

CAPITAL UNIVERSITY OF SCIENCE AND
TECHNOLOGY, ISLAMABAD



**Investigating Protein Semantic
Similarity Measurement and its
Correlation with Sequence
Similarity**

by

Najmul Ikram

A thesis submitted in partial fulfillment for the
degree of Doctor of Philosophy

in the

Faculty of Computing

Department of Computer Science

March 2018

Copyright © 2017 by Najmul Ikram

All rights reserved. No part of this thesis may be reproduced, distributed, or transmitted in any form or by any means, including photocopying, recording, or other electronic or mechanical methods, by any information storage and retrieval system without the prior written permission of the author.

This work is dedicated to my beloved father and my caring mother.

Investigating Protein Semantic Similarity Measurement and its Correlation with Sequence Similarity

By
Najmul Ikram
(PC103999)

Foreign Evaluator 1
Dr. Finn Drablos
Norwegian University of Science and Technology
Trondheim, Norway

Foreign Evaluator 2
Dr. Mexhid Ferati
Oslo and Akershus University
Oslo, Norway

Dr. Muhammad Abdul Qadir
(Thesis Supervisor)

Dr. Nayyer Masood
(Head, Department of Computer Science)

Dr. Muhammad Abdul Qadir
(Dean, Faculty of Computing)

DEPARTMENT OF COMPUTER SCIENCE
CAPITAL UNIVERSITY OF SCIENCE AND TECHNOLOGY
ISLAMABAD
2018



**CAPITAL UNIVERSITY OF SCIENCE & TECHNOLOGY
ISLAMABAD**

Expressway, Kahuta Road, Zone-V, Islamabad
Phone: +92-51-111-555-666 Fax: +92-51-4486705
Email: info@cust.edu.pk Website: <https://www.cust.edu.pk>

CERTIFICATE OF APPROVAL

This is to certify that the research work presented in the thesis, entitled “**Investigating Protein Semantic Similarity Measurement and its Correlation with Sequence Similarity**” was conducted under the supervision of **Dr. Muhammad Abdul Qadir**. No part of this thesis has been submitted anywhere else for any other degree. This thesis is submitted to the **Department of Computer Science, Capital University of Science and Technology** in partial fulfillment of the requirements for the degree of Doctor in Philosophy in the field of **Computer Science**. The open defence of the thesis was conducted on **15 February, 2018**.

Student Name : Mr. Najmul Ikram
(PC103999)

The Examination Committee unanimously agrees to award PhD degree in the mentioned field.

Examination Committee :

(a) External Examiner 1: Dr. Amir Ali Abbasi,
Associate Professor
NCB, QAU, Islamabad

(b) External Examiner 2: Dr. Jamil Ahmed,
Assistant Professor
RCMS, NUST, Islamabad

(c) Internal Examiner : Dr. Muhammad Tanvir Afzal,
Associate Professor,
CUST, Islamabad

Supervisor Name : Dr. Muhammad Abdul Qadir,
Professor,
CUST, Islamabad

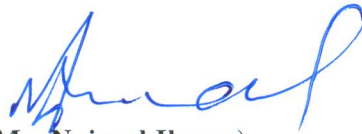
Name of HoD : Dr. Nayyer Masood,
Professor,
CUST, Islamabad

Name of Dean : Dr. Muhammad Abdul Qadir,
Professor,
CUST, Islamabad

AUTHOR'S DECLARATION

I, **Mr. Najmul Ikram (Registration No. PC103999)**, hereby state that my PhD thesis titled, '**Investigating Protein Semantic Similarity Measurement and its Correlation with Sequence Similarity**' is my own work and has not been submitted previously by me for taking any degree from Capital University of Science and Technology, Islamabad or anywhere else in the country/ world.

At any time, if my statement is found to be incorrect even after my graduation, the University has the right to withdraw my PhD Degree.



(**Mr. Najmul Ikram**)

Dated: February, 2018

Registration No : PC103999

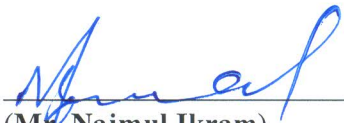
PLAGIARISM UNDERTAKING

I solemnly declare that research work presented in the thesis titled “**Investigating Protein Semantic Similarity Measurement and its Correlation with Sequence Similarity**” is solely my research work with no significant contribution from any other person. Small contribution/ help wherever taken has been duly acknowledged and that complete thesis has been written by me.

I understand the zero tolerance policy of the HEC and Capital University of Science and Technology towards plagiarism. Therefore, I as an author of the above titled thesis declare that no portion of my thesis has been plagiarized and any material used as reference is properly referred/ cited.

I undertake that if I am found guilty of any formal plagiarism in the above titled thesis even after award of PhD Degree, the University reserves the right to withdraw/ revoke my PhD degree and that HEC and the University have the right to publish my name on the HEC/ University Website on which names of students are placed who submitted plagiarized thesis.

Dated: February, 2018



(Mr. Najmul Ikram)
Registration No. PC103999

Acknowledgements

All Praises be to Allah Almighty who enabled me to complete this research successfully and my utmost respect to His last Prophet (SAW). I acknowledge the help and guidance of my supervisor Dr. Abdul Qadir throughout my PhD research. Without his direction, it was not possible. The input from members of CDSC research group during group meetings was very useful, especially from Dr. Tanvir Afzal. I acknowledge the help of Mr. Haroon Khan in identifying protein pairs for evaluation experiments. The observations of Pakistani and foreign evaluators were highly valuable to improve the dissertation. Staff of CUST, especially Graduate Office was very cooperative. But my PhD was not possible without the cooperation of my family, who supported me throughout this venture. My special thanks to all friends for their good wishes and cooperation.

List of Publications

It is certified that following publications have been made/ submitted out of the research work that has been carried out for this thesis.

Journal Papers

1. Ikram N., Qadir M.A., Afzal M.T. (2017) Investigating correlation between protein sequence similarity and semantic similarity using gene ontology annotations. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*. DOI: 10.1109/TCBB.2017.2695542. I.F 1.955
2. Ikram N., Qadir M.A. (2011) Algorithms for the evaluation of ontologies for extended error taxonomy and their application on large ontologies. *J. UCS*, 17(7), 1005-1020. I.F 0.466

Conference Papers

1. Ikram N., Qadir M.A. (2010) Evaluation of ontologies for Extended Error Taxonomy: algorithms and their application on large ontologies. *Proceedings of I-KNOW'10: 10th international conference on knowledge management and knowledge technologies*.

Abstract

Protein sequence similarity is commonly used to compare proteins, and to search for proteins similar to a query protein. With the growing use of biomedical ontologies, especially Gene Ontology (GO), semantic similarity between ontology terms, proteins and genes is getting attention of researchers. Protein semantic similarity measurement has many applications in bioinformatics, including protein function prediction and protein-protein interactions. Semantic similarity measures were proposed by Resnik, Jiang and Conrath, and Lin. Recent measures include Wang and AIC.

The question whether the semantic similarity has a strong correlation with sequence similarity, has been addressed by some authors. It has been reported that such correlation exists, and it has been used for the evaluation of semantic similarity computation methods as well as for protein function prediction. We investigate the correlation between semantic similarity and sequence similarity using graphs, Pearson's correlation coefficient and example proteins. We find that there is no strong correlation between the two similarity measures. Pearson's correlation coefficient is not sufficient to explain the nature of this relationship, if not accompanied by graph analysis. We find that there are several pairs with low sequence similarity and high semantic similarity, but very few pairs with high sequence similarity and low semantic similarity. Interestingly, the correlation coefficient depends only on the number of common GO terms in proteins under comparison.

We propose a novel method SemSim for semantic similarity measurement. It addresses the limitations of existing methods, and computes similarity in two steps. In the first step, SimGIC like approach is used where contribution of common ancestors is divided by contribution of all ancestors. In the second step, we use two new factors: Specificity computed from ontology based information content, and Uniqueness computed from annotation based information content. The final result, after applying these two factors, makes clear distinction between the generalized and specialized terms. We conducted experiments on protein pairs having evidence of high similarity, and the ones having evidence of low similarity. Experiments show that SemSim performs better than the previous measures in both cases.

When semantic similarity is used for searching proteins from large databases, the speed issue becomes significant. To search for proteins similar to a query protein

having m annotations, from the database of p proteins, $p \times m \times n \times g$ comparisons would be required. Here n is the average annotations per protein, g is the complexity of GO term similarity computation algorithm, and it is assumed that each term of one protein is compared with each term of the other. We propose a method SimExact that is suitable for high speed searching of semantically similar proteins. Although SimExact works on common terms only, our experiments show that it gives correct results required for protein semantic searching. SimExact can be used as a pre processor, generating candidate list for the existing methods, which proceed for further computation. Such arrangement will gain high speed while retaining the accuracy of the given method. We provide online tool that generates a ranked list of the proteins similar to a query protein, with a response time of less than 8 seconds in our setup. We use SimExact to search for protein pairs having high disparity between semantic similarity and sequence similarity. SimExact makes such searches possible, which would be NP-hard otherwise.

Contents

Author’s Declaration	iii
Plagiarism Undertaking	iv
Acknowledgements	v
List of Publications	vi
Abstract	vii
List of Figures	xii
List of Tables	xiv
1 Introduction	1
1.1 Background	1
1.2 Statement of Problem and Research Questions	5
1.3 Methodology	7
1.4 Selection of Data Sets	8
1.4.1 Dataset 1	8
1.4.2 Dataset 2	9
1.4.3 Dataset 3	9
1.4.4 Dataset 4	10
1.4.5 Dataset 5	10
1.4.6 Dataset 6	10
1.4.7 Dataset 7	11
1.4.8 Dataset 8	11
1.5 Organization	11
2 Literature Review	12
2.1 Semantic Similarity Measures	12
2.1.1 Information Content	12
2.1.2 Resnik	13
2.1.3 Jiang Conrath	14

2.1.4	Lin	14
2.1.5	SimUI and SimGIC	14
2.1.6	GraSM	15
2.1.7	Wang	15
2.1.8	AIC	16
2.1.9	SSA	17
2.1.10	InteGO	18
2.1.11	Mazandu and Mulder	19
2.1.12	GOSemSim	19
2.1.13	Seco et al.	20
2.1.14	Yang et al.	21
2.1.15	Bien et al.	21
2.1.16	HRSS	23
2.1.17	GAP	24
2.1.18	Santos et al.	25
2.1.19	Measuring Semantic Similarity between Proteins	26
2.2	Correlation between Semantic Similarity and Sequence Similarity	26
2.2.1	Lord et al.	27
2.2.2	Louie et al.	28
2.2.3	Lee et al.	28
2.2.4	Joshi and Xu	29
2.2.5	Anscombe	29
2.2.6	Pesquita et al.	30
2.2.7	Other	31
2.3	Evaluation of Semantic Similarity Measures	31
2.4	Critical Analysis of Existing Methods	35
2.4.1	Motivating Case Study	35
2.4.2	Old Measures	36
2.4.3	Recent Measures	37
2.5	Protein Sequence Similarity	40
2.6	Conclusion	41
3	Proposed Methods	43
3.1	Semantic Similarity Measure: SemSim	43
3.2	Protein Functional Similarity Method - SimExact	50
3.3	Implementation and Experimental Environment	53
3.4	Protein Query Algorithm/ Tool	55
3.4.1	Searching Protein Pairs	58
3.4.2	Sequence-Semantic Conflict	60
3.4.3	Similarity Inferred from Sequence Similarity	61
4	Comparison And Correlation Between Similarity Measures	62
4.1	Experiment 4.1: Distribution of Similarity Measures	62
4.2	Experiment 4.2: Correlation Between Similarity Measures	65

4.3	Experiment 4.3: Rank Correlation Between Similarity Measures . . .	68
4.4	Experiment 4.4: Correlation Between Semantic Similarity and Semantic Similarity	69
4.5	Experiment 4.5: Effect of Common GO Terms	72
4.6	Experiment 4.6: Correlation at Sub Ontology Level	74
4.7	Discussion	75
5	Results Of Novel Similarity Measures	78
5.1	Experiment 5.1: Evaluation Using Highly Similar Proteins	79
5.2	Experiment 5.2: Evaluation Using Highly Dissimilar Proteins	80
5.3	Experiment 5.3: Evaluation With General GO Terms	81
5.4	Experiment 5.4: Comparing SemSim With Existing Measures	82
5.5	Discussion	89
5.6	Experiment 5.5: Results of Protein Query Tool	89
5.7	Experiment 5.6: Searching Protein Pairs	90
6	Investigation and Discussion	94
6.1	Experiment 6.1: Investigation on Weak Correlation	96
6.2	Discussion	98
6.3	Research Questions Summary	101
7	Conclusion	103
	Bibliography	105
	Appendices	109

List of Figures

1.1	A Part of Gene Ontology.	4
1.2	A schematic diagram of Gene Ontology and protein annotations. . .	6
2.1	An example taxonomy of store items.	37
2.2	An example taxonomy of things.	38
2.3	Blosum62 matrix.	41
3.1	A hypothetical ontology.	49
3.2	Algorithm for processing protein query using SimBinary	56
3.3	Modified algorithm for processing protein query	57
3.4	SQL implementation of protein query processing	57
3.5	Screen shot of protein query tool SemQuery showing SimExact sim- ilarity	59
3.6	Screen shot of protein query tool SemQuery showing SimExact and other similarities	60
4.1	Frequency distribution of GO term semantic similarity values for various methods.	64
4.2	Frequency distribution of protein semantic similarity for various methods.	66
4.3	Scatter plots of GO term similarities: Wang against Resnik GraSM, AIC against SimGIC.	67
4.4	Scatter plot of protein semantic similarity against sequence similar- ity for Resnik, Lin, SimGIC and AIC for Dataset 1.	71
4.5	Scatter plots of SimGIC, Lin against sequence similarity for non- IEA annotations.	71
4.6	Scatter plot of protein semantic similarity against sequence similar- ity for SimBinP. Correlation: 0.72.	72
4.7	Scatter plot of SimGIC against sequence similarity for Dataset 4. Correlation: 0.	73
4.8	Scatter plot of SimGIC against sequence similarity for Dataset 5. Correlation: 0.69.	74
4.9	Scatter plot of Jiang Conrath Fractional, AIC Fractional against sequence similarity	74
5.1	Similarity between pairs of globins by various methods.	80
5.2	Similarity between protein pairs of Trad et al. by various methods. .	81

5.3	Similarity between protein pairs having generalized common terms.	82
5.4	Scatter plot of SemSim A against Resnik GraSM.	84
5.5	Scatter plot of SemSim B against Resnik GraSM.	84
5.6	Scatter plot of SemSim A against Wang.	84
5.7	Scatter plot of SemSim A against AIC.	85
5.8	Scatter plot of protein semantic similarity against sequence similarity for SemSim B.	85
5.9	Scatter plot of SimBinP against Jiang Conrath GraSM.	90
5.10	Graph of time consumed for protein query processing.	91
5.11	Time consumed for protein query processing with full computation.	91
5.12	Scatter plot of semantic similarity against sequence similarity for searched pairs.	92
6.1	Distribution of sequence similarity scores between protein pairs belonging to distant GO terms.	99
6.2	Distribution of sequence similarity scores between protein pairs belonging to sibling GO terms.	99
6.3	Distribution of sequence similarity scores between protein pairs sharing a common GO term.	100
6.4	Bar Diagram of SISS of leaf GO terms with themselves.	100
6.5	Distribution of SISS of leaf GO terms with themselves.	100

List of Tables

2.1	Approaches used for evaluation of semantic similarity measures.	34
2.2	Similarity computed by existing methods.	35
3.1	Assumed ICs of concepts of Figure 2.3.	47
3.2	Similarity between generalized concepts by SimSim A.	47
3.3	Similarity computed by SemSim and existing methods (r=6, Figure 3.1).	50
3.4	Database table ‘TermPair’.	54
4.1	GO calibration of semantic similarity measures.	65
4.2	Calibration of protein semantic similarity for various measures.	65
4.3	Correlation between various semantic similarity measures.	67
4.4	Spearman’s rank correlation between similarity ranks given by semantic similarity measures.	69
4.5	Top ten terms similar to GO:0000053 with their similarity (rank).	69
4.6	Pearson’s correlation coefficient between sequence similarity and semantic similarity.	75
5.1	Globins having high secondary and tertiary structural similarity.	79
5.2	Similarity computed by SemSim and existing methods.	87
5.3	Protein similarity computed by SemSim and existing methods.	88
5.4	Protein pairs having low sequence similarity and high semantic similarity.	93
6.1	Relationship between research questions and experiments.	102

Chapter 1

Introduction

1.1 Background

Proteins are the functional units of life, taking part essentially in every structure and activity of life. They are the basic building materials for living cells, and consist of 20 amino acids combined in different ways, which makes more than 100,000 different proteins in a human body.

The sequencing of genomes from various organisms and X-Ray structure analysis has brought to us a large amount of data about the sequences and structures of thousand of proteins. This information can be used for biological research only if one can extract functional insight from it. Bioinformatics uses the analysis of protein sequences and structures to help annotate the genome, to understand their function, and to predict their structure when only sequence is known. Bioinformatics methods are used in fundamental research on theories of evolution as well as in more practical works on protein design. Algorithms and techniques used in these areas include sequence alignments, secondary structure prediction, functional classification of proteins, threading and modeling of distantly-related homologous proteins, modeling the progress of protein expression and so on.

Homology means existence of two or more proteins that evolved from a common ancestor. It can be used to predict a gene's function. Sequence alignment is used to

find homology between two polypeptide chains. Finding out sequence alignments can help detect evolutionary origins and trace back the function, structure, and mechanism of a genome. Repeated motifs can be detected by aligning a sequence with itself. Newly discovered protein sequences can be matched by inputting the sequence to a system having large database of previously known proteins using some alignment tool, like BLAST (Basic Local Alignment Search Tool). Using BLAST, homology of a newly sequenced protein can be found, and function and tertiary structure of a protein can be predicted.

Proteins demonstrate diverse sequence-structure relationships. Understanding protein similarity relationships is vital for the annotation of gene products. Proteins with high sequence similarity and high structural similarity tend to have some functional similarity and evolutionary relationships. However, examples of proteins having low sequence similarity and high structural and functional similarity exist.

Sequence similarity of proteins means comparing them based on what they appear to be. There are other ways to measure protein similarity. Many bioinformatics applications would need to compare proteins based on what they do (function). There is an increasing use of ontology (e.g. Gene Ontology) in bioinformatics research. Ontology models a domain of interest. It defines concept hierarchies, object properties, data type properties, and many other constructs that enable automated agents to understand and process the information according to the semantics of the domain. For example in biology, ontology can organize concepts (biological terms) as taxonomy where each term is inherited from some parent concept (is a relationship). Database technology is normally used to store ontology in an efficient manner, where Structured Query Language (SQL) is used to retrieve information.

The Gene Ontology (GO) [1] aims at fulfilling the requirements of research community for correct descriptions of gene products (genes, proteins). It consists of three vocabularies that describe gene products in terms of biological processes, cellular components and molecular functions in a species-independent way. It defines

over 38000 biological terms and the relationship (is-a, part-of) between pairs of terms. For example, term protein C-terminus binding (GO:0008022) is a protein binding (GO:0005515), which is a molecular function (GO:0003674). This forms a large taxonomy of terms (concepts in the context of ontology) where each term can be traced for its inheritance of characteristics. Terms can have multiple parents, so the Gene Ontology is a Directed Acyclic Graph (DAG) rather than a tree. An example taxonomy from Gene Ontology is shown in Figure 1.1. It shows parent-child relationship between terms, and that a term may have multiple parents.

Protein annotation means assigning a protein to one or more GO terms. This means to assert that a protein takes part in a certain biological process, performs a molecular function, and is found in a cellular component. Annotation database is a mapping of protein ids to GO term ids in the form of a many-to-many relationship. Annotation of proteins with GO terms makes a valuable knowledge base that integrates the knowledge of many scientists on a single platform which can be understood by computers. Research organizations are constantly adding to this valuable database.

The combination of Gene Ontology and annotations offers a new opportunity for computation of protein similarity based upon their function, often called semantic similarity. Protein semantic similarity takes into account the role of proteins in biological activities. To enable such comparison, we need to compare terms from Gene Ontology that annotate the proteins to be compared. The location of a concept (GO term) in the Gene Ontology graph holds biological meaning (semantics) of that concept, which is understandable to computer programs. A number of researchers have proposed methods for measuring semantic similarity between the concepts (terms) of Gene Ontology [2]. Among the early proposed methods are Resnik [3], Jiang and Conrath [4], and Lin [5].

To compute semantic similarity between two proteins, the annotated GO terms of both proteins are compared. The schematic diagram of Figure 1.2 shows the elements involved in semantic similarity measurement in a generic way. Gene Ontology forms taxonomy of terms for example, g_1 , g_2 etc. To measure semantic



FIGURE 1.1: A Part of Gene Ontology.

similarity between two proteins p_1 and p_2 , each GO term annotated with p_1 is compared with each GO term annotated with p_2 . Then the similarity between p_1 and p_2 is computed using one of three aggregation approaches: average, maximum and best match average [2]. In average approach, the similarity between p_1 and p_2 is the average of all GO term similarity values. In maximum approach, the protein similarity is the maximum of the term similarity values. Best Match Average (BMA) is a balance between these two approaches. For each term in one set, the best matching term in the other set is determined. Protein similarity is computed by taking average of the similarity values of the best matching pairs.

If we take several protein pairs, and compute sequence similarity and semantic similarity between each pair, we can check the correlation between the two measures. Pearson's correlation coefficient is commonly used to determine the extent of correlation between two variables, where 0 means no correlation and 1 means perfect correlation.

Rank correlation coefficients are also used to determine the relationship between two variables. They measure the extent to which, when one variable increases, the other also increases, without being sensitive to how much each increases. For an example, let us consider the (x, y) pairs: $(1, 5)$, $(2, 20)$, $(3, 200)$, $(4, 2000)$, $(5, 2001)$. It can be seen that when x increases, y always increases, although the increments are not uniform. Rank correlation between x and y will be 1 whereas, Pearson's correlation will be 0.89. As the name indicates, rank correlation works on the ranks of data which, in the above example are $(1, 1)$, $(2, 2)$, $(3, 3)$, $(4, 4)$, $(5, 5)$. Spearman's correlation coefficient and Kendall's correlation coefficient are commonly used rank correlation coefficients.

The nature of this relationship has more important implications in this area. It can help us to answer a basic question: how the sequence of a protein can determine its biological function. Moreover, the current protein querying can be improved significantly. A very high correlation would suggest the use of semantic based query answering tools as alternate to the current sequence matching tools. A low correlation would make the semantic similarity an independent measure to be explored further.

1.2 Statement of Problem and Research Questions

Semantic similarity measuring methods frequently disagree with one another. There is no established criterion for the evaluation of these methods. Correlation with sequence similarity has been used as an evaluation criterion in some

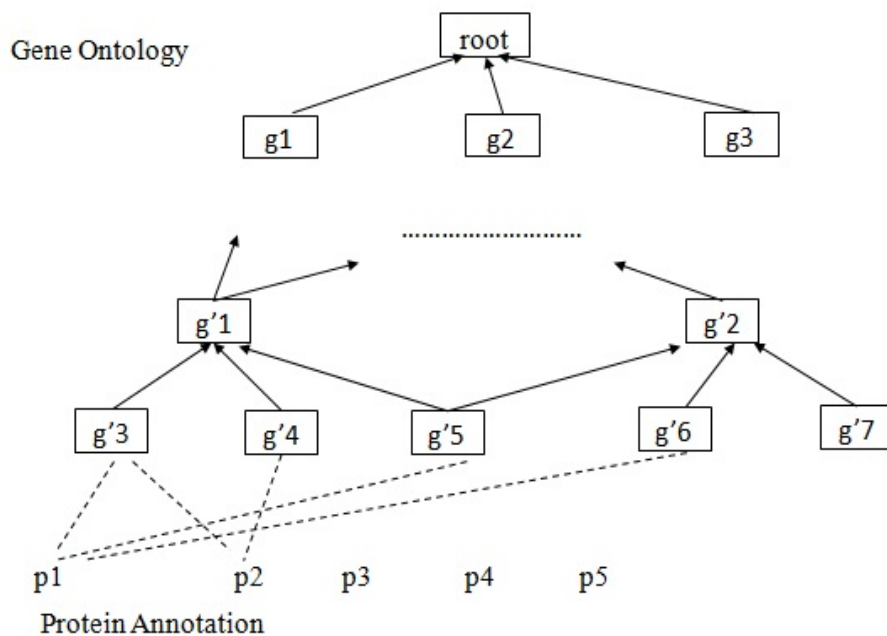


FIGURE 1.2: A schematic diagram of Gene Ontology and protein annotations.

studies. Assumption of a significant correlation between semantic similarity and sequence similarity is used for the evaluation. However, there is no thorough experimental evidence of the significant correlation, and there are studies reporting on the contrary. Tools to search semantically (functionally) similar proteins do not exist, and the current methods are not fast enough to support such search.

To solve the above problems, we need to answer the following questions, which are the research questions of this dissertation:

1. RQ1: Are the semantic similarity measuring methods consistent? If not, then what are the reasons? If various methods give significantly different results, then there is a possibility that some of them have problems.
2. RQ2: What is the relationship between semantic similarity of protein pairs and their sequence similarity? There are varying opinions about this relationship in the literature, and therefore a detailed study is required.
3. RQ3: Can the correlation with sequence similarity be used as a standard criterion for evaluation of novel semantic similarity measures or there is a

need of some other benchmark? This correlation has been used for evaluation of similarity methods, but there are views on the contrary.

4. RQ4: Can we devise a semantic similarity method which gives results consistent with the benchmark / dataset provided by domain experts? If the presence of problems in the existing methods is established, then there would be a need to fix the problems, and design new method.
5. RQ5: Can a software tool be implemented to measure the semantic similarity with acceptable accuracy and in a faster way (in real time)? Given the large protein databases, the problem of searching proteins similar to a query protein would require fast comparisons.
6. RQ6: Can the pattern of sequence-semantic similarity graph be used to understand or investigate the behavior of proteins? After assessing the correlation between semantic similarity and sequence similarity, we would need to explore the reasons of such relationship.

1.3 Methodology

Following are the major steps of methodology adopted in the research:

1. Literature review to study the existing semantic similarity measures, and the relationship between semantic similarity and sequence similarity
2. Implementation of existing methods for similarity measurement in order to have a test bed for experimentation
3. Selection of data set for experiments
4. Performing experiments by using the test bed
5. Result analysis and evaluation of correlation between sequence and semantic similarity. First, the existing semantic similarity measures were compared and correlated with one another to observe their agreement or disagreement.

Then the correlation between semantic and sequence similarities were studied.

6. Formulation of new semantic similarity measures SemSim and SimExact. SemSim incorporates specificity and uniqueness of GO terms in accordance with the semantics of the ontology.
7. Experiments and comparison of SemSim and SimExact with existing similarity measures

Steps 1 to 4 are going to answer the research question 1. Step 5 is going to be helpful in answering the research question 2. The philosophy behind the selection of data for the evaluation of correlation explains the answer to question 3. We need a data set which is annotated by domain experts in order to evaluate the similarity methods. Based upon the results of experiments and our understanding of the problem, we devised new methods to find semantic similarity. Step 6 answers question 4 to 6.

1.4 Selection of Data Sets

We carried out experiments with multiple datasets to strengthen the conclusions. In order to minimize the bias, some data sets are selected from the published literature. When there was a need of a particular type of data set according to the requirements of an experiment, then the data set is selected randomly, or generated by a program using random pattern. The explanation of the selection of each dataset is given in the following sections:

1.4.1 Dataset 1

Protein pairs for experiments on semantic-sequence correlation were taken from CESSM (<http://xldb.di.fc.ul.pt/tools/cessm/downloaddata.php>) [6]. CESSM data

is particularly created for experiments related with protein similarity measurement. The proteins of this data set have reasonable number of GO annotations, and are therefore suitable for our experiments. Every protein is manually annotated with at least one GO term in all three GO sub ontologies with IC of at least 0.5. The proteins BLAST e-values for both directions are below 10^{-4} . The dataset consists of 13430 protein pairs. Some later works on semantic similarity measures e.g. Couto et al. [7] and Alvarez and Yan [8] have used this data, too. Use of dataset from an independent source helps make our experimental results unbiased. GO term pairs are generated during the computation of semantic similarity for protein pairs. They consist of GO terms annotated by CESSM proteins. These pairs are used for experiments on GO terms.

1.4.2 Dataset 2

Non-IEA (IEA means Inferred from Electronic Annotation) or manually assigned annotations are considered to be most reliable, because they are established by experts on the basis of experiments. Experiments with these annotations are particularly useful for our research. To perform such experiment, we required additional pairs to supplement CESSM dataset. This is because many proteins in CESSM dataset did not have manual annotations, whereas we required those with manual annotations. Our program generated protein pairs having varying values of sequence similarity, and varying number of common GO terms in the non-IEA annotations. This dataset consists of 4270 protein pairs.

1.4.3 Dataset 3

If different methods produce different similarity values, it is important to check if these methods rank GO terms similar to a given term in the same order or not. This dataset is prepared to evaluate the correlation between the semantic similarity rankings of different methods. That is, if method A rank of similarity is same as that for method B. It is a set of GO term pairs where a reference

term is picked randomly, and then n most similar terms with the reference term (according to each method) are selected. We used $n=50$, while total number of pairs is 10800.

1.4.4 Dataset 4

This dataset has been built based on Dataset 1. For experiments concerning with the effect of common terms, two more datasets were required: one with zero common terms, and the other with non-zero common terms. In CESSM dataset, most of pairs have at least one common GO term. However, by shuffling the CESSM proteins, Dataset 4 could easily be generated, having zero common terms. The purpose of this dataset is to study the correlation in the absence of common terms. These are 1771 protein pairs.

1.4.5 Dataset 5

Dataset 5 has been built based on Dataset 1. It is taken from CESSM pairs, having at least one common term. This was required to study the effect of common terms. It consists of 9355 protein pairs.

1.4.6 Dataset 6

It consists of pairs of globins from Lesk and Chothia [9]. These protein pairs have evidence of having very high function similarity. This dataset is used to evaluate our method, and compare it with the existing methods to compute semantic similarity. These are 9 globins, making 72 protein pairs.

1.4.7 Dataset 7

It consists of protein pairs from Trad et al. [10]. The protein pairs have evidence of having very low function similarity. They are used in our novel evaluation method. These are 9 protein pairs.

1.4.8 Dataset 8

This dataset has been built based on Dataset 1. Its purpose is to test the behaviour of similarity methods on generalized GO terms. To produce this dataset, CESSM protein pairs having 25% to 75% common terms were sorted on IC value in descending order. Then top 100 pairs were selected. This gives us the protein pairs sharing most generalized (low IC) GO terms.

1.5 Organization

Here is a description of the rest of this dissertation. Chapter 2 gives a review of the existing literature about semantic similarity measurement, and its correlation with sequence similarity. It points out limitations of the existing methods. Chapter 3 describes our proposed similarity methods and reasoning behind them. The tools developed for protein query answering are also described. Chapters 4, 5 and 6 furnish the results of our experiments and the analysis of these results. Chapter 4 is about comparison of semantic similarity measures and their correlation with sequence similarity. Results of our novel similarity measures SemSim and SimExact are given in Chapter 5. Comparison of our method with existing similarity measures is also given. The results are traced and checked to find out which measure performs better than others. Chapter 6 further investigates why the correlation between semantic similarity and sequence similarity is not strong. Chapter 7 gives the final conclusions of our research.

Chapter 2

Literature Review

2.1 Semantic Similarity Measures

Gene Ontology is extensively covered in the recent literature of bioinformatics as well as semantic computing. A number of authors have contributed in the area of semantic similarity and proposed measurement methods for it. Some well known measures are described here.

2.1.1 Information Content

Many similarity measures are based on Information Content (IC), a measure of specificity of a concept in ontology. Generic concepts have less information while specific concepts have more information. Frequency of a concept in Gene Ontology is the number of gene products (genes or proteins) annotated with the concept (GO term) and its descendants. The probability that a concept occurs is computed from its frequency $Freq$ divided by the maximum frequency (frequency of the root concept) found in the ontology.

$$Prob(c) = \frac{Freq(c)}{MaxFreq} \quad (2.1)$$

The information content IC of a concept is defined as

$$IC(c) = -\log(Prob(c)) \quad (2.2)$$

Specialized concepts will have high IC , while generalized concepts will have lower IC . The leaf concepts (GO terms) with least protein annotations will have highest IC , whereas the root concept will have lowest IC .

2.1.2 Resnik

Resnik [3] defined semantic similarity measure (SimRes) between two concepts c_1 and c_2 as the IC of the most informative common ancestor.

$$SimRes(c_1, c_2) = MAX(IC(c)|c \text{ is ancestor of } c_1 \text{ and } c_2) \quad (2.3)$$

To find similarity between concepts c_1 and c_2 , we would move upwards from c_1 and c_2 and stop on the concept where the two paths meet (the common ancestor). This is going to be the most informative common ancestor (MICA) of c_1 and c_2 . The IC of this concept measures the similarity between c_1 and c_2 . For two sibling terms, parent term is the immediate common ancestor. Grand parent and above terms are all common ancestors, but the immediate parent, having least IC , is the MICA. Two distantly located concepts will meet at a high level, giving low similarity, whereas closely located concepts will meet at a lower level, giving higher similarity.

The minimum Resnik similarity is 0, which is between such concepts that have only one common ancestor that is the root concept. Maximum value of similarity is not fixed, and depends upon the ontology size and annotations.

2.1.3 Jiang Conrath

Jiang and Conrath [4] computed the similarity from the semantic distance between two concepts. The distance is defined as

$$DistJiang(c_1, c_2) = IC(c_1) + IC(c_2) - 2 \cdot SimRes(c_1, c_2) \quad (2.4)$$

Jiang Conrath similarity (SimJiang) is defined as the reciprocal of semantic distance

$$SimJiang(c_1, c_2) = \frac{1}{DistJiang(c_1, c_2) + 1} \quad (2.5)$$

The similarity value ranges from 0 to 1.

2.1.4 Lin

Lin [5] defined semantic similarity as

$$SimLin(c_1, c_2) = \frac{2 \cdot SimRes(c_1, c_2)}{IC(c_1) + IC(c_2)} \quad (2.6)$$

The IC of the most informative common ancestor (MICA) is divided by the average of the IC s of the two concepts to be compared. Lin's similarity between two close terms located near root will be higher as compared to Resnik similarity. Moreover, the similarity of a term with itself will be 1, whether the term is general or specific. The similarity value ranges from 0 to 1.

2.1.5 SimUI and SimGIC

Gentleman [11] proposed SimUI and Pesquita et al. [12] proposed simGIC, using the Jaccard index.

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (2.7)$$

SimUI and SimGIC similarity is obtained by dividing the sum of contribution from common ancestors by the sum of contribution from all ancestors of the two terms. SimUI simply counts the ancestor terms, while SimGIC takes IC of each ancestor as the contribution. The similarity ranges from 0 to 1.

2.1.6 GraSM

Couto et al. [13] made adjustments in the above three methods and proposed Graph-based Similarity Measure (GraSM) technique. They suggested that the similarity measurement techniques should not consider the ontology, especially Gene Ontology, as a tree. Since many concepts in Gene Ontology have multiple parents, the ancestors of a concept make a complex formation. They use the notion of disjunctive common ancestors and take average of the IC of these common ancestors. They applied their adjustment on the three existing methods.

Two common ancestors are disjunctive if there are independent paths from both ancestors to the concept. GraSM takes average of IC values of the disjunctive common ancestors of the two concepts to define their shared information.

$$ShareGraSM(c_1, c_2) = Avg\{IC(a) | a \in CommonDisjAnc(c_1, c_2)\} \quad (2.8)$$

Resnik similarity GraSM (SimResG) is equal to ShareGraSM. In (Eq 2.4) and (Eq 2.6), SimRes is replaced with SimResG to compute SimJiangG and SimLinG respectively. Couto et al. [7] defined their technique DiShIn, which is an improvement in GraSM. DiShIn determines the disjunctive common ancestors from the number of distinct paths from the concepts to their common ancestors.

2.1.7 Wang

Wang et al. compute semantic similarity without using information content [14]. They consider all ancestors of both the concepts to be compared, and then take

common ancestors. The S-Value for GO term t related to term x (where t is an ancestor of x), is defined as,

$$\begin{aligned} S_x(t) &= 1 && \text{if } t = x \\ & \text{Max}\{w_e * S_x(t') | t' \text{ children of } t\} && \text{if } t \neq x \end{aligned} \quad (2.9)$$

Where w_e (edge weight) is a fraction (typically 0.8 or 0.6). Semantic value of a GO term x is defined as,

$$SV(x) = \sum_{t \in Tx} S_x(t) \quad (2.10)$$

Where Tx is the set of all ancestor terms of x , including x itself. Wang similarity between terms a and b is defined as,

$$Wang(a, b) = \frac{\sum_{t \in Ta \cap Tb} S_a(t) + S_b(t)}{SV(a) + SV(b)} \quad (2.11)$$

Where $S_a(t)$ is the S-value of term t related to term a and $S_b(t)$ is the S-value of term t related to term b .

2.1.8 AIC

Song et al. [15] proposed their measure based on aggregate information content (AIC). They define the knowledge of a term t as the reciprocal of Information Content(IC),

$$K(t) = \frac{1}{IC(t)} \quad (2.12)$$

The semantic weight SW of a term t is defined as,

$$SW(t) = \frac{1}{1 + e^{-k(t)}} \quad (2.13)$$

Then the semantic value SV of a term is computed by adding the semantic weights of all the ancestor terms.

$$SV(x) = \sum_{t \in T_x} SW(t) \quad (2.14)$$

AIC similarity between term a and b is defined as

$$AIC(a, b) = \frac{\sum_{t \in T_a \cap T_b} 2 \cdot SW(t)}{SV(a) + SV(b)} \quad (2.15)$$

The weight of terms increases as we move upwards. The weight of root is highest due to its lowest IC. This is because of Equation 2.12.

2.1.9 SSA

Alvarez and Yan [8] proposed a semantic similarity algorithm (SSA) that does not depend upon external databases (annotations). SSA computes semantic similarity purely from Gene Ontology. This implies that the similarity value between any two GO terms will not change unless the Gene Ontology is revised.

SSA computes the semantic similarity between two GO terms t_1 and t_2 as:

$$SSA(t_1, t_2) = \frac{spsim(t_1, t_2) + nca(t_1, t_2) + ld(t_1, t_2)}{3} \quad (2.16)$$

Where $spsim$ is the similarity computed from the distance of the shortest path between t_1 and t_2 in GO, nca is a score proportional to the depth of the nearest common ancestor of t_1 and t_2 , and ld is a similarity score between the definitions of the two terms. The resultant value lies between 0 and 1.

2.1.10 InteGO

Peng et al. [16] proposed a rank-based gene semantic similarity measure InteGO (Integration of GO similarities) that integrates three existing measures. InteGO maintains a background set of genes BG for each desired species that is used to unify the similarity scores from different seed measures. BG is sufficiently large, and unbiased, and it includes full range of similarity scores. Pre calculated similarity scores for all seed measures are stored in a database. When the system takes a gene set G as input the similarity scores of all the pairs in G and all the pairs between G and BG are calculated using all seed measures and merged into the database. If G is a subset of BG , the system outputs the results directly. All the pairs in the database are sorted incrementally based on their similarity scores and are ranked. The ranked similarity score $RankSim(g_1, g_2, m)$ for genes g_1 and g_2 is computed as

$$RankSim(g_1, g_2, m) = \frac{2 \cdot r^m}{|BG \cup G|^2} \quad (2.17)$$

Where r^m is the rank of gene pair g_1, g_2 using seed measure m . InteGO provides an adaptive mechanism to automatically select the appropriate integration method from a set of available methods. Each method in the pool of available integration methods is given a score so that the best method can be selected. InteGO provides four integration approaches: minimum, maximum, mean and median. Users are allowed to add their own integration method. To determine the best method, all pair similarities in BG are computed based on each candidate integration method and are systematically evaluated using biological data.

InteGO was tested on three model organisms with different levels of GO annotation scale and complexity. EC numbers and protein sequences were adopted as independent biological evidence. For demonstration, its results were compared with three existing similarity measures: Yu, Schlicker and Wang. The authors reported that taking the maximal ranks from all of the seed measures performs the best.

2.1.11 Mazandu and Mulder

Mazandu and Mulder [17] proposed a unified mathematical framework for all IC-based semantic similarity measures and compared these methods based on theoretical parameters. They performed experimental evaluation of different correction factors and assessed the impact of different normalization approaches. They compared various topology-based approaches and parent versus child based approaches. They used protein pairs from CESSM. To compute protein similarity from term similarity, Best Match Average approach was used.

The authors found that GraSM approach improved the performance of the existing methods. However, they observed that identifying disjunctive common ancestors is computationally expensive, especially for a complex DAG. They find that considering only the parents or only children has negative impact on the computation of IC and semantic similarity. They also observed that since the annotation based approaches make use of annotation data, the similarity scores will be biased. Some GO terms have more protein annotations than the others. This may be due to research focus on these terms. But the annotation based approaches will treat them like less specific terms, like those lying higher in the taxonomy.

2.1.12 GOSemSim

Yu et al. [18] developed a software package GOSemSim for computation of existing semantic similarity measures. GOSemSim is developed as a package for the statistical computing environment R and is released under GNU General Public License with Bioconductor project. The IC is computed from Bioconductor annotation packages and is species specific. Five measures: Resnik, Lin, Jiang Conrath, Schlicker and Wang were integrated in GOSemSim. User can select the required method in parameter. GOSemSim provides six functions. The function `goSim`, `mgoSim`, `geneSim` and `clusterSim` compute the semantic similarity among GO terms, sets of GO terms, GO descriptions of gene products, and GO descriptions of gene clusters respectively. Functions `mgeneSim` and `mclusterSim` compute

the similarity matrix of a set of genes and gene clusters. The result of goSim is between 0 and 1.

By mapping gene products to GO terms, the functions geneSim, mgeneSim, clusterSim, mclusterSim are used to compute semantic similarity between gene products. Species and gene id's are required for this computation. The functions mgeneSim, clusterSim and mclusterSim are designed for computations at large scale.

2.1.13 Seco et al.

Seco et al. [19] proposed that IC computation based only on the hierarchical structure of concepts is better than that employing corpus analysis. They performed experiments on WordNet Ontology. Their method is based on the assumption that the taxonomy of WordNet ontology is built in a structured and meaningful manner. The concepts with many hyponyms convey less information as compared to leaf concepts. Since leaf concepts are the most specific in the taxonomy, they convey maximum information. Therefore, the IC value of a WordNet concept can be expressed as a function of the hyponyms it has.

$$IC(c) = 1 - \frac{\log(hypo(c) + 1)}{\log(max_{wn})} \quad (2.18)$$

Where function $hypo(c)$ returns the number of hyponyms of c and max_{wn} is the maximum number of concepts in the ontology. The information content decreases monotonically as we traverse from a leaf node to root. Information content of root will be 0.

They evaluated their IC measure by correlating the similarity computed by their IC with human judgment. They found that their measure outperformed the previous ones.

2.1.14 Yang et al.

Yang et al. [20] proposed their similarity measure that takes into account the descendant concepts in addition to ancestors concepts. The authors point out that the descendant concepts should not be disregarded when measuring similarity between non-leaf concepts. Their approach is based on downward random walks. It can be used to improve the existing similarity measures. Suppose a term p has four child terms c_1 , c_2 , c_3 and c_4 , and a gene g is annotated with p . The method adopts a reasoning which says that g is indirectly annotated with c_1 with some probability, say 0.2. Similarly, g is assumed annotated with c_2 , c_3 and c_4 with a probability, say 0.2 each. The combined probability is 0.8, not 1. The remaining 0.2 probability is for uncertainty, which is for unknown child of p , that may be missing in the ontology due to our incomplete knowledge about the domain.

Now when one compares a gene annotated with p and another gene annotated with a term x , there is 20% chance that a gene annotated with c_1 is being compared. This allows us to measure semantic similarity between c_1 and x from the semantic similarity between p and x , using a factor of 0.2. The idea is to decompose the similarity of a parent with some concept, into the similarities of the children with that concept. The method computes the random walk contribution to the similarity. This contribution is finally combined with the similarity obtained by a host measure, that may be any existing similarity measure. The host measure only considers the taxonomy above the two concepts being compared, and the random walk contribution only considers the taxonomy below them. The combined similarity will have both the components, and thus, the host similarity measure will be improved.

2.1.15 Bien et al.

Bien et al. [21] propose a bi-directional semantic similarity measure that includes ancestors as well as descendant concepts. Since a child concept is a special case of the parent concept, they assume that the semantics of a parent concept are union

of the semantics of all its children. GO frequently contains multiple parents of a child concept. Therefore, it is likely that two concepts being compared share a common child concept. The authors argue that, in such case, the children are also relevant to similarity measurement. In cases where concept pairs have same ancestral topology, descendant topology may be quite different.

They extend Wang similarity measure to include the descendant concepts in addition to ancestors. They define the similarity as:

$$S_{BSV}(t_A, t_B) = \frac{\alpha \cdot S_{wang}(t_A, t_B) + \beta \cdot S_{DSV}(t_A, t_B)}{\alpha + \beta} \quad (2.19)$$

Where α is the number of total ancestors and β the number of total descendants of t_A and t_B . Swang is the Wang's similarity between the two concepts, which is derived from ancestors, and SDSV is the descending semantic value, derived from common descendants. Due to recursive dependence on common descendants, SDSV is expected to impact comparisons between concepts that are high in the hierarchy. However, terms may not have common descendants just because more specific child concepts are not yet created. SBSV gives higher weight to SDSV when terms are higher in the taxonomy, and less when they are lower in the taxonomy. Total number of descendants β , complements the weakness of SDSV by reducing the weight of SDSV when the terms have few descendants, with a reduced chance of having common descendants, regardless of their similarity.

Swang is the Wang's similarity between the two concepts, which is derived from ancestors, and SDSV is the descending semantic value, derived from common descendants. Due to recursive dependence on common descendants, SDSV is expected to impact comparisons between concepts that are high in the hierarchy. However, terms may not have common descendants just because more specific child concepts are not yet created. SBSV gives higher weight to SDSV when terms are higher in the taxonomy, and less when they are lower in the taxonomy.

To evaluate this measure, the authors explored the GO taxonomy to look for the terms having highest discrepancies between the ascending and descending similarities. They based their evaluation on comparison with sequence similarity. They computed correlation coefficient for gene expression data, and showed that their measure is better than the existing ones.

2.1.16 HRSS

Wu et al. [22] proposed an edge based similarity measure known as Relative Specificity Similarity (RSS). The measure used both the taxonomy above the given terms and that below the terms to compute similarity.

They defined distance between two concepts as the number of edges in the shortest path from one concept to the other. The similarity has three components. The first one is a measure of how specific the most recent common ancestor (MRCA) of the concepts is. The second component is a measure of how general the two concepts are, in terms of the minimum distance from the concepts to the leaves. The third component is a measure of the local distance between the concepts and MRCA.

After RSS, the authors proposed further improved measure called Hybrid Relative Specificity Similarity (HRSS). For this measure, the conventional IC was used, which is computed from annotations. They defined IC based specificity of MRCA as the IC of MICA (Most Informative Common Ancestor). They defined IC based generality of a concept as the distance between the concept and the most informative leaf nodes (MIL). They combine IC-based computations with the edge-based computations of RSS to formulate the new measure HRSS.

They evaluated the new measure by the correlation with sequence similarity, using CESSM data set. They also used protein-protein interactions for the evaluation. They find that HRSS and simGIC measures outperform all other measures.

2.1.17 GAP

Vafaee et al. [23] proposed a method called GAP (Gene functional Association Predictor) for predicting gene functional associations. The measure enhances information gain from the gene annotations. The authors designed two types of term similarity measures: data driven similarity and ontology based similarity. In case the terms are identical, data driven similarity is defined as the inverse of the frequency of the term. When the terms are different, the similarity is zero. The reasoning behind this measure is that, two different features do not contribute to the similarity between two objects, and if the objects share a common feature, it contributes to their similarity depending upon how informative the feature is. The informativeness of a term is determined from its inverse frequency in the data set used. The data driven similarity is used when the features are not present in an ontology.

When the features correspond to the concepts in an ontology, like GO, the other type of similarity i.e. ontology based similarity is used. For this purpose, the idea of the most informative common ancestor is used, like many previous studies. GAP uses multiple semantic similarity measures. Authors propose two new concepts: Leaves and Specificity-descendant.

The proposed Leaves measure defines information content of a concept c in terms of the number of leaf concepts that descend from c . The specificity-descendant information content is defined as

$$IC(t) = f(f_1(depth(t)), f_2(local_{density}(t))) \quad (2.20)$$

(Eq 2.20) Where $f(f_1, f_2) = f_1 * f_2$. f_1, f_2 quantify the contribution of the depth and local density and take into account all the descendant concepts of t .

$$f_1(depth(t)) = \frac{depth(t)}{depth(v) \max_{v \in \text{descendants}(t)}} \quad (2.21)$$

$$f_2(local_{density}(t)) = \frac{max_{terms}}{descendants(t)} \quad (2.22)$$

Where max_{terms} is the total number of terms in the whole ontology. The similarity between two genes is defined as the weighted sum of the gene feature scores. To aggregate feature similarity scores into resultant gene similarity, two transformation strategies were used. In Decile-only aggregation, each feature similarity is mapped to an integer $h \in [1, 10]$ where the score is at the h_{th} decile of the sample. In Decile-weighted aggregation, a weighted sum is used where the weights are decile values as described in Decile-only aggregation, and the scores are scaled from 0 to 1.

To evaluate GAP's performance, a set of human genes was randomly selected that had some known interactions. GAP was then queried by those genes, and scores were computed for the predicted interactions.

2.1.18 Santos et al.

Santos et al. [24] used Gene Ontology annotations to develop semantic-based measures for the selection of druggable target proteins. They constructed several binary vectors that represented protein drug targets retrieved from public databases. Each vector was a set of 2700 binary descriptors, where each descriptor represented an InterPro annotation. Some targets were used for training and validation, and the rest were used to generate a representative vector space using SVD. The similarity between a pair of drug targets was computed as the cosine of the angle between the respective representing vectors on a reduced space. The similarity relationships were analyzed with the help of clustering techniques available in a software called Multi Experiment Viewer.

A similarity matrix was constructed from the values of the computed cosines, and this matrix was used as input to the software. Hierarchical clustering algorithm (HCL) was applied to the input, which produced a heatmap with the targets reordered semantically.

The results were compared with those produced by using sequence similarity, and were found to be better. To find druggable candidates, other proteins were projected into the reduced space.

2.1.19 Measuring Semantic Similarity between Proteins

The semantic similarity measures generally find similarity between GO term pairs by using the methods discussed in the previous section. To translate term similarity into protein similarity, annotation data is used. Suppose we want to compute similarity between proteins p_1 and p_2 , assume p_1 has a set G_1 of GO terms annotated to it, and p_2 has a set G_2 , similarity between p_1 and p_2 will come from the similarities between the GO terms of G_1 and G_2 by using a particular method. Each term in the set G_1 is compared with the each term in the set G_2 , and then the aggregated similarity is computed by using one of the three ways.

In the first approach, the maximum of all term similarity scores (max approach) would be the similarity between p_1 and p_2 . The second approach takes average of all term similarity scores (average approach) to determine the overall similarity between p_1 and p_2 . The third approach named as Best Match Average (BMA), computes the maximum (best) similarity of each term in G_1 with the each term in G_2 (named the best similarity set as G_1-G_2), and computes the average of G_1-G_2 , $AvgG1-G2$. Then computes the best similarity of each term in G_2 with the each term in G_1 (named the best similarity set as G_2-G_1), and computes the average of G_2-G_1 , $AvgG2-G1$. Then the BMA is computed by taking the average of both the measures $(AvgG1 - G2 + AvgG2 - G1/2)$.

2.2 Correlation between Semantic Similarity and Sequence Similarity

If two proteins are similar in sequence, one would expect that they would be performing similar functions. This means that certain degree of correlation should

exist between sequence similarity and semantic (function) similarity. However, both processes and data used for the semantic similarity measurement may not be accurate, resulting in some uncertainty.

There are multiple reasons for such correlation not being strong. One of them is incompleteness of information. Annotation data is generally incomplete for most proteins since it is very difficult to define all possible functions a protein performs. Annotations are not uniform over proteins: some proteins are more heavily annotated than others. Even the existing annotations may not be perfectly accurate.

Many studies have applied the assumed good correlation between sequence similarity and semantic similarity to evaluate novel semantic similarity measures. In these studies, it is assumed that a better semantic similarity measure would have better correlation with sequence similarity.

2.2.1 Lord et al.

Lord et al. [25] were the first to apply semantic similarity measures on GO and proteins, and to evaluate the measures based on their correlation with sequence similarity. They assumed that highly similar sequences should be highly semantically similar. They used average method to translate term similarity into protein similarity. They reported a good correlation between sequence and semantic similarity. The correlation was especially high in the molecular function GO aspect.

They also found protein pairs going against the correlation. They found that such protein pairs lie in three categories. Polymorphic groups mean diverse classes of protein are involved in the same process. Hyper variable protein families have similar reasons as above. Mis-annotations mean some error in annotation, and was responsible for half of the protein pairs.

2.2.2 Louie et al.

Louie et al. [26] use the assumption of a good correlation between sequence similarity and semantic similarity for prediction of protein function from sequence. This is a well known problem in bioinformatics. The authors address particularly the problem of overly specific function prediction using a statistical model based on the relationship between sequence similarity and semantic similarity. The model is based on sets of proteins with experimentally tested functions and measures of function specificity and similarity.

Their experiments show almost exact function similarity when proteins have high sequence similarity. They find small probability of function match when proteins have low sequence similarity. They report that function similarity increases with increasing sequence similarity, but shows considerable variation. The authors applied their model to a collection of proteins of unknown function and predicted their function.

They created test and training data sets using single function proteins with IDA evidence code (experimentally characterized). Proteins in each set were sequence aligned with each other. The model predicts the function similarity between two proteins when their sequence similarity is known.

2.2.3 Lee et al.

Lee et al. [27] review protein function prediction methods. There is a large number of proteins having unknown functions. The experimental methods to determine their functions are costly and time consuming. However, computational methods to predict protein functions are effective. The most common approach for function prediction is based on the knowledge that proteins with similar sequences generally have similar functions. Since protein structures are mostly not known, sequence based prediction appears to be most practical.

To enhance the value of these methods, acquiring more experimental data and validation procedures are important. It is therefore beneficial to compare and analyze the results of various prediction methods. Meta-servers make it possible by providing access to such methods. Improving these meta-servers, and providing improved software environment would result in more powerful interfaces, which would be helpful for researchers. Validation by independent sources will increase the confidence of biologists on the predictions.

2.2.4 Joshi and Xu

Joshi and Xu [28] perform a study on the relationship between sequence similarity and function similarity for proteins of four organisms. They work on probability based on number of gene pairs sharing the same function at a certain GO index level against the total pairs having any functions at that level. They report a consistent correlation between the two types of similarity. They observe cases where function similarity reaches very high even when sequence similarity is low. They report that if two proteins have sequence similarity more than 70%, they have about 90% probability to have the same biological process for index levels 1 to 8.

They study the behavior of annotations based on computational techniques as compared to those based on experimental evidence. They observe a difference between the quality of the two types of annotations, and infer that many annotations based on computational techniques may be incorrect. Some of the incorrect annotations may be caused by over extension of function details while inferring a protein from a protein hit with a known function.

2.2.5 Anscombe

Anscombe [29] comments on the use of graphs as compared to correlation coefficient. Numeric calculations are exact in contrast to graphs, and may be helpful in making concrete decisions. However, in many situations, numeric calculations

may lead to incorrect conclusions. He gives convincing arguments, along with an example as evidence.

The example includes four fictitious data sets, each having some (x, y) pairs. Scatter plots of all four data sets are created. The data sets are constructed such that a typical regression program gives the same output for all. In particular, the Pearson's correlation coefficient for each set is the same i.e., 0.8. However, when we look at the scatter plots, they are quite different from each other. Some of them have really good correlation between x and y while others are weakly correlated.

The example shows that, numerical values such as correlation coefficient, may be misleading in many situations. It would be more appropriate that graphs are also studied in addition to the correlation coefficient, so that various trends and patterns can be observed. This is very relevant to our study of the relationship between sequence similarity and function similarity.

2.2.6 Pesquita et al.

Pesquita et al. [30] conducted an evaluation of semantic similarity measures based on the correlation between sequence similarity and semantic (function) similarity. They reported that the relationship between protein sequence similarity and function similarity is not linear. However, it can be approximated by a normal cumulative distribution function. They reported that different semantic similarity measures do not demonstrate significantly different behavior with respect to correlation with sequence similarity. Therefore, other metrics such as resolution may be used to evaluate the performance of semantic similarity measures. They also found that average and maximum combination approaches are not suitable for computation of protein similarity from term similarity.

2.2.7 Other

Yang et al. [20] use the correlation between sequence similarity and function similarity as one criterion to evaluate the performance of their technique, assuming that a better performing measure should have better correlation coefficient with sequence similarity.

2.3 Evaluation of Semantic Similarity Measures

Due to the presence of so many approaches and measurement methods for semantic similarity, an important question is: how accurate each measure captures the functional similarity between two GO terms, proteins or genes?

The selection of the approach to validate similarity measures is still a subject of debate since no accepted standard and few comparative studies exist. There are related measures, like sequence similarity and coexpression data, which may be considered for the validation of semantic similarity methods. By correlating semantic similarity with such measures it may be possible to determine how well a measure captures the similarity in gene products.

The first assessment exercise was done by Lord et al. [25], who evaluated Resnik, Lin, and Jiang Conrath measures by correlating them against sequence similarity, and by using the average approach. They used Pearson's correlation coefficient as the evaluation criterion, and reported that it is highest for molecular function aspect of gene ontology. Resnik's measure was reported to have the highest correlation with sequence similarity, and accordingly, considered to be the best measure.

Sevilla et al. [31] used gene coexpression data, and evaluated the same three measures using the maximum approach. They found that Resnik's measure performs best, and that the biological process aspect has the highest correlation with gene coexpression.

Couto et al. [13] evaluated the GraSM variation of these measures based on correlation with sequence similarity. They used best-match average (BMA) combination approach to compute protein similarity. They found the highest correlation with Jiang and Conrath's measure for biological process sub-ontology of GO.

Schlicker et al. [32] compared their measure with Resnik's measure by viewing the distribution of sequence similarity over semantic similarity, considering discrete levels. They reported that their measure performed better, especially in distinguishing orthologous gene products from the gene products with other levels of sequence similarity.

Lei et al. [33] evaluated several semantic similarity measures in an application for predicting subnuclear location, using the multiclass classification results (Matthews correlation coefficient) as the criterion. They compared Resnik's measure with the simLP measure and found them not to be significantly different in performance. They also tested several combination strategies for applying simLP to gene product similarity and concluded that the sum of exact-matching term pair similarities produced the best results.

Guo et al. [34] evaluated simUI, simLP, Resnik, Lin, and Jiang Conrath measures on their ability to characterize human regulatory pathways. They reported that Resnik's measure performs the best.

Wang et al. [14] evaluated their measure against Resnik's measure. They clustered gene pairs by semantic similarity computed by their method, and compared the resulting clusters with human perception. They reported that their measure is closer to human expectations as compared to Resnik's measure.

Pesquita et al. [30] conducted an evaluation of semantic similarity measures by modeling their behavior as function of sequence similarity. They compared Resnik, Lin, and Jiang Conrath measures using several combination approaches, and also the group wise measures simUI and simGIC. They reported that the correlation between semantic similarity and sequence similarity is not linear. They reported that the best-match average (BMA) combination approach gives better results

than the average and maximum approaches. They also found that the majority of semantic similarity measures show identical behavior, and suggested that resolution is better criterion for evaluation. They identified hybrid simGIC to be the best method on the basis of resolution.

Mistry et al. [35] evaluated eleven measures: Resnik, Lin, and Jiang Conrath with both the average and maximum approaches, three vector-based measures (cosine, weighted cosine, and kappa), and TO and NTO. They studied correlation between the measures and the correlation with sequence similarity. The TO measure was found to have best correlation with sequence similarity.

Song et al. [15] evaluated their proposed measure AIC by comparing it with Resnik, Jiang Conrath, Lin and Wang. They used correlation with gene expression data as the evaluation criterion. They found AIC having the best correlation, and therefore having the best performance. Table 2.1 gives a summary of evaluation studies with their results.

What can be understood from these studies is that there is no agreement upon the most accurate measure for comparing GO terms or gene products. Different measures performed differently under different situations. A measure can be well suited for one task but perform poorly for another. For instance, simUI was found by Guo et al. to be the weakest measure for its ability to characterize human regulatory pathways, while Pesquita et al. found it to be fairly good when evaluated for correlation with sequence similarity. However, one result was consistently found that pair wise measures using Resnik similarity outperform Lin and Jiang Conrath measures.

There are arguments for and against various strategies used, and there is no gold standard for evaluation of semantic similarity measures. Sequence similarity is well known to be correlated to semantic similarity, but it is also well known that there are gene products with similar function but different sequence.

TABLE 2.1: Approaches used for evaluation of semantic similarity measures.

Study Name	Evaluation Approach	Methods Compared	Result
Lord et al.	Correlation with sequence similarity	Resnik, Jiang Conrath, Lin	Resnik identified as best
Sevilla et al.	Correlation with gene co-expression data	Resnik, Jiang Conrath, Lin	Resnik identified as best
Couto et al.	Correlation with sequence similarity	Resnik, Jiang Conrath, Lin and their GraSM variants	Jiang Conrath GraSM the best
Schlicker et al.	Correlation with sequence similarity	Schlicker, Resnik	Schlicker found better
Lei et al.	Matthews correlation coefficient	Resnik, SimLP	No significant difference
Guo et al.	Ability to characterize human regulatory pathways	SimUI, SimLP, Resnik, Lin, and Jiang Conrath	Resnik's measure performs best
Wang et al.	Comparison with human perception	Resnik, Wang	Wang's measure performs better
Pesquita et al.	Resolution of similarity measures	Resnik, Lin, Jiang Conrath, SimUI, SimGIC	SimGIC performs best
Mistry et al.	Correlation with sequence similarity	Resnik, Lin, Jiang Conrath, cosine, weighted cosine, kappa, TO, NTO	TO measure performs best
Song et al.	Correlation with gene expression data	Resnik, Jiang Conrath, Lin, Wang	AIC performs best

TABLE 2.2: Similarity computed by existing methods.

Term1	Term2	Resnik GraSM	Jiang GraSM	Lin GraSM	Wang
GO_0048308	GO_0048311	0.3	0.02	0.3	0.56
GO_0007005	GO_0051646	0.02	0.01	0.03	0.2
GO_0016043	GO_0071841	0.04	0.11	0.2	0.51
GO_0051179	GO_0051641	0.19	0.22	0.64	0.76
GO_0051179	GO_0051179	0.19	1	1	1
GO_0051641	GO_0051641	0.07	1	1	1

2.4 Critical Analysis of Existing Methods

2.4.1 Motivating Case Study

In order to analyze the existing methods, we take example taxonomy from Gene Ontology as shown in Figure 1.1. We select four pairs of concepts making different formations. Then we compute the similarity between them using the existing methods (normalized in case of Resnik) and the computed similarities are shown in Table 2.1.

We can observe that the similarity computed by these methods for same pairs of GO terms is not consistent. Some methods give higher similarity than others, for the same pairs.

We discuss limitations of various measures, and the need to develop new measure. We divide the measures into two parts: old measures and recent measures.

2.4.2 Old Measures

Resnik's measure has distinct characteristics among all similarity measures. To elaborate, let us take an example of a pair of GO terms GO:0051641-cellular localization and GO:0051179-localization located near root (Figure 1.1). Resnik similarity between these two terms is 0.19. The similarity of GO:0051179-localization with itself is also 0.19, not 1. Lin gives 0.64 for the first comparison and 1 for the second one. The reason for significantly low similarity value by Resnik is that the terms are located near root, where Resnik drastically differs from other measures.

Let us now consider a taxonomy of store items as shown in Figure 2.1. Resnik similarity between 'food item' and 'food item' is less than 100%. For other measures, it is 100% because a concept is compared with itself. The logic behind Resnik can be explained in real life: one food item can be dissimilar with another. This factor should not be ignored when formulating similarity measures. Let us call it Resnik Factor. Now we cannot say for sure that 'food item' is dissimilar with 'food item'. It can be similar in some cases. In fact, some uncertainty is involved in this comparison, and many other comparisons. It would be more appropriate to say that concept 'food item' is similar to itself, subject to some uncertainty. Further, apple is similar to apple with some (little) uncertainty.

From the point of view of semantic computing, the structure of ontology is more relevant to measure similarity as compared to the instances or any external data. However, most measures make use of annotation of gene products for this purpose. Whether this external data should be used for computation of term similarity, is another important issue that needs to be addressed. Suppose there are three sibling terms a , b and c . The Resnik (and other IC-based measures) similarity between a and b will change if more annotations of c are added. It is possible that two distantly located concepts in ontology have higher Resnik similarity value as compared to two sibling concepts because of too many annotations of the sibling concepts.

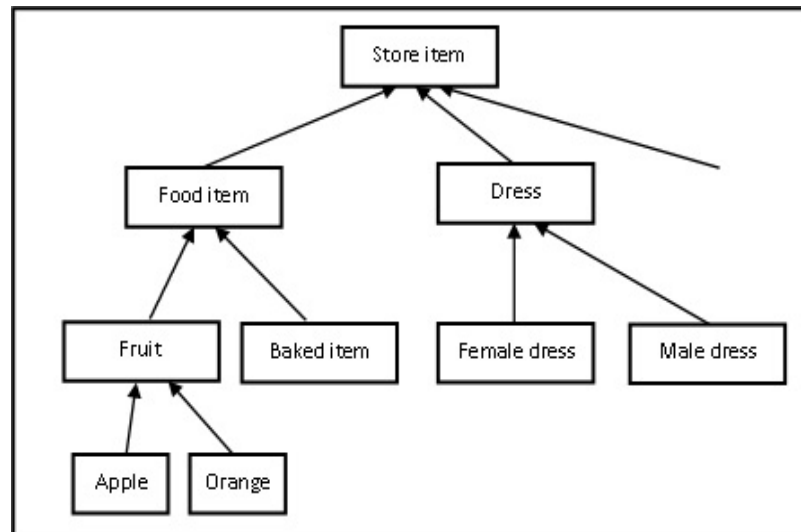


FIGURE 2.1: An example taxonomy of store items.

Lin's similarity between two close terms located near root will be higher as compared to Resnik similarity between the same two terms. Lin's similarity of a term with itself will be 1, whether the term is general or specific. Jiang Conrath similarity measure too shows the same behavior. According to these two measures, the similarity between 'apple' and 'apple', between 'fruit' and 'fruit', and between 'food item' and 'food item' is 100%. From one point of view, Lin and Jiang Conrath appear to make correction in the Resnik's measure. From other point of view, they appear to eliminate a good characteristic of that measure. Taking another example, similarity between 'fruit' and 'baked item' using Lin's measure, may be nearly same as that between orange and apple. This does not appear to be consistent with the human judgment.

2.4.3 Recent Measures

The above issues were addressed by later similarity measures. There is a class of measures that compute similarity from some values taken from all the ancestors of the two terms. The contribution from common ancestors is divided by that from all ancestors. SimUI uses number of ancestors, while simGIC uses IC of ancestors as contribution, reducing the contribution from general terms. Wang reduces the weight of ancestors by multiplying the weight of child concept by a fraction, for

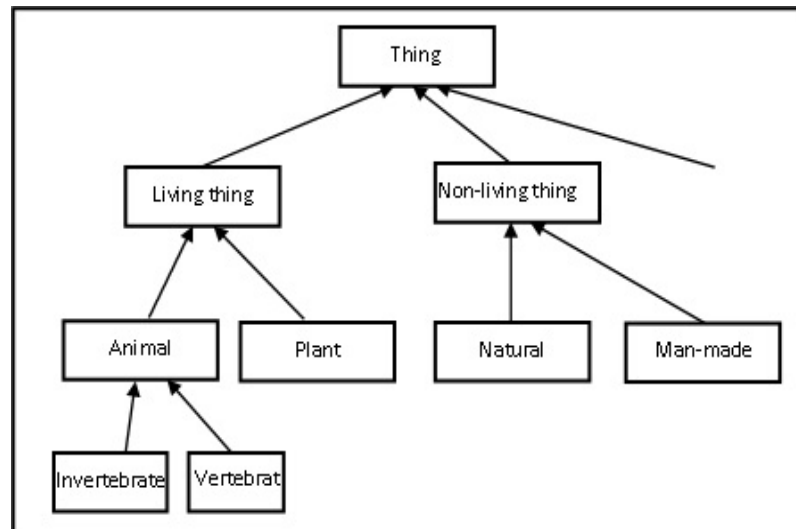


FIGURE 2.2: An example taxonomy of things.

example, 0.8. AIC, the most recent measure, assigns higher weights to ancestors, with maximum weight for the root.

The above measures sound logical, with a clear advantage of handling complex DAG structure of ontologies like Gene Ontology. The close concepts near leaves will have more common ancestors than the uncommon ones, giving a high similarity value due to a high numerator. Similar close concepts near root will have less common ancestors and less similarity value. This addresses the issue with Lin measure. There are few but quite significant issues which are not addressed by the above measures, including the most recent ones. In the example taxonomy of Figure 2.2, let us analyze how the similarity is measured by various methods. Similarity between ‘Living thing’ and ‘Non-living thing’ will be 33% by SimUI due to 1 common ancestor out of 3. Wang and AIC will count the common ancestor ‘Thing’ twice, raising the similarity to around 50%. Wang will reduce the contribution of ‘Thing’ by a factor, e.g. 0.8, dropping the value to 44%. AIC will give higher weight to ‘Thing’, raising the similarity above 50%. SimGIC will give zero weight to ‘Thing’, bringing the similarity to 0. The semantics of the ontology, and our knowledge about things suggest that living thing and non-living thing are highly dissimilar.

Now we compare ‘Animal’ with ‘Man-made’ thing, having 1 common ancestor

‘Thing’ (counted twice by Wang and AIC) out of 5. SimUI will give 20%, SimGIC 0%, while Wang will give somewhat less than 33%. AIC, with the highest weight of ‘Thing’, will give more than 33%. We cannot prove which similarity value is correct. The most effective tool we have is the logical reasoning, using the principles of semantic computing. Root is the least specific concept in an ontology, with information content of 0. Animal and man-made thing may not be given high similarity for both being ‘things’.

SimGIC gives low similarity (in agreement with semantics) so far. However, when we move down, it will change. It will give slightly less than 33% similarity between ‘Plant’ and ‘Animal’ due to the common ancestor ‘Living thing’. The similarity will grow to above 50% one or two steps further down, for quite general siblings in a large ontology.

The result of Resnik needs more analysis as it does not give 100% similarity to same concepts, and has been often criticized for this. However, it has one feature that may guide us to better design of a new similarity measure, called as ‘Resnik factor’ earlier in this dissertation. Due to this factor, Resnik gives low (although too low) similarity between Plant and Plant, and relatively high similarity between Apple and Apple. This is not only logical, but may also be useful for some semantic processing applications. For example, in protein comparison, if two proteins share a general GO term, it should contribute less to their similarity, while sharing a specialized term should contribute more.

We have analyzed why the similarity computed is not consistent across different methods, i.e., there is a high degree of disagreement among them. This answers research question 1 as given in section 1.1.

2.5 Protein Sequence Similarity

Protein sequences frequently have some similarity among them depending upon their relationship. Such similarity is due to common evolutionary origin. Therefore, measuring this similarity is important for the research on evolution. Comparing two sequences is not a simple string comparison. For example, sequences ACCDEDD and EACCPTDEDE have quite significant similarity. To find out similarity between them, the sequences should be aligned together such that ACC of the first sequence matches with ACC of the second, and DED of first, with DED of the second. To do this, gaps will be inserted in the first sequence before ACC and before DED. Gaps will have penalty associated with them, reducing the similarity value.

Not all matching amino acids contribute the same to the similarity of sequences. C matching with C contributes more to the sequence similarity than A matching with A. When there are mismatches, they generally result in the reduction of similarity (negative contribution). However, this negative contribution also depends upon the mismatching amino acids, and in some cases, it is actually positive contribution (for example, comparison of D with E). This is because the amino acids are related evolutionary. The overall similarity score is computed from the matching and non-matching amino acids in the sequence alignment. The contribution of each match and mismatch is determined from a scoring matrix. BLOSUM62, shown in Figure 2.3 is a commonly used scoring matrix. There are other scoring matrices including BLOSUM50, BLOSUM80, PAM10, PAM50 and others.

Sequence similarity can be expressed as percent identity. A high sequence similarity between two proteins is unlikely to exist just by chance. However, little similarity may exist by chance. We say that two proteins have significant sequence similarity if their similarity is high to the extent that is unlikely to happen by chance. This is crucial for research because the significant similarity will mean the two proteins diverged from a common ancestor. In this context, alternative methods have been developed to express similarity score.

Ala	4																			
Arg	-1	5																		
Asn	-2	0	6																	
Asp	-2	-2	1	6																
Cys	0	-3	-3	-3	9															
Gln	-1	1	0	0	-3	5														
Glu	-1	0	0	2	-4	2	5													
Gly	0	-2	0	-1	-3	-2	-2	6												
His	-2	0	1	-1	-3	0	0	-2	8											
Ile	-1	-3	-3	-3	-1	-3	-3	-4	-3	4										
Leu	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4									
Lys	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5								
Met	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5							
Phe	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6						
Pro	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7					
Ser	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4				
Thr	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5			
Trp	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11		
Tyr	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	
Val	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4
	Ala	Arg	Asn	Asp	Cys	Gln	Glu	Gly	His	Ile	Leu	Lys	Met	Phe	Pro	Ser	Thr	Trp	Tyr	Val

FIGURE 2.3: Blosum62 matrix.

Expect value (E) is a value that represents the number of proteins having a given similarity (or higher) that one can expect by chance. It decreases exponentially as the similarity score increases. E-value can be 10^{-6} or less for proteins having significant similarity. P-value is the probability of finding a protein having given similarity (or higher) by chance.

2.6 Conclusion

The similarity values computed are not consistent across various methods, and considerable disagreement exists. Resnik's measure has a unique feature that properly treats general concepts, lowering their similarity. It makes logical use of IC. Later measures removed some limitations of Resnik's measure, but all of them, including the most recent ones, miss this important feature. There is a need to address the issues in the existing semantic similarity measures.

The relationship between protein sequence similarity and function similarity has been explored. There are indications that a strong correlation exists between the two. However, evidence exists that proteins with low sequence similarity have high

function similarity, and that the relationship is not linear. It is crucial that this relationship be studied in more detail, since it is applied in the areas like prediction of protein function, and evaluation of new similarity measures. There is a need to check whether this correlation can be used for evaluation of semantic similarity measures.

There is no agreed upon criterion to evaluate the correctness of semantic similarity measures. Some authors evaluated semantic similarity measures using the correlation with sequence similarity as the criterion. However, there is no strong reason for this correlation to be used for evaluation.

Chapter 3

Proposed Methods

We propose two methods for computing semantic similarity: SemSim and SimExact. SemSim is a general purpose method to measure semantic similarity between concepts of ontology. It can be used for any ontology including Gene Ontology. SimExact has a specific purpose: Finding similar proteins with high efficiency (fast speed). This is crucial for many applications. However, it is not appropriate for measuring semantic similarity between the terms of Gene Ontology.

3.1 Semantic Similarity Measure: SemSim

In order to correctly compare concepts of ontology, especially Gene Ontology, it is necessary to resolve the issues discussed earlier. First of all, we would treat the structure of ontology and annotation data separately. Annotations are not integral part of ontology. In OWL, individuals or instances are not mixed with ontology structure. Although protein or gene annotations are not the same as instances, still they are databases external to Gene Ontology. Similarity between two concepts of ontology should depend upon the structure of ontology rather than external databases. Number of proteins annotated to a GO term determines its uniqueness.

There is a need to formulate a mechanism to compute Information Content and similarity purely from the ontology structure. We define Ontology-based Information Content IC_{ont} , which is computed in the same way as IC , except that the frequency of leaf concepts is considered to be 1, rather than the number of annotations. Therefore, IC_{ont} is independent of annotation data, and is solely based on the ontology structure. For distinction, we denote IC as IC_{ann} which reminds that it is based on annotations of proteins or genes.

Now we propose our method SemSim which addresses the above mentioned issues. The initial part of this method uses Jaccard index like simUI and simGIC. Similarity is a fraction in which numerator is computed from the common ancestors and denominator from all ancestors of the two concepts. This way, the DAG structure of ontology does not pose problems.

We define $Ancestors^+(c)$ as the union of $Ancestors(c)$ and the concept c itself.

$$Ancestors^+(c) = \{c\} \cup Ancestors(c) \quad (3.1)$$

The initial form (SemSimI) takes common ancestors of both concepts, sums up their IC_{ont} , and divides this sum by the sum of IC_{ont} of all ancestors of both the concepts.

$$SemSimI(c_1, c_2) = \frac{\sum IC_{ont}(Cmn(c_1, c_2))}{\sum IC_{ont}(Cmb(c_1, c_2))} \quad (3.2)$$

where

$$Cmn(c_1, c_2) = Ancestors^+(c_1) \cap Ancestors^+(c_2)$$

$$Cmb(c_1, c_2) = Ancestors^+(c_1) \cup Ancestors^+(c_2)$$

This is initial unadjusted form of SemSim, which has some issues to be addressed. It does not take into account the annotation of proteins or genes. Similarity between fruit and fruit (Figure 2.1) will be 1. Similarity between plant and animal (Figure 2.2) will be slightly less than 0.33 (IC of thing being zero). This is low as

compared to Lin, Wang and AIC, but still high in the context of the taxonomy. Similarity between vertebrate and invertebrate will be near 0.5 because of two common ancestors out of four, again too high for generalized concepts. Overall, the similarity between two close generalized concepts will be high. So this form of SemSim does not fully incorporate what we called Resnik Factor.

Our approach to deal with this factor is to introduce confidence level in addition to similarity value. For example, the similarity between fruit and fruit is high, with a low confidence level, and the similarity between orange and orange is also high, with a higher confidence level. Two oranges are more likely to be similar than two things which are only known to be fruits. The advantage of this approach is that we can measure similarity using graphical methods, and confidence level using some other measure like annotation-based Information Content.

We introduce two types of confidence levels, which can be combined together. The concepts near root generally represent broad classes of things like ‘living thing’ and ‘non-living thing’, ‘natural thing’ and ‘man made thing’ etc. So they are different from one another at very basic level, even if they are located close to each other. If the concept is compared with itself, similarity should be high, but there is some uncertainty. The first type of confidence level is a measure of specificity of the concepts. This is computed from IC_{ont} .

$$Spec(c_1, c_2) = \frac{IC_{ont}(c_1) + IC_{ont}(c_2)}{2 \cdot MaxIC_{ont}} \quad (3.3)$$

where c_1 and c_2 are the two concepts to be compared, and $MaxIC_{ont}$ is the maximum IC_{ont} in the ontology. Specialized concepts have higher Information Content, and higher confidence level, whereas generalized concepts have lower Information Content and lower confidence level. Part-of relationship was not used for computing information content. The reason is that two items being part of an assembly will be frequently different from each other. For instance, chair and computer are part of an office, but are not similar.

The second type of confidence level is a measure of uniqueness of a concept in the context of annotations. A concept having thousands of protein annotations is less unique as compared to a concept (GO term) having only few annotations. It will be relevant only when using SemSim for comparing proteins or genes. When a unique concept is compared with itself, it should give high similarity and high certainty, but when we compare a less unique concept with itself, it should give high similarity but less certainty. Uniqueness is computed from IC_{ann} .

$$Uniq(c_1, c_2) = \frac{IC_{ann}(c_1) + IC_{ann}(c_2)}{2 \cdot MaxIC_{ann}} \quad (3.4)$$

This system gives three values for a pair of concepts: similarity value, specificity and uniqueness. In some situations, a single similarity value is required by the client application. For this purpose, adjustment is applied on initial similarity value. Adjustment factor Ajd is defined as

$$Adj = w_1 \cdot (1 - Spec) + w_2 \cdot (1 - Uniq) \quad (3.5)$$

w_1 and w_2 are the weights (positive) given to the two types of confidence levels respectively, such that $w_1 + w_2 \leq 1$. Each weight will be set to 0 for no adjustment and 1 for maximum adjustment, with their sum not exceeding 1. We define SemSimA as

$$SemSimA = SemSimI(1 - Adj) \quad (3.6)$$

The adjustment reduces the similarity value of generalized concepts and the concepts having many annotations. This adjusted similarity measure addresses the above mentioned issues. In Figure 2.2, let us assume that the ICs of various concepts are as shown in Table 3.1.

Then, the similarity between various generalized concepts will be as shown in Table 3.2. Here, the weight w_2 is set to zero, because uniqueness is not modeled in

TABLE 3.1: Assumed ICs of concepts of Figure 2.3.

Thing	0
Living thing	1
Plant, Animal	2
Vertebrate, invertebrate	3

TABLE 3.2: Similarity between generalized concepts by SimSim A.

Concept 1	Concept2	$w_1 = 0$	$w_1 = 0.4$	$w_1 = 0.8$	$w_1 = 1$
Plant	Animal	0.2	0.13	0.07	0.04
Vertebrate	Invertebrate	0.33	0.24	0.15	0.1
Living thing	Living thing	1	0.64	0.28	0.1
Vertebrate	Vertebrate	1	0.72	0.44	0.3

the example. The heading shows various values of w_1 , which represents specificity. It can be seen that the adjustment reduces the similarity between generalized concepts by a factor proportional to the weight.

The unadjusted similarities of the pairs (Living thing, Living thing) and (Vertebrate, Vertebrate) are both 1. Vertebrate is less likely to be dissimilar with another vertebrate, while a living thing is more likely to be dissimilar with another living thing. However, the low similarity or high dissimilarity of a concept with itself, especially for high weight, cannot be justified logically. Resnik measure also behaves the same way, but this is not consistent with the reality.

To explain the issue, let us take an example of two person p_1 and p_2 , who are both IT professionals, working in banking applications, both play tennis, both are young. Assume that all these characteristics have high specificity, and are shared by p_1 and p_2 . Our system based on SemSimA will give high similarity between them. The overall similarity between two person is the average of (high) individual similarity values resulting from the comparison of a special concept with itself. A

new fact is registered in the system, saying both of them are Africans. Assume that this characteristic is a general one. Now comparison of a general concept with itself gives low similarity (using SemSimA or Resnik) due to small IC. This low similarity value will be contributed to the overall similarity between two persons, which is the average of individual similarities. Therefore, this new fact will drop the overall similarity between p_1 and p_2 .

This scenario applies to proteins where, instead of person, we have two proteins and instead of common characteristics, we have common functions. We would consider two proteins having common functions corresponding to special GO terms, and therefore having high similarity. Assigning a new general GO term to both proteins will drop their similarity. When two different concepts are compared, the above adjustment will work fine for specialized as well as generalized concepts.

We define SemSimB which, instead of making a generalized concept dissimilar with itself, gives 100% similarity, but lowers the confidence level Conf. For $c_1 \neq c_2$,

$$SemSimB(c_1, c_2) = SemSimA(c_1, c_2), c_1 \neq c_2 \quad (3.7)$$

For $c_1 = c_2 = c$

$$SemSimB(c, c) = 1 \quad (3.8)$$

$$Conf = \frac{w_1 \cdot Spec + w_2 \cdot Uniq}{w_1 + w_2} \quad (3.9)$$

Rather than returning a single value, this model returns two values: similarity and confidence level. Confidence level determines how much certainty or uncertainty is present in the similarity value, when a concept is compared with itself. In case of two different concepts, there is no uncertainty. The client application will process the result accordingly. When computing similarity between two proteins using Best Match Average for example, weighted average of best matching pairs

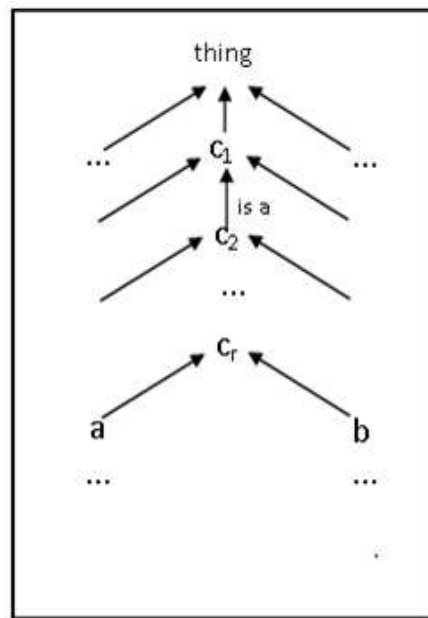


FIGURE 3.1: A hypothetical ontology.

may be taken with Conf being used as weight. A generalized GO term common in two proteins will contribute to their similarity rather than dissimilarity. For maximum effect of specificity and uniqueness, both weights would be set to 0.5. We recommend using SemSimB, unless the client application requires a single similarity value.

Let us consider a hypothetical ontology shown in Figure 3.1. There is a chain of descendants such that c_1 is a thing, c_2 is a c_1 , c_r is a c_{r-1} . Assume that ICs are, root:0, c_1 :1, c_2 :2, c_3 :3, and so on. Ellipses mean further taxonomy of concepts.

Similarities computed by SemSim and existing methods between two pairs (a, b) and (a, a) are listed in Table 3.3, assuming r is 6, depth of ontology is 12 ($d=12$) and maximum IC in the ontology is 12. Lin, SimGIC, Wang and AIC compute similarity regardless of the descendants of a and b . Normalized Resnik similarity shown in parentheses, and SemSim are sensitive to the taxonomy below a and b . SemSim returns two values: similarity and confidence (certainty). Since a and b are general concepts in this example, SemSim gives low similarity between them. It gives similarity=1 and confidence=0.58 for pair (a, a) .

TABLE 3.3: Similarity computed by SemSim and existing methods (r=6, Figure 3.1).

Pair	Resnik	Lin	SimGIC	Wang	AIC	SemSim
a, b	6(0.50)	0.86	0.6	0.76	0.91	(0.35, 1)
a, a	7(0.58)	1	1	1	1	(1, 0.58)

3.2 Protein Functional Similarity Method - SimExact

To compute semantic similarity between two proteins, GO terms that annotate these proteins are compared, and then the Best Match Average (BMA) of the term similarities is taken. Protein pairs, especially those with high similarity, have some common GO term annotations. A component of protein semantic similarity results from the exact match between such GO terms. Most methods give a semantic similarity of 1 when a GO term is compared with itself. To study this, we split protein similarity into two components: the first one resulting from exact match, and the second one resulting from partial match of GO terms. We define semantic similarity measure *SimBinary*, that returns 1 for identical GO terms (exact match) and 0 otherwise.

$$\text{SimBinary}(c_1, c_2) = 1 \text{ for } c_1 = c_2, 0 \text{ for } c_1 \neq c_2 \quad (3.10)$$

In Best Match Average, only the best matching term pairs are significant, and all other term pairs are ignored. For each common term between two proteins, the best match is exact match, and all other terms are ignored. Coming to the second component, we compute semantic similarity using various methods, but ignore the 1's resulting from the comparison of identical GO terms. We therefore define a fractional component of each measure that computes similarity only if it is a fraction, and ignores the exact matching case of $c_1 = c_2$. *SimResFrac*, the fractional component of Resnik is defined as:

$$SimResFrac(c_1, c_2) = SimRes(c_1, c_2) \text{ for } c_1 \neq c_2, \text{ for } c_1 = c_2 \quad (3.11)$$

Fractional components of Jiang Conrath, Lin and other measures are computed in the same manner. The purpose is to study each component of similarity in isolation. We apply these variants as well as SimBinary to protein pairs by taking BMA of the similarity values of individual term pairs, and therefore define the following measures for protein comparison:

$$SimBinP(p_1, p_2) = BMA(SimBinary(c_i, c_j) \forall c_i \in Annot(p_1), c_j \in Annot(p_2)) \quad (3.12)$$

$$SimRFP(p_1, p_2) = BMA(SimResFrac(c_i, c_j) \forall c_i \in Annot(p_1), c_j \in Annot(p_2)) \quad (3.13)$$

SimBinP is SimBinary as applied to proteins p_1 and p_2 , and $Annot(p_i)$ is the set of concepts (GO terms) annotated to p_i .

From initial experiments, it was hypothesized that the semantic similarity is mainly influenced by common terms. We explored the possibility to exploit this for high speed similarity computation applications. We computed SimBinP for our data set, and observed its behavior as compared to other similarity measures.

SimBinary gives similarity of 1 for identical GO terms. If two proteins share a common term, it will contribute 1 to their similarity, no matter how informative the term is. Most existing measures work the same way. However, this means ignoring the information content. Sharing a general term and sharing a specific term should have different effects. We propose SimExact, which is enhanced form of SimBinary, and is based on contribution from common terms between the proteins to be compared. We combine IC_{ont} and IC_{ann} to get IC_{comb} .

$$IC_{comb}(c) = w_1 \cdot IC_{ont}(c) + w_2 \cdot IC_{ann}(c) \quad (3.14)$$

Where w_1 and w_2 are weights given to two ICs respectively such that $w_1 + w_2 = 1$. In our experiments, $w_1=0.5$, $w_2=0.5$ was used, in which case combined IC is the mean of two ICs. Two leaf GO terms, one having few annotations and the other having many annotations will have same IC_{ont} , but the former will have higher IC_{ann} than the latter.

SimExact makes an improvement in SimBinary. Instead of giving 1 for all matching terms, SimExact discriminates general terms from specific terms. SimExact at term level (SimEx) is defined as

$$SimEx(c_1, c_2) = IC_{comb}(c_1) \text{ for } c_1 = c_2, 0 \text{ for } c_1 \neq c_2 \quad (3.15)$$

Now we define SimExact at protein level. MatchScore for proteins p_1 and p_2 is a score based on the terms shared by the two proteins.

$$MatchScore(p_1, p_2) = \sum_{t \in T_{p_1 \cap p_2}} IC_{comb}(t) \quad (3.16)$$

MatchScore is based on number and IC of common terms rather than the percentage of common terms. An advantage of this scheme is that it handles the case of under-annotated proteins. A pair having few but 100% common terms will have lower score than a pair having many and 100% common terms. However, the effect of uncommon terms is not counted so far. We define Penalty as a score based on the uncommon terms in proteins p_1 and p_2

$$Penalty(p_1, p_2) = \sum_{t \in (T_{p_1 \cup p_2}) - (T_{p_1 \cap p_2})} IC_{comb}(t) \quad (3.17)$$

SimExact is then defined as

$$\text{SimExact}(p_1, p_2) = \text{MatchScore}(p_1, p_2) - \text{Penalty}(p_1, p_2) \quad (3.18)$$

SimExact has no definite upper and lower limits. It can be negative for very dissimilar proteins, and can be more than hundred for proteins with many common GO terms. It will make distinction between pairs all having 100% common terms, but having different annotation strengths.

SimBinary and SimExact can be used to design high speed similarity computation programs, especially, protein query answering programs. It can be safely assumed that the similar proteins do have some common GO terms. When we are interested in most similar proteins, common GO terms would not only be present, they would be in majority. This can be used for formulation of efficient search algorithm. The algorithm would only look for proteins that have GO terms common with the query protein, rather than scanning the whole database. With the use of powerful database queries, we can select the proteins similar to a query protein with an order of high efficiency. Experiments are required to assess whether SimBinary and SimExact can produce results with reasonable accuracy.

3.3 Implementation and Experimental Environment

We implemented Resnik, Jiang Conrath and Lin, which are relatively old measures, and are referenced frequently in the literature. GraSM technique was applied on these three measures. Thus we had Resnik GraSM, Jiang Conrath GraSM and Lin GraSM in addition to the above three measures. We implemented two later measures SimUI and SimGIC that are based on Jaccard index, and are suitable for ontology having multiple parents. SimGIC, which is based on IC, was frequently used in our experiments. We also need to experiment with latest measures. We implemented two recent measures Wang and AIC. Thus we have 10 well known measures implemented for our experiments.

TABLE 3.4: Database table ‘TermPair’.

term1	term2	semsim	simres	simlin	wang	aic
GO:0000166	GO:0000287	0.015593	1.21927	0.210385	0.195372	0.401487
GO:0000166	GO:0003677	0.080613	2.66626	0.639772	0.500182	0.67147
GO:0000166	GO:0003785	0.017212	1.21927	0.164286	0.195372	0.401111
GO:0000166	GO:0005102	0.017268	1.21927	0.277177	0.267437	0.457796
GO:0000166	GO:0005515	0.017234	1.21927	0.339025	0.326116	0.495353
GO:0000166	GO:0005524	0.086848	4.73321	0.826889	0.46713	0.54415
GO:0000166	GO:0008022	0.022774	1.21927	0.18828	0.267437	0.45971
GO:0000166	GO:0017124	0.01992	1.21927	0.181942	0.225794	0.428496

Our implementation strategy was to integrate all relevant data into a single database. We chose GO database as our primary database, created new tables, and added new fields in the existing tables where required. This way, we were able to use the power of SQL to process and retrieve the required data from a unified database.

Table 3.4 shows a table ‘TermPair’ that we added to this database. For comparing two proteins, we need to compare $m \cdot n$ GO term pairs, where m and n are numbers of GO term annotations of the proteins. Instead of throwing away the results of term comparison, we save them in this table. On subsequent calls to term comparison function, this table is searched. If the term pair exists in the table, its results are reused rather than computed again. The table, which has more fields than shown, is also used for the analysis of term similarity. In the same way, new fields were added in the existing tables to save the values of IC, semantic weight etc., so that these values are reused instead of computing them again.

We computed semantic similarity between protein pairs using Best Match Average (BMA) approach. Sequence similarity was computed using BLOSUM62 matrix and scores were divided by sequence length.

3.4 Protein Query Algorithm/ Tool

To compute semantic similarity between two proteins having m and n annotations respectively, $m \cdot n$ term comparisons are required. To process a query for proteins similar to a given protein from the database of p proteins, $p \cdot n^2 \cdot g$ operations would be required where n is the average annotations of a protein, and g is the complexity of GO term similarity computation algorithm. This will be practically too hard for applications.

Tradeoff between accuracy and speed: There is an obvious need for increased speed of protein query processing. One possibility is to compromise on accuracy and gain acceptable speed. The same is done in case of BLAST search. Of course, the compromise on accuracy should be kept to a minimum. We design a method for protein query processing based on SimBinary (or SimExact), with an order of high speed. Our technique eliminates the need to visit every protein in the database. The algorithm is shown in Figure 3.2. The loss of accuracy is assessed experimentally. This method can be combined with state-of-the-art semantic similarity methods where it is used as a pre-processor, to short list the similar proteins for state-of-the-art method.

If there are t terms annotated with a protein, and r proteins annotated with a term (on average), $t \cdot r$ operations will be required to process the query. In general, t and r are too small as compared to p , the number of proteins in the database.

The algorithm can be used in combination with state of the art similarity measures. In this scheme, our algorithm acts as a guide, and calls other similarity measurement methods in a smart way, only for selected pairs. The modified algorithm is shown in Figure 3.3. After computing similarity $simp_b$ using SimBinary, it is checked against a preset threshold. If the similarity is more than threshold, methods are called to compute similarities $simp_i$ by existing measures i , which may be slow but more accurate. At the end, the result is sorted by $simp_u$, the similarity by measure u , selected by user.


```

SemQuery ( $p_q$ )
{
  initialize result as empty list
  query for  $T = \text{terms}(p_q)$ , the set of terms annotated with  $p_q$ 
  for each  $t$  in  $T$ 
  {
    query for  $P = \text{prot}(t)$ , the set of proteins annotated with  $t$ 
    for each  $p$  in  $P$ 
      If  $p$  not in result
      {
         $c =$  number of terms common in  $p$  and  $p_q$ 
         $a =$  number of all terms annotated with  $p$ 
         $\text{sim}_p = c / a * 100$ 
        add  $p, \text{sim}_p$  to result
      }
    }
  }
  sort result on  $\text{Sim}_p$  in descending order
  output top  $n$  proteins of result
}

```

FIGURE 3.2: Algorithm for processing protein query using SimBinary

Let us determine the complexity of this algorithm. The algorithm needs to compute the number of terms common in two proteins, and also the total number of terms annotated with the proteins. This is determined from the database of annotations. Assuming that the fields are indexed, reading from the database is of logarithmic complexity. The algorithm involves database read operations proportional to $t \cdot r$ where t is the number of terms annotated to a protein, and r is the number of proteins annotated with a term. Each read operation has complexity $\log(a)$ where a is the number of rows in the annotation database. Therefore, the complexity of our algorithm is $t \cdot r \cdot \log(a)$.

The power of relational database technology can be exploited to implement the algorithm. Let us consider the SQL given in Figure 3.4. This SQL is a complete and efficient implementation of the algorithm. When SimExact is used instead of

```

SemQueryFull ( $p_q$ )
{
  initialize result as empty list
  query for  $T = \text{terms}(p_q)$ , the set of terms annotated with  $p_q$ 
  for each  $t$  in  $T$ 
  {
    query for  $P = \text{prot}(t)$ , the set of proteins annotated with  $t$ 
    for each  $p$  in  $P$ 
      If  $p$  not in result
      {
        compute  $a$  and  $c$ , number of annotations of  $p$  and terms
        common with  $p_q$ 
         $\text{sim}_{pb} = c / a * 100$ 
        if  $\text{sim}_{pb} > \text{threshold}$ 
        {
           $\text{Sim}_i = \text{FullSim}(p, p_q)$ 
          add  $p, \text{sim}_{pb}, \text{sim}_i$  to result
        }
      }
    }
  }
  sort result on  $\text{Sim}_{pu}$  in descending order
}

```

FIGURE 3.3: Modified algorithm for processing protein query

```

SELECT prot, count(*) AS cnt FROM annotation WHERE term IN
(SELECT term FROM annotation
WHERE prot = <query protein>)
GROUP BY prot ORDER BY cnt DESC

```

FIGURE 3.4: SQL implementation of protein query processing

SimBinary, ICont and ICann are also involved. The two ICs are computed and saved in the database against each GO term, and this is done only once in the start. The term table is then joined in the SQL. This affects the execution speed, though not drastically.

We have developed our query tool in PHP and mySql, and it is available online at www.datafurnish.com/protsem.php. After successful trial run, it can be provided

as a unique service to the scientists working in this area. A screen shot is shown in Figure 3.5. User enters a query protein id and clicks on the Submit button. The proteins having high semantic similarity with the query protein are listed in the lower part of the page, ranked according to their similarity.

The first column is protein id, the second one is protein name, and the third one is the similarity (SimExact) with query protein. Id and name of the query protein is displayed below the submit button. The similar proteins are retrieved in few seconds, even faster than the state-of-the-art BLAST tools. User may want to run a second query on one of the proteins in the result set displayed on the page. To do this, he will simply click on the id of the desired protein in the list. To query for some other protein, he will type its id in the text box and press the submit button.

This page only gives SimExact, and is therefore fast. The user may opt to compute similarity by other implemented methods. To do this, he checks the box labeled ‘compute full similarity’ located on the upper right part of the page. The tool now computes similarity using other measures too. Because of SimExact acting as a guide, the other measures are only computed on the selected pairs. This adds to the efficiency to a large extent. Still the user experiences a considerable delay in the response. The resulting page is shown in Figure 3.6.

Everything on this page is same as in the previous one, except that new column are shown for SimGIC, Resnik, Lin and Jiang Conrath measures. Now user can see and easily compare similarity values from multiple measures. The ranking is still on SimExact. In future versions, the user will be allowed to choose another measure for ranking.

3.4.1 Searching Protein Pairs

Processing all possible protein pairs is computationally expensive. SimBinary can be used to identify protein pairs making interesting patterns. For example, we can fix a window of interest on semantic similarity, and find large number of pairs

Protein Query Tool	
Find related proteins based on semantic similarity	
Enter Query Protein	<input type="text"/> Submit <input type="checkbox"/> Compute full similarity
Q9JKJ9 24-hydroxycholesterol 7-alpha-hydroxylase	<input type="checkbox"/> Use Information Content
Protein	Similarity
Q9NYL5 24-hydroxycholesterol 7-alpha-hydroxylase	174.4
Q60991 25-hydroxycholesterol 7-alpha-hydroxylase	111.6
Q64505 Cholesterol 7-alpha-monooxygenase	82.4
Q9NR63	46.2
P23219 Prostaglandin G/H synthase 1	36.8
P35354 Prostaglandin G/H synthase 2	28.2
P27786 Steroid 17-alpha-hydroxylase/17.20 lyase	21.9
G4NG13	9.9
P26439 3 beta-hydroxysteroid dehydrogenase/Delta 5-->4-isomerase type 2	8.7
P10635 Cytochrome P450 2D6	6.7
P08684 Cytochrome P450 3A4	5.7
P29475 Nitric oxide synthase, brain	4.5
P29476 Nitric oxide synthase, brain	4.5
P29474	4.5
Q9Z0J4 Nitric oxide synthase, brain	4.5

FIGURE 3.5: Screen shot of protein query tool SemQuery showing SimExact similarity

falling within that window. We applied this method to find large number of pairs with semantic similarity between 0.8 and 1.

A useful outcome of this method is to identify the pairs, in large number, with high disparity between sequence similarity and semantic similarity. These are the pairs having semantic similarity near 1 and sequence similarity near 0.1, meaning different sequences performing highly similar functions. Some of such pairs may help identify annotation errors. An example is P02185 Myoglobin and P69891 Hemoglobin subunit gamma-1, having very high semantic similarity of 1 and very low sequence similarity of 0.1. This may be due to some error or incompleteness in annotations. Another example is P02239 Leghemoglobin-1 and P01958 Hemoglobin subunit alpha having high semantic similarity of 0.88 and very low sequence similarity of 0.1.

Protein Query Tool						
Find related proteins based on semantic similarity						
Enter Query Protein		Submit	<input checked="" type="checkbox"/> Compute full similarity			
Q60991 25-hydroxycholesterol 7-alpha-hydroxylase			<input type="checkbox"/> Use Information Content			
Protein	SimExact	SimBin	SimGIC	Resnik	Lin	Jiang
Q9NYL5 24-hydroxycholesterol 7-alpha-hydroxylase	119.6	89	96	65	97	92
Q9JKJ9 24-hydroxycholesterol 7-alpha-hydroxylase	119.6	89	94	64	97	92
Q64505 Cholesterol 7-alpha-monooxygenase	90.5	78	84	57	91	81
Q9NR63	59.2	61	70	57	85	67
P23219 Prostaglandin G/H synthase 1	50.8	56	68	53	84	63
P27786 Steroid 17-alpha-hydroxylase/17.20 lyase	36.9	53	65	53	83	60
P35354 Prostaglandin G/H synthase 2	43.2	50	63	57	82	59
G4NG13	25.9	50	62	45	81	58
Q61263 Sterol O-acyltransferase 1	31.6	47	54	45	70	54
P26439 3 beta-hydroxysteroid dehydrogenase/Delta 5->4-isomerase type 2	25.7	39	57	60	79	51
Q9Z0J4 Nitric oxide synthase, brain	21.5	41	54	49	73	50
P29476 Nitric oxide synthase, brain	21.5	41	56	50	74	50
P08684 Cytochrome P450 3A4	22.7	39	59	65	78	50
P29474	21.5	39	57	50	76	49
P10635 Cytochrome P450 2D6	23.7	39	56	62	79	49
Q01237 3-hydroxy-3-methylglutaryl-coenzyme A reductase	15.1	33	54	69	81	48

FIGURE 3.6: Screen shot of protein query tool SemQuery showing SimExact and other similarities

3.4.2 Sequence-Semantic Conflict

Experiments indicated many protein pairs having low sequence similarity but high semantic similarity. We propose formal method for identifying this conflicting behaviour of proteins. Let us denote sequence similarity between proteins p_1 and p_2 by $seq(p_1, p_2)$ and semantic similarity between them by $sem(p_1, p_2)$. We consider protein triples (p_0, p_1, p_2) where p_0 is the reference protein, and each of p_1 and p_2 are compared with it. We define sequence-semantic conflict to exist if

$$(seq(p_0, p_1) - seq(p_0, p_2)) \cdot (sem(p_0, p_1) - sem(p_0, p_2)) < 0 \quad (3.19)$$

$$(seq(p_0, p_1) - seq(p_0, p_2)) \cdot (sem(p_0, p_2) - sem(p_0, p_1)) < 0 \quad (3.20)$$

The expression on the left hand side of these equations will quantify the conflict. The conflict will be significant if it is greater than a threshold t . SimBinary was used to find triples with maximum conflict. The conflict could also be found simply by searching for pairs having high semantic and low sequence similarity. However, the above model is more convincing because it would show that one protein is closer to p_0 in a sequence-based query, while the other is closer in a semantic-based query.

The strategy of this search algorithm is to start with the candidate pairs (p_0, p_1) that lie on the upper left portion of a scatter plot between sequence similarity and semantic similarity. That is, p_1 has high semantic similarity but low sequence similarity with p_0 . Once we find such pair, we use SimBinary to find candidates for p_2 such that p_2 has relatively lower semantic similarity with p_0 . We compute sequence similarity of p_2 with p_0 and test for conflict. If conflict is found, but its magnitude is too small as compared to the maximum conflict found so far, the candidate is dropped.

3.4.3 Similarity Inferred from Sequence Similarity

For some investigations, we measure similarity between GO terms inferred from the sequence similarity of proteins that annotate the GO terms. To compare terms t_1 and t_2 , each protein that annotates t_1 is compared with each protein that annotates t_2 . All sequence similarity values thus obtained are averaged. The average shows how similar the proteins corresponding to t_1 and t_2 are. We call it similarity inferred from sequence similarity (SISS), defined below.

$$SISS(t_1, t_2) = \frac{\sum_{p_i \in P_1, p_j \in P_2} seq(p_i, p_j)}{|P_1 \times P_2|} \quad (3.21)$$

Where P_1 is the set of proteins annotating t_1 , P_2 is the set of proteins annotating t_2 , and Seq is sequence similarity between two proteins. This should be useful for some investigations on behavior of proteins.

Chapter 4

Comparison And Correlation Between Similarity Measures

This chapter furnishes results of the experiments related with the existing semantic similarity measures. Semantic similarity measures were compared to find out agreement or disagreement between them. The measures were calibrated so that the results can be interpreted in a better way. Correlation between sequence similarity and semantic similarity was studied from different aspects.

4.1 Experiment 4.1: Distribution of Similarity Measures

This experiment was done to see the distribution of the similarity values computed by different methods. There may be a difference in the numeric value returned by these methods. Then question arises, that how these values would be compared. The distributions will help us to observe the pattern of similarities of different methods. Then this pattern can be used to normalize the data of similarity measures in five different classes (very low, low, moderate, high and very high). This calibration will help us to convert the quantitative measure of similarity into qualitative measure given by domain experts in their research papers and then compare

the methods with the data by domain experts. Distributions of GO term similarity can be different from those of protein similarity. This is because in BMA, which is used for proteins, similarity values other than best match are not included. Therefore, the experiment is performed first on GO terms and then on proteins.

Semantic similarity was computed for the GO term pairs of Dataset 1, using various methods (Resnik GraSM, Lin GraSM, SimGIC, Wang, AIC), and the distribution of the similarity was plotted (frequency of pairs against the similarity value). The same was done for protein pairs of Dataset 1 using BMA. Figure 4.1 shows these distributions for GO term pairs, where the similarity range (0 to 1) is divided into 10 bins. The columns represent the percentage of pairs that fall in the respective bin. The graphs show quite different distributions for different methods. A high frequency in low similarity bins was expected since being dissimilar, in general, is a common case whereas being similar is a special case. That is, if we pick two GO terms randomly, they would have high probability of being dissimilar.

Since the similarity values computed (or normalized) by the methods are between 0 and 1, the applications using these methods would interpret the values near 0 as very low, and those near 1 as very high. However, it can be observed from the graphs that very low or very high may be different for different methods. A similarity value, say 0.5, may not mean the same in every measure. Therefore, it would be useful to calibrate each measure based on its distribution in the range 0 to 1. We propose a calibration in which the similarity values output by various methods are mapped to 5 categories: very low, low, moderate, high and very high. A frequency of less than 1% in a bin (Figure 4.1) is considered insignificant. Starting from left, the first bin with significant frequency is mapped to very low, and the next one to low. Similarly, the first bin from right with significant frequency is mapped to very high, and the next one to high. The remaining bins in the middle are mapped to moderate. Resnik GraSM is an exception, where frequency is concentrated in four bins (0 to 0.4), which is equally divided into five categories. The calibration is shown in Table 4.1, we call it GO calibration.

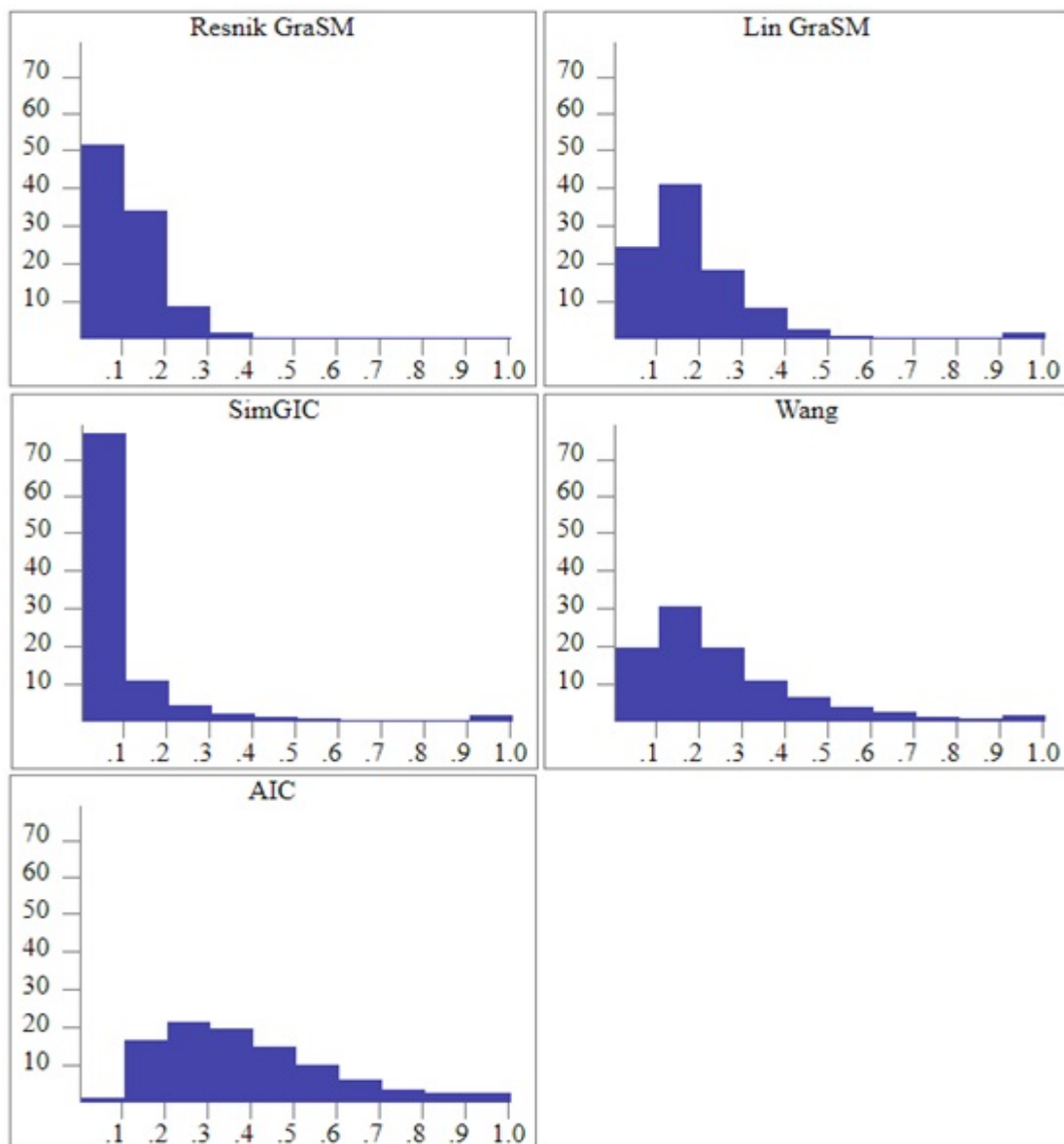


FIGURE 4.1: Frequency distribution of GO term semantic similarity values for various methods.

The distribution of similarity values for proteins is shown in Figure 4.2. Calibration of similarity measures was done as explained above, which is shown in Table 4.2.

This calibration is useful when we compare semantic similarity computed by the above methods with some independent evidence, especially when the evidence is expressed qualitatively. For instance, protein pairs of Dataset 6 have evidence of very high similarity according to Lesk and Chothia [20]. Protein pairs of Dataset 7 have evidence of very low similarity according to Trad et al. [21]. These very low or very high similarities given by domain experts are qualitative measures.

TABLE 4.1: GO calibration of semantic similarity measures.

	Very low	Low	Moderate	High	Very high
Normal	0 to 0.1	0.1 to 0.2	0.2 to 0.8	0.8 to 0.9	0.9 to 1
Resnik G.	0 to 0.08	0.08 to 0.16	0.16 to 0.24	0.24 to 0.3	0.3 to 1
Lin G.	0 to 0.1	0.1 to 0.2	0.2 to 0.5	0.5 to 0.9	0.9 to 1
SimGIC	0 to 0.1	0.1 to 0.2	0.2 to 0.4	0.4 to 0.9	0.9 to 1
Wang	0 to 0.1	0.1 to 0.2	0.2 to 0.8	0.8 to 0.9	0.9 to 1
AIC	0 to 0.2	0.2 to 0.3	0.3 to 0.8	0.8 to 0.9	0.9 to 1

TABLE 4.2: Calibration of protein semantic similarity for various measures.

	Very low	Low	Moderate	High	Very high
Normal	0 to 0.1	0.1 to 0.2	0.2 to 0.8	0.8 to 0.9	0.9 to 1
Resnik G.	0 to 0.2	0.2 to 0.3	0.3 to 0.6	0.6 to 0.7	0.7 to 1
Lin G.	0 to 0.3	0.3 to 0.4	0.5 to 0.8	0.8 to 0.9	0.9 to 1
SimGIC	0 to 0.1	0.1 to 0.2	0.3 to 0.8	0.8 to 0.9	0.9 to 1
Wang	0 to 0.4	0.4 to 0.5	0.5 to 0.8	0.8 to 0.9	0.9 to 1
AIC	0 to 0.5	0.5 to 0.6	0.7 to 0.8	0.8 to 0.9	0.9 to 1

To compare these quantitative measures with quantitative results of the similarity measuring methods (Resnik, Lin, Jiang Conrath etc.), the calibration is essential.

4.2 Experiment 4.2: Correlation Between Similarity Measures

This experiment was performed to explore whether different similarity measures agree or disagree with one another. If various methods give significantly different results, then there is a possibility that some or most of them have problems. We study the correlation between semantic similarity measures at GO term level.

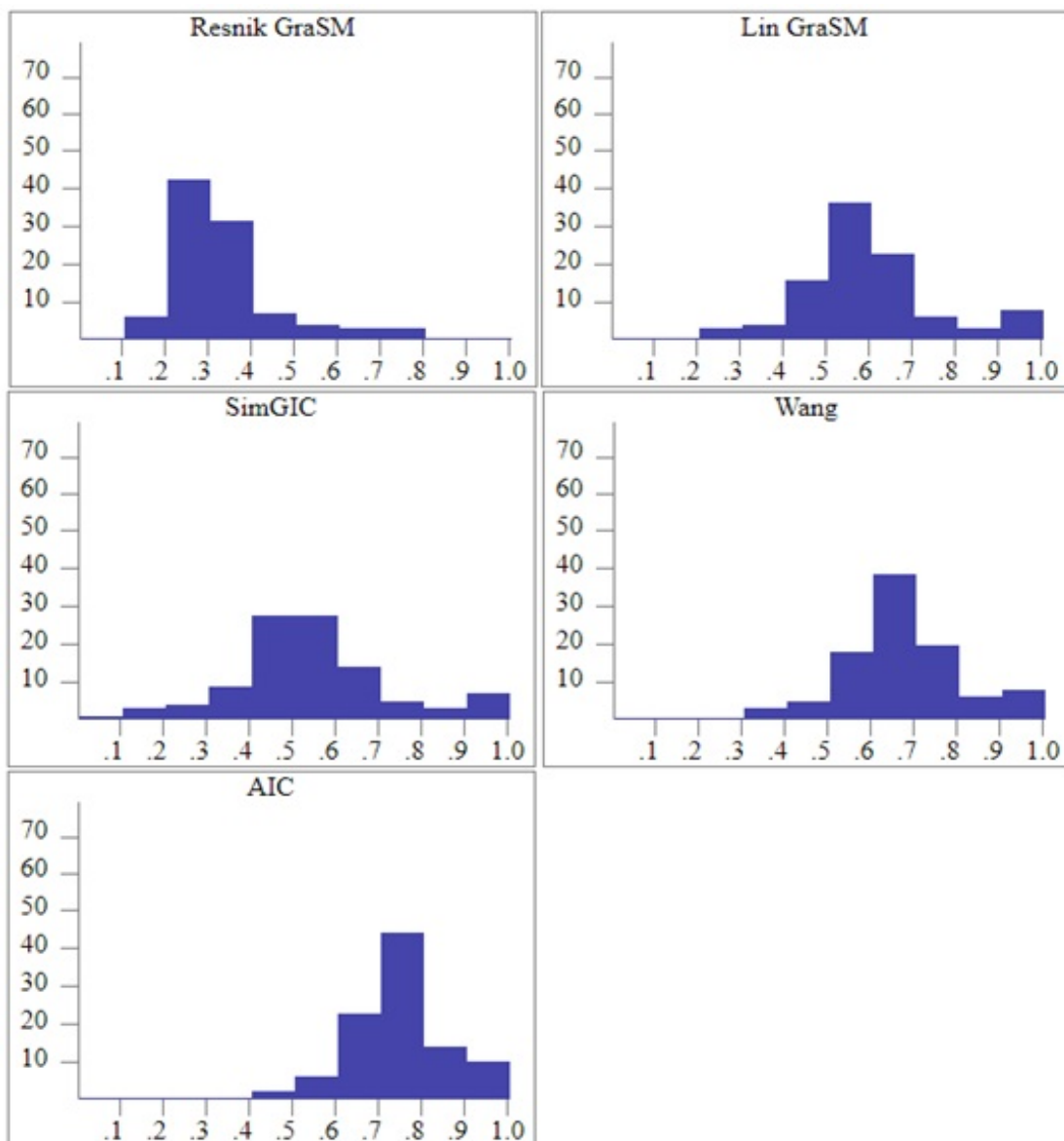


FIGURE 4.2: Frequency distribution of protein semantic similarity for various methods.

Semantic similarity was computed between GO term pairs of Dataset 1, and the results of all the methods were compared. Each method is compared with all other methods as shown in table 4.3.

Figure 4.3 shows only two scatter plots of these term pairs (due to limitations of space, all the combinations are not shown): Wang against Resnik GraSM, and AIC against SimGIC. The graphs show that the computed similarity is not consistent across different methods. There are several pairs with low Resnik GraSM similarity but high Wang similarity. SimGIC and AIC also show a considerable disagreement.

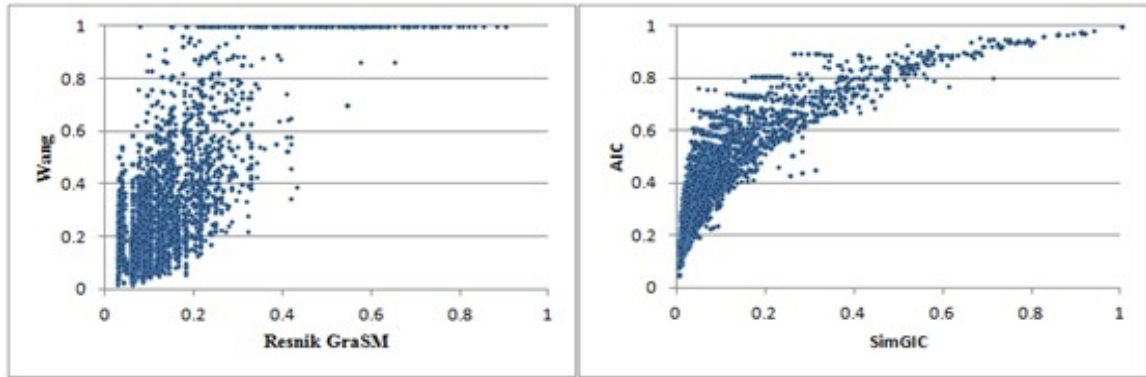


FIGURE 4.3: Scatter plots of GO term similarities: Wang against Resnik GraSM, AIC against SimGIC.

TABLE 4.3: Correlation between various semantic similarity measures.

	Resnik G.	Lin G.	SimGIC	Wang	AIC
Resnik G.	1	0.86	0.79	0.69	0.62
Lin G.		1	0.88	0.82	0.75
SimGIC			1	0.9	0.81
Wang				1	0.97
AIC					1

Table 4.3 shows Pearson’s correlation coefficient between the results of various methods. The experiment shows that there is significant disagreement between different methods of semantic similarity measurement. Appendices A, B, C and D list term pairs with high disagreement among different measures. This supports results of our initial experiment and the reasoning given in Section 2.4. This also answers the first part of research question 1 as given in section 1.1.

4.3 Experiment 4.3: Rank Correlation Between Similarity Measures

Different similarity values computed by various methods for same pairs may be attributed to different scale and distribution used by these methods. If two similarity measures rank the terms similar to a reference term in identical order, the measures would be seen as compatible by most applications, even if the similarity values are different. The experiment is carried out to explore this dimension, using rank correlation coefficient.

In this experiment, we take a reference GO term r , find n most similar terms to r , and rank them by their similarity to r using each method. The rank (1 for most similar term, 2 for next, and so on) according to each method is saved in the database, and this process is repeated for considerable number of reference terms (Dataset 3). We observed that different methods rank the terms differently. We compute Spearman's rank correlation coefficient, which is equal to Pearson's correlation coefficient between the ranks of terms given by two methods. This determines the agreement or disagreement between the methods, without being sensitive to their scale or distribution. The Spearman's rank correlation coefficients between various methods are given in Table 4.4. Table 4.5 lists top 10 terms similar to an example reference term GO:0000053 along with their similarity to the reference term and the rank given by each method. The terms are sorted on the average similarity given by all methods in descending order.

It can be seen from Table 4.4 and Table 4.5, that there is considerable disagreement among the methods even on ranking of similar terms. This also answers the first part of research question 1 as given in section 1.1.

TABLE 4.4: Spearman’s rank correlation between similarity ranks given by semantic similarity measures.

	Resnik G.	Lin G.	SimGIC	Wang	AIC
Resnik G.	1	0.89	0.78	0.69	0.72
Lin G.		1	0.79	0.72	0.68
SimGIC			1	0.96	0.94
Wang				1	0.97
AIC					1

TABLE 4.5: Top ten terms similar to GO:0000053 with their similarity (rank).

Term	Resnik G.	Lin G.	SimGIC	Wang	AIC
GO:0046418	0.263 (1)	0.312 (4)	0.754 (1)	0.891 (1)	0.956 (1)
GO:0000052	0.227 (5)	0.289 (14)	0.674 (2)	0.840 (2)	0.932 (3)
GO:0006600	0.227 (6)	0.287 (16)	0.673 (3)	0.840 (3)	0.932 (2)
GO:0050667	0.243 (3)	0.312 (3)	0.586 (8)	0.776 (7)	0.891 (8)
GO:0009066	0.193 (41)	0.292 (12)	0.618 (4)	0.784 (6)	0.907 (6)
GO:0009064	0.193 (40)	0.290 (13)	0.617 (5)	0.784 (6)	0.907 (5)
GO:0046442	0.245 (2)	0.295 (10)	0.602 (7)	0.768 (8)	0.878 (12)
GO:0006591	0.193 (39)	0.267 (27)	0.606 (6)	0.784 (4)	0.907 (4)
GO:0016259	0.227 (7)	0.277 (24)	0.556 (9)	0.752 (9)	0.892 (7)
GO:0006573	0.213 (24)	0.281 (21)	0.548 (10)	0.748 (12)	0.886 (11)

4.4 Experiment 4.4: Correlation Between Semantic Similarity and Semantic Similarity

In this experiment, we study the relationship between protein semantic (function) similarity and sequence similarity. This is required for answering RQ2. Some earlier studies have reported that there is a significant correlation between the two

measures. Some studies expected that as the accuracy of semantic similarity measures improves, this correlation will get stronger. Accordingly, it has been used for evaluating the correctness of semantic similarity measures. There is also evidence of protein pairs that deviate from this trend, and go against the expectation of a significant correlation.

Semantic similarity and sequence similarity were computed for protein pairs of Dataset 1 and Dataset 2. To show the relationship between the two types of similarity, we make scatter plots, where sequence similarity is plotted along x-axis, and semantic similarity along y-axis. Figure 4.4 shows scatter plots for Resnik, Lin, SimGIC and AIC for Dataset 1. Pearson's correlation coefficients for various measures are: Resnik: 0.76, Lin: 0.67, SimGIC: 0.69, and AIC: 0.66.

All the graphs depict the absence of a significant correlation between the two similarity measures. In Resnik graph, there are several points having low sequence similarity and varying values of semantic similarity. Lin similarity shows a similar behavior, with a large number of points clustering between 0.3 and 0.9 semantic similarity and less than 0.2 sequence similarity. Wang and AIC graphs have the similar formation as well, with a column of points from 0.4 to 1 semantic similarity and low sequence similarity. Upper left region (1) in all graphs shows protein pairs that have low sequence similarity but high semantic similarity. Lower right region (2) is empty, indicating that there are no protein pairs with high sequence similarity and low semantic similarity. Protein pairs in upper right region (3) have high values of both sequence and semantic similarity.

Figure 4.5 shows the scatter plots of semantic similarity against sequence similarity for SimGIC and Lin similarity measures for non-IEA annotations and protein pairs of Dataset 2. Correlation coefficients are: SimGIC: 0.64, Lin: 0.62. It can be observed from these graphs that proteins can have varying values of semantic similarity for low values of sequence similarity. Similarly, varying values of sequence similarity for high values of semantic similarity are also observed.

The Pearson's correlation coefficient alone is not sufficient to explain the nature of this relationship. It can be misleading if graphs are not studied carefully. Several

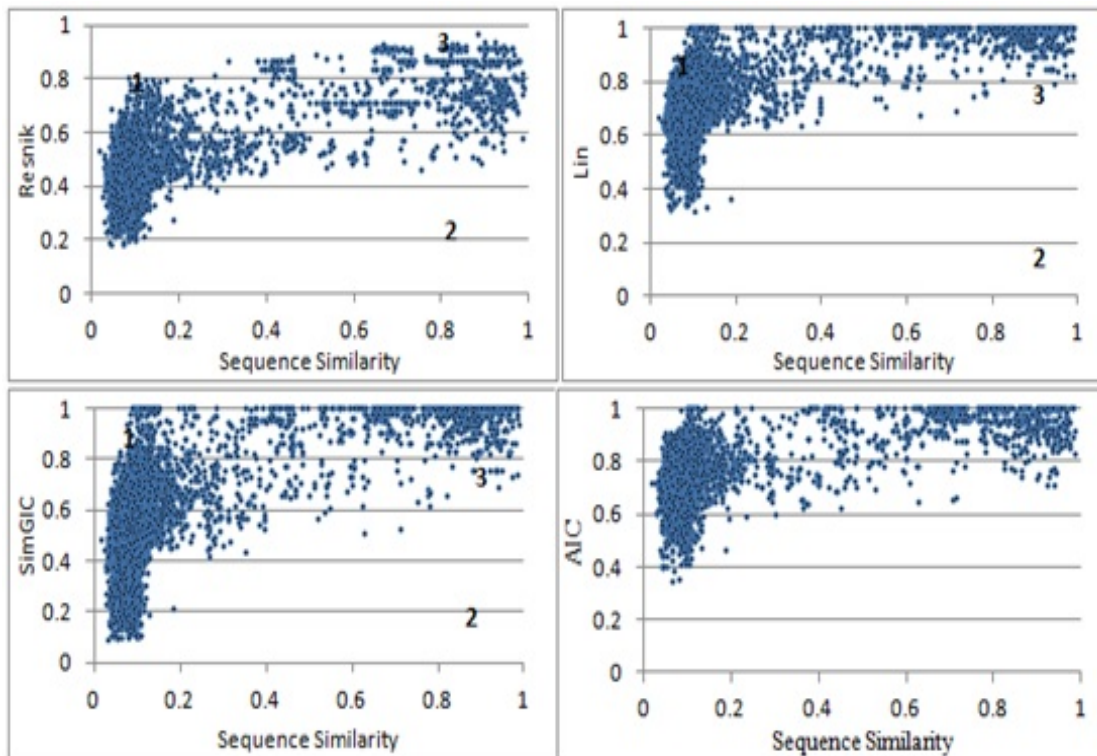


FIGURE 4.4: Scatter plot of protein semantic similarity against sequence similarity for Resnik, Lin, SimGIC and AIC for Dataset 1.

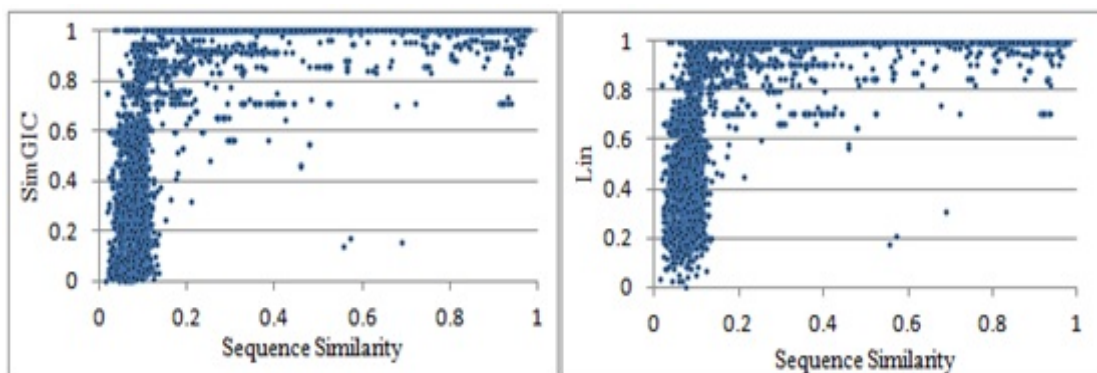


FIGURE 4.5: Scatter plots of SimGIC, Lin against sequence similarity for non-IEA annotations.

graph formations are possible for a given value of correlation coefficient, say 0.7. This is in agreement with the findings of Anscombe [33] who show the scatter plots of 4 sets of data, each having Pearson's correlation coefficient of 0.8. After looking at the graphs, one can observe that some of them have really good correlation while others are weakly correlated.

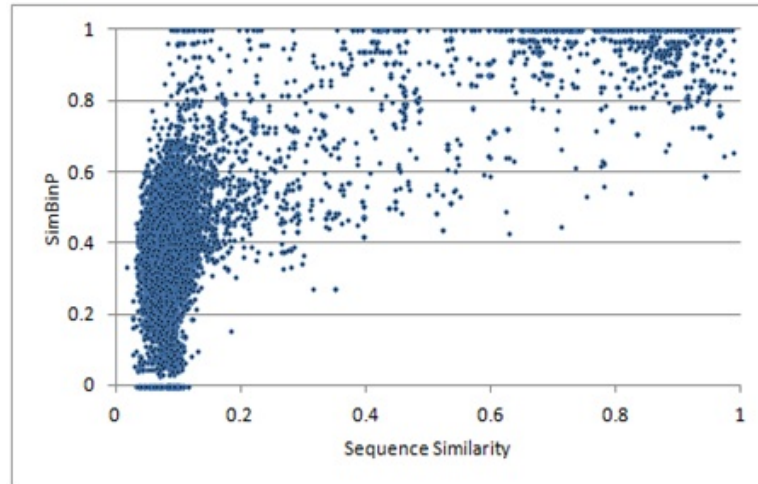


FIGURE 4.6: Scatter plot of protein semantic similarity against sequence similarity for SimBinP. Correlation: 0.72.

4.5 Experiment 4.5: Effect of Common GO Terms

This experiment was performed to investigate the effect of common GO terms on the correlation between semantic similarity and sequence similarity. For common terms, the semantic similarity measures compare a term with itself. This is a simple comparison, where similarity is 1 for most methods. Dataset 1, Dataset 4 and Dataset 5 were used for various investigations in this experiment. Semantic similarity between the pairs of these datasets was computed using various methods, and plotted against sequence similarity. The scatter plot for SimBinP against sequence similarity is shown in Figure 4.6.

SimBinP computes similarity of identical terms as 1 and all other terms as 0. The result is quite surprising, since it exhibits a correlation similar to that of figure 4.4. The correlation coefficient is 0.72, which is not very different from the previous ones.

Figure 4.7 shows the scatter plot of SimGIC for protein pairs having zero common GO terms (Dataset 4). Figure 4.8 shows the similar graph for protein pairs having non-zero common GO terms (Dataset 5). Dataset 4 exhibits no correlation, while Dataset 5 exhibits correlation of 0.69, comparable with Dataset 1. This shows that

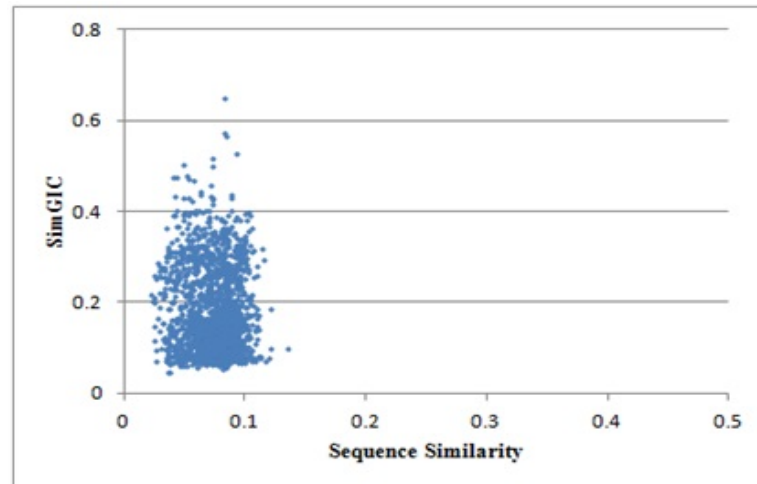


FIGURE 4.7: Scatter plot of SimGIC against sequence similarity for Dataset 4.
Correlation: 0.

the correlation is actually contributed by the 1's resulting from the comparison of a GO term with itself, which is due to the common terms.

To filter out the effect of exact match between GO terms, we computed semantic similarity using SimResFrac, SimJiangFrac, SimLinFrac etc. Figure 4.9 shows the scatter plots for SimJiangFrac (GraSM version) and that of SimAICFrac. Both the graphs are flat, showing almost same semantic similarity pattern for the lowest sequence similarity pairs and for the highest sequence similarity pairs. This again shows that the correlation is actually contributed by the 1's resulting from the comparison of a GO term with itself, which is due to the common terms. This suggests that the correlation is not a suitable criterion for the evaluation of semantic similarity measures, because the similarity is computed between not exactly same terms rather between somewhat different terms. This answers research question 3 given in Section 1.1.

Clearly, SimBinary is not a precise similarity measure, and is an approximation. The well known semantic similarity measures are far more precise than SimBinary at term level. However, in the given system of Gene Ontology and protein annotations, the term similarity values above 0 and below 1 do not seem to play a role in improving the correlation between semantic similarity and sequence similarity. This interesting behavior is further investigated in Chapter 6.

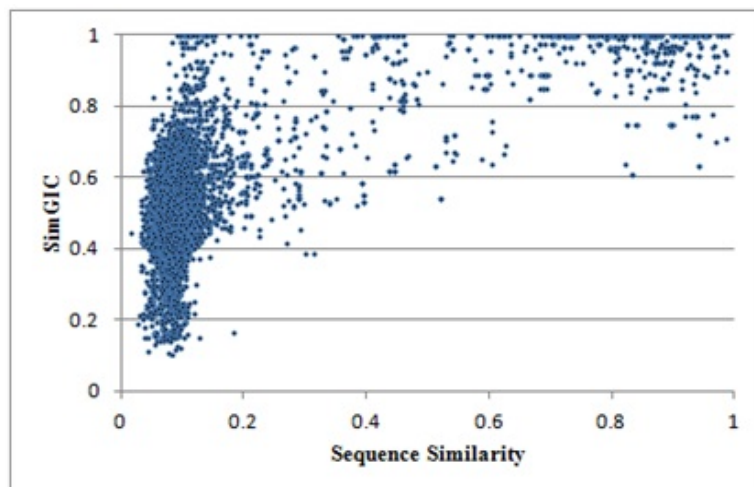


FIGURE 4.8: Scatter plot of SimGIC against sequence similarity for Dataset 5. Correlation: 0.69.

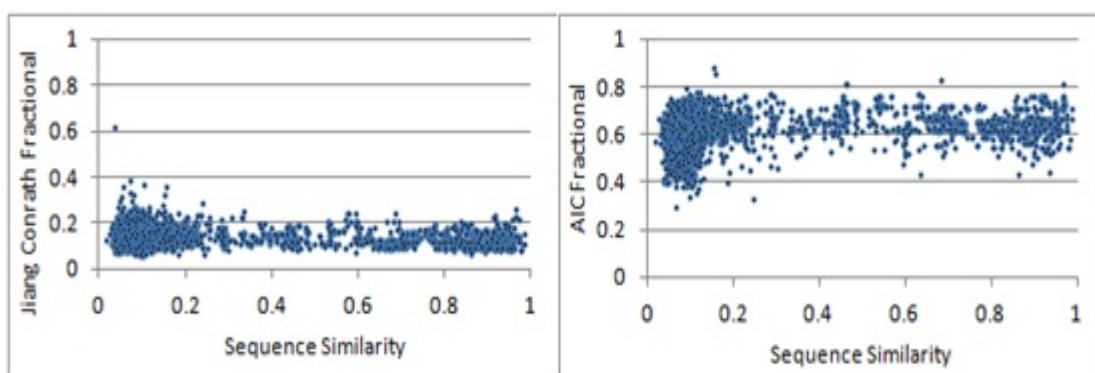


FIGURE 4.9: Scatter plot of Jiang Conrath Fractional, AIC Fractional against sequence similarity .

4.6 Experiment 4.6: Correlation at Sub Ontology Level

In this experiment, we compute the correlation for three sub ontologies of Gene Ontology. This is required for investigation on RQ2 at sub ontology level. Semantic similarity and correlation coefficient were computed for protein pairs of Dataset 1 for sub ontologies: molecular function, biological process and cellular component. Table 4.6 shows the Pearson's correlation coefficient between sequence similarity and semantic similarity. The correlation coefficients are in the same range as observed in previous experiments.

TABLE 4.6: Pearson’s correlation coefficient between sequence similarity and semantic similarity.

	Molecular Function	Biological Process	Cellular Component
SimRes	0.579	0.671	0.677
SimLin	0.554	0.657	0.642
SimJC	0.642	0.749	0.731
SimResG	0.602	0.737	0.728
SimLinG	0.547	0.695	0.661
SimJCG	0.646	0.752	0.732
SimGIC	0.558	0.683	0.664

4.7 Discussion

The results show a weak or moderate correlation between semantic similarity and sequence similarity. The results differ from some earlier studies assuming good correlation between sequence and semantic similarities. Though Pearson’s correlation coefficient is a commonly used indicator of such relationship, it hides fine details about the formation of points on the 2-D space of a graph. A small increment in the correlation coefficient may not mean an improvement in the performance of similarity measurement method; it could be due to fluctuations.

All scatter plots show a strong bias towards cases with low sequence similarity. There are more pairs with sequence similarity less than 0.2, and less pairs with high sequence similarity. This was expected since the sequences would be similar in special cases, while they would be generally dissimilar (i.e. there are few recent paralogs). However, non-linearity can be clearly seen.

A careful observation of the scatter plots suggests that there are two separate regions in the graphs, corresponding to two ranges of sequence similarity. Left region corresponds to sequence similarity values from 0 to 0.2, whereas right region corresponds to the values from 0.2 to 1. In left region, semantic similarity varies from 0.1 to 1 almost independent of sequence similarity. In contrast to left region

where points appear vertically, right region has most points appearing horizontally. In this region, semantic similarity remains high, showing some correlation with sequence similarity. The graphs show that semantic similarity has its own significance and uniqueness. Its characteristics are distinct from sequence similarity. There are very few points showing high sequence similarity and low semantic similarity. This means that if a protein has high sequence similarity with another protein, it will most likely have high semantic similarity with it. In short, high sequence similarity means probably high semantic similarity.

The points showing low sequence similarity but high semantic similarity represent the missing results from a protein query to state of the art tools. This issue is very important because most modern research in the area uses BLAST search tools that are based on sequence similarity. In contrast to sequence similarity, which is static, semantic similarity is dynamic. With new versions of Gene Ontology and ever growing annotation knowledge, the semantic comparisons are expected to be more meaningful.

We have thus answered research question 2 given in Section 1.1. Absence of significant correlation can have different explanations.

The most convincing reason is that the nature of semantic similarity is distinct from sequence similarity, and the two actually are not correlated. A point appearing near the upper left corner of the plot might move slightly down when the similarity measure, annotations or GO improves, but one would not expect it to move altogether to the lower left corner, even with the perfect tools and knowledge resources. There are several indications in the literature, that proteins with different sequences can have similar functions. Moonlighting proteins, for example, can switch between distinct functions. In convergent evolution process, proteins of organisms, not closely related, may have independently evolved to perform similar functions. Polymorphic protein systems consist of different proteins performing same function [25]. Proteins having evolutionary relationships can have different sequences, but similar functions. During divergent evolution, the sequences can change and diverge from one another, while performing functions.

Results of Experiment 4.5 have useful implications. Surprisingly, there is a good contribution to correlation from common terms, and no contribution from uncommon ones. Annotation problems are common in both datasets. Can we suspect that GO relationships or semantic similarity measures are responsible for this? While we guess that GO relationships are mostly free from errors, problem may lie in the interpretation of these relationships, and converting them to numeric values. For example, can it be guaranteed that all pairs of sibling terms across GO will have uniform similarity? GO:0000024: maltose biosynthetic process, and GO:0000025: maltose catabolic process are two sibling terms, having one common and one distinct parent each. A similarity measure would yield high similarity between them, while they are semantically opposite. To overcome such problems, ontologies with rich semantics, and similarity measures capable of capturing these semantics would be required.

Researchers have studied the role of protein domains in similarity. Thompson et al. [36] find that structure similarity searches using domain boundaries based on conserved sequence information may give us additional method to identify similarities between proteins. This results in improvement in performance of structure similarity searches.

Pathways can also be used in the context of protein similarity. Biological pathways and networks are based on protein-protein interactions. Guo et al. [34] evaluated semantic similarity measures on their ability to characterize human regulatory pathways.

Chapter 5

Results Of Novel Similarity Measures

It is important to assess the performance of our methods as compared to existing methods, using some benchmark. As discussed in Chapter 2, there is no gold standard for the evaluation of semantic similarity methods. Different studies have used different approach for the evaluation, and the community does not appear to agree on a single approach which we should use in our research. The most commonly used approach is the correlation with sequence similarity, which is not suitable as discussed earlier.

if, by some means, we can acquire set(s) of protein pairs with known functional similarity, that can be used for the evaluation of semantic similarity methods. The reliability of evaluation will depend upon how well the similarity within the pairs is known. Our approach is to look for pairs having confirmed experimental evidence of either very low or very high similarity. We took help of a biologist and searched the published literature for protein pairs with reliable evidence of functional similarity between them. We found one paper providing protein pairs with very high similarity, and another providing pairs with very low similarity. Thus we got protein pairs whose functional similarity is confirmed by domain experts. We used these pairs for our evaluation experiments.

TABLE 5.1: Globins having high secondary and tertiary structural similarity.

Id	Name
P69905	Hemoglobin subunit alpha
P68871	Hemoglobin subunit beta
P01958	Hemoglobin subunit alpha
P02062	Hemoglobin subunit beta
P02185	Myoglobin
P02221	Globin CTT-I/CTT-IA
P02239	Leghemoglobin-1
P69891	Hemoglobin subunit gamma-1
P68080	Myoglobin

Assessing SimBinary and SimExact is not difficult, since their objective is to search for proteins similar to a query protein in a minimum possible time. We perform experiments with SemSim, SimBinary and SimExact and show their results in this chapter. Results are traced for sample pairs using GO graphs and annotations.

5.1 Experiment 5.1: Evaluation Using Highly Similar Proteins

This experiment is performed to evaluate our semantic similarity method SemSim. Evaluating semantic similarity measures is difficult. No globally accepted criterion exists for such evaluation. We found two studies that list proteins with evidence of function similarity between them. Lesk and Chothia [9] discuss proteins that have different sequence, but highly similar secondary and tertiary structure. They identify nine globins that have high structure and function similarity. They are listed in Table 5.1.

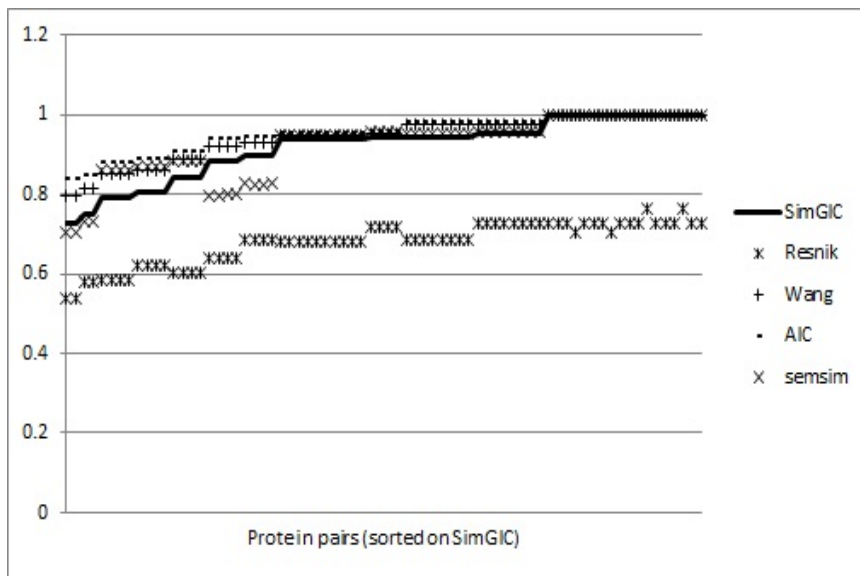


FIGURE 5.1: Similarity between pairs of globins by various methods.

We performed this experiment based on evidence from Lesk and Chothia (Dataset 6). We measured function similarity between all possible protein pairs using Sem-Sim and state-of-the-art semantic similarity methods. Figure 5.1 shows the graph of similarity measured by various methods. Using normal calibration (Table 4.2), SimGIC, Wang, AIC and SemSim give very high and high similarity between these proteins, in agreement with the evidence given in the study. Resnik GraSM gives lower similarity as compared to other measures, with no pair above 80%, and few pairs below 60%.

5.2 Experiment 5.2: Evaluation Using Highly Dissimilar Proteins

This experiment too is performed to evaluate our semantic similarity method Sem-Sim. This experiment is based on the evidence from de Trad et al. [10] who devised a protein comparison method that examines a protein sequence at different spatial resolutions. They identified 8 protein pairs with dissimilar functions (Dataset 7). For example, Lysozyme and hemoglobin do not share any biological function, which is very low function similarity. The graph of similarity values is shown in

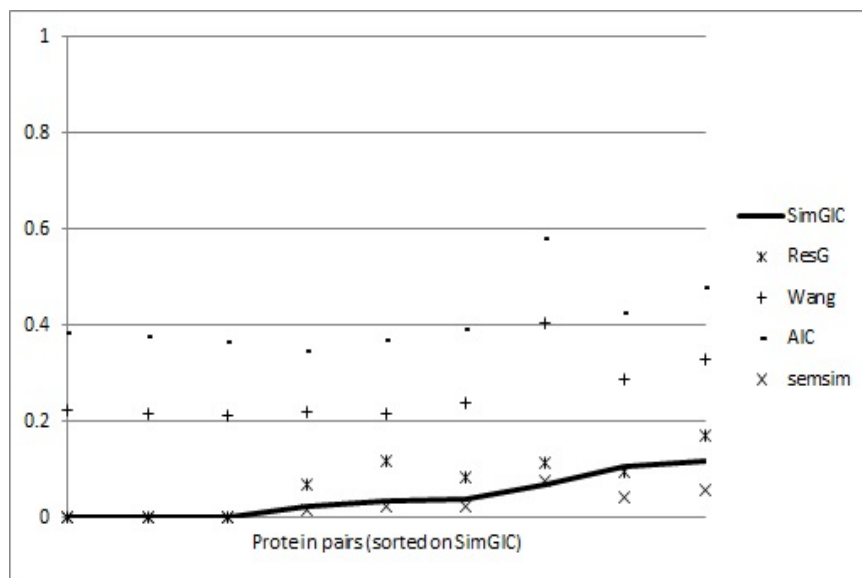


FIGURE 5.2: Similarity between protein pairs of Trad et al. by various methods.

Figure 5.2. SimGIC, Resnik GraSM and SemSim give very low similarity, in agreement with the evidence given in the study. Wang and AIC give higher similarity values.

5.3 Experiment 5.3: Evaluation With General GO Terms

When a term is compared with itself, the similarity is generally 1. Protein pairs frequently have terms common between them. The case of general terms (located near root) is interesting. Resnik gives very low similarity when a general term is compared with itself, while SimGIC, Wang and AIC give similarity value of 1. To observe the behavior of various measures on generalized common terms, this experiment is focused on such terms. Protein pairs of Dataset 8 were used for this experiment.

Figure 5.3 shows the graph of similarities measured by various methods for these pairs. AIC gives highest similarity, followed by Wang and then SimGIC. SemSim and Resnik GraSM give lower similarity.

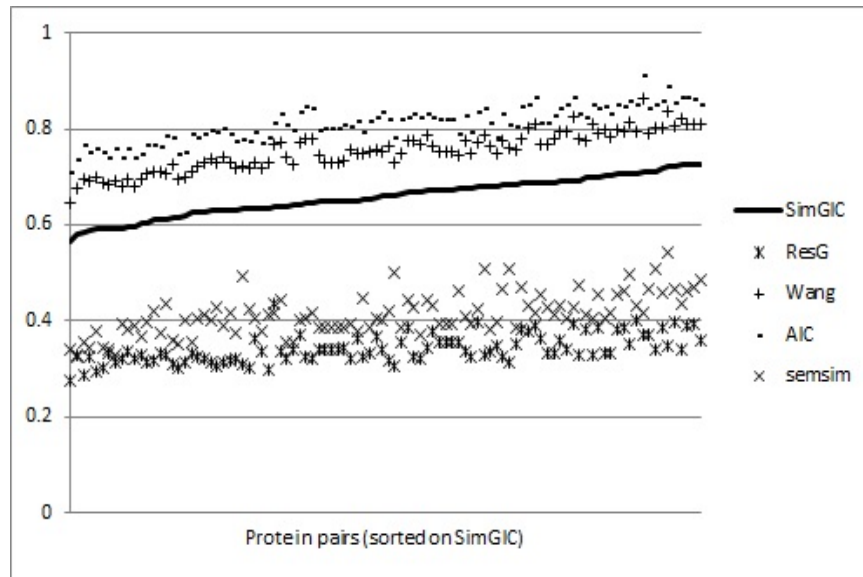


FIGURE 5.3: Similarity between protein pairs having generalized common terms.

SemSim is the only measure that produces valid results according to the published evidence and in case of general terms.

5.4 Experiment 5.4: Comparing SemSim With Existing Measures

In this experiment, we compare SemSim with existing measures, and with sequence similarity. We need to observe what difference SemSim makes as compared to existing methods. We also trace the reasons why SemSim differs with the existing methods in each case, and if it is justified. Semantic similarity was computed for GO term pairs of Dataset 1 using SemSim and the existing methods. SemSim similarity was plotted against other measures, and against sequence similarity. Then GO term pairs were picked from various regions of scatter plot for detailed inspection. These pairs were traced using Gene Ontology graphs to inspect whether the computed similarity conforms to the semantics of the ontology. The GO graphs are not shown due to the limitations of space. Finally, some protein pairs were picked from the graphs of previous experiment, and their similarity was checked. With

this exercise, we could also verify that our implementation of various methods is free from programming errors.

Figure 5.4 shows scatter plot of GO term semantic similarity SemSim A against Resnik GraSM. Figure 5.5 shows similar graph for SemSim B against Resnik GraSM, the weights being $w_1 = 0.5, w_2 = 0.5$. SemSim A and SemSim B differ only when $c_1 = c_2$. In this case, SemSim A gives a fraction while SemSim B gives 1, along with a confidence level. Wang and AIC give a flat value of 1 (without confidence level) when $c_1 = c_2$. Figure 5.6 shows scatter plots for SemSim A against Wang and Figure 5.7 for SemSim A against AIC. It can be seen that SemSim A gives low similarity as compared to Wang and AIC in many cases.

Correlation with sequence similarity for SemSim A ($w_1=w_2=0.3$) is 0.77, and for SemSim A ($w_1 = w_2 = 0.5$) it is 0.83. We used maximum weights ($w_1 = 0.5, w_2 = 0.5$) for SemSim B. The scatter plot is shown in Figure 5.8, and correlation is 0.84. However, we do not rely on this correlation to validate SemSim. In Figure 5.6, we inspected pairs from various regions of the graph. Table 5.2 lists 9 of such pairs with semantic similarity values from various measures.

In the first row, there is not much difference among various measures, except AIC giving a higher similarity. Manual inspection shows that the terms have mostly uncommon ancestors, justifying low similarity. In the second row, Wang gives higher similarity of 0.41, while AIC gives still higher value of 0.54. Manual inspection of graph shows many uncommon ancestors, asking for a low semantic similarity. Relatively higher value by Wang is because the common ancestors contribute twice, and because the root may have considerable semantic value if the terms are not too deep. Further higher value by AIC is because of root having the highest weight of 1. The scenario of third row is the same as that of second row.

In the fourth row, Wang gives similarity of 0.60 while AIC gives higher similarity of 0.79. Manual inspection finds many uncommon ancestors, agreeing with the measures giving low similarity. Fifth row also has the same scenario. The terms of sixth row are GO:0003677-DNA binding and GO:0003723-RNA binding, which

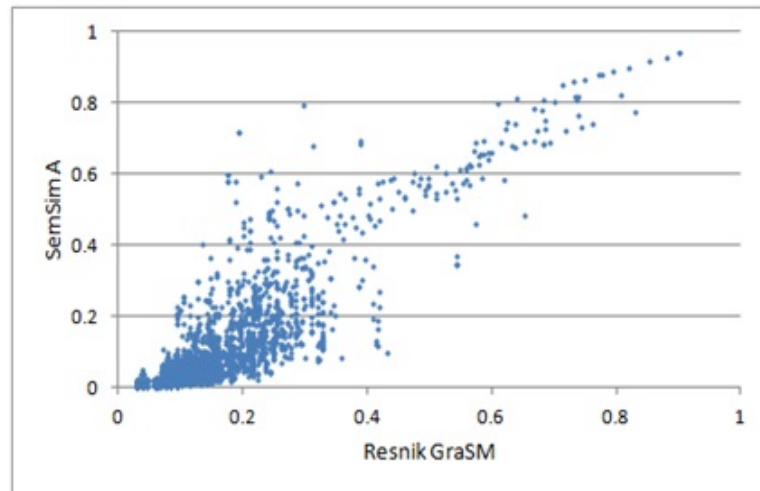


FIGURE 5.4: Scatter plot of SemSim A against Resnik GraSM.

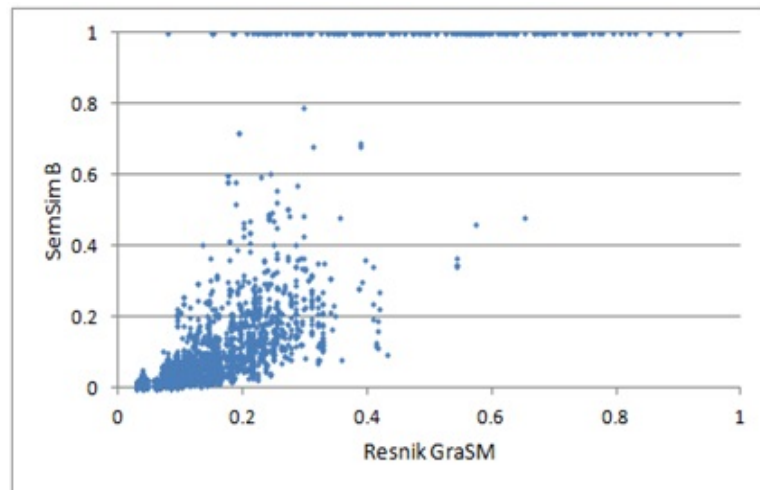


FIGURE 5.5: Scatter plot of SemSim B against Resnik GraSM.

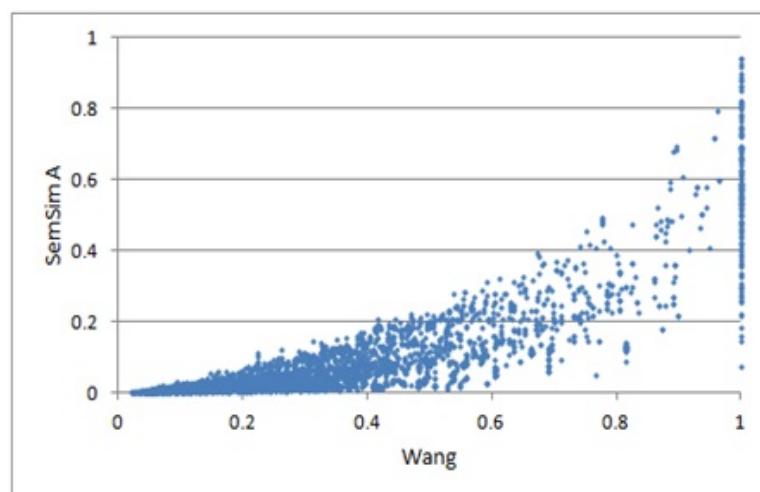


FIGURE 5.6: Scatter plot of SemSim A against Wang.

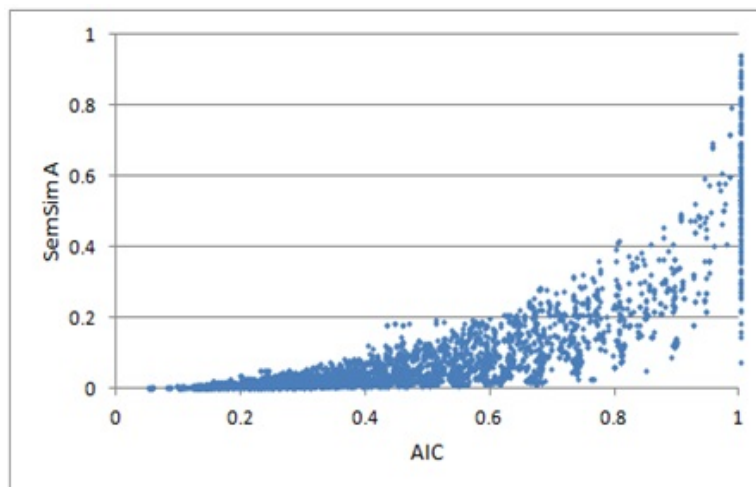


FIGURE 5.7: Scatter plot of SemSim A against AIC.

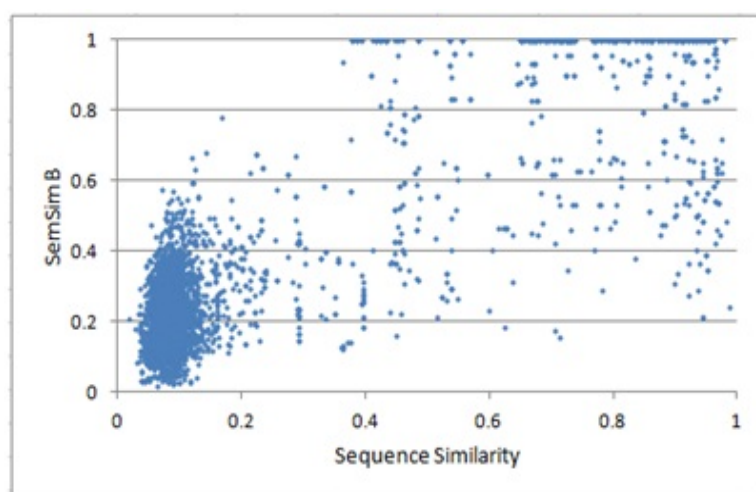


FIGURE 5.8: Scatter plot of protein semantic similarity against sequence similarity for SemSim B.

are generalized sibling terms, 4 levels below root. The low similarity computed by Resnik GraSM and SemSim is close to human perception. These two measures incorporate Resnik Factor, giving low similarity between close generalized concepts (like plant and animal). In the seventh row, Wang gives 0.81 while AIC gives a higher similarity of 0.90. SemSim gives a lower similarity of 0.21. Manual observation shows that the terms are not far apart, with few uncommon ancestors. However, the terms are general, falling near root. A high share in AIC value, and a reasonable share in Wang is from the root being common ancestor of the two terms. SimGIC yields a lower value of 0.31 because it gives zero weight to root, and counts the common ancestors once. The low value by SemSim is due to the

low IC of the terms, and low value by Resnik is due to low IC of the MICA.

In the eighth row, a term ‘Protein binding’ is compared with itself. All measures except Resnik GraSM and SemSim A give similarity of 1. This is a general term, having a small IC. Resnik GraSM gives a low similarity of 0.15, and it has a reason for this, as discussed before. However, this may have adverse effect in some applications since it may contribute to dissimilarity. SemSim, in its initial stage, works like SimGIC, SimUI, Wang and AIC, yielding similarity of 1. In the later stage, it computes confidence level according to the specificity and uniqueness. SemSim A makes adjustment in the similarity and makes it 0.19, close to Resnik GraSM. SemSim B (not listed in the table) returns similarity of 1 along with confidence of 0.19, and lets the client application treat the two values.

In the ninth row too, a term is compared with itself, but this time the term is more specific. Both Resnik GraSM and SemSim A now give higher similarity due to the higher IC of the term. SemSim B gives similarity of 1 and confidence of 0.89. Resnik GraSM gives same value of 0.15 for row 7 and 8, although row 7 has two different terms while row 8 has identical terms.

We also traced some protein pairs from the graphs of Figure 4.4, especially those having sequence similarity not in line with semantic similarity. In these graphs, there are pairs having low sequence similarity and high semantic similarity. We want to trace the measurement to investigate whether this clash is due to some weakness in the method, or it actually exists in the system. Some of the traced pairs are listed in Table 5.3. The pair of first row is picked based on high disparity between sequence similarity and SemSim B. Both proteins of this pair have exactly same GO terms annotated to them. Therefore, similarity of 1 given by Wang, AIC and SemSim B seems more logical as compared to SemSim A and Resnik GraSM. It also shows that two proteins having low sequence similarity may have high semantic similarity. The pair of second row is picked for high disparity between sequence similarity and Wang similarity. The proteins have mix of common and uncommon GO terms. Wang and AIC give similarity of 1 for common terms. SemSim A and Resnik GraSM give low similarity for generalized common terms.

TABLE 5.2: Similarity computed by SemSim and existing methods.

Term1	Term2	ResG	JiangG	Wang	AIC	GIC	SSimA
GO:0051220 cytoplasmic sequestering of protein	GO:0050764 regulation of phagocytosis	0.14	0.07	0.19	0.37	0.07	0.05
GO:0005739 mitochondrion	GO:0032839 dendrite cytoplasm	0.07	0.09	0.41	0.54	0.09	0.05
GO:0097038 perinuclear endoplasmic reticulum	GO:0005798 Golgi-associated vesicle	0.10	0.06	0.42	0.56	0.13	0.07
GO:0009755 hormone-mediated signaling pathway	GO:0032886 regulation of microtubule-based process	0.14	0.07	0.60	0.79	0.15	0.09
GO:0004674 protein serine/threonine kinase activity	GO:0016361 activin receptor activity, type I	0.18	0.09	0.61	0.67	0.31	0.17
GO:0003677 DNA binding	GO:0003723 RNA binding	0.13	0.21	0.75	0.86	0.54	0.16
GO:0008022 protein C-terminus binding	GO:0005515 protein binding	0.15	0.15	0.81	0.90	0.31	0.13
GO:0005515 protein binding	GO:0005515 protein binding	0.15	1	1	1	1	0.19
GO:0071773 cellular response to BMP stimulus	GO:0071773 cellular response to BMP stimulus	0.79	1	1	1	1	0.89

SemSim B makes correction in SemSim A by giving similarity of 1 for common terms, but reducing weight of generalized terms. For instance, a term GO:0003824-catalytic activity is common between the two proteins. The term is generalized, located one level below root. Wang and AIC give a flat similarity value of 1 for comparison of this term with itself. Resnik GraSM and SemSim A give 0.09 and 0.08 respectively. SemSim B gives similarity of 1 and weight of 0.08. Rather than decreasing overall similarity, this pair slightly increases it, like a course in an academic transcript with 100% score but small weight (credits) slightly raises the overall grade.

The pair of third row is picked for high disparity between sequence similarity and Resnik GraSM. Again, the proteins have mix of common and uncommon terms.

TABLE 5.3: Protein similarity computed by SemSim and existing methods.

Protein 1	Protein 2	Seq.	ResG	Wang	AIC	SSimA	SSimB
P31347 Angiogenin	Q8WN62 Angiogenin	0.40	0.50	1	1	0.58	1
P98194 Calcium-transporting ATPase type 2C member 1	Q03669 Sarcoplasmic/endoplasmic reticulum calcium ATPase 2	0.12	0.29	0.85	0.89	0.35	0.52
P45975 Histone-lysine N-methyltransferase Su(var)3-9	P38827 Histone-lysine N-methyltransferase, H3 lysine-4 specific	0.08	0.32	0.80	0.85	0.39	0.42
P50993 Sodium /potassium-transporting ATPase subunit alpha-2	Q6PIE5 Sodium /potassium-transporting ATPase subunit alpha-2	0.93	0.34	0.80	0.86	0.36	0.50
P48734 Cyclin-dependent kinase 1	P06493 Cyclin-dependent kinase 1	0.94	0.24	0.62	0.71	0.25	0.46
Q5RA62 TFIIH basal transcription factor complex helicase XPB subunit	Q60HG1 TFIIH basal transcription factor complex helicase XPB subunit	0.94	0.47	1	1	0.55	1

In the fourth row, proteins have quite high sequence similarity, but semantic similarity by all methods is not much different from third row. The proteins have mix of common and uncommon terms. One of the term pairs consists of GO:0055085-transmembrane transport and GO:0006811-ion transport. These are generalized sibling terms, three levels below root. Wang and AIC give high similarity for this pair, while Resnik GraSM, SemSim A and SemSim B give low similarity due to generality of the terms, as discussed earlier. The pair has 14 generalized common terms, having confidence level less than 0.5. In Resnik GraSM and SemSim A, these contribute to dissimilarity whereas in Wang and AIC, they contribute to similarity, but their generality is ignored. In SemSim B, they contribute to similarity with low weight. Fifth row too shows high sequence similarity, but all methods agree on low semantic similarity. Pair of sixth row is one of those pairs that have high sequence similarity as well as high semantic similarity. Both proteins are annotated to exactly same GO terms, like row 1. Result of Wang, AIC and SemSim B is closer to reality than Resnik GraSM and SemSim A.

5.5 Discussion

SimUI, SimGIC, Wang, AIC and SemSim have advantage over Resnik GraSM and Jiang Conrath GraSM that they consider contribution of all common and uncommon ancestors, and are therefore suitable for ontologies having DAG structure. These methods represent a class of semantic similarity measures that are futuristic, and are widely accepted by researchers. SemSim and Resnik GraSM have an edge over all other measures that they effectively distinguish the general terms from specific terms. SemSim has a unique advantage of being a member of the class of modern similarity measures, while retaining a good characteristic of the original measure, Resnik.

It cannot be proved that a computed similarity between two GO terms or proteins is correct. No standard benchmark is available for evaluation of similarity measures. We try to evaluate our method in best possible way. Experiment 5.1 and 5.2 show that the similarity computed by SemSim is in agreement with that reported in the published literature. During inspection of the results, we did not find a case where SemSim B similarity value appears to disagree with the taxonomy. Other measures do disagree in one case or another. SemSim is the only measure that produces valid results according to the published evidence. Results of SemSim are close to reality in all cases. Especially, it treats the generalized concepts according to the semantics of the domain. This answers research question 4 given in Section 1.1.

5.6 Experiment 5.5: Results of Protein Query Tool

In this experiment, we assess the performance of protein query tool ProtQuery, and the possible loss of accuracy in the methods used for it. This is required for addressing RQ5. First, we compare SimBinP with Jiang Conrath GraSM, the measure closest to SimBinP. Similarities were computed for protein pairs of

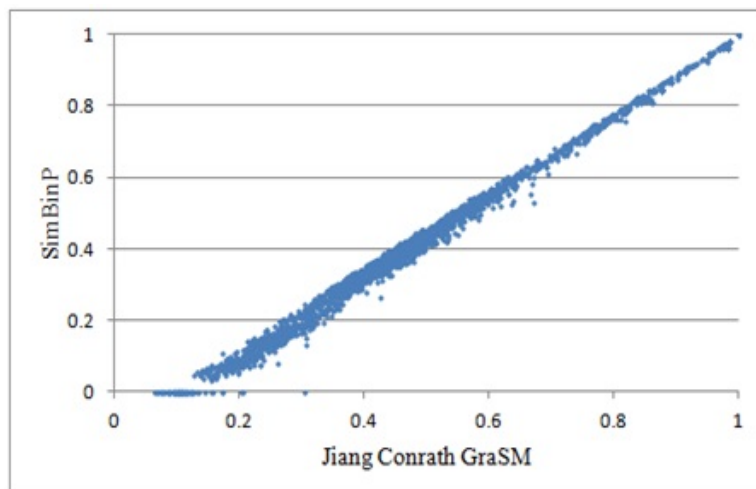


FIGURE 5.9: Scatter plot of SimBinP against Jiang Conrath GraSM.

Dataset 1. Figure 5.9 shows the scatter plot of SimBinP against Jiang Conrath GraSM. Although SimBinP is based on common terms only, the graph shows that, its results are very close to Jiang Conrath GraSM. SimExact is also based on common terms, and in addition, it makes use of Information Content to improve the similarity measurement. Therefore, SimBinP and SimExact can be used for protein query processing without significant loss of accuracy.

Figure 5.10 shows a bar graph of the time consumed by our tool in processing protein queries. Most of the queries are processed in less than 8 seconds. Figure 5.11 shows similar graph when ‘Compute full similarity’ is checked. The time consumed goes above 100 seconds, although the existing methods are applied to few short-listed proteins. If these methods are applied on all proteins in the database, the time consumed would jump to extreme high, not feasible for practical use. This answers research question 5 given in Section 1.1.

5.7 Experiment 5.6: Searching Protein Pairs

This experiment demonstrates some further uses of our protein querying mechanism. It can be used to identify protein pairs making interesting patterns. For example, we can fix a window of interest on the scatter plot of Figure 4.6, and find large number of pairs falling within that window. The graph of Figure 4.6 is

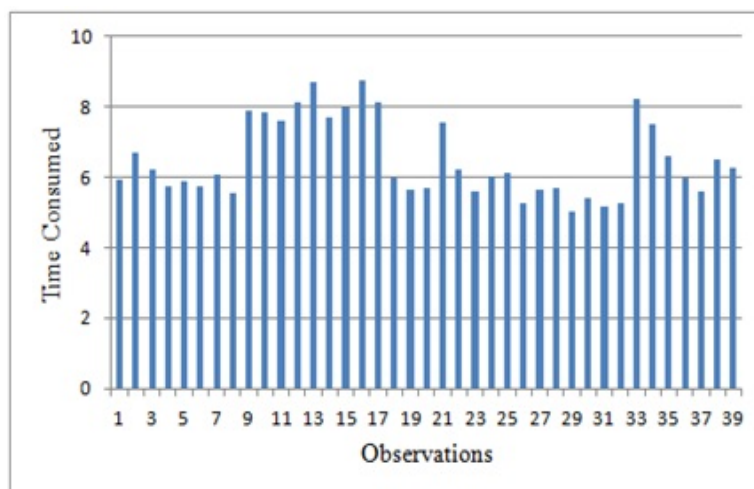


FIGURE 5.10: Graph of time consumed for protein query processing.

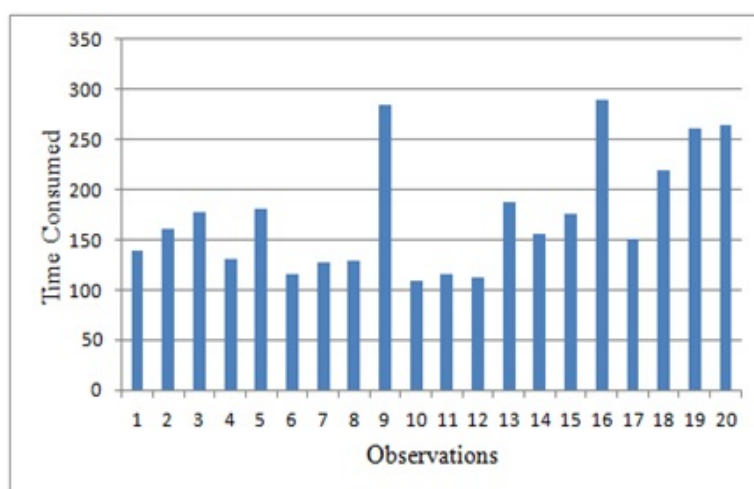


FIGURE 5.11: Time consumed for protein query processing with full computation.

prepared from a dataset of protein pairs, which may be biased. The measurement of similarities for large dataset requires intensive computation. Let us consider a window with semantic similarity between 0.8 and 1, and sequence similarity between 0 and 1 (full range) and apply our method to find large number of pairs within it. The results, containing 7,800 pairs, are plotted in Figure 5.12. The method can find more pairs, maybe all possible pairs within the window. These, and similar results, are useful in our analysis on behavior of proteins. For example they verify the results of Figure 4.4 and Figure 4.6, that protein pairs with high semantic similarity can have sequence similarity ranging from 0.1 to 1.

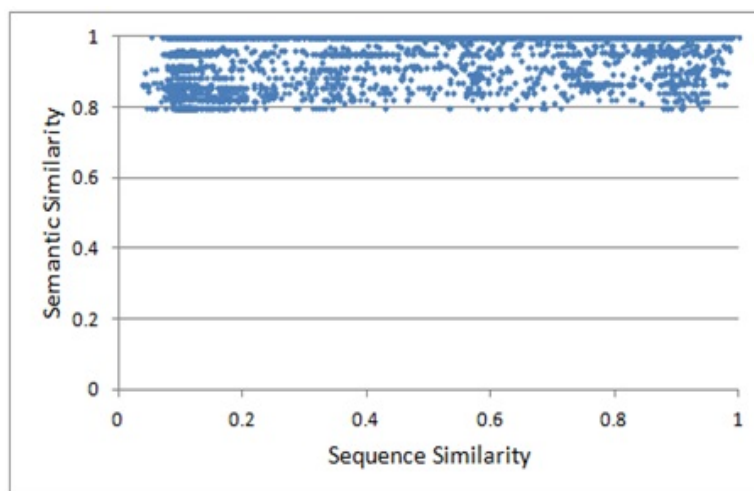


FIGURE 5.12: Scatter plot of semantic similarity against sequence similarity for searched pairs.

Another useful outcome of this method is to identify the pairs, in large number, with high disparity between sequence similarity and semantic similarity. These are the pairs having semantic similarity near 1 and sequence similarity near 0.1, meaning different sequences performing highly similar functions. Some of such pairs may help identify annotation errors. We identified such pairs in large number, Table 5.4 lists 10 of them. More such pairs are listed in Appendix F. Appendix E lists protein pairs with highest sequence-semantic conflict. This was not possible without SimBinP and SimExact.

Going further, a large portion of Figure 4.4 and Figure 4.6, with sequence similarity from 0.2 to 1 and semantic similarity from 0 to 0.6 is empty. The pairs in this region might be non-existent or rare. We used the method to search for pairs having semantic similarity ranging from 0.5 to 0.8 and full range of sequence similarity. We found that such pairs exist, although not frequently. Among the overall retrieved pairs, there are 2165 pairs inside a 0.2×0.2 square located at the upper right corner i.e., with sequence similarity and semantic similarity both between 0.8 and 1. There are 1179 pairs inside the square of same dimensions on left side of this square, and only 158 pairs inside the square below the corner one.

TABLE 5.4: Protein pairs having low sequence similarity and high semantic similarity.

Protein 1	Protein 2	Seq	Sem
P13043 Protein gvpD 1	Q8ZYR9 DNA repair and recombination protein RadA	0.07	1
P74503 KaiC-like protein 2	Q57702 DNA repair and recombination protein RadB	0.07	1
Q9HHT2 Protein gvpD 2	Q49593 DNA repair and recombination protein RadA	0.07	1
Q9WYK4 UPF0273 protein TM_0370	Q9HHT2 Protein gvpD 2	0.08	1
Q5B4H8 ATP-dependent DNA helicase II subunit 2	P0CO50 ATP-dependent DNA helicase II subunit 1	0.09	1
P08280 Protein RecA	Q58754 UPF0273 protein MJ1359	0.07	0.95
P03433 Polymerase acidic protein	P03431 RNA-directed RNA polymerase catalytic subunit	0.09	0.97
Q75D34 Autophagy-related protein 21	Q640T2 WD repeat domain phosphoinositide-interacting protein 3	0.08	0.95
Q7SG97 SVP1-like protein 2	Q75D34 Autophagy-related protein 21	0.09	0.95
Q8ZTQ5 UPF0273 protein PAE3143	Q89I84 Protein RecA	0.10	0.95

Chapter 6

Investigation and Discussion

We have found that semantic similarity has only weak correlation with sequence similarity, and that too is due to the common terms. This interesting behavior of proteins needs further investigation. We now attempt to find out the reasons for the weak correlation and for the pattern of sequence-semantic graphs. In Figure 4.4, the points near origin, and those near top right corner (1, 1) are most obvious. Any proteins picked randomly would most likely be unrelated both sequence wise and semantic wise, falling near origin. Highly similar sequences are expected to have similar functions, falling near point (1, 1). The points corresponding to low sequence similarity but high semantic similarity were also expected to some extent. Proteins having different sequences can have similar functions in some cases. Convergent evolution, for instance, may result in such proteins. There are no points corresponding to high sequence similarity but low semantic similarity, which was also expected.

Some aspects of the graphs, however, need attention. As protein pairs exceed a limit of 15% sequence similarity, they abruptly change behavior, and jump to very high semantic similarity, rather than gradually changing the behavior. In the right region, the increasing sequence similarity does not have effect on semantic similarity. Semantic similarity measurement of GO terms is an active research area in bioinformatics and semantic computing, and assumption of a good correlation

between sequence and semantic similarities has been used for evaluation of similarity measures. Therefore, absence of a strong correlation has significance. We refer to this as the weak correlation issue. But the most interesting aspect is that the term semantic similarity values above 0 and below 1 do not play significant role in protein sequence-semantic graph, as depicted in Figure 4.5 and Figure 4.8. One would not expect this. This is referred to as fractional similarity issue.

The reasons for the above facts are hypothesized to be one or more of the following:

1. Incorrect annotations
2. Incorrect, less precise or incomplete Gene Ontology structure
3. Weak method of semantic similarity measurement
4. Protein function, by nature, is independent of sequence, and a good correlation does not exist in nature.

For further investigation, we measure similarity between GO terms inferred from the sequence similarity of proteins that annotate the GO terms. The scatter plots of Figure 4.4 and Figure 4.5 depend on the selection of protein pairs. Bias in this dataset would lead to incorrect conclusions. We can pick GO terms with known semantic similarity, particularly sibling leaf terms, and measure SISS between them. Sibling leaf terms with SISS deviating too much from average would then be identified as problematic.

Sequence similarity was computed as percentage, as it was done in previous chapters. BLOSUM62 matrix was used for scoring. Alternative scoring matrices e.g. PAM matrices can be used for measuring sequence similarity. Future works may use probability (P-value) or expect (E-value) to represent sequence similarity.

6.1 Experiment 6.1: Investigation on Weak Correlation

In this experiment, several pairs of sibling leaf GO terms were picked randomly, and proteins annotating one term were compared with those annotating the other (sibling term), without taking average. The individual sequence similarity scores were saved in the database. This allowed us to observe the distribution of sequence similarity of proteins annotated to sibling GO terms. The same computation was done for pairs of distant (non-sibling, randomly picked) terms so that a comparison can be made. Lastly, a similar computation was performed, comparing a term with itself, i.e., proteins that annotate the same term being compared with one another. In all these computations, a protein was never compared with itself.

Figure 6.1, Figure 6.2 and Figure 6.3 show these distributions. It should be noted that a protein pair having some relationship with respect to the term considered, would have other relationships with respect to other terms. In Figure 6.3 for example, the proteins share one term, while they may annotate other terms, which may be distant ones. In other words, two proteins sharing one term have some semantic similarity, but may not be highly similar. Same reasoning is applicable to proteins annotating sibling terms. So a peak at low sequence similarity is quite expected in all three graphs. However, some considerable fraction of proteins would be expected at high sequence similarity range in Figure 6.2 and Figure 6.3. Graph of Figure 6.2 would be expected to be close to that of Figure 6.3, and far from that of Figure 6.1, because sibling GO terms have high semantic similarity. However, the graph of Figure 6.2 is very close to Figure 6.1. The average for distant terms is 6.6%, for sibling terms 7%, and for same term 21%. This suggests that only the terms common between proteins are significant in the context of this study. This supports the results obtained from Figure 4.5 and Figure 4.8 with more clarity and confidence.

The experiment does not involve a semantic similarity measure, so reason 3 should not be responsible for the results. We can rule out reason 2 and reason 3 since the

experiment involved sibling terms, whose semantic similarity can easily be assumed to be high as compared to distant terms. Reason 4 seems to be the main reason responsible for the pattern of Figure 4.4 and Figure 4.5 (weak correlation). The first three reasons may cause slight displacement of points along y-axis. However, it cannot be expected that the points appearing near upper-left corner, would move altogether to lower-left corner, even if annotations, ontology and similarity measure are perfect.

Our experiments suggest that protein function, by nature, is independent of sequence. This can be seen in the horizontal band of points in Figure 4.5. The points correspond to protein pairs mostly having high function similarity, but sequence similarity ranging from very low to very high. Gene duplication can result into such proteins. Due to evolution, the sequences have changed, while the function has remained the same. The pairs on the extreme left of this band correspond to distant relatives. In case of paralogs, the function would also be dissimilar, but such pairs may not be part of our dataset.

It is quite difficult to understand why the sibling terms have same effect as the distant terms. Reasons for ineffectiveness of close terms are hypothesized to be one or more of the following:

1. In Gene Ontology structure, close location of terms does not mean biological similarity
2. Annotations are not precise enough to make distinction between close GO terms
3. Proteins either perform same function or altogether different functions: performing similar functions has no significance.

The first hypothesized reason is GO structure. Closely located terms may not mean that they are biologically similar. Semantic similarity measures return high similarity values for close terms e.g. sibling ones, and low similarity for distant terms. The applications using these measures assume that closely located terms

are biologically similar, and so are the proteins associated with them. The second hypothesized reason is that annotations are not precise enough to map proteins to accurate terms. We took average of a large number of pairs, and it is unlikely that in all or most of cases, annotations were inaccurate. According to the third reason, the ineffectiveness of close terms is inherent in the proteins. As discussed earlier, when sequences get little similar, their semantic similarity jumps to very high. At this level, proteins have mostly common terms. The only region where proteins have uncommon terms corresponds to very low sequence similarity. This suggests that reason 3 is responsible for ineffectiveness of close terms. In presence of reason 3, it is difficult to assess reason 1 and 2.

SISS was computed between pairs of GO terms. In one experiment, each leaf term of molecular function was compared with itself. The purpose was to see if all terms behave uniformly or they have some diversity. Figure 6.4 shows bar graph, and Figure 6.5 shows frequency distribution of these SISS values. It can be observed that many terms have low SISS with themselves, meaning that each term is annotated by diverse sequences, but some have higher values, meaning annotation by similar sequences. The average SISS is 0.216.

Similar computation was repeated for non-leaf terms of molecular function to compare the results with those of leaf terms. The average value is 0.177. The lower average for the general terms was expected. However, this average is higher than that of the sibling terms, which was unexpected.

6.2 Discussion

For sequence similarity below 0.15, semantic similarity varies from very low to very high, independent of sequence similarity. When sequence similarity exceeds 0.15, Semantic similarity jumps to above 0.8. For sequence similarity ranging from 0.15 to 1, semantic similarity remains above 0.8. The sequence-semantic scatter plot mainly consists of a left vertical band and a top horizontal band of points. Semantic similarity between proteins is mainly influenced by terms

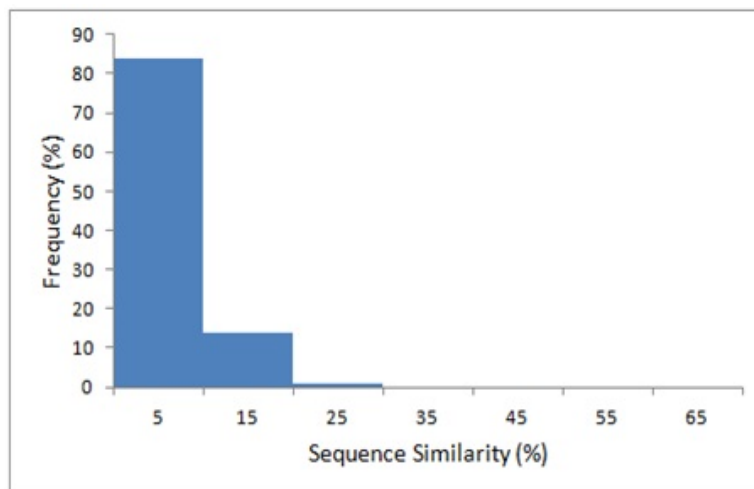


FIGURE 6.1: Distribution of sequence similarity scores between protein pairs belonging to distant GO terms.

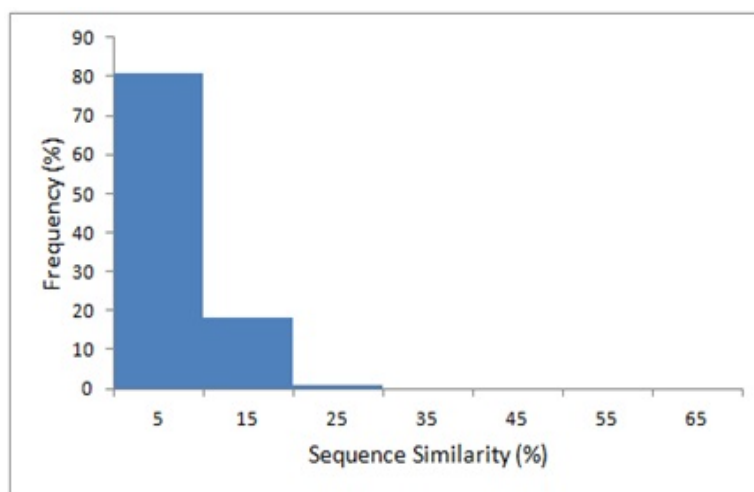


FIGURE 6.2: Distribution of sequence similarity scores between protein pairs belonging to sibling GO terms.

common between them, and other terms do not play a role in the context of correlation. Experiments were designed to explore such interesting behavior of protein pairs, and to try to find reasons of such behavior. We suspected that this could be attributed to weakness of GO structure or semantic similarity measures.

We have computed similarity between GO terms inferred from sequence similarity (SISS), which helps explore some behaviors with more clarity and confidence. This approach showed that proteins annotated to sibling GO terms have as different sequences as those annotated to distant terms. This verified that the correlation

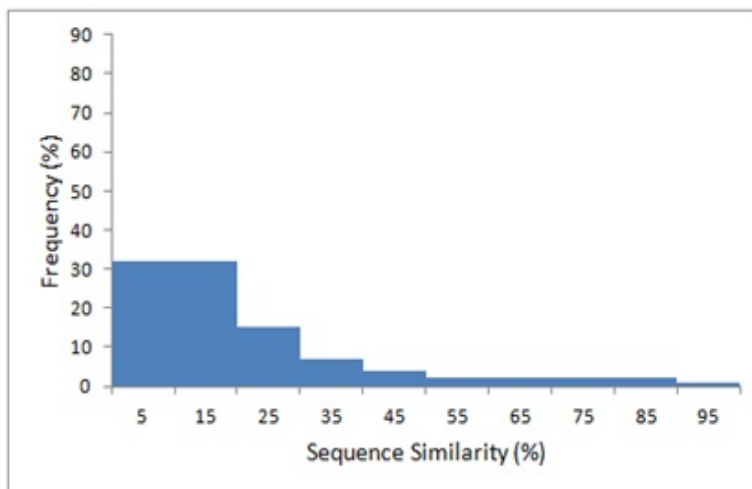


FIGURE 6.3: Distribution of sequence similarity scores between protein pairs sharing a common GO term.

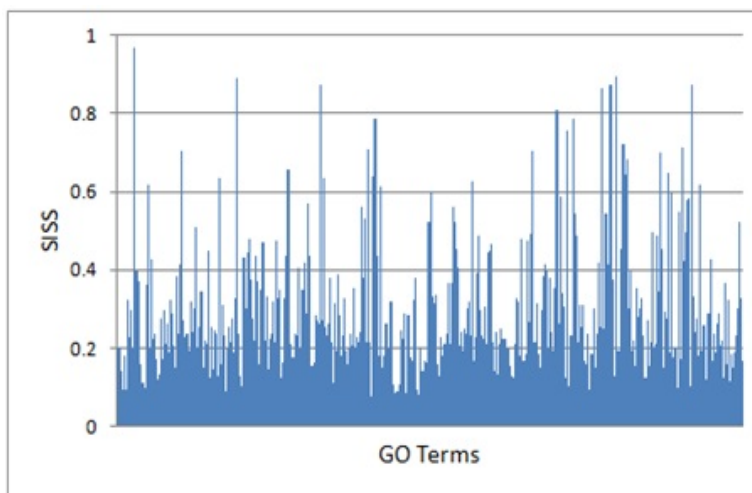


FIGURE 6.4: Bar Diagram of SISS of leaf GO terms with themselves.

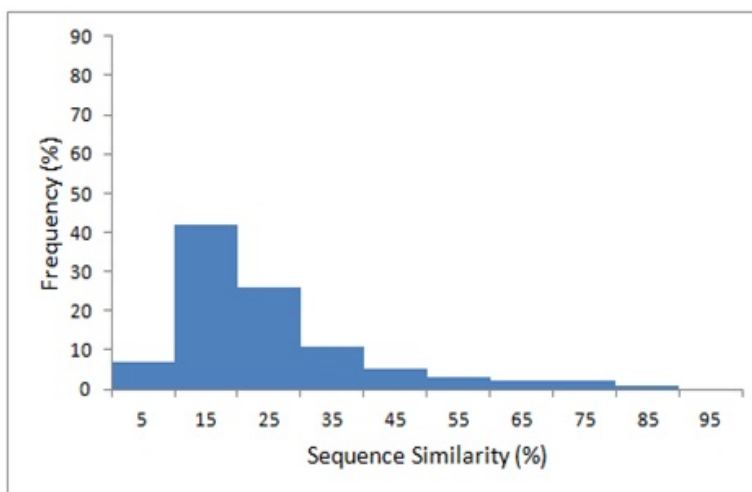


FIGURE 6.5: Distribution of SISS of leaf GO terms with themselves.

only depend upon the terms common between proteins, and semantically close terms do not play a role in it.

One thing is clear that errors in annotation, Gene Ontology structure, or semantic similarity measures are not the primary reasons for weak correlation. The main reason appears to be that such strong correlation does not exist in nature.

6.3 Research Questions Summary

Research questions given in Section 1.1 have been investigated and answered in various parts of the dissertation, supported by the results of experiments. RQ1 is answered in Experiments 4.1, 4.2 and 4.3, where the results show disagreement among the existing semantic similarity measures. RQ2 is answered in Experiment 4.4 and Experiment 4.6 where a weak correlation between semantic similarity and sequence similarity is shown. It is revisited in Experiment 5.6. Taking advantage of fast searching method, a large number of protein pairs have been shown, that have high semantic similarity, while sequence similarity varying from very low to very high. RQ3 is answered in Experiment 4.5. RQ4 is answered in best possible way in Experiments 5.1, 5.2, 5.3 and 5.4 that show the results of SemSim. RQ5 is answered in Experiment 5.5 that gives results of protein query tool. Chapter 6 mainly investigates RQ6 with the help of Experiment 6.1. Table 6.1 gives the relationship between research questions and experiments.

TABLE 6.1: Relationship between research questions and experiments.

Experiment	Research Question
Experiment 4.1	RQ1
Experiment 4.2	RQ1
Experiment 4.3	RQ1
Experiment 4.4	RQ2
Experiment 4.5	RQ3
Experiment 4.6	RQ2
Experiment 5.1	RQ4
Experiment 5.2	RQ4
Experiment 5.3	RQ4
Experiment 5.4	RQ4
Experiment 5.5	RQ5
Experiment 5.6	RQ2
Experiment 6.1	RQ6

Chapter 7

Conclusion

We have proposed calibration for semantic similarity measures. Experiments show that the similarity values are not consistent across various methods, and there is significant variability among them. Different similarity measures rank the GO terms similar to a reference term in different order. We have shown that there is no significant correlation between protein semantic similarity and sequence similarity. If two proteins have high sequence similarity with each other, they will most likely have high semantic similarity too. On the other hand, if two proteins have high semantic similarity, they may or may not have high sequence similarity. The points in the graphs corresponding to low sequence similarity but high semantic similarity show that state of the art tools will miss some proteins while answering a similarity query. The correlation coefficient depends only upon the number of common GO terms in protein pairs, and the semantic similarity measurement method does not influence it. Therefore, it may not be used for evaluation of semantic similarity measures.

We have shown that Resnik gives too low whereas Lin, Wang and AIC give too high similarity when comparing generalized terms, and the terms having many protein annotations. Addressing these issues, a novel method SemSim is proposed, and its results are reported. In the first step, similarity is computed without involving annotations. Then the adjusted similarity is computed using specificity and uniqueness of terms, giving them appropriate weights. SemSim addresses

the limitations of the existing methods, and conforms to GO taxonomy better than other methods. SemSim results are closer to domain expert's judgment as compared to other measures. SemSim A combines adjustment with similarity to give a single similarity value, which will be required by some applications. SemSim B returns two values: similarity and confidence level, and the decision of how to treat them is left to the client application.

SimExact is suitable for searching semantically similar proteins in situations where high speed is required. Although SimExact involves approximation, our experiments show that it produces correct results. Using SimExact, a protein query tool has been developed, and is available online. The tool finds proteins similar to a query protein in less than 8 seconds. Simexact can be used in combination with an existing similarity measure to short list candidate proteins on which the other measure is applied. A sequence-semantic conflict frequently exists in the proteins. SimExact is used to identify proteins that demonstrate such conflict.

We have attempted to find the reasons behind the weak correlation between semantic similarity and sequence similarity, which could be due to annotation errors, weakness of Gene Ontology or semantic similarity methods. We have shown that proteins annotated to sibling GO terms have as different sequences as those annotated to distant terms. Results and analysis suggest that a significant correlation does not exist in nature.

Future works should refine the protein query tool based on SimExact. Tool for pair wise semantic similarity based on SemSim should also be developed. Research is suggested on protein pairs with very high functional similarity and very low sequence similarity. What is the structural similarity of such interesting pairs should also be investigated.

Bibliography

- [1] T. G. O. Consortium, “Gene ontology tool for the unification of biology,” *Nat. Genet.*, vol. 25, pp. 25–29, 1964.
- [2] C. Pesquita, D. Faria, A. Falca, P. Lord, and C. F., “Semantic similarity in biomedical ontologies,” *PLoS Comput Biol*, pp. 25–29, 2009.
- [3] P. Resnik, “Using information content to evaluate semantic similarity in a taxonomy,” in *14th International Joint Conference on Artificial Intelligence*, 1995, pp. 57–60.
- [4] J. Jiang and D. Conrath, “Semantic similarity based on corpus statistics and lexical taxonomy,” in *10th International Conference on Research on Computational Linguistics*, 1997.
- [5] D. Lin, “An information-theoretic definition of similarity,” in *10th International Conference on Research on Computational Linguistics*, 1998.
- [6] C. Pesquita, D. Pessoa, D. Faria, and F. Couto, “Cessm: Collaborative evaluation of semantic similarity measures,” *JB2009: Challenges in Bioinformatics*, vol. 157, p. 190, 2009.
- [7] F. Couto and M. Silva, “Disjunctive shared information between ontology concepts: application to gene ontology,” *Journal of Biomedical Semantics*, pp. 2–5, 2011.
- [8] M. Alvarez and C. Yan, “A graph-based semantic similarity measure for the gene ontology,” *Journal of Bioinformatics and Computational Biology*, vol. 9, no. 6, pp. 681–695, 2011.

-
- [9] A. Lesk and C. Chothia, “How different amino acid sequences determine similar protein structures: the structure and evolutionary dynamics of the globins,” *Journal of molecular biology*, pp. 225–270, 1980.
- [10] C. Trad, Q. Fang, and I. Cosic, “Protein sequence comparison based on the wavelet transform approach,” *Protein Engineering*, pp. 193–203, 2002.
- [11] R. Gentleman, “Visualizing and distances using go,” <http://www.bioconductor.org/docs/vignettes.html>, 2005.
- [12] C. Pesquita, D. Faria, H. Bastos, A. Falco, and F. Couto, “Evaluating go-based semantic similarity measures,” in *10th Annual Bio-Ontologies Meeting*, 2007.
- [13] F. Couto, M. Silva, and P. Coutinho, “Measuring semantic similarity between gene ontology terms,” *Data & Knowledge Engineering*, vol. 61, pp. 25–29, 2007.
- [14] J. Wang, Z. Du, R. Payattakool, P. Yu, and C. Chen, “A new method to measure the semantic similarity of go terms,” *Bioinformatics*, vol. 23, pp. 1274–1281, 2007.
- [15] X. Song, L. Lin, P. Srimani, P. Yu, and J. Wang, “Measure the semantic similarity of go terms using aggregate information content,” *IEEE/ACM Trans. Comput. Biol. Bioinformatics*, vol. 11, no. 3, pp. 468–476, 2014.
- [16] J. Peng, Y. Wang, and J. Chen, “Towards integrative gene functional similarity measurement,” *BMC Bioinformatics*, vol. 15, no. 2, 2014.
- [17] G. Mazandu and N. Mulder, “Information content-based gene ontology semantic similarity approaches: Toward a unified framework theory,” *Hindawi Publishing Corporation BioMed Research*, vol. 2013, 2013.
- [18] G. Yu, F. Li, Y. Qin, X. Bo, Y. Wu, and S. Wang, “Gosemsim: an r package for measuring semantic similarity among go terms and gene products,” *Bioinformatics*, vol. 26, no. 7, pp. 976–978, 2010.

-
- [19] N. Seco, V. T., and J. Hayes, “An intrinsic information content metric for semantic similarity in wordnet,” in *ECAI 2004*, 2004.
- [20] H. Yang, T. Nepusz, and A. Paccanaro, “Improving go semantic similarity measures by exploring the ontology beneath the terms and modelling uncertainty,” *Bioinformatics*, vol. 28, no. 10, pp. 1383–1389, 2012.
- [21] S. e. a. Bien, “Bi-directional semantic similarity for gene ontology to optimize biological and clinical analyses,” *J Am Med Inform Assoc*, vol. 19, pp. 765–774, 2012.
- [22] X. Wu, E. Pang, K. Lin, and Z. Pei, “Improving the measurement of semantic similarity between gene ontology terms and gene products: Insights from an edge- and ic-based hybrid method,” *PLOS ONE*, vol. 8, no. 5, 2013.
- [23] F. Vafaei, D. Rosu, F. Broackes-Carter, and I. Jurisica, “Novel semantic similarity measure improves an integrative approach to predicting gene functional associations,” *BMC systems biology*, vol. 7, no. 1, p. 22, 2013.
- [24] S. E. et al., “Oss a semantic-based similarity measure for human druggable target proteins,” in *BIOTECHNO 2013*, 2013.
- [25] P. Lord, R. Stevens, A. Brass, and C. Goble, “Investigating semantic similarity measures across the gene ontology: the relationship between sequence and annotation,” *Bioinformatics*, vol. 25, no. 2, pp. 25–29, 2003.
- [26] B. Louie, R. Higdon, and E. Kolker, “A statistical model of protein sequence similarity and function similarity reveals overly-specific function predictions,” *PLoS One*, vol. 4, no. 10, p. e7546, 2009.
- [27] D. Lee, O. Redfern, and C. Orengo, “Predicting protein function from sequence and structure,” *Nature reviews. Molecular cell biology*, vol. 8, no. 12, p. 995, 2007.
- [28] T. Joshi and D. Xu, “Quantitative assessment of relationship between sequence similarity and function similarity,” *BMC genomics*, vol. 8, no. 1, p. 222, 2007.

-
- [29] F. J. Anscombe, “Graphs in statistical analysis,” *The American Statistician*, vol. 27, no. 1, pp. 17–21, 1973.
- [30] C. Pesquita, D. Faria, H. Bastos, A. E. Ferreira, A. O. Falcão, and F. M. Couto, “Metrics for go based protein semantic similarity: a systematic evaluation,” *BMC bioinformatics*, vol. 9, no. 5, p. S4, 2008.
- [31] J. L. Sevilla, V. Segura, A. Podhorski, E. Guruceaga, J. M. Mato, L. A. Martinez-Cruz, F. J. Corrales, and A. Rubio, “Correlation between gene expression and go semantic similarity,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, vol. 2, no. 4, pp. 330–338, 2005.
- [32] A. Schlicker, F. S. Domingues, J. Rahnenführer, and T. Lengauer, “A new measure for functional similarity of gene products based on gene ontology,” *BMC bioinformatics*, vol. 7, no. 1, p. 302, 2006.
- [33] Z. Lei and Y. Dai, “Assessing protein similarity with gene ontology and its use in subnuclear localization prediction,” *BMC bioinformatics*, vol. 7, no. 1, p. 491, 2006.
- [34] X. Guo, R. Liu, C. D. Shriver, H. Hu, and M. N. Liebman, “Assessing semantic similarity measures for the characterization of human regulatory pathways,” *Bioinformatics*, vol. 22, no. 8, pp. 967–973, 2006.
- [35] M. Mistry and P. Pavlidis, “Gene ontology term overlap as a measure of gene functional similarity,” *BMC bioinformatics*, vol. 9, no. 1, p. 327, 2008.
- [36] K. Thompson, Y. Wang, T. Madej, and S. Bryant, “Improving protein structure similarity searches using domain boundaries based on conserved sequence information,” *BMC Structural Biology*, pp. 9–33, 2009.

Appendices

A. Term pairs with high disagreement among different measures

term1	term2	resg	jcg	ling	GIC	wang	Aic	ss0	ss5
GO:0000012	GO:0006281	0.17	0.06	0.27	0.88	0.96	0.98	0.9	0.65
GO:0006283	GO:0006289	0.19	0.06	0.29	0.91	0.96	0.98	0.9	0.72
GO:0006281	GO:0006284	0.17	0.08	0.32	0.91	0.96	0.98	0.89	0.62
GO:0006281	GO:0006289	0.17	0.08	0.32	0.91	0.96	0.98	0.9	0.6
GO:0006289	GO:0006281	0.17	0.08	0.32	0.91	0.96	0.98	0.9	0.6
GO:0097680	GO:0006303	0.22	0.05	0.28	0.9	0.95	0.98	0.91	0.8
GO:0006298	GO:0006281	0.17	0.08	0.32	0.91	0.96	0.98	0.9	0.59
GO:0006281	GO:0006302	0.17	0.08	0.34	0.92	0.96	0.98	0.91	0.58
GO:0070371	GO:0000165	0.21	0.06	0.32	0.9	0.96	0.98	0.9	0.7
GO:0005388	GO:0086039	0.23	0.06	0.32	0.91	0.95	0.98	0.92	0.65
GO:0007254	GO:0051403	0.24	0.06	0.35	0.93	0.96	0.99	0.93	0.76
GO:0016486	GO:0034230	0.24	0.05	0.29	0.87	0.96	0.98	0.89	0.78
GO:0016486	GO:0034231	0.24	0.05	0.29	0.87	0.96	0.98	0.89	0.78
GO:0006493	GO:0016266	0.22	0.06	0.32	0.9	0.95	0.98	0.91	0.67
GO:0032455	GO:0016486	0.24	0.05	0.3	0.88	0.96	0.98	0.89	0.77
GO:0006290	GO:0006289	0.17	0.05	0.23	0.81	0.92	0.97	0.82	0.67
GO:0006290	GO:0006301	0.17	0.05	0.23	0.81	0.92	0.97	0.82	0.67
GO:0000733	GO:0006259	0.16	0.07	0.27	0.81	0.94	0.97	0.82	0.55
GO:0006631	GO:0001676	0.18	0.08	0.32	0.88	0.95	0.97	0.87	0.52
GO:0006290	GO:0006302	0.17	0.05	0.24	0.82	0.92	0.97	0.82	0.65
GO:0043969	GO:0016573	0.27	0.06	0.35	0.93	0.96	0.99	0.94	0.75

GO:0006342	GO:0030466	0.29	0.05	0.35	0.93	0.97	0.99	0.95	0.82
GO:0006631	GO:0033559	0.18	0.08	0.33	0.88	0.95	0.97	0.88	0.51
GO:0000729	GO:0006259	0.16	0.07	0.29	0.83	0.94	0.97	0.83	0.52
GO:0019369	GO:0019371	0.23	0.06	0.33	0.9	0.95	0.98	0.9	0.74
GO:0006493	GO:0006486	0.21	0.07	0.35	0.9	0.96	0.98	0.9	0.6
GO:0006486	GO:0006487	0.21	0.08	0.36	0.9	0.96	0.98	0.9	0.6
GO:0006342	GO:0030702	0.29	0.06	0.37	0.94	0.97	0.99	0.95	0.79
GO:0006284	GO:0006289	0.17	0.06	0.28	0.83	0.92	0.97	0.81	0.63
GO:0008380	GO:0000375	0.19	0.08	0.33	0.88	0.93	0.98	0.87	0.59
GO:0006284	GO:0006301	0.17	0.06	0.28	0.83	0.92	0.97	0.81	0.63
GO:0006289	GO:0006301	0.17	0.06	0.28	0.84	0.92	0.97	0.82	0.61
GO:0006396	GO:0008380	0.16	0.09	0.34	0.87	0.94	0.97	0.86	0.5
GO:0006342	GO:0006348	0.29	0.06	0.38	0.94	0.97	0.99	0.95	0.79
GO:0016070	GO:0010501	0.15	0.1	0.34	0.86	0.94	0.97	0.81	0.47
GO:0005789	GO:0030867	0.14	0.07	0.25	0.73	0.93	0.97	0.79	0.47
GO:0043629	GO:0043630	0.26	0.04	0.28	0.85	0.94	0.98	0.87	0.8
GO:0006310	GO:0006259	0.16	0.09	0.34	0.86	0.94	0.97	0.86	0.47
GO:0006284	GO:0006302	0.17	0.07	0.29	0.84	0.92	0.97	0.82	0.61
GO:0006289	GO:0006302	0.17	0.07	0.3	0.84	0.92	0.97	0.83	0.59
GO:0060389	GO:0006468	0.19	0.08	0.34	0.86	0.94	0.97	0.85	0.58
GO:0000183	GO:0006342	0.29	0.07	0.4	0.95	0.97	0.99	0.95	0.77
GO:0006281	GO:0006283	0.17	0.07	0.29	0.83	0.93	0.97	0.81	0.58
GO:0048478	GO:0043111	0.29	0.05	0.32	0.89	0.94	0.98	0.9	0.82
GO:0033540	GO:0006635	0.3	0.06	0.38	0.94	0.96	0.99	0.94	0.79
GO:0006379	GO:0090501	0.2	0.07	0.34	0.87	0.94	0.98	0.87	0.58
GO:0006470	GO:0035335	0.21	0.07	0.34	0.87	0.94	0.97	0.87	0.64
GO:0030702	GO:0030466	0.29	0.04	0.31	0.88	0.94	0.98	0.9	0.85
GO:0016573	GO:0043967	0.27	0.08	0.41	0.95	0.96	0.99	0.94	0.68
GO:0001682	GO:0008033	0.23	0.06	0.33	0.87	0.94	0.98	0.88	0.67
GO:0018108	GO:0007260	0.23	0.07	0.35	0.89	0.94	0.98	0.9	0.66
GO:0005789	GO:0033017	0.14	0.09	0.3	0.78	0.93	0.97	0.81	0.43

GO:0006348	GO:0030466	0.29	0.05	0.32	0.88	0.94	0.98	0.9	0.85
GO:0006919	GO:0008635	0.29	0.07	0.43	0.96	0.97	0.99	0.96	0.78
GO:0032777	GO:0035267	0.21	0.07	0.33	0.87	0.92	0.98	0.88	0.56
GO:0005671	GO:0000123	0.18	0.07	0.32	0.83	0.93	0.97	0.82	0.47
GO:0000183	GO:0030466	0.29	0.05	0.33	0.89	0.94	0.98	0.9	0.84
GO:0030702	GO:0006348	0.29	0.05	0.33	0.89	0.94	0.98	0.9	0.82
GO:0006584	GO:0042414	0.23	0.05	0.29	0.81	0.93	0.97	0.82	0.67
GO:0008203	GO:0016125	0.21	0.07	0.33	0.85	0.93	0.97	0.84	0.59
GO:0006104	GO:0006637	0.21	0.06	0.29	0.8	0.93	0.96	0.81	0.59
GO:0000329	GO:0005774	0.16	0.09	0.33	0.81	0.93	0.97	0.8	0.43
GO:0006584	GO:0042415	0.23	0.05	0.3	0.83	0.93	0.97	0.82	0.66
GO:0030702	GO:0000183	0.29	0.05	0.34	0.89	0.94	0.98	0.9	0.81
GO:0046777	GO:0006468	0.19	0.1	0.4	0.89	0.94	0.97	0.87	0.52
GO:0032872	GO:0046328	0.24	0.09	0.42	0.92	0.95	0.98	0.92	0.62
GO:0010008	GO:0031901	0.16	0.09	0.34	0.82	0.93	0.97	0.81	0.45
GO:0010008	GO:0031902	0.16	0.09	0.35	0.83	0.93	0.97	0.82	0.44
GO:0000183	GO:0006348	0.29	0.06	0.36	0.9	0.94	0.98	0.9	0.81
GO:0016567	GO:0006513	0.21	0.08	0.35	0.86	0.92	0.97	0.87	0.61
GO:0006476	GO:0070846	0.25	0.05	0.32	0.83	0.93	0.97	0.86	0.72
GO:0007409	GO:0021955	0.26	0.06	0.36	0.89	0.92	0.97	0.9	0.67
GO:0006486	GO:0042125	0.22	0.06	0.31	0.82	0.93	0.96	0.84	0.61
GO:0005774	GO:0005765	0.16	0.11	0.38	0.85	0.93	0.97	0.83	0.4
GO:0006476	GO:0090042	0.25	0.06	0.33	0.84	0.93	0.97	0.86	0.71
GO:0050730	GO:0001932	0.18	0.1	0.38	0.85	0.93	0.97	0.87	0.45
GO:0006464	GO:0018149	0.18	0.08	0.34	0.8	0.93	0.97	0.84	0.46
GO:0034230	GO:0032455	0.24	0.04	0.26	0.78	0.91	0.96	0.8	0.77
GO:0034231	GO:0032455	0.24	0.04	0.26	0.78	0.91	0.96	0.8	0.77
GO:0046459	GO:0000038	0.18	0.05	0.25	0.75	0.89	0.95	0.75	0.57

B. Pairs of general terms with high disagreement among different measures

term1	term2	resg	jcj	ling	GIC	wang	aic	ss0	ss5
-------	-------	------	-----	------	-----	------	-----	-----	-----

GO:0006302	GO:0006281	0.17	0.08	0.34	0.92	0.96	0.98	0.91	0.58
GO:0005388	GO:0086039	0.23	0.06	0.32	0.91	0.95	0.98	0.92	0.65
GO:0000733	GO:0006259	0.16	0.07	0.27	0.81	0.94	0.97	0.82	0.55
GO:0006631	GO:0001676	0.18	0.08	0.32	0.88	0.95	0.97	0.87	0.52
GO:0006631	GO:0033559	0.18	0.08	0.33	0.88	0.95	0.97	0.88	0.51
GO:0000729	GO:0006259	0.16	0.07	0.29	0.83	0.94	0.97	0.83	0.52
GO:0006493	GO:0006486	0.21	0.07	0.35	0.9	0.96	0.98	0.9	0.6
GO:0006486	GO:0006487	0.21	0.08	0.36	0.9	0.96	0.98	0.9	0.6
GO:0006396	GO:0008380	0.16	0.09	0.34	0.87	0.94	0.97	0.86	0.5
GO:0016070	GO:0010501	0.15	0.1	0.34	0.86	0.94	0.97	0.81	0.47
GO:0005789	GO:0030867	0.14	0.07	0.25	0.73	0.93	0.97	0.79	0.47
GO:0006259	GO:0006310	0.16	0.09	0.34	0.86	0.94	0.97	0.86	0.47
GO:0006310	GO:0006259	0.16	0.09	0.34	0.86	0.94	0.97	0.86	0.47
GO:0006379	GO:0090501	0.2	0.07	0.34	0.87	0.94	0.98	0.87	0.58
GO:0005789	GO:0033017	0.14	0.09	0.3	0.78	0.93	0.97	0.81	0.43
GO:0032777	GO:0035267	0.21	0.07	0.33	0.87	0.92	0.98	0.88	0.56
GO:0005671	GO:0000123	0.18	0.07	0.32	0.83	0.93	0.97	0.82	0.47
GO:0006104	GO:0006637	0.21	0.06	0.29	0.8	0.93	0.96	0.81	0.59
GO:0000329	GO:0005774	0.16	0.09	0.33	0.81	0.93	0.97	0.8	0.43
GO:0006468	GO:0046777	0.19	0.1	0.4	0.89	0.94	0.97	0.87	0.52
GO:0010008	GO:0031901	0.16	0.09	0.34	0.82	0.93	0.97	0.81	0.45
GO:0010008	GO:0031902	0.16	0.09	0.35	0.83	0.93	0.97	0.82	0.44
GO:0021955	GO:0007409	0.26	0.06	0.36	0.89	0.92	0.97	0.9	0.67
GO:0005774	GO:0005765	0.16	0.11	0.38	0.85	0.93	0.97	0.83	0.4
GO:0050730	GO:0001932	0.18	0.1	0.38	0.85	0.93	0.97	0.87	0.45
GO:0006464	GO:0018149	0.18	0.08	0.34	0.8	0.93	0.97	0.84	0.46
GO:0007220	GO:0044267	0.15	0.08	0.29	0.71	0.92	0.96	0.7	0.44
GO:0000725	GO:0006281	0.2	0.09	0.38	0.85	0.93	0.97	0.84	0.54
GO:0032784	GO:0006355	0.21	0.1	0.42	0.87	0.95	0.98	0.88	0.5
GO:0004714	GO:0005009	0.22	0.07	0.36	0.83	0.93	0.97	0.85	0.52
GO:0005010	GO:0004714	0.22	0.07	0.36	0.83	0.93	0.97	0.85	0.52

GO:0005006	GO:0004714	0.22	0.07	0.36	0.83	0.93	0.97	0.86	0.52
GO:0016266	GO:0006486	0.21	0.07	0.32	0.81	0.92	0.96	0.82	0.57
GO:0006475	GO:0006473	0.22	0.08	0.38	0.86	0.92	0.97	0.87	0.56
GO:0005790	GO:0005783	0.13	0.08	0.28	0.68	0.91	0.96	0.73	0.41
GO:0046459	GO:0001676	0.18	0.06	0.28	0.77	0.89	0.95	0.77	0.53
GO:0005783	GO:0005791	0.13	0.08	0.28	0.68	0.91	0.96	0.73	0.41
GO:0005783	GO:0016529	0.13	0.08	0.28	0.68	0.91	0.96	0.73	0.41
GO:0005007	GO:0004714	0.22	0.08	0.37	0.84	0.93	0.97	0.85	0.52
GO:0006464	GO:0043687	0.18	0.1	0.38	0.82	0.93	0.97	0.83	0.44
GO:0000717	GO:0006355	0.17	0.76	0.95	0.02	0.23	0.06	0.02	0

C. Pairs of specific terms with high disagreement among different measures

term1	term2	resg	jcj	ling	GIC	wang	Aic	ss0	ss5
GO:0043630	GO:0043629	0.26	0.04	0.28	0.85	0.94	0.98	0.87	0.8
GO:0048478	GO:0043111	0.29	0.05	0.32	0.89	0.94	0.98	0.9	0.82
GO:0030702	GO:0030466	0.29	0.04	0.31	0.88	0.94	0.98	0.9	0.85
GO:0030466	GO:0006348	0.29	0.05	0.32	0.88	0.94	0.98	0.9	0.85
GO:0000183	GO:0030466	0.29	0.05	0.33	0.89	0.94	0.98	0.9	0.84
GO:0030702	GO:0006348	0.29	0.05	0.33	0.89	0.94	0.98	0.9	0.82
GO:0030702	GO:0000183	0.29	0.05	0.34	0.89	0.94	0.98	0.9	0.81
GO:0034230	GO:0032455	0.24	0.04	0.26	0.78	0.91	0.96	0.8	0.77
GO:0034231	GO:0032455	0.24	0.04	0.26	0.78	0.91	0.96	0.8	0.77
GO:0030070	GO:0016486	0.24	0.05	0.29	0.78	0.93	0.96	0.81	0.71
GO:0007262	GO:0042991	0.33	0.05	0.38	0.88	0.93	0.97	0.89	0.81
GO:0032788	GO:0032789	0.16	0.04	0.17	0.6	0.83	0.94	0.62	0.6
GO:0002554	GO:0001820	0.3	0.05	0.34	0.83	0.92	0.96	0.84	0.76
GO:0030070	GO:0032455	0.24	0.04	0.26	0.71	0.88	0.94	0.74	0.7
GO:0007232	GO:0007231	0.36	0.06	0.43	0.85	0.94	0.97	0.86	0.77
GO:0006228	GO:0006241	0.4	0.06	0.45	0.89	0.94	0.97	0.9	0.82
GO:0016076	GO:0016077	0.26	0.04	0.28	0.74	0.86	0.94	0.76	0.71
GO:0042414	GO:0042415	0.23	0.05	0.27	0.7	0.86	0.94	0.7	0.61

GO:0019276	GO:0097359	0.21	0.04	0.23	0.65	0.82	0.93	0.67	0.62
GO:0046638	GO:0051138	0.45	0.07	0.52	0.92	0.95	0.98	0.92	0.8
GO:0031156	GO:0031285	0.34	0.05	0.38	0.8	0.9	0.93	0.81	0.7
GO:0018242	GO:0018243	0.25	0.05	0.28	0.7	0.83	0.93	0.72	0.66
GO:0006060	GO:0051164	0.21	0.04	0.24	0.6	0.8	0.91	0.62	0.55
GO:0046398	GO:0097359	0.2	0.04	0.23	0.62	0.77	0.9	0.63	0.58
GO:0034065	GO:0031297	0.23	0.04	0.26	0.62	0.83	0.88	0.64	0.58
GO:0000348	GO:0000354	0.3	0.04	0.32	0.65	0.83	0.92	0.69	0.67
GO:0006703	GO:0006705	0.31	0.05	0.37	0.68	0.86	0.91	0.68	0.62
GO:0021860	GO:0021954	0.32	0.05	0.38	0.7	0.83	0.93	0.7	0.62
GO:0014002	GO:0014044	0.31	0.05	0.35	0.65	0.82	0.92	0.65	0.61
GO:0048709	GO:0014037	0.29	0.05	0.33	0.61	0.8	0.91	0.62	0.57
GO:0032470	GO:0032468	0.41	0.06	0.45	0.76	0.83	0.94	0.79	0.74
GO:0031154	GO:0031152	0.35	0.05	0.39	0.61	0.85	0.9	0.6	0.56
GO:0045401	GO:0045425	0.31	0.04	0.32	0.53	0.8	0.88	0.57	0.56
GO:0042063	GO:0001570	0.2	0.05	0.24	0.46	0.72	0.86	0.45	0.41
GO:0000038	GO:0032788	0.17	0.04	0.2	0.48	0.71	0.83	0.47	0.42
GO:0000038	GO:0032789	0.17	0.04	0.2	0.48	0.71	0.83	0.47	0.42
GO:0015014	GO:0006075	0.29	0.05	0.33	0.59	0.77	0.88	0.62	0.56
GO:0050653	GO:0006075	0.29	0.05	0.33	0.59	0.77	0.88	0.62	0.56
GO:0045401	GO:0045086	0.31	0.05	0.35	0.55	0.8	0.88	0.58	0.54

D. Pairs of identical terms with high disagreement among different measures

term1	term2	resg	jcg	ling	GIC	wang	aic	ss5	conf
GO:0016787	GO:0016787	0.15	1	1	1	1	1	1	0.16
GO:0016740	GO:0016740	0.15	1	1	1	1	1	1	0.15
GO:0005515	GO:0005515	0.15	1	1	1	1	1	1	0.18
GO:0043234	GO:0043234	0.18	1	1	1	1	1	1	0.15
GO:0016020	GO:0016020	0.18	1	1	1	1	1	1	0.22
GO:0016772	GO:0016772	0.18	1	1	1	1	1	1	0.2
GO:0003676	GO:0003676	0.18	1	1	1	1	1	1	0.23

GO:0043231	GO:0043231	0.18	1	1	1	1	1	1	0.25
GO:0005737	GO:0005737	0.21	1	1	1	1	1	1	0.4
GO:0004871	GO:0004871	0.21	1	1	1	1	1	1	0.24
GO:0016818	GO:0016818	0.21	1	1	1	1	1	1	0.26
GO:0004872	GO:0004872	0.22	1	1	1	1	1	1	0.24
GO:0017111	GO:0017111	0.22	1	1	1	1	1	1	0.27
GO:0016021	GO:0016021	0.22	1	1	1	1	1	1	0.33
GO:0016491	GO:0016491	0.22	1	1	1	1	1	1	0.19
GO:0003677	GO:0003677	0.22	1	1	1	1	1	1	0.28
GO:0016301	GO:0016301	0.23	1	1	1	1	1	1	0.26
GO:0004888	GO:0004888	0.23	1	1	1	1	1	1	0.27
GO:0008233	GO:0008233	0.24	1	1	1	1	1	1	0.31
GO:0006810	GO:0006810	0.24	1	1	1	1	1	1	0.27
GO:0005634	GO:0005634	0.24	1	1	1	1	1	1	0.36
GO:0016773	GO:0016773	0.24	1	1	1	1	1	1	0.27
GO:0050896	GO:0050896	0.24	1	1	1	1	1	1	0.25
GO:0005886	GO:0005886	0.24	1	1	1	1	1	1	0.42
GO:0034641	GO:0034641	0.25	1	1	1	1	1	1	0.27
GO:0009058	GO:0009058	0.25	1	1	1	1	1	1	0.27
GO:0004672	GO:0004672	0.25	1	1	1	1	1	1	0.3
GO:0005102	GO:0005102	0.25	1	1	1	1	1	1	0.27
GO:0016887	GO:0016887	0.25	1	1	1	1	1	1	0.29
GO:0005887	GO:0005887	0.26	1	1	1	1	1	1	0.42
GO:0006139	GO:0006139	0.27	1	1	1	1	1	1	0.3
GO:0004518	GO:0004518	0.27	1	1	1	1	1	1	0.3
GO:0005829	GO:0005829	0.27	1	1	1	1	1	1	0.49
GO:0003723	GO:0003723	0.28	1	1	1	1	1	1	0.31
GO:0044281	GO:0044281	0.28	1	1	1	1	1	1	0.29
GO:0051246	GO:0051246	0.28	1	1	1	1	1	1	0.32

E. Sequence semantic conflict

p0	p1	p2	p1_sem	p2_sem	p1_seq	p2_seq
P49615	Q92643	P48609	0.94	0.16	0.4	0.78
P26439	Q12389	P49768	1	0.25	0.08	0.1
P26439	Q12389	Q64442	1	0.25	0.08	0.1
P26570	Q12389	Q9R101	1	0.29	0.08	0.1
P23578	P32639	Q96T60	0.87	0.17	0.06	0.1
P33431	Q08548	P36010	0.78	0.08	0.07	0.13
P40471	P36165	P36060	1	0.31	0.1	0.13
P25808	Q12389	Q12039	1	0.31	0.08	0.1
P23595	P32639	P36132	0.87	0.2	0.06	0.1
P23595	P32639	P53164	0.87	0.2	0.06	0.1
P23764	P32639	Q99LB2	0.87	0.2	0.06	0.11
P23595	P32639	P36132	0.87	0.2	0.06	0.1
P23764	P32639	Q99LB2	0.87	0.2	0.06	0.11
P23594	P32639	P53982	0.87	0.21	0.06	0.1
P23594	P32639	P36897	0.87	0.21	0.06	0.1
P23594	P32639	P28241	0.87	0.21	0.06	0.1
P28005	P41812	P42574	0.91	0.27	0.04	0.1
P23542	P32639	P53164	0.87	0.23	0.06	0.1
P23542	P32639	Q9WV60	0.87	0.23	0.06	0.1
P23542	P32639	P53164	0.87	0.23	0.06	0.1
P46964	P39007	Q00535	1	0.38	0.04	0.08
P23804	P32639	Q10743	0.87	0.25	0.06	0.11
P33609	Q08548	Q9JIL8	0.78	0.17	0.07	0.1
P53119	Q12464	P47912	0.78	0.17	0.05	0.09
P24529	Q07478	P41743	0.72	0.12	0.06	0.1
P32639	P23394	Q54CS9	0.87	0.27	0.06	0.09
P23394	P32639	P43405	0.87	0.27	0.06	0.09
P21954	P28834	P53598	0.87	0.27	0.09	0.11
P32639	P23394	Q54CS9	0.87	0.27	0.06	0.09
P23394	P32639	P67775	0.87	0.27	0.06	0.09

P23394	P32639	Q6PIE5	0.87	0.27	0.06	0.08
P39007	P41543	Q5IGR6	0.9	0.3	0.08	0.1
P32657	P23394	Q9URU2	0.87	0.27	0.06	0.09
P32657	P23394	Q9URU2	0.87	0.27	0.06	0.09
P97494	P70428	Q9W3K5	0.73	0.15	0.13	0.29
P41812	P28005	Q12389	0.83	0.25	0.04	0.09
P48025	P39518	P42683	0.82	0.24	0.12	0.17
P27619	P54132	Q12852	0.67	0.09	0.08	0.1
P32383	Q99P84	A6QLJ0	0.69	0.12	0.07	0.09
P32383	Q99P84	P22735	0.69	0.12	0.07	0.09
P41743	P48439	P48734	0.83	0.27	0.08	0.12
P40376	P36165	P34099	1	0.45	0.1	0.2
P22413	Q1RMT1	Q12852	0.69	0.14	0.07	0.09
P22413	Q1RMT1	P38993	0.69	0.14	0.07	0.09
P22303	Q1RMT1	Q9NDJ2	0.69	0.14	0.07	0.09
P22147	Q1RMT1	Q9NDJ2	0.69	0.14	0.07	0.09
P22147	Q1RMT1	P32657	0.69	0.14	0.07	0.1
P22147	Q1RMT1	Q1RKN3	0.69	0.14	0.07	0.09
P22303	Q1RMT1	Q1RKN3	0.69	0.14	0.07	0.09
P22303	Q1RMT1	P32657	0.69	0.14	0.07	0.1
P42356	Q9UBF8	P34756	0.77	0.23	0.07	0.09
P32561	Q99P84	Q06151	0.69	0.15	0.07	0.1
P22830	Q1RMT1	Q13011	0.69	0.18	0.07	0.1
P22830	Q1RMT1	Q06651	0.69	0.18	0.07	0.11
P41543	P48439	P46985	0.83	0.33	0.08	0.11
P27695	P54132	Q4PIR3	0.67	0.17	0.08	0.1
P22735	Q1RMT1	P22543	0.69	0.19	0.07	0.1
P34575	P38859	P50440	0.67	0.17	0.08	0.11
P67775	Q04592	P23595	0.75	0.25	0.2	0.63
P22735	Q1RMT1	Q9Y6K1	0.69	0.19	0.07	0.1
P27695	P54132	Q9USN7	0.67	0.17	0.08	0.1

P22735	Q1RMT1	Q9Y6K1	0.69	0.19	0.07	0.1
P38827	P36120	Q09811	0.67	0.18	0.08	0.1
P56523	Q54QQ1	Q9VH48	0.61	0.13	0.05	0.08
P38295	P36051	P41543	0.67	0.19	0.07	0.11
P22057	P29703	P53223	0.62	0.15	0.09	0.11
P22887	Q1RMT1	Q03114	0.69	0.22	0.07	0.11
P22887	Q1RMT1	P51166	0.69	0.22	0.07	0.08
P23196	Q1RMT1	P29466	0.69	0.22	0.07	0.09
P32603	Q9UEF7	Q9Y646	0.62	0.15	0.07	0.1
P32603	Q9UEF7	Q8VEB4	0.62	0.15	0.07	0.1
P22694	Q1RMT1	P36582	0.69	0.23	0.07	0.1
P22543	Q1RMT1	P36582	0.69	0.23	0.07	0.1
P23219	Q1RMT1	P25045	0.69	0.23	0.07	0.09
P22694	Q1RMT1	P54199	0.69	0.23	0.07	0.1
P22543	Q1RMT1	P54199	0.69	0.23	0.07	0.1
P22694	Q1RMT1	P36582	0.69	0.23	0.07	0.1
P23219	Q1RMT1	Q64FW2	0.69	0.23	0.07	0.09
P23219	Q1RMT1	Q8VHE9	0.69	0.23	0.07	0.09

F. Protein pairs with high semantic and low sequence similarity (maximum difference)

p0	p1	sem	seq
P25808	P32892	1	0.1
P38112	Q06218	1	0.1
P32892	P25808	1	0.1
P36120	Q12389	1	0.1
P40308	P36165	1	0.1
P38112	P53734	1	0.1
P25808	P36120	1	0.1
P36120	P25808	1	0.1
P36120	P38112	1	0.11

P38112	P36120	1	0.11
P36120	P53734	1	0.11
P32892	Q06218	1	0.11
P25808	Q06218	1	0.11
P47124	P50108	1	0.11
P32892	P36120	1	0.11
P36120	P32892	1	0.11
P36120	Q06218	1	0.11
P34163	Q07804	1	0.12
P32892	P53734	1	0.12
P25808	P38112	1	0.12
P38112	P25808	1	0.12
P32892	Q12389	1	0.12
P28005	P41812	0.91	0.04
P32892	P38112	1	0.14
P38112	P32892	1	0.14
P36120	P38719	0.92	0.08
P90648	P42527	0.93	0.1
P53734	Q06218	0.93	0.1
P53734	Q12389	0.93	0.1
P53734	P38112	0.93	0.1
P38112	P38719	0.92	0.1
P53734	P25808	0.93	0.1
P39007	P41543	0.9	0.08
P53734	P36120	0.93	0.11
P53734	P32892	0.93	0.12
P23394	P32639	0.87	0.06
P32639	P23394	0.87	0.06
P23394	Q9BUQ8	0.93	0.13
P46985	P50108	0.9	0.1
P47124	P46985	0.89	0.09

P32639	Q9BUQ8	0.87	0.07
P28005	P38786	0.91	0.11
P56951	Q60524	1	0.21
P41812	P28005	0.83	0.04
P28241	P28834	1	0.21
P34218	P40963	0.87	0.08
P25045	P40970	0.94	0.18
P40970	P25045	0.94	0.18
P53734	P38719	0.86	0.1
P70428	Q16394	0.88	0.13
P39007	P46964	0.8	0.05
P41543	P48439	0.83	0.08
P29476	P29475	0.92	0.17
P40559	Q12271	0.86	0.12
P41543	P33767	0.83	0.1
P28834	P28241	0.94	0.21
P38929	Q9R0K7	0.86	0.13
P39007	P48439	0.8	0.07
P53115	Q12464	0.78	0.05
P39518	P47912	0.82	0.09
P32639	P53131	0.8	0.07
P40559	P50942	0.86	0.14
P39104	P42356	0.8	0.08
P32639	P24384	0.8	0.08
P39518	P30624	0.82	0.1
P23394	P24384	0.8	0.08
P38929	P98194	0.81	0.1
P38929	Q5R5K5	0.81	0.1
P38929	Q64566	0.81	0.1
P38929	P57709	0.81	0.1
P23394	P53131	0.8	0.09

P46985	P47124	0.8	0.09
P33767	P41543	0.83	0.13
P33333	Q08548	0.78	0.07
P38929	Q03669	0.81	0.11
P47912	P39518	0.82	0.12
P42356	Q9UBF8	0.77	0.07
P41812	P38786	0.75	0.05
P53668	A2ASS6	0.75	0.06
P48609	Q00534	0.85	0.16
P53223	Q9JMF7	0.9	0.21
P41410	Q09811	0.77	0.09
P48609	P50582	0.77	0.09
P68181	A2ASS6	0.71	0.04
P53668	P24583	0.75	0.08
P36582	P36583	0.9	0.23
P24384	Q07478	0.72	0.06
P41543	P39007	0.75	0.09
P53668	Q02156	0.75	0.09
P41543	P46977	0.75	0.09
P33767	P48439	0.75	0.09
P38929	Q64578	0.76	0.11
P53668	P41743	0.75	0.09
P53115	Q60HG1	0.72	0.07
P53115	Q5RA62	0.72	0.07
P38929	Q92105	0.76	0.11
P53115	Q1RMT1	0.72	0.07
P24384	P53131	0.78	0.13
P40963	P34218	0.72	0.08
P51652	Q9Z0X1	0.71	0.08
P33767	P39656	0.75	0.11
P24384	Q9BUQ8	0.72	0.09

P41903	P58137	0.78	0.15
P58137	P41903	0.78	0.15
P42881	Q9H3H5	0.8	0.17
P42881	Q5EA65	0.8	0.17
P36582	P41743	0.75	0.12
P53355	P28482	0.69	0.06
P87231	Q04149	0.74	0.11
P42881	P24140	0.8	0.17
P42527	P90648	0.72	0.1
P68181	P49841	0.71	0.09
P42881	Q9JK82	0.7	0.08
P53355	Q02156	0.69	0.07
P42881	Q5IGR7	0.7	0.08
P53096	Q9UBN7	0.69	0.07
P36165	Q12043	0.85	0.23
P41410	P53115	0.69	0.07
P32383	Q99P84	0.69	0.07
P42881	Q5IGR8	0.7	0.08
P42881	Q5IGR6	0.7	0.08
Q01887	Q01279	0.72	0.11
