**CS4220 Project**

Member:

Vu Phuc Tho

Hoang Quang Minh – A0099943J

**An aggregate scoring scheme to integrate pathway databases**

**I. Introduction**

A biological pathway is a series of biochemical reactions, which can be spontaneous or driven by catalytic activities of enzymes. Pathway data offer important information about how genes and proteins interact with each other within living organisms, which in turn allow us to analyze, hypothesize and make predictions regarding observable biological phenomena. Motivated by the above mentioned fact (and facilitated with advanced experimental technologies), many databases have been established to record and amass new pathway data of different species (e.g. KEGG, Reactome, Wikipathways, etc.) in support of scientific researches. Over the years, these pathway databases have significantly grown in size and coverage. However, as we find a surge in the volume of data, we also find ourselves facing a new challenge: full exploitation of pathway databases is often hindered due to their heterogeneity (i.e the inconsistency in labelling, incompatible data formats, non-overlapping pathways, etc.). In this project, we explore an approach to overcome the difficulty of unifying heterogeneous pathway databases.

**II. Method**

1. Problem definition and notations:

We formally define the simplified problem of integrating pathway databases as followed:

Let D1 = {p1,p2, … pn) and D2 = {q1,q2, … qm) denote two pathway databases with size n and m respectively, where pi is the ith pathway appearing in D1 and qj is the jth pathway appearing in D2. Let us also define score(i,j) as the compatibility score (which is fully described below in section 3) between pathway pi and pathway qj (1 < i < n , 1 < j < m). The objective of the integration problem, therefore, is for each pathway pi in D1, find a pathway qj in D2 such that score(i,j) is maximized.

1. Extraction and normalization of pathway data:

For the project, we choose three sources of data: KEGG, Wikipathways and IntPathway, and extract pathway data of Saccharomyces cerevisiae in the three databases. Since data format of each database is different from others, further normalizations are required to unify the format before integration process.

Retrieving data from Wikipathways and Intpathway is straight-forward, since both databases allow user to directly download their data. On the other hand, KEGG does not provide the same functionality as Wikipathways and Intpathway, but requires user to use their own API or download data in KGML format. However, in our program, we scrap data directly from HTML files of GenomeNet, which contains the full and completed pathway data of KEGG. Since the data format is more clear and easier to extract than KGML, we achieved the same good result with less processing time.

Normalization of data is required for Wikipathways, since they save genes not by their systematic names, but by gene id in various data sources, including BioCyc, NCBI, and Uniprot. For each gene id in pathway, we need to map it to the gene systematic names by scraping data from the databases shown above. After the normalization, it is ensured that the genes are stored in same format for all three databases in order to integrate efficiently.

1. A linear weighted scoring scheme

We define score(i,j) as a linear weighted sum of k metrics, i.e:

score(i,j) = w0 + w1M1 + w2M2 + … + wkMk

where wi denotes the weight of normalized metric Mi. In this project we consider three popular metrics to measure the compatibility between two pathways: (i) Gene agreement, (ii) Gene-pair agreement and (iii) String comparison of pathway names (Longest common substring)

3.1 Gene agreement:

Let us define Gi as the set of genes present in pi and G’jas the set of genes present in qj. Let us further define NG as the total number of unique genes in both databases. The chance that 2 sets of size |Gi| and |G’j| selected at random in a pool of size NG shares r elements is given by:

This probability coincides with the p.m.f of the Hypergeometric distribution with population size NG, |Gi| success states and |G’j| draws. Let X ~ Hypergeometric(NG, |Gi| , |G’j|) and r = |Gi ∩ G’j|, we proceed to choose our metric M1 = Pr(X < r) (p-value at X = r).

3.2 Gene-pair agreement

A gene-pair is a tuple consisting of 2 closely related genes appearing in the same pathway. Similarly to section 3.1, let us define GPi and GP’j as the sets of gene-pairs present in pi and qj respectively; NP is the total number of unique gene-pairs in both databases. In doing so, we likewise obtain a hypergeometric distribution whose p-value is our metric M2.

3.3 Longest common substring (LCS)

Although naming convention varies across databases, we believe important keywords are conserved throughout all references of a pathway, forming identical substrings. The longer the length of two sequences’ LCS, therefore, implies a higher probability of them representing the same pathway. This justifies the choice of our third metric M3 as length of the LCS of pi and qj’s referral names.

1. Weight-training using Genetic Algorithm

4.1 Background on Genetic Algorithm (GA)

Genetic algorithm is a heuristic search algorithm, inspired by the natural selection process. The genetic algorithm keeps a population of individuals with different chromosomes (weight sets) and repeatedly modified them. In each iteration, the genetic algorithm randomly chooses individuals as parents and use them to produce children for next generation. Individuals with better performances have more opportunity to be chosen as parent, hence the successive generation will inherit good characteristics of their parents and ‘evolve’ towards the optimal solution.

For each iteration, the genetic algorithm produces next generation through three operations:

* Selection: select the individuals as parents for the next generation
* Crossover: produce offsprings from each pair of parents selected from previous step
* Mutation: apply random changes to a portion of offsprings.

4.2 Weight-training:

We treat each tuple of weights **W**i = {w0,w1,w2,w3} as an individual and generate a population pool from random sampling. After every round of GA, the fitness of each individual is evaluated using a fitness function given by f(D1,D2,**W**i) = stop / sremain where stop is the average of top 1% normalized scores and sremain is the average of the remaining normalized scores. Individuals with high fitness scores crossover to generate offsprings which make up the population of next round of GA. This process is repeated until the the average fitness scores of the population reaches a pre-determined threshold.

1. **Justifications:**
2. Metrics and scoring scheme

Gene agreement and gene-pair agreement are direct measurements of compatibility between pathways. Ideally, two versions of the same pathway stored in different databases should have high gene and gene-pair overlapping percentage (which leads to high p-value). In reality, however, due to incompleteness of databases, gene overlapping percentage and gene-pair overlapping percentage are often much smaller than expected (in extreme cases, databases store disjoint segments of a pathway), giving rise to false negatives during prediction as a consequence.

LCS has been shown empirically to produce better result than gene agreement and gene-pair agreement [*Zhou et al., IntPath--an integrated pathway gene relationship database for model organisms and important pathogens*]. However, during our research, we have discovered cases whereby two different pathways can have unexpectedly high LCS score. For example, let us consider two pathways: “**T**r**y**pt**o**pha**n** **biosynthesis**” and “**Ty**r**o**si**n**e **biosynthesis**” which essentially are different yet induce a very high LCS score.

In conclusion, each metric has its own advantages and disadvantages. By introducing an aggregated scoring scheme, we hope to combine the strengths of all metrics and filter out false positives which are hard to discern using any single metric.

1. Weight-training

Due to the total absence of training data, supervised learning is not a feasible option. To work around this constraint, we propose using the popular heuristic method Genetic Algorithm to learn a suitable weight-set for our scoring scheme. To justify the choice of Genetic Algorithm, we try to justify our choice of the fitness function described above (section II.4.2).

Assuming we already possess a weight set **W**, let S denote the set of compatibility scores produced when applying **W** on databases D1 and D2, S = {score(i,j)}ij . We know that only a small portion of scores in S should be regarded as “good” matches whereas the rest are likely to represent non-correlated pairs of pathways. Explicitly, let us assume that for each pathway p in D1 has at most one “good” match and vice-versa for each pathway q in D2, then the proportion between “good” and “bad” matches are (at most, since we must account for repeated match) . Intuitively, a good set of weights will be able to clearly separate “good” and “bad” matches. Following this intuition, we believe that, given a good set of weights, the average normalized score of top matches should be much higher than the average normalized score of remaining matches. This justifies the choice of our fitness function as ratio between stop and sremain. Note that we used normalized scores to compute stop and sremain to prevent GA from scaling up weights infinitely. The cut-off for top matches is empirically chosen to be one percent of the score population.

1. **Validation:**

To verify the effectiveness of our method, we proceed to compare the consistency of predictions made on Wikipathways and KEGG databases using the following methods: (i) LCS, (ii) Gene agreement, (iii) Gene-pair agreement and (iv) aggregate scoring method, with LCS as the baseline for comparison. Consistency is measured by calculating percentage overlap between top scoring matches of each method above and top scoring matches of LCS.

To further compare the effectiveness of our method and LCS, we also compare the average gene-pair agreement score and average gene agreement score of the unique top matches (non-overlapping matches) made by our method and the unique top matches made by LCS.

1. **Results:**

1. Weight set obtained by GA training:

w0=0.00901413316794486,

w1=0.31051462633709526,

w2=0.2167698714962747,

w3=0.7032772432533334.

Where w0 denotes offset constant, w1 denotes weight of gene-agreement score, w­2 denotes weight of gene-pair agreement score and w3 denotes weight of LCS score.

2. Consistency:

Comparing top scoring 1% matches (121) generated using 4 methods above with LCS as the baseline, we obtain the following results:

|  |  |  |
| --- | --- | --- |
|  | Number of overlapping matches (with LCS) | Percentage overlap (with LCS) (%) |
| Gene agreement | 1 | 0.00826 |
| Gene-pair agreement | 1 | 0.00826 |
| Aggregate scoring | 81 | 66.9 |
| LCS (baseline) | 121 | 100.0 |

Out of top 121 matches, LCS and Aggregate scoring each singled out 40 matches. The following graph summarizes the gene-agreement score and gene-pair agreement score of these 80 matches.

*Figure 1. Gene agreement score and gene pair agreement score distribution of 80 non-overlapping top matches ( horizontal axis – gene pair agreement score, vertical axis – gene agreement score, orange – matches made by LCS, blue – matches made by aggregate scoring method)*

1. **Analysis and Discussion:**

1. Result analysis

First of all, we notice that the weight set **W** obtained using GA training showed an approximate ratio of 1:1:3 between weights of gene agreement score, gene pair agreement score and LCS score. This is consistent with the fact that LCS yields better performance than gene agreement and gene pair agreement, whereas the latter are of equal importance. The convergence of “evolution” of **W** and our prior belief, to some extent, shows the effectiveness of our fitness function and as a result, reinforces the soundness of our hypothesis on qualities of a good weight set.

The above results also show a relatively high agreement between top matches made by LCS and top matches made by our method. Approximately two thirds of high scoring matches appear in LCS matching also appear in aggregate score matching. Since there is no ground truth for this matching problem, it is difficult to validate the accuracy of the remaining one third top matches of each method. However, if we take a further look at the distribution of gene and gene-pair score of the remaining matches, we can observe a very clear separation of the two groups: matches made by LCS have scores ranging from very low to average whereas most matches made by the aggregate method have very high scores (except 1 point with 0 gene-pair agreement score).

To assess the relative accuracy of two methods, we consider the following four possibilities.

1: The non-overlapping matches made by LCS are true positives. They have low-average GA scores and GPA scores because each database records a different segment of a same pathway.

2: The non-overlapping matches made by LCS are false positives for the reason explained in section III.1

3: The non-overlapping matches made by aggregate score method are true positives missed out by LCS because pathway names are relatively different.

4: The non-overlapping matches made by aggregate score method are false positives because in reality, certain pathways contain a high percentage of overlapping genes and gene-pairs.

In the context of this experiment, we worked on yeast genome which contain approximately 1500 genes. In a small genome, the probability for two pathways having similar content of genes and gene-pairs is relatively high. Hence, there is a higher chance for possibility 4 to happen compare possibility 3. It is uncertain whether 1 or 2 is more likely than the other. Overall, we conclude that LCS’s predictions are likely to be more accurate than that of our method with respect to this dataset. However, we do hope that in more complete databases, where the probability of possibility 1 is lower than probability of possibility 2, our method can single out some false negatives made by LCS.

2. Areas for improvement:

In our implementation of this method, we have taken the raw LCS score as our metric M3. However, we realize that raw LCS score is not a good measurement of similarity between pathway names because the upper bound for LCS score is the minimum length of the two pathway names (i.e. LCS < min(length(s1), length(s2))). This means matches between pathways with long names generally will produce higher scores than matches between pathways with short names, even if the latter pathways are very similar. To improve this, we could have used the alignment ratio, given by raw score divided by average length of two pathway names, as our metric. With this change, we expect the method will produce less false negatives.

1. **Contributions:**

Vu Phuc Tho: data extraction, data pre-processing, implementation of LCS and Genetic Algorithm training, result post-processing.

Hoang Quang Minh: result analysis, implementation of gene agreement scoring and gene-pair agreement scoring, final report.