TIGERSearch attacks Proteins*

Jasmin Šarić
European Media Laboratory
Heidelberg
Jasmin.Saric@eml.villa-bosch.de

Uwe Reyle
Institute for Computational Linguistics
University of Stuttgart
Uwe.Reyle@ims.uni-stuttgart.de

Abstract

Protein Databases (e.g. Swissprot - http://www.expasy.ch) contain information on regions and other sites of interest in the sequence of proteins. We present TIGERSearch to search efficiently the content of Swissprot FT lines and browse the results via a graphical user interface. The major search facilities comprise regular expressions, dominance/part-whole and precedence relations between domains as well as variables to connect queries. The translation from Swissprot to full structured TIGER-XML is automated and ontologically motivated.

Overview It is common usage to associate particular features to regions and other sites of interest in the sequence of proteins. The Swissprot feature table (FT) lists among others post-translational modifications, binding sites, enzyme active sites, local secondary structure and non-covalent bindings between amino acid residues that may be located at any distance in the sequence. The information is presented by triples consisting of a key word, an interval of the sequence and a short natural language description providing additional information about the feature (see Figure 1).

Key word	Start	End	Description
SIGNAL	1	21	
CHAIN	22	480	11S GLOBULIN BETA SUBUNIT
...	...	...	...
MOD_RES	22	22	PYRROLIDONE CARBOXYLIC ACID
DISULFID	124	303	INTERCHAIN (GAMMA-DELTA)

Figure 1. Part of 11SB_CUCMA Swissprot FT entry.

Our approach supports retrieval of proteins that are characterised by particular features. In contrast, SRS (http://srs.ebi.ac.uk), e.g., allows for querying feature values by means of regular expressions and boolean combinations of the retrieved results. In order to find all phosphorylated proteins with cytoplasmic location, for example, one may first ask for FT-lines containing "cytoplasmic" as its value, and then ask for those sequences among them that are phosphorylated. The result will show cytoplasmic domains as well as phosphorylated residues, but without, indicating whether the cytoplasmic domain is included in the phoshporylated residue. This example shows that the expressivity of the query-language underlying SRS is not powerful enough to allow to retrieve all the information that is represented in the FT-lines. And to our knowledge this is also true for other search tools on FT-lines. The reason is twofold:

(i) The information that a feature value *is associated with* a particular subsequence (a chain, domain etc.) of the protein is lost, when queries are combined by boolean expressions.

(ii) The part-whole information between subsequences (hence modification sites, motifs, domains and chains) that is encoded by the beginning and end positions of these sequences is not accessible to the query engine. It is, therefore only possible to ask for phosphorylated proteins with cytoplasmic tails, but not possible to ask in addition if the phosphorylation site is actually part of the cytoplasmic tail.

Our approach overcomes both of these shortcomings. It has a representation and query language that is powerful enough to deal with arbitrarily complex feature combinations and it is supported by the underlying ontological

*Part of this work has been supported by the TIGER project (http://www.ims.uni-stuttgart.de/projekte/TIGER/) and by the Klaus Tschira Foundation gGmbH, Heidelberg (http://www.kts.villa-bosch.de).

features of the tables, in particular part-whole precedence relations between sequence elements as well as between any subsequences of interest. For the time being our focus lies on the sophisticated retrieval of information wrt. one particular database. Accessing all relevant databases (like SRS) is in principle possible, but requires adaptation of the translation module to each of these databases.

Translating FT-line entries Starting out from Swissprot entries we automatically construct a database of proteins that represents the information given by the Swissprot feature tables. We translate Swissprot feature table lines (and other features like relevant sequence parts) into TIGER-trees, according to the following characteristics:

1. Relevant regions of a peptide sequence are represented as nodes with start and end point given in the Swissprot FT lines (e.g. *transmembrane region*), see Table 1.
2. Additional features for single amino acids may be associated to the terminals (as well to non-terminal nodes) of the tree by means of feature-value matrices (e.g. *Metal binding site*, *Phosphorylation*), see Table 2.
3. Relations between single amino acids (and thus between the appropriate domains) are represented with secondary edges (e.g. *Disulfide bonds*), see Table 3.

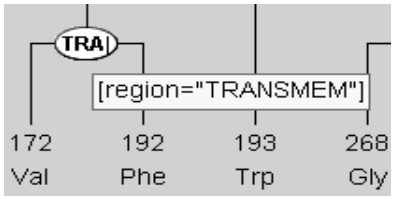

Table 1. Transmembrane

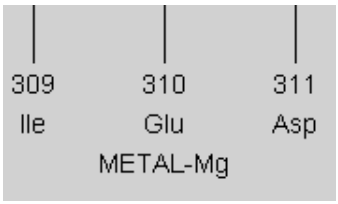

Table 2. AA Modification

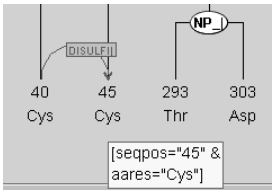

Table 3. Disulfid bridge

In TIGERSearch (see (Lezius 2002), (Lezius, Biesinger & Gerstenberger 2002)) the query that asks for proteins that have a phosphorylated cytoplasmic location is expressed as follows:

```
[location="CYTOPLASMIC"] >* [aa\-mod="PHOSPHORYLATION"]
```

Here ">*" expresses the (transitive) part-whole relation, i.e. that the phosphorylation site has to be within a cytoplasmic region.

The following query shows how we deal with restriction (i) above. In order to find all subsequences of interest (i.e. motifs, domains etc.) that are connected by a disulfide bond, we conjunctively combine the following four queries: (a) find all non-trivial subsequences (NT) that precede one another (precedence-relation is expressed by ".*"); (b) and (c) find amino acid residues (T) that are part of these sequences, respectively; (d) show that there is a disulfide bridge between these residues. That sub-queries (a) and (b), (a) and (c) as well as (b) and (d), and (c) and (d) talk about the same subsequences is guaranteed by the use of variables #v, etc.

```
#v:[NT] .* #w:[NT] & #v > #vt:[T] & #w > #wt:[T] & #vt >~DISULFID #wt
```

It is important to note that in our approach there is no crucial difference in the language representing the data and the query language - in contrast to SQL as query language for relational databases. TIGERSearch avoids this separation by taking the query language as a simple but powerful extension of the representation language. As an advantage the user has to learn just one language. The query language is based on regular expressions, and is supported by pull-down-menus for key word search. TIGERSearch is transparent, user-friendly and displays the results in a coloured tree-like graph structure.

References

Lezius, Wolfgang (2002): Ein Suchwerkzeug für syntaktisch annotierte Textcorpora. PhD thesis, IMS, University of Stuttgart.

Lezius, Wolfgang, Hannes Biesinger & Ciprian Gerstenberger (2002): *TIGER-XML Quick Reference Guide*. IMS, University of Stuttgart.