

Introduction

When determining 3D structures of proteins, approaches incorporating software for prediction are often lower in cost and higher in throughput than conventional experimental methods. Currently, the most practical computational approach is the threading method, which matches the target sequence against each known protein structure in a database (e.g. PDB) to identify similar folds. RAPTOR is a threading package that ranked no. 1 non-meta server in CAFASP3. Using a new linear programming approach, it has consistently demonstrated high quality predictions and is used in structure modeling, data mining and conservation discovery. This poster presents an overview of a recent industry application where a pipeline system incorporates RAPTOR and a protein/DNA homology search program PatternHunter for high-throughput functional annotation generation.

Functional Annotation

Two important premises of protein structure prediction are:

1. The number of unique structures in nature is fairly small compared to the theoretically infinitely many permutations of amino acid sequences. It follows that sequentially dissimilar proteins could share similar structures.
2. The structure of a protein determines its function.

It therefore seems a practical and feasible idea to infer the function of an unknown target protein from that of structurally similar template proteins via the threading technique. Furthermore, it is worth noting that functional annotations in structure databases are not always the most detailed or up-to-date compared to a DNA sequence database. It is hence beneficial to complement the threading technique by running Translated PatternHunter, our protein/DNA homology search program, against a DNA sequence database such as GenBank. This is the basic idea of the RAPTOR/PatternHunter pipeline.

A critical component in a prediction system is to assess the quality of results. We measure the confidence by the following metrics:

- **Support Vector Machines (SVM)** regression for RAPTOR. In short, SVM is a type of machine learning technique that extracts "features" from the alignments - attributes that describe the alignments. A threading pair is treated as a positive pattern only if they are of at least fold-level similarity. Over 60,000 threading pairs are employed to train RAPTOR's SVM model. By using this model, 5% more targets can be recognized than the traditional Z-score. RAPTOR normalizes the SVM scores to ensure they are comparable. Such normalized SVM scores are called Z-scores, which are not the same as the traditional Z-scores.
- **Expected Value (E-value)** of an alignment for PatternHunter is the expected number of alignments that (1) have the same raw score as the output alignment; (2) are generated purely because of randomness (instead of homology). The smaller the E-value is, the more significant this alignment is, and the more likely this alignment is a true homolog.

The next step would be to compare and evaluate these two scores to select the better prediction. Results from RAPTOR and PatternHunter (PH) are selected by the following algorithm (z-score (i.e. normalized SVM) from RAPTOR, E-value from PatternHunter)

```

if z-score >= 4.0
    choose RAPTOR result
else if E-value <= 0.1
    choose PH result
else if z-score >= 2.0
    choose RAPTOR result
else
    output nothing
    
```

The two scores from the programs are not directly combined into a weighted score because the nature of the two programs are fundamentally disparate -- PatternHunter is a homology search program involving only sequence-related data, whereas RAPTOR builds on top of both sequential and structural information.

We then filter out results by cut-off values that concern only the individual programs. For RAPTOR, a two-tier filter is justified as follows: if a threading pair has a Z-score >= 4.0, then there is a very good chance that this pair is at least similar at the superfamily level. If a threading pair is unrelated, then it is highly likely that its Z-score <= 2.0. For matches with a z-score between 2.0 and 4.0, the result is contestable with that from PatternHunter. For PatternHunter, the value of 0.1 is chosen based on empirical data. These parameters can be trained over a large test set.

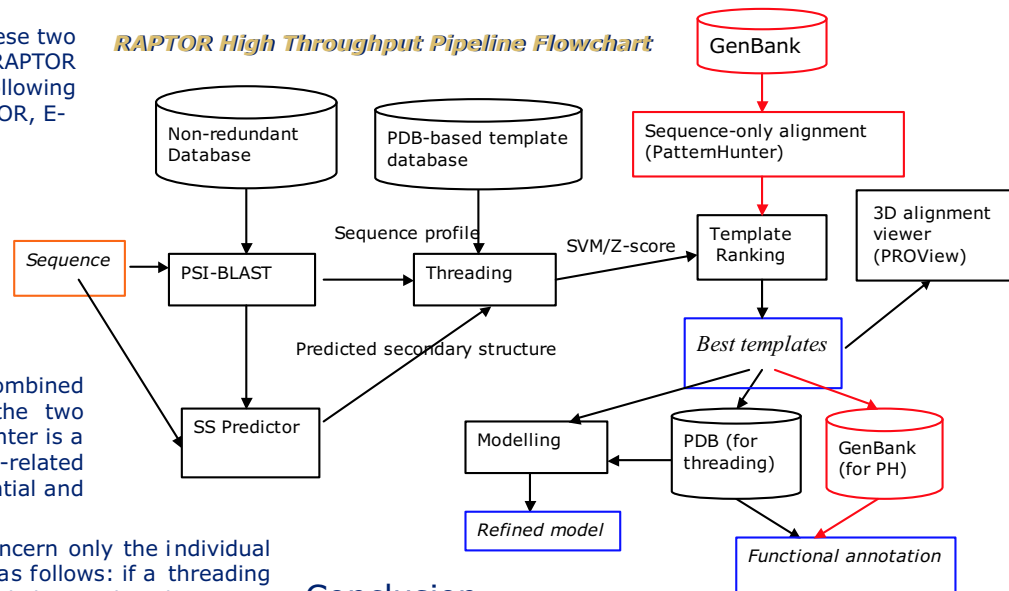
Lastly, RAPTOR's structural nature implies that it is capable of detecting unobvious fold-level similarities (oftentimes implying functional relatedness) that are otherwise overlooked by a sequence homology program. Therefore, RAPTOR is chosen to be the primary engine in the pipeline, with PatternHunter as an auxiliary tool to pick up homologies not available to RAPTOR due to database limitation (size and quality of annotation).

Assuming the pipeline is for annotating protein sequences, we can further improve the pipeline as follows:

PSI-BLAST (against NR or PDB) --> HMM (Hidden Markov Model against protein families) --> RAPTOR (against known protein structures)

Easy targets (at the family/superfamily levels) are identified by the sequence/profile-level tools (PSI-BLAST, HMM) and hence filtered out, and the hard targets (typically at the fold level) are picked up by RAPTOR. This scheme is expected to reduce inaccuracy for easy targets by employing more appropriate tools (sequence/profile-level) while preserving RAPTOR's power for hard targets, and hence improves overall prediction performance.

RAPTOR High Throughput Pipeline Flowchart



Conclusion

This poster presents an overview of a recent industry application where a pipeline system incorporates a threading program RAPTOR and a protein/DNA homology search program PatternHunter for high-throughput functional annotation generation. This system utilizes the threading method's fold recognition capability to infer functional roles of target proteins from predicted homologous or structurally similar templates. As a complement, PatternHunter runs against sequential (as opposed to structural) databases (e.g. GenBank), which widens the search space and enables discovery of relevant templates absent in structural databases that would otherwise be overlooked. A quality assessment scheme is in place to benchmark confidence scores from both approaches to determine the better annotation.

1. Bioinformatics Solutions Inc., Waterloo ON Canada
2. University of Waterloo, Computer Science, Waterloo ON Canada

J. Xu, M. Li, D. Kim, Y. Xu, *Journal of Bioinformatics and Computational Biology*, 1:1(2003), 95-118.
 J. Xu, M. Li, *PROTEINS: Structure, Function, and Genetics*, CASP5 special issue (2003).
 Bin Ma, John Tromp, Ming Li *Bioinformatics*, 18(3):440-445 March 2002.