

BAB 1. PENDAHULUAN

1.1. Latar Belakang Masalah

Web semantik bukan merupakan teknologi web yang terpisah akan tetapi merupakan perluasan dari teknologi web saat ini dimana informasi yang disampaikan memiliki definisi arti yang baik, memungkinkan komputer dan manusia bekerja sama lebih baik [1]. Web semantik memungkinkan data tidak hanya ditujukan atau dapat dimengerti oleh manusia sebagai pembaca tetapi juga agar bisa diproses dan dimengerti oleh mesin atau komputer. Perubahan teknologi web semantik yang sangat signifikan adalah perubahan dari *web of document* menjadi *web of data*. Salah satu perkembangan teknologi yang digunakan pada web semantik adalah *ontology*. Menurut Gruber, *Ontology* adalah sebuah spesifikasi secara eksplisit dari konseptualisasi [2]. *Ontology* mendeskripsikan data pada web dan keterhubungan data pada web.

Dengan semakin berkembangnya teknologi web semantik, maka akan semakin banyak pula jumlah *ontology* yang tersedia di Internet. Hal ini akan menimbulkan permasalahan heterogenitas, misalnya terdapat dua buah *ontology* dengan nama yang berbeda, struktur yang berbeda, didefinisikan dengan cara yang berbeda atau memiliki data dengan bahasa yang berbeda padahal kedua buah *ontology* tersebut mengekspresikan domain pengetahuan yang sama. Cara yang digunakan untuk mengurangi permasalahan heterogenitas *ontology* pada web semantik adalah *ontology matching*. *Ontology matching* merupakan proses untuk menemukan hubungan diantara *ontology* [3]. *Ontology matching* membandingkan dua buah *ontology* sehingga dapat menemukan keterhubungan diantara kedua *ontology* tersebut.

Terdapat beberapa teknik yang dapat digunakan untuk menyelesaikan permasalahan heterogenitas dengan menggunakan *ontology matching* diantaranya adalah *terminological techniques (name-based techniques)*, *structure-based techniques*, *instance-based techniques (extensional techniques)* dan *semantic-based techniques*. Dari beberapa teknik yang digunakan dalam *ontology matching* tersebut, salah satu teknik yang akan digunakan dalam menyelesaikan permasalahan heterogenitas pada tugas akhir ini adalah *instance-based ontology matching (IBOM)*. *Instance-based ontology matching* memanfaatkan perluasan dari kelas berupa *instance* yang merupakan contoh dari suatu kelas yang digunakan dalam proses *matching* dua buah *ontology* [4]. *Instance-based ontology matching* memiliki kelebihan yakni fokus pada bagian aktif dari *ontology* dan memiliki kemampuan untuk menangani ambiguitas bahasa seperti sinonim ataupun homonim. Kekurangan dari metode *instance-based ontology matching* ini adalah harus terdapat *instance* yang sesuai agar teknik ini dapat digunakan.

Dalam penerapan *instance-based ontology matching* ini digunakan dua buah parameter yakni parameter *Top N* dan *Similarity Threshold (ST)* yang berpengaruh pada proses *instance enrichment*. *Instance enrichment* merupakan proses memperkaya anotasi atau informasi sebuah *instance* pada suatu *ontology* dengan memanfaatkan *instance* yang dimiliki oleh *ontology* lain [4]. *Top N* akan menentukan jumlah maksimum *instance* pada saat proses *instance enrichment* dan

ST akan menentukan batas minimum nilai kemiripan *instance* sebelum *instance* tersebut dilakukan proses *instance enrichment*. Kedua parameter tersebut berpengaruh terhadap nilai performansi *recall*, *precision* dan *f-measure* dari IBOM. *Instance-based ontology matching* digunakan karena dengan semakin banyaknya jumlah *instance* pada suatu kelas yang terdapat pada suatu *ontology* maka akan semakin banyak proses perbandingan yang dilakukan untuk mencari kemiripan diantara dua buah *ontology* tersebut sehingga akan menghasilkan hasil yang lebih akurat dalam menentukan keterhubungan suatu *ontology*.

1.2. Perumusan Masalah

Dalam tugas akhir ini dirumuskan masalah sebagai berikut :

1. Bagaimana menerapkan metode *Instance-based ontology matching* untuk menganalisa dan menentukan keterhubungan *ontology* pada web semantik.
2. Bagaimana nilai parameter *TopN* dan *Similarity Threshold* mempengaruhi nilai performansi *recall*, *precision* dan *f-measure* dari metode *Instance-based ontology matching* dalam menentukan keterhubungan *ontology* pada web semantik.

1.3. Tujuan

Tujuan dari penelitian tugas akhir ini adalah :

1. Menerapkan metode *Instance-based ontology matching* untuk menganalisa dan menentukan keterhubungan *ontology* pada web semantik.
2. Melakukan analisa keterhubungan *ontology* pada web semantik berdasarkan parameter *TopN* dan *Similarity Threshold* yang mempengaruhi nilai performansi *recall*, *precision* dan *f-measure* metode *Instance-based ontology matching*.

1.4. Batasan Masalah

Adapun batasan masalah dalam tugas akhir ini sebagai berikut :

1. Data input yang digunakan berekstensi *.owl atau *.rdf.
2. Implementasi menggunakan bahasa pemrograman Java.
3. Proses pencarian kemiripan *instance* (*instance similarity measure*) menggunakan Lucene API atau tidak mengimplementasikan dari awal.

1.5. Metodologi Penyelesaian Masalah

Dalam penelitian tugas akhir ini penulis menerapkan pendekatan sistematis untuk memecahkan masalah diatas. Adapun metode penelitian yang dilakukan adalah sebagai berikut :



Gambar 1-1 : Metode penyelesaian masalah

Gambar 1-1 mendeskripsikan tahapan-tahapan yang akan dilakukan dalam pelaksanaan tugas akhir ini, yaitu :

1. Studi literatur
 - a. Pencarian materi atau referensi yang dibutuhkan dan berhubungan dengan permasalahan seperti *Semantic Web*, *Ontology*, *Ontology Matching*, *Instance-Based Ontology Matching*, *Ontology Alignment*, *Precision*, *Recall* dan *F-measure*.
 - b. Pendalaman materi atau referensi yang diperoleh untuk membantu pemecahan masalah
2. Pengumpulan dan pembangunan data

Tahap ini merupakan proses membangun data yang akan digunakan sebagai inputan dalam sistem. Data input yang akan digunakan pada tugas akhir ini terdiri dari enam buah yakni empat buah data *ontology* dengan format *.owl yang akan dicari keterhubungan kelasnya dalam *ontology matching* dan dua buah data *ontology* dengan format *.rdf yang akan digunakan sebagai referensi terhadap pencarian nilai performansi *recall*, *precision* dan *f-measure* dari *output* yang dihasilkan oleh proses *ontology matching*. Informasi data yang akan digunakan dalam membangun data input ini berasal dari <http://gramediaonline.com> .
3. Perancangan sistem

Tahap penerapan metode *Instance-based ontology matching* yang mencari keterhubungan dua buah *ontology* yang berbeda dengan cara mencocokkan kelas-kelas yang terdapat pada dua buah *ontology* tersebut berdasarkan *Instance* yang dimilikinya. Inputan dari sistem berupa dua buah *ontology* yang berekstensi *.owl atau *.rdf. Kedua *ontology* tersebut kemudian dicari keterhubungannya dengan menggunakan metode *Instance-based ontology matching* yang telah diimplementasikan pada sistem. Hasil *output* dari proses pencocokan

ini berupa pemetaan keterhubungan kelas (*class mapping*) yang dibuat dalam format *ontology alignment* dengan ekstensi file *.rdf dan menampilkan nilai performansi *recall*, *precision* dan *f-measure*.

4. Implementasi sistem

Pada tahap ini sistem akan dibangun berdasarkan perancangan yang telah dibuat sebelumnya. Sistem akan dibangun menggunakan bahasa pemrograman Java dan berbasis desktop dengan memanfaatkan *library* Jena API, Lucene API dan *Alignment* API.

5. Pengujian

Tahap ini merupakan pengujian sistem yang akan dibangun dengan inputan *dataset* yang telah dibuat sebelumnya untuk mengukur pengaruh dari parameter *Top N* dan *Similarity Treshold* terhadap metode *Instance-based ontology matching*. Parameter *TopN* dan *Similarity Treshold* digunakan karena kedua parameter ini merupakan parameter yang berpengaruh dalam proses memperbanyak anotasi atau informasi *Instance* pada *ontology* yang akan berpengaruh juga terhadap metode *Instance-based ontology matching*. Pengujian akan dilakukan berdasarkan skenario yang telah dibuat.

6. Analisa

Analisa pengaruh dari parameter *TopN* dan *Similarity Threshold* terhadap nilai performansi *recall*, *precision* dan *f-measure* dari metode *Instance-based ontology matching* yang menentukan keterhubungan dua buah *ontology* terhadap hasil pengujian dari skenario yang telah dibuat.

7. Dokumentasi

Dokumentasi dilakukan untuk mencatat setiap hal penting dalam pelaksanaan tugas akhir ini.