Indexing protein sequences with MINOS.

H. Ripoche [1] hr@lirmm.fr

E. Mephu Nguifo [2] mephu@lens.lifl.fr

J. Sallantin [3] js@lirmm.fr

[1,3] LIRMM
UMR 9928 CNRS – Montpellier II
161 rue Ada
F-34392 Montpellier

[2] Université d'Artois - IUT de LENS
Département d'Informatique
Rue de l'Université - SP 16
62307 LENS cedex

Abstract

This paper concerns the use of an object-oriented database for the analysis of protein sequences. We describe proteins either by bibliographic information or by prediction function such as Prosite patterns [2, 5]. We propose to use concept lattices—a tool used in information retrieval to build thesauruses—to classify protein sequences. This classification of proteins may help finding sequence alignments, or discussing about them. Conversely, sequence alignments can be used to criticize the structuration of sequences.

Keywords: *Knowledge Discovery in Databases, Concept Lattices, Object-Oriented Databases, Sequence Alignments, Protein Data Bank, Prosite.*

1 Knowledge discovery in a genetic database

Knowledge discovery has been defined as the *nontrivial extraction of implicit, previously unknown, and potentially useful information from data* [9]. It is also called data mining. These techniques have already been applied to the analysis of financial data, but genome projects, with their growing flows of data, constitute an attractive application domain.

We propose to use MINOS [17] (MINing Object System). This system uses machine learning techniques at two levels: to describe biological sequences, and—when they are described—to structure a set of sequences.

We apply this system to the discovery of patterns in genetic sequences. These patterns permit to detect potentially significant regions in sequences. The discovery of patterns constitutes an attractive domain in computational biology because the computing techniques involved to discover patterns are relatively simple whereas these patterns have a biological interest [12].

Then, we use concept lattices [18], a clustering method based on a binary representation of data to form groups of related objects (concepts). For us, these objects are sequences described by patterns or bibliographic information. There exists a generalization – specialization relationship between concepts that defines a kind of network (index) between concepts and thus between sequences.

Our approach combines three domains: object systems (we use an object-oriented database), information retrieval – data analysis (concept lattices) and genetic sequences. We present MINOS in the next section. Then we apply it to the structuration of a set of sequences and to alignment problems.

2 Overview of MINOS

This system takes advantage of an underlying object-oriented database management system to build prediction functions that recognizes properties in genetic sequences. Then, we show how these prediction functions can be used as descriptors to qualify sequences and structure a set of sequences.

2.1 Building and using prediction functions with OSQL

OSQL (Object-SQL) is useful to build prediction functions about genetic sequences. We have shown that these two steps—the production of consensus patterns and their use to classify new sequences—could be performed with an object-oriented query language (Figure 1).

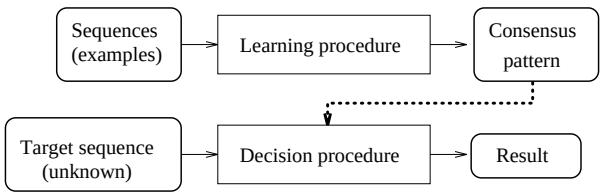

Figure 1: Construction and use of a consensus pattern. Rounded boxes represent sequences or results ; they are implemented as objects. Squared boxes represent procedures and are implemented as functions or methods.

Let's suppose that our "learning procedure" is called **makePattern** and the "decision procedure" **matchPattern** (these functions are analogous to Gribskof's **profileMake** and **profileSearch** functions [11] that deal with profiles, but we can use other prediction rules as well). O_2SQL, the query language of O_2 [3, 15] permits to trigger functions or methods in a query. For example, we can build a motif that recognizes the sequences of the "Globin" family, and use it in a two steps process as follows:

1. Define a pattern named "globin-finder" with a function that creates a pattern from a set of sequences[1]:

```
makePattern("select s from s in Sequences
             where s->family = GLOBIN",
            "globin-finder")
```

2. Use this pattern to detect globin-like sequences (**matchPattern** quantifies the proximity between a pattern and a sequence):

```
select s
from s in Sequences, p in Patterns
where p->name = "globin-finder"
and matchPattern(p,s) > 0.9
```

2.2 Clustering sequences with concept lattices

In this system, genetic sequences are represented by database objects. However, these objects have a special behaviour: they have a method that apply prediction functions on the sequence ; the result of this method is a set of patterns that matches the sequence. So, each sequence is described by a set of patterns. Then, we use concept lattices [18] to classify the sequences according to the patterns they have in common. In the vocabulary we adopt to define concept lattices, a sequence is an *example* and a pattern is an *attribute* describing an example. We define concept lattices in the next paragraphs.

A *concept* is a couple (E, A) where E is the set of all the examples sharing the attributes of A, and A is the set of all the attributes verified by the examples of E. There exists a relationship between concepts: A concept (E_1, A_1) is more specific than a concept (E_2, A_2) if and only if $E_1 \subset E_2$ (or $A_2 \subset A_1$). A concept lattice is a lattice that has concepts as nodes, and specialization – generalization relationships between concepts as links between nodes. We use the LEGAL system [14] to build concept lattices.

The following table represents the relationships between the examples $(1 \ldots 7)$ and their attributes $(a \ldots f)$. For instance, the example 5 is described by the attributes c and d. The corresponding lattice is displayed in Figure 2.

Examples	a	b	c	d	e	f
1		x	x	x	x	
2	x	x	x			
3	x	x				x
4				x	x	
5			x	x		
6	x					
7		x	x			

[1] A function such as **makePattern** can be called from the query interpreter ; this function has a parameter that is itself a query.

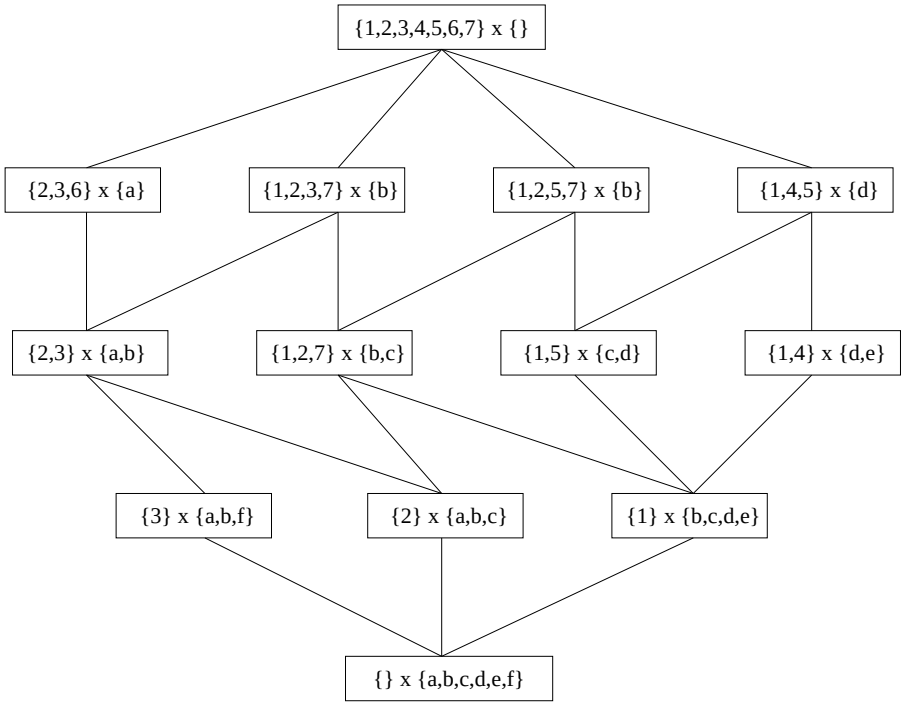

Figure 2: Concept lattice. Each concept is represented by a rectangle. Links between rectangles represent generalization – specialization links.

3 Applications

In this section, we see how genetic sequences are stored as objects of the database, and then, we provide two examples of structuration of the database with concept lattices. Our database contains a real-size data set of 964 protein sequences. In the first example, the sequences are described by bibliographic information. In the second one, the descriptors are 336 biological patterns.

3.1 Genetic sequences

A biological sequence is represented by a string of characters, an ident, and a set of properties. In our application, we use the genetic sequences of the Protein Data Bank [1, 4] (Figure 3)

3.2 Sequence descriptors

3.2.1 Bibliographic information

Bibliographic information is a means of describing sequences. In this example, we take authors' names as sequence descriptors.

3.2.2 Prosite patterns

In order to describe the protein sequences stored in our database, we have used A. Bairoch's Prosite data bank [2]. This data bank contains a set of patterns. Each pattern has been built

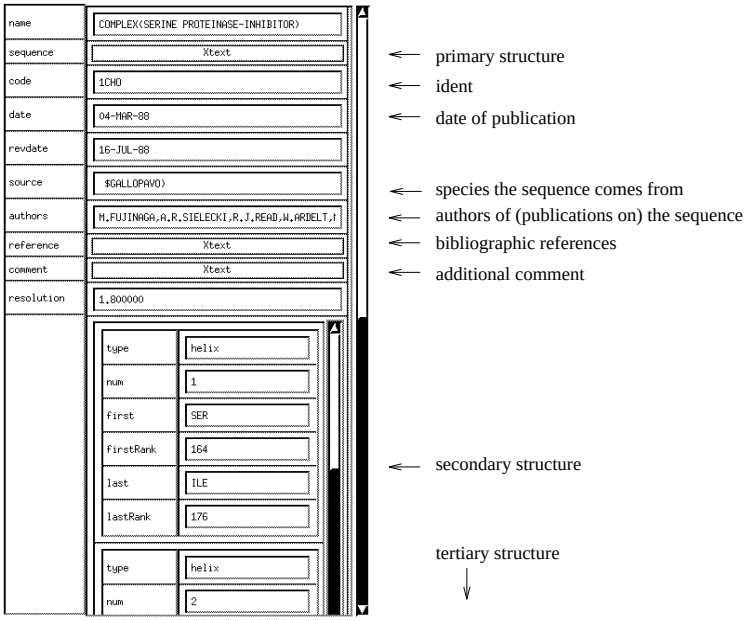

Figure 3: A Protein Data Bank sequence (1CHO).

from an alignment of a group of related proteins and can detect biologically significant regions in proteins. As A. Bairoch states, these significant regions are generally:

- Enzyme catalytic sites.
- Prostethic group attachment sites (heme, pyridoxal-phosphate, biotin, etc.).
- Amino acids involved in binding a metal ion.
- Cysteines involved in disulfide bonds.
- Regions involved in binding a molecule (ADP/ATP, GDP/GTP, calcium, DNA, etc.) or another protein.

And the criterions for a good pattern signature are: short size, ability to detect all or most of the sequences it is designed to describe without giving too many false positive results. The method that identifies Prosite's patterns in our database calls L.F. Kolakowski's ProSearch program [13].

Other knowledge bases based on Prosite patterns have also been studied. Ogiwara [16] defines a mail server that query protein sequences with Prosite patterns and returns 3D information on the sequence. Hirosawa [12] uses a deductive object-oriented database (*Quixote*) to manage patterns (user defined or coming from Prosite) and sequences. This system takes advantage of the inheritance mechanism to combine biological and computed information in a transparent manner.

3.3 Interpretation of concepts

3.3.1 Using authors' names

We have selected from the database the sequences having the string "proteinase" in their name. Then the sequences described by their authors are structured in a concept lattice. When we study the lattice, we discover three types of interpretation for the concepts:

- Identical sequences: sequences of identical primary structure but distinct names. They reveal redundancies in the Protein Data Bank.
- Homologous sequences.
- Non homologous sequences. In this case we can see from the names of the sequences that there is often a relation of inhibition between the sequences: one sequence is the inhibitor of the other ; or one sequence is a complex proteinase-inhibitor and the other the proteinase or the inhibitor alone.

Figure 4 gives an example of the second family of concepts. The sequences {1CHO, 2SEC, 2SNI, 3SGB} are grouped in a concept because they have all been studied by M.N.J. James. A phylogenetic tree representing the sequences of this concept is displayed (it shows that the sequences are weakly homologous). The algorithm that generates the phylogenetic tree has been designed by J. Gracy [10]. This phylogenetic tree permits to criticize the results given by our symbolic learning method (grouping sequences in concepts) with a numerical method (classifying the sequences of a concept by homology). Then the user interprets these results and accepts or rejects the new organization of knowledge.

3.3.2 Using Prosite patterns

In this example, we consider the sequences of the Protein Data Bank that are toxins. 13 sequences are concerned[2]. Each sequence responds to at least one of the following patterns of Prosite:

N-myristosylation site	a
Protein kinase C phosphorylation site	b
Casein kinase II phosphorylation site	c
Amidation site	d
Snake toxins signature	e
Glycosaminoglycan attachment site	f
Pancreatic trypsin inhibitor (Kunitz) family signature	g
N-glycosylation site	h
Tyrosine kinase phosphorylation site	i

The relationships between sequences and patterns are displayed in the table:

[2]We have used the April 93 version of the data bank.

Protein	a	b	c	(d)	e	(f)	(g)	(h)	(i)
(1ATX)	x								
1CTX	x	x	x	x	x				
(2MLT)									
3EBX	x	x	x		x				
(1SH1)	x								
2ABX	x	x			x				
6EBX	x	x	x		x				
1CDT		x	x		x				
1DTX	x	x	x			x	x		
(2SH1)	x								
1NXB	x	x	x		x				
(1SN3)	x							x	x
5EBX	x	x	x		x				

We can try to find the most significant relationships by the examination of:

- The **boolean matrix** (above): For instance, we can remove the patterns that appear in only one sequence (patterns d, f, g, h, i) and then remove the sequences that are described by at most one pattern (sequences 1ATX, 2MLT, 1SH1, 2SH1, 1SN3).
- The **concept lattice** (not displayed): Three kinds of concepts can be distinguished according to their position in the lattice:
  - Concepts with fewer objects than attributes. Since the sequences of these concepts have many common patterns, they are likely to be highly homologous.
  - Concepts having approximately as many objects as attributes.
  - Concepts with more objects than attributes. The sequences of these concepts are weakly or not homologous (the relation between sequences may result of a functional convergence).

Let's consider the concept ({1CTX, 3EBX, 1NXB, 5EBX, 6EBX} , {a, b, c, e}): These 5 sequences share at least 4 patterns[3]. This suggests that they are homologous. The alignment of the sequences of this concept (Figure 5) shows that 1CTX is weakly homologous to the four other sequences.

Then, if we look carefully at the patterns of 6EBX, we can see that they are duplicated. To get a better alignment, we can split 6EBX as follows:

```
1CTX    : IRCFIT...PDITSKDCPNGH.VCYTKTWCDAFCSIRGKRVDLGCAATCPTVKTGVDIQCCSTDNCNPFPTRKR
6EBX(1) : .ICFNHQSSQPQTTKTCSPGESSCYHKQWSD....FRGTIIERGCG..CPTVKPGIKLSCCESEVCNN......
6EBX(2) : RICFNHQSSQPQTTKTCSPGESSCYHKQWSD....FRGTIIERGCG..CPTVKPGIKLSCCESEVCN.......
1NXB    : RICFNQHSSQPQTTKTCSPGESSCYHKQWSD....FRGTIIERGCG..CPTVKPGIKLSCCESEVCN.......
3EBX    : RICFNHQSSQPQTTKTCSPGESSCYHKQWSD....FRGTIIERGCG..CPTVKPGIKLSCCESEVCN.......
5EBX    : RICFNHQSSQPQTTKTCSPGESSCYNKQWSD....FRGTIIERGCG..CPTVKPGIKLSCCESEVCN.......
```

To conclude about this example, we can say that our method finds groups of sequences that are likely to be homologous because they have some patterns in common ; moreover, these patterns may serve as *anchor points* in the alignment. Each group of sequences or "potentially interesting alignment" corresponds to a concept in the lattice. This example provides several results and prospects:

[3]In fact the similarity is higher because some patterns are repeated.

- A method that helps the discovery of potentially interesting sequence alignments.
- This method may help to *explain* the result of an alignment. For instance, the concept we have studied is linked to a more specific concept: ({1CTX}, {a, b, c, d, e}), which justifies the weak homology of 1CTX by the fact that it has an amidation site (pattern d).
- Concerning knowledge revision: Alignments can be used to criticize a group of sequences: in our example we have seen that 1CTX cannot be aligned easily with the other sequences.
- In the context of these toxin proteins, patterns a, b, and c are frequently associated with e (snake toxin). None of these three patterns alone does imply that the sequence is a toxin, but this could suggest that the conjunction of these three patterns imply that the sequence is a toxin. (This reasoning does not apply here actually because this conjunction of patterns is much less specific than e.)

4 Conclusion

Many systems appear that allow to index or cross-reference genetic data banks. For instance, *SRS* [7, 8] has been developed at the EMBL and references a large set of biological data banks. Other systems reference sequences according to homology relationship [6] in order to speed up homology searches.

Our system is flexible because it relies on an object-oriented database and is compatible with the other existing approaches. However, our purpose is to automate the process of knowledge acquisition from genetic sequences. We have shown how data fusion could be used to create valid knowledge through an interaction with a user who compares the results of several methods. More precisely, sequence homology helps the user in criticizing the structuration proposed by the system. The user may either agree or disagree on the structuration proposed, and his opinion can be used to enrich the description of existing objects. This yields to another structuration that can be criticized again in a reflexive manner.

References

[1] Abola E., Bernstein F.C., Bryant S.H., Koetzle T.F., and Weng J., *Protein Data Bank* in Crystallographic Databases - Information Content, Software Systems, Scientific Applications, eds. Allen F.H., Bergerhoff G., and Sievers R., Data Commission of the International Union of Crystallography, Bonn/Cambridge/Chester, 1987, pp. 107-132.

[2] Bairoch A., *The PROSITE dictionary of sites and patterns in proteins, its current status*, Nucleic Acids Res. 21:3097-3103(1993).

[3] Bancilhon F., Delobel C., Kanellakis P., *The O_2 book*, GIP Altair, 1989.

[4] Bernstein F.C., Koetzle T.F., Williams G.J.B., Meyer E.F., Jr., Brice M.D., Rodgers J.R., Kennard O., Shimanouchi T., and Tasumi M., *The Protein Data Bank: A Computer-based Archival File for Macromolecular Structures*, J.Mol.Biol., 112, 535-542 (1977).

[5] Bucher P., Bairoch A., *A generalized profile syntax for biomolecular sequence motifs and its function in automatic sequence intrepretation*, Proc. Second International Conference on Intelligent Systems for Molecular Biology, AAAI Press, 1994, Altman R. et al. eds, 53–61.

[6] Califano A., Rigoutsos I., *FLASH: A Fast Look-up Algorithm for String Homology*, Hunter L., Searls D., Shavlik J. eds., Proc. First International Conference on Intelligent Systems for Molecular Biology, AAAI Press, 1993, 145–153.

[7] Etzold T., Argos P., *SRS, an indexing and retrieval tool for flat file data librairies*, Comput. Appl. Biosci. 9:49–57, 1993.

[8] Etzold T., Argos P., *Transforming a set of biological flat file librairies to a fast access network*, Comput. Appl. Biosci. 9:59–64, 1993.

[9] W.J. Frawley, G. Piatetsky-Shapiro, C.J. Matheus, *Knowledge Discovery in Databases: An Overview*, introductory chapter of Knowledge Discovery in Databases, AAAI Press - The MIT Press, 1991.

[10] Gracy J., Chiche L., Sallantin J., *A Modular Learning Environment for Protein Modeling*, Proc. First International Conference on Intelligent Systems for Molecular Biology, Ed. Hunter L., Searls D., Shavlik J., AAAI Press, 1993, 145–153.

[11] Gribskof M., McLachlan A. D., Eisenberg D., *Profile analysis: Detection of distantly related proteins*, Proc. Natl. Acad. Sci. USA, vol. 84, 4355–4358, July 1987, Biochemistry.

[12] Hirosawa M., Tanaka R., Ishikawa M., *Motif Knowledge Base Based on Deductive Object-Oriented Database Language*, Proc. Genome Informatics Workshop IV, Universal Academic Press, Yokohama, 1993, 10–16.

[13] Kolakowski L.F. Jr., Leunissen J.A.M., Smith J.E., *ProSearch: fast searching of protein sequences with regular expression patterns related to protein structure and function*, Biotechniques 13:919-921(1992).

[14] Mephu E., Sallantin J., *Prediction of Primate Splice Junction Gene Sequences with a Cooperative Knowledge Acquisition system*, Proc. First International Conference on Intelligent Systems for Molecular Biology, Ed. Hunter L., Searls D., Shavlik J., AAAI Press, 1993, 292–300.

[15] *O_2: Reference Manual*, O_2 Technology, 1994.

[16] Ogiwara A., Uchiyama I., Kanehisa M., *Sequence Motif Analysis and Retrieval Tool*, Proc. Genome Informatics Workshop IV, Universal Academic Press, Yokohama, 1993, 402–410.

[17] Ripoche H., Sallantin J., *Knowledge discovery in a genetic database: The MINOS system*, Proc. of the 28th Hawaii International Conference on System Sciences, 1995, Forthcoming.

[18] Wille R., *Concept lattices and conceptual knowledge systems*, Semantic Networks in Artificial Intelligence, Ed. Lehman F., Pergamon Press, 1992, 493–515.

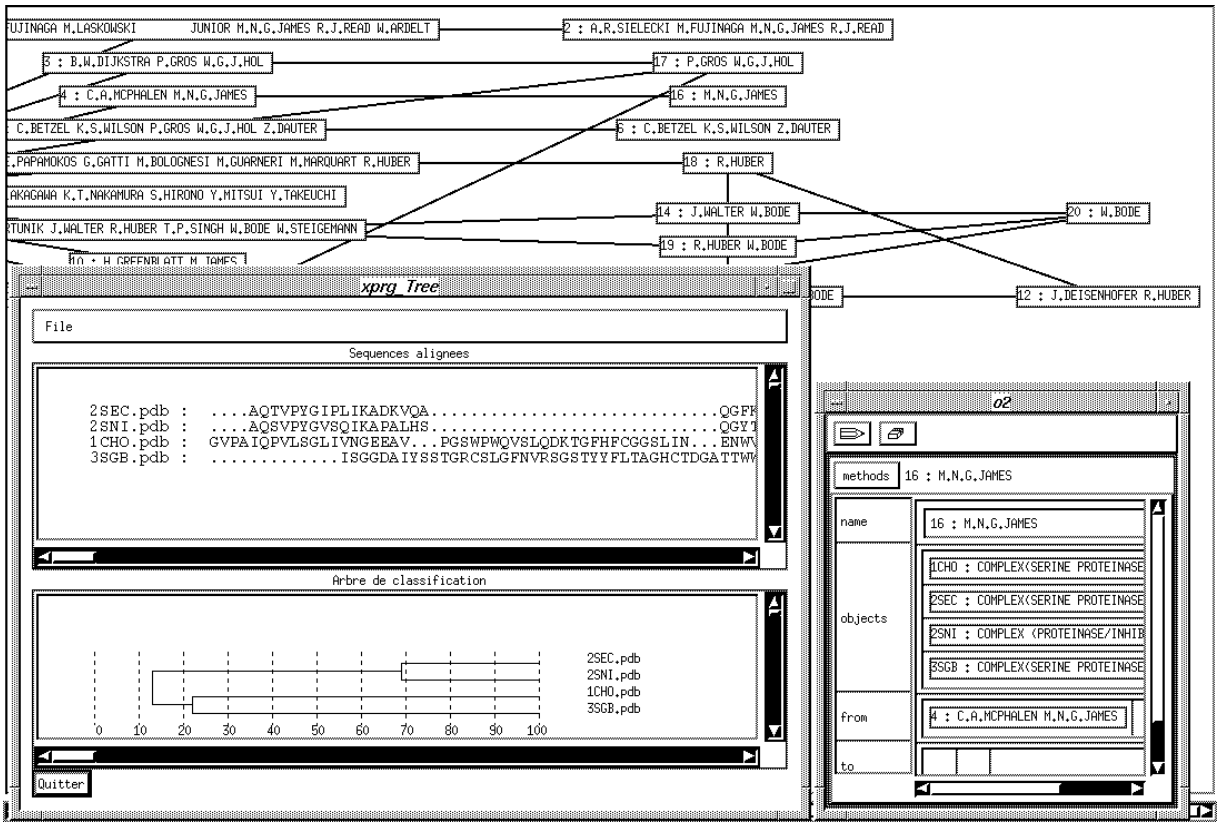

Figure 4: The concept lattice is displayed in the background. Each node (concept) displays the name of the attributes describing the objects it contains. In the bottom right window, a concept is open ; we can see its objects. The bottom left window shows the alignment of the proteins in a concept and the related phylogenetic tree.

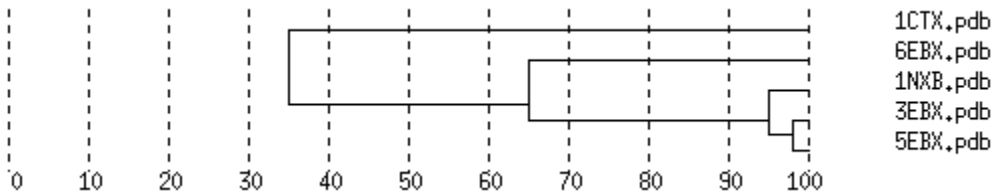

Figure 5: Phylogenetic tree of concept ({1CTX, 3EBX, 1NXB, 5EBX, 6EBX} , {a, b, c, e}).