NATURAL LANGUAGE PROCESSING (NLP) BAHASA INDONESIA SEBAGAI *PREPROCESSING* PADA *TEXT MINING*

Nur Indrawati
Departemen Teknik Informatika Institut Tinggi Teknologi Telkom
Jalan Telekomunikasi No. 1 Dayeuh Kolot Bandung 40257 Indonesia
nurindrawati_jogja@yahoo.co.id

Abstrak

Semakin besarnya volume berita elektronik berbahasa Indonesia mengakibatkan informasi tersedia dalam jumlah yang besar, beraneka ragam, dan pada umumnya tidak terstruktur. Hal ini mendorong terjadinya peningkatan kebutuhan untuk mencari dan mengelola informasi dengan baik sehingga dihasilkan pengetahuan yang bermanfaat.

Text mining memberikan solusi pada masalah-masalah dalam memproses, mengorganisasi, dan menganalisis *unstructured text* dalam jumlah besar. Dalam memberikan solusi, *text mining* mengadopsi dan mengembangkan teknik dan solusi dari bidang lain, salah satunya *Natural Language Processing*.

Natural language processing (NLP) digunakan sebagai *preprocessing* dalam *text mining* karena user menentukan relevansi dari dokumen dengan membaca dan menganalisisnya. Jika sistem dapat melakukan analisis dokumen secara otomatis, maka proses pencarian dokumen yang relevan akan lebih mudah. *Preprocessing* pada penelitian ini dibatasi hanya pada tiga modul atau tiga tahapan, yaitu: POS tagging, syntax parsing, dan semantic role labeling.

Dalam paper ini dibahas teori dan gambaran implementasi *semantic role labeling* berdasarkan teori *case grammar* dan melakukan studi perbandingan dengan pelabelan secara manual.

Kata kunci : *text mining, preprocessing, natural language processing, semantic role labeling, case grammar.*

1. Pendahuluan

1.1 Latar Belakang

Semakin besarnya volume berita elektronik berbahasa Indonesia mengakibatkan informasi tersedia dalam jumlah yang besar, beraneka ragam, dan pada umumnya tidak terstruktur. Hal ini mendorong terjadinya peningkatan kebutuhan untuk mencari dan mengelola informasi dengan baik sehingga dihasilkan pengetahuan yang bermanfaat.

Text mining merupakan upaya pencarian atau penambangan data yang berupa teks dimana sumber data biasanya diperoleh dari dokumen, dengan tujuan mencari kata-kata yang dapat mewakili isi dokumen sehingga dapat dilakukan analisis keterhubungan antar dokumen. *Text mining* memberikan solusi pada masalah-masalah dalam memproses, mengorganisasi, dan menganalisa *unstructured text* dalam jumlah besar. *Text mining* mengadopsi dan mengembangkan banyak teknik dan solusi dari bidang lain, seperti *Data Mining, Information Retrieval,* Statistik dan Matematik, *Machine Learning*, *Linguistic, Visualization*, dan *Natural Language Processing* [15].

Natural language processing (NLP) pada aplikasinya berkaitan dengan bagaimana komputer dapat digunakan untuk memahami dan memanipulasi teks bahasa alami *(natural language)* untuk mendapatkan informasi tertentu. Dengan perantaraan bahasa alami *(natural language)* inilah, manusia dapat berinteraksi dengan komputer. *Natural language processing (NLP)* digunakan dalam pemrosesan dokumen karena user menentukan relevansi dari dokumen dengan membaca dan menganalisisnya. Jika sistem dapat melakukan analisis dokumen secara otomatis, maka proses pencarian dokumen yang relevan akan lebih mudah [11].

Sebelum memasuki proses text mining, dokumen akan melewati tahapan preprocessing. Pada tahap ini, dokumen teks diproses dengan menggunakan pengetahuan umum mengenai bahasa alami *(natural language)*. *Preprocessing* pada peelitian ini dibatasi hanya pada tiga modul atau tiga tahapan *preprocessing*, yaitu: *POS tagging*, *syntax parsing*, dan *semantic role labeling*.

POS tagging bertujuan untuk memberi jabatan (POS) aturan grammar pada string atau kata dalam sebuah kalimat. *Syntax parsing* akan menunjukkan analisis sintaks pada kalimat berdasarkan pada teori grammar. *Semantic role labeling* digunakan untuk memberikan label pada kalimat, yang membedakan kontribusi suatu kata atau frasa terhadap *semantic role*-nya dalam sebuah kalimat.

Pada penelitian ini akan dianalisis dan diimplementasikan semantic role labeling berdasarkan pada teori case grammar dan melakukan studi perbandingan dengan pelabelan secara manual.

1.2 Tujuan

Tujuan yang hendak dicapai dalam penelitian ini adalah :

1. Mengimplementasikan *natural language processing* bahasa Indonesia sebagai *preprocessing* pada *text mining*, khususnya *semantic role labeling* pada kalimat berbahasa

Indonesia dengan menggunakan *case grammar* sebagai landasan teori.
2. Menganalisis hasil pelabelan kalimat dari *semantic role labeling* yang sudah diimplementasikan dan membandingkannya dengan hasil pelabelan secara manual.

1.3 Identifikasi Masalah

Objek pada penelitian ini adalah NLP bahasa Indonesia, yaitu pemrosesan kalimat berbahasa Indonesia yang digunakan sebagai *preprocessing* pada *text mining*. Dalam penelitian ini, lebih diutamakan dan difokuskan pada tahap *semantic role labeling.*

Batasan masalah pada penelitian ini adalah :

1. Dataset yang digunakan adalah artikel berita berbahasa Indonesia yang didapatkan dari web dan bersifat *offline*.
2. Pembahasan dalam penelitian ini hanya terletak pada bagaimana mengimplementasikan *semantic role labeling* pada kalimat tunggal yang diambil dari artikel berita berbahasa Indonesia pada poin (1) di atas.
3. Implementasi *semantic role labeling* didasarkan pada teori *case grammar*.
4. Input adalah kalimat baku yang diambil dari artikel berita berbahasa Indonesia. Jumlah kalimat yang diambil sebagai input adalah satu buah kalimat dan disimpan dalam file dengan format .log.
5. Output dari perangkat lunak adalah kalimat yang sudah diberi label sesuai dengan peran semantiknya.

1.4 Metode Penelitian

Metode yang digunakan dalam penyelesaian penelitian ini adalah :

1. Studi literatur
2. Pencarian dan pengumpulan data
3. Analisis kebutuhan dan implementasi sistem perangkat lunak
4. Pengujian sistem dan analisa hasil
5. Pengambilan keputusan dan penyusunan laporan penelitian.

2. *Text Mining*

Text mining memiliki definisi menambang data yang berupa teks di mana sumber data biasanya didapatkan dari dokumen, dan tujuannya adalah mencari kata - kata yang dapat mewakili isi dari dokumen sehingga dapat dilakukan analisa keterhubungan antar dokumen. *Text mining* juga merupakan proses penemuan akan informasi atau trend baru yang sebelumnya tidak terungkap dengan memproses dan menganalisa data dalam jumlah besar.

Dalam [15] disebutkan bahwa *text mining* bisa dianggap subjek riset yang tergolong baru. *Text mining* dapat memberikan solusi dari permasalahan seperti pemrosesan, pengorganisasian / pengelompokkan dan menganalisis *unstructured text* dalam jumlah besar [15].

Dalam memberikan solusi, *text mining* mengadopsi dan mengembangkan banyak teknik dari bidang lain, seperti *Data mining, Information Retrieval*, Statistik dan Matematik, *Machine Learning, Linguistic, Natural Languange Processing*, dan *Visualization*. Kegiatan penelitian untuk *text mining* antara lain ekstraksi dan penyimpanan teks, *preprocessing* akan konten teks, pengumpulan data statistik dan *indexing* dan analisis konten [15].

Dalam menganalisis sebagian atau keseluruhan *unstructured text*, *text mining* mencoba untuk mengasosiasikan satu bagian *text* dengan yang lainnya berdasarkan aturanaturan tertentu. Hasil yang di harapkan adalah informasi baru atau "insight" yang tidak terungkap jelas sebelumnya [15].

Permasalahan yang dihadapi pada *text mining* sama dengan permasalahan yang terdapat pada *data mining*, yaitu jumlah data yang besar, dimensi yang tinggi, data dan struktur yang terus berubah, dan data *noise.* Perbedaan di antara keduanya adalah pada data yang digunakan. Pada *data mining*, data yang digunakan adalah *structured data*, sedangkan pada *text mining*, data yang digunakan *text mining* pada umumnya adalah *unstructured data*, atau minimal *semistructured*. Hal ini menyebabkan adanya tantangan tambahan pada *text mining* yaitu struktur *text* yang komplek dan tidak lengkap, arti yang tidak jelas dan tidak standard, dan bahasa yang berbeda ditambah translasi yang tidak akurat [15].

Struktur data yang baik dapat memudahkan proses komputerisasi secara otomatis. Sehingga *text mining* perlu melakukan *preprocessing* yang lebih kompeks daripada *data mining* karena data yang diolah berbentuk *unstructured* [15].

2.1. Concept-Based Text Representation

Sebagian besar penelitian yang dilakukan terkait dengan *text mining* dalam kurun waktu 30 tahun didominasi oleh penggunaan metode statistikal untuk mencocokkan antara bahasa alami yang digunakan oleh *user* saat memasukkan *query* dengan data yang disimpan. Pendekatan *concept-based* merupakan solusi baru dalam *text mining*. [9]

Salah satu hal yang umum dalam riset *text mining* adalah dengan merepresentasikan teks sebagai kumpulan kata-kata atau lebih dikenal dengan istilah pendekatan *Bag-of-Words* (BoW). Kumpulan teks dalam suatu dokumen secara sederhana dianggap sebagai kumpulan kata-kata tanpa memperhatikan sintaksis yang merupakan salah satu cabang tata bahasa yang membicarakan struktur kalimat, klausa, dan frase. Hal ini berarti pada pendekatan BoW tidak mempedulikan urutan kata dalam dokumen karena kita hanya

memperhatikan suatu kata sebagai entitas tersendiri yang tidak dipengaruhi oleh struktur kalimat dimana kata itu berada. Pendekatan BoW juga tidak memperhatikan unsur semantik yang merupakan cabang linguistik yang membahas tentang arti dan makna, yang artinya sebuah kata diabaikan maknanya dan juga kedudukannya dalam sebuah dokumen secara semantik apakah berperan besar dalam mewakili isi dari suatu dokumen atau tidak.

Pendekatan BoW mengabaikan sintaksis dan semantik, jadi dalam BoW dokumen direpresentasikan sebagai vektor yang elemennya merupakan hasil pembobotan berdasarkan frekuensi kemunculan suatu kata dalam dokumen saja. Pendekatan BoW menggunakan analisa statistik dari frekuensi kemunculan term untuk mengetahui peran suatu kata dalam mewakili isi dokumen. Cara tersebut mempunyai kekurangan, salah satunya adalah bisa jadi dua buah kata mempunyai jumlah kemunculan yang sama dalam satu dokumen padahal kata yang satu lebih besar kontribusinya dalam mewakili isi dokumen dibandingkan dengan kata yang lain.

Untuk mengatasi kekurangan yang dialami oleh pendekatan BoW, maka muncul sebuah pendekatan baru yang dikenal dengan *Bag-of-Concepts* (BoC). Pendekatan BoC menangkap kata-kata yang merupakan konsep dari sebuah dokumen, yang nantinya dapat mewakili isi dokumen.

Sebuah kalimat yang lengkap dan berarti biasanya terdiri dari konsep utama yang memberikan gambaran tentang kandungan kalimat. Mendapatkan keterkaitan antara *verb* dan argumen-argumennya dalam sebuah kalimat mempunyai potensi yang menjanjikan dalam usaha kita untuk mengetahui arti dari sebuah kaimat. Informasi yang kita dapatkan tentang siapa sedang mengerjakan apa dan untuk siapa dapat menjelaskan besarnya kontribusi setiap *term* dalam suatu kalimat. Verba dan argumen-argumennya merupakan konsep seperti yang telah disebutkan sebelumnya. Kedudukan konsep ini mempunyai peranan yang penting dalam pengelompokkan sebuah dokumen karena konsep-konsep tersebut merepresentasikan semantik dari dokumen dan semantik merepresentasikan arti dari dokumen [9].

2.2 Concept-Based Mining Model [9]

Input dari c*oncept-based mining model* berupa sebuah dokumen berbentuk teks yang telah terlabeli sesuai dengan peran semantiknya, yang memperlihatkan struktur *verb-argument*. Pelabelan tersebut dilakukan berdasarkan peran semantik kata dalam kalimat, sehingga prosesnya disebut dengan *semantic role labeling*. Contoh sederhana *semantic role lebeling* adalah sebagai berikut:

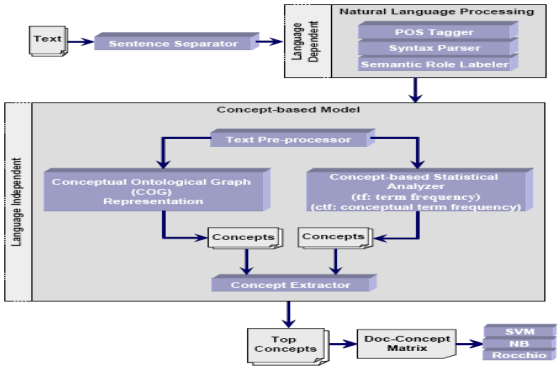
"Ibu membeli roti."

Pada kalimat di atas terdapat sebuah verba yaitu membeli dan argumen-argumennya yaitu "Ibu" dan "roti" . Pemberian label berdasarkan peran semantik dilakukan terhadap argumen dimana "Ibu" mendapatkan label subjek dan "roti" mendapat label objek, sehingga struktur *verb-argument* yang terbentuk dari kalimat tersebuat adalah sbb:

"[ARG0 Ibu] [TARGET membeli] [ARG1 roti]"

Pada c*oncept-based mining model*, baik verba maupun argumen-argumennya dianggap sebagai *term,* berbeda dari metode *text mining* tradisional yang menganggap sebuah kata sebagai *term*. *Concept-based mining model* terdiri dari tiga bagian utama yaitu: *graphical concept-based*, *statistical concept-based*, dan *concept extractor*.

Graphical concept-based menggambarkan struktur kalimat dengan tetap mempertahankan semantik kalimat sesuai dengan dokumen aslinya. Setiap konsep pada *graphical concept-based* diberi bobot berdasarkan posisi konsep pada struktur kalimat yang digambarkan. Sedangkan tujuan dari *statistical concept-based* adalah untuk memberikan bobot pada setiap konsep pada level kalimat dan dokumen. *Concept extractor* mengkombinasikan dua macam bobot yang dihasilkan oleh *graphical concept-based* dan *statistical concept-based* untuk dicari bobot konsep yang maksimal.



Gambar 2-1. *Concept-based Mining Model*

Pada gambar di atas, dapat dilihat bahwa sebelum memasuki sistem *concept-based model,* teks akan dilakukan *preprocessing* dengan *Natural Language Processing (NLP)* yang meliputi *POS tagging, syntax parsing*, dan *semantic role labeling.*

3. *Natural Language Processing* sebagai *Preprocessing* pada *Text Mining*

3.1 Definisi

Natural language atau bahasa alami adalah bahasa yang dapat dimengerti oleh manusia. Dalam [5] disebutkan bahwa menurut Peter Coxhead, 'Natural Language Processing' atau yang biasa disingkat NLP adalah semua usaha yang dilakukan untuk mengolah *natural language* menggunakan komputer. NLP

merupakan bagian dari *Artificial Intelligence* dan *Linguistic* yang bertujuan memahami sebuah atau beberapa kalimat deklaratif maupun kalimat tanya yang disampaikan secara lisan maupun tertulis.

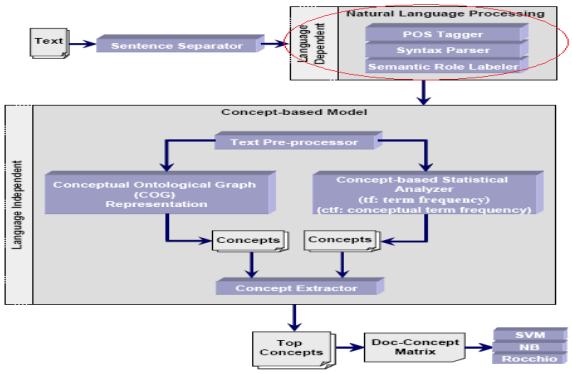
3.2 PPermasalahan dalam *Natural Language Processing*

Bahasa merupakan permasalahan rumit yang melibatkan proses seperti pengenalan suara atau tulisan cetak, parsing sintaks, inferensi semantik tingkat tinggi, dan bahkan komunikasi emosional melalui irama dan perubahan nada suara. Untuk mengatasi kompleksitas tersebut, ahli bahasa mendefinisikan beberapa tingkat dalam analisis *Natural Language* [12]:

1. **Prosody** berhubungan dengan irama dan intonasi bahasa. Level analisa ini sulit untuk dirumuskan dan sering diabaikan. **Phonology** menentukan jenis-jenis suara yang digabungkan untuk membentuk bahasa. Cabang bahasa ini berperan penting dalam speech recoginition dan speech generation.
2. **Morphology** membahas mengenai komponen (morfem) yang membentuk suatu kata. Meliputi aturan yang mengatur formasi kata, seperti pengaruh prefiks (un-,non-, anti-,etc.) dan sufiks (-ing,-ly,etc) dalam bahasa inggris yang merubah arti akar kata. Analisa Morfology sangat penting dalam menentukan peran sebuah kata dalam kalimat seperti tenses kalimat (lampau, sekarang, atau bentuk masa depan), jumlah (jamak atau tunggal) dan logat atau cara pengucapan.
3. **Syntax** mempelajari aturan-aturan yang merangkai kata menjadi frase dan kalimat baku dan penggunaan aturan tersebut untuk memparsing dan men-generate kalimat. Syntax merupakan level analisa bahasa yang paling baik dirumuskan dan yang paling berhasil diterapkan.
4. **Semantik** menentukan arti dari kata, frase dan kalimat dan cara bagaimana maksud atau artinya disampaikan dalam bahasa alami.
5. **Pragmatics** adalah pembelajaran mengenai bahasa mana yang digunakan dan pengaruhnya kepada pendengar. Sebagai contoh, pragmatics akan menyatakan bahwa “Yes”[ya] merupakan jawaban yang tidak lazim untuk pertanyaan “Do you know what time it is ?” [Apakah anda tahu sekarang jam berapa ?].
6. **World Knowledge** menyimpan pengetahuan dari dunia nyata, lingkungan interaksi sosial dan tujuan atau maksud sebuah komunikasi. Latar belakang pengetahuan umum ini sangat penting untuk memahami arit text atau percakapan secara keseluruhan.

Sesuai dengan definisi di atas, maka penelitian ini termasuk dalam analisis semantik pada *Natural Language Processing*.

3.3 *NLP* sebagai *Preprocessing* pada *Text Mining*



Natural language processing (NLP) digunakan dalam pemrosesan dokumen karena user menentukan relevansi dari dokumen dengan membaca dan menganalisisnya. Jika sistem dapat melakukan analisis dokumen secara otomatis, maka proses pencarian dokumen yang relevan akan lebih mudah.

Sebelum memasuki proses *text mining*, dokumen akan melewati tahapan *preprocessing*. Pada tahap ini, dokumen teks diproses dengan menggunakan pengetahuan umum mengenai bahasa alami *(natural language)*. *Preprocessing* pada penelitian ini dibatasi hanya pada tiga modul atau tiga tahapan, yaitu: *POS tagging, syntax parsing,* dan *semantic role labeling*.

Pada gambar di atas, dapat dilihat bahwa sebelum memasuki sistem concept-based model, teks akan dilakukan preprocessing dengan Natural *Language Processing (NLP)* yang meliputi *POS tagging, syntax parsing,* dan *semantic role labeling*.

3.3.1 *PPOS Tagging*

POS tagging bertujuan untuk memberi jabatan (POS) aturan grammar pada string atau kata dalam sebuah kalimat.

3.3.2 *SSyntax Parsing*

Syntax parsing akan menunjukkan analisis sintaks pada kalimat berdasarkan pada teori grammar.

3.3.2.1 *Syntax Parsing* dengan Menggunakan PC-PATR [5]

Parser merupakan *software* yang menganalisis input kalimat secara sintaktik. Setiap kata dan bagian–bagian ujarannya diidentifikasi. *Parser* kemudian membuat peta kata-kata dalam struktur yang dibuat pohon *parser*. Pohon *parser* menunjukkan makna dari semua kata dan bagaimana cara menggabungkan kata-kata tersebut. Sistem matematikal yang umum digunakan untuk memodelkan struktur konstituen pada bahasa alami adalah *context-free grammar*.

Context-free grammar disebut juga *phrase-structure grammar*. *Phrase-structure grammar* terdiri dari aturan/produksi (*phrase-structure rule*)

dan kamus kata (*lexicon*). Pada setiap *phrase-structure rule*, bagian sebelah kiri tanda panah (→) adalah sebuah simbol nonterminal, sedangkan bagian sebelah kanannya adalah urutan satu atau lebih simbol terminal atau nonterminal. Simbol nonterminal didefinisikan di *rule* itu, sedangkan simbol terminal digantikan di *lexicon*. Untuk memodelkan fenomena *context-free grammar* yang lebih kompleks, digunakan *constraint-based formalism*.

PC-PATR merupakan implementasi dari metode *constraint-based formalisms* pada *personal computer* (PC). PC-PATR ini dapat dijalankan pada beberapa sistem operasi seperti misalnya : MS-DOS, Microsoft Windows, Macintosh, dan Unix. *Constraint-based formalism* yang digunakan pada PC-PATR disebut dengan istilah PATR-II formalism.

Grammar PATR-II terdiri dari himpunan *rule* (aturan) dan *lexicon* (kamus kata). Setiap rule terdiri dari *phrase structure rule* dan himpunan *feature constraint*. *Feature constraint* adalah *unification* pada *feature structure* yang berhubungan dengan konstituen dari *phrase structure rule*. Sedangkan *lexicon* menyediakan *record-record* yang menggantikan simbol terminal pada *phrase structure rule*. Setiap record ini berisi kata beserta feature-nya.

3.3.2.2 File-file pada PC-PATR [5]

PC-PATR dijalankan dengan terlebih dahulu membaca (*load*) file grammar dan file lexicon-nya. Kemudian PC-PATR akan membaca kalimat masukan, menguraikannya berdasarkan *grammar* dan *lexicon*, lalu menghasilkan keluaran berupa pohon urai kalimat tersebut. Kalimat-kalimat masukan ini bisa dituliskan di dalam file.sen. Hasil keluarannya dapat disimpan dalam file.log.

File *grammar* adalah file yang berisi *phrase structure rule*, termasuk *feature constraint*-nya. Sebelum definisi *rule-rule*, pada file *grammar* dapat dibuat suatu *feature template*. *Feature template* digunakan sebagai makro di *lexicon*. *Feature template* dapat digunakan untuk memberikan nilai *default feature structure* terhadap suatu kategori. Contoh: Let N be <subcat> = !istilah , artinya kategori N didefinisikan memiliki subcat yang nilai default-nya adalah istilah. Efek dari template ini adalah membuat semua nomina (N) mempunyai subkategori istilah, kecuali didefinisikan lain. Jika tidak ada tanda seru (!) maka semua nomina didefinisikan mempunyai subkategori istilah, tanpa terkecuali.

Lexicon merupakan file basis data bentuk standar yang memiliki sejumlah *record*. Masing-masing *record* merepresentasikan kata. Setiap *record* dibagi atas beberapa *field*, dimana masing-masing *field* didahului oleh suatu penanda bentuk standar pada awal barisnya. Penanda ini didahului oleh karakter "\" dan diikuti oleh satu atau lebih karakter alfanumerik. Penanda yang digunakan dalam penelitian ini adalah:

\w, adalah penanda untuk bentuk leksikal kata, dimana pengejaannya sama persis dengan pengejaan kata yang akan dijadikan masukan PC-PATR.
\c, adalah penanda untuk kategori/kelas kata
\f, adalah penanda untuk ciri tambahan dari kata
Contoh isi file lexicon untuk kata menyiapkan:
\w menyiapkan
\c Vtran
\f <subcat> = ekatransitif

File *sentence* berisi kalimat masukan yang hendak diuraikan. Satu kalimat terletak di satu baris. Jika kalimat yang ingin diuraikan ada lebih dari satu, maka kalimat-kalimat tersebut harus disusun baris per baris. Kalimat yang sangat panjang tidak boleh dipisah menjadi 2 baris atau lebih, karena PC-PATR menguraikan file sentence ini baris demi baris.

File log adalah file keluaran. File ini berisi hasil penguraian file *sentence* terhadap *grammar* dan *lexicon* yang digunakan. Hasil penguraiannya digambarkan dengan pohon urai.

3.3.3 *SSemantic Role Labeling*

3.3.3.1 Definisi

Semantic role labeling merupakan proses pengidentifikasian argumen dari predikat dalam suatu kalimat, dan menentukan *semantic role* atau peran semantiknya [7].

3.3.3.2 Automatic Semantic Role Labeling

Mengidentifikasi *semantic role* dapat memberikan level analisis semantik yang bermanfaat dalam menyelesaikan *task natural language processing (NLP)*. *Semantic role* merepresentasikan partisipan dalam aksi atau keterhubungan yang digambarkan dengan *semantic frame*. Sebagai contoh, *frame* untuk kata kerja "crash", menyertakan peran AGENT, VEHICLE, dan TO-LOCATION.

Dalam penelitiannya[3], Daniel Gildea dan Daniel Jurafsky mencoba melakukan analisis semantik secara otomatis untuk menyelesaikan permasalahan pada *semantic role labeling*, dengan mengaplikasikan teknik statistik. Sistem yang dibangun berdasarkan pada *training classifier* pada *training set* yang sudah diberi label, dan untuk pengetesannya dilakukan pada *test set* yang belum berlabel.

Analisis semantik secara otomatis akan menentukan peran semantis atau *semantic role* dari unsur pokok dalam suatu kalimat. *Feature* yang digunakan dalam menentukan *semantic role* ada 5 *feature*, yaitu *phrase type*, *grammatical function*, *position*, *voice*, dan *head word* [3].

Teknik statistik yang diterapkan dalam [3] adalah *probabilistic parsing* dan *statistical classification*. Jadi, dalam *parsing* dan klasifikasi digunakan teknik statistik. Sedangkan data latih yang digunakan adalah FrameNet, yang merupakan *hand-labeled dataset*. *Database* FrameNet

mendefinisikan *tagset* dari *semantic role*, yang disebut sebagai *frame element* dan mengumpulkan sekitar 50.000 kalimat dari British National Corpus yang telah dilabeli (*hand-labeled*) dengan *frame element*-nya.

Automatic semantic role labeling dapat dibagi menjadi dua *subtasks* utama, yaitu (1) mengidentifikasi *boundaries* dari *frame element* dalam kalimat dan (2) melabeli setiap *frame element*, yang sudah diidentifikasi *boundaries*-nya, dengan *role* yang benar.

3.3.3.3 *Case Grammar* dan *Semantic Role Labeling* [1]

Pada pemrosesan bahasa alami atau *natural language processing (NLP)*, *semantic role labeling*, seperti *parsing* semantik pada umumnya, merupakan bidang yang sedang terus diteliti. *Semantic role labeling* digunakan dalam berbagai macam *task*, seperti tanya-jawab (*question-answering*), *information extraction*, dan *machine translation*. Permasalahan dalam *semantic role labeling* adalah menentukan relasi semantik antara kata kerja (*verb*) dengan bagian lain dalam suatu kalimat. Penentuan relasi semantik tersebut dilakukan dengan mengidentifikasi kata kerja *(verb)* dan argumennya, dan kemudian meng-*assign* setiap argumen dengan label semantik.

Salah satu teori yang dapat menyelesaikan permasalahan *semantic role labeling* adalah *case grammar*. Dalam tulisannya, Emil Albright [1] menjelaskan bahwa *case grammar* merupakan salah satu teori mengenai representasi peran semantik. *Case grammar* dapat mendeskripsikan relasi semantik.

Relasi semantik dalam *case grammar* dideskripsikan sebagai *case relation*, atau relasi antar kasus. Kasus dalam *case grammar* sendiri dapat mendeskripsikan relasi antara *verb* dan *noun* atau argumennya. Struktur yang berkaitan dengan makna dari kata kerja *(verb)* dideskripsikan dalam frame kasus *(case frame)*. Kata kerja akan diikuti dengan sekumpulan kasus yang mungkin, yang berkaitan dengannya.

Salah satu implementasi dari pengembangan teori representasi peran semantik adalah PropBank, yang menambahkan lapisan semantik pada Penn TreeBank[1]. Dalam PropBank, didefinisikan *role* mulai dari Arg0 hingga Arg5, serta ArgM-X untuk argumen yang spesifik. Misalnya, ArgM-LOC untuk argumen yang menerangkan lokasi, dan ArgM-TEMP untuk argumen waktu.

4 Tata Bahasa Kasus *(Case Grammar)*

Case grammar atau tata bahasa kasus merupakan suatu pendekatan terhadap tata bahasa yang memberi penekanan pada hubungan-hubungan semantik dalam suatu kalimat[11]. Dalam tata bahasa kasus, kata kerja dianggap sebagai bagian kalimat yang paling penting, dan memiliki sejumlah hubungan semantik dengan satu atau lebih frasa nomina (kata benda). Hubungan-hubungan inilah yang disebut kasus [13]. Kata kerja dipandang sebagai suatu peristiwa atau event dan kasus merupakan peran frasa nomina pada peristiwa tersebut. Tata bahasa kasus atau *case grammar* memelihara serta mempertahankan suatu pembedaan antara struktur dalam *(semantik deep)* dan struktur permukaan dari tata bahasa generatif.

4.1 Dua Belas Tipe Kata Kerja Dasar [5]

Tabel 2-1 : Tipe Kata Kerja Dasar

	Tipe Kata Kerja Dasar	Kasus yang diperlukan	Case Frame
1.	KK Keadaan	Objek keadaan	+ [- Ok]
2.	KK Keadaan – Pengalam	Pengalam, Objek keadaan	+ [- P, Ok]
3.	KK Keadaan – Benefaktif	Benefaktif, Objek keadaan	+ [- B, Ok]
4.	KK Keadaan – Lokatif	Objek keadaan, Lokatif	+ [- Ok, L]
5.	KK Proses	Objek	+ [- O]
6.	KK Proses – Pengalam	Pengalam, Objek	+ [- P, O]
7.	KK Proses – Benefaktif	Benefaktif, Objek	+ [- B, O]
8.	KK Proses – Lokatif	Objek, Lokatif	+ [- O, L]
9.	KK Aksi	Agen, Objek	+ [- A, O]
10.	KK Aksi – Pengalam	Agen, Pengalam, Objek	+ [- A, P, O]
11.	KK Aksi – Benefaktif	Agen, Benefaktif, Objek	+ [- A,B,O]
12.	KK Aksi – Lokatif	Agen, Objek, Lokatif	+ [- A,O,L]

Urutan kasus setiap KK disebut kerangka kasus (*case frame*) KK itu, dan secara konvensional dituliskan: + [- Ok], + [- P, Ok], + [- B, Ok], + [- Ok, L], dan seterusnya. Tanda garis (-) dalam kerangka kasus tersebut menandakan bahwa ada Kk tertentu yang dapat dimasukkan dalam kerangka kasus bersangkutan. Tanda “+” menyatakan ciri semantik. Urutan kasus setiap KK disebut kerangka kasus (*case frame*) KK itu, dan secara konvensional dituliskan: + [- Ok], + [- P, Ok], + [- B, Ok], + [- Ok, L], dan seterusnya. Tanda garis (-) dalam kerangka kasus tersebut menandakan bahwa ada Kk tertentu yang dapat dimasukkan dalam kerangka kasus bersangkutan. Tanda “+” menyatakan ciri semantik. Contoh penerapan aturan pada table 2.1 di atas dapat dilihat pada table 2.2

Tabel 2-2 : Contoh Penerapan Case grammar [5]

	Tipe Kata Kerja Dasar	Contoh Kalimat
1.	KK Keadaan	Dinding rumah itu putih. Ok KK
2	KK Keadaan-Pengalam	Anak itu takut akan hantu. P KK Ok
3	KK Keadaan-Benefaktif	Bapak sudah punya sepasang bendi. B KK Ok
4	KK Keadaan Lokatif	Ibu ada di dapur. Ok KK L
5	KK Proses	Gunung itu longsor. O KK
6	KK Proses-Pengalam	Kami was-was atas keterlambatan bapak. P KK O
7	KK Proses-Benefaktif	Pak Surya menang dua juta rupiah. B KK O
8	KK Proses-Lokatif	Matahari sedang terbit dari ufuk Timur. O KK L
9	KK-Aksi	Mereka sedang main kartu. A KK O
10	KK-Aksi Pengalam	Bapak bilang "tidak" kepada Hadi. A KK O P
11	KK-Aksi Benefaktif	Anwar meminjami saya uang A KK B O
12	KK-Aksi Lokatif	Saya mengembalikan buku ke perpustakaan A KK O L

4.2 **Performansi Case Grammar**

Dalam [5] disebutkan beberapa performansi *case grammar* :

1. Kasus-kasus pada *case grammar* merepresentasikan karakteristik dasar dari pemikiran manusia, sehingga representasi semantik yang dihasilkan mudah dimengerti manusia.
2. Parsing menggunakan *case grammar* biasanya bersifat *expectation-driven*. Satu kata kerja yang telah ditentukan tipe semantiknya, dapat digunakan untuk menentukan hubungan dari kata benda terhadap verba, hubungan tesebut direpresentasikan ke dalam kasus tertentu.

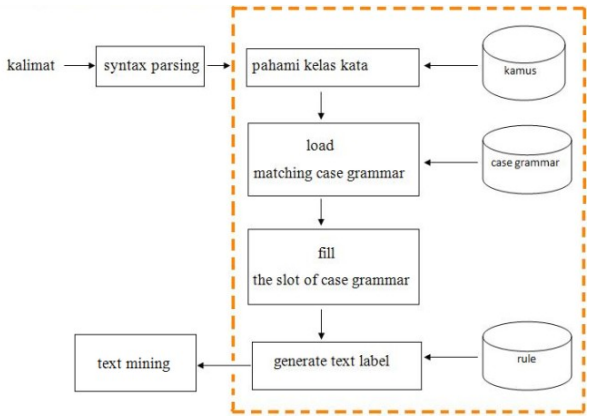
5 **Analisis dan Perancangan Perangkat Lunak**

5.1 **Gambaran Umum Perangkat Lunak**

Dalam sistem perangkat lunak yang akan dibangun pada penelitian ini, masukan ke sistem adalah sebuah kalimat tunggal yang diambil dari artikel berita yang berasal dari website media cetak yang diambil secara offline. Kalimat yang diambil dari artikel berita tersebut akan dimasukkan ke dalam file berekstensi .log. Kemudian kalimat tersebut akan di-*parsing* dengan menggunakan PC-PATR.

Berdasarkan hasil *syntax parsing* dengan menggunakan PC-PATR, sistem akan menentukan kelas kata dari kata kerja dalam kalimat yang telah di-*parsing* tadi. Proses berikutnya adalah *load case grammar* yang sesuai dengan kata kerja, dari basis pengetahuan. Setelah ditemukan *case grammar* yang sesuai, akan dilakukan pengisian slot dari *case grammar* tersebut. Terakhir, akan di-*generate* label berdasarkan peran semantik masing-masing kata dalam kalimat.

Jika digambarkan, secara umum prosesnya adalah sebagai berikut:



Gambar1: Gambaran Umum Perangkat Lunak

Bagian yang berada dalam garis putus-putus merupakan sistem yang dibangun dalam penelitian ini. Secara umum, hasil akhir dari sistem yang dibuat adalah kalimat yang sudah diberi label sesuai bagian yang diberi garis putus-putus.

5.2 **Analisis Masukan dan Keluaran**

Masukan perangkat lunak adalah sebuah kalimat hasil parsing yang berupa file .log. dan keluaran sistem adalah kalimat yang telah diberi label sesuai dengan *semantic role*-nya.

5.3 **Analisis Fungsionalitas Perangkat Lunak**

Dari gambaran perangkat lunak diatas maka dapat dirumuskan fungsionalitas yang ada dalam perangkat lunak. Fungsionalitas-fungsionalitas tersebut adalah :

1. Kelola kamus
   Memasukkan data yang akan diproses oleh perangkat lunak, mengubah data yang tersimpan dalam basis data yang disebabkan oleh adanya kesalahan dalam hal teknik penulisan data pada saat melakukan proses input data, dan menghapus data katalog yang dirasakan sudah tidak dapat digunakan lagi. Menggolongkan kata kerja sesuai dengan tipe semantiknya.
2. *Load* file hasil *parsing*
3. Memahami kelas kata
   Menentukan fungi sintaksis kalimat hasil *parsing*

4. *Search case grammar*
   Load matching case grammar dan kemudian mengisi slot dari *case grammar*
5. *Generate text label*
   Menentukan peran semantis masing-masing fungsi sintaksis kalimat berdasarkan rule *case grammar*

Daftar Pustaka:

[1] Albright, Emil. 2006. *An Overview of Historical Development of Semantic Role Representation.*

[2] Alwi, Hasan dan Dardjowidjojo, Soenjono. 2000, *Tata Bahasa Baku Bahasa Indonesia*. Jakarta : Balai Pustaka.

[3] Daniel Gildea dan Daniel Jurafsky. *Automatic Labeling of Semantic Roles.*2002. http://www.cs.rochester.edu/~gildea/gildea-cl02.pdf. Didownload pada tanggal 16 Oktober 2008.

[4] Joice. 2002. *Pengurai Struktur Kalimat Bahasa Indonesia yang Mengunakan Constraint-Based Formalism.* Skripsi Sarjana Ilmu Komputer Universitas Indonesia.

[5] Musfirah, Ulfah. 2007. *Penentuan Peran Semantis Kata dalam Kalimat Bahasa Indonesia Menggunakan Case Grammar*. Bandung: Institut Teknologi Telkom.

[6] Parunak, Van. *Case Grammar: A Linguistic Tool for Engineering Agent-Based System.* http://www.newvectors.net/staff/parunakv/casegram.pdf. Didownload pada tanggal 24 Oktober 2008.

[7] Richard Johansson dan Pierre Nugues. *A FrameNet-based Semantic Role Labeler for Swedish.*http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.102.2924. Didownload pada tanggal 29 September 2008.

[8] Shady Sheheta, Fakhri Karray, Mohamed Kamel. 2005. *Concept Mining using Conceptual Ontological Graph (COG).*

[9] Shady Sheheta, Fakhri Karray, Mohamed Kamel. 2006. *A Conceptual-based Model for Enhancing Text Categorization*, KDD ’07, 12-15 Agustus 2007.

[10]Shady Sheheta, Fakhri Karray, Mohamed Kamel. 2006. *Enhancing Text Clustering using Concept-based Mining Model*, KDD ’07, 12-15 Agustus 2007.

[11]Shady Sheheta, Fakhri Karray, Mohamed Kamel. 2006. *Enhancing Text Retrieval Performance using Conceptual Ontological Graph*, KDD ’07, 12-15 Agustus 2007.

[12]Suciadi, James. *Studi Analisis Metode-Metode Parsing dan Interpretasi Semantik Pada Natural Language Processing.* Journal of Artificial Intelligence Research.

[13]Tarigan, Henry Guntur. 1990. *Pengajaran Tata Bahasa Kasus*. Bandung: Penebit Angkasa.

[14]Wibisono, Yudi., & Khodra, M. L. 2006. *Clustering Berita Berbahasa Indonesia*

[15]Adiwijaya, Igg. 2006. *Texi Mining dan Knowledge Discovery.* Komunitas *Data mining* Indonesia & Soft-omputing Indonesia