

Penggunaan Pohon dalam Merepresentasikan Struktur Kalimat Bahasa Alami

Dita Anindhika - 13509023¹
 Program Studi Teknik Informatika
 Sekolah Teknik Elektro dan Informatika
 Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
¹13509023@std.stei.itb.ac.id

Abstrak—Pohon merupakan salah satu terapan graf yang konsepnya banyak dimanfaatkan dalam memecahkan berbagai permasalahan. Dalam kehidupan sehari-hari orang telah lamamenggunakan pohon untuk menggambarkan hirarkhi. Beberapa contohnya adalah pohon silsilah keluarga, struktur organisasi, dan organisasi pertandingan. Kali ini, penulis mencoba membahas tentang penggunaan pohon dalam merepresentasikan struktur kalimat bahasa natural. Yang dimaksud dengan bahasa natural tak lain adalah bahasa manusia yang sering kita gunakan sehari-hari. Pohon memang sudah lama digunakan oleh para ahli bahasa untuk mengurai kalimat, pohon yang demikian umumnya disebut pohon parsing (*parse tree*). Makalah ini membahas tentang pemrosesan bahasa natural dalam pembuatan pohon parsing (*parse tree*).

Kata Kunci—Pohon, bahasa natural, graf, pohon parsing (*parse tree*).

I. PENDAHULUAN

Pohon merupakan salah satu terapan graf. Untuk itu, sebelum kita melangkah lebih jauh mengenai pohon ada baiknya kita membahas tentang pengertian dari graf secara singkat.

A. Teori Graf

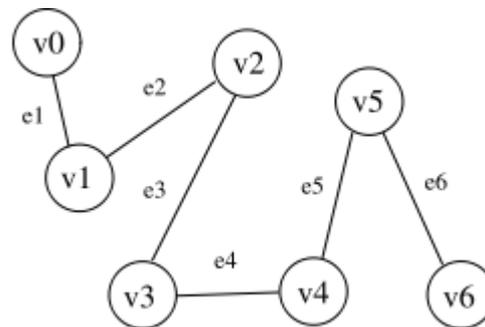
Graf secara informal dapat didefinisikan sebagai himpunan benda-benda yang disebut simpul yang terhubung oleh sisi atau busur. Biasanya graf digambarkan sebagai kumpulan titik-titik (melambangkan simpul) yang dihubungkan oleh garis-garis (melambangkan sisi) atau garis berpanah (melambangkan busur). Suatu sisi dapat menghubungkan suatu simpul dengan simpul yang sama.

Sedangkan secara matematis, Graf G didefinisikan sebagai pasangan himpunan (V,E) yang dalam hal ini :

- V = himpunan tidak kosong dari simpul-simpul (*vertices* atau *node*)
- E = himpunana sisi (*edges* atau *arcs*) yang menghubungkan sepasang simpul.

Untuk itu, graf dapat dinyatakan dalam notasi $G=(V,E)$. Seperti untuk gambar 1 bawah ini, Graf G dapat dinyatakan dengan $V = \{1, 2, 3, 4, 5, 6\}$ dan $E =$

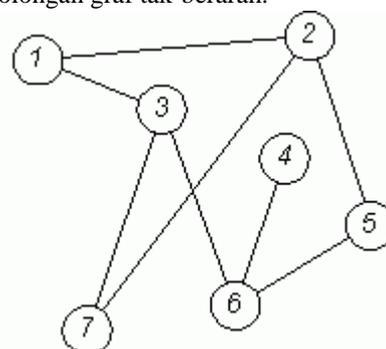
$\{(1,2), (1,5), (2,3), (3,4), (4,5), (5,2), (4,6)\}$ sehingga, $G = (\{1, 2, 3, 4, 5, 6\}, \{(1,2), (1,5), (2,3), (3,4), (4,5), (5,2), (4,6)\})$



Gambar 1. Graf

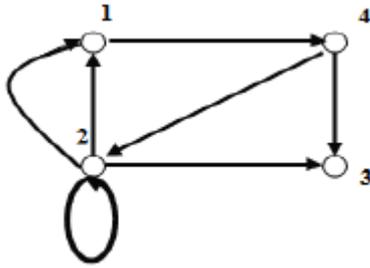
Berdasarkan arah pada sisi graf,graf dibagimenjadi dua kategori, yakni :

1. Graf tak-berarah (*undirected graph*)
 Graf tak-berarah adalah graf yang sisinya tidak memiliki orientasi arah. Pada kasus ini berlaku $(v_1, v_2) = (v_2, v_1)$. Pohon sendiri termasuk kedalam golongan graf tak-berarah.

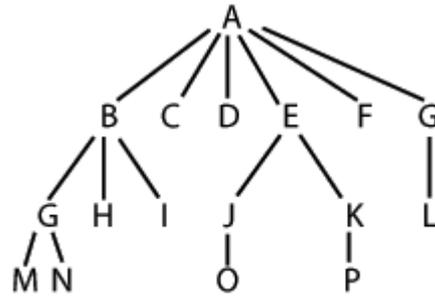


Gambar 2. Graf Tak-Berarah

2. Graf berarah (*directed graph* atau *digraph*)
 Graf berarah adalah graf yang setiap sisinya diberi orientasi arah. Hal tersebut menyebabkan $(v_1, v_4) \neq (v_4, v_1)$.



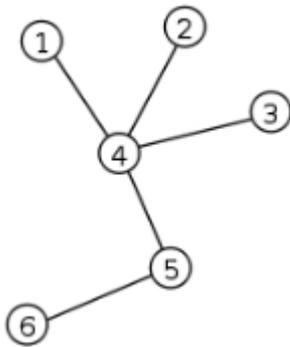
Gambar 3. Graf Berarah



Gambar 5. Pohon Berakar

B. Pohon

Pohon didefinisikan sebagai graf tak-berarah terhubung yang tidak mengandung sirkuit. Graf yang ditunjukkan oleh gambar 2 walaupun tak-berarah dan terhubung tidak dapat diklasifikasikan sebagai pohon karena mengandung sirkuit.



Gambar 4. Pohon

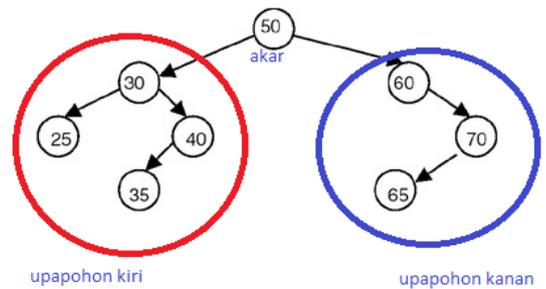
Berikut ini adalah sifat-sifat pohon :

1. G adalah pohon
2. Setiap pasang simpul di dalam G terhubung dengan lintasan tunggal.
3. G terhubung dan memiliki $m = n - 1$ buah sisi
4. G tidak mengandung sirkuit $m = n - 1$ buah sisi
5. G tidak mengandung sirkuit dan penambahan satu sisi pada graf akan membuat satu sirkuit.
6. G terhubung dan semua sisinya adalah jembatan (jembatan adalah sisi yang bila dihapus menyebabkan graf terpecah menjadi dua komponen).

Pohon sendiri dapat diklasifikasikan menjadi pohon bebas dan pohon berakar. Pada literatur, pohon yang didefinisikan sebagai graf yang terhubung dan tidak mengandung sirkuit adalah pohon bebas sedangkan pohon yang simpulnya diperlakukan sebagai akar dan sisi-sisinya diberi arah sehingga menjadi graf berarah.

Dalam penggunaan pohon dikenal beberapa terminologi dasar sebagai berikut :

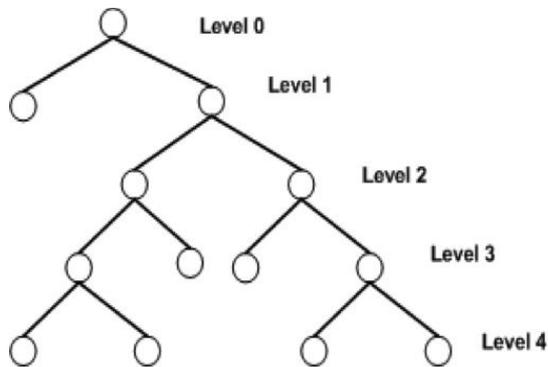
1. Anak (*child* atau *children*) dan Orangtua (*parent*)
Seperti yang terlihat pada gambar 5, G adalah sebuah simpul di dalam pohon berakar. Simpul G dikatakan anak simpul A jika ada sisi dari simpul ke G dan B. Dalam hal demikian, B disebut orangtua (*parent*) G.
2. Lintasan (*path*)
Lintasan dari simpul v_1 ke simpul v_k adalah runtunan simpul-simpul $v_1, v_2, v_3, \dots, v_k$ sedemikian sehingga v_i adalah orangtua dari v_{i+1} untuk $1 \leq i < k$. Seperti contoh pada gambar 5, lintasan dari simpul A ke simpul N adalah A, B, G, N.
3. Keturunan (*descendant*) dan leluhur (*ancestor*)
Jika terdapat lintasan dari simpul P ke simpul E didalam pohon, maka P adalah leluhur dari simpul E, dan E adalah keturunan dari simpul P.
4. Saudara kandung (*sibling*)
Simpul yang berorangtua sama adalah saudara kandung satu sama lain. Misalnya, G, H, dan I adalah saudara kandung.
5. Upapohon (*subtree*)
Upapohon sebuah simpul dalam pohon yang dapat membentuk pohon. Misalnya pada pohon biner, terdapat akar, upapohon kiri, dan upapohon kanan.



Gambar 6. Pohon Biner

6. Derajat (*degree*)
Derajat sebuah simpul pada pohon berakar adalah jumlah upapohon (atau jumlah anak) pada simpul tersebut. Misalnya pada gambar 5, derajat E adalah 4.

7. Daun (*leave*)
Simpul yang berderajat nol (atau tidak mempunyai anak) disebut daun. Pada gambar 5, L adalah daun.
8. Simpul dalam (*internal nodes*)
Simpul yang mempunyai anak disebut simpul dalam. G adalah sebuah simpul dalam.
9. Aras (*level*) atau tingkat
Akar mempunyai aras = 0, sedangkan aras simpul lainnya = 1 + panjang lintasan dari akar ke tempat tersebut.



Gambar 7. Pendefinisian aras tiap simpul

10. Tinggi (*height*) dan kedalaman (*depth*)
Aras maksimum dari suatu pohon disebut tinggi atau kedalaman pohon tersebut. Dengan kata lain, aras adalah panjang maksimum lintasan dari akar ke daun. Seperti pada gambar 7, aras maksimum dari pohon adalah 4.

C. Bahasa Natural

Bahasa Natural adalah bahasa yang muncul secara tidak disengaja sebagai hasil dari pemikiran manusia yang umumnya dipelajari dalam ilmu bahasa. Dalam mengidentifikasi bahasa, ilmu bahasa mengenal beberapa tingkatan, yaitu :

1. Fonetik (*phonetics*)
Fonetik adalah bunyi yang dihasilkan alat ucap. Bahasa dari tiap individu membentuk sebuah satuan bunyi yang dinamakan *phonemes*.
2. Fonologi (*phonology*)
Fonologi adalah ilmu tentang perbendaharaan fonem sebuah bahasa dan distribusinya. Hal yang mendasari dibentuknya fonologi adalah perubahan fonem. Dalam pelaksanaan bunyi-bunyi ujaran, terjadilah pengaruh timbal-balik antara bunyi-bunyi ujaran yang berdekatan yang menyebabkan perubahan-perubahan bunyi-ujaran; misalnya fonem /a/ yang berada dalam suku kata /a/ yang berada dalam suku kata terbuka terdengar lebih nyaring bila dibandingkan dengan fonem /a/ yang terdapat dalam suku kata tertutup. Bandingkan antara /a/ pada kata: *pada*, *kata*, *rata*, dengan pada kata: *bedak*, *tidak*, *sempat*, dan lain-lain.
3. Morfologi (*morphology*)

Morfologi adalah tingkatan yang membahas tentang morfem dan kombinasinya. Morfem sendiri adalah satuan bentuk bahasa terkecil yg mempunyai makna secara relatif stabil dan tidak dapat dibagi atas bagian bermakna yg lebih kecil.

4. Sintaksis (*syntax*)
Sintaksis adalah pengaturan dan hubungan kata dengan kata atau dengan satuan lain yg lebih besar.
5. Semantik (*semantics*)
Semantik adalah bagian struktur bahasa yg berhubungan dengan makna ungkapan atau struktur makna suatu wicara.
6. Pragmatik
Pragmatik adalah hal-hal yang berkenaan dengan syarat-syarat yang mengakibatkan serasi tidaknya pemakaian bahasa dalam komunikasi.

Keenam tingkatan yang disebut diatas akan membangun sintaksis terbesar yaitu kalimat. Kelengkapan struktur kalimat sendiri dalam bahasa Indonesia sekurangnya adalah memiliki subjek (S) dan predikat (P). Subjek pada kalimat adalah pelaku dan untuk mencari subjek dalam suatu kalimat dapat digunakan kata tanya apa dan siapa, sedangkan predikat umumnya adalah kata kerja dan untuk mencarinya dapat dipergunakan kata tanya bagaimana.

Selain subjek dan predikat, tata kalimat dalam bahasa Indonesia juga mengenal objek (O), pelengkap (Pel), dan keterangan (Ket). Objek merupakan kalimat yang dipermutasikan dengan subjek dalam kalimat pasif. Dalam kalimat, biasanya objek dapat diikuti oleh pelengkap ataupun keterangan. Pelengkap adalah unsur kalimat yang kehadirannya bersifat wajib mengikuti verba intransitif.

Keterangan bukanlah unsur utama dalam kalimat, sifatnya hanya untuk menerangkan. Keterangan sendiri dapat berada pada awal mau akhir kalimat mengikuti objek maupun predikat. Keterangan sendiri dapat dikategorikan menjadi keterangan waktu, tempat, cara, sebab, tujuan, dan aposisi/pendampingan.

II. PEMBAHASAN

Penggunaan pohon dalam merepresentasikan bahasa natural sudah lazim dipergunakan oleh para ahli bahasa untuk menguraikan struktur kalimat dengan metode parsing. Parsing adalah proses menganalisis teks yang terbuat dari urutan *token* (misalnya kata-kata), untuk menenukan struktur tata bahasa berdasarkan tata bahasa baku. Hasil dari parsing tersebutlah yang nantinya dipergunakan untuk direpresentasikan dalam bentuk pohon. Pohon tersebut dikenal dengan nama pohon parsing (*parse tree*).

A. Pohon Parsing

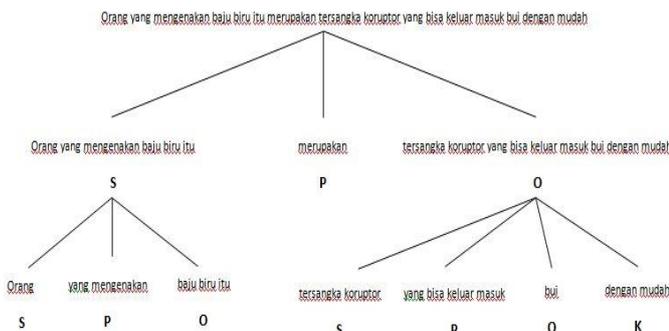
Pohon parsing (*parse tree*) adalah pohon berakar yang terdiri dari struktur sintaksis suatu kata sesuai dengan tata bahasa formal. Pohon parsing juga sering disebut sebagai

concrete syntax tree namun ia berbeda dengan *abstract syntax tree* karena struktur dan elemennya lebih mencerminkan sintaks dari bahasa masukan.

Pohon ini berfungsi untuk menggambarkan bagaimana memperoleh suatu string dengan cara menurunkan simbol-simbol variabel menjadi simbol-simbol terminal, sampai tidak ada yang belum tergantung. Simbol awal akan menjadi akar dan setiap penurunan simbol variabel akan dipilih aturan produksi yang menuju ke solusi.

Proses parsing dapat dilakukan dengan cara melakukan penurunan terkiri (*leftmost derivation*) dan penurunan terkanan (*rightmost derivation*).

Para ahli bahasa menggunakan pohon ini untuk mengurai kalimat. Contohnya dapat dilihat pada gambar berikut.



Gambar 8. Contoh penerapan pohon parsing dalam tata kalimat bahasa Indonesia

Pohon parsing membagi kalimat majemuk bertingkat tersebut menjadi subjek (S), predikat (P), objek (O).

- S : orang yang mengenakan baju biru itu
(S) (P) (O)
- P : merupakan
- O : tersangka koruptor yang bisa keluar masuk
(S) (P)
bui dengan mudah
(O) (K)

B. Pemrosesan Bahasa Natural

Pemrosesan bahasa natural dapat dibagi atas beberapa proses, yaitu:

1. *Morphological analysis*, dimana kata secara individu dianalisis ke dalam komponen-komponennya dan yang bukan kata (seperti tanda baca) dipisahkan dari kata
2. *Syntactic analysis*, urutan linier dari kata ditransformasikan ke dalam struktur yang menunjukkan bagaimana kata saling berhubungan. Urutan kata yang melanggar aturan bahasa dalam pengkombinasian kata, akan ditolak.
3. *Semantic analysis*, dibuat mapping antara struktur sintaksis & objek dalam domain kerja.

Struktur dimana tidak ada mapping yang memungkinkan, akan ditolak.

4. *Discourse integration*, dimana arti dari kalimat secara individu bergantung pada kalimat-kalimat yang mendahului & mungkin dapat mempengaruhi arti dari kalimat-kalimat yang mengikutinya
5. *Pragmatic analysis*, dimana struktur yang merepresentasikan apa yang diucapkan, diinterpretasikan kembali untuk menentukan apa arti sebenarnya.

C. Metode Parsing

1. Top Down Parsing

Top down parser memulai pemeriksaan dari simbol awal kalimat dan mencoba untuk mencari bentuk simbol terminal berikutnya yang sesuai dengan jenis kata dari kalimat masukan. Dikenal dua kelas dalam *top down parsing*, yakni metode dengan *back up* dan metode tanpa *back up*.

Metode brute force merupakan metode yang tergolong dalam kelas metode dengan *back up*. Kelas metode ini merupakan kelas metode parsing yang menggunakan produksi alternatif, jika ada, ketika hasil penggunaan sebuah produksi tidak sesuai dengan simbol input. Penggunaan produksi sesuai dengan nomor urut produksi.

Untuk kelas metode tanpa *back up*, dikenal metode *recursive descent*. Kelas metode ini tidak menggunakan produksi alternative seperti metode *brute force* yang diaparkan sebelumnya. Jika produksi A mempunyai dua buah ruas kanan atau lebih maka produksi yang dipilih untuk digunakan adalah produksi dengan simbol pertama ruas kanannya sama dengan input yang sedang dibaca. Jika tidak ada produksi yang demikian maka dikatakan bahwa parsing tidak dapat dilakukan.

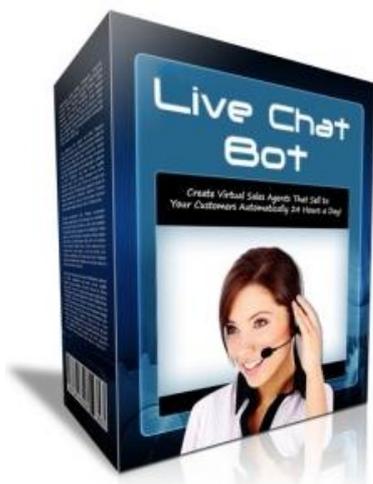
Ketentuan produksi yang digunakan metoda *recursive descent* adalah jika terdapat dua atau lebih produksi dengan ruas kiri yang sama maka karakter pertama dari semua ruas kanan produksi tersebut tidak boleh sama. Ketentuan ini tidak melarang adanya produksi yang bersifat rekursi kiri.

2. Bottom Up Parsing

Bottom Up parser mencari dari simbol-simbol terminal menuju ke arah pembentukan simbol awal sebuah kalimat. Salah satu metodenya adalah metode grammar presenden sederhana.

III. ANALISIS

Penggunaan pohon dalam parsing bahasa natural memudahkan pemahaman mengenai struktur kalimat yang relatif rumit, terutama untuk kalimat majemuk bertingkat sedangkan terapannya pada teknologi masa kini dapat kita temukan pada *translator* serta *artificial intelligence* seperti *chat bot*.



Gambar 9. Chat Bot

Untuk menghasilkan terjemahan, translator memanfaatkan pohon parsing untuk mencari pola dalam ratusan juta dokumen untuk membantu memutuskan pada terjemahan terbaik. Dengan mendeteksi pola dalam dokumen-dokumen yang telah diterjemahkan oleh penerjemah manusia, *translator* dapat membuat terjemahan cerdas seperti layaknya terjemahan yang dibuat oleh penerjemah. Namun, dalam pemrosesan bahasa natural, seringkali ditemui kesulitan pemahaman dari makna kalimat karena dalam kalimat arti dari tiap kata ditentukan oleh kata-kata sebelum dan sesudahnya. Hal tersebut juga menyebabkan tingkat akurasi translator yang masih belum bisa menghasilkan terjemahan yang setara dengan terjemahan manusia.

IV. KESIMPULAN

Penggunaan pohon pada pemrosesan bahasa natural adalah untuk merepresentasikan struktur dari bahasa alamit tersebut. Pohon digunakan karena memiliki karakteristik yang *men-support* struktur kompleks. Seperti dijelaskan pada bab sebelumnya, dengan menggunakan pohon, struktur dari kalimat bahasa natural dapat lebih mudah dipahami.

Dalam pemrosesan bahasa natural, sampai saat ini masih terhalang kendala banyaknya variasi makna yang dimiliki tiap kalimat, sehingga dalam penerapannya *translator* masih kesulitan untuk dapat menghasilkan terjemahan terbaik.

REFERENSI

- [1] Munir, Rinaldi. *Diktat Kuliah IF2091 Struktur Diskrit*. Bandung: Program Studi Teknik Informatika ITB, 2008.
- [2] Halaman web
http://en.wikipedia.org/wiki/Concrete_syntax_tree
Tanggal akses : 13 Desember 2010
- [3] Halaman web
<http://www-rohan.sdsu.edu/~ling354/nat-lang.html#Phonetics>
Tanggal akses : 13 Desember 2010
- [4] Halaman web

<http://en.wikipedia.org/wiki/Parsing>

Tanggal akses : 15 Desember 2010

- [5] Halaman web
[en.wikipedia.org/wiki/Tree_\(graph_theory\)](http://en.wikipedia.org/wiki/Tree_(graph_theory))
Tanggal akses : 16 Desember 2010
- [6] Halaman web
http://en.wikipedia.org/wiki/Natural_language_processing
Tanggal akses : 16 Desember 2010
- [7] Staf Pengajar Sosiokomunikasi ITB. *Metode Pulisan Ipteks*. Bandung : KK Ilmu Kemanusiaan ITB, 2007.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 16 Desember 2010

Dita Anindhika - 13509023