

ONTOLOGY SEBAGAI SOLUSI PENCARIAN MAKNA AMBIGU DALAM DALAM SISTEM YANG HETEROGEN

Herlina Jayadianti*, Lukito Edi Nugroho*, Paulus Insap Santosa*,
Wahyu Widayat**, Carlos Sousa Pinto***

*Teknik Elektro dan Teknologi Informasi, Fakultas Teknik, Universitas Gadjah Mada,
Yogyakarta, Indonesia

** Fakultas Ekonomi, Universitas Gadjah Mada, Yogyakarta, Indonesia

***Departemento de Sistemas da Informacao, Universidade do Minho, Campus de Azurem,
Guimaraes, Portugal

Email : herlinajayadianti@gmail.com

Abstract

Human beings as creatures of God has tremendous advantages in terms of process knowledge . Knowledge , ideas , predictions , and beliefs intertwined managed to produce a greater range of knowledge in the human brain . With its experience a human can easily distinguish them said the same but have different meanings , or otherwise different words but the same meaning . Of all the advantages it has, on one hand, humans also have various limitations in managing large amounts of data , the limitations of doing a quick search , to the limitations in considering complex data . This is the basis of this study . The research in this paper will focus on an ambiguous word search prototype system , where the system is expected later to understand user queries performed more easily . SPARQL , RDF and ontology is the technology that will be used to support the course of this study .

Keywords : *Ontology, Ambiguous, Knowledge, Term*

Manusia sebagai makhluk ciptaan Tuhan memiliki kelebihan yang luar biasa dalam hal memproses pengetahuan. Pengetahuan, ide, prediksi, dan keyakinan dikelola saling terkait sehingga menghasilkan rangkaian pengetahuan yang lebih besar didalam otak manusia. Dengan pengalaman yang dimilikinya manusia dengan mudah dapat membedakan kata yang sama namun diantaranya memiliki makna yang berbeda, atau sebaliknya kata yang berbeda namun memiliki makna yang sama. Dari segala kelebihan yang dimilikinya, disatu sisi manusia juga memiliki berbagai keterbatasan dalam mengelola data dalam jumlah besar, keterbatasan dalam melakukan pencarian secara cepat, hingga keterbatasan dalam mengingat data yang kompleks. Hal tersebut menjadi dasar dilakukannya penelitian ini. Penelitian dalam paper ini akan fokus pada sebuah prototype system pencarian kata ambigu, dimana system yang diharapkan nantinya dapat memahami query yang dilakukan user dengan lebih mudah. SPARQL, RDF dan ontology adalah teknologi yang akan digunakan untuk mendukung jalannya penelitian ini.

Keywords: Ontology, Ambigu, Pengetahuan, Term

1. PENDAHULUAN

Pengetahuan yang dimiliki manusia dapat berasal dari pengalaman yang dialaminya baik secara langsung ataupun tidak, menyimpannya dan akan digunakan untuk melengkapi pengetahuan baru lainnya dikemudian hari. Dapat dikatakan bahwa manusia menyusun setiap hal yang dialaminya kemudian mengelolanya. Manusia sebagai makhluk yang cerdas dapat dengan mudah mengelola dan menghubungkan antara pengalaman-pengalaman tersebut ke dalam sebuah *network* yang sangat besar didalam otak yang dimilikinya, yang akan digunakan untuk menganalisis persoalan di kemudian hari. Sebagai contoh : Manusia ketika dilahirkan belum mengetahui buah bernama "Apel", kemudian sejalan dengan waktu, dan pengalaman yang diperolehnya dari memakan buah "Apel" kemudian manusia mulai mengetahui rasanya, jenisnya, warnanya serta bentuk dari buah tersebut. Namun disaat yang berbeda manusia juga menemui kata "Apel" lainnya yang bermakna berbeda. "Apel" sebagai kata kerja "kunjungan spesial" dan bukan "Apel" sebagai kata benda "Buah berwarna merah".

Contoh : “Saya ingin Apel di malam minggu”

Apakah manusia dapat membedakan “Apel” yang dimaksud pada contoh diatas? Tentu saja jawabannya adalah dapat, karena manusia dapat berargumen, berdebat, serta berdiskusi untuk menyampaikan maksud yang diinginkannya, misalnya dengan menambahkan kata keterangan baru didalam kalimat contoh sebelumnya.

Contoh : “Saya ingin Apel (berwarna merah) di malam minggu”

Tentu ketika ditambahkan keterangan “Berwarna merah”, maka manusia lainnya sebagai lawan bicara dapat memahami dengan mudah kalimat yang disampaikan oleh lawan bicaranya tersebut. Sebab didalam otak mereka telah tersusun pengalaman yang sama mengenai warna buah apel dan apel adalah buah.

Contoh lainnya adalah : “Blackberry” sebagai buah dan “Blackberry” sebagai *mobilephone*. Untuk contoh ini akan dibahas lebih lanjut pada sub bab implementasi.

Contoh: “Blackberry berwarna hitam”

Dari contoh diatas, akan muncul dua makna ambigu, apakah buah blackberry berwarna hitam? Ataukah jenis *mobilephone* “Blackberry” yang memiliki warna hitam. Bagaimana membuat komputer dapat memahami makna ambigu dari kata “Blackberry” dan kemudian memahami query yang dilakukan user, sehingga hasil yang diberikan juga tepat.

Jika kita menelaah persoalan ambigu, maka hal ini tidak hanya terjadi dalam konteks Bahasa Indonesia saja, namun bisa terjadi di dalam bahasa manapun, seperti bahasa Inggris misalnya.

Contoh: “The tank is expensive”

Jika pengalaman yang kita miliki dan terkelola didalam otak adalah : “Perang di Iraq dengan menggunakan tank”, lalu “Korupsi pembelian kendaraan tempur berjenis tank” – maka lalu kita menyimpulkan bahwa “Tank itu mahal harganya”, “Kendaraan tank untuk perang itu mahal harganya”

Contoh: “The tank (for war) is expensive”

Namun apa yang terjadi jika pembicaraan itu terjadi di toko ikan atau di pasar hewan. Seorang pembeli berkata kepada penjual ikan:

Contoh: “The tank is expensive”

Maka tentu yang dimaksud bukanlah “Tank” sebagai kendaraan perang tetapi “tank” sebagai wadah atau aquarium. Masih banyak kata-kata ambigu lainnya seperti “Mouse”, “Body” dan lain sebagainya. Namun bukan fokus kita membericarakan masalah ambigu didalam percakapan antara manusia. Fokus utama kita adalah apakah jika masalah ambigu ini muncul didalam aplikasi mesin pencari, Maka komputer dapat dengan mudah mengenali kata yang dimaksud. Kami akan menunjukkan sebuah projek implementasi yang menunjukkan bahwa permasalahan ini dapat diselesaikan dengan ontology. Paper ini disusun dalam beberapa bagian yakni (1) pendahuluan, (2) Ontology dan teori dasar yang digunakan dalam menyusun paper ini, (3) Implementasi dan pembahasan, (5) Penutup yang berisi kesimpulan dan saran.

2. ONTOLOGY

Pengertian *ontology* sangat beragam dan berubah sesuai dengan berjalannya waktu, ada beberapa definisi *ontology* yakni:

- Neches (Neches, Fikes, Finin, Gruber, & Patil, 1991) : “Sebuah *ontology* merupakan definisi dari pengertian dasar dan relasi kosakata dari sebuah area sebagaimana aturan dari kombinasi istilah dan relasi untuk mendefinisikan kosakata”.

- Gruber (Gruber, 1993) memberikan definisi yang sering digunakan oleh beberapa orang, definisi tersebut adalah " *Ontology* merupakan sebuah spesifikasi eksplisit dari konseptualisme".
- Borst (P. Borst, Akkermans, & Top, 1997; W. N. Borst, 1997) melakukan penambahan dari definisi Gruber dengan mengatakan "Sebuah *ontology* adalah spesifikasi formal dari sebuah konseptual yang diterima (*share*)".
- Studer (Decker, Erdmann, Fensel, & Studer, 1998; Fensel, Decker, Erdmann, & Studer, 1998) mengemukakan bahwa *ontology* adalah "Konseptualisasi mengacu kepada sebuah model abstrak dari beberapa fenomena di dunia dengan memiliki identifikasi konsep yang relevan dari fenomena tersebut.

Secara umum dapat dikatakan bahwa *Ontology* merupakan suatu teori tentang makna dari suatu obyek, *property* dari suatu obyek, serta relasi obyek tersebut yang mungkin terjadi pada suatu domain pengetahuan. Pada tinjauan filsafat, *ontology* adalah studi tentang sesuatu yang ada. Selain itu *ontology* adalah sebuah konsep yang secara sistematis menjelaskan tentang segala sesuatu yang ada atau nyata. Dalam bidang kecerdasan buatan, *ontology* memiliki dua pengertian yang berkaitan yakni: (1) *Ontology* merupakan kosakata representasi yang sering dikhususkan untuk domain atau subyek pembahasan tertentu. (2) *Ontology* sebagai suatu *body of knowledge* untuk menjelaskan suatu bahasan tertentu.

2.1 KOMPONEN ONTOLOGY

Komponen dari *Ontology* diantaranya adalah: (1) Class atau disebut juga konsep, koleksi, tipe dari objek. (2) Sub Class adalah hirarki dari kelas, dan memiliki / mewarisi sifat dari kelas di atasnya. (3) Individu adalah Instances atau objek. (4) Attribute disebut juga properti atau karakteristik, dan (5) Relasi yakni cara dimana masing-masing kelas dan individu dapat terhubung.

2.2 MENGAPA ONTOLOGY MENJADI PENTING

Menurut (Lanfranchi, Ciravegna, & Petrelli, 2005; Uren et al., 2006) :

- Membagi pemahaman yang sama mengenai struktur dari suatu informasi dan pengetahuan diantara orang-orang ataupun software agents.
- Memungkinkan dilakukannya penggunaan kembali domain pengetahuan.
- Membuat asumsi domain pengetahuan menjadi eksplisit.
- Menganalisis domain pengetahuan.
- Memungkinkan diberikannya label dan antonym terhadap obyek dalam domain.
- Mendefinisikan relasi antara resource-resource yang berbeda.
- Representasi dari konseptualisasi yang ada dapat dilakukan dengan menggunakan berbagai macam bahasa.

Seringkali pembuatan ontologi dari suatu domain bukanlah tujuan akhir pengembangan. Pembuatan antonym sama halnya dengan pendefinisian sekumpulan data dan struktur-strukturnya yang akan digunakan oleh program yang lain. Setelah ontologi selesai dibuat, perlu dilakukan evaluasi dengan menggunakan kriteria-kriteria berupa: (Gruber, 1993) (1) Kelengkapan (2) fungsional (*functional completeness*), (3) Generalitas (*generality*), (4) Efisiensi (*efficiency*), (5) Kejelasan (*perspicuity*), (6) Keakuratan (*precision/granularity*), dan (7) Minimalitas (*minimality*).

3. IMPLEMENTASI

Wordnet¹ adalah aplikasi 'library' yang dapat digunakan secara online. Wordnet terdiri dari berbagai terms atau vocabulary yang bisa dilihat persamaan katanya (*sinonim*) atau perbedaan katanya (*antonym*) secara lengkap. Didalam Wordnet bobot ukuran persamaan dan perbedaan makna diukur dalam 'sense'. (Lihat Gambar 1)

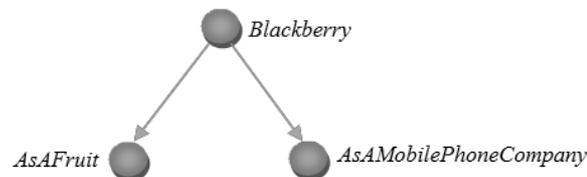
¹ <http://wordnet.princeton.edu/>

Search Word: <input type="text" value="blackberry"/>	Senses: <input type="text"/>
Searches for fruit: <input type="button" value="Noun"/> <input type="button" value="Verb"/>	
The noun blackberry has 2 senses (first 1 from tagged texts)	
1. (2) blackberry -- (large sweet black or very dark purple edible aggregate fruit of any of various bushes of the genus <i>Rubus</i>)	
2. blackberry, blackberry bush -- (bramble with sweet edible black or dark purple berries that usually do not separate from the receptacle)	
The verb blackberry has 1 sense (no senses from tagged texts)	
1. blackberry -- (pick or gather blackberries; "The children went blackberrying")	

Gambar 1. Sense dalam Wordnet

Namun didalam Wordnet "*Blackberry*" dikenali sebagai 'buah' atau '*fruit*', tidak ada yang mengenali *blackberry* dengan 'sense' '*Mobile phone*'. *Blackberry*" sebagai buah tentu tidak equivalent dengan "*Blackberry*" sebagai sebuah perusahaan mobile company, walaupun masing-masing memiliki 'term' yang sama (Lihat Gambar 2). Seperti telah dibahas di sub bab pendahuluan, bahwa terms yang sama dapat memiliki arti yang sama atau bahkan berbeda (ambigu) [3] atau sebaliknya terms yang berbeda dapat memiliki arti yang sama [4], [5]. Dalam Gambar 2, term *Blackberry* dibagi menjadi dua bagian, *Blackberry* sebagai buah (*fruit*) dan *Blackberry* sebagai perusahaan telepon genggam (*Mobile phone company*).

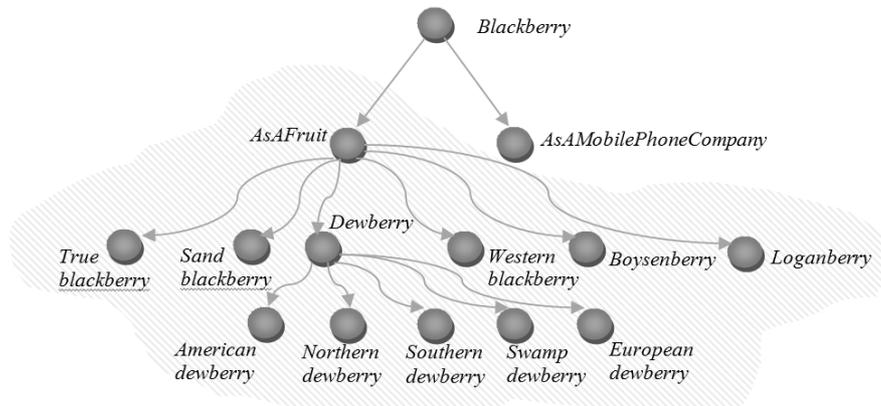
Blackberry (sebagai contoh dalam Gambar 2) memiliki ambigu kata yang bisa jadi menyulitkan bagi komputer dalam melakukan pencarian untuk term terkait.



Gambar 2. Perbedaan Term 'Blackberry'

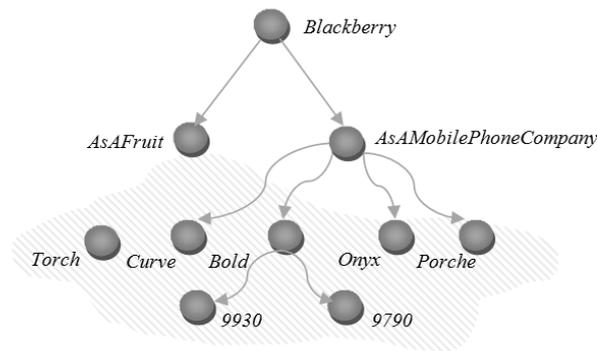
Step yang harus diambil adalah memilah kedua term 'blackberry' yang sama tersebut menjadi Ontology yang berbeda. Ontology 1 untuk domain *Blackberry* sebagai buah dan Ontology 2 untuk domain *Blackberry* sebagai Jenis telepon genggam.

Susunan kelas dan sub kelas dalam ontology 1 dapat dilihat dalam Gambar 3. Terdapat berbagai jenis atau persepsi mengenai *blackberry* sebagai buah seperti : "*True blackberry (Rubus fruticosus)*", "*Sand blackberry (Rubus cuneifolius)*", "*Dewberry (dewberry bush)*", "*American dewberry (Rubus Canadensis)*", "*Northern dewberry (Rubus flagellaris)*", "*Southern dewberry (Rubus trivialis)*", "*Swamp dewberry (Rubus hispidus)*", "*European dewberry (Rubus caesius)*", "*Western blackberry (Rubus ursinus)*", "*Boysenberry (boysenberry bush)*", "*Loganberry (Rubus loganobaccus or Rubus ursinus loganobaccus)*". Untuk "*Dewberry (dewberry bush)*" diantaranya terdiri atas subclass "*American dewberry*", "*Northern dewberry*" dan "*Southern dewberry*".



Gambar 3.. Berbagai jenis buah 'Blackberry' (Ontology 1)

Susunan kelas dan sub kelas dalam ontology 2 dapat dilihat dalam Gambar 4. Terdapat jenis Blackberry seperti : Torch, Curve, Bold, Onyx dan porche. "Bold blackberry" adalah salah satu tipe handphone blackberry dengan Type DataProperties 9930 dan 9790.



Gambar 4.. Berbagai jenis HandPhone 'Blackberry' (Ontology 2)

Untuk lebih jelasnya digunakan tools Protégé² untuk menyusun kelas-kelas di tiap ontology (Ontology 1 dan 2). (Lihat gambar 5). Selanjutnya untuk term yang sama dengan makna berbeda seperti 'Blackberry' diberikan syntax 'owl:disjointWith'.



Gambar 5.. Desain Class antara dua Ontology (Ontology 1 dan Ontology 2)

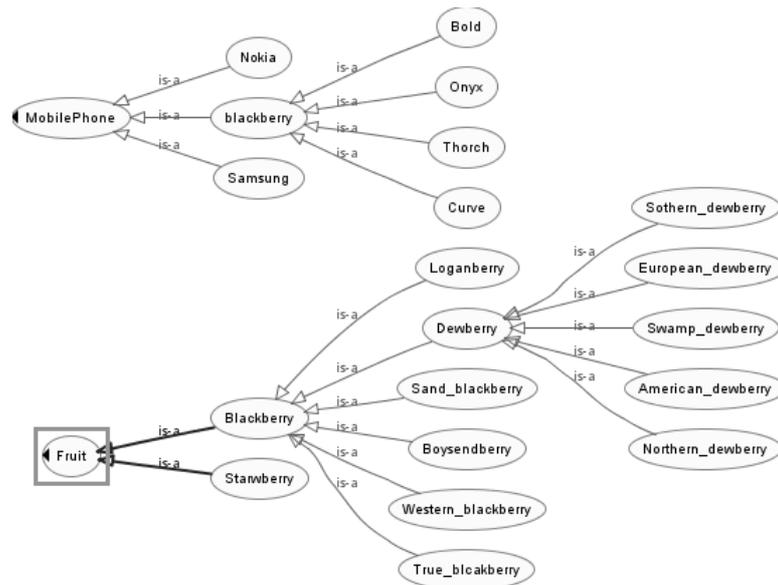
² <http://protege.stanford.edu/download/registered.html#p4.3>

```

<owl:Class rdf:about="&blackberry;Fruit">
<rdfs:label xml:lang="en">Blackberry</rdfs:label>
<rdfs:subClassOf rdf:resource="&Fruit"/>
<owl:disjointWith rdf:resource="&blackberry"/>
<owl:disjointWith rdf:resource="&blackberry;MobilePhone"/>
</owl:Class>

```

Proses ini dilakukan melalui proses import ontology. Ontology 1 di import ke dalam ontology 2 kemudian proses mapping diantara term-term tersebut di lakukan didalam ontology yang sama.



Gambar 6.. Desain Class

Dalam domain Blackberry sebagai ‘buah’ (Ontology 1) terdapat kelas “True_blackberry”. “True_blackberry” memiliki beberapa properties (datatype):

```

hasColor(True_blackberry,black);
hasLatin(True_blackberry,Rubus fruticosus);
hasOriginFrom(True_blackberry,Europe).
hasTaste(True_blackberry,sweet)

```

Dalam domain mobilephone (Ontology 2) terdapat subclass “Bold”. SubClass “Bold” memiliki properties (datatype): hasColor(bold,black). Selanjutnya setelah mendesain Kelas, Instances (individual), dan dataproperties di masing-masing ontology, langkah selanjutnya adalah pengujian dengan menggunakan SPARQL³ sebagai query language bagi Resource Description Framework (RDF)⁴ database.

```

Prefix : <http://www.semanticweb.org/user/blackberry#>
SELECT ?Blackberry ?Category
WHERE {?Blackberry :HasTaste ?Category.
WHERE {?Blackberry :HasColor ?Category.
?Category :TypeCategory ?value.
...
}

```

³ <http://www.w3.org/TR/rdf-sparql-query/>

⁴ RDF is a standard model for data interchange on the web. <http://www.w3.org/RDF/>

Blackberry	Category
western_blackberry	Fruit
true_blackberry	Fruit
sand_blackberry	Fruit
loganberry	Fruit
swamp_dewberry	Fruit
southern_dewberry	Fruit
northern_dewberry	Fruit
european_dewberry	Fruit
american_dewberry	Fruit
boysendberry	Fruit

Blackberry	Category
Torch	MobilePhone
Curve	MobilePhone
Bold	MobilePhone

Gambar 7.Hasil Query SPARQL

Untuk membangun sebuah "*artificial brain*" di dalam komputer agar dapat berfikir seperti "otak manusia" bukanlah hal yang sederhana. Memetakan semua persepsi, visualisasi, dan argumen dari otak manusia menjadi pengetahuan didalam program komputer adalah pekerjaan yang sulit. Sebagai contoh diatas, pengguna memasukkan nilai "blackberry" berdasarkan persepsi, pengalaman dan pengetahuan yang mereka miliki, seperti warna, aroma, rasa, jenis, dan lain-lain. Komputer akan memahami apa yang pengguna meminta hanya berdasarkan dataProperty sebagai nilai, seperti "HasTaste".

```
WHERE {?Blackberry :HasTaste ?Category.
```

```
WHERE {?Blackberry :HasColor ?Category.
```

berdasarkan DataProperty "HasTaste (Sweet)" maka targetnya adalah blackberry sebagai buah atau "*fruit*" dan bukan "*mobilePhone*".

Penelitian ini berbeda dari penelitian-penelitian sebelumnya tentang kesetaraan semantik [2],[3]. Dalam paper ini kami menunjukkan bahwa representasi pengetahuan dengan ontologi dapat mengatasi beberapa masalah semantik dalam konteks ambigu. Ontologi adalah teknologi yang tepat untuk menemukan hubungan semantik antara konsep-konsep dari berbagai model informasi.

5. KESIMPULAN DAN SARAN

Komputer tidak bisa belajar melalui pengamatan dan pengalaman seperti yang manusia dapat lakukan. Komputer juga tidak dapat dengan mudah berbagi dan memahami makna setiap query dari pengguna. Tidaklah mudah untuk membangun sebuah sistem yang dapat mengakomodasi apa semua pengguna inginkan, terutama jika kita harus mempertimbangkan ambiguitas kata. Dua situasi yang dapat terjadi adalah: (1) istilah yang sama dengan arti yang berbeda (istilah ambiguitas) atau (2) istilah yang berbeda dengan makna yang sama. Melalui tulisan ini, kami fokus pada hal ambiguitas. Untuk membangun pengetahuan tersebut kami menggunakan RDF dan Ontologi untuk membuat proses pencarian kata ambigu lebih mudah.

DAFTAR PUSTAKA

- C. Brewster and K. O'Hara, "Knowledge representation with ontologies: the present and future," *Intell. Syst. Ieee*, vol. 19, no. 1, pp. 72–81, 2004.
- H. Jayadianti, C. Pinto, L. Nugroho, and P. Insap, "Integrating large knowledge repositories in multiagent ontologies," *Proceedings of the 2nd International Workshop on Exploiting Large Knowledge Repositories (E-LKR'12), in conjunction with the 1st International Workshop on Automatic Text Summarization for the Future (ATSF-2012), CEUR-WS.org/Vol-882 - Exploiting Large Knowledge Repositories (E-LKR'12)*, 2012. [Online]. Available: <http://ceur-ws.org/Vol-882/>. [Accessed: 18-Jan-2013].
- H. Jayadianti, L. Nugroho, C. Pinto, P. Insap, and W. Widayat, "Solving Problems of Data Heterogeneity, Semantic Heterogeneity and Data Inequality using Ontologies," *Elpub Conference 2012*, 2012. [Online]. Available: http://www3.dsi.uminho.pt/elpub2012/Elpub_2012/Programme.html. [Accessed: 18-Jan-2013].
- N. F. Noy, "Semantic integration: a survey of ontology-based approaches," *Acm Sigmod Rec.*, vol. 33, no. 4, pp. 65–70, 2004.
- Y. Xue, "Ontological View-driven Semantic Integration in Open Environments," Book, The University of Western Ontario, 2010
-