

## Daftar Istilah

Dataset	Kumpulan data
Fitur ciri	Atribut atau masukan klasifikasi
Fitur produk	Suatu aspek produk, target dari sebuah sentimen dalam analisis sentimen. Atau bisa disebut juga ciri dari suatu produk tertentu
Polaritas	Orientasi fitur produk berupa positif maupun negatif

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Data ulasan suatu produk sangat penting untuk konsumen yang ingin membeli sebuah produk dan untuk produsen yang ingin melihat tanggapan konsumen. Konsumen dan produsen tidak lagi terbatas meminta teman atau keluarga berpendapat dalam membeli ataupun meminta tanggapan suatu produk, karena sudah banyaknya ulasan yang dikemukakan di forum publik web dan jejaring sosial. Banyak konsumen dan produsen yang ingin tahu apakah ulasan produk tersebut bersifat negatif atau positif. Maka diperlukan analisis sentimen untuk mendeteksi secara otomatis label positif dan negatif dari ulasan produk dan menjadikan ringkasan. Ringkasan tersebut dapat memudahkan konsumen menyaring informasi dan dapat membantu mengambil keputusan dalam membeli sebuah produk maupun memudahkan produsen untuk melihat tanggapan konsumen dan menarik kesimpulan dari produk.

Pada penelitian analisis sentimen yang telah dilakukan pada tugas akhir sebelumnya [8], khususnya pada bagian klasifikasi masih dapat untuk dikembangkan. dengan menambahkan beberapa fitur ciri seperti semantik dan negasi yang dapat menambah informasi *classifier*. Pada tugas akhir sebelumnya [8] masih belum terdapat informasi fitur ciri semantik, sehingga sulit mendapatkan informasi fitur produk yang memiliki makna semantik, sehingga masih terdapat fitur produk yang khususnya memiliki makna semantik masih sulit dikenali yang menyebabkan *classifier* tidak dapat mengklasifikasi fitur produk semantik. Selanjutnya, masih belum terdapat informasi fitur ciri bernegasi yang dapat membalikan orientasi fitur produk yaitu dari positif ke negatif atau dari negatif ke positif sehingga *classifier* masih sulit untuk mengkategorisasikan fitur produk yang bernegasi dan dapat menyebabkan kesalahan kategorisasi.

Pada tugas akhir ini menawarkan solusi untuk menangani masalah yang dipaparkan di atas yaitu dengan menggunakan pendekatan *Semantic role labeling* dan *Negation handling*. *Semantic role labeling* bertujuan untuk mengidentifikasi

hubungan semantik pada fitur produk yang memiliki makna implisit yang digunakan sebagai salah satu fitur ciri klasifikasi. *Negation handling* bertujuan untuk mengidentifikasi kata yang bernegasi yang digunakan sebagai salah satu fitur ciri klasifikasi. Penambahan fitur ciri *Semantic role labeling* dan *Negation handling* pada algoritma *Random forest* bertujuan untuk menambahkan informasi pemodelan pada klasifikasi agar algoritma *random forest* dapat lebih mudah mengklasifikasikan fitur produk yang memiliki makna semantik dan negasi. Algoritma *random forest* memiliki akurasi baik pada analisis sentimen jika menggunakan karakteristik data unigram [21] seperti yang digunakan pada tugas akhir ini sebagai fitur produk. Maka diharapkan hasil klasifikasi ini dapat lebih akurat agar nantinya dapat dijadikan ringkasan yang lebih baik untuk meningkatkan ketepatan dalam penarikan kesimpulan dari ulasan produk.

## 1.2 Rumusan Masalah

Perumusan masalah berfokus kepada beberapa hal yaitu sebagai berikut:

1. Bagaimana mengekstrak informasi dari *semantic role labeling* dan *negation handling* agar dapat dijadikan fitur ciri pada klasifikasi?
2. Bagaimana akurasi klasifikasi setelah diterapkan *semantic role labeling* dan *negation handling*?

Adapun batasan masalah pada tugas akhir ini yaitu sebagai berikut:

1. Bahasa pemrograman yang digunakan adalah bahasa pemrograman java.
2. Dataset yang digunakan adalah ulasan produk berbahasa inggris dari data ulasan amazon.com yang digunakan pada penelitian Minqing Hu dan Bing Liu.
3. Hanya berfokus pada pengembangan klasifikasi fitur produk dari Tugas Akhir sebelumnya [8] yang berjudul “Analisis Sentimen dan Peringkasan Opini pada Ulasan Produk Menggunakan Algoritma Random Forest”.
4. Pada penerapan metode *semantic role labeling* digunakan *library Clear NLP* dan pada penerapan Algoritma *Random forest* digunakan *library Weka*.

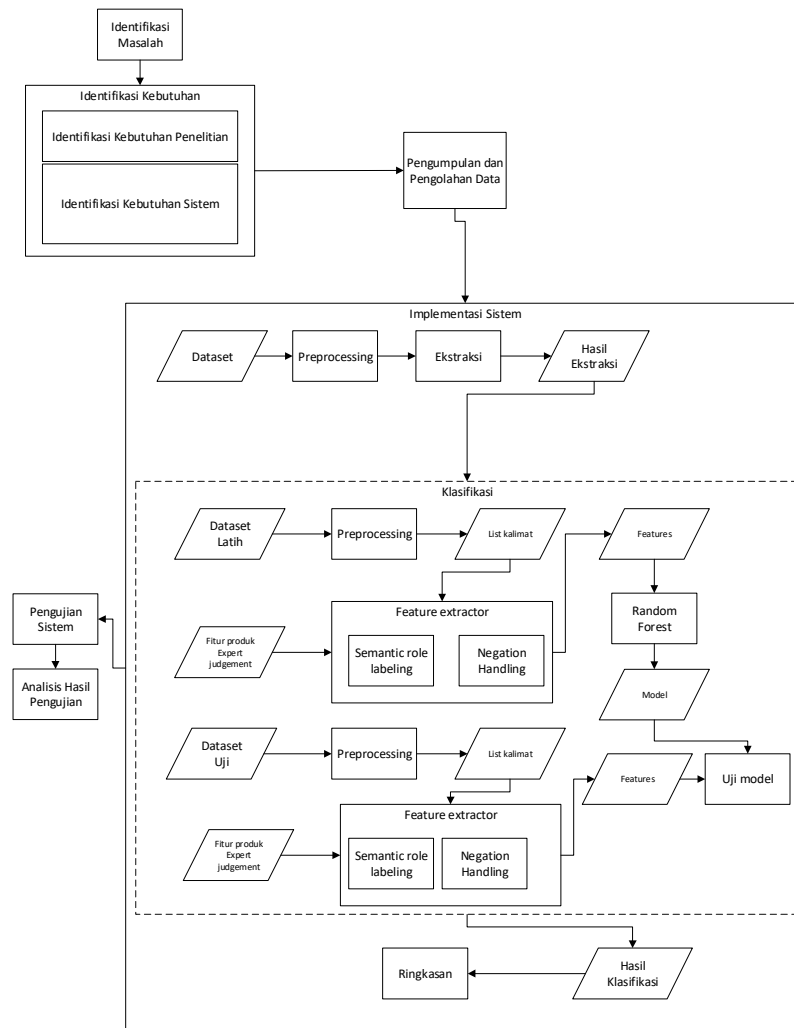
### 1.3 Tujuan

Terdapat beberapa tujuan dari tugas akhir ini yaitu sebagai berikut:

1. Menganalisis akurasi klasifikasi fitur produk setelah menggunakan semua fitur ciri.
2. Menganalisis akurasi klasifikasi setelah menambahkan fitur ciri dari *Negation handling* dan *semantic role labeling*.
3. Menganalisis pengaruh dataset yang memiliki karakteristik yang berbeda-beda terhadap *negation handling* dan *semantic role labeling*.

### 1.4 Metodologi Penyelesaian Masalah

Metodologi penyelesaian masalah dalam penelitian ini adalah:



Gambar 1- 1 Metodologi penelitian tugas akhir

#### **1.4.1 Identifikasi Masalah**

Masalah pada Tugas Akhir sebelumnya[8] seperti yang sudah dipaparkan pada latar belakang yaitu masih belum terdapat informasi hubungan semantik dari fitur produk belum terdapat informasi dari fitur produk yang bernegasi yang bisa menyebabkan kesalahan dalam menentukan orientasi dari fitur produk.

#### **1.4.2 Identifikasi Kebutuhan**

Terdapat dua kebutuhan pada tugas akhir ini yaitu:

##### **1.4.2.1 Identifikasi kebutuhan penelitian**

Identifikasi kebutuhan dari penelitian ini yaitu melakukan studi literatur dengan mempelajari tugas akhir sebelumnya[8], dari paper dan buku terkait yang dapat menangani masalah yang ditemukan pada tugas akhir yang sebelumnya[8]. Studi literatur juga dilakukan dengan dosen pembimbing dan dosen di kelas yang berhubungan dengan tugas akhir ini.

##### **1.4.2.2 Identifikasi kebutuhan sistem**

Identifikasi kebutuhan sistem adalah kebutuhan akan penggunaan metode dalam menyelesaikan masalah. Metode yang digunakan pada penelitian ini yaitu *Semantic role labeling* dan *Negation handling*. *Semantic role labeling* bertujuan untuk mengidentifikasi hubungan semantik dari fitur produk. *Negation handling* bertujuan untuk mengidentifikasi kata yang bernegasi yang dapat merubah polaritas dari ulasan. *Semantic role labeling* dan *Negation handling* akan dijadikan sebagai fitur ciri klasifikasi yang bertujuan untuk menambah informasi dari *classifier*.

#### **1.4.3 Pengumpulan dan Pengolahan Data**

Data yang digunakan dalam tugas akhir ini adalah melanjutkan dataset yang digunakan pada tugas akhir sebelumnya[8] yang diambil dari ulasan produk yang terdapat pada situs amazon.com yang digunakan pada penelitian Minqing Hu dan Bing Liu. Format data yang digunakan merupakan *file txt* yang berupa kalimat *review* beserta label fitur dan opini untuk setiap kalimatnya. Ulasan produk yang akan digunakan dalam tugas akhir ini terdapat 5 jenis , yaitu Apex DVD Player, Canon G3, Nikon Coolpix 4300, Nokia 6610, Zen Mp3 Player.

#### **1.4.4 Implementasi Sistem**

Implementasi sistem pada Tugas Akhir ini yaitu melakukan perancangan sistem yang akan di paparkan pada bab 3. Implementasi meliputi dalam pengembangan klasifikasi dari tugas akhirnya sebelumnya[8] dengan menggunakan beberapa metode yang diusulkan. Pada tugas akhir ini di fokuskan pada proses klasifikasi dan akan mengembangkan dari proses klasifikasi sebelumnya[8]. Pada tahap awal yaitu sistem membaca dataset dan melakukan pengekstrakan fitur produk dengan *expert judgment* dan dilakukan pengekstrakan label dari fitur produk. Dataset yang sudah di *tokenize* dijadikan sebagai inputan *semantic role labeling*. *Semantic role labeling* untuk mengidentifikasi hubungan semantik dari setiap kata. *Dependency parser* yang digunakan dari *semantic role labeling* akan digunakan juga pada *Negation handling* untuk mengidentifikasi kata yang bernegasi. Setelah melakukan beberapa metode tersebut akan diekstrak informasi dari setiap metode yang dapat dijadikan sebagai fitur ciri klasifikasi yang akan di proses oleh algoritma *random forest*.

#### **1.4.5 Pengujian Sistem**

Pengujian yang dilakukan yaitu dengan menguji akurasi setiap atribut pada klasifikasi dan pengujian terhadap akurasi dari klasifikasi setelah menggunakan beberapa metode meliputi *semantic role labeling* dan *negation handling*. Pengujian juga dilakukan pada semua dataset untuk mengetahui akurasi masing-masing dari klasifikasi.

#### **1.4.6 Analisis Pengujian Sistem**

Analisis yang dilakukan meliputi dari hasil pengujian yang dilakukan yaitu menganalisis pengaruh dari setiap atribut yang digunakan dan menganalisis hasil setelah diterapkan *semantic role labeling* dan *Negation Handling* sebagai fitur ciri klasifikasi. Menganalisis pengaruh dari dataset yang memiliki karakteristik yang berbeda-beda.

## 1.5 Sistematika Penulisan

Tugas akhir yang berjudul “Pengembangan Klasifikasi Opini Pada Ulasan Produk Menggunakan Algoritma *Random Forest*” disusun dengan sistematika penulisan sebagai berikut:

1. Pendahuluan

Pada Bab 1, menjelaskan latar belakang penulisan dan masalah yang diangkat, perumusan masalah, tujuan, batasan masalah, metologi penyelesaian masalah serta sistematika penulisan dari tugas akhir ini.

2. Tinjauan Pustaka

Pada Bab 2, menjelaskan landasan teori yang mendukung dan terkait dalam tugas akhir ini.

3. Perancangan Sistem

Pada Bab 3, berisikan gambaran umum sistem yang akan dibangun, langkah-langkah pengumpulan data, tahapan setiap proses dalam *semantic role labeling*, *negation handling* dan klasifikasi *random forest*.

4. Pengujian dan Analisis

Pada Bab 4, memuat hasil analisis terhadap data yang sudah dilakukan klasifikasi dengan melakukan perhitungan berupa akurasi.

5. Kesimpulan dan Saran

Pada Bab 5, menjelaskan kesimpulan hasil dari penelitian yang sudah dilakukan dan saran untuk pengembang penelitian ini.

## **BAB II**

### **KAJIAN PUSTAKA**

#### **2.1 Analisis sentimen**

Analisis sentimen adalah proses pengolahan data tekstual secara otomatis untuk mendeteksi apakah segmen teks berisi konten positif atau negatif. Hasil dari proses pengolahan teks nantinya dapat di jadikan ringkasan untuk mendapatkan informasi sentimen yang terkandung dalam teks. Banyaknya ulasan yang menyebabkan konsumen dan produsen sulit menyaring informasi yang terkandung dalam ulasan tersebut sehingga tidak dapat menarik kesimpulan dari ulasan. Maka dibutuhkan analisis sentimen untuk menyaring informasi teks. Salah satu contoh analisis sentimen dalam dunia nyata adalah identifikasi kecenderungan opini terhadap sebuah opini fitur produk, apakah fitur produk cenderung beropini negatif atau positif [9].

#### **2.2 Perbedaan level dari analisis sentimen**

Secara umum, analisis sentimen telah diteliti terutama pada tiga tingkatan yaitu:

##### **2.2.1 Tingkat Dokumen**

Pada tingkat dokumen bertugas untuk mengklasifikasikan apakah sebuah dokumen opininya mengungkapkan sentimen positif atau negatif. Pada tingkat dokumen biasanya diambil contoh pada ulasan sebuah produk untuk menentukan konten secara menyeluruh pada dokumen apakah bersifat negatif atau positif. Misalnya, pada ulasan produk, sistem akan menentukan apakah ulasan mengungkapkan pendapat positif atau negatif secara keseluruhan pada dokumen [9].

##### **2.2.2 Tingkat kalimat**

Pada tingkat kalimat bertugas untuk menentukan apakah setiap kalimat menyatakan pendapat positif atau negatif. Pada level kalimat yang paling populer adalah pada twitter [9].

##### **2.2.3 Tingkat aspek dan Entitas**

Pada tingkat dokumen dan tingkat kalimat tidak dapat menemukan secara maksimal aspek apa yang sebenarnya terkandung dalam kalimat. Contohnya pada “layanan



restoran ini tidak baik namun saya tetap menyukai restoran ini”. Maka, tidak dapat disimpulkan bahwa opini tersebut adalah positif karena masih terdapat entitas yang negatif. Tingkat aspek biasanya disebut tingkat fitur (*feature based opinion mining and summarization*). Maka level aspek bisa disebut batasannya lebih kecil dari *level* kalimat dan dokumen karena bisa saja dalam satu kalimat memiliki beberapa aspek. Dengan demikian, tujuan dari tingkat aspek adalah untuk menemukan entitas atau aspek pada ulasan [9].

### 2.3 Part Of Speach Tagging

*Part Of Speach* Tagging (POS tagging) adalah proses memberi label pada setiap kata dalam sebuah kalimat dengan POS atau *tag* yang sesuai dengan gambaran kalimat tersebut. *POS Tagger* adalah perangkat lunak yang membaca teks dengan beberapa bahasa dan memberikan tanda untuk setiap kata, seperti kata benda, kata kerja, kata sifat dll [17]. Adapun contoh dari *POS Tagging* sebagai berikut:

Tabel 2- 1 Part Of Speach Tagging [17]

Tag	Deskripsi	Contoh
CC	<i>Coordin, conjunction</i>	<i>And, but, or</i>
CD	<i>Cardinal number</i>	<i>One, two</i>
DT	<i>Determiner</i>	<i>A, the</i>
EX	<i>Existensial 'there'</i>	<i>There</i>
FW	<i>Foreign word</i>	<i>Mea culpa</i>
IN	<i>Preposition/sub-conj</i>	<i>Of, in, by</i>
JJ	<i>Adjective</i>	<i>Yellow</i>
JJR	<i>Adj., comparative</i>	<i>Bigger</i>
JJS	<i>Adj., superlative</i>	<i>Wildest</i>
LS	<i>List item marker</i>	<i>1, 2, one</i>
MD	<i>Modal</i>	<i>Can, should</i>
NN	<i>Noun, sing or mass</i>	<i>Llama</i>
NNS	<i>Noun, plural</i>	<i>Llamas</i>
NNP	<i>Proper noun, sing</i>	<i>IBM</i>
NNPS	<i>Proper noun, plural</i>	<i>Carolinas</i>
PDT	<i>Predeterminer</i>	<i>Ll,both</i>
POS	<i>Possessive ending</i>	<i>S</i>
PRP	<i>Personal pronoun</i>	<i>I, you, he</i>
PRPS	<i>Possessive pronoun</i>	<i>your, one's</i>
RB	<i>Adverb</i>	<i>Quick, never</i>
RBR	<i>Adverb, comperative</i>	<i>Faster</i>
RBS	<i>Adverb, superlative</i>	<i>Fastest</i>
RP	<i>Particle</i>	<i>Up, off</i>
SYM	<i>Symbol</i>	<i>%, &amp;</i>
TO	<i>"to"</i>	<i>To</i>
UH	<i>Interjection</i>	<i>Ah,oops</i>
VB	<i>Verb base form</i>	<i>Eat</i>
VBD	<i>Verb past tense</i>	<i>Ate</i>

Tag	Deskripsi	Contoh
<b>VBG</b>	<i>Verb gerund</i>	<i>Eating</i>
<b>VBN</b>	<i>Verb past participle</i>	<i>Eaten</i>
<b>VBP</b>	<i>verb non-3sg pres</i>	<i>Eat</i>
<b>VBZ</b>	<i>Verb 3sg pres</i>	<i>Eats</i>
<b>WDT</b>	<i>Wh-determiner</i>	<i>Which, that</i>
<b>WP</b>	<i>Wh-pronoun</i>	<i>What, who</i>
<b>WPS</b>	<i>Possessive wh-</i>	<i>Whose</i>
<b>WRB</b>	<i>Wh-adverb</i>	<i>How, where</i>

## 2.4 Lemmatization

*Lemmatization* adalah proses untuk mengembalikan kata ke bentuk kata dasarnya. Proses *lemmatization* bertujuan melakukan normalisasi pada teks atau kata dengan berdasarkan pada bentuk dasar kata tersebut.

Adapun contoh *lemmatization* sebagai berikut:

<p><b>Kalimat:</b></p> <p>They fired away and the picture turned out quite nicely as all of my pictures have thusfar</p> <p><b>Hasil setelah di lemmatization:</b></p> <p>They fire away and the picture turn out quite nicely as all of my picture have thusfar</p>
--

Gambar 2- 1 Contoh Lemmatization

## 2.5 Stop word Removal

*Stop word removal* adalah proses menghilangkan kata yang dianggap tidak memiliki makna dalam sebuah kalimat. *Stop word removal* bertujuan untuk mengurangi *noise* pada suatu kalimat.

Adapun contoh *stopword* sebagai berikut:

<p><b>Kalimat:</b></p> <p>They fired away and the picture turned out quite nicely as all of my pictures have thusfar</p> <p><b>Hasil Setelah di stop word removal:</b></p> <p>fired picture turned nicely pictures thusfar</p>
--

Gambar 2- 2 Contoh Stop word removal

## 2.6 Supervised Learning

*Supervised learning* adalah suatu teknik pembelajaran terawasi dimana jika *output* yang diharapkan telah diketahui sebelumnya. Biasanya data yang digunakan pada pembelajaran ini menggunakan data yang sudah ada. Pada *supervised learning* biasanya data sudah di ketahui atribut dan labelnya, dari atribut dan label tersebut maka bisa membuat sebuah model. Model tersebut akan digunakan untuk menguji data testing, agar nantinya dapat menentukan pola dari klasifikasi. Misalnya pada sebuah penelitian, setiap pola yang diberikan kedalam metode yang telah diketahui *output*nya. Satu pola akan di inputkan, pola ini akan di proses oleh sebuah metode dari *supervised learning*. Lapisan *training* akan membangkitkan pola *output* yang nantinya akan dicocokkan dengan pola *output* targetnya. Apabila terjadi perbedaan antara pola output hasil pembelajaran dengan pola *output* target, maka akan muncul *error*. Jika nilai *error* ini masih cukup besar, itu berarti masih perlu dilakukan pembelajaran yang lebih lanjut. Klasifikasi sentimen biasanya dirumuskan sebagai masalah klasifikasi pelabelan positif dan negatif. Pelatihan dan pengujian data yang biasa digunakan adalah opini fitur produk. Adapun metode klasifikasi dari *supervised learning* adalah *Naive Bayes*, *SVM*, *Decision Tree*, *Random forest* dll [19].

## 2.7 Negation Handling

Penanganan negasi dapat menjadi perhatian dalam sentiment analysis. Pada *bag-of-word* pernyataan “*I like this book*” dan “*I don’t like this book*” dianggap sangat mirip dilihat dari kata yang digunakan namun yang membedakan adalah kata “*don’t*” yang berarti negasi yang menyebabkan kalimat memiliki makna yang terbalik [11]. Pada penelitian [11] menggunakan kata-kata negasi seperti *not*, *no*, *never*, *none*, *nowhere*, *don’t*, *didn’t*, *cann’t* dll. Negasi tersebut adalah jenis yang paling umum dari *explicit negation* yang berfungsi untuk membalikan makna kalimat. Negasi perlu mengidentifikasi ruang lingkupnya, negasi dapat bersifat lokal seperti “*not good*” dan dapat berupa kata yang memiliki hubungan jauh seperti “*does not look very good*” ataupun negasi dari subjeknya seperti “*no one thinks that it’s good*”. Ungkapan negasi dalam sebuah kalimat sangat bervariasi. Ini bisa berupa kata kerja, kata keterangan, akhiran atau awalan. Negasi juga dapat muncul lebih dari satu kali dalam sebuah kalimat. Oleh karena itu banyak pendekatan yang

dapat digunakan untuk mengidentifikasi negasi salah satunya *dependency parser* yang digunakan pada tugas akhir ini.

## 2.8 Dependency Parser

Dependency parser adalah proses menganalisis struktur gramatikal sebuah kalimat, kata-kata yang menyusun kalimat tersebut membangun hubungan antara kata-kata dengan dengan "*head*" dan kata-kata yang memodifikasi *head-head* tersebut. Gambar 2-3 menunjukkan penguraian ketergantungan kalimat. Panah dari kata yang bergerak ke kata lain menunjukkan bahwa kemana Bergeraknya relasi yang memodifikasi kata tersebut, dan label yang diberikan dengan panah menggambarkan sifat sebenarnya dari ketergantungan tersebut [18].

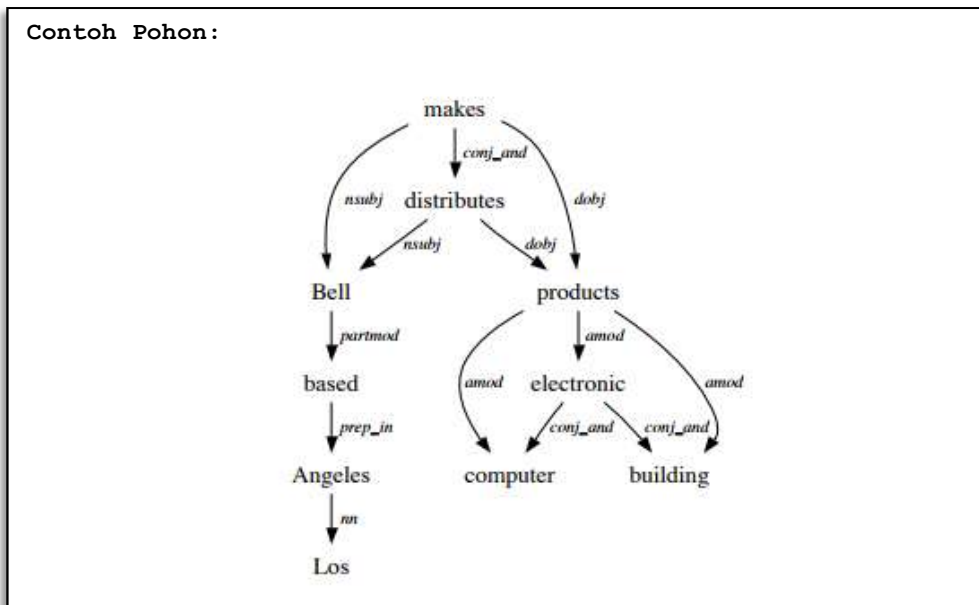
**Contoh Kalimat:**

Bell, based in Los Angeles, makes and distributes electronic, computer and building products.

**Hasil Dependency:**

```
nsubj(makes-8, Bell-1)
nsubj(distributes-10, Bell-1)
vmod(Bell-1, based-3)
nn(Angeles-6, Los-5)
prep in(based-3, Angeles-6)
root(ROOT-0, makes-8)
conj and(makes-8, distributes-10)
amod(products-16, electronic-11)
conj and(electronic-11, computer-13)
amod(products-16, computer-13)
conj and(electronic-11, building-15)
```

Gambar 2- 3 Contoh dependency parser



Gambar 2- 4 Contoh pohon dependency parser

Adapun deskripsi dari setiap label beserta contohnya yaitu sebagai berikut:

Tabel 2- 2 Dependency parser [18]

Label	Deskripsi	Kalimat	Contoh
<b>Acomp</b>	adjectival complement	<i>She looks very beautiful</i>	acomp( <i>beautiful, looks</i> )
<b>Advcl</b>	adverbial clause modifier	<i>The accident happened as the night was falling</i>	advcl( <i>happened, falling</i> )
<b>Advmod</b>	adverb modifier	<i>Genetically modified food</i>	advmod( <i>modified, genetically</i> )
<b>Agent</b>	agent	<i>The man has been killed by the police</i>	agent( <i>killed, police</i> )
<b>Amod</b>	adjectival modifier	<i>Sam eats red meat</i>	amod( <i>meat, red</i> )
<b>Oppos</b>	appositional modifier	<i>Bill ( John 's cousin )</i>	appos ( <i>john 's cousin, Bill</i> )
<b>Aux</b>	Auxiliary	<i>He should leave</i>	aux( <i>should, leave</i> )
<b>Auxpass</b>	passive auxiliary	<i>Kennedy has been killed</i>	auxpass( <i>killed, been</i> )
<b>Cc</b>	coordination	<i>Bill is big and honest</i>	cc( <i>Big, and</i> )
<b>Ccomp</b>	clausal complement	<i>He says that you like to swim</i>	ccomp( <i>Says, like</i> )
<b>Conj</b>	conjunct	<i>Bill is big and honest</i>	conj( <i>big, honest</i> )
<b>Cop</b>	copula	<i>Bill is big</i>	cop( <i>big, is</i> )
<b>Csubj</b>	clausal subject	<i>What she said makes sense</i>	csubj( <i>makes, said</i> )
<b>Csubjpass</b>	clausal passive subject	<i>That she lied was suspected by everyone</i>	csubjpass( <i>suspected, lied</i> )
<b>Dep</b>	dependent	<i>Then, as if to show that he could, . . .</i>	dep( <i>show, if</i> )
<b>Det</b>	determiner	<i>The man is here</i>	det( <i>man, the</i> )

<b>Label</b>	<b>Deskripsi</b>	<b>Kalimat</b>	<b>Contoh</b>
<b>Discourse</b>	<i>discourse element</i>	<i>Iguazu is in Argentina :)</i>	discourse(is, :)
<b>Dobj</b>	<i>direct object</i>	<i>She gave me a raise</i>	dobj(gave, raise)
<b>Expl</b>	<i>expletive</i>	<i>There is a ghost in the room</i>	expl(is, There)
<b>Goeswith</b>	<i>goes with</i>	<i>They come here with out legal permission</i>	goeswith(with, out)
<b>Iobj</b>	<i>indirect object</i>	<i>She gave me a raise</i>	iobj(gave, me)
<b>Mark</b>	<i>marker</i>	<i>He says that you like to swim</i>	mark(swim, that)
<b>Mwe</b>	<i>multi-word expression</i>	<i>I like dogs as well as cats</i>	mwe(well, as)
<b>Neg</b>	<i>negation modifier</i>	<i>Bill is not a scientist</i>	neg(scientist, not)
<b>Nn</b>	<i>noun compound modifier</i>	<i>Oil price futures</i>	nn(futures, oil)
<b>Npadvmod</b>	<i>noun phrase as adverbial modifier</i>	<i>The director is 65 years old</i>	npadvmod(old, years)
<b>Nsubj</b>	<i>nominal subject</i>	<i>The baby is cute</i>	nsubj(cute, baby)
<b>Nsubjpass</b>	<i>passive nominal subject</i>	<i>Dole was defeated by Clinton</i>	nsubjpass(defeated, Dole)
<b>Num</b>	<i>numeric modifier</i>	<i>Sam ate 3 sheep</i>	num(sheep, 3)
<b>Number</b>	<i>element of compound number</i>	<i>I have four thousand sheep</i>	number(thousand, four)
<b>Parataxis</b>	<i>Parataxis</i>	<i>The guy, John said, left early in the morning</i>	parataxis(left, said)
<b>Pcomp</b>	<i>prepositional complement</i>	<i>They heard about you missing classes</i>	pcomp(about, missing)
<b>Pobj</b>	<i>object of a preposition</i>	<i>I sat on the chair</i>	pobj(on, chair)
<b>Poss</b>	<i>possession modifier</i>	<i>their offices</i>	poss(offices, their)
<b>Possessive</b>	<i>possessive modifier</i>	<i>Bill's clothes</i>	possessive(John, 's)
<b>Preconj</b>	<i>preconjunct</i>	<i>Both the boys and the girls are here</i>	preconj(boys, both)
<b>Predet</b>	<i>predeterminer</i>	<i>All the boys are here</i>	predet(boys, all)
<b>Prep</b>	<i>prepositional modifier</i>	<i>I saw a cat in a hat</i>	prep(cat, in)
<b>Prepc</b>	<i>prepositional clausal modifier</i>	<i>He purchased it without paying a premium</i>	prepc without(purchased, paying)
<b>Prt</b>	<i>phrasal verb particle</i>	<i>hey shut down the station</i>	prt(shut, down)
<b>Punct</b>	<i>Punctuation</i>	<i>Go home!</i>	punct(Go, !)
<b>Quantmod</b>	<i>quantifier phrase modifier</i>	<i>About 200 people came to the party</i>	quantmod(200, About)
<b>Remod</b>	<i>relative clause modifier</i>	<i>I saw the man you love</i>	remod(man, love)
<b>Ref</b>	<i>Referent</i>	<i>saw the book which you bought</i>	ref(book, which)
<b>Root</b>	<i>Root</i>	<i>I love French fries</i>	root(ROOT, love)
<b>Tmod</b>	<i>temporal modifier</i>	<i>Last night, I swam in the pool</i>	tmod(swam, night)
<b>Vmod</b>	<i>reduced non-finite verbal modifier</i>	<i>Points to establish are . . . ”</i>	vmod(points, establish)

Label	Deskripsi	Kalimat	Contoh
Xcomp	<i>open clausal complement</i>	<i>He says that you like to swim</i>	xcomp( <i>like, swim</i> )
Xsubj	<i>controlling subject</i>	<i>Tom likes to eat fish</i>	xsubj( <i>eat, Tom</i> )

## 2.9 Semantic Role Labeling

*Semantic role labeling* (SRL) bertujuan untuk menetapkan label secara otomatis untuk menemukan hubungan semantik yang tersirat dalam kalimat dan secara otomatis menemukan *semantic role* dari setiap ulasan masing-masing predikat dalam kalimat untuk konstituen antar kalimat. Predikat dapat berupa kategori leksikal misalkan *verb* dan *noun* yang membentuk struktur argumen predikatnya sendiri. Sebagai contoh dalam kalimat “*Andi sold his lunch to Bela in school yesterday*” menghasilkan identifikasi predikat “*sell*” yang melibatkan “*Andi*” sebagai “penjual” (*Agent*), “*Bela*” sebagai “pembeli” (*Recipient*), dan “*lunch*” sebagai “barang yang terjual” (*Theme*). Selain itu, “*school*” tersebut diidentifikasi sebagai “lokasi dimana acara penjualan berlangsung”, dan “*yesterday*” adalah “waktu acara jual beli terjadi”. Hubungan semacam itu (misalnya *agent, recipient*) disebut peran semantik [10]. Pendekatan *semantic role labeling* didasarkan pada *supervised machine learning*, sering menggunakan sumber daya FrameNet dan PropBank untuk menentukan apa yang dianggap sebagai predikat, dan mendefinisikan dataset *role* yang digunakan dalam training dari dataset yang di uji. Algoritma umumnya mulai dengan parsing kalimat dan kemudian secara otomatis menandai setiap node pohon *parsing* dengan peran semantis. PropBank adalah sumber kalimat yang menjelaskan peran semantik. Pada Propbank terkadang masih sulit dalam mendefenisikan universal dari tematik *role* maka dihubungkan *individual verb sense*. *Individual verb sense* adalah *role* semantik yang lebih spesifik. FrameNet menggunakan kesamaan *inferece* atau menggunakan kesamaan semantik dengan kata kerja yang berbeda dan kata kerja dengan kata benda [5]. Pada Tugas akhir ini digunakan Propbank sebagai *semantic role*. Representasi *arguments label* dan *modifier* dapat dilihat pada tabel 2-3 berikut:

Tabel 2- 3 Representasi Arguments labelling dan arguments modifier [5]

Arguments labelling		Arguments modifier	
V	<i>Verb</i>	ArgM-ADV	<i>Adverbial modification</i>
Arg0	<i>PROTO-AGENT, Subject</i>	ArgM-DIR	<i>Direction</i>
Arg1	<i>PROTO-PATIENT, Object</i>	ArgM-DIS	<i>Discourse marker</i>
Arg2	<i>Indirect object</i>	ArgM-EXT	<i>Extent marker</i>
Arg3	<i>Start point</i>	ArgM-LOC	<i>Location</i>
Arg4	<i>End point</i>	ArgM-MNR	<i>Manner</i>
Arg5	<i>Direction</i>	ArgM-MOD	<i>General modification</i>
R	<i>Reference</i>	ArgM-NEG	<i>Negation</i>
		ArgM-PRD	<i>Secondary predicate</i>
		ArgM-PRP	<i>Purpose</i>
		ArgM-REC	<i>Reciprocal</i>
		ArgM-TMP	<i>Temporal marker</i>

Contoh *Semantic role* dengan menggunakan PropBank pada kalimat:

<p><b>Kalimat 1</b> : Sales fell to \$25 million from \$27 million</p> <p>Hasil : [Arg1 Sales] fell [Arg4 to \$25 million] [Arg3 from \$27 million]</p> <p><b>Kalimat 2</b>: The average junk bond fell by 4.2%</p> <p>Hasil : [Arg1 The average junk bond] fell [Arg2 by 4.2%]</p> <p><b>fall.01</b></p> <p><b>Arg1</b>: Logical subject, patient, thing falling</p> <p><b>Arg2</b>: Extent, amount fallen</p> <p><b>Arg3</b>: start point</p> <p><b>Arg4</b>: end point, end state of arg1</p>
--

Gambar 2- 6 Contoh *Semantic role* [5]

## 2.10 Random Forest

*Random forest* adalah pengembangan dari *decision tree* dengan menggabungkan beberapa *decision tree*. Setiap *tree* ditanamkan dengan cara yang sama dan setiap *decision tree* telah dilakukan training menggunakan sampel individu dan setiap atribut dipecah pada *tree* yang dipilih antara atribut subset yang dipilih secara acak. Pada proses klasifikasi, individunya didasarkan pada *vote* dari suara terbanyak pada