

# Smoothing 3D protein structure motifs through graph mining and amino-acids similarities

**Wajdi Dhifli**

Blaise Pascal University  
Clermont-Ferrand, France

**Rabie Saidi**

European Bioinformatics Institute  
Cambridge, United Kingdom

**Engelbert Mephu Nguifo**

Blaise Pascal University  
Clermont-Ferrand, France

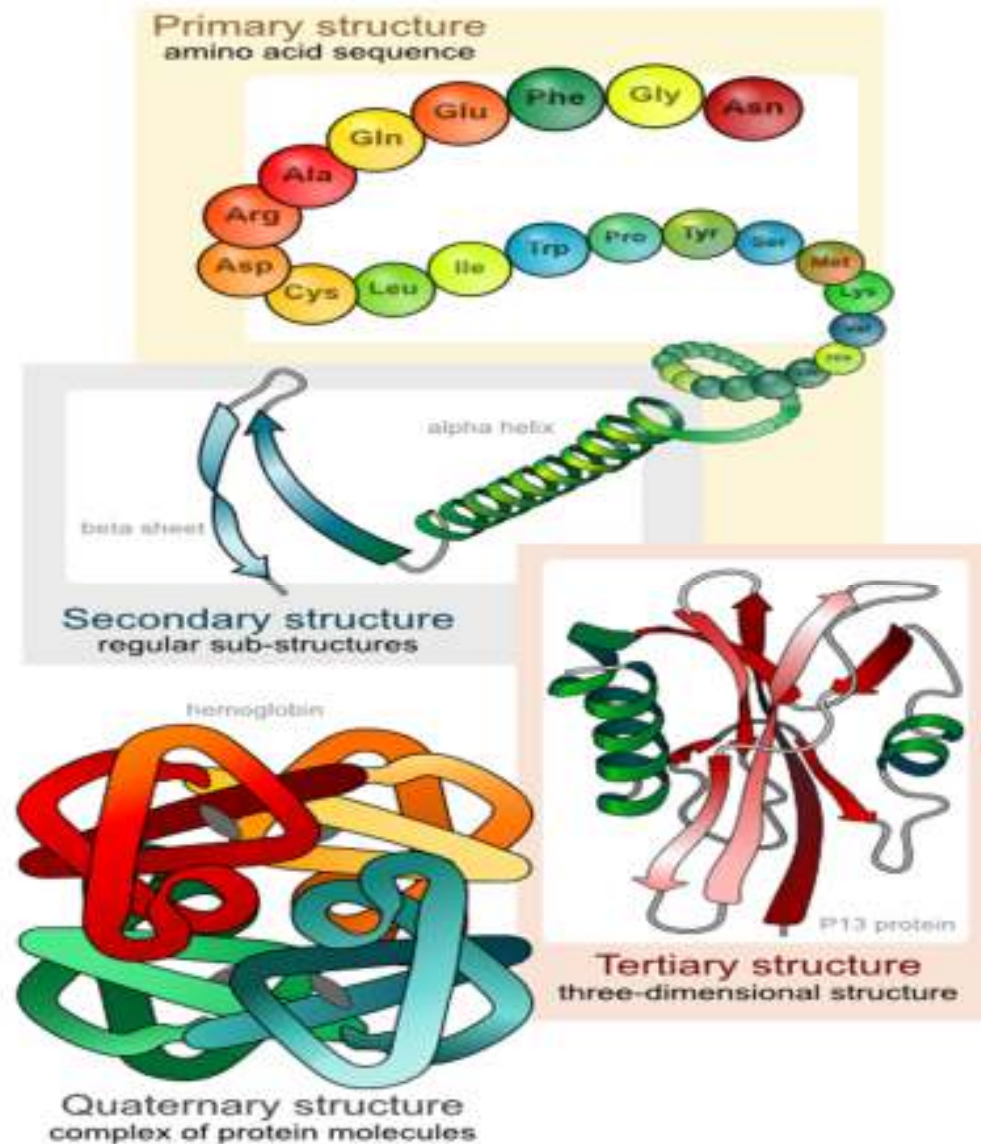
**JOBIM**  
**Juillet 2013**

# Context and motivations

## Proteins

A combination of amino acids within an alphabet of 20 :

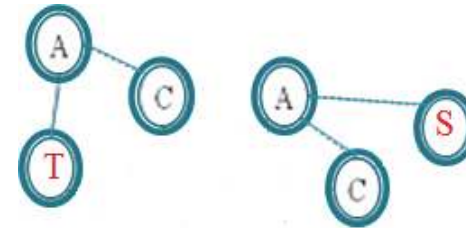
- ▶ Calcium
- ▶ Arginine
- ▶ Glutamine
- ▶ Glycine
- ▶ ...



# Context and motivations

- ▶ During the evolution, proteins go through changes.
  - **Mutation** : is a substitution that exchanges one amino acid to another

CTGGAG  
CTGGGG



In the literature, there exist substitution matrices expressing scores of substitution between each possible pair of amino acids.

**Substitution matrix :**  
**Blosum 62**

A	Ala	4																							
R	Arg	-1	5																						
N	Asn	-2	0	6																					
D	Asp	-2	-2	1	6																				
C	Cys	0	-3	-3	-3	9																			
Q	Gln	-1	1	0	0	-3	5																		
E	Glu	-1	0	0	2	-4	2	5																	
G	Gly	0	-2	0	-1	-3	-2	-2	6																
H	His	-2	0	1	-1	-3	0	0	-2	8															
I	Ile	-1	-3	-3	-3	-1	-3	-3	-4	-3	4														
L	Leu	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4													
K	Lys	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5												
M	Met	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5											
F	Phe	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6										
P	Pro	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7									
S	Ser	1	-1	1	0	-1	0	0	-1	-2	-2	0	-1	-2	-1	4									
T	Thr	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5							
W	Trp	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11						
Y	Tyr	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7					
V	Val	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-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					
	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V					

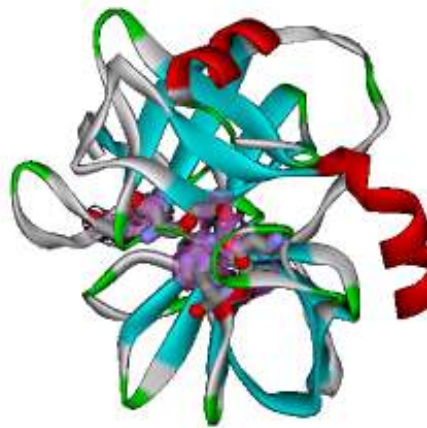
# Context and motivations

- ▶ The primary structure has been extensively studied, unlike the other structures
- ▶ However, the tertiary (3D) structure is more interesting:
  - It contains the primary + interactions between amino acids
  - The function of a protein is highly related to its 3D structure

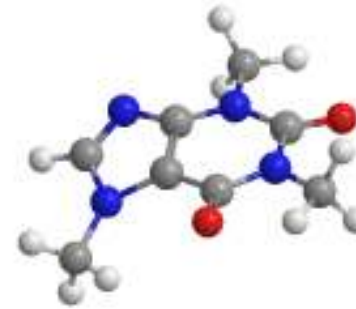


# Context and motivations

- ▶ Graphs are powerful representation framework
- ▶ A protein 3D structure can be represented by a graph of amino acids (protein contact map)
  - Amino acids => Nodes (labeled with the amino acid type)
  - Interactions between amino acids => Edges



Protein 3D structure




Protein graph

➔ Use graph mining techniques to study protein 3D structure

# Context and motivations

## Frequent subgraph mining

- ▶ One current trend in graph mining is frequent subgraph discovery
  - ▶ It consists on finding subgraphs that frequently occur in graph data
- 
- ▶ Among the most powerful techniques to study proteins is to look for recurrent substructures then use them for analysis
- 
- ▶ Protein **3D structures** → Protein **graphs** → Use the frequent subgraphs as patterns to describe proteins → Each subgraph represents a 3D-motif

# Context and motivations

## Frequent subgraph discovery approaches

### ▶ **ILP approaches**

- WARMR : King R.D., Srinivasan A. and Dehaspe L. (J. of Computer-Aided Molecular Design 2001)
- FARMER : Nijssen, S. and Kok, J. (IJCAI 2001)
- ...

### ▶ **Apriori based approaches**

- AGM/AcGM : Inokuchi et al (PKDD 2000)
- FFSM : Huan et al (ICDM 2003)
- ...

### ▶ **Pattern growth based approaches**

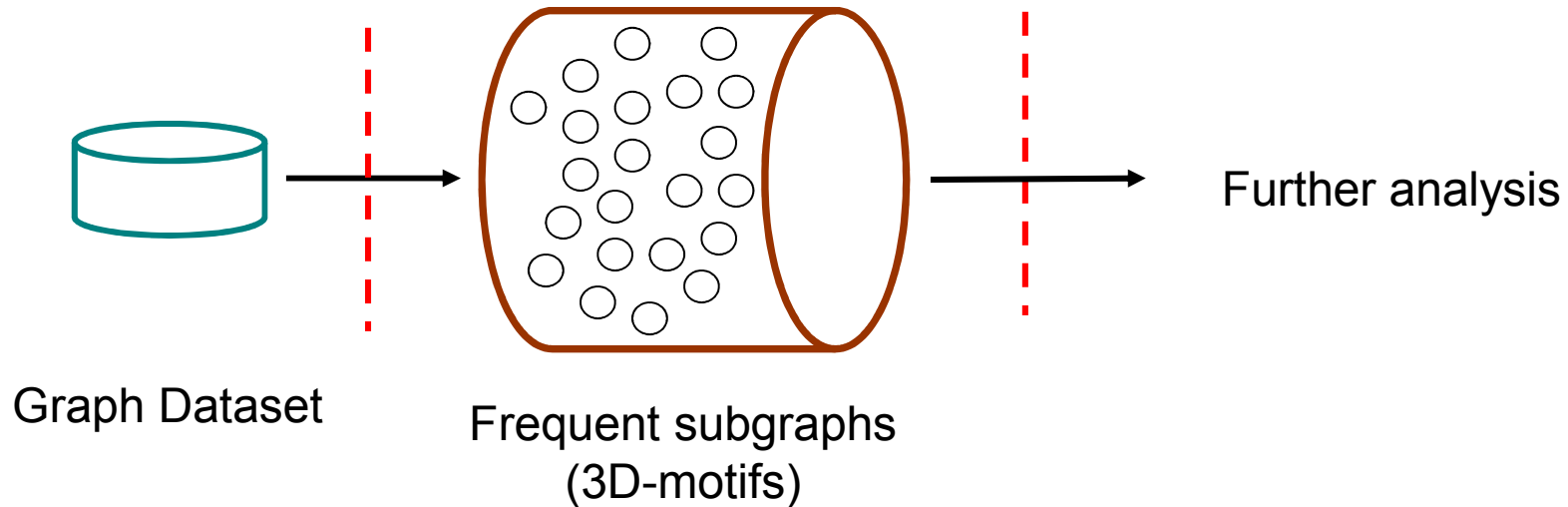
- Gspan : Yan and Han (ICDM 2002)
- Gaston : Nijssen and Kok (KDD 2004)
- ...

### ▶ **Closed and maximal**

- Closegraph : Yan and Han (KDD 2003)
- Margin : Thomas, L.T., Valluri, S.R. and Karlapalem, K. (ICDM 2006)
- ...

# Context and motivations

## Frequent subgraph issues



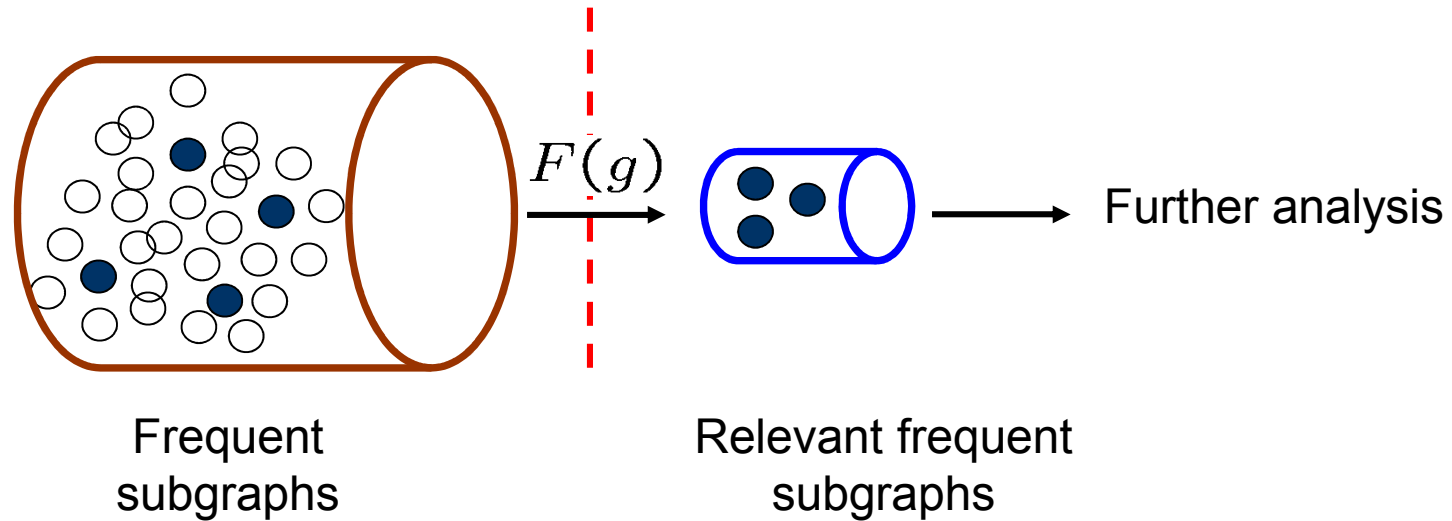
### Main issues:

- ✘ Exponential Pattern Set : huge number of subgraphs
- ✘ Interpretation : role of each subgraphs ?
- ✘ No guarantee of the relevance of the discovered subgraphs: redundancy due to structural or semantic similarities



# Feature selection

## General framework of feature selection



### Aims:

- ▶ Decreasing the exponential number of discovered frequent subgraphs
- ▶ Enhancing (or at least maintaining) the quality of the feature set

# Feature selection

## Feature selection techniques

### ▶ Learning task dependent selection

- Find a subset of features that preserves/enhance the output prediction capabilities

### ▶ Learning task independent selection

- Reduce the features without regard to the learning task

#### 1. Filter approaches (univariate / multivariate)

- Assess the relevance of features based on their properties

#### 2. Wrapper approaches

- Various subsets of features are generated and evaluated by training and testing a specific learning model

#### 3. Embedded approaches

- The selection is made into the model construction by searching in the combined space of feature subsets. Thus, they are specific to a given learning algorithm

# Feature selection

## Existing feature selection approaches for subgraphs

### ▶ Random sampling

- MUSK: Geng Li, Murat Semerci, Bulent Yener, and Mohammed J. Zaki (SDM 2009)
- ...

### ▶ Top-k and Clustering based approaches

- Extracting redundancy-aware top-k patterns: Dong Xin, Hong Cheng, Xifeng Yan, and Jiawei Han (KDD 2006)
- RING: Shijie Zhang, Jiong Yang, and Shirong Li. (ICDM 2009)
- TGP: Yuhua Li, Quan Lin, Ruixuan Li, and Dongsheng Duan (ADMA 2010)
- ...

### ▶ Constraints based approaches

- D&D : Yuanyuan Zhu, Jeffrey Xu Yu, Hong Cheng, and Lu Qin (CIKM 2012)
- CORK : Marisa Thoma, Hong Cheng, Arthur Gretton, Jiawei Han, Hans-Peter Kriegel, Alex Smola, Le Song, Philip S. Yu, Xifeng Yan, and Karsten M. Borg-wardt. (SADM 2010)
- MIP : Frédéric Pennerath, and Amedeo Napoli (ECML-PKDD 2009)
- COM : Ning Jin, Calvin Young, and Wei Wang (CIKM 2009)
- ...

**What about the domain knowledge ?  
Can we use them in the selection ?**

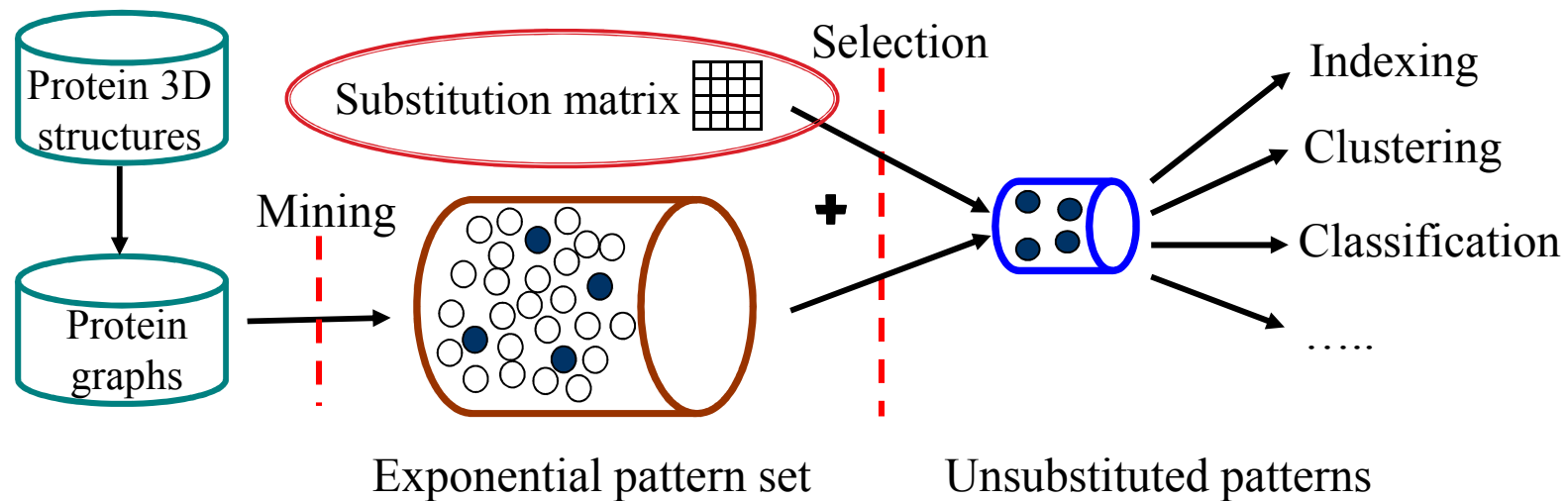
# Selection of Representative Unsubstituted subgraphs by means of substitution matrix

## Contribution:

- ▶ Some amino acids have similar proprieties
  - Some substitution can be without effect on the function nor the structure of the protein
- Same thing can be deduced for subgraphs
- ▶ **Idea** : use substitution matrices to define similarity between the discovered frequent subgraphs
  - Number of features will be reduced ?
  - Any impact on the quality of the pattern set?

# Selection of Representative Unsubstituted subgraphs by means of substitution matrix

## General framework of the selection approach



- ▶ Incorporate a domain knowledge (the substitution matrix) in the selection
- ▶ Keep only one subgraph from every set of substitutable subgraphs
- ▶ The selected subgraphs represents the set of **representative unsubstituted patterns**



# Selection of Representative Unsubstituted subgraphs by means of substitution matrix

## Preliminaries

1. Ranking function :

$$M_{el} = \frac{M(l, l)}{\sum_{i=1}^{|L|} M(l, l_i)}$$

$$M_{patt}(P) = 1 - \prod_{i=1}^{|V_P|} M_{el}(P[i])$$

2. Similarity function :

$$S_{el}(v, v') = \frac{\mathcal{M}(l, l')}{\mathcal{M}(l, l)}$$

$$S_{patt}(P, P') = \frac{\sum_{i=1}^{|V_P|} S_{el}(P[i], P'[i])}{|V_P|}$$

# Selection of Representative Unsubstituted subgraphs by means of substitution matrix

## Main algorithm

**Data:**  $\Omega$ ,  $M$ ,  $\tau$  (set of frequent subgraphs, substitution matrix, substitution threshold)

**Result:** :  $\Omega^*$  (unsubstituted patterns)

### Begin UnSubPatt

1. **Divide** the set of frequent subgraphs into groups of subgraphs having the same size and order
2. **For each** group of subgraphs
3. **Sort** the subgraphs by descending order of  $M_{patt}$
4. **For each** subgraph  $Sg_i$
5. **Delete** all the other subgraphs  $Sg_j$  it substitutes
6. Occurences  $(Sg_i) = \text{Occurences}(Sg_i) \cup \text{Occurences}(Sg_j)$

End.

# Selection of Representative Unsubstituted subgraphs by means of substitution matrix

## Substitution cases:

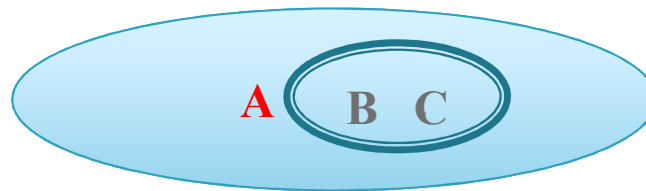
A, B, C and D: four structurally isomorphic subgraphs |  $M_{patt}(A) > M_{patt}(B) > M_{patt}(C) > M_{patt}(D)$  :

- Independency



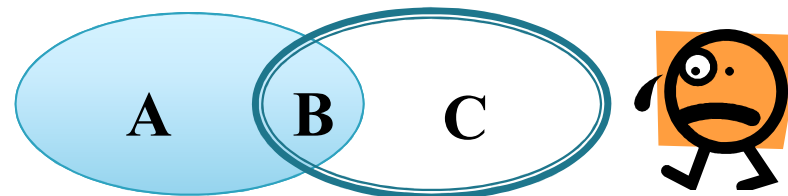
A substitutes B  
C substitutes D

- Inclusion



A substitutes B  
B substitutes C  
A substitutes C

- Intersection

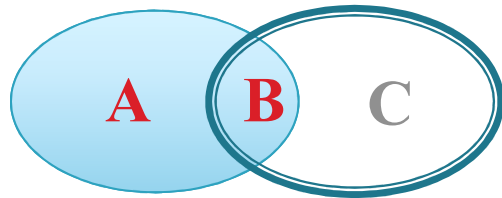


A substitutes B  
B substitutes C

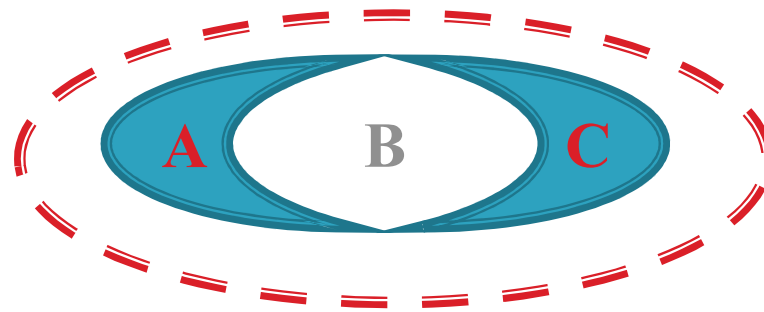
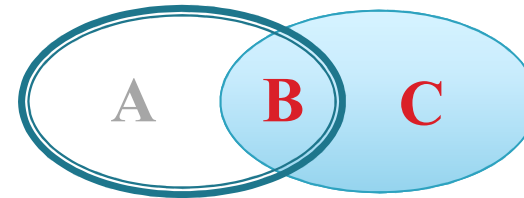
# Selection of Representative Unsubstituted subgraphs by means of substitution matrix

## Intersection

### ► Solution ?



OR ???



- Considering more distinct features
- Better description
- More independent features

# Experiments & results

## Experimental data

Dataset	SCOP ID	Family name	Pos	Neg	#Proteins	Avg # nodes	Avg # edges
DS1	52592	G proteins	33	33	66	246	971
DS2	48942	C1 set domains	38	38	76	238	928
DS3	56437	C-type lectin domains	38	38	76	185	719
DS4	88854	Protein kinases, catalytic subunit	41	41	82	275	1077

**SCOP ID**: identifier of protein family in SCOP, **Pos**: positive proteins sampled from a selected protein family, **Neg**: negative proteins randomly sampled from the Protein Data Bank, **# Proteins**: the number of protein structures in the whole dataset, **Avg# nodes**: average number of nodes, **Avg# edges**: average number of edges.



# Experiments & results

## Evaluation methodology

### ▶ Reduction

- Selection Rate =  $\text{Number of selected subgraphs} * 100 / \text{Number of frequent subgraphs}$

### ▶ Interestingness

- Average Classification Accuracy : 5 runs \* 5 CV

# Experiments & results

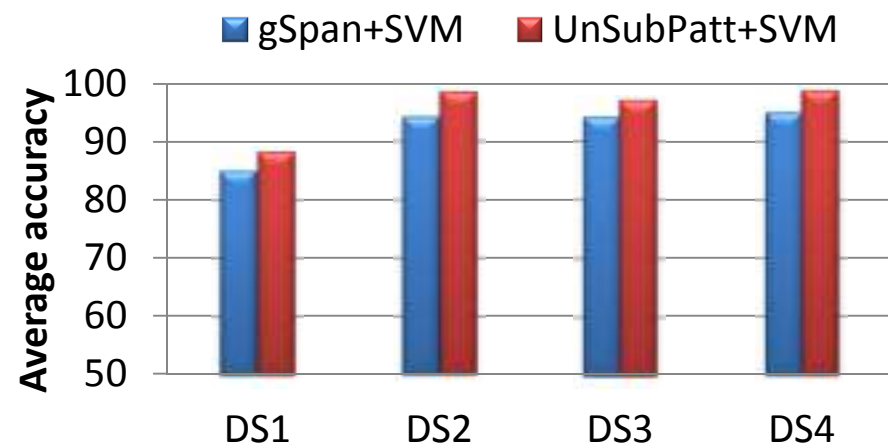
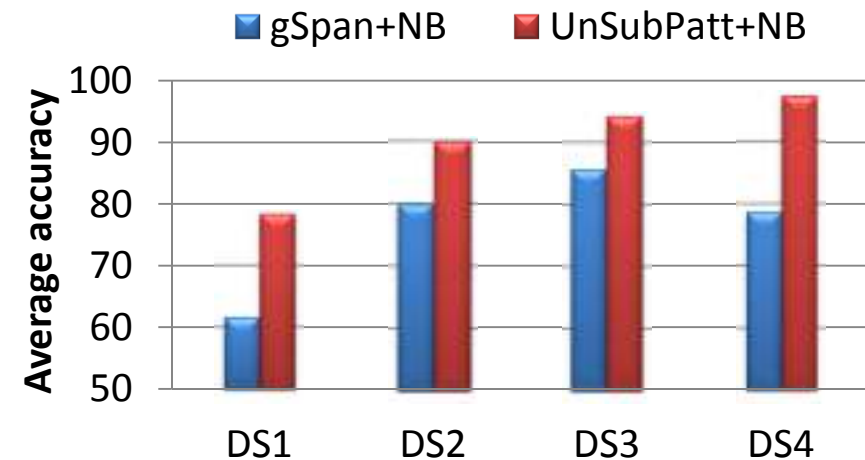
## Results

- The frequent subgraphs  $\Omega$  are extracted using **gSpan** with a frequency  $\geq 30\%$
- Substitution matrix: **Blusom62**
- Substitution threshold: **30%**

$|\Omega|$  : Number of frequent subgraphs,  $|\Omega^*|$  : unsubstituted patterns, and the selection rate.

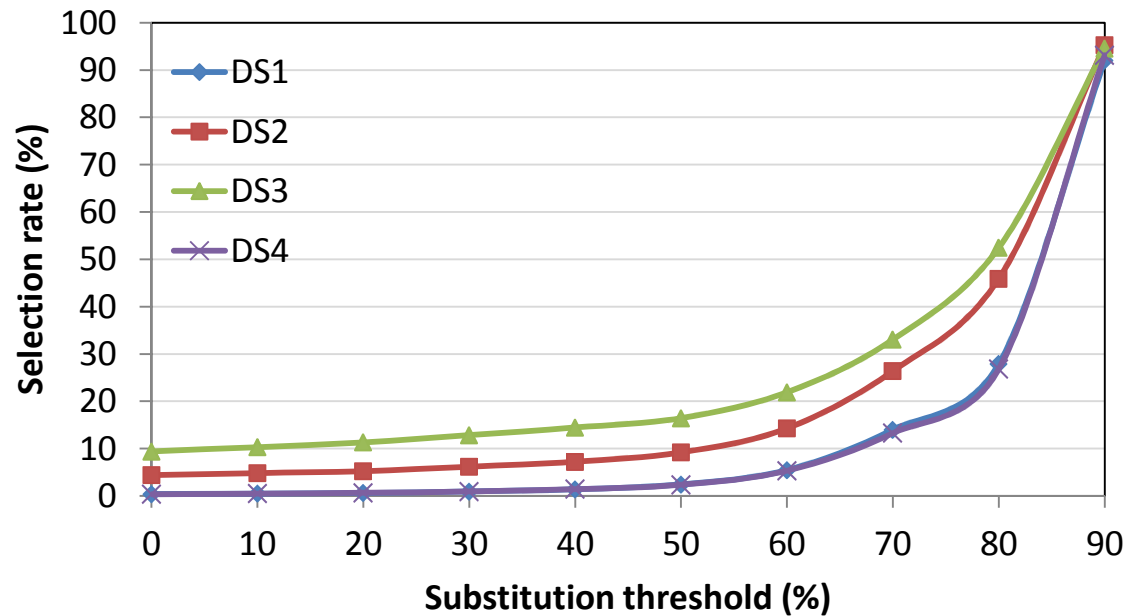
Dataset	$ \Omega $	$ \Omega^* $	Selection rate (%)
DS1	799 094	7291	0.91
DS2	258371	15898	6.15
DS3	114792	14713	12.82
DS4	1073393	9958	0.93

Classification accuracy by NB and SVM using frequent subgraphs (**gSpan**) and unsubstituted patterns (**UnSubPatt**).



# Experiments & results

## Impact of variation of the substitution threshold on the selection rate

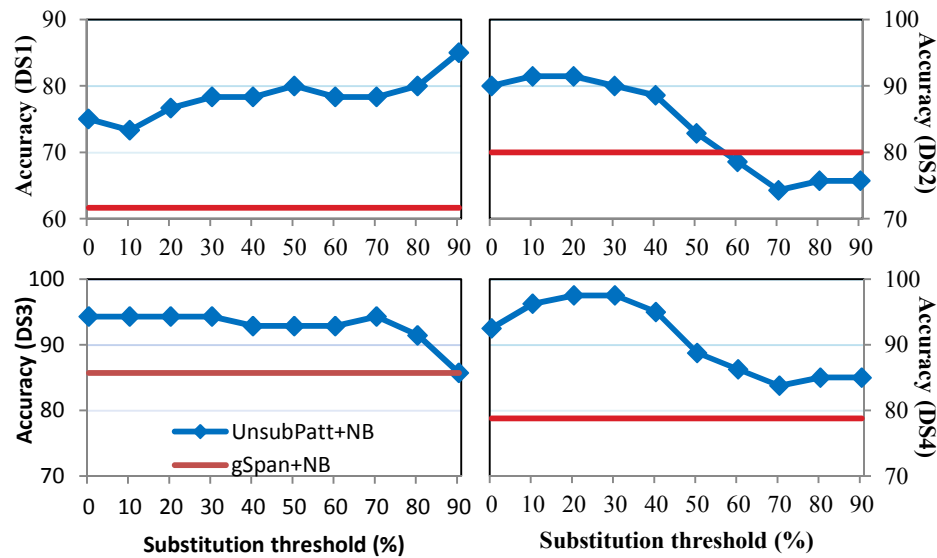


Rate of unsubstituted patterns ( $\Omega^*$ ) from the initial set of frequent subgraphs ( $\Omega$ ) depending on the substitution threshold.

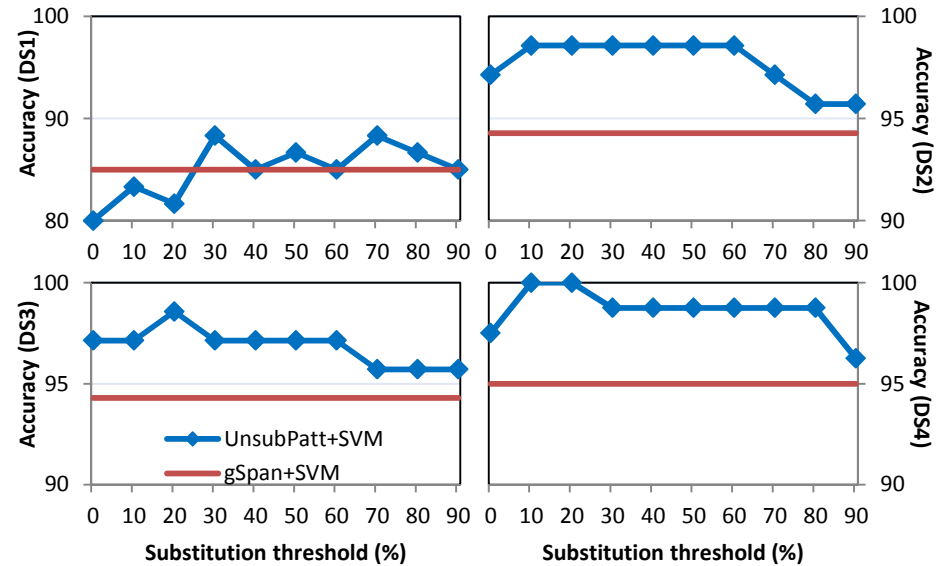
# Experiments & results

## Impact of variation of the substitution threshold on the classification accuracy

### Classification accuracy by NB.

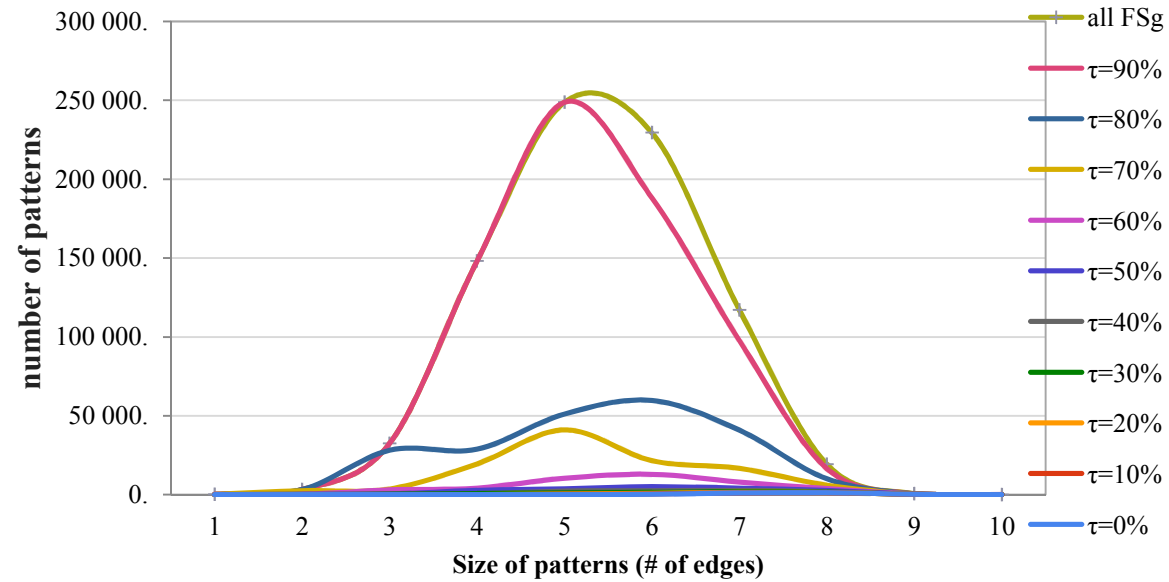


### Classification accuracy by SVM.



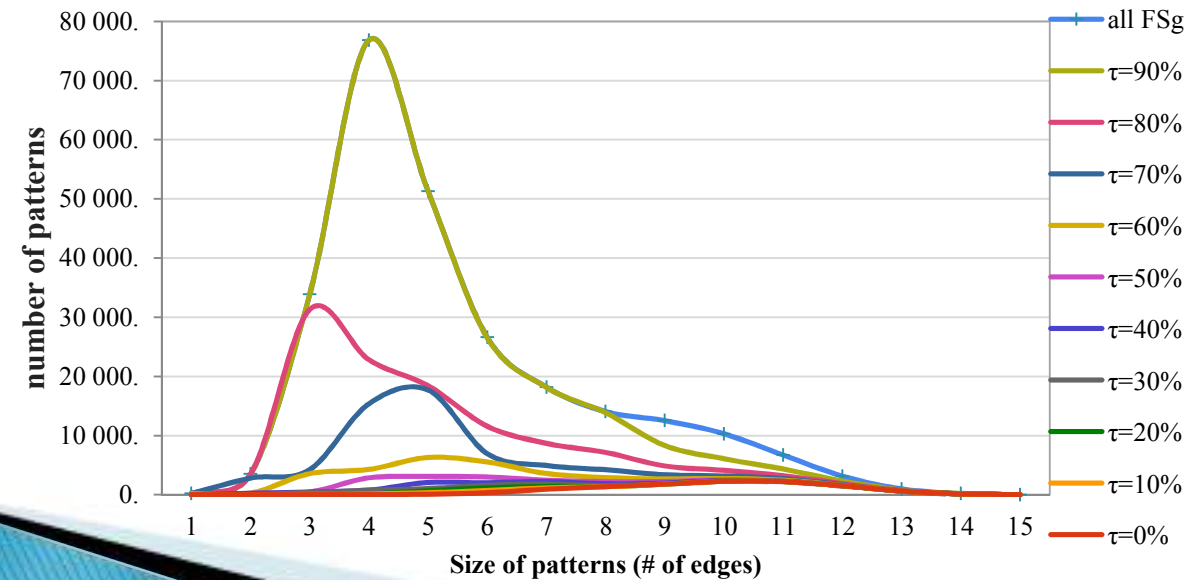
# Experiments & results

**DS1**



**Patterns distribution**  
**(# of edges)**

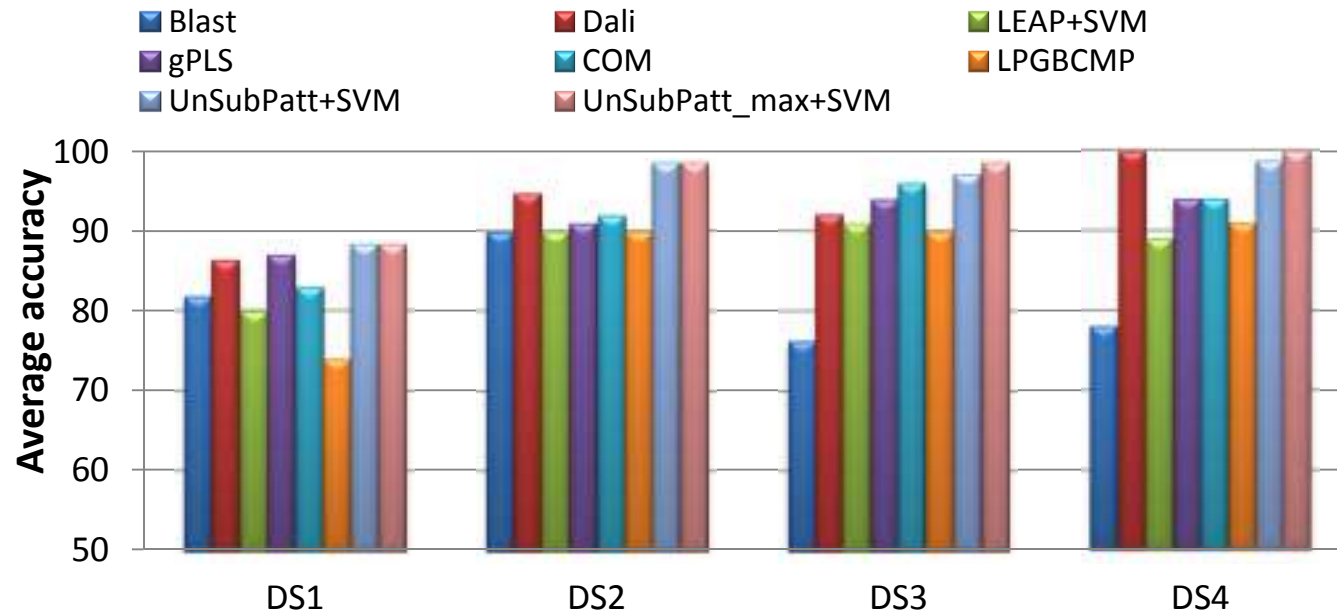
**DS2**





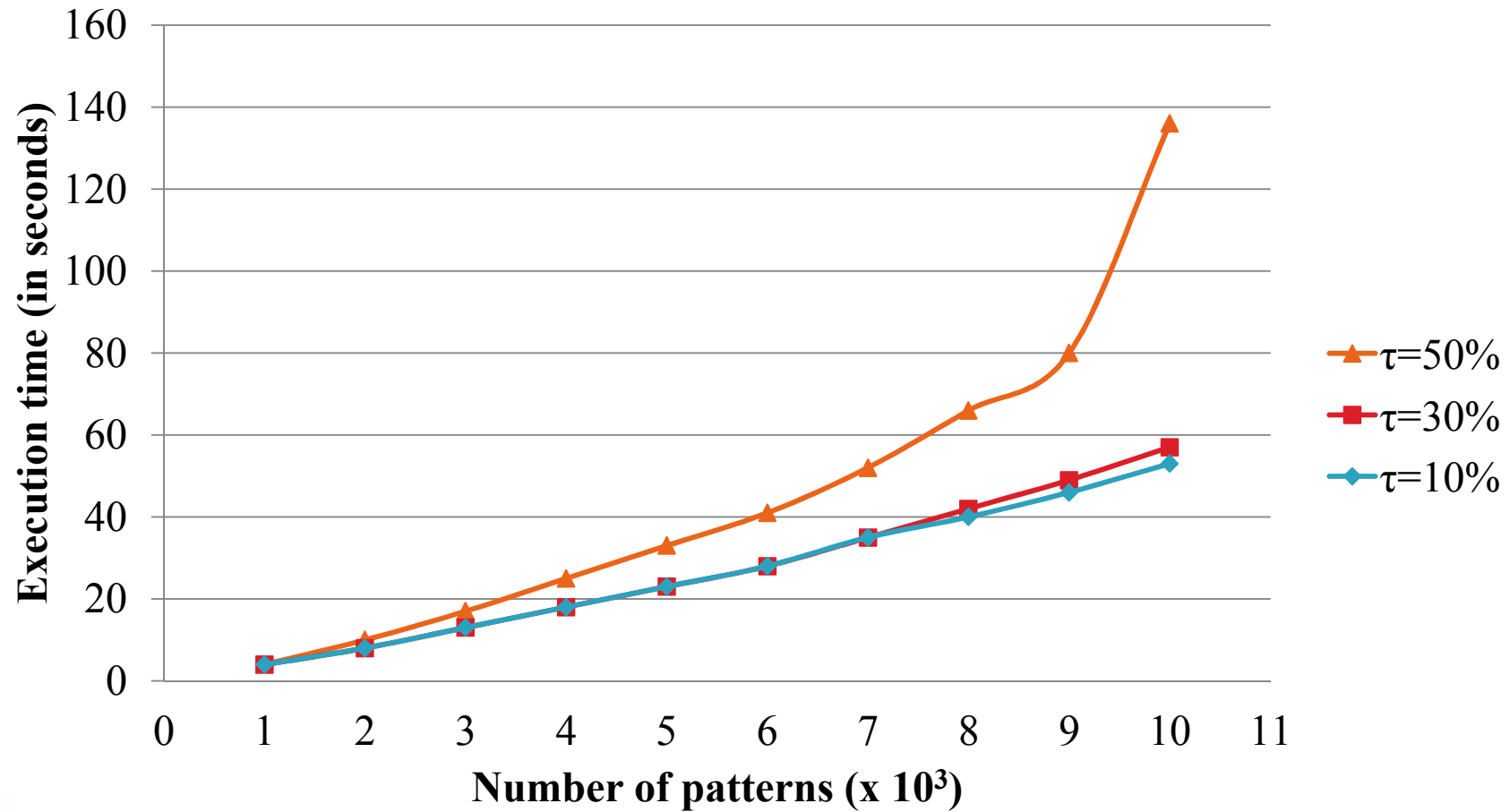
# Experiments & results

## Comparison with other pattern selection methods



# Experiments & results

## Runtime analysis



# Experiments & results

## Parallelization of « UnSubPatt »

**Data:**  $\Omega$ ,  $M$ ,  $\tau$  (set of frequent subgraphs, substitution matrix, substitution threshold)

**Result:**  $\Omega^*$  (unsubstituted patterns)

### Begin UnSubPatt

1. **Divide** the set of frequent subgraphs into groups of subgraphs having the same size and order
2. **For each** group of subgraphs
3. **Sort** the subgraphs by descending order of  $M_{patt}$
4. **For each** subgraph  $Sg_i$
5. **Delete** all the other subgraphs  $Sg_j$  it substitutes
6. **Occurrences** ( $Sg_i$ ) = Occurences ( $Sg_i$ )  $\cup$  Occurences ( $Sg_j$ )

End.

**Compute the substitution of each group in  
Parallel processes**

# Conclusion

- ▶ Considering the substitution between amino acids:
  - Enhance the selection results in terms of reduction and quality
  - Allows detecting similarities between patterns that current selection approaches do not detect
- ▶ The proposed approach :
  - Is scalable and can be easily parallelized
  - Can be used on protein 3D structures as well as sequences (seen as line graphs)
  - Is not a learning-task driven approach => can be used in different mining tasks

## Prospects

- ▶ Embed the selection approach within the extraction process
- ▶ Consider also the insertions and deletions over subgraphs with different sizes

# Thanks