SGPS: A Semantic Scheme for Web Service Similarity

Sourish Dasgupta
University of Missouri – Kansas City
5100 Rockhill Rd.
Kansas City, MO 64110
sdwb7@umkc.edu

Satish Bhat
University of Missouri – Kansas City
5100 Rockhill Rd.
Kansas City, MO 64110
ssbmvd@umkc.edu

Yugyung Lee
University of Missouri – Kansas City
5100 Rockhill Rd.
Kansas City, MO 64110
leeyu@umkc.edu

ABSTRACT
Today's Web becomes a platform for services to be dynamically interconnected to produce a desired outcome. It is important to formalize the semantics of the contextual elements of web services. In this paper, we propose a novel technique called Semantic Genome Propagation Scheme (SGPS) for measuring similarity between semantic concepts. We show how SGPS is used to compute a multi-dimensional similarity between two services. We evaluate the SGPS similarity measurement in terms of the similarity performance and scalability.

Categories and Subject Descriptors
H.3.5 [Online Information Services]: *Web-based services.*

General Terms
Measurement, Design, Theory

Keywords
Context, Web Services, Similarity, Semantics

1. INTRODUCTION
A major contribution of recent Web research is the hosting of a platform where cross-organizational computing devices can host loosely coupled component services in a dynamic manner to produce a desired outcome. However, the intrinsic complexity of many other environmental factors (like uncertainty in user preferences, user context, network health) makes it necessary for web service composition to be context-aware. We stress that apart from functional descriptions web services need context descriptions as well. Hence, it is important to formalize the context semantics of web services. At the heart of such semantic formalism lies the notion of service similarity. We propose a novel technique for measuring similarity between semantic concepts. The technique is inspired by the biological genome model and is based on the principle that semantic concepts are identified with three sets of properties: (i) properties that are inherited from ancestors, (ii) properties that are inherited but changed to more specific ones, and (iii) properties that are newly added. We have named this model as *Semantic Genome Propagation Scheme* (SGPS).

2. SGPS: FORMAL FOUNDATION
Semantic similarity measures have been studied extensively for the last two decades. Most previous researches focused on two perspectives: (i) similarity based on subsumption path lengths between two comparable concepts [1] and (ii) similarity based on information content of the parent of two comparable concepts [2]. However, the role of the properties contained by the concepts was largely underestimated [1-3]. This causes the problem of false positive. Although the semantic distances between a concept and its ancestors are usually greater than the distances with its siblings yet this is true only under certain circumstances. In SGPS a concept is defined as a vector in an application specific n-dimensional space. To compute the similarity between two concept vectors we introduce the concept of ***genes***. A single gene represents a particular dimension of a concept vector. These genes together form the genome of that vector. For an n-dimensional concept vector we will have *n* number of genes in the concept genome. Each gene may have several ***Genome Factors*** (denoted as *GF*) that characterize the gene. We can perceive these genome factors as concepts that help us to understand the gene completely. In this way we can represent a service concept by its functional profile and by its context. Hence, the service genome constitutes two sets of genes: (i) *functional gene* (Input, Output, Pre-conditions, and Result) (ii) *contextual gene* (Spatial Context, Temporal Context, Actor Context, Object Context, Background Context, and Information Context).

For any particular gene of a concept genome, the different constituent *GF*s may have different relative importance for that concept. Even for a particular service this relative importance may vary with time because of newly added importance to some relatively less important *GFs*. For any domain ontology a concept genome is passed along from a parent concept to its children concepts (***inheritance***). This genome may undergo two operations: ***mutation*** and ***addition***. However, in SGPS, *inheritance* is not only the acquiring of properties from one's parents but also adding its own properties (including mutated and added properties) to the inherited *GF*. A *mutation* occurs when some of the inherited *GF*s in a particular gene are changed to more specific concepts. There may be several such mutations in a single inheritance. The genome may also have *addition*s in its contents. An *addition* occurs when some completely new *GFs* are added to a particular gene. Just like mutations we may have multiple additions to a particular concept genome.

A concept genome may undergo extensive diversification as it is distributed down an ontology hierarchy. The higher the degree of this diversification the greater is the semantic distance between any two concept vectors. Any comparison between two concept vectors is done by singling out each of their corresponding genes and computing the semantic distance between the individual *GFs* therein. For any concept ontology the root concept is assigned a genome. Formally, a gene within such genome is represented as a collection of three *GF* sets: mutated, additional, and inherited. The first two sets are collectively called the ***Diversity Factor*** (*DF*) because they contribute to the genetic diversification of a concept vector. The

WWW 2009, April 20–24, 2009, Madrid, Spain.
ACM 978-1-60558-487-4/09/04.

third set (i.e. the inherited *GF*) is important because it provides the ancestral genes into a concept vector and thus increases the chance of similarity to another concept vector that has part of or all of the ancestral genes. Hence, we add this set to the *DF* to get a **C*umulative Diversity Factor*** (*CDF*) for each concept vector in the hierarchy. A particular gene may be distributed over all the three *GF* sets. Similarity between *GFs* is computed based on a generic upper ontology. For each of the *GF* hierarchies, we adopt and modify the prime number based encoding technique [4].

GF Distance (denoted as *GF-dist*) is used to measure the semantic distance between two *GFs* of a particular gene common in two concept genomes. The distance is used as an inverse weight for the semantic match that may be discovered between two such *GFs*. The *GF* distance is calculated by adding up all the primes in the codes of each of the *GF* concepts and then subtracting that from each other. Such a comparison is not valid if these concepts do not have any common hierarchy. Contribution by the *DF* to the diversification of the genes of a concept genome is the foremost. We argue that as mutation is just a change in the *inherited GFs*, the *mutated GF set* does not contribute much to this diversification in comparison to the *additional GF* set. In order to capture the importance factors, we use a pre-defined set of weights (α, β, γ) for additional, mutated and inherited *GF*s respectively such that $\alpha > \beta > \gamma$, that sum up to 1. We hereby propose two operators: (i) ***Gene Intersection*** (⊖) and (ii) ***Gene Union*** (⊕). An intersection may occur over the *GF x* for a particular *GF* set if either there is an exact semantic match between two *GFs* defined in the hierarchy *x* or there is a subsumption between them (tested by the finding whether their codes divide each other or not). For any two concept vectors c_i and c_j, the Gene Intersection over the *GF x* is given as follows:

$$CDF_x(c_i) \ominus CDF_x(c_j) = [\varphi^{mGF}{}_x(c_i) \cap^{GF}{}_x \varphi^{mGF}{}_x(c_j)] \cup [\varphi^{aGF}{}_x(c_i) \cap^{GF}{}_x \varphi^{aGF}{}_x(c_j)] \cup [\varphi^{iGF}{}_x(c_i) \cap^{GF}{}_x \varphi^{iGF}{}_x(c_j)]$$

where $\varphi^{mGF}{}_x$ is the *mutated GF* set for the *GF* x, $\varphi^{aGF}{}_x$ is the additional *GF* set for the *GF* x, $\varphi^{iGF}{}_x$ is the inherited *GF* set for the *GF x*, $\cap^{GF}{}_x$ is the intersection over the *GF x* for a particular *GF* set.

Similarly, for any two concepts c_i and c_j, the Gene Union over the *GF x* is given as follows:

$$CDF_x(c_i) \oplus CDF_x(c_j) = [\varphi^{mGF}{}_x(c_i) \cup \varphi^{mGF}{}_x(c_j)] \cup [\varphi^{aGF}{}_x(c_i) \cup \varphi^{aGF}{}_x(c_j)] \cup [\varphi^{iGF}{}_x(c_i) \cup \varphi^{iGF}{}_x(c_j)]$$

We now define the similarity of two concept vectors c_i and c_j over the *GF x* as follows:

$$Sim_x(c_i, c_j) = 1/(1+\bar{w}_x(c_i,c_j)) \times [|CDF_x(c_i) \ominus CDF_x(c_j)| / |CDF_x(c_i) \oplus CDF_x(c_j)|],$$

where, $|r^c{}_x|$ is the number of relations a concept *c* has with the *GF* category *x* and $\bar{w}_x(c_i, c_j) = | (|r^{ci}{}_x| / \sum_x |r^{ci}{}_x|)^2 - (|r^{cj}{}_x| / \sum_x |r^{cj}{}_x|)^2 |$ (the weight difference factor for a particular *GF* category x). The overall similarity between two concepts c_i and c_j over a particular gene '*z*' can be computed as follows:

$$z\text{-}Sim(c_i, c_j) = \sum^{n(z)}_x Sim_x(c_i, c_j)$$, where n(z) is the total number of *GF* categories that constitute the gene z.

The metric of ***Contextual Similarity*** measures the degree of closeness of two services in terms of their context information. We define the contextual similarity metric as follows:

$$Sim^{con}(s_i,s_j) = \sum_p 1/(1+\bar{w}_p(s_i,s_j)) \times [|CDF_p(s_i) \ominus CDF_p(s_j)| / |CDF_p(s_i) \oplus CDF_p(s_j)|]$$

For grouping functionally similar candidate services, we now define the **Functional Similarity** metric as follows

$$Sim^{func}(s_i,s_j) = \sum_q 1/(1+\bar{w}_q(s_i,s_j)) \times [|CDF_q(s_i) \ominus CDF_q(s_j)| / |CDF_q(s_i) \oplus CDF_q(s_j)|] \text{ and } \bar{w}_q(s_i, s_j) = | (|r^{si}{}_q| / \sum_q |r^{si}{}_q|)^2 - (|r^{sj}{}_q| / \sum_q |r^{sj}{}_q|)^2 |$$

3. EVALUATION

The experimental platform was a machine with a CPU cycle of 1.4 GHz and a RAM of 2 GB. The performance is computed in terms of the average execution time for each of the three different types of similarity measurement (functional, contextual, dependency) for a random collection of service node pairs over 10 service networks. Computation of functional similarity is faster than that of contextual similarity. This is because contextual similarity has the highest number of intersection computation and hence, the highest summation overload for the contextual gene.

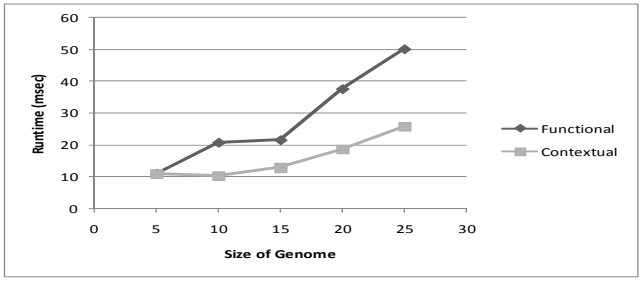

Figure 1. SGPS Similarity Performance

4. CONCLUSION

In this paper we have discussed a novel technique for computing similarity between web services. We have evaluated its performance with respect to the service network size and the genome size of the service nodes.

5. ACKNOWLEDGEMENT

This work has been partially supported by National Science Foundation (IIS #0742666).

6. REFERENCES

[1] R. Rada, et al., "Development and application of a metric on semantic nets", IEEE Transactions on Systems, Man, and Cybernetics, p. 19, 1989.

[2] P. Resnik, "Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language", Journal of Artificial Intelligence Research, vol 11, p. 95–130, 1999.

[3] G. Hirst, D. StOnge, "Lexical chains as representation of context for the detection and correction of malaproprisms, in WordNet: An electronic lexical database", MIT Press. p. 305–332, 1998.

[4] D. Preuveneers, Y. Berbers, "Prime numbers considered useful: ontology encoding for efficient subsumption testing". Department of Computer Science, K.U.Leuven, Leuven, Belgium, 2006.