow down button.
please make it!"
}

```

### Kode 5.2 Contoh Format Data yang Memuat Data Ulasan Pelanggan

Untuk menampilkan data tersebut, parameter yang dimasukkan adalah *sort by helpfulness* dan *appID* yang sesuai dengan aplikasi *Plants vs Zombies 2* tersebut. Hasil dari data yang ditampilkan disesuaikan dengan kolom yang ada pada basis data yang telah dibuat pada tahap perancangan. Data tersebut lalu akan dimasukkan ke basis data sesuai dengan kolom yang telah disediakan.

Selanjutnya, akan dilakukan *scraping* data tersebut dengan menggunakan program Java, *library org-json* untuk membaca tipe data JSON, dan MySQL untuk membuat koneksi ke basis data dan memasukkan data hasil *scraping* tersebut ke dalam basis data.

- Langkah pertama adalah membuat koneksi ke basis data dengan *library* yang telah tersedia oleh Netbeans, yaitu MySQL yang dapat dilihat sebagai berikut :

```

static final String JDBC_DRIVER =
"com.mysql.jdbc.Driver";

static final String DB_URL =
"jdbc:mysql://localhost/dataeki";

static final String USER = "root";

static final String PASS = "";

Connection conn = null;

Statement stmt = null;

```

### Kode 5.3 Melakukan Koneksi ke Database

- Setelah itu, langkah selanjutnya adalah dengan membaca data dari aplikasi tersebut dengan menggunakan *library* org-json yang membuat program membaca data dengan bentuk JSON seperti berikut :

```
InputStream is = new URL(url).openStream();

JSONArray json = null;

try {

    BufferedReader rd = new
BufferedReader(new InputStreamReader(is,
Charset.forName("UTF-8")));

    String jsonText = readAll(rd);

    json = new JSONArray(jsonText);

} catch (JSONException e) {

    e.printStackTrace();

} finally {

    is.close();

}

return json;

}
```

#### **Kode 5.4 Membaca Data dalam Format JSON**

- Langkah selanjutnya adalah dengan mengambil data dari aplikasi menggunakan *library* org-json yang membuat program dapat mengambil data dengan bentuk JSON seperti berikut :

```

try {
int i;
for (i = 0; i <= 111; i++) {
JSONArray json =
readJsonFromUrl("https://gooplayapi.herokuapp.com/reviews/com.ea.game.pvz2_row/en/helpfulness/0"+i);
for(int x = 0; x < json.length(); x = x + 1)
{
JSONObject obj = (JSONObject)json.get(x);
String id = obj.get("id").toString();
String uname =
obj.get("userName").toString();
String uimage =
obj.get("userImage").toString();
String date = obj.get("date").toString();
String url = obj.get("url").toString();
int score = (Integer) obj.get("score");
String title = obj.get("title").toString();
String test = obj.get("text").toString();

```

#### **Kode 5.5 Membaca Data Bentuk JSON**

- Langkah terakhir adalah dengan memasukkan data dan menyimpan data tersebut ke dalam basis data yang telah dilakukan koneksi pada langkah sebelumnya seperti berikut :

```

try{

Class.forName("com.mysql.jdbc.Driver");
System.out.println("Connecting to
database...");
conn =
DriverManager.getConnection(DB_URL,USER,PAS
S);
System.out.println("Creating
statement...");
Calendar calendar =
Calendar.getInstance();

```

```
        java.sql.Date startDate = new
java.sql.Date(calendar.getTime().getTime());
// the mysql insert statement
        String query = " insert into reviews (id,
app_id, userName, userImage, date, url, score,
title, text) "
            + " values (?, ?, ?, ?, ?, ?, ?, ?, ?)";
        PreparedStatement preparedStmt =
conn.prepareStatement(query);
        int a=1;
        String appid = "com.ea.game.pvz2_row";
        preparedStmt.setString(a++, id);
        preparedStmt.setString(a++, appid);
        preparedStmt.setString(a++, uname);
        preparedStmt.setString(a++, uimage);
        preparedStmt.setString(a++, date);
        preparedStmt.setString(a++, url);
        preparedStmt.setInt(a++, score);
        preparedStmt.setString(a++, title);
        preparedStmt.setString(a++, test);
preparedStmt.execute();
        conn.close();
    }catch(SQLException se){
        se.printStackTrace();
    }catch(Exception e){
        e.printStackTrace();
    }finally{
        try{
            if(stmt!=null)
                stmt.close();
        }catch(SQLException se2){
        }
    }
```

```

try{
    if(conn!=null)
        conn.close();
}catch(SQLException se){
    se.printStackTrace();
};//end finally try
};//end try
}
}
} catch (IOException e) {
    e.printStackTrace();
} catch (JSONException e) {
    e.printStackTrace();
}
}
}
//System.out.println(callURL("https://goo
playapi.herokuapp.com/reviews/com.facebook.kata
na/en/sort.NEWEST/0"));
public static String readAll(Reader
rd) throws IOException {
    StringBuilder sb = new StringBuilder();
    int cp;
    while ((cp = rd.read()) != -1) {
        sb.append((char) cp);
    }
    return sb.toString();
}
}

```

### **Kode 5.6 Memasukkan Data**

Setelah melakukan proses tersebut, data akan tersimpan di dalam basis data dengan jumlah yang sesuai dengan limit maksimum halaman yang ditampilkan, yaitu 121 halaman. Jika satu halaman memuat 40 ulasan, maka total data yang didapatkan dan dimasukkan ke dalam basis data adalah 4840 ulasan pelanggan. Basis data yang memuat data ulasan pelanggan tersebut dapat dilihat pada gambar berikut :

id	app_id	userName	userImage
gp:AOqpTOE-2ddcAL9rit2UG3y	com.ea.game.pvz2_row	Kevin Isaiah Pascual	https://lh4.googleusercontent.com/gPmG-e7VYg/AAA...
gp:AOqpTOE-76v165uC8YRgzF	com.ea.game.pvz2_row	Varsha Shadaw	https://lh4.googleusercontent.com/xgAvKHWTRc/AAA...
gp:AOqpTOE-9_I4BDIGNeLGYv	com.ea.game.pvz2_row	Alex Mercader	https://lh6.googleusercontent.com/JP6es1fDv5Y/AAA...
gp:AOqpTOE-H264tySJgziDKIS	com.ea.game.pvz2_row	Sheik Madina	https://lh4.googleusercontent.com/VQaP6Q4UIHw/AAA...
gp:AOqpTOE-ic45z7hcBFjLE2kd5v7A6w7Vjp3C6	com.ea.game.pvz2_row	deepali Jadhav	https://lh4.googleusercontent.com/B7Z5NMPsGtE/AAA...
gp:AOqpTOE-JdaCLw1dRwc_R	com.ea.game.pvz2_row	Sweden Alvarez	https://lh5.googleusercontent.com/-b8LCuUYt2E/AAA...
gp:AOqpTOE-mdvlel-sCl-Q3XfLEi-	com.ea.game.pvz2_row	pl mAge	https://lh3.googleusercontent.com/RFuOQ-qeYLk/AAA...

**Gambar 5.2** Contoh Hasil *Scraping* Data (1)

date	url	score	title	text
May 4, 2017	https://play.google.co.id=com....	2		My plant vs zombie 2 app has stopped, how to fix i...
April 28, 2017	https://play.google.co.id=com....	1		Game is not allowing me to log onto after the upda...
April 22, 2017	https://play.google.co.id=com....	5		think the other premium plants can bought using re...
May 4, 2017	https://play.google.co.id=com....	4		It nice
April 25, 2017	https://play.google.co.id=com....	5		Best game for our minds
May 4, 2017	https://play.google.co.id=com....	4		It was a very nice game. More challenges.
April 6, 2017	https://play.google.co.id=com....	4		Everything in this update is awesome specially the...

**Gambar 5.3** Contoh Hasil *Scraping* Data (2)

### 5.3. Implementasi Praproses Data

Praproses data pada penelitian ini terdiri dari beberapa tahap, antara lain memuat data teks dari basis data, segmentasi kalimat menjadi kata, serta melakukan tokenisasi dan *labeling* menggunakan program berbasis Java dan *library* Apache OpenNLP.

### 5.3.1. Memuat Data Teks

Sebelum melakukan praproses data lebih lanjut, perlu dilakukan proses pemuatan data teks yang terdapat di dalam basis data agar dapat dibaca oleh program dan dilakukan proses-proses selanjutnya. Proses pemuatan data teks tersebut dilakukan dengan program berikut :

```
public static ArrayList<String> getTextFromDB(String uname,
String pass, String db, String table, int row){
ArrayList<String> textDB = new ArrayList<>();
try
{
Class.forName("com.mysql.jdbc.Driver");
Connection con=(Connection) DriverManager.getConnection(
"jdbc:mysql://localhost:3306/"+db,uname,pass);
String sqlSelect = "select app_id, text "
+ "from "+table+" limit 0,"+row;
Statement stmtSelect=con.createStatement();
ResultSet rs=stmtSelect.executeQuery(sqlSelect);
while(rs.next()){
if(rs.getString(2)!=null)
{
System.out.println(rs.getString(2))
textDB.add(rs.getString(2)
.replaceAll("https?://\\S+\\s?", ""))
.replaceAll("\n", " ").trim());}
else
{rs.next();}}
con.close();}
catch(ClassNotFoundException | SQLException e)
{System.out.println(e);}
return textDB;
}}
```

#### Kode 5.7 Fungsi getTextFromDB Untuk Pemuatan Data Teks

Untuk dapat memuat data teks, dibutuhkan koneksi Java ke basis data MySQL menggunakan *library* MySQL yang terdapat pada Netbeans dengan mysql-connector. Dengan melakukan koneksi dengan database MySQL, akan didapatkan setiap baris dari hasil query yang dimasukkan tersebut. Setelah itu, setiap kalimat yang ada pada setiap baris dalam basis data yang

diambil tersebut akan dilakukan perubahan untuk menyesuaikan dengan bahasa pemrograman Java, diantaranya menghapus baris baru, dan menghapus tanda baca seperti tanda tanya, tanda koma dan tanda seru. Kemudian, setiap kalimat tersebut akan disimpan kembali dalam class `ArrayList<String> textDB`.

Setelah itu, untuk memuat data yang terdapat pada basis data tersebut, maka dimasukkan kode untuk melakukan pemuatan data pada Fungsi *Main*. Data yang dibuat sejumlah 4840 data menggunakan pemanggilan fungsi `getTextFromDB` dengan *argument row* bernilai 4000. Lalu, akan dilakukan pemuatan kalimat agar dapat dilakukan proses lebih lanjut dengan menggunakan variabel `Textku` sebagai berikut :

```
int rowNum = 1000;
ArrayList<String> myText =
getTextFromDB("root", "", "dataeki", "reviews"
, rowNum);
String Textku = myText.toString();
```

#### Kode 5.8 Pemanggilan Fungsi `getTextFromDB` pada *Main*

Hasil ekstraksi data pemuatan data teks ke dalam program dapat dilihat pada tabel berikut :

**Tabel 5.1 Sampel Hasil Pemuatan Data**

My plant vs zombie 2 app has stopped, how to fix it without clearing the data.

Game is not allowing me to log onto after the update. Please fix

think the other premium plants can bought using real money or gems, and the others can only be bought with gems please read this!

It is a nice game

Best game for our minds

It was a very nice game. More challenges.

Everything in this update is awesome specially the new purchasable plants that costs a hundred gems but I just noticed that Sun Bean is not doing it's thing. I think he better get fixed. Also, can I have my Peashooter? It's been a long time without a basic Peashooter.

For some reason I'm not getting gem jam 3 or 4 but I have the update could you fix that

Great

Very pleased that Sun Bean is back like the old days. Homing Thistle still attacking air a lot. But with recent level up, it's attack is upgraded and it's not bad. Sadly Surfer Zombie and Balloon Zombie also upgraded. Surfer Zomble able to smash a plant and throwing another surf board while dying. While Balloon Zombie sometime able to re inflate the balloon after it's been exploded. Pinata is missing (the pinata party and seed pinata timer) when the seed pinata timer reaching zero, only to be available the next seed pinata travelog quest which can be the next 4 hours, it's very vexing.

### 5.3.2. Melakukan Segmentasi Kalimat

Setelah pemuatan data teks berhasil dilakukan, maka langkah selanjutnya yang merupakan tahapan awal yang dilakukan untuk membuat pemodelan *chunk parser* ini adalah dengan melakukan segmentasi kalimat. Segmentasi kalimat dilakukan untuk memecah ulasan pelanggan menjadi kalimat-kalimat agar mendapatkan frase yang tepat pada tahap-tahap selanjutnya. Untuk melakukan hal ini, digunakan salah satu model yang terdapat pada *library* OpenNLP, yaitu Sentence Detector. Model ini mampu mendeteksi kalimat-kalimat yang terdapat pada data mentah ulasan pelanggan yang masih berupa paragraf-paragraf.

Berdasarkan penjelasan dan perancangan yang telah dilakukan pada tahap sebelumnya, untuk melakukan segmentasi kalimat pada prototipe ini, kode yang digunakan dapat dilihat sebagai berikut :

```

InputStream input = new
FileInputStream("C:\\Users\\Almira
F\\Documents\\NetBeansProjects\\NyobaNLP\\
en-sent.bin");
SentenceModel model2 = new
SentenceModel(input);
SentenceDetectorME sdetector = new
SentenceDetectorME(model2);
String[] sentences =
sdetector.sentDetect(Textku);
List<String> tokens = new
ArrayList<String>();

```

### **Kode 5.9 Kode yang Digunakan untuk Segmentasi Kalimat**

Pemisahan kalimat dilakukan untuk mendapatkan frase yang sesuai pada setiap ulasan pelanggan, karena pada satu kalimat dengan kalimat lainnya dapat menciptakan frase-frase yang berbeda dan perbedaan tersebut dapat menyebabkan interpretasi hasil ekspresi kata kerja yang berbeda sehingga perlu dilakukan segmentasi terhadap kalimat-kalimat tersebut untuk mendapatkan setiap frase yang sesuai. Sampel hasil segmentasi kalimat dapat dilihat pada tabel berikut :

**Tabel 5.2 Hasil Segmentasi Kalimat**

My plant vs zombie 2 app has stopped, how to fix it without clearing the data.

Game is not allowing me to log onto after the update.

Please fix.

think the other premium plants can bought using real money or gems, and the others can only be bought with gems please read this!

It nice

Best game for our minds.

It was a very nice game.

More challenges.

Everything in this update is awesome specially the new purchasable plants that costs a hundred gems but I just noticed that Sun Bean is not doing it's thing

I think he better get fixed.

Also, can I have my Peashooter?

It's been a long time without a basic Peashooter.

For some reason I'm not getting gem jam 3 or 4 but I have the update could you fix that

great.

Great....

Very pleased that Sun Bean is back like the old days.

Homing Thistle still attacking air a lot.

But with recent level up, it's attack is upgraded and it's not bad.

Sadly Surfer Zombie and Balloon Zombie also upgraded.

Its all gone!

Surfer Zomble able to smash a plant and throwing another surf board while dying.

While Balloon Zombie sometime able to re inflate the balloon after it's been exploded.

Pinata is missing (the pinata party and seed pinata timer) when the seed pinata timer reaching zero, only to be available the next seed pinata travelog quest which can be the next 4 hours, it's very vexing.

This game is awesome.

.I am addicted with this game.,

Boring Your plants are expensive and some of them can't be exchanged by gems.

Please make it possible more plants by gems only.

No more plant is made to purchase by gems Disgusting.

### 5.3.3. Melakukan Tokenisasi

Tokenisasi dilakukan dengan menggunakan *library* yang telah disediakan an oleh Apache OpenNLP pada *class*

WhitespaceTokenizer dengan fungsi tokenizing. Ketika terdapat suatu kalimat yang akan menjadi masukan pada fungsi tokenizing tersebut, maka kata-kata pada suatu kalimat tersebut akan diubah menjadi bentuk token. Kemudian, token-token tersebut akan disimpan kembali dalam suatu ArrayList. Implementasi segmentasi kalimat dan tokenisasi tersebut adalah sebagai berikut :

```
String[] token = new String[]{};
WhitespaceTokenizer whitespaceTokenizer =
WhitespaceTokenizer.INSTANCE;
for(int l = 0; l < token.length; l++){
tokens.add(token[l]); }
//System.out.println("Sente: "+tokens);
}
String[] tokenAll = new String[tokens.size()];
tokenAll = tokens.toArray(tokenAll);
```

#### Kode 5.10 Fungsi **tokenizing** pada *Main Class*

Hasil dari implementasi kode yang digunakan untuk melakukan tokenisasi pada prototipe ekstraksi ekspresi kata kerja pada ulasan pelanggan ini dapat dilihat pada tabel berikut ini :

**Tabel 5.3 Hasil Tokenisasi pada Beberapa Ulasan Pelanggan**

Teks	Hasil Tokenisasi
Hasil Segmentasi Kalimat	More challenges., Everything in this update is awesome specially the new purchasable plants that costs a hundred gems but I just noticed that Sun Bean is not doing it's thing
Hasil Tokenisasi	[More, challenges, ., ,, Everything, in, this, update, is, awesome, specially, the, new, purchasable, plants, that, costs, a, hundred, gems, but, I, just, noticed, that, Sun, Bean, is, not, doing, it, 's, thing]

Hasil Segmentasi Kalimat	think the other premium plants can bought using real money or gems, and the others can only be bought with gems please read this!
Hasil Tokenisasi	[Think, the, other, premium, plants, can, bought, using, real, money, or, gems, and, the, others, can, only, be, bought, with, gems, please, read, this, !]
Hasil Segmentasi Kalimat	Everything in this update is awesome specially the new purchasable plants that costs a hundred gems but I just noticed that Sun Bean is not doing it's thing
Hasil Tokenisasi	[Everything, in, this, update, is, awesome, specially, the, new, purchasable, plants, that, costs, a, hundred, gems, but, I, just, noticed, that, Sun, Bean, is, not, doing, it, 's, thing]
Hasil Segmentasi Kalimat	Also, can I have my Peashooter?
Hasil Tokenisasi	[Also, can, I, have, my, Peashooter, ?]

## 5.4. Implementasi Pengembangan Prototipe

Pada bagian ini, akan dilakukan implementasi terhadap perancangan dari pengembangan prototipe yang akan dibuat dengan menggunakan *library* dari Apache OpenNLP dan menggunakan bahasa pemograman Java.

### 5.4.1. Melakukan *Labeling* dengan POS Tagging

Setelah seluruh kalimat yang terdapat pada basis data tersebut diubah ke dalam bentuk token, maka selanjutnya akan dilakukan pemberian *label* terhadap token-token pada kalimat tersebut dengan menggunakan model yang telah disediakan oleh Apache OpenNLP dengan bahasa pemograman Java. Proses *labeling* dilakukan dengan memanggil model yang telah diunduh dan melakukan *load* pada model tersebut agar dapat

melakukan proses POS Tagging tersebut. Proses *labeling* tersebut dilakukan sebagai berikut :

```
File file = new File("C:\\Users\\Almira
F\\Documents\\NetBeansProjects\\NyobaNLP\\en-pos-
maxent.bin");
POSModel model = new POSModelLoader().load(file);
//Constructing the tagger
POSTaggerME tagger = new POSTaggerME(model);
```

#### Kode 5.11 Labeling Token

Hasil dari *labeling* token tersebut akan disimpan kembali dalam sebuah Array yang berisi token dan hasil *label* dari setiap token tersebut dengan variabel `tags` seperti berikut :

```
//Generating tags from the tokens
String[] tags = tagger.tag(tokens);
```

#### Kode 5.12 Menyimpan Hasil Labeling ke Array

Sampel hasil dari pemberian label pada setiap kata yang telah ditokenisasi dengan menggunakan model POS Tagging OpenNLP dapat dilihat pada tabel berikut ini :

Tabel 5.4 Sampel Hasil Labeling Kata

No	Kata	Label yang Dihasilkan
1	<b>Very</b>	RB (Adverb)
2	<b>Pleased</b>	VBD (Verb, Past Tense)
3	<b>That</b>	IN (Preposition)
4	<b>Sun</b>	NNP (Proper Noun, Singular)
5	<b>Bean</b>	NNP (Proper Noun, Singular)
6	<b>Is</b>	VBZ (Verb, 3rd person singular present)
7	<b>Back</b>	RB (Adverb)
8	<b>Like</b>	IN (Preposition)
9	<b>The</b>	DT (Determiner)

10	<b>Old</b>	JJ (Adjective)
11	<b>Days</b>	NNS (Noun, plural)
12	<b>Homing</b>	NN (Noun, singular or mass)
13	<b>Thistle</b>	NN (Noun, singular or mass)
14	<b>Still</b>	RB (Adverb)
15	<b>Attacking</b>	VBG (Verb, gerund or past participle)
16	<b>Air</b>	NN (Noun, singular or mass)
17	<b>A</b>	DT (Determiner)
18	<b>Lot</b>	NN (Noun, singular or mass)
19	<b>But</b>	CC (Coordinating Conjunction)
20	<b>With</b>	IN (Preposition)
21	<b>Recent</b>	JJ (Adjective)
22	<b>Level</b>	NN (Noun, singular or mass)
23	<b>Up</b>	RB (Adverb)
24	<b>It</b>	PRP (Personal Pronoun)
25	<b>'s</b>	VBZ (Verb, 3rd person singular present)
26	<b>Attack</b>	NN (Noun, singular or mass)
27	<b>Is</b>	VBZ (Verb, 3rd person singular present)
28	<b>Upgraded</b>	VBN (Verb, past participle)
29	<b>And</b>	CC (Coordinating Conjunction)
30	<b>It</b>	PRP (Personal Pronoun)
31	<b>'s</b>	VBZ (Verb, 3rd person singular present)
32	<b>Not</b>	RB (Adverb)
33	<b>Bad</b>	JJ (Adjective)
34	<b>Sadly</b>	NNP (Proper Noun, Singular)
35	<b>Surfer</b>	NNP (Proper Noun, Singular)
36	<b>Zombie</b>	NNP (Proper Noun, Singular)
37	<b>And</b>	CC (Coordinating Conjunction)
38	<b>Ballon</b>	NNP (Proper Noun, Singular)
39	<b>Zombie</b>	NNP (Proper Noun, Singular)
40	<b>Also</b>	RB (Adverb)
41	<b>Upgraded</b>	VBD (Verb, Past Tense)
42	<b>Surfer</b>	NNP (Proper Noun, Singular)

43	<b>Zombie</b>	NNP (Proper Noun, Singular)
44	<b>Able</b>	JJ (Adjective)
45	<b>To</b>	TO
46	<b>Smash</b>	VB (Verb, base form)
47	<b>A</b>	DT (Determiner)
48	<b>Plant</b>	NN (Noun, singular or mass)
49	<b>And</b>	CC (Coordinating Conjunction)
50	<b>Throwing</b>	VBG (Verb, gerund or past participle)
51	<b>Another</b>	DT (Determiner)
52	<b>Surf</b>	NN (Noun, singular or mass)
53	<b>Board</b>	NN (Noun, singular or mass)
54	<b>While</b>	IN (Preposition)
55	<b>Dying</b>	VBG (Verb, gerund or past participle)
56	<b>While</b>	IN (Preposition)
57	<b>Ballon</b>	NNP (Proper Noun, Singular)
58	<b>Zombie</b>	NNP (Proper Noun, Singular)
59	<b>Sometime</b>	RB (Adverb)
60	<b>Able</b>	JJ (Adjective)
61	<b>To</b>	TO
62	<b>Re</b>	VB (Verb, base form)
63	<b>Inflate</b>	VB (Verb, base form)
64	<b>The</b>	DT (Determiner)
65	<b>Ballon</b>	NN (Noun, singular or mass)
66	<b>After</b>	IN (Preposition)
67	<b>It</b>	RP (Particle)
68	<b>'s</b>	VBZ (Verb, 3rd person singular present)
69	<b>Been</b>	VCN (Verb, past participle)
70	<b>Exploded</b>	VCN (Verb, past participle)
71	<b>Pinata</b>	NNP (Proper Noun, Singular)
72	<b>Is</b>	VBZ (Verb, 3rd person singular present)
73	<b>Missing</b>	VBG (Verb, gerund or past participle)
74	<b>The</b>	DT (Determiner)
75	<b>Pinata</b>	NN (Noun, singular or mass)

76	<b>Party</b>	NN (Noun, singular or mass)
77	<b>And</b>	CC (Coordinating Conjunction)
78	<b>Seed</b>	NN (Noun, singular or mass)
79	<b>Pinata</b>	NN (Noun, singular or mass)
80	<b>Timer</b>	NN (Noun, singular or mass)
81	<b>When</b>	WRB (Wh-adverb)
82	<b>The</b>	DT (Determiner)
83	<b>Seed</b>	NN (Noun, singular or mass)
84	<b>Pinata</b>	NN (Noun, singular or mass)
85	<b>Timer</b>	NN (Noun, singular or mass)
86	<b>Reaching</b>	VBG (Verb, gerund or past participle)
87	<b>Zero</b>	CD (Cardinal Number)
88	<b>Only</b>	RB (Adverb)
89	<b>To</b>	TO
90	<b>Be</b>	VB (Verb, base form)
91	<b>Available</b>	JJ (Adjective)
92	<b>The</b>	DT (Determiner)
93	<b>Next</b>	JJ (Adjective)
94	<b>Seed</b>	NN (Noun, singular or mass)
95	<b>Pinata</b>	NN (Noun, singular or mass)
96	<b>Travelog</b>	NN (Noun, singular or mass)
97	<b>Quest</b>	NN (Noun, singular or mass)
98	<b>Which</b>	WDT (Wh- determiner)
99	<b>Can</b>	MD (Modal)
100	<b>Be</b>	VB (Verb, base form)
101	<b>The</b>	DT (Determiner)
102	<b>Next</b>	JJ (Adjective)
103	<b>4</b>	CD (Cardinal Number)
104	<b>Hours</b>	NNS (Noun, Plural)
105	<b>It</b>	PRP (Personal Pronoun)
106	<b>'s</b>	VBZ (Verb, 3rd person singular present)
107	<b>Very</b>	RB (Adverb)
108	<b>Vexing</b>	JJ (Adjective)

### 5.4.2. Implementasi Pemodelan Chunk Parser

Pada penelitian ini, akan dilakukan ekstraksi terhadap ekspresi kata kerja yang terdapat pada ulasan pelanggan. Definisi singkat dari ekspresi kata kerja tersebut adalah sebuah gabungan dari beberapa *chunks* yang berpusat pada frase kata kerja (VP) dengan ditambahkan kata benda (NP) pada bagian kiri dari frase kata kerja, dan partikel (PRT), frase kata sifat (ADJP), frase kata keterangan (ADVP) juga kata benda (NP) pada bagian kanan dari frase kata kerja tersebut.

Setelah melakukan proses *labeling* terhadap token yang dihasilkan dari segmentasi kalimat menjadi kata per kata dari basis data, maka langkah selanjutnya yang dapat dilakukan adalah mengimplementasikan pemodelan Chunk Parser yang akan dikembangkan pada penelitian ini. proses pembuatan model chunk parser ini menggunakan *library* dari Apache OpenNLP dan menggunakan bahasa pemograman Java seperti pada proses sebelumnya. Model yang digunakan adalah *chunker* yang dapat melakukan *chunking* terhadap model dengan cara menggabungkan *label* yang telah diberikan pada proses *labeling* dengan menggunakan POS Tagging tersebut menjadi beberapa jenis kata yang lebih utuh dan mudah dicerna informasinya.

Pemodelan chunk parser dilakukan dengan memanggil model yang telah diunduh dan melakukan *load* terhadap model tersebut. Model yang digunakan telah dilakukan proses *training* sebelumnya dan akan melakukan proses *chunking* terhadap token-token yang telah memiliki *label* tersebut yang dapat dilihat sebagai berikut :

```

InputStream inputStream = new
FileInputStream("C:\\Users\\Almira
F\\Documents\\NetBeansProjects\\NyobaNLP\\en-
chunker.bin");
ChunkerModel chunkerModel = new
ChunkerModel(inputStream);
ChunkerME chunkerME = new ChunkerME(chunkerModel);

```

### **Kode 5.13 Pemanggilan Model Chunk Parser**

Setelah memanggil model, maka langkah selanjutnya adalah melakukan *chunk* tersebut dan menyimpan hasil *chunk* tersebut menjadi rentang (*span*) yang dimasukkan ke dalam sebuah `ArrayList`. Proses ini berguna untuk mempermudah proses pembuatan model untuk mengekstraksi kata kerja sesuai dengan tujuan penelitian ini seperti berikut :

```
Span[] span = chunkerME.chunkAsSpans(tokens, tags);
```

### **Kode 5.14 Memasukkan Hasil *Chunk* ke dalam `ArrayList`**

Sebuah ekspresi kata kerja merupakan bagian kalimat yang lebih kecil dari sebuah klausa maupun kalimat utuh tetapi telah cukup untuk menggambarkan pendapat yang berguna. Penelitian ini juga akan membatasi *chunk* yang akan dimasukkan ke dalam ekspresi kata kerja tersebut, sehingga *chunk* yang terdapat dalam satu kerja hanya berjumlah *K chunks*.

Pada penelitian ini, nilai *K* dibuat menjadi 5 *chunks* dikarenakan ekspresi kata kerja biasanya hanya mempunyai 5 *chunks*, sebagai contoh satu VP, satu NP dan PP, ADJP atau ADVP yang bersifat opsional. Setelah *label* hasil dari *chunking* disimpan dalam sebuah `Array`, maka langkah selanjutnya adalah menggabungkan *label chunk* yang sesuai untuk membentuk sebuah ekspresi kata kerja berdasarkan data yang ada sebagai berikut :

```

int count = 0;
int start = 0;
int end = 0;
for (int i = 0; i < span.length; i++) {
if ("VP".equals(span[i].getType())) {
count = 1;
start = span[i].getStart();
end = span[i].getEnd();
if(i>1 && "NP".equals(span[i-1].getType())){
start = span[i-1].getStart();
count++;
}
for (int j = i+1; j <= span.length; j++) {
if(count==5){
break;
}
}
}
}

```

#### **Kode 5.15 Membuat *Chunk* untuk Ekspresi Kata Kerja (1)**

Kode 5.14 menunjukkan kode program yang berfungsi untuk membuat *loop* yang akan mengecek setiap *label* pada *chunk* tersebut. Jika terdapat *chunk* dengan *label* kata kerja (VP), maka program akan mengambil *chunk* tersebut, mengecek Array tersebut dari awal hingga akhir dan menghitungnya. Lalu, setelah mendapatkan *chunk* dengan *label* kata kerja, maka selanjutnya adalah mengecek *chunk* dengan *label* kata benda (NP). Jika terdapat *chunk* dengan *label* kata benda, maka program akan menambahkan kata tersebut setelah kata kerja yang telah didapatkan sebelumnya yaitu di bagian kanan dari kata kerja tersebut.

Lalu, akan dilakukan *looping* lagi terhadap *label* kata benda dan kata kerja yang telah digabungkan tersebut. Jika penggabungan kata kerja dan kata benda tersebut terdiri dari 5 *chunks*, maka penggabungan kata akan dihentikan hanya sampai pada tahap tersebut. Jika tidak, maka akan dilanjutkan dengan :

```

else if (count < 5 && ("PP".equals(span[j].getType()
|| "PRT".equals(span[j].getType()) || "ADVP".equals(span[j].getType()) || "ADJP".equals(span[j].getType()) || "NP".equals(span[j].getType()))){
end = span[j].getEnd();
count++;
break;
}
}
for (int j = start; j < end; j++) {
System.out.print(tokens[j]+" ");
}
System.out.println(count);
System.out.println("");
}
}

```

#### Kode 5.16 Membuat Chunk untuk Ekspresi Kata Kerja (2)

Kode 5.15 menunjukkan kode program untuk menambahkan *chunk* dengan *label* lainnya pada penggabungan kata yang telah terbentuk tersebut yang diletakkan setelah kata benda. *Label* kata yang ditambahkan antara lain kata sifat (ADJP), dan beberapa jenis kata keterangan seperti ADVP, PP dan PRT. Setelah mengambil kata dengan *label* yang sesuai tersebut, maka selanjutnya adalah menampilkan hasil ekspresi kata kerja yang terbentuk dengan melakukan *looping* terhadap token-token kata tersebut dalam satu Array berbentuk *span*.

### 5.4.3. Implementasi Validasi Pemodelan Chunk Parser

Setelah pemodelan terhadap Chunk Parser yang sesuai dengan tujuan penelitian ini, yaitu untuk mengekstraksi ekspresi kata kerja dengan lebih detail dilakukan, maka langkah selanjutnya yang dapat dilakukan adalah melakukan uji validasi terhadap model tersebut. Uji validasi dilakukan dengan data berdasarkan CoNLL-2000 yang dibuat berdasarkan partisi dari *WallStreet Journal Corpus* (WSJ) dan telah digunakan secara luas untuk pelatihan dan uji validasi terhadap pemodelan *chunk parser*. Data tersebut dibagi menjadi dua bagian, yaitu bagian *training*

data dengan jumlah token sebanyak 211727 dan *testing* data dengan jumlah token sebanyak 47377 token. Data tersebut dapat diunduh pada website resmi CoNLL-2000 [25].

<http://www.cnts.ua.ac.be/conll2000/chunking/train.txt.gz>  
<http://www.cnts.ua.ac.be/conll2000/chunking/test.txt.gz>

#### **Gambar 5.4 Data *Training* dan *Testing* untuk Uji Validasi**

Setelah mengunduh data tersebut, maka dilakukan ekstraksi untuk membuka data. Contoh tampilan data berupa satu kalimat yang telah diberi *tag* untuk setiap *chunk* yang muncul pada *training* model sebagai contoh dapat dilihat sebagai berikut :

```
Confidence NN B-NP
in IN B-PP
the DT B-NP
pound NN I-NP
is VBZ B-VP
widely RB I-VP
expected VBN I-VP
to TO I-VP
take VB I-VP
another DT B-NP
sharp JJ I-NP
dive NN I-NP
if IN B-SBAR
trade NN B-NP
figures NNS I-NP
for IN B-PP
September NNP B-NP
, , O
due JJ B-ADJP
for IN B-PP
release NN B-NP
tomorrow NN B-NP
, , O
fail VB B-VP
to TO I-VP
```

```

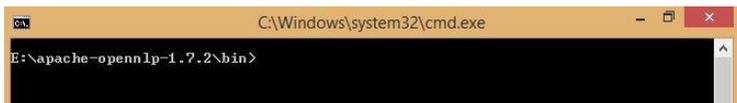
show VB I-VP
a DT B-NP
substantial JJ I-NP
improvement NN I-NP
from IN B-PP
July NNP B-NP
and CC I-NP
August NNP I-NP
's POS B-NP
near-record JJ I-NP
deficits NNS I-NP
. . O

```

**Gambar 5.5 Contoh Data yang Terdapat pada CoNLL-2000 untuk Melakukan Uji Validasi**

Agar data dapat digunakan untuk melakukan *training* model, maka data tersebut harus diubah menjadi format ekstensi `.train` terlebih dahulu. Cara yang dapat dilakukan adalah dengan membuka file tersebut dengan menggunakan Notepad++, lalu menyimpan data tersebut dengan ekstensi file `.train`. Hal ini perlu dilakukan karena OpenNLP hanya dapat membaca data dengan ekstensi file `.train` untuk melakukan *training* pada model tersebut.

Untuk melakukan validasi terhadap model, maka langkah awal yang dilakukan adalah dengan membuka file Apache OpenNLP yang terdapat pada folder `bin` pada Command Prompt yang akan menghasilkan tampilan berikut :



**Gambar 5.6 Tampilan Awal Uji Validasi Model**

Setelah itu, maka langkah selanjutnya adalah dengan memunculkan fitur-fitur dari Apache OpenNLP tersebut sehingga dapat dipilih fitur yang digunakan untuk melakukan *training* dan validasi terhadap model tersebut sebagai berikut :

ChunkerME	learnable chunker
ChunkerTrainerME	trainer for the learnable chunker
ChunkerEvaluator	Measures the performance of the Chunker mode
ChunkerCrossValidator	K-fold cross validator for the chunker
ChunkerConverter	converts ad data format to native OpenNLP fo

**Gambar 5.7** Fitur yang Tersedia Terkait dengan *Chunker*

Untuk melakukan *training* pada model, maka fitur yang dibutuhkan adalah `ChunkerTrainerME` yang akan menghasilkan model. Parameter yang dimasukkan dapat dilihat pada gambar berikut :

```
C:\Windows\system32\cmd.exe - opennlp ChunkerTrainerME -model en-chunk...
E:\apache-opennlp-1.7.2\bin>opennlp ChunkerTrainerME -model en-chunker.bin -lang
en -data traindatachunk.train -encoding UTF-8
Indexing events using cutoff of 5
Computing event counts...
```

**Gambar 5.8** Parameter yang Dimasukkan untuk Melakukan *Training* pada Model

Dapat dilihat pada gambar tersebut, OpenNLP akan menghitung setiap *event* yang akan dilakukan untuk melakukan *training* pada model tersebut. Event tersebut akan sesuai dengan jumlah token yang terdapat pada data, yaitu 211727. Lalu, OpenNLP akan melakukan indexing untuk menggabungkan beberapa *event* yang mungkin untuk iterasi yang digunakan pada model ini adalah sebanyak 100 iterasi. Lebih jelasnya dapat dilihat pada gambar berikut :

```
Computing event counts... done. 211727 events
Indexing... done.
Sorting and merging events... done. Reduced 211727 events to 197252.
Done indexing.
Incorporating indexed data for training...
done.
Number of Event Tokens: 197252
Number of Outcomes: 22
Number of Predicates: 107838
...done.
Computing model parameters ...
Performing 100 iterations ...
```

**Gambar 5.9** Hasil Dari *Training* Terhadap Pemodelan Chunk Parser

OpenNLP akan melakukan perhitungan berdasarkan parameter-parameter yang telah ditentukan untuk membuat model tersebut. Berdasarkan perhitungan tersebut, akan dihasilkan

nilai dari *loglikelihood* dari setiap iterasi yang telah dilakukan. Nilai dari *loglikelihood* merupakan faktor yang akan menentukan tingkat akurasi dari model yang dihasilkan. Contoh hasil *loglikelihood* dari setiap iterasi tersebut dapat dilihat pada gambar berikut :

```

Computing model parameters ...
Performing 100 iterations
1:  -- log likelihood = -54457.145521283      0.26015104335608118
2:  -- log likelihood = -239513.5583724217    0.92600376990044255
3:  -- log likelihood = -141313.1386347234     0.9443387003074745
4:  -- log likelihood = -181803.50883437743    0.9543752899385223
5:  -- log likelihood = -79503.42433952962     0.9603640537106746
6:  -- log likelihood = -66102.71404580091     0.9642416885895516
7:  -- log likelihood = -5956.26275967000      0.9673480100621813
8:  -- log likelihood = -50289.4480193626     0.9695937642246856
9:  -- log likelihood = -45191.66474660764     0.971789143669518
10: -- log likelihood = -41152.072230657305    0.9734941693784921
11: -- log likelihood = -3761.148681609484     0.975388799703893
12: -- log likelihood = -35121.2208384663     0.9765074036936244
13: -- log likelihood = -32799.86067974924     0.9777638185021278
14: -- log likelihood = -30004.79523202407     0.9790612610578747
15: -- log likelihood = -29069.18684952929     0.9802009191008967
16: -- log likelihood = -27544.031862135682    0.9812116546307273
17: -- log likelihood = -26191.889279893687    0.9821098365368012
18: -- log likelihood = -24983.66519338117     0.98294026758217
19: -- log likelihood = -23897.254608339066    0.983762108753253
20: -- log likelihood = -22913.972427533994    0.9844989966108716
21: -- log likelihood = -22019.425221197505    0.9852640820476293
22: -- log likelihood = -21201.692370132845    0.9858874871886911
23: -- log likelihood = -20450.926897685684    0.9865487160352718
24: -- log likelihood = -19758.51466295537     0.9871187605548655
25: -- log likelihood = -19118.944091097902    0.9875830668738517
26: -- log likelihood = -18524.55593877613     0.9881262191406859
27: -- log likelihood = -17971.350264962235    0.9886268638388113
28: -- log likelihood = -17454.83510011787     0.9891780916054355
29: -- log likelihood = -16971.306074681943    0.9896706608037209
30: -- log likelihood = -16517.54956425916     0.99002016747798207
31: -- log likelihood = -16098.764808146195    0.9904085201037184
32: -- log likelihood = -15688.496062665647    0.9908514266012365
33: -- log likelihood = -15308.587166577601    0.9911489795821978
34: -- log likelihood = -14949.136776332851    0.9914323633735895
35: -- log likelihood = -14608.435965917946    0.9917065931128292
36: -- log likelihood = -14284.997484913669    0.9920510846514615
37: -- log likelihood = -13977.470640872889    0.9922872378109546
38: -- log likelihood = -13684.65817685939     0.9925942261029257
39: -- log likelihood = -13405.452422581959    0.992853728456928
40: -- log likelihood = -13138.900145292935     0.993091585532312
41: -- log likelihood = -12884.189629563436    0.99330974194593649
42: -- log likelihood = -12640.297615836464    0.9934916189243695
43: -- log likelihood = -12406.681788181833    0.993680541451964
44: -- log likelihood = -12182.65256240386     0.9939166946114572
45: -- log likelihood = -11957.53896914803     0.9940442173175335
46: -- log likelihood = -11760.92746306771     0.9942236937187983
47: -- log likelihood = -11562.16851856701     0.9944504007519117
48: -- log likelihood = -11376.841896540847    0.994539812052784
49: -- log likelihood = -11186.517485959335    0.9947290614801135
50: -- log likelihood = -11008.798639967052    0.9948801995021891
51: -- log likelihood = -10837.31893879739     0.9949982760819357
52: -- log likelihood = -10671.29323810977    0.9951163826616823
53: -- log likelihood = -10511.745550448913    0.9952863829265173

```

Gambar 5.10 Hasil dari *loglikelihood* Pada Setiap Iterasi

Setelah berhasil melakukan perhitungan *loglikelihood* pada setiap iterasi, maka akan dihasilkan model yang dapat digunakan pada penelitian ini, dengan nama file en-chunker.bin dengan waktu eksekusi 190.079 detik seperti yang tertera pada gambar berikut :

```
Writing chunker model ... done (2.010s)
Wrote chunker model to
path: E:\apache-opennlp-1.7.2\bin\en-chunker.bin
Execution time: 190.079 seconds
```

**Gambar 5.11 Model yang Dihasilkan dari Proses *Training***

Setelah berhasil menghasilkan model yang digunakan pada penelitian, maka langkah selanjutnya adalah menguji validasi terhadap model tersebut dengan menggunakan parameter `ChunkerEvaluator` yang telah disediakan oleh OpenNLP. Untuk melakukan uji validasi ini, maka data yang digunakan adalah data *test*. Data *test* berisi *chunks* dari korpus CoNLL2000 juga, namun diambil dari bagian yang berbeda dari data yang digunakan untuk data *training* dengan ekstensi file berupa *.eval*. Parameter yang dimasukkan untuk melakukan uji validasi tersebut dapat dilihat pada gambar berikut :

```
E:\apache-opennlp-1.7.2\bin>opennlp ChunkerEvaluator -model en-chunker.bin -data
testdatachunk.eval -encoding UTF-8
```

**Gambar 5.12 Parameter yang Dimasukkan Untuk Melakukan Validasi**

Maka, OpenNLP akan melakukan *load* terhadap model tersebut dan menghitung Precision, Recall dan F-Measure dari model tersebut. dari 23852 entitas yang terdapat pada data, OpenNLP berhasil mengevaluasi 2012 sampel dari data dengan rata-rata pengiriman data akan diperjelas pada gambar di bawah ini :

```
Loading Chunker model ... done (0.772s)
current: 214.4 sent/s avg: 214.4 sent/s total: 215 sent
current: 377.4 sent/s avg: 295.6 sent/s total: 592 sent
current: 427.4 sent/s avg: 339.3 sent/s total: 1019 sent
current: 450.5 sent/s avg: 367.0 sent/s total: 1469 sent
current: 459.5 sent/s avg: 385.6 sent/s total: 1929 sent
```

**Gambar 5.13 Rata-Rata Evaluasi Entitas Sampel dari Model**

Dari hasil sampel yang berhasil dievaluasi tersebut, terdapat jumlah data yang diprediksi benar adalah valid sejumlah 21995 entitas yang lebih jelasnya dapat dilihat pada gambar berikut :

```

Evaluated 2012 samples with 23852 entities; found: 23760 entities; correct: 2199
5.
TOTAL: precision: 92.58%; recall: 92.22%; F1: 92.40%.
PP: precision: 94.21%; recall: 97.69%; F1: 95.92%. [target: 481
1; tp: 4700; fp: 289]
NP: precision: 93.22%; recall: 93.21%; F1: 93.21%. [target: 124
22; tp: 11578; fp: 8421]
UP: precision: 93.06%; recall: 92.94%; F1: 93.00%. [target: 465
8; tp: 4329; fp: 3231]
ADJP: precision: 80.79%; recall: 77.71%; F1: 79.22%. [target: 866
; tp: 673; fp: 460]
SBAR: precision: 87.84%; recall: 66.17%; F1: 75.48%. [target: 535
; tp: 354; fp: 49]
ADJP: precision: 79.67%; recall: 67.12%; F1: 72.86%. [target: 438
; tp: 294; fp: 75]
PRI: precision: 75.00%; recall: 59.43%; F1: 66.32%. [target: 106
; tp: 63; fp: 21]
INTJ: precision: 50.00%; recall: 50.00%; F1: 50.00%. [target: 2
; tp: 1; fp: 1]
CONJP: precision: 57.14%; recall: 44.44%; F1: 50.00%. [target: 9
; tp: 4; fp: 3]
LST: precision: 0.00%; recall: 0.00%; F1: 0.00%. [target: 5
; tp: 0; fp: 1]
Execution time: 6.370 seconds

```

**Gambar 5.14 Hasil Validasi Model Secara Keseluruhan Oleh OpenNLP**

Dapat dilihat pada gambar tersebut, hasil Precision, Recall dan F-measure secara keseluruhan adalah :

- Precision : 92,50%
- Recall : 92.22%
- F-Measure : 92.40%

#### **5.4.4. Implementasi Sorting Data**

Setelah dilakukan validasi terhadap model, maka langkah selanjutnya yang dapat dilakukan adalah melakukan *sorting* atau mengurutkan hasil dari *chunking* yang telah dilakukan oleh model tersebut. Proses *sorting* dilakukan dengan tujuan untuk memudahkan analisis dari *chunk* yang dihasilkan. Hasil dari *chunk* tersebut diurutkan dari yang paling banyak muncul hingga ke yang paling sedikit.

Proses *sorting* data dilakukan dengan metode HashMap dan juga List yang merupakan *class* yang digunakan untuk menyimpan data dalam model Array pada bahasa pemrograman Java. Hasil dari *sorting* pada data hasil dari *chunking* ini lalu akan disimpan ke dalam sebuah file dalam format .csv. Agar dapat menyimpan file tersebut ke dalam bentuk .csv, maka diperlukan bantuan *library* lain. Dalam hal ini, *library* yang digunakan adalah Open CSV. Open CSV merupakan *library*

yang dibuat oleh SourceForge, yang bersifat *Open Source* dan dapat digunakan secara bebas [26].

Untuk melakukan *sorting* ini, kode yang digunakan secara detail akan dijelaskan pada uraian berikut.

```
HashMap<String, Integer> map = new HashMap<>();
int count, start, end = 0;
List<String> tampung = new ArrayList<String>();
List<String> string = new ArrayList<String>();
final Map<String, Integer> counter = new
HashMap<String, Integer>();
```

#### **Kode 5.17 Inisiasi *HashMap* dan *ArrayList* untuk Menampung Kalimat**

Kode diatas menggambarkan proses inisiasi pembuatan *HashMap* baru yang digunakan untuk proses *sorting ini*. setelah itu, dibuat inisiasi terhadap *ArrayList* yang berfungsi untuk menampung kalimat dari hasil *chunking* pada tahap sebelumnya. Hasil *chunking* tersebut yang lalu akan di *sorting* dari kalimat yang paling banyak muncul hingga kalimat yang paling sedikit. Setelah itu, proses *sorting* akan dilakukan dengan kode berikut.

```
if (map.containsKey(kalimat)) {
map.put(kalimat, map.get(kalimat) + 1);
} else {
map.put(kalimat, 1);
}
tampung.add(kalimat.toLowerCase().trim());
string.add("");
}
tampung.removeAll(string);
}
```

#### **Kode 5.18 *Hashmap* Untuk Melakukan *Sorting***

Kode diatas digunakan untuk menyimpan seluruh hasil *chunking* yang telah berhasil dilakukan pada tahap sebelumnya kedalam sebuah *HashMap*. Setiap kalimat yang berhasil dilakukan *chunking* tersebut, maka *HashMap* tersebut akan menyimpannya dengan sebuah nomor, untuk menandai kalimat tersebut. Kalimat tersebut lalu diubah ke bentuk huruf kecil agar

dapat dilakukan *sorting* secara akurat. Karena, jika tidak diubah ke huruf kecil, sistem akan menilai kalimat tersebut berbeda dan akan diberikan nomor yang berbeda. Padahal, kalimat tersebut memiliki maksud yang sama. Oleh karena itu, semua kalimat diubah menjadi huruf kecil. Langkah selanjutnya yang dilakukan dapat dilihat pada kode berikut :

```
for (String str : tampung) {
    counter.put(str, 1 +
        (counter.containsKey(str) ? counter.get(str) : 0));
}
```

#### **Kode 5.19 Menghitung Nomor yang Telah Dibuat *HashMap***

Setelah *HashMap* memberikan nomor pada setiap kalimat tersebut. Nomor tersebut akan semakin bertambah jika kalimat yang ditemukan pada setiap masukan hasil *chunking* tersebut memiliki kesamaan, sehingga nantinya dapat dilihat kalimat mana yang paling banyak muncul dari hasil *chunking* tersebut. Setelah menghitung nomor yang diberikan pada setiap kalimat, selanjutnya adalah menghitung frekuensi dari nomor tersebut. Kode untuk melakukan hal tersebut dapat dilihat sebagai berikut :

```
List<String> list = new ArrayList<String>(counter.keySet());
    Collections.sort(list, new Comparator<String>() {
        @Override
        public int compare(String x, String y) {
return counter.get(y) - counter.get(x);
        }
    });
    for (String str : list) {
        int frequency = counter.get(str);
        System.out.println(str + " : " +
frequency);
        String[] kata2 = {str + " : " +
frequency};
```

#### **Kode 5.20 Melakukan *Sorting* Terhadap Data**

Setelah menghitung frekuensi dari tiap kalimat yang muncul pada *chunk* yang dihasilkan, maka hal yang selanjutnya yang

dapat dilakukan adalah mengurutkan frekuensi tersebut mulai dari frekuensi terbesar ke frekuensi terkecil dari tiap kalimat. Untuk melakukan hal tersebut, kode yang dibuat adalah sebagai berikut :

```
for (String str : list) {
    int frequency = counter.get(str);
    System.out.println(str + " : " + frequency);
    String[] kata2 = {str + " : " + frequency};
```

#### **Kode 5.21 Mengurutkan Kalimat**

Setelah kalimat-kalimat tersebut berhasil diurutkan berdasarkan frekuensi munculnya kalimat tersebut pada data, mulai dari yang paling sering muncul hingga yang paling jarang, maka langkah selanjutnya yang dapat dilakukan adalah dengan menyimpan hasil *sorting* kalimat tersebut ke dalam sebuah file baru dengan format .csv. Hal ini dilakukan agar mempermudah proses selanjutnya yang akan dilakukan dalam penelitian ini. Untuk menyimpan hasil kalimat tersebut dalam sebuah file baru, maka dilakukan kode berikut ini :

```
String pathFile = "E:\\text02.csv";
    if(writeCSV(kata2, pathFile)){
        System.out.println();}
else {
System.out.println("Cannot write the file");}
```

#### **Kode 5.22 Menyimpan Hasil *Sorting* ke File Baru**

Kode tersebut akan menghasilkan sebuah file baru dimana kalimat yang berhasil di *sorting* akan disimpan. Proses ini diharapkan akan mempermudah analisis terhadap *chunking* yang dihasilkan oleh model dan tahapan-tahapan selanjutnya dalam penelitian ini.

Prototipe ini telah melakukan 3 kali percobaan terhadap data yang diolah, dengan mengubah parameter *chunks* yang diambil, yaitu 3, 4 dan 5 *chunks*. Hasil dari percobaan tersebut dapat dilihat sebagai berikut :

#### 5.4.4.1. Hasil Percobaan dengan 3 *Chunks* pada Aplikasi *Plant vs Zombies*

Percobaan pertama yang dilakukan pada prototipe ini adalah dengan membuat prototipe mengambil 3 *chunks* pada setiap kalimat yang terdapat pada ulasan pelanggan. Hasil ekstraksi ekspresi kata kerja pada 3 *chunks* dapat dilihat pada tabel berikut :

**Tabel 5.5 Hasil Percobaan dengan 3 *Chunks***

Hasil <i>Chunks</i>	Frekuensi Muncul
i love this game	18
i love it	13
i love the game	11
i love it,	9
i don't like	6
i think it	5
i play it	5
i love this game	5
the game is awesome	5
this is the best game	5
i lost all	4
this game is awesome	4
i think this game	4
i get it	4
i loved it,	4
it came out	4
this game is great,	3
this game is amazing	3
i installed it	3
this is the second time	3
i lost my progress	3
what happened to	3
i logged in	3
i will give you	3

i think this	3
the game is good	3
you make it	3
i don't know what	3
it would be better	3
i loved it	3
i worked hard	3
it would make the game	3
i find it	3
i play this game	3
i opened it	3
i updated it	2
i hate this game	2
you should add multiplayer	2
i have to play it	2
i have some problem	2
i think you	2
a pay to win game.	2
it shows me	2
i have lost all my progress	2
i will rate it	2
i lose my account	2
the game is great.,	2
you progress through	2
i pay for	2
you were able	2
i update it	2
to level up plants.,	2
i got it	2
i open it	2
i cant open it	2
i love it, i	2
we can buy with	2
i have to start all	2
it takes too long	2

it will make the game	2
i hate it	2
i like it	2
i came back	2
some levels are impossible	2
i tried to	2
you know i	2
all over again	2
this is so	2
i stuck at	2
i uninstalled it	2
it makes me	2
this is my favourite game	2
is it that	2
i think it's	2
i changed my phone	2
i want to play it	2
i am speechless	2
i have a problem	2
"you need in	2
i dont know why	2
to get it back.	2
i hope that	2
i like this game,	2
i am unable	2
you have to pay for	2
i loved this game	2
i spent a lot	2
it is way	2
it is a good game,	2
i like this game	2
i lost my profile	2
know how it	2

Berdasarkan hasil ekstraksi ekspresi kata kerja yang dapat dilihat pada tabel diatas, model telah dapat bekerja dengan baik,

namun pada kenyataannya ulasan pelanggan yang dijadikan sebagai data pada percobaan kali ini juga memiliki kecatatan secara tata bahasa, seperti penggunaan tanda baca dan *grammar* yang tidak tepat sehingga menyulitkan model untuk mengekstrak frase ekspresi kata kerja dengan tepat. Terdapat beberapa jenis kalimat dari hasil ekstraksi tersebut yang di luar kontrol dari penelitian ini diantaranya kalimat seperti contoh pada tabel berikut :

**Tabel 5.6 Kesalahan pada Hasil Ekstraksi Ekspresi Kata Kerja yang Dihasilkan**

Kalimat yang Dihasilkan	Kesalahan yang Terjadi
what happened to	Kesalahan tata bahasa
this is the second time	Kesalahan tata bahasa
i will give you	Kesalahan tata bahasa
this game is great,	Kesalahan tata bahasa dan tanda baca
to level up plants.,	Kesalahan tata bahasa dan tanda baca

#### 5.4.4.2. Hasil Percobaan dengan 4 *Chunks* pada Aplikasi Plant Vs Zombies

Percobaan kedua yang dilakukan pada prototipe ini adalah dengan membuat prototipe mengambil 4 *chunks* pada setiap kalimat yang terdapat pada ulasan pelanggan. Hasil ekstraksi ekspresi kata kerja pada 4 *chunks* dapat dilihat pada tabel berikut :

**Tabel 5.7 Hasil Percobaan dengan 4 *Chunks***

Hasil <i>Chunks</i>	Frekuensi Muncul
i love it so	10
i lost all of	8
i love this game it	7

i opened it again,	7
i worked hard for	6
it is one of	5
i have to start all over	5
i play this game for	5
the game is awesome but i	5
i get it back?,	5
this is the second time i	4
all over again from	3
i find it hard	3
i am sure that it	3
i installed it again	2
again, but ea was kind enough to compensate me	2
what to say, maybe just....	2
i lost 1500 diamonds 48million	2
the stage.. and i have trophies. i	2
crashes. no ""submit crash report"" window, asome, pls pls	2
it is way to	2
the entire game and finished all the world's. i	2
my new device, please fix., even my plant level	2
the levels and unlocked most of	2
the travel log. please fix this! in	2
2 års progress..., i	2
had 800 hundred gems and none of	1
an odd reason is takes like	1
have a ridiculous idea but im gonna share it	1
perfect. keep it better	1
you take a premium plant like	1
this is that i cant	1
you made, locks up	1
it was a very nice game. more challenges., everything	1
had to re-install the game, only to have to start again	1

pls help, love it hard	1
suits and play as them et, its cool i	1
the level. that's all. thank you for	1
ancient egypt is difficult than	1
i finish since the last update. samsung galaxy s6,	1
the only problem is no multiplayer, it's just	1
you win the frost pea and	1
this game is very addictive.i like	1
played!!! and this game is cool, and i	1
a refund. thanks. update: it's been a few days	1
already and yet	
to even load.sinclair zx spectrum games run	1
faster than	
gems and time just to buy those plants.	1
i couldn't log in to	1
i know that i must complete quests	1
to experience and play pvz since when	1
that could be unlocked using gems, not just	1
money all the time.	
another thing is the game keeps on	1
me think that i am very power	1
i couldn't access my previous account. what	1
crashes on popcap loading screen. please,	1
it is fun., only one problem new players	1
i bought my new s8 and installed the game.	1
it needs a pc release as	1
it is really awful to win again and again	1
ini membuat saya jengkel, great game, very	1
amazing game!	
bad. anything can do to reinstate my earlier	1
stage? if	
these while and games are linked to my google	1
account.	
you start playing it shuts down pls	1
some levels are ridiculously hard	1

i can get free gems, costume and coins *sigh. whats	1
unlock all achievements when i	1
no new world is coming., super, next update please, add the viking world...,	1
i entered my account get me	1
i put 4 stars. why? i	1
this game has lag a little, i	1
u please don't decrease more than 100mb++? my space is not enough	1
a bong choy is funny. go punch., adictivo. no puedo	1
weeks ago. yesterday i logged in	1
this games is pretty slow and hard	1
there would be more games like	1
some levels are just impossible	1
that as soon as you	1
the zombie eats it, i've noticed that when i hate it it	1
that need real money, with	1
the levels are already played...!!!, crazy dave is so crazy...,	1
the fire peashooter can be found in the frostbite	1
it i'm just saying anyway thank you guys for everything to find a way to play the game	1
need plant vs zombie 3,	1
to uninstalling it. please fix this so that i	1
it ends too soon b4	1
the zombies disappeared in my screen lucky	1
to play my old account again. how	1
more levels to play, cool game, it	1
it, its so good but why	1
dark ages lv 14 impossible due	1

Hasil ekstraksi ekspresi kata kerja pada 4 *chunks* ini, seperti halnya pada ekstraksi ekspresi kata kerja pada 3 *chunks*,

ternyata juga masih memiliki kesalahan yang sama. Karena tetap menggunakan data ulasan pelanggan yang sama, kesalahan tersebut akan teteap terjadi dan tidak dapat dihindari. Contoh kalimat yang mengalami permasalahan dan tidak dapat diatasi pada penelitian ini dapat dilihat pada tabel berikut :

**Tabel 5.8 Kesalahan pada Kalimat yang Dihasilkan**

Kalimat yang Dihasilkan	Kesalahan yang Terjadi
the level. that's all. thank you for	Kesalahan tata bahasa
i am sure that it	Kesalahan tata bahasa
this is that i cant	Kesalahan tata bahasa
the levels are already played...!!!, crazy dave is so crazy...,	Kesalahan tata bahasa dan tanda baca
what to say, maybe just....	Kesalahan tata bahasa dan tanda baca

#### 5.4.4.3. Hasil Percobaan dengan 5 *Chunks* pada Aplikasi Plant vs Zombies

Percobaan ketiga yang dilakukan pada prototipe ini adalah dengan membuat prototipe mengambil 5 *chunks* pada setiap kalimat yang terdapat pada ulasan pelanggan. Hasil ekstraksi ekspresi kata kerja pada 5 *chunks* dapat dilihat pada tabel berikut :

**Tabel 5.9 Sampel Hasil Percobaan dengan 5 *Chunks***

Hasil <i>Chunks</i>	Frekuensi Muncul
this game is amazing. i love it	7
only reward is 1000 coins if i'm not getting wrong	5
i clear the data and i came back	4
happy plants, won't work on me	4

my second gmail account same result.it had to do something with recent updates.	4
i emailed you's no reply back	4
superb, fun way to waste some time, the best game, 100%	3
i was a big fan of the original	3
this game was good but the map changed and everything	3
i love this game because it is a old game	2
the sun bean's nerf, please either lower the level's required sun or bring the old sun bean	2
one and kee, i love the game but when	2
surfer zomble able to smash a plant and throwing another surf board while dying.	2
this game but life is too short for the hassle	2
plants and changed phones and my progress was not saved. so i	2
you realise it's way harder. after	2
the ef. gets annoying now. and another thing is the game	2
i open it so good., my profile	2
the game has too many glitch but very good gameplay, excellent,	2
are some premium plants like dandelion need to be bought with	2
love this game but right now i cant install it	2
the game but creator can i choose my own plants to use in the 11 worlds	2
i lose all of my data, kudos	1
u give them free.i will give it	1
i pay for and it won't show	1
i liked how the powers are, i	1
addictive. do yourself a favour, don't	1
there are no new worlds. new worlds would be awesome or just pvz3 with less	1

your stupid game deleted my account permanent.this happen 2 times i just waste my time	1
startup.., new upgrades and locked plants are purchased through gems not the real money	1
coins and gems... recently i updated	1
my own team i can select which one i want (plant).i never	1
i lost my progress so i had to start all	1
starting over., good graphics bestest game ever., it's	1
i ned to go back to the 1st level,	1
i found no solution and now i	1
my data is lost, i couldnt recover or find my account,	1
i wish there are more gems to be given everyday.	1
i loved it but please make a plant that we	1
i have problem it is written resources	1
you please help me?? i'm really frustrated here. my hard work	1
i have some problem on this game	1
plantvlevels is rubbish, no replays for a long time	1
quests though. there are few plants that you	1
i reinstalled the game. i had given 5 stars	1
it's good game but level 23 (the matching one) of	1
being offline same goes to the travel logs, nice gameplay, nice, fun for	1
i love it, i have changed my handset. i	1
it it's ok it's your choice.??,	1
sometime, wont let me update garden but very good game kids	1
i opened it again, i	1
a limited 14 plant and was almost wining and place a frozen lettuce	11

use another phone since my tab is not working anymore.. kindly check my acc.	1
the graphics are the best in the university!!!!!!!!!!!!, awesome epic	1
stops until i win., help. mine won't open. it	1
was new mode come??. too much ads, really nice game, file	1
i can't play anymore the app crashes, over	1
it is not better game graphics is very cool, i	1
the loading screen pls fix or help me :), best game i	1
all data and saves were gone. was close to 100%	1
i do to get my progress back?, its a great game, good graphics but its there	1
which are almost impossible to win unless we buy many leaves	1
the pinata is not satisfying..i actually have a ridiculous idea	1
i came back at the start!!!!!!!!??	1
love this game, awesome game, please bring back coin quests, with	1
i restored my mobile and i installed this app again and guess	1
zombies r funny, i have downloaded the game once	1
this? many people complain about this too, this game	1
i am at level 34 on	1
people to play this game fix any single thing that	1
this game is the lagging. it always	1
buying upgrades and not having the real advantage that i	1
users pay to win. deleting this game., love	1
you have to spend money on plants	1

the paid plants could be paid in a gem or make another world	1
we should be given some good premium plants to play for free	1
iam going to uninstall it ., kya game h	1
the developers adjust the difficulty because it is so freaking	1
ever played. keep up and the good work and god bless., cool,	1
this game., please help... all of my data	1
i wonder why not all plants are free	1
that i purchased with real money, well	1
it will always stay in my memories as	1
lost it again after another re-install	1
to open this application, crashes too often lately, best mobile game. power plants is great.	1
any support to help me ..., this is awesome,	1
coins.going to uninstall this game, good game :), by far one	1
i like pvp, loveeeeeed it, no problem	1
i will give 5* when it	1
there are several achievements like complete tutorial,	1
seed pockets. but now its gone :(( still a great game	1
all the plants have level caps and different too.ive seen the video	

Seperti halnya pada hasil ekstraksi ekspresi kata kerja dengan menggunakan 3 dan 4 *chunks*, permasalahan yang sama juga masih terjadi pada kalimat yang dihasilkan dengan 5 *chunks*. Permasalahan kalimat yang dihasilkan tersebut dapat dilihat pada tabel berikut :

Tabel 5.10 Kesalahan pada Kalimat yang Dihasilkan

Kalimat yang Dihasilkan	Kesalahan yang Terjadi
plants and changed phones and my progress was not saved. so i	Kesalahan tata bahasa
plants and changed phones and my progress was not saved. so i	Kesalahan tata bahasa
i liked how the powers are, i	Kesalahan tata bahasa
there are several achievements like complete tutorial,	Kesalahan tata bahasa dan tanda baca
it it's ok it's your choice.??,	Kesalahan tata bahasa dan tanda baca

#### 5.4.4.4. Hasil Percobaan dengan 3 *Chunks* pada Aplikasi Instagram

Percobaan keempat yang dilakukan pada prototipe ini adalah dengan membuat prototipe mengambil 3 *chunks* pada setiap kalimat yang terdapat pada ulasan pelanggan di dalam aplikasi Instagram. Hasil ekstraksi ekspresi kata kerja pada 3 *chunks* dapat dilihat pada tabel berikut :

Tabel 5.11 Hasil Percobaan dengan 3 *Chunks* pada Instagram

Hasil <i>Chunks</i>	Frekuensi Muncul
i love it	45
i love instagram	21
i don't like	13
it won't let me	12
i think it	11
i think it's	11
i love this app	9
i loved it	8

it deleted my account	7
i love it, i	7
i love this app,	6
i use it	6
i like it	6
there was a problem	6
i am unable	5
i love instagram,	5
i love it.,	5
it would be nice	5
it makes me	4
it kicked me	4
i think instagram	4
i love you	4
it is easy	4
this app is really	4
me log in	4
it wont let me	4
there is no way	3
i didn't do anything	3
it is good,	3
it is good	3
i like how	3
i love insta	3
my account got deleted for	3
instagram is great	3
you deleted my account	3
me get it	3
it disabled my account	3
i love it!,	3
to keep in touch	3
i love you,	3
i can't log into	3
i love how	3
it keeps on	3

i love ig,	3
i can't get on	3
i hate it	3
you can communicate with	3
i put in	3
i hate instagram	3
i have to uninstall it	3
i woke up	3
instagram is a great app	3
it is fun	3
i am not	3
i cannot turn off	2
i try to log in	2
i love this	2
it was easier	2
i can't log back	2
social things .. i	2
the app is great	2
me log in.	2
to log in to	2
i hope u	2
i feel like	2
you can take pics	2
i can get on	2
i got it	2
it takes too long	2
i don't understand why	2
i would give it	2
i like it,	2
they told me	2
i got disabled	2
i hate that it	2
i can't get back	2
i'm missing out	2
you can talk to	2

i hope this	2
a awesome way to talk to	2
it would be better	2
i get to	2
it keeps telling me	2
i know what	2
i hope it	2
i can't see who	2
it is awesome,	2
me figure out	2
there isn't a way	2
i dont know why	2
i can see what	2

Berdasarkan hasil ekstraksi ekspresi kata kerja yang dapat dilihat pada tabel diatas, model telah dapat bekerja dengan baik seperti pada percobaan yang telah dilakukan pada aplikasi sebelumnya, yaitu *Plant vs Zombies*. Namun pada kenyataannya ulasan pelanggan yang dijadikan sebagai data pada percobaan kali ini juga memiliki kecatatan secara tata bahasa, seperti penggunaan tanda baca dan *grammar* yang tidak tepat sehingga menyulitkan model untuk mengekstrak frase ekspresi kata kerja dengan tepat. Terdapat beberapa jenis kalimat dari hasil ekstraksi tersebut yang di luar kontrol dari penelitian ini diantaranya kalimat seperti contoh pada tabel berikut :

**Tabel 5.12 Kesalahan pada Kalimat yang Dihasilkan**

Kalimat yang Dihasilkan	Kesalahan yang Terjadi
it wont let me	Kesalahan tata bahasa
me log in	Kesalahan tata bahasa
me get it	Kesalahan tata bahasa
i love ig,	Kesalahan tata bahasa dan tanda baca

i love you,	Kesalahan tata bahasa dan tanda baca
-------------	--------------------------------------

#### 5.4.4.5. Hasil Percobaan dengan 4 *Chunks* pada Aplikasi Instagram

Percobaan keempat yang dilakukan pada prototipe ini adalah dengan membuat prototipe mengambil 4 *chunks* pada setiap kalimat yang terdapat pada ulasan pelanggan di dalam aplikasi Instagram. Hasil ekstraksi ekspresi kata kerja pada 4 *chunks* dapat dilihat pada tabel berikut :

**Tabel 5.13 Hasil Percobaan dengan 4 *Chunks* pada Instagram**

Hasil <i>Chunks</i>	Frekuensi Muncul
i didn't post anything	32
i enjoy seeing my friends and beautiful people	17
it helps me keep in touch	12
me get my account back	12
i love it so	11
me log in to	11
it disabled my account for	11
i love it, it's	11
jesus christ!!, cant login even after	10
there was a problem with	10
i am simply amazed at the excellent technology	10
crashing is observed arrow sign not performing any action, most interesting aap	9
instagram is an excellent app, my fav apps	9
me suggestions for that account	9
it would btr, ...love instagram. i?expressing thoughts!, they	8
the next myspace. pages selling shout outs should be against community guidelines "	8

i am locked out of	8
the option to have a black/dark interface. also	7
posting pictures!, since the new update.	7
there should be features of	6
edited my photo using adobe lightroom, and everytime i	6
this bring bad thing for	6
all my friends are on it!, lul,	5
i had to make another account. ????????????????, good, the app	5
u have to improve it more, its a really good app	5
blocked temporarily ..."", it's an amazing way to keep in	5
it's good ????, fun!, amazing app, can someone please, bring switch camera on	5
love this app. very easy to use i	4
become family ????, amazing, nothing	4
to delete the account., i love the posts	4
posts, awesome, tolong buat supaya pengguna android jellybean juga bisa menggunakan fitur live	4
no doubt, friendly user, awesome way to make new friends and not just locally but all around	4
it is one of	3
my account got deleted for no reason plz	3
it does not work properly and now i	3
will give 5 stars ., good, favorite site!, malah makin lelet ??, since	3
it would be a lot better	3
you, easy and fun to use, omg, good ., there	2
i have violated the terms, and i	2
i bet you know you	2
me make a new instagram please fix this this app	2
guys just adding new feature like	2

it is very good, i	2
my account twice saying ""it looks like you	2
it really helps me show everyone	2
i love instagram it	2
i don't understand why y'all	2
the best social media application, nice, cant change push notifications and videos don't load.,	2
i hahaha nice, fascinating. cant post full- length pics. the frame	2
freezes with my lg g6, love it	2
i like it this app bcoz	2
must it, easy to use. i	1
anyone i've been blocked... 2 days	1
editing multiple pictures to share. reinstalled the app but still	1
it is that it makes me	1
friends are up to	1
we put in all the following	1
friends and to get new friends by	1
your friends, instagram is asking me for \	1
this gets fixed, can you	1
i hope that you never	1
the last update, u cn see a glimpse of	1
it does, ????????, u guys deleted my account! and i	1
its not working! ??, got banned from commenting for 23 hours	1
no reason people worked hard for	1
what to do with., grest lighting	1
it is amazing nothing i	1
my favorite accounts have ""12 new posts"", "6 new posts	1
i love insta...., really easy	1
i know from my past	1
the hell am i suppose to give it	1

i have not done anything wrong. so i	1
instagram fir spam you just	1
an option to choose between most important and most recent for	1
an option to send individually. problem in	1
you follow 600. have to force stop all the time	1
gave insta a try and it	1
your friends, great app to connect with, download pic	1
this feature becomes more and more useless	1
something that i have to deal with and it's annoying	1
a tablet and i cant get on anymore plz help, portrait photo	1
this is my favourite app?, lovely	1
you can't turn off and recently	1
new version videos not played, new update is not letting me	1
you improve the picture quality.	1
ur bio?, very nice, cool app!, y'all had delete my account by	1
facebook. please help instagram team, best social app, the new one	1
my account, delete people pages??. love it., its the best,	1
i love it!, awesome,	1
you can send emails non lime other	1
i'm begging you, it's	1

Hasil ekstraksi ekspresi kata kerja pada 4 *chunks* ini, seperti halnya pada ekstraksi ekspresi kata kerja pada 3 *chunks*, ternyata juga masih memiliki kesalahan yang sama. Karena tetap menggunakan data ulasan pelanggan yang sama, kesalahan tersebut akan teteap terjadi dan tidak dapat dihindari. Contoh kalimat yang mengalami permasalahan dan tidak dapat diatasi pada penelitian ini dapat dilihat pada tabel berikut :

**Tabel 5.14 Kesalahan pada Kalimat yang Dihasilkan**

Kalimat yang Dihasilkan	Kesalahan yang Terjadi
friends are up to	Kesalahan tata bahasa
i'm begging you, it's	Kesalahan tata bahasa
you can't turn off and recently	Kesalahan tata bahasa
all my friends are on it!, lul,	Kesalahan tata bahasa dan tanda baca
it would btr, ...love instagram. i?expressing thoughts!, they	Kesalahan tata bahasa dan tanda baca

#### 5.4.4.6. Hasil Percobaan dengan 5 *Chunks* pada Aplikasi Instagram

Percobaan ketiga yang dilakukan pada prototipe ini adalah dengan membuat prototipe mengambil 5 *chunks* pada setiap kalimat yang terdapat pada ulasan pelanggan. Hasil ekstraksi ekspresi kata kerja pada 5 *chunks* dapat dilihat pada tabel berikut :

**Tabel 5.15 Hasil Percobaan dengan 5 *Chunks* pada Instagram**

Hasil <i>Chunks</i>	Frekuensi Muncul
nice application to share pics with	20
share photos to the app	16
i can take a nice hd video from	8
i can text my friends on	7
i tried to turn off the notifications	7
please make on and off option	7
the app has been deleting a lot of	6
i tried recovering but it wouldn't let me	5
that lets you post videos and picture	5

please add an option to check the offline/online status of	4
to fool the customers badly..uninstalling right now...!!	4
my account was just disabled along with it won't even upload! just glitches and shuts	3
it works ok, its the best	3
instagram was nice and not rude, great app	3
ig but lately it's been freezing, blacking out	3
i got a problem after updated in	3
i love instagram. u can c what	3
i turned off push	2
i want u to do is	2
hello instagram is cool but plz add photo	2
this app is very good. i liked it	2
u can better advance a way for	2
i can't log back into	2
story, please fix it. device : xiaomi redmi	2
everyone, either...i'll go to the search page	2
did y'all really have to copy snapchat?, ?????	2
i hope it doesn't	2
the dam bleeding app keeps crashing on me!!!,	2
is the timeline not showing most recent posts ?? turning into	2
i love instagram, but it's telling me	2
they are beind stuck when im	2
i loved it a lot.	2
5 stars bc its a good app but stop deleting accounts pleeeeee, love it	2
i see everyone is having the same problem	2
your best photos and love the filters, my favourite,	2
the app does not need an update	2
the videos can't be favst forwarded.,	2
people account!!!!, it's an amazing app	2

to load even with strong signals., stop deleting account plz...??, nice app, the only thing	2
freeze which is annoying., it	2
instagram and hasnt posted not 1 pic and she	2
a great way to interact with people,	2
to solve it. have change image quality on my app	2
very nice app, keeps crashing on android 7.0 nougat, didn't	1
supb, on windows phone not work .....,	1
it is easy to use and useful, it	1
this, instagram is an amazing app with	1
i love it, it is very good,	1
a great way to share your pictures i	1
a great way to see my friends, the best app ever	1
to make sharing stories easier and be able	1
keeping in touch and seeing how ppl	1
request it's been doing it for	1
my phone is off really	1
annoying having to deal with some of	1
i shouldn't need. i'm missing out	1
little posted from like	1
it said 'put in	1
posting your best moments!, mine doesn't	1
it is happening to some	1
you can take pics send them	1
account @darkskiin.pwiincess was disabled please let me	1
know by instagram, it's fun, instagram disabled	1
i can't even look at my old posts.	1
random ass photo order is sirously the dumbest feature ever!! and get rid	1
to me?, good, awesome app, share your moment with	1

sucking. it's super annoying., it's a good app	1
i love it, awesome app,	1
they shoot.. happy to be a part	1
it it my life changed, stop deleting accounts	1
i love this app, put more photo	1
the last updated deleted my account without	1
the other woman be attentive. the world is a bad place	1
i love this child safe way of	1
liked it, nice, really like	1
i forgot it. but it	1
i was a fandon account. do not get instagram	1
i loved it ??, it's	1
you stuff instagram., lots of	1
i do give kudos for	1
a good way to share your day, whats up	1
dms and try to go back it	1
the pic... wont allow a pic to post as is. it crops	1
your follows to get in contact	1
this new update i have to have a phone number to access my account	1
it have tried restarting my phone, uninstalling it	1
hi to 2 people it	1
this app is very helpful for	1
some memes, useful to promote my work and myself, app doesn't let ne upload my location	1
i can't find her anywhere..., sering ngecrash pas	1

Seperti halnya pada hasil ekstraksi ekspresi kata kerja dengan menggunakan 3 dan 4 *chunks*, permasalahan yang sama juga masih terjadi pada kalimat yang dihasilkan dengan 5 *chunks*. Permasalahan kalimat yang dihasilkan tersebut dapat dilihat pada tabel berikut :

Tabel 5.16 Kesalahan pada Hasil *Chunking*

Kalimat yang Dihasilkan	Kesalahan yang Terjadi
little posted from like	Kesalahan tata bahasa
it said 'put in	Kesalahan tata bahasa
posting your best moments!, mine doesn't	Kesalahan tata bahasa
to load even with strong signals., stop deleting account plz...??, nice app, the only thing	Kesalahan tata bahasa dan tanda baca
a great way to interact with people,	Kesalahan tata bahasa dan tanda baca

### 5.5. Implementasi *Usability Testing*

*Usability Testing* pada penelitian ini dilakukan untuk menguji apakah sistem yang dikembangkan telah memuat informasi yang berguna terkait dengan ulasan pelanggan yang digunakan sebagai objek penelitian. *Usability Testing* dilakukan dengan cara melakukan survei terhadap responden. Responden pada survei penelitian ini merupakan pengguna dari aplikasi yang dijadikan objek penelitian, yaitu *Plant vs Zombies 2*, telah menggunakan aplikasi ini selama kurang lebih 1 bulan dan diutamakan merupakan mahasiswa.

Penyesuaian konsep utama *usability testing* dengan tujuan penelitian ini dilakukan untuk agar pernyataan yang terdapat dalam survei akan menggambarkan sistem dengan tepat dan sesuai dengan sasaran dilakunnya penelitian, yaitu untuk mengetahui apakah ekstrasi ekspresi kata kerja ini dapat dijadikan acuan informasi yang cukup bagi pengguna Google Play, sehingga pengguna tidak perlu membaca ulasan pelanggan hingga memakan waktu yang lama.<sup>2</sup>

Survei yang dilakukan berupa memberikan kuisioner terkait dengan informasi yang disajikan pada sistem yang ada. Pernyataan yang disajikan untuk responden dapat dilihat pada tabel berikut.

**Tabel 5.17** Kuisioner *Usability Testing*

Pernyataan	1	2	3	4	5
Saya merasa bahwa saya informasi mengenai ulasan pelanggan yang terdapat pada sistem ini berguna					
Saya berpikir sistem ini sangat rumit dan tidak perlu					
Saya berpikir bahwa sistem ini mudah digunakan					
Saya berpikir saya akan membutuhkan bantuan seseorang untuk dapat menggunakan sistem ini					
Saya merasa bahwa informasi yang terdapat dalam sistem ini memberikan pengetahuan yang lebih terkait dengan ulasan pelanggan					
Saya berpikir terlalu banyak inkonsistensi terkait dengan informasi yang ditampilkan dari sistem ini					
Saya merasa banyak orang akan memahami informasi yang dimuat dalam sistem ini secara efektif					
Saya merasa informasi yang dimuat dalam sistem ini sulit untuk dipahami					
Saya merasa sangat percaya diri dalam menggunakan sistem ini					
\Saya perlu belajar banyak hal sebelum dapat menggunakan sistem ini					

1. Apakah informasi mengenai ulasan pelanggan telah cukup menggambarkan aplikasi Plant vs Zombies 2? Jelaskan

---

---

---

2. Adakah informasi selain ulasan pelanggan yang menurut anda perlu ditambahkan agar dapat menggambarkan aplikasi Plant vs Zombies 2 lebih mudah?

---

---

---

3. Saran anda untuk perbaikan kedepannya?

---

---

---

Dalam survei ini, berhasil didapatkan 10 responden yang rata-rata telah menggunakan aplikasi ini selama lebih dari 2 bulan, sehingga sudah memenuhi syarat untuk menjadi responden survei ini. Responden akan mengisi pernyataan-pernyataan dalam survei ini dalam bentuk *hardcopy* yang telah dibagikan sebelumnya. Penelitian ini berhasil memperoleh 10 responden yang berasal dari mahasiswa di lingkungan Jurusan Sistem Informasi ITS. 10 responden tersebut telah menggunakan aplikasi Plant vs Zombies 2 selama lebih dari 2 bulan dan sangat familiar dengan lingkungan dari aplikasi. 10 responden tersebut juga memiliki kemampuan Bahasa Inggris cukup baik sehingga dapat memahami kalimat dengan Bahasa Inggris dengan cukup baik.

## **BAB VI**

### **HASIL DAN PEMBAHASAN**

Bagian ini akan menjelaskan hasil serta analisis terhadap hasil yang telah diperoleh dari proses implementasi yang telah dilakukan pada bab sebelumnya.

#### **6.1. Hasil dan Pembahasan Ekstraksi Data Ulasan Pelanggan**

Setelah dilakukan ekstraksi terhadap data ulasan pelanggan pada Aplikasi Google Play. Aplikasi yang digunakan pada penelitian ini merupakan game Plant vs Zombies 2. Ulasan pelanggan tersebut menggunakan Bahasa Inggris, karena model yang dikembangkan hanya mampu mendeteksi Bahasa Inggris dan belum mampu mendeteksi Bahasa Indonesia. Pada penelitian ini, aplikasi berjenis game dipilih karena ulasan pelanggan diperkirakan akan mengandung ekspresi kata kerja lebih banyak jika dibandingkan aplikasi berjenis sosial media, maupun aplikasi lainnya. Berikut ini merupakan sampel hasil dari ekstraksi data tersebut :

**Tabel 6.1 Sampel Data Ulasan Pelanggan pada Aplikasi Plant vs Zombies 2**

My plant vs zombie 2 app has stopped, how to fix it without clearing the data.

Everything in this update is awesome specially the new purchasable plants that costs a hundred gems but I just noticed that Sun Bean is not doing it's thing. I think he better get fixed. Also, can I have my Peashooter? It's been a long time without a basic Peashooter.

For some reason I'm not getting gem jam 3 or 4 but I have the update could you fix that

Great

---

Very pleased that Sun Bean is back like the old days. Homing Thistle still attacking air a lot. But with recent level up, it's attack is upgraded and it's not bad. Sadly Surfer Zombie and Balloon Zombie also upgraded. Surfer Zomble able to smash a plant and throwing another surf board while dying. While Balloon Zombie sometime able to re inflate the balloon after it's been exploded. Pinata is missing (the pinata party and seed pinata timer) when the seed pinata timer reaching zero, only to be available the next seed pinata travelog quest which can be the next 4 hours, it's very vexing.

---

Dapat dilihat pada data ulasan pelanggan tersebut, meskipun sorting yang dilakukan pada saat mengambil data berdasarkan *helpfulness* dari ulasan tersebut yang berarti dari kegunaan ulasan pelanggan tertinggi berdasarkan vote dari pengguna lain, tidak menjamin bahwa ulasan pelanggan tersebut memuat informasi yang berguna. Selain itu, masih terdapat banyak kesalahan yang dilakukan pengguna saat menulis ulasan, seperti kesalahan tata bahasa dalam Bahasa Inggris dan ejaan.

Banyak ulasan pelanggan yang masih hanya memberikan ulasan standar yang biasanya berisi “Great”, “Good” maupun ulasan-ulasan standar lain yang seharusnya memiliki tingkat *helpfulness* rendah, namun dikarenakan tetap mendapat vote yang tinggi dari pengguna lain, maka tetap masuk sebagai ulasan yang membantu. Ulasan tersebut tidak mendeskripsikan aplikasi sama sekali sehingga menyebabkan relevansi data yang digunakan dalam penelitian ini berkurang.

## **6.2. Hasil dan Pembahasan Praproses Data**

Pada bagian ini akan dijelaskan mengenai hasil dari praproses data yang telah dilakukan pada tahap implementasi sebelumnya. Hasil praproses data tersebut berupa hasil muatan data teks dan tokenisasi.

### **6.2.1. Muatan Data Teks**

Data ulasan pelanggan yang didapatkan dari hasil *scraping* pada data tersebut berjumlah sebanyak 4840 ulasan pelanggan dari

121 halaman yang berhasil dimuat. Ketika dilakukan pemuatan data teks tersebut, dilakukan pengecekan terhadap data yang kosong atau bernilai *NULL*. Data tersebut tidak akan dimasukkan ke dalam pemrosesan data berikutnya. Setelah dilakukan pengecekan, ternyata tidak ditemukan data bernilai *NULL* tersebut sehingga jumlah data total adalah 4840 ulasan.

### 6.2.2. Muatan Data Ulasan Pelanggan

Data ulasan pelanggan yang dimuat pada program didapat pada satu sumber, yaitu data teks yang dimuat sebelumnya yang telah dimasukkan ke dalam satu basis data. Dari 4830 ulasan pelanggan, data yang akan digunakan untuk diteliti lebih lanjut berjumlah 4000 ulasan pelanggan.

### 6.2.3. Hasil dan Pembahasan Segmentasi Kalimat

Segmentasi kalimat merupakan tahapan dimana ulasan pelanggan akan dipecah menjadi kalimat-kalimat yang sebelumnya tergabung dalam paragraf-pragraf dalam ulasan pelanggan tersebut. Segmentasi kalimat perlu dilakukan agar prototipe ekstraksi ekspresi kata kerja akan menghasilkan frase yang sesuai dari setiap kalimat-kalimat yang dihasilkan tersebut. Sampel hasil segmentasi kalimat dapat dilihat pada tabel berikut :

**Tabel 6.2 Sampel Hasil Segmentasi Kalimat**

My plant vs zombie 2 app has stopped, how to fix it without clearing the data.

Game is not allowing me to log onto after the update. Please fix

think the other premium plants can bought using real money or gems, and the others can only be bought with gems please read this!

It is a nice game

Model yang digunakan untuk melakukan segmentasi kalimat ini berasal dari *library* yang sama, yaitu OpenNLP yang telah dijelaskan pada tahapan implementasi. Berdasarkan segmentasi kalimat yang dihasilkan, dapat dilihat pada tabel diatas bahwa kalimat-kalimat yang dihasilkan telah sesuai berdasarkan data ulasan pelanggan Plant vs Zombies 2. Setiap kalimat yang dihasilkan tersebut lalu akan diproses ke tahapan selanjutnya yaitu tokenisasi.

#### 6.2.4. Hasil dan Pembahasan Tokenisasi

Tokenisasi merupakan proses yang dilakukan untuk merubah kalimat menjadi kata per kata yang disebut dengan token agar dapat dilakukan proses *labeling*. Sebagai contoh, Hasil tokenisasi pada satu kalimat ulasan dapat dilihat pada tabel berikut :

**Tabel 6.3 Sampel Hasil Tokenisasi**

Teks	Hasil Tokenisasi
Hasil Segmentasi Kalimat	More challenges., Everything in this update is awesome specially the new purchasable plants that costs a hundred gems but I just noticed that Sun Bean is not doing it's thing
Hasil Tokenisasi	[More, challenges, ., ,, Everything, in, this, update, is, awesome, specially, the, new, purchasable, plants, that, costs, a, hundred, gems, but, I, just, noticed, that, Sun, Bean, is, not, doing, it, 's, thing]

Pada sampel salah satu hasil dari tokenisasi yang berhasil dilakukan oleh model, dapat dilihat bahwa model yang digunakan pada prototipe ini telah dapat memisahkan kata-kata yang terdapat dalam satu kalimat dengan baik, terutama pada tanda baca yang banyak digunakan pada ulasan pelanggan ini. Tokenisasi dilakukan terhadap setiap kalimat yang dihasilkan pada tahapan sebelumnya. Karena hasil dari tokenisasi telah

dianggap sesuai untuk penelitian ini, maka dapat dilanjutkan ke tahap berikutnya.

### 6.3. Hasil Pengembangan Prototipe

Pada bagian ini, akan dijelaskan mengenai hasil dari pengembangan prototipe yang telah dilakukan pada tahap eliminasi.

#### 6.3.1. Hasil dan Pembahasan *Labeling* dengan POS Tagging

*Labeling* dilakukan dengan menggunakan bahasa pemrograman Java dan dengan model untuk POS Tagging dari Apache OpenNLP. Model POS Tagging pada Apache OpenNLP akan memberikan label berdasarkan token-token yang telah dihasilkan pada tahap sebelumnya. Hasil dari *labeling* tersebut disesuaikan dengan konsep POS Tagging yang telah dibahas pada tahap sebelumnya dan melakukan pengecekan apakah hasil dari label yang diberikan telah sesuai dengan label yang ada pada konsep POS Tagging dan telah diberikan pada kata yang sesuai. Untuk memastikan bahwa model POS Tagging telah memberikan label yang sesuai pada setiap kata, maka dilakukan pengecekan secara manual terhadap sampel hasil *labeling* yang telah dihasilkan pada tahap implementasi sebelumnya. Pengecekan kesesuaian tersebut dapat dilihat pada tabel berikut :

**Tabel 6.4 Pengecekan Kesesuaian Hasil *Labeling* dengan Model POS Tagging**

No	Kata	Label yang Dihasilkan	Kecocokan kata yang diberikan dengan label yang dihasilkan
1	<b>Very</b>	RB (Adverb)	Telah sesuai, karena mendeskripsikan kata keterangan

2	<b>Pleased</b>	VBD (Verb, Past Tense)	Telah sesuai, karena mendeskripsikan kata kerja dalam bentuk <i>past tense</i>
3	<b>That</b>	IN (Preposition)	Telah sesuai, karena mendeskripsikan konjungsi
4	<b>Sun</b>	NNP (Proper Noun, Singular)	Telah sesuai, karena mendeskripsikan kata benda tunggal
5	<b>Bean</b>	NNP (Proper Noun, Singular)	Telah sesuai, karena mendeskripsikan kata benda tunggal
6	<b>Is</b>	VBZ (Verb, 3rd person singular present)	Telah sesuai, karena mendeskripsikan kata kerja tunggal
7	<b>Back</b>	RB (Adverb)	Telah sesuai, karena mendeskripsikan kata keterangan
8	<b>Like</b>	IN (Preposition)	Telah sesuai, karena mendeskripsikan konjungsi
9	<b>The</b>	DT (Determiner)	Telah sesuai
10	<b>Old</b>	JJ (Adjective)	Telah sesuai, karena mendeskripsikan kata sifat tunggal
11	<b>Days</b>	NNS (Noun, plural)	Telah sesuai, karena mendeskripsikan kata benda jamak
12	<b>Homing</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal

13	<b>Thistle</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal
14	<b>Still</b>	RB (Adverb)	Telah sesuai, karena mendeskripsikan kata keterangan
15	<b>Attacking</b>	VBG (Verb, gerund or past participle)	Telah sesuai, karena telah mendeskripsikan kata kerja
16	<b>Air</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal
17	<b>A</b>	DT (Determiner)	Telah sesuai
18	<b>Lot</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal
19	<b>But</b>	CC (Coordinating Conjunction)	Telah sesuai, karena mendeskripsikan konjungsi penghubung
20	<b>With</b>	IN (Preposition)	Telah sesuai, karena mendeskripsikan konjungsi
21	<b>Recent</b>	JJ (Adjective)	Telah sesuai, karena mendeskripsikan kata sifat tunggal
22	<b>Level</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal
23	<b>Up</b>	RB (Adverb)	Telah sesuai, karena mendeskripsikan kata keterangan
24	<b>It</b>	PRP (Personal Pronoun)	Telah sesuai, karena mendeskripsikan kata ganti orang

25	<b>'s</b>	VBZ (Verb, 3rd person singular present)	Telah sesuai, karena mendeskripsikan kata kerja tunggal
26	<b>Attack</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal
27	<b>Is</b>	VBZ (Verb, 3rd person singular present)	Telah sesuai, karena mendeskripsikan kata kerja tunggal
28	<b>Upgraded</b>	VCN (Verb, past participle)	Telah sesuai, karena telah mendeskripsikan kata kerja lampau
29	<b>And</b>	CC (Coordinating Conjunction)	Telah sesuai, karena mendeskripsikan konjungsi penghubung
30	<b>It</b>	PRP (Personal Pronoun)	Telah sesuai, karena mendeskripsikan kata ganti orang
31	<b>'s</b>	VBZ (Verb, 3rd person singular present)	Telah sesuai, karena mendeskripsikan kata kerja tunggal
32	<b>Not</b>	RB (Adverb)	Telah sesuai, karena mendeskripsikan kata keterangan
33	<b>Bad</b>	JJ (Adjective)	Telah sesuai, karena mendeskripsikan kata sifat tunggal
34	<b>Sadly</b>	NNP (Proper Noun, Singular)	Telah sesuai, karena mendeskripsikan kata benda tunggal
35	<b>Surfer</b>	NNP (Proper Noun, Singular)	Telah sesuai, karena mendeskripsikan kata benda tunggal

36	<b>Zombie</b>	NNP (Proper Noun, Singular)	Telah sesuai, karena mendeskripsikan kata benda tunggal
37	<b>And</b>	CC (Coordinating Conjunction)	Telah sesuai, karena mendeskripsikan konjungsi penghubung
38	<b>Ballon</b>	NNP (Proper Noun, Singular)	Telah sesuai, karena mendeskripsikan kata benda tunggal
39	<b>Zombie</b>	NNP (Proper Noun, Singular)	Telah sesuai, karena mendeskripsikan kata benda tunggal
40	<b>Also</b>	RB (Adverb)	Telah sesuai, karena mendeskripsikan kata keterangan
41	<b>Upgraded</b>	VBD (Verb, Past Tense)	Telah sesuai, karena mendeskripsikan kata kerja dalam bentuk <i>past tense</i>
42	<b>Surfer</b>	NNP (Proper Noun, Singular)	Telah sesuai, karena mendeskripsikan kata benda tunggal
43	<b>Zombie</b>	NNP (Proper Noun, Singular)	Telah sesuai, karena mendeskripsikan kata benda tunggal
44	<b>Able</b>	JJ (Adjective)	Telah sesuai, karena mendeskripsikan kata sifat tunggal
45	<b>To</b>	TO	Telah sesuai
46	<b>Smash</b>	VB (Verb, base form)	Telah sesuai, karena mendeskripsikan kata kerja dasar
47	<b>A</b>	DT (Determiner)	Telah sesuai

48	<b>Plant</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal
49	<b>And</b>	CC (Coordinating Conjunction)	Telah sesuai, karena mendeskripsikan konjungsi penghubung
50	<b>Throwing</b>	VBG (Verb, gerund or past participle)	Telah sesuai, karena telah mendeskripsikan kata kerja
51	<b>Another</b>	DT (Determiner)	Telah sesuai
52	<b>Surf</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal
53	<b>Board</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal
54	<b>While</b>	IN (Preposition)	Telah sesuai, karena mendeskripsikan konjungsi
55	<b>Dying</b>	VBG (Verb, gerund or past participle)	Telah sesuai, karena telah mendeskripsikan kata kerja
56	<b>While</b>	IN (Preposition)	Telah sesuai, karena mendeskripsikan konjungsi
57	<b>Ballon</b>	NNP (Proper Noun, Singular)	Telah sesuai, karena mendeskripsikan kata benda tunggal
58	<b>Zombie</b>	NNP (Proper Noun, Singular)	Telah sesuai, karena mendeskripsikan kata benda tunggal
59	<b>Sometime</b>	RB (Adverb)	Telah sesuai, karena mendeskripsikan kata keterangan

60	<b>Able</b>	JJ (Adjective)	Telah sesuai, karena mendeskripsikan kata sifat tunggal
61	<b>To</b>	TO	Telah Sesuai
62	<b>Re</b>	VB (Verb, base form)	Telah sesuai, karena mendeskripsikan kata kerja dasar
63	<b>Inflate</b>	VB (Verb, base form)	Telah sesuai, karena mendeskripsikan kata kerja dasar
64	<b>The</b>	DT (Determiner)	Telah sesuai
65	<b>Ballon</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal
66	<b>After</b>	IN (Preposition)	Telah sesuai, karena mendeskripsikan konjungsi
67	<b>It</b>	RP (Particle)	Telah sesuai, karena mendeskripsikan kata partikel
68	<b>'s</b>	VBZ (Verb, 3rd person singular present)	Telah sesuai, karena mendeskripsikan kata kerja tunggal
69	<b>Been</b>	VCN (Verb, past participle)	Telah sesuai, karena telah mendeskripsikan kata kerja lampau
70	<b>Exploded</b>	VCN (Verb, past participle)	Telah sesuai, karena telah mendeskripsikan kata kerja lampau
71	<b>Pinata</b>	NNP (Proper Noun, Singular)	Telah sesuai, karena mendeskripsikan kata benda tunggal

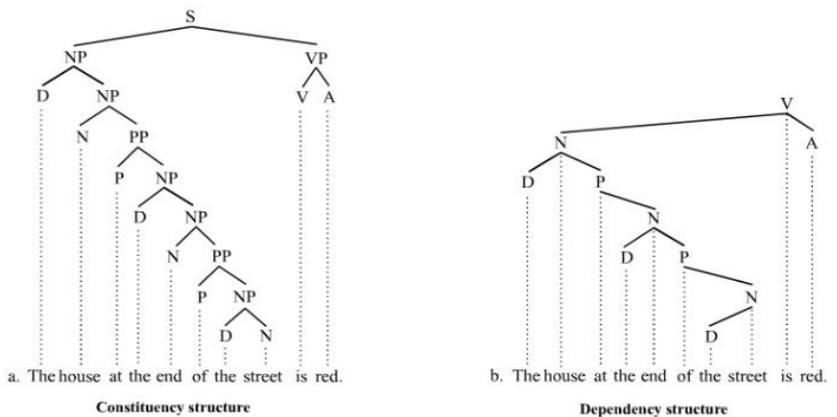
72	<b>Is</b>	VBZ (Verb, 3rd person singular present)	Telah sesuai, karena mendeskripsikan kata kerja tunggal
73	<b>Missing</b>	VBG (Verb, gerund or past participle)	Telah sesuai, karena telah mendeskripsikan kata kerja
74	<b>The</b>	DT (Determiner)	Telah sesuai
75	<b>Pinata</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal
76	<b>Party</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal
77	<b>And</b>	CC (Coordinating Conjunction)	Telah sesuai, karena mendeskripsikan konjungsi penghubung
78	<b>Seed</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal
79	<b>Pinata</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal
80	<b>Timer</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal
81	<b>When</b>	WRB (Wh-adverb)	Telah sesuai, karena mendeskripsikan kata keterangan waktu
82	<b>The</b>	DT (Determiner)	Telah sesuai
83	<b>Seed</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal

84	<b>Pinata</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal
85	<b>Timer</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal
86	<b>Reaching</b>	VBG (Verb, gerund or past participle)	Telah sesuai, karena telah mendeskripsikan kata kerja
87	<b>Zero</b>	CD (Cardinal Number)	Telah sesuai, karena mendeskripsikan angka
88	<b>Only</b>	RB (Adverb)	Telah sesuai, karena mendeskripsikan kata keterangan
89	<b>To</b>	TO	Telah sesuai
90	<b>Be</b>	VB (Verb, base form)	Telah sesuai, karena mendeskripsikan kata kerja dasar
91	<b>Available</b>	JJ (Adjective)	Telah sesuai, karena mendeskripsikan kata sifat tunggal
92	<b>The</b>	DT (Determiner)	Telah sesuai
93	<b>Next</b>	JJ (Adjective)	Telah sesuai, karena mendeskripsikan kata sifat tunggal
94	<b>Seed</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal
95	<b>Pinata</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal
96	<b>Travelog</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal

97	<b>Quest</b>	NN (Noun, singular or mass)	Telah sesuai, karena mendeskripsikan kata benda tunggal
98	<b>Which</b>	WDT (Wh-determiner)	Telah sesuai
99	<b>Can</b>	MD (Modal)	Telah sesuai
100	<b>Be</b>	VB (Verb, base form)	Telah sesuai, karena mendeskripsikan kata kerja dasar
101	<b>The</b>	DT (Determiner)	Telah sesuai
102	<b>Next</b>	JJ (Adjective)	Telah sesuai, karena mendeskripsikan kata sifat tunggal
103	<b>4</b>	CD (Cardinal Number)	Telah sesuai, karena mendeskripsikan angka
104	<b>Hours</b>	NNS (Noun, Plural)	Telah sesuai, karena mendeskripsikan kata benda jamak
105	<b>It</b>	PRP (Personal Pronoun)	Telah sesuai, karena mendeskripsikan kata ganti orang
106	<b>'s</b>	VBZ (Verb, 3rd person singular present)	Telah sesuai, karena mendeskripsikan kata kerja tunggal
107	<b>Very</b>	RB (Adverb)	Telah sesuai, karena mendeskripsikan kata keterangan
108	<b>Vexing</b>	JJ (Adjective)	Telah sesuai, karena mendeskripsikan kata sifat tunggal

*Labeling* merupakan proses pemberian label terhadap kata sesuai dengan jenis kata tersebut. Pada penelitian ini digunakan Bahasa Inggris, maka label yang digunakan merupakan jenis-

jenis kata yang terdapat pada bahasa tersebut. Dalam konsep linguistik, label ini diambil dari konsep *treebank*. *Treebank* adalah sebuah bagian dari korpus teks yang menganotasi struktur kalimat yang sintetis atau semantik [27]. Konstruksi dari bagian korpus ini dikembangkan sejak akhir tahun 1990an sebagai bagian dari revolusi di linguistik komputasional yang merupakan keuntungan dari besarnya data empiris. Lebih jelasnya, konsep *treebank* tersebut dapat dilihat pada gambar berikut :



**Gambar 6.1 Konsep *Treebank* Pada Linguistik Bahasa Inggris [10]**

Contoh kalimat yang diambil pada *treebank* yang dimuat pada gambar diatas adalah “The house at the end of the street is red.” Terdapat dua metode untuk melakukan strukturalisasi terhadap kalimat tersebut, yaitu secara konstitusional atau hierarki dan secara ketergantungan antar kata yang terdapat pada kalimat tersebut. Pada Apache OpenNLP, struktur yang digunakan adalah berdasarkan konstitusional.

Dari satu kalimat ulasan pelanggan tersebut terdapat 108 token yang berisi kata. Dari 108 token tersebut, *pre-trained* model akan memberikan label yang sesuai untuk kata tersebut, sebagai hasil dari *training* sebelumnya yang telah dilakukan Apache OpenNLP terhadap model tersebut. Dari 108 token yang

dihasilkan dari sampel kalimat ulasan pelanggan, *pre-trained* model POS Tagging yang telah disediakan oleh Apache OpenNLP telah berhasil mendeteksi dan memberikan label kepada 108 token tersebut dengan benar.

Oleh karena itu, hasil *labeling* menggunakan program telah dirasa memuaskan karena akurasi yang tinggi, yang dapat dilihat dari tidak adanya kesalahan dalam pemberian label terhadap setiap kata yang terdapat pada token pada kalimat yang dijadikan sampel ini. Oleh karena itu, pemberian *label* secara manual tidak perlu dilakukan dan tahap selanjutnya, yaitu melakukan pemodelan Chunk Parser dapat dilakukan.

### 6.3.2. Hasil dan Pembahasan Validasi Model

Validasi model dilakukan untuk memastikan bahwa model yang dibuat untuk melakukan ekstraksi ekspresi kata kerja ini telah mampu melakukan proses *chunking* tersebut dengan baik. Untuk melakukan validasi terhadap model, maka digunakan Apache OpenNLP, dengan *tools* validasi yang telah disediakan oleh *library* tersebut dengan menggunakan konsep Precision, Recall dan F-Measure sesuai dengan rancangan validasi pada penelitian ini. Pada tahap perancangan sebelumnya, target validasi yang telah dibuat adalah :

- Precision : 0.80 (80%)
- Recall : 0.80 (80%)
- F-Measure : 0.70 (70%)

Setelah menentukan target tersebut, maka dilakukan validasi terhadap model yang digunakan tersebut. Hasil validasi yang dilakukan terhadap model dapat dilihat pada tabel berikut :

**Tabel 6.5 Hasil Validasi Model**

Model yang Digunakan	Hasil Precision	Hasil Recall	Hasil F-Measure
<b>en-chunker.bin</b>	92,50%	92,22%	92,40%

Precision, Recall dan F-Measure juga telah dilakukan terhadap setiap *tag* yang terdapat pada konsep *chunker*. Hasil dari validasi tersebut dapat dilihat pada gambar berikut :

<i>Tag/Label</i>	<i>Deskripsi</i>	<i>Hasil Precision</i>	<i>Hasil Recall</i>	<i>Hasil F-Measure</i>
<b>NP</b>	<i>Noun Phrase</i>	93,22%	93,21%	93,21%
<b>PP</b>	<i>Prepositional Phrase</i>	94,21%	97,69%	95,92%
<b>VP</b>	<i>Verb Phrase</i>	92,52%	96,20%	91,20%
<b>ADVP</b>	<i>Adverb Phrase</i>	80,79%	77,71%	79,22%
<b>ADJP</b>	<i>Adjective Phrase</i>	79,67%	67,12%	72,86%
<b>CONJF</b>	<i>Subordinating Conjunction</i>	57,14%	44,44%	50,00%
<b>PRT</b>	<i>Particle</i>	75,00%	59,43%	66,32%
<b>INTJ</b>	<i>Interjection</i>	50,00%	50,00%	50,00%

Pada tabel tersebut, dapat dilihat label-label penting seperti NP, VP dan VP telah memiliki persentase Precision, Recall dan F-Measure yang tinggi. Sedangkan label lain, cenderung memiliki persentase validasi yang rendah. Hal ini dikarenakan data yang digunakan pada saat *training* kemungkinan besar tidak memasukkan cukup data pada label-label tersebut sehingga pada saat memprediksi kata dengan label-label CONF, PRT dan INTJ hasil persentase validasi cenderung rendah.

Meskipun begitu, secara keseluruhan model telah menghasilkan persentase akurasi tertinggi, terutama untuk label NP, VP, dan PP yang sebagian besar akan mendominasi data ulasan pelanggan dan tujuan utama dari dibuatnya model ini. Karena target persentase validasi telah tercapai, maka dapat dilakukan langkah selanjutnya pada penelitian ini, yaitu melakukan *chunking*. Hasil dari *chunking* tersebut akan dijelaskan pada bagian selanjutnya.

### 6.3.3. Hasil dan Pembahasan Pemodelan Chunk Parser pada Aplikasi *Plant vs Zombies*

Setelah proses POS Tagging dilakukan, maka langkah selanjutnya adalah melakukan *chunking* dengan model Chunk Parser yang telah dibuat. Proses *chunking* yang dilakukan oleh model tersebut meliputi pemberian label *chunks* yang sesuai. Perbedaan label *chunks* dengan label yang diberikan pada POS Tagging pada dasarnya terletak pada kata yang dilabeli. *Chunks* akan membentuk frase-frase yang terdiri dari beberapa kata, sedangkan POS Tagging hanya mengandung satu kata yang dilabeli dengan jenis tertentu. Hal ini juga yang menyebabkan label yang terdapat pada konsep *chunking* jauh lebih sedikit dari label yang akan diberikan pada model POS Tagging. Karena itulah, meskipun *chunking* juga merupakan proses *labeling* terhadap token dari kata-kata yang terbentuk dari segmentasi kalimat, namun *label* yang diberikan oleh model *chunking* merupakan *label* hasil dari penggabungan beberapa label dari POS Tagging

Secara sederhana, *chunking* menggabungkan label-label yang telah dihasilkan oleh POS Tagging menjadi frase-frase yang memuat informasi sederhana yang dapat lebih mudah dipahami oleh orang awam. Untuk lebih jelasnya, struktur dari sebuah *chunk* dapat dilihat pada tabel berikut :

**Tabel 6.6 Struktur dari Sebuah *Chunk***

Bagian Pertama	Bagian Kedua	Bagian Ketiga (Opsional)
B : Menandai mulainya sebuah <i>chunk</i>	NP : <i>chunk</i> berjenis frase kata benda	NP : <i>chunk</i> berjenis frase kata benda
I : menandai kelanjutan dari sebuah <i>chunk</i>	VP : <i>chunk</i> berjenis frase kata benda	VP : <i>chunk</i> berjenis frase kata benda

E : Menandai akhir dari sebuah <i>chunk</i>	ADJP : <i>chunk</i> berjenis frase kata sifat	ADJP : <i>chunk</i> berjenis frase kata sifat
	ADVP : <i>chunk</i> berjenis frase kata keterangan	ADVP : <i>chunk</i> berjenis frase kata keterangan
	PP : <i>chunk</i> berjenis frase kata penunjuk tempat	PP : <i>chunk</i> berjenis frase kata penunjuk orang
	PRT : <i>chunk</i> berjenis frase kata partikel	PRT : <i>chunk</i> berjenis frase kata partikel
	SBAR : <i>chunk</i> berjenis frase kata konjungsi	SBAR : <i>chunk</i> berjenis frase kata konjungsi
	INTJ : <i>chunk</i> berjenis frase kata sapaan	INTJ : <i>chunk</i> berjenis frase kata sapaan

Label pada *chunking* yang terdapat pada tabel diatas merupakan gabungan dari berbagai macam label kata per kata setelah dilakukan tokenisasi dan diberikan oleh model POS Tagging pada langkah sebelumnya. Penjelasan mengenai gabungan label dari POS Tagging yang akan menghasilkan suatu *chunk* tertentu dapat dilihat pada tabel berikut [28] :

**Tabel 6.7 Penggabungan Label POS Tagging Menjadi *Chunk***

Tag	Deskripsi	Label Kata pada POS Tagging	Contoh
NP	<i>Noun Phrase</i>	DT+RB+JJ+NN+PR	“ <i>The strange bird</i> ”
PP	<i>Prepositional Phrase</i>	TO+IN	“ <i>in between</i> ”
VP	<i>Verb Phrase</i>	RB+MD+VB	“ <i>was looking</i> ”

ADVP	<i>Adverb Phrase</i>	RB	“also”
ADJP	<i>Adjective Phrase</i>	CC+RB+JJ	“warm and cozy”
SBAR	<i>Subordinating Conjunction</i>	IN	“whether or not”
PRT	<i>Particle</i>	RP	“up the stairs”
INTJ	<i>Interjection</i>	UH	“hello”

Namun, pada penelitian ini, hanya melakukan *chunking* pada frase kata kerja yang dilabeli dengan VP pada setiap kata yang terdapat kalimat ulasan pelanggan saja dirasa tidak cukup untuk membentuk ekspresi kata kerja yang dapat menggambarkan aplikasi dengan baik. Oleh karena itu, perlu dibuat pemodelan baru untuk melakukan *chunking* yang dapat memenuhi kebutuhan penelitian ini.

Pemodelan untuk melakukan *chunk parser* yang sesuai dimulai dengan mengambil semua frase kata yang mengandung label VP (Kata Kerja). Setelah itu, model akan mendeteksi di bagian kiri dari kata tersebut. Jika mengandung kata berlabel NP (Kata Benda), maka kata tersebut juga akan diambil dan diletakkan di sebelah kiri dari kata dengan label VP tersebut. Setelah itu, dilakukan pendeteksian kembali pada bagian kanan dari kata berlabel VP. Jika dideteksi kata dengan label lain selain dengan VP, maka kata tersebut juga akan diambil dan diletakkan pada bagian kanan dari kata berlabel VP. Kata-kata tersebut lalu digabungkan menjadi satu kesatuan yang lalu dapat disebut sebagai ekspresi kata kerja.

Agar tidak menghasilkan satu kalimat yang terlalu panjang dan rumit sehingga sulit dipahami secara cepat oleh pengguna, maka *chunk* yang diambil dibatasi hanya sampai *chunks* dengan jumlah tertentu. Maka dari itu, dilakukan percobaan dengan mengubah nilai *chunks* yang diambil untuk melihat jumlah *chunks* yang paling efektif untuk menggambarkan ekspresi kata kerja tersebut.

### 6.3.3.1. Hasil dan Pembahasan Pemodelan Chunk Parser dengan 3 *Chunks* pada Aplikasi *Plant vs Zombies*

Hasil kalimat pemodelan chunk parser dengan batas maksimal *chunks* sejumlah 3 dan kalimat-kalimat tersebut telah dilakukan *sorting* dengan frekuensi paling sering muncul dapat dilihat pada tabel berikut :

Tabel 6.8 Sampe 10 teratas Hasil *Chunking* dan *Sorting* dengan 3 *chunks*

No	Hasil <i>Chunks</i>	Frekuensi Muncul
1	i love this game	18
2	i love it	13
3	i love the game	11
4	i love it,	9
5	i don't like	6
6	i think it	5
7	i play it	5
8	i love this game	5
9	the game is awesome	5
10	this is the best game	5

Pada percobaan dengan 3 *chunks* ini, dapat dilihat bahwa kalimat yang dihasilkan dapat dipahami dengan mudah. Struktur kata yang dihasilkan juga telah sesuai dengan tata bahasa sederhana yang ingin didapatkan dari ulasan pelanggan yang biasanya terdiri dari paragraf panjang yang membuat pengguna dengan waktu terbatas kesulitan untuk memahami ulasan-ulasan tersebut. Kalimat-kalimat ini diambil sesuai dengan hasil segmentasi kalimat dan tokenisasi, sehingga satu kalimat dan token yang dihasilkan akan diambil satu frase sesuai dengan konsep ekstraksi ekspresi kata kerja yang diinginkan.

*Sorting* yang dihasilkan juga telah menunjukkan kalimat sederhana dengan frekuensi terbanyak berada di posisi teratas,

dan terus menurun sesuai dengan jumlah frekuensi yang semakin sedikit.

Kalimat yang paling sering muncul dapat dengan mudah dipahami maknanya, namun belum menggambarkan ekspresi kata kerja yang sempurna. Kalimat “*i love this game*”, “*i love the game*”, dan sebagainya tersebut lebih menggambarkan pendapat pengguna mengenai game tersebut secara keseluruhan. Sedangkan, yang ingin ditonjolkan pada penelitian ini adalah ekstraksi ekspresi kata kerja yang mampu menggambarkan apa yang dapat pengguna lakukan dengan game tersebut.

Oleh karena itu, perlu dilakukan percobaan kembali terhadap *chunks* yang diambil untuk mengetahui hasil mana yang paling baik dan menghasilkan ekspresi kata kerja yang dapat menggambarkan penggunaan game tersebut dengan mengubah jumlah *chunks* yang akan diambil pada setiap kalimat yang terdapat pada ulasan pelanggan tersebut.

### 6.3.3.2. Hasil dan Pembahasan Pemodelan Chunk Parser dengan 4 *Chunks* pada Aplikasi *Plant vs Zombies*

Hasil kalimat pemodelan chunk parser dengan batas maksimal *chunks* sejumlah 3 dan kalimat-kalimat tersebut telah dilakukan *sorting* dengan frekuensi paling sering muncul dapat dilihat pada tabel berikut :

**Tabel 6.9 Sampel 10 Teratas Hasil Chunking dan Sorting dengan 4 *chunks***

No	Hasil <i>Chunks</i>	Frekuensi Muncul
1	i love it so	10
2	i lost all of	8
3	i love this game it	7
4	i opened it again,	7
5	i worked hard for	6
6	it is one of	5
7	i have to start all over	5

8	i play this game for	5
9	the game is awesome but i	5
10	i get it back?,	5

Hasil *chunks* pada tabel diatas menghasilkan kalimat-kalimat yang telah menggambarkan ekspresi kata kerja yang lebih akurat dibandingkan dengan pada saat *chunks* yang diambil hanya berjumlah 3. Namun, tata bahasa yang dihasilkan juga menjadi lebih buruk dibandingkan dengan tata bahasa saat 3 *chunks* diambil. Selain itu, dapat dilihat juga, frekuensi dari tiap kalimat yang muncul dan memiliki kesamaan menurun.

Dapat dilihat pada tabel tersebut, kalimat yang memiliki kesamaan dengan frekuensi tertinggi hanya berjumlah 4, yang sangat berbeda dengan pada saat *chunks* yang diambil berjumlah 3, yang memiliki frekuensi kesamaan kalimat dengan jumlah 18. Selain itu, pada kalimat dengan frekuensi yang tertinggi sebagian besar memiliki persamaan dengan kalimat dengan frekuensi tertinggi pada saat menggunakan 3 *chunks* pada saat mengekstrak hasil.

Hasil dari *chunks* diatas telah cukup menggambarkan ekspresi kata kerja, namun dikarenakan tata bahasa yang dihasilkan, dilihat dari perspektif *grammar* pada Bahasa Inggris. Hasil ekstraksi kalimat tersebut mengalami penurunan dalam kualitas tata bahasa yang cukup mengganggu pembaca. Meskipun tidak terdapat perbedaan yang berarti dari arti kalimat secara harfiah yang berhasil diekstrak pada saat 3 *chunks* dengan 4 *chunks* ini. Oleh karena itu, dilakukan percobaan kembali untuk mendapatkan hasil ekspresi kata kerja yang lebih baik dengan metode yang sama, yaitu mengubah jumlah *chunks* pada prototipe tersebut.

### 6.3.3.3. Hasil dan Pembahasan Pemodelan Chunk Parser dengan 5 *Chunks* pada *Plant vs Zombies*

Hasil kalimat pemodelan chunk parser dengan batas maksimal *chunks* sejumlah 5 dan kalimat-kalimat tersebut telah dilakukan *sorting* dengan frekuensi paling sering muncul dapat dilihat pada tabel berikut :

Tabel 6.10 Hasil Chunking dan Sorting dengan 5 *chunks*

No	Hasil <i>Chunks</i>	Frekuensi Muncul
1	this game is amazing, i love it	7
2	only reward is 1000 coins if i'm not getting wrong	5
3	i clear the data and i came back	4
4	happy plants, won't work on me	4
5	my second gmail account same result.it had to do something with recent updates.	4
6	i emailed you's no reply back	4
7	superb, fun way to waste some time, the best game, 100%	3
8	i was a big fan of the original	3
9	this game was good but the map changed and everything	3
10	i love this game because it is a old game	2

Hasil 5 *chunks* pada tabel diatas menghasilkan kalimat-kalimat yang menggambarkan ekspresi kata kerja yang lebih jelas dibandingkan dengan pada saat *chunks* yang diambil hanya berjumlah 3 maupun 4. Namun, meskipun ekspresi kata kerja yang dihasilkan lebih jelas dibandingkan dengan *chunks* yang berjumlah 3 maupun 4, tata bahasa yang dihasilkan dengan mengambil 5 *chunks* memiliki tingkat kesamaan frekuensi yang lebih rendah jika dibandingkan dengan hasil sebelumnya.

Hal ini dapat dilihat pada tabel tersebut, frekuensi kalimat yang memiliki kesamaan dengan frekuensi tertinggi hanya berjumlah 7, meskipun sedikit lebih tinggi dibandingkan dengan saat mengambil dengan 4 *chunks* namun tetap saja frekuensi tersebut lebih rendah dibandingkan ketika menggunakan 3 *chunks* pada program. Namun, ketika menggunakan 5 *chunks* ini, kalimat yang dihasilkan jauh berbeda dengan saat menggunakan 4 *chunks* maupun 3 *chunks*.

Kemungkinan penyebab perbedaan frekuensi kalimat yang memiliki kesamaan antara 3 dan 4 *chunks* dengan 5 *chunks*

karena semakin banyak *chunks* yang diambil akan menyebabkan semakin banyak kata yang terdapat pada suatu kalimat semakin banyak. Hal tersebut menyebabkan kalimat-kalimat akan menjadi semakin unik sehingga perbedaan antar kalimat semakin besar dan menyebabkan program semakin sulit menemukan perbedaan antar kalimat tersebut. Sehingga, frekuensi kesamaan yang dihasilkan menjadi semakin sedikit antara satu kalimat dengan kalimat lainnya.

Perbedaan kalimat-kalimat yang dihasilkan saat menggunakan 5 *chunks* ini kemungkinan besar diakibatkan oleh jumlah *chunks* yang diambil tersebut. Semakin banyak jumlah *chunks* yang diambil, maka kata yang terdapat dalam sebuah kalimat yang menghasilkan ekspresi kata kerja tersebut semakin banyak. Hal tersebut menyebabkan perbedaan yang cukup signifikan terhadap kesatuan kalimat yang berhasil diambil ketika menggunakan banyak *chunks*.

Hasil ekstraksi ekspresi kata kerja yang dihasilkan oleh prototipe dengan 5 *chunks* ini merupakan hasil yang paling sesuai menurut penelitian ini. Meskipun kesalahan tata bahasa tetap terjadi, seperti yang telah dijelaskan pada tahapan sebelumnya, kesalahan tersebut tidak dapat dihindari dalam penelitian ini dan tidak dapat ditangani. Namun, kalimat-kalimat yang telah memiliki tata bahasa yang benar lebih besar pada percobaan dengan 5 *chunks* ini. Selain itu, hasil ekstraksi ekspresi kata kerja yang dihasilkan juga lebih sesuai dengan tujuan dari penelitian ini dibandingkan dengan percobaan sebelumnya. Sehingga, prototipe dengan 5 *chunks* ini merupakan hasil yang paling sesuai dan akan digunakan untuk melakukan tahapan selanjutnya.

#### **6.3.4. Hasil dan Pembahasan Pemodelan Chunk Parser pada Aplikasi *Instagram***

Setelah dilakukan *chunking* terhadap salah satu aplikasi bergenre game yang terdapat pada Google Play, yaitu *Plant vs Zombies*, maka dilakukan *chunking* terhadap salah satu aplikasi lain yang bergenre *social media*, yaitu *Instagram*. Tujuan

dilakukannya tahapan ini adalah untuk melakukan perbandingan hasil dari genre aplikasi yang berbeda tersebut, apakah *chunking* yang dihasilkan menggambarkan pengalaman penggunaan produk dengan baik atau belum menghasilkan ekstraksi ekspresi kata kerja yang kurang tepat seperti pada aplikasi Plant vs Zombies seperti pada tahap sebelumnya. Berikut adalah hasil *chunking* yang telah dilakukan pada aplikasi Instagram.

#### 6.3.4.1. Hasil dan Pembahasan Pemodelan Chunk Parser dengan 3 *Chunks* pada Aplikasi Instagram

Hasil kalimat pemodelan chunk parser dengan batas maksimal *chunks* sejumlah 3 dan kalimat-kalimat tersebut telah dilakukan *sorting* dengan frekuensi paling sering muncul dapat dilihat pada tabel berikut :

Tabel 6.11 Hasil *Chunking* dan *Sorting* dengan 3 *chunks* pada Instagram

No	Hasil <i>Chunks</i>	Frekuensi Muncul
1	i love it	45
2	i love instagram	21
3	i don't like	13
4	it won't let me	12
5	i think it	11
6	i think it's	11
7	i love this app	9
8	i loved it	8
9	i love ig,	3
10	i can't get on	3

Pada percobaan dengan 3 *chunks* ini, dapat dilihat bahwa kalimat yang dihasilkan dapat dipahami dengan mudah, sama halnya seperti pada hasil ulasan pelanggan pada Plant vs Zombies. Konteks kalimat yang dihasilkan juga sama. Struktur kata yang dihasilkan juga telah sesuai dengan tata bahasa

sederhana yang ingin didapatkan dari ulasan pelanggan yang biasanya terdiri dari paragraf panjang yang membuat pengguna dengan waktu terbatas kesulitan untuk memahami ulasan-ulasan tersebut. Kalimat-kalimat ini diambil sesuai dengan hasil segmentasi kalimat dan tokenisasi, sehingga satu kalimat dan token yang dihasilkan akan diambil satu frase sesuai dengan konsep ekstraksi ekspresi kata kerja yang diinginkan.

*Sorting* yang dihasilkan juga telah menunjukkan kalimat sederhana dengan frekuensi terbanyak berada di posisi teratas, dan terus menurun sesuai dengan jumlah frekuensi yang semakin sedikit.

Kalimat yang paling sering muncul dapat dengan mudah dipahami maknanya, namun belum menggambarkan ekspresi kata kerja yang sempurna. Kalimat “*i love it*”, “*i love instagram*”, dan sebagainya tersebut lebih menggambarkan pendapat pengguna mengenai *social media* tersebut secara keseluruhan. Sedangkan, yang ingin ditonjolkan pada penelitian ini adalah ekstraksi ekspresi kata kerja yang mampu menggambarkan apa yang dapat pengguna lakukan dengan *social media* tersebut, sama halnya seperti pada percobaan yang telah dilakukan pada aplikasi sebelumnya.

#### **6.3.4.2. Hasil dan Pembahasan Pemodelan Chunk Parser dengan 4 *Chunks* pada Aplikasi Instagram**

Hasil kalimat pemodelan chunk parser dengan batas maksimal *chunks* sejumlah 4 dan kalimat-kalimat tersebut telah dilakukan *sorting* dengan frekuensi paling sering muncul dapat dilihat pada tabel berikut :

**Tabel 6.12 Hasil Chunking dan Sorting dengan 4 chunks pada Instagram**

No	Hasil <i>Chunks</i>	Frekuensi Muncul
1	i didn't post anything	32
2	i enjoy seeing my friends and beautiful people	17

3	it helps me keep in touch	12
4	me get my account back	12
5	i love it so	11
6	me log in to	11
7	it disabled my account for	11
8	i love it, it's	11
9	jesus christ!., cant login even after	10
10	there was a problem with	10

Hasil *chunks* pada tabel diatas menghasilkan kalimat-kalimat yang telah menggambarkan ekspresi kata kerja yang lebih akurat dibandingkan dengan pada saat *chunks* yang diambil hanya berjumlah 3. Namun, tata bahasa yang dihasilkan juga menjadi lebih buruk dibandingkan dengan tata bahasa saat 3 chunks diambil, yang ternyata tidak jauh berbeda saat percobaan pada aplikasi game Plant vs Zombies. Selain itu, dapat dilihat juga, frekuensi dari tiap kalimat yang muncul dan memiliki kesamaan menurun sama seperti percobaan sebelumnya.

Dapat dilihat pada tabel tersebut, kalimat yang memiliki kesamaan dengan frekuensi tertinggi hanya berjumlah 32, yang tidak jauh berbeda dengan saat menggunakan 3 chunks. Hal ini berbeda pada saat percobaan dengan menggunakan aplikasi Plant vs Zombies, dimana kesamaan kalimat yang dihasilkan mengalami penurunan drastis. Hal ini kemungkinan besar disebabkan oleh kesamaan topik yang dibahas pada ulasan sehingga kalimat yang dihasilkan mempunyai banyak kesamaan. Hasil dari *chunks* diatas telah cukup menggambarkan ekspresi kata kerja, namun dikarenakan tata bahasa yang dihasilkan, dilihat dari perspektif *grammar* pada Bahasa Inggris. Hasil ekstraksi kalimat tersebut mengalami penurunan dalam kualitas tata bahasa yang cukup mengganggu pembaca. Meskipun tidak terdapat perbedaan yang berarti dari arti kalimat secara harfiah yang berhasil diekstrak pada saat 3 *chunks* dengan 4 *chunks* ini.

### 6.3.4.3. Hasil dan Pembahasan Pemodelan Chunk Parser dengan 5 *Chunks* pada Aplikasi Instagram

Hasil kalimat pemodelan chunk parser dengan batas maksimal *chunks* sejumlah 5 dan kalimat-kalimat tersebut telah dilakukan *sorting* dengan frekuensi paling sering muncul dapat dilihat pada tabel berikut :

**Tabel 6.13 Hasil Chunking dan Sorting dengan 5 chunks pada Instagram**

No	Hasil <i>Chunks</i>	Frekuensi Muncul
1	nice application to share pics with	20
2	share photos to the app	16
3	i can take a nice hd video from	8
4	i can text my friends on	7
5	i tried to turn off the notifications	7
6	please make on and off option	7
7	the app has been deleting a lot of	6
8	i tried recovering but it wouldn't let me	5
9	that lets you post videos and picture	5
10	please add an option to check the offline/online status of	4

Hasil 5 *chunks* pada tabel diatas menghasilkan kalimat-kalimat yang menggambarkan ekspresi kata kerja yang lebih jelas dibandingkan dengan pada saat *chunks* yang diambil hanya berjumlah 3 maupun 4. Namun, meskipun ekspresi kata kerja yang dihasilkan lebih jelas dibandingkan dengan *chunks* yang berjumlah 3 maupun 4, tata bahasa yang dihasilkan dengan mengambil 5 *chunks* memiliki tingkat kesamaan frekuensi yang lebih rendah jika dibandingkan dengan hasil sebelumnya. Hal tersebut juga terjadi pada percobaan dengan aplikasi game Plant vs Zombies

Hal ini dapat dilihat pada tabel tersebut, frekuensi kalimat yang memiliki kesamaan dengan frekuensi tertinggi hanya berjumlah 7, meskipun sedikit lebih tinggi dibandingkan dengan saat

mengambil dengan 4 *chunks* namun tetap saja frekuensi tersebut lebih rendah dibandingkan ketika menggunakan 3 *chunks* pada program. Namun, ketika menggunakan 5 *chunks* ini, kalimat yang dihasilkan jauh berbeda dengan saat menggunakan 4 *chunks* maupun 3 *chunks*.

Namun, pada percobaan aplikasi Instagram ini, hasil kalimat yang berhasil diekstrak lebih menggambarkan ekspresi kata kerja dibandingkan pada percobaan dengan menggunakan aplikasi *game* Plant vs Zombies 2. Hal ini kemungkinan disebabkan karena pengguna yang memberikan ulasan pada halaman ulasan pelanggan yang terdapat pada aplikasi ini sebagian besar membahas topik yang memiliki korelasi satu sama lain. Selain itu, tata bahasa yang terdapat pada ulasan pelanggan pada aplikasi ini kemungkinan lebih baik dibandingkan dengan ulasan pelanggan pada aplikasi Plant vs Zombies sehingga menghasilkan ekstraksi ekspresi kata kerja yang lebih dibandingkan dengan percobaan aplikasi sebelumnya.

Hasil ekstraksi ekspresi kata kerja yang dihasilkan oleh prototipe dengan 5 *chunks* ini merupakan hasil yang paling sesuai menurut penelitian ini. Meskipun kesalahan tata bahasa tetap terjadi, seperti yang telah dijelaskan pada tahapan sebelumnya, kesalahan tersebut tidak dapat dihindari dalam penelitian ini dan tidak dapat ditangani. Namun, kalimat-kalimat yang telah memiliki tata bahasa yang benar lebih besar pada percobaan dengan 5 *chunks* ini. Selain itu, hasil ekstraksi ekspresi kata kerja yang dihasilkan juga lebih sesuai dengan tujuan dari penelitian ini dibandingkan dengan percobaan sebelumnya. Sehingga, prototipe dengan 5 *chunks* ini merupakan hasil yang paling sesuai dan akan dilakukan perbandingan dengan aplikasi sebelumnya secara keseluruhan.

### **6.3.5. Perbandingan Hasil Pada Aplikasi Plant vs Zombies dengan Instagram**

Dalam melakukan perbandingan hasil ekstraksi ekspresi kata kerja yang telah dilakukan pada penelitian ini, maka yang hasil *chunking* yang digunakan adalah hasil ekstraksi 5 *chunks*.

Perbandingan hasil ekstraksi ekspresi kata kerja dapat dilihat pada tabel berikut :

**Tabel 6.14 Perbandingan Ekspresi Kata Kerja yang Dihasilkan**

<b>Plant vs Zombies</b>	<b>Frekuensi</b>	<b>Instagram</b>	<b>Frekuensi</b>
this game is amazing, i love it	7	nice application to share pics with	20
only reward is 1000 coins if i'm not getting wrong	5	share photos to the app	16
i clear the data and i came back	4	i can take a nice hd video from	8
happy plants, won't work on me	4	i can text my friends on	7
my second gmail account same result.it had to do something with recent updates.	4	i tried to turn off the notifications	7
i emailed you's no reply back	4	please make on and off option	7
superb, fun way to waste some time, the best game, 100%	3	the app has been deleting a lot of	6
i was a big fan of the original	3	i tried recovering but it wouldn't let me	5
this game was good but the map changed and everything	3	that lets you post videos and picture	5
i love this game because it is a old game	2	please add an option to check the offline/online status of	4

Beberapa hal yang dapat dicermati dari perbandingan hasil tersebut adalah :

- Hasil ekstraksi kata kerja pada Instagram secara subjektif dari sudut pandang penelitian ini lebih menggambarkan ekspresi kata kerja dibandingkan hasil ekstraksi kata kerja pada aplikasi Plant vs Zombies 2. Dapat dilihat pada salah satu ekspresi kata kerja pada Instagram, yaitu “*nice application to share pics with.*” yang telah mengekspresikan kata kerja “aplikasi bagus untuk berbagi gambar” dan dapat dipahami langsung dengan mudah oleh pengguna. Jika dibandingkan dengan ekspresi kata kerja “*this game is amazing, i love it*” yang berarti “game ini bagus, saya menyukainya” yang lebih menggambarkan pendapat dibandingkan pengalaman penggunaan produk.
- Selain itu, tata bahasa dari ekspresi kata kerja pada Instagram juga dinilai lebih baik dan dapat lebih mudah dipahami maksudnya dibandingkan dengan Plant vs Zombies. Penyebab terjadinya hal ini kemungkinan dikarenakan ulasan pelanggan yang terdapat pada Instagram memiliki tata bahasa yang lebih baik dibandingkan dengan Plant vs Zombies. Selain itu, ulasan pelanggan pada Instagram kemungkinan besar juga membahas topik yang memiliki kemiripan yang lebih besar satu dengan lainnya sehingga lebih memudahkan model dalam melakukan *chunking*.
- Frekuensi yang dihasilkan oleh ekstraksi pada aplikasi Instagram juga lebih besar dibandingkan pada aplikasi Plant vs Zombies. Hal ini juga kemungkinan disebabkan karena topik yang dibahas pada aplikasi Instagram yang memiliki kemiripan yang lebih besar satu dengan lainnya sehingga lebih memudahkan model dalam melakukan *chunking* dan menghasilkan kalimat-kalimat yang tingkat kemiripannya lebih besar dibandingkan dengan aplikasi Plant vs Zombies.

Berdasarkan uraian tersebut, secara keseluruhan, hasil ekstraksi ekspresi kata kerja pada Instagram lebih baik dibandingkan dengan Plant vs Zombies.

#### 6.4. Hasil dan Pembahasan Validasi Prototipe

Validasi prototipe dilakukan untuk memastikan bahwa prototipe yang dihasilkan telah sesuai dengan kebutuhan dan tujuan dari penelitian ini. Metode yang digunakan untuk melakukan validasi prototipe ini adalah *Usability Testing*. *Usability Testing* merupakan salah satu metode yang digunakan untuk melakukan validasi terhadap prototipe suatu sistem. Dalam hal ini, *usability testing* digunakan untuk menguji informasi yang dihasilkan dari *chunks* yang telah dilakukan oleh model ke pengguna.

*Usability testing* akan mengukur apakah informasi yang dihasilkan oleh model yang telah dibuat secara garis besar berguna dan dapat mempermudah pengguna dalam memahami aplikasi dan mengurangi waktu yang dibutuhkan untuk membaca ulasan pelanggan yang cukup banyak jumlahnya. Hasil dari *usability testing* akan menunjukkan apakah model dan tujuan penelitian ini, yaitu untuk melakukan ekstraksi ekspresi kata kerja telah tercapai.

*Usability testing* dilakukan dengan menyebarkan kuisisioner yang berisi 10 pertanyaan mengenai prototipe yang telah dibuat, dan difokuskan pada informasi yang dihasilkan oleh model yang telah dibuat. Kuisisioner tersebut lalu diberikan kepada sampel berupa 10 pengguna aplikasi *Plant vs Zombies* versi Android, yang otomatis mengunduh aplikasi tersebut melalui Google Play dan telah membaca ulasan pelanggan tersebut. 10 pengguna tersebut berasal dari mahasiswa di lingkungan Jurusan Sistem Informasi ITS dan telah menggunakan aplikasi tersebut selama lebih dari 2 bulan. Selain itu, sampel mahasiswa yang diambil tersebut dapat memahami Bahasa Inggris dengan cukup baik, dan pernah membaca ulasan pelanggan yang berada di halaman utama Google Play mengenai aplikasi tersebut.

Hasil dari kuisisioner tersebut lalu dihitung sesuai dengan aturan dari *System Usability Scale* yang telah dijabarkan pada tahap sebelumnya. Hasil dari SUS ini yang akan menentukan apakah prototipe tersebut telah valid dan dapat ditarik kesimpulan dari

penelitian ini. Hasil dari SUS tersebut dapat dilihat pada gambar berikut :

SUS Calculation												
Responden	Jenis Kelamin	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	Total
Ashma Hanifah	2	3	4	3	3	2	3	1	3	3	4	77,5
Nadya Chandra	2	2	3	1	4	3	3	3	3	2	3	72,5
Faisal S Putra	1	3	3	2	4	1	4	1	4	1	5	72,5
M Alvin Darari	1	3	4	2	3	2	3	3	3	1	4	72,5
Umar Al Aqsho	1	2	3	2	4	2	3	1	4	2	3	67,5
Pri Rezki	1	3	3	1	3	3	4	2	3	3	4	75
Ni Ketut Ratih L	2	3	4	3	5	2	3	1	3	2	3	77,5
Robbigh F	1	2	3	3	5	2	3	3	3	2	4	77,5
Hanum Fitriani Ayu K	2	3	5	1	3	3	3	2	3	3	3	77,5
Bambang Setiawan	1	3	3	1	4	2	5	2	4	2	3	75
Rata-rata SUS											74,5	

**Gambar 6.2 Hasil Perhitungan Usability Testing**

Hasil *System Usability Scale* (SUS) yang diberikan oleh pengguna terhadap prototipe pada penelitian ini memiliki kegunaan sebesar 74,5% berdasarkan rata-rata nilai SUS yang diberikan oleh setiap pengguna. Hasil tersebut menunjukkan bahwa setiap rata-rata pengguna telah setuju bahwa prototipe ini berguna bagi mereka dalam mempermudah pemahaman informasi mengenai ulasan pelanggan terhadap aplikasi Plant vs Zombies 2 yang terdapat pada *platform* Google Play. Agar lebih jelasnya, justifikasi dari setiap pertanyaan dengan rata-rata hasil jawaban pengguna dari aplikasi tersebut dapat dilihat pada tabel di bawah ini :

**Tabel 6.15 Justifikasi *System Usability Scale* Terhadap Kuisisioner**

Pertanyaan	Justifikasi
Saya merasa bahwa saya informasi mengenai ulasan pelanggan yang terdapat pada sistem ini berguna	Setuju
Saya berpikir sistem ini sangat rumit dan tidak perlu	Tidak Setuju
Saya berpikir bahwa sistem ini mudah digunakan	Setuju

<b>Saya berpikir saya akan membutuhkan bantuan seseorang untuk dapat menggunakan sistem ini</b>	Tidak Setuju
<b>Saya merasa bahwa informasi yang terdapat dalam sistem ini memberikan pengetahuan yang lebih terkait dengan ulasan pelanggan</b>	Setuju
<b>Saya berpikir terlalu banyak inkonsistensi terkait dengan informasi yang ditampilkan dari sistem ini</b>	Tidak Setuju
<b>Saya merasa banyak orang akan memahami informasi yang dimuat dalam sistem ini secara efektif</b>	Setuju
<b>Saya merasa informasi yang dimuat dalam sistem ini sulit untuk dipahami</b>	Tidak Setuju
<b>Saya merasa sangat percaya diri dalam menggunakan sistem ini</b>	Setuju
<b>Saya perlu belajar banyak hal sebelum dapat menggunakan sistem ini</b>	Tidak Setuju

Berdasarkan tabel justifikasi di atas, dapat dilihat bahwa pada setiap pertanyaan yang dihasilkan telah sesuai dengan hasil yang diharapkan. Justifikasi pada tabel tersebut didapatkan dari rata-rata hasil dari perhitungan *SUS scale* yang telah dilakukan dan hasilnya dapat dilihat pada tabel sebelumnya. Secara garis besar, menurut pengguna sistem yang dikembangkan pada penelitian ini berguna dan dibutuhkan untuk mereka. Selain itu, informasi yang ditampilkan pada sistem ini juga berguna menurut pengguna. Justifikasi ini juga menunjukkan bahwa pengguna memiliki pandangan yang positif mengenai aplikasi yang sedang dikembangkan.

*Halaman ini sengaja dikosongkan*

## **BAB VII**

### **KESIMPULAN DAN SARAN**

Pada bagian ini akan dijelaskan mengenai kesimpulan dari hasil penelitian yang telah dilakukan serta saran yang dapat dijadikan referensi bagi penelitian lain di bidang yang sama maupun penelitian lain yang ingin melanjutkan dan memperbaiki penelitian ini.

#### **7.1. Kesimpulan**

Dari hasil pengembangan penelitian yang telah dilakukan, pengembangan rancang bangun prototipe untuk ekstraksi kata kerja dengan metode *text chunking* dan menggunakan studi kasus ulasan pelanggan dari aplikasi Plant vs Zombies 2 pada *platform* Google Play. Tujuan dari penelitian ini adalah untuk memaparkan pengalaman penggunaan produk yang berfungsi untuk memudahkan pelanggan dalam memahami ulasan pelanggan sehingga dapat Kesimpulan yang dapat diambil dari penelitian ini antara lain :

1. Pada penelitian ini, digunakan metode *scraping* untuk mendapatkan data dengan bantuan *library* Jsoup, salah satu *library* berbasis JSON yang dapat membaca dan mengambil data dalam bentuk JSON.
2. Metode yang dilakukan untuk melakukan ekstraksi informasi adalah dengan melakukan *text chunking* dengan bantuan *library* dari Apache OpenNLP. *Labeling* yang dilakukan pada model *text chunking* dengan
3. Struktur penyimpanan data mengenai ulasan produk yang terdapat pada Google Play tersebut disesuaikan dengan struktur yang terdapat pada halaman ulasan pelanggan pada *platform* tersebut, dan disimpan ke dalam sebuah basis data, yaitu MySQL. Secara garis besar, atribut data yang digunakan pada penyimpanan data untuk ulasan pelanggan ini adalah ID, *app\_id*, *userName*, *userImage*, *date*, *url*, *score*, *title*, dan *text*.

4. Proses yang telah dilakukan untuk menghasilkan pengalaman penggunaan produk pada penelitian ini terdiri dari beberapa tahap, yaitu segmentasi kalimat yang berfungsi untuk memecah paragraf-paragraf pada ulasan pelanggan menjadi kalimat-kalimat, tokenisasi yang berfungsi untuk memecah kalimat-kalimat tersebut menjadi kata per kata, pemodelan Chunk Parser yang berfungsi untuk melakukan *chunking*, uji validasi model untuk menguji apakah model dapat menghasilkan ekspresi kata kerja yang valid, pembuatan prototipe yang berfungsi untuk mengekstrak informasi beruka ekspresi kata kerja dan validasi prototipe yang berfungsi untuk menguji apakah prototipe memiliki tingkat usability yang sesuai.
5. Metode yang digunakan untuk melakukan validasi adalah dengan menggunakan *library* yang sama dengan ekstraksi informasi pada tahap sebelumnya, yaitu Apache OpenNLP. Validasi dilakukan terhadap model *chunker* yang digunakan telah menghasilkan tingkat akurasi yang cukup memuaskan secara keseluruhan dengan hasil Precision yaitu 92,50%, Recall yaitu 92,22% dan F-Measure yaitu 92,40%. Selain itu, dilakukan juga uji validasi terhadap prototipe dengan menggunakan metode *Usability Testing* yang menghasilkan persentase sebesar 74,5%, yang menandakan bahwa secara keseluruhan informasi yang dihasilkan pada penelitian ini telah cukup berguna bagi pengguna.
6. Pada penelitian ini, telah dilakukan ekstraksi informasi terhadap dua aplikasi dengan genre yang berbeda, yaitu Plant vs Zombies 2 yang merupakan aplikasi dengan genre *game* dan Instagram yang merupakan aplikasi dengan genre *social media*. Hasil aktivitas yang sering dibahas dapat dilihat pada tabel berikut :

Tabel 7.1 Aktivitas yang Sering Dibahas Pada Aplikasi

Plant vs Zombies	Frekuensi	Instagram	Frekuensi
this game is amazing, i love it	7	nice application to share pics with	20
only reward is 1000 coins if i'm not getting wrong	5	share photos to the app	16
i clear the data and i came back	4	i can take a nice hd video from	8
happy plants, won't work on me	4	i can text my friends on	7
my second gmail account same result.it had to do something with recent updates.	4	i tried to turn off the notifications	7

7. Hasil ekstraksi kata kerja pada Instagram secara subjektif dari sudut pandang penelitian ini lebih menggambarkan ekspresi kata kerja dibandingkan hasil ekstraksi kata kerja dibandingkan pada aplikasi Plant vs Zombies 2. Dapat dilihat pada salah satu ekspresi kata kerja pada Instagram, yaitu “*nice application to share pics with.*” yang telah mengekspresikan kata kerja “aplikasi bagus untuk berbagi gambar” dan dapat dipahami langsung dengan mudah oleh pengguna. Jika dibandingkan dengan ekspresi kata kerja “*this game is amazing, i love it*” yang berarti “game ini bagus, saya menyukainya” yang lebih menggambarkan pendapat dibandingkan pengalaman penggunaan produk.
8. *Library* OpenNLP pada dasarnya telah memiliki tingkat akurasi yang cukup tinggi dan dapat melabeli kata sesuai dengan konsep *text chunking* dengan benar, namun ekstraksi ekspresi kata kerja yang dihasilkan kurang sesuai dengan tujuan penelitian ini. Ekspresi kata kerja yang dihasilkan memiliki tata bahasa yang kurang baik sehingga meskipun masih dapat dipahami

dengan mudah, namun cukup mengganggu pengguna ketika membaca kalimat-kalimat tersebut. Sehingga, perlu dilakukan eksplorasi lebih lanjut terkait dengan model *chunking* yang digunakan

## 7.2. Saran

Saran yang dapat diberikan berdasarkan hasil yang diperoleh dari penelitian ini adalah sebagai berikut :

- Data yang digunakan untuk melakukan penelitian ini sebaiknya dilakukan *cleansing* terlebih dahulu, yang bertujuan untuk mengeliminasi ulasan-ulasan pelanggan dengan kesalahan ejaan maupun tata bahasa yang dapat memperbesar peluang kesalahan ekstraksi pada ekspresi kata kerja yang telah dilakukan oleh model.
- Perlu dilakukan eksplorasi kembali terhadap model *chunker* yang digunakan, apakah telah sesuai dan dapat digunakan untuk melakukan ekstraksi ekspresi kata kerja yang berfungsi untuk memaparkan pengalaman penggunaan produk sehingga dapat menghasilkan ekstraksi ekspresi kata kerja yang dapat memaparkan pengalaman penggunaan produk lebih baik
- Untuk pengembangan yang lebih baik, perlu dilakukan ekplorasi pelabelan lain agar meningkatkan efektivitas dan akurasi dari kalimat yang dihasilkan agar dapat memaparkan pengalaman penggunaan produk secara lebih akurat.

## DAFTAR PUSTAKA

- [1] D. Mayzlin, Y. Dover dan J. A. Chevalier, *An Empirical Investigation of Online Review Manipulation*, Cambridge: National Bureau of Economic Research, 2012.
- [2] A. Sil, F. Huang dan A. Yates, "Extracting Action and Event Semantics from Web Text," *Commonsense Knowledge: Papers from the AAAI Fall Symposium*, vol. 10, no. 2, pp. 108-113, 2010.
- [3] L. Huayi, A. Mukherjee, J. Si dan B. Liu, "Extracting Verb Expressions Implying Negative Opinions," *Association for the Advancement of Artificial Intelligence*, pp. 1-7, 2015.
- [4] D. McClosky, M. Surdenau dan C. D. Manning, "Event Extraction as Dependency Parsing," pp. 1-10, 2012.
- [5] J. Newbery, "Online Reviews Are the New Word of Mouth," *The Huffington Post*, 19 August 2015. [Online]. Available: [http://www.huffingtonpost.com/jorge-newbery/online-reviews-are-the\\_e\\_b\\_8009634.html](http://www.huffingtonpost.com/jorge-newbery/online-reviews-are-the_e_b_8009634.html). [Diakses 10 March 2017].
- [6] N. Solutions, "What Is E-Commerce?," *Network Solutions*, [Online]. Available: <http://www.networksolutions.com/education/what-is-commerce/>. [Diakses 3 February 2015].
- [7] K. Arline, "What Is E-Commerce?," *Business News Daily*, 2015 February 2015. [Online]. Available: <http://www.businessnewsdaily.com/4872-what-is-e-commerce.html>. [Diakses 3 February 2016].
- [8] BloomIdea, "Types of E-Commerce," *BloomIdea*, 5 May 2014. [Online]. Available: <http://bloomidea.com/en/blog/types-e-commerce>. [Diakses 2 February 2016].
- [9] J. Callaham, "The new Google Store is your one-stop web shop for Chromebooks, Nexus devices and more," *AndroidCentral*, 11 March 2015. [Online]. Available: <http://www.androidcentral.com/new-google-store-your-one-stop-web-shop-chromebooks-nexus-devices-and-more>. [Diakses 15 February 2017].

- [10] G. G. Chowdhury, "Natural Language Processing," dalam *Annual review of Information*, 2003, pp. 51-90.
- [11] E. D. Liddy, "Natural Language Processing," dalam *Encyclopedia of library and Information Science, 2nd Ed*, NY, Marcel Decker Inc., 001.
- [12] A. Ratnaparkhi, "A Maximum Entropy for Part-of-Speech Tagging," *Journal of Artificial Intelligence*, vol. 2, no. 1, p. 10, 2009.
- [13] A. R. Martinez, "Part-of-speech tagging," *WIREs Comp Stat*, vol. 4, o. 1, pp. 107-114, 2012.
- [14] A. Voutilainen, "Part Of Speech Tagging," dalam *The Oxford handbook of Computational Linguistics*, Oxford, Oxford University press, 2009, p. 220.
- [15] L. A. Ramshaw dan M. P. Marcus, "Text Chunking using Transformation-Based Learning," *Journal of Computer Science*, pp. 82-5.
- [16] D. Zhu, "Text Chunking Using NLTK," Microsoft Corporation, China, 014.
- [17] J. D. Kelleher, B. M. Namee dan A. D'Arcy, *Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies*, London, England: MIT Press, 2015.
- [18] J. Nielsen, *Usability 101 : Introduction to Usability*, 2012.
- [19] N. Y. Wirawan, *Rancang Bangun Ekstraksi Topik Fitur Produk dari Ulasan Pengguna Online dengan Latent Dirichlet Allocation*, Surabaya: Jurusan Sistem Informasi, 2017.
- [20] S. N. L. Processing, "Tokenization," [Online]. Available: <http://nlp.stanford.edu/IR-book/html/htmledition/tokenization-1.html>. [Diakses 13 February 2017].
- [21] O. D. C. R. 2. (9.2), "The Data Dictionary," 2002. [Online]. Available: [https://docs.oracle.com/cd/B10501\\_01/server.920/a96524/c05dicti.htm](https://docs.oracle.com/cd/B10501_01/server.920/a96524/c05dicti.htm). [Diakses 28 May 2017].
- [22] S. Abney, "Chunking," [Online]. Available: <http://www.cnts.ua.ac.be/conll2000/chunking/>. [Diakses 07 June 2017].

- [23] Oracle, “Class HashMap<K,V>,” [Online]. Available: <https://docs.oracle.com/javase/7/docs/api/java/util/HashMap.html>. [Diakses 03 June 2017].
- [24] T. Point, “Java - The HashMap Class,” [Online]. Available: [https://www.tutorialspoint.com/java/java\\_hashmap\\_class.htm](https://www.tutorialspoint.com/java/java_hashmap_class.htm). [Diakses 3 June 2017].
- [25] CoNLL-2000, “Chunking,” [Online]. Available: <http://www.cnts.ua.ac.be/conll2000/chunking/train.txt.gz>. [Diakses 07 June 2017].
- [26] S. Forge, “opencsv,” 20 January 2017. [Online]. Available: <http://opencsv.sourceforge.net/>. [Diakses 01 June 2017].
- [27] C. F. a. S. L. Alexander Clark, *The handbook of computational linguistics and natural language processing*, Wiley, 2010.
- [28] C. L. & P. R. Center, “Penn Treebank II tag set,” 2002. [Online]. Available: <http://www.clips.ua.ac.be/pages/mbsp-tags>. [Diakses 1 June 2017].

*Halaman ini sengaja dikosongkan*

## BIODATA PENULIS



Penulis lahir di Medan pada tanggal 3 Desember 1995. Merupakan anak pertama dari 3 bersaudara. Penulis dibesarkan di Medan dan telah menempuh beberapa pendidikan formal yaitu; SD Harapan 2 Medan, SMP Harapan 2 Medan dan SMA Negeri 1 Medan.

Pada tahun 2013 pasca kelulusan SMA, penulis melanjutkan pendidikan dengan jalur PKM (Mandiri) di Jurusan Sistem Informasi FTIf – Institut Teknologi Sepuluh Nopember (ITS) Surabaya dan terdaftar sebagai mahasiswa dengan NRP 5213100184. Selama menjadi mahasiswa, penulis mulai aktif di berbagai kegiatan kemahasiswaan semenjak tahun pertama, dimulai dari menjadi panitia dan *volunteer* untuk beberapa kegiatan Himpunan Mahasiswa Sistem Informasi terkait dengan event internal, dan kegiatan BEM ITS yang berafiliasi dengan IECC. Pada tahun kedua, penulis aktif menjadi staff di dua organisasi, yaitu BEM ITS dan HMSI ITS dan menangani ranah internal di kedua organisasi tersebut. Setelah itu, penulis aktif menjadi Ketua Divisi di HMSI pada tahun ketiga. Pada tahun ketiga ini, penulis mendapatkan kesempatan menjadi salah satu *exchange student* di Shibaura Institute of Technology, Tokyo, Jepang. Selain itu, kegiatan seperti Latihan Ketrampilan Manajemen Mahasiswa pun pernah diikuti hingga Tingkat Menengah. Di bidang akademik, penulis aktif menjadi asisten dosen dan asisten praktikum pada beberapa mata kuliah seperti Perencanaan Sumber Daya Perusahaan dan Matematika Diskrit.

Pada tahun keempat, karena penulis memiliki ketertarikan di bidang *enterprise*, maka penulis mengambil bidang minat Sistem Enterprise (SE). Penulis dapat dihubungi melalui *email* di [almiradhara@gmail.com](mailto:almiradhara@gmail.com).

## LAMPIRAN 1 : SAMPEL HASIL SCRAPING ULASAN DATA PELANGGAN

Id	app_id	userName	userImage	date	url	score	title	reviews
gp:AOqpTOE-	com.ea.game.pvz2_row	Kevin Isaiah Pascual	<a href="https://lh4.googleusercontent.com">https://lh4.googleusercontent.com</a>	May 4, 2017	<a href="https://play.google.com/">https://play.google.com/</a>	2		My plant vs zombie 2 app has stopped, how to fix it without clearing the data.
gp:AOqpTOE-	com.ea.game.pvz2_row	Varsha Shadaw	<a href="https://lh4.googleusercontent.com">https://lh4.googleusercontent.com</a>	April 28, 2017	<a href="https://play.google.com/">https://play.google.com/</a>	1		Game is not allowing me to log onto after the update. Please fix.
gp:AOqpTOE-	com.ea.game.pvz2_row	Alex Mercader	<a href="https://lh4.googleusercontent.com">https://lh4.googleusercontent.com</a>	April 22, 2017	<a href="https://play.google.com/">https://play.google.com/</a>	5		think the other premium plants can bought using real money or gems, and the others can only be bought with gems please read this!
gp:AOqpTOE-	com.ea.game.pvz2_row	Sheik Medina	<a href="https://lh4.googleusercontent.com">https://lh4.googleusercontent.com</a>	May 4, 2017	<a href="https://play.google.com/">https://play.google.com/</a>	4		It nice.
gp:AOqpTOE-	com.ea.game.pvz2_row	deepali Jadhav	<a href="https://lh4.googleusercontent.com">https://lh4.googleusercontent.com</a>	April 25, 2017	<a href="https://play.google.com/">https://play.google.com/</a>	5		Best game for our minds.
gp:AOqpTOE-	com.ea.game.pvz2_row	Sweden Alvarez	<a href="https://lh4.googleusercontent.com">https://lh4.googleusercontent.com</a>	May 4, 2017	<a href="https://play.google.com/">https://play.google.com/</a>	4		It was a very nice game. More challenges.
gp:AOqpTOE-	com.ea.game.pvz2_row	pl mAge	<a href="https://lh4.googleusercontent.com">https://lh4.googleusercontent.com</a>	April 6, 2017	<a href="https://play.google.com/">https://play.google.com/</a>	4		Everything in this update is awesome specially the new purchasable plants that costs a hundred gems but I just noticed that Sun Bean is not doing it's thing. I think he better get fixed. Also, can I have my Peashooter? It's been a long time without a basic Peashooter.
gp:AOqpTOE-	com.ea.game.pvz2_row		<a href="https://lh4.googleusercontent.com">https://lh4.googleusercontent.com</a>	April 10, 2017	<a href="https://play.google.com/">https://play.google.com/</a>	5		For some reason I'm not getting gem jam 3 or 4 but I have the update could you fix that.
gp:AOqpTOE-	com.ea.game.pvz2_row	Bernard Bell Romarate	<a href="https://lh4.googleusercontent.com">https://lh4.googleusercontent.com</a>	April 28, 2017	<a href="https://play.google.com/">https://play.google.com/</a>	5		great.
gp:AOqpTOE-	com.ea.game.pvz2_row		<a href="https://lh4.googleusercontent.com">https://lh4.googleusercontent.com</a>	April 26, 2017	<a href="https://play.google.com/">https://play.google.com/</a>	5		Great....

## LAMPIRAN 2 : CONSENT FORM

### USABILITY TESTING

#### **Pendahuluan**

Pengujian ini dilakukan untuk melakukan penilaian terhadap kegunaan (usabilitas) dari prototipe yang akan dibuat pada penelitian ini.

Prototipe aplikasi ini bertujuan untuk memaparkan pengalaman penggunaan produk berdasarkan ulasan pelanggan. Studi kasus yang digunakan pada penelitian ini adalah aplikasi *Plant Vs Zombies 2* pada *platform* Google Play. Metode yang digunakan adalah *text chunking* dengan memanfaatkan *library* dari Apache OpenNLP.

Penelitian ini mengadopsi algoritma yang digunakan pada penelitian yang telah dilakukan sebelumnya, yaitu *Extracing Verb Expressions Implying Negative Opinions* oleh Huayi Li untuk membuat prototipe *text chunking* pada ulasan pelanggan. Pengembangan prototipe ekstraksi ekspresi kata kerja ini mempunyai tujuan sebagai berikut :

1. Membuat prototipe yang mampu mengekstraksi ekspresi kata kerja yang dapat dimanfaatkan untuk mempermudah pengguna dalam memahami ulasan pelanggan terkait dengan pengalaman penggunaan produk.
2. Mempersingkat waktu pengguna dalam pemilihan produk yang diinginkan karena dengan prototipe ini dapat membantu pengguna sehingga hanya membutuhkan waktu singkat untuk memahami ulasan pengguna.
3. Memperkaya informasi yang diberikan oleh e-commerce terkait dengan produk atau jasa sehingga dapat memberikan nilai tambah bagi e-commerce tersebut.
4. Mampu memberikan tambahan fitur yang dapat membantu pelanggan e-commerce untuk menampilkan

informasi mengenai aktivitas yang dilakukan pada saat penggunaan produk.

*Usability Testing* ini akan memakan waktu sekitar 10 menit dan media yang digunakan adalah paper prototipe sebagai gambaran dari hasil akhir dari rancang bangun prototipe yang akan dibuat. Seluruh fungsi dari prototipe yang dibuat berdasarkan literatur dan penelitian sebelumnya yang terkait dengan penelitian ini.

Terdapat 3 Bagian pertanyaan yang harus dijawab. Pada bagian pertama menggunakan tipe soal likert dengan skala 1 – 5 yang bertujuan mengukur usabilitas dari aplikasi. Pada bagian kedua menggunakan tipe soal bebas, untuk mengambil evaluasi dan saran untuk pengembangan aplikasi.

*Terimakasih atas partisipasi anda*

### **FORM PERSETUJUAN**

*Saya mengerti bahwa partisipasi dalam studi kegunaan ini bersifat sukarela. Silahkan isi form berikut untuk menunjukkan bahwa Anda telah membaca dan Anda memahami informasi pada formulir ini dan bahwa setiap pertanyaan yang mungkin Anda miliki tentang sesi telah dijawab.*

*Tanggal* : \_\_\_\_\_

*Nama Lengkap* : \_\_\_\_\_

*Jenis Kelamin* : \_\_\_\_\_

*Kemampuan Bahasa Inggris* : \_\_\_\_\_

*Lama Menggunakan Aplikasi* : \_\_\_\_\_

*Tanda Tangan* : \_\_\_\_\_

### LAMPIRAN 3: SYSTEM USABILITY SCALE FORM

**Intruksi** : Isikan setiap jawaban pertanyaan dengan memberikan tanda *checklist* (✓) atau *cross* (x) pada salah satu kolom 1-5 yang menurut Anda paling sesuai dengan pendapat anda.

1 = Sangat tidak setuju

5 = Sangat setuju

Pernyataan	1	2	3	4	5
Saya merasa bahwa saya informasi mengenai ulasan pelanggan yang terdapat pada sistem ini berguna					
Saya berpikir sistem ini sangat rumit dan tidak perlu					
Saya berpikir bahwa sistem ini mudah digunakan					
Saya berpikir saya akan membutuhkan bantuan seseorang untuk dapat menggunakan sistem ini					
Saya merasa bahwa informasi yang terdapat dalam sistem ini memberikan pengetahuan yang lebih terkait dengan ulasan pelanggan					
Saya berpikir terlalu banyak inkonsistensi terkait dengan informasi yang ditampilkan dari sistem ini					
Saya merasa banyak orang akan memahami informasi yang dimuat dalam sistem ini secara efektif					
Saya merasa informasi yang dimuat dalam sistem ini sulit untuk dipahami					
Saya merasa sangat percaya diri dalam menggunakan sistem ini					
Saya perlu belajar banyak hal sebelum dapat menggunakan sistem ini					

Bagian II Evaluasi

1. Apakah informasi mengenai ulasan pelanggan telah cukup menggambarkan aplikasi Plant vs Zombies 2? Jelaskan

---

---

---

2. Adakah informasi selain ulasan pelanggan yang menurut anda perlu ditambahkan agar dapat menggambarkan aplikasi Plant vs Zombies 2 lebih mudah?

---

---

---

3. Saran anda untuk perbaikan kedepannya?

---

---

---