MATCH
Communications in Mathematical and in Computer Chemistry

MATCH Commun. Math. Comput. Chem. **74** (2015) 575-590

ISSN 0340 - 6253

n-Gram Analysis of COG Categorized Protein Sequences

Ulfeta A. Marovac[1*], Nenad S. Mitić[2]

[1]*State University of Novi Pazar, Vuka Karadžića bb., 36300 Novi Pazar, Serbia*
[2]*University of Belgrade, Faculty of Mathematics, Studentski trg 16, 11000 Belgrade, Serbia*

(Received April 6, 2015)

Abstract

The classification of proteins categorized in the Cluster of Orthologous Groups (COGs) is important for better understanding of biological processes, as well as for various pathological conditions in human and other organisms. In this paper, a model for classification of proteins in the COG categories based on characteristic amino acid *n*-grams is proposed. A novel method, based on Boolean algebra, for extracting *n*-grams which characterize proteins belonging to a certain COG category is presented. The presented method significantly reduces the number of processed *n*-grams, which implies the reduction of required storage space and processing time. The obtained results show that the proteins of a certain COG category contain *n*-grams which satisfy specific patterns; such *n*-grams are unique, related to different COG categories. The model for classification based on the proposed method assigns a correct COG category to a protein with the confidence of 96%.

1 Introduction

The COG database [1] is an attempt of phylogenetic classification of proteins encoded by the whole genome sequences on the basis of the orthology concept [2]. Each COG is a set of at least three or more proteins that are inferred to be orthologs. Ortholog proteins are defined as proteins in different species that evolved from a common ancestral protein [3, 4], and retained the same function during the evolution. Identification of an orthology is required for predicting the exact function of a new protein. COG constructive procedure (which uses BLAST as a sequence comparison method) is based on simple notion: any set of at least three or more proteins from different genomes that are more similar to each other than to any other proteins from their source genomes, are most likely to belong to an orthologous family [5, 6]. Today, proteins are classified in 23 COG categories (shown in Table 1) according to the function of belonging proteins.

Table 1: COG functional categories

COG category	Function
A	RNA processing and modification
B	Chromatin Structure and dynamics
C	Energy production and conversion
D	Cell cycle control and mitosis
E	Amino Acid metabolism and transport
F	Nucleotide metabolism and transport
G	Carbohydrate metabolism and transport
H	Coenzyme metabolism
I	Lipid metabolism
J	Translation
K	Transcription
L	Replication and repair
M	Cell wall/membrane/envelop biogenesis
N	Cell motility
O	Post-translational modification, protein turnover, chaperone functions
P	Inorganic ion transport and metabolism
Q	Secondary Structure
T	Signal Transduction
U	Intracellular trafficking and secretion
Y	Nuclear structure
Z	Cytoskeleton
R	General Functional Prediction only
S	Function Unknown

Due to a large number of known proteins, their manual classification is practically unmanageable in real time. Biologists play a major role in protein classification; however, in order to make an effective classification of all available proteins along with various tools and techniques, they need assistance of a computer scientist. The cooperation between computer scientists and biologists resulted in a number of new branches which investigate the biological data, such as bioinformatics.

An *n*-gram, as introduced by Shannon in 1948 [7], is a subsequence of the length *n* of a sequence of the length *m* ($m \geq n$) over the given alphabet. An *n*-gram analysis calculates the probability of an *n*-gram occurrence in a sequence, the relative frequency of the different *n*-grams occurrence, or more sophisticated statistical properties of *n*-grams. Various *n*-gram based analyses were used for text compression [8], automatic text categorization and prediction [9], language identification [10], author attribution [11], etc.

Protein and genome sequences can be considered the sequences of symbols which can be the object of an *n*-gram analysis. Faouzi et al. [12] showed the hierarchical *n*-grams extraction approach in protein classifications which was consistent with the biological domain knowledge. Osmanbeyoglu et al. analysed 970 microbial organisms and extracted the *n*-grams which were overrepresented in one organism and very rarely presented in other organisms, hence they could be used as proteome signatures [13]. In their papers Ganapathiraju et al. showed [14-16] that biological sequences could be processed using the same methods as natural language. They set up an analogy between multiple genome sequences and raw texts, function of proteins and meaning of words, complex interaction in biological systems and document topics. Different variants of *n*-gram-based methods were successfully applied to measure sequence similarity and reconstruction of phylogenetic trees without sequence alignment [17], classification of unknown proteins [18], comparison of properties of coding and noncoding regions in genomes [19], linguistic complexity of genomic sequences [20], classification and unsupervised hierarchical clustering of genome sequences [21], promoter recognition problem [22], characterization of genomic islands [23, 24], etc.

In this paper, the analysis of the *n*-gram structure of amino acid sequences of proteins, associated with the category from COG collection was carried out. Biologists assign certain COG category to the protein based on existing specific amino acids patterns, which can often be 7 or more AA long. The problem of formal algorithm to recognize new characteristic patterns is still open. As a solution, BLAST or other sequence comparison methods are applied on proteins to find their similarity with the existing ones already classified in some COG categories. The following study was aimed at providing a method for classification of proteins, based on amino acid *n*-grams according to COG categories. This could be done by extracting *n*-grams characterized by the proteins belonging to a specific COG category. The extraction was established by using mathematical model based on functions and equations in Boolean algebra. The obtained results can be used as supplementary method for classification of new proteins by function.

2 The proposed method

A sequential pattern is a relatively short sequence that occurs significantly more (or less) in a given set of sequences. Sequential analysis, i.e. the determination of sequential patterns is developed as a separate research direction of data mining. Most sequential pattern models belong to one of the following four categories: frequent patterns, periodic patterns, statistically significant patterns, and approximate patterns [25]. This paper includes statistically significant patterns (those that occurred in a given set of sequences more than expected) and approximate patterns (the ones that did not appear in full composition, but with minor changes). The aim of this study was to find sequential pattern models that characterized proteins belonging to the same COG category.

2.1 The idea of method construction

Generally, a sequence is an ordered collection of elements X = '$x_1x_2...x_{k-1}x_k$' such that each element x_i belongs to the set A. The set A is called the alphabet. An n-gram is a segment of n consecutive symbols of the sequence X of length k ($n \leq k$) which is defined over a given alphabet A of cardinality $|A| = r$. The number of different n-grams L of alphabet A is equal to the number of variations with repetition of n different elements when r by r elements are taken, i.e. $L = r^n$. There are $k - n + 1$ overlapping n-grams in the sequence X with length k ('$x_1...x_{n-1}x_n$', '$x_2...x_nx_{n+1}$', ..., '$x_{n-k+1}...x_{k-1}x_k$').

Proteins can be considered the sequences which are defined over 20 letters (amino acid) alphabet A = {A, E, Q, D, N, L, G, K, S, V, R, T, P, I, M, F, Y, C, W, H}. An analysis of overlapping n-grams of all proteins in the dataset should be done to determine the n-grams that characterize proteins in a certain COG category. The main idea in finding characteristic n-grams is to extract overlapping n-grams a_i of certain length n from each protein, count their occurrences and determine significant n-grams for each COG category. The significance of the occurrence of a sequence a_i in the COG category COG_k is expressed by two measures: support (s) - the percentage of protein in the category COG_k containing a_i, and confidence (c) - the percentage of proteins which belong to the category COG_k and contain sequence a_i. Among the generated models (for a_i and n), only those with support and confidence greater than predefined values s and c are taken for testing phase.

Although not so complicated, the proposed method suffers from one serious shortcoming. The number of different n-grams of length 1, 2, ..., n is 20^1, 20^2, ..., 20^n, respectively. Data processing and storing very quickly overcome technical abilities, take too much time, but without reliability that the process will be effective for a large set of proteins in the model set. For this reason, the transformations on n-grams were made. A transformation can be considered a specific kind of dimensional reduction which enables determination of characteristics of overlapping n-grams. A complete description of the method will be given according to the previously mentioned 20 amino acids, but it can be easily extended to include additional amino acids which can be found in proteins, e.g. selenocysteine and pyrrolysine.

2.2 Method description and implementation

Let A be a set of amino acids. Each subset of the alphabet A can be represented as a sequence that is sorted according to the defined amino acid order. For instance, {A, E} and {E, A} represent the same subset of A and they will be presented with a sequence 'AE'. Throughout the text, such a sequence is referred to as a basic amino acid sequence (BAA sequence). There is a bijection between basic amino acid sequences and elements of the powerset of *A*, further denoted as *P(A).* The cardinality of the power set of *A* is $|P(A)| = 2^{|A|}$; thus the cardinality of the set of basic amino acid sequences is 2^{20}.

Let $B = P(A)$ be the powerset of *A.* Then structure (B, ∩, ∪, ', ∅, **1**) is Boolean algebra, where ∅ (i.e. the empty BAA sequence) and **1** (i.e. the BAA sequence of length 20 which contains all amino acids) are, respectively, the bottom and top elements. The result of the operation union (∪) applied on two BAA sequences is the BAA sequence which contains amino acids from both operands. The result of the operation intersection (∩) applied on two BAA sequences is the BAA sequence which contains amino acids that appear in both operands. Complement (') of a BAA sequence gives the sequence consisting of all amino acids which are not included in the operand [26].

The algorithm

1. The first step in model construction is defining mapping *n*-grams in the BAA sequences. For each *n*-gram a_i, sequence $b_j = BAA(a_i)$ is defined as BAA sequence of length from 1 to 20, depending on the number of different amino acids which are contained in a_i. For example, *n*-gram 'MIKRADF' is mapped to BAA sequence 'ADFIKMR', while *n*-gram 'QQSQSNNHHT' is mapped to BAA sequence 'HNQST'. Applying the function BAA reduces the number of possible *n*-grams to the number of possible subsets of the set *A* of *n, n*-1, …, 1 elements:

$$L_{\mathrm{BAA}} = \binom{20}{n} + \binom{20}{n-1} + \binom{20}{n-2} + \cdots + \binom{20}{1} \qquad (1)$$

because each set of 20 elements has exactly $\binom{20}{r} = \frac{20!}{(20-\mathrm{r})!\mathrm{r}!}$ different subsets of r (r = 1, …, *n*) elements. For example, for $n = 3$ the number of different trigrams is 8000, while the number of different BAA sequences is

$$\binom{20}{3}+\binom{20}{2}+\binom{20}{1}=1350.$$

2. Set initial value of *n* (*n*-gram length) to 1.

3. Increase *n* by 1 and fix it until the end of the complete process. For each protein *P,* a set of overlapping *n*-grams $O_n = \{a_i \mid a_i \in P, |a_i| = n\}$ is extracted. Based on the extracted set we calculate a set of corresponding BAA sequences $B_n = \{b_j \mid b_j = BAA(a_i), a_i \in O_n\}$. The process is completed with construction of ordered 4-typle $T_{jkn} = (n, COG_k, b_j, \#a_i)$, where *n* is the length of processed *n*-grams, COG_k is *COG category,* $b_j \in B_n$ is calculated BAA sequence, and $\#a_i$ is the number of a_i sequences such that exists protein *P* and $(b_j = BAA(a_i) \wedge a_i \in P \wedge P \in COG_k)$. The significance of the occurrence of a BAA sequence b_j in the COG category COG_k is expressed by support and confidence related to predefined values *s* and *c*. BAA sequences b_j that are found to be significant represent sequential pattern which can be characteristics of COG category COG_k.

4. The fourth step covers selecting sequential patterns b_j that are eligible to be characteristic pattern for COG category COG_k. From the set of constructed 4-tuples T_{jkn} , only those where $\#a_i$ have ordered pair *(support, confidence)* greater than or equal to the pair of predefined values (*s,c*) were selected for further processing. For COG category COG_k a vector of *l* characteristics sequential patterns is determined by taking first *l* BAA sequences b_i from ordered list of 4-tuples T_{jkn} related to COG_k, where the list is ordered in decreasing order related to pair (support, confidence).

Among selected characteristic sequential patterns, those with confidence equal to 100% have a special role. Such patterns are called *descriptors* of *n*-grams of length *n* for specific COG category. Formally, descriptor d_{jkn} is defined by equation $d_{jkn} = b_j$ where exists $T_{jkn} = (n, COG_k, b_j, \#a_i)$ and $c(b_j) = 100\%$, $s(b_j) \geq s_{def}$, while corresponding descriptor set D_{jkn} is defined as $D_{jkn} = \{a_i \mid b_j = BAA(a_i), d_{jkn} = b_j\}$, e.g. D_{jkn} is a set of *n*-grams that maps to b_j (d_{jkn}).

5. If maximum *n*-gram length is not achieved, go to the step 2. Otherwise, go to the next step.

6. Take all descriptors found in step 4. Construct the classification model based on the fact that any individual descriptor unambiguously determines unique COG category. Based on this feature, if some *n*-grams from new protein with unknown COG category is mapped to descriptor d_{jkn}, such protein can be classified in COG category COG_k.

Models of sequential patterns do not have to be completely determined. Moreover, they can be more general, with approximate patterns. The generalization of sequential patterns in our case is the replacement of specific amino acids from the BAA sequences with wildcard (variable which can represent any amino acid). Combined sequential patterns (described above) which consist of BAA sequence b_j with mark '*' as wildcard in *n*-grams whose (*n-1*) character is mapped in the BAA sequence, and a single character in the position of '*' can be any amino acid (wildcard '*'). These models will be referred as *approximate descriptors* of *n*-grams of length *n*. An *n*-gram described by descriptors (approximate descriptors) will be here in referred to as a *characteristic n-gram*.

Instead of applying the process described in step 4, determining the set of descriptors related to specific COG category can be done by using a solution from Boolean algebra. Let b_j be a BAA sequence extracted from the protein that belongs to the COG category COG_k, e_i be a BAA sequence extracted from the proteins belonging to the other COG category (but not to COG category COG_k), and let x be a descriptor or approximate descriptor of *n*-grams in the COG category COG_k. Then x is contained in some b_j and there is no e_i which contains x. The previous sentence is equivalent to $b_j' \cap x = 0$ for some b_j and $e_i' \cap x \neq \emptyset$ holds for each e_i. Then the descriptor x can be found as the solution of the generalized system of Boolean equations:

$$(b_1' \cap x = 0 \vee \cdots \vee b_{k1}' \cap x = 0) \wedge (e_1' \cap x \neq 0 \wedge \cdots \wedge e_{k2}' \cap x \neq 0). \qquad (2)$$

The solutions of this system represent BAA sequences which describe *n*-grams that are contained only in proteins from COG category COG_k. The problem of solving generalized systems of Boolean equations (systems which are built using conjunctions and disjunctions of Boolean equations and Boolean inequations) still stays open for further discussion. Some results of solving these systems were summarized in Rudeanu's books [26, 27]. Banković gave all the solutions related to Boolean inequations [28], systems of a Boolean equation and a Boolean inequation [29], and systems of two Boolean inequations [30]. Marovac has considered systems of *k* Boolean inequations [31] and disjunction of Boolean equations [32]. The results obtained in these papers were used in the implementation of the program for extracting descriptors and approximate descriptors.

Implementation

The previously described algorithm for extracting BAAs and determining descriptors and approximate descriptors was implemented as a computer program written in C language. BAA sequences are represented as binary words of length 20, e.g. by 20-digits binary number. Each position in the binary word of length 20, from the largest to the smallest weight, presents the one amino acid from the set *A* (1-presence, 0-absence). Thus A = 2^0, E = 2^1, Q = 2^2, D = 2^3, ..., W = 2^{18}, H = 2^{19}. Boolean operation of union and intersection of BAA sequences has been implemented as bitwise 'or' (|) and 'and' (&) operation in C language. Complement of a BAA sequence is obtained by subtracting the BAA sequence from binary equivalent of (2^{20} - 1) which is binary equivalent of **1.** The empty set ∅ is represented by zero.

The quality of constructed classification model is evaluated based on the counts of test records correctly and incorrectly predicted by the model [33] obtained from confusion matrix. Confusion matrix consists of four values: TP (true positive), TN (true negative), FP (false positive) and FN (false negative). The program calculates three most commonly used performance measures:

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \qquad (3)$$

which is the proportion count of correctly associated and non-associated proteins to appropriate COG category in relation to the testing set of proteins. Also, precision and recall are calculated for each category. They are defined with:

$$precision = \frac{TP}{TP+FP}, \qquad recall = \frac{TP}{TP+FN}.$$

Based on the defined model we also implemented a predictor for classification of (previously unclassified) proteins to specific COG category. A protein is classified to some COG category COG_k if it contains the appropriate characteristic *n*-grams for this category in a quantity higher than the threshold *h* – predefined number that can be supplied as an input parameter.

Advantages and disadvantages of proposed method

The advantages of the proposed model in comparison to the previously used methods of protein classification, according to COG categories, are simpler creation of models in the case of the target class of organisms, shorter time of realization and decreased usage of computer resources. The proposed model of classification requires the time proportional to the product of the length of sequence that has to be classified and the total length of a set of characteristic n-grams (whereas n∈{5,6,...,10}) that is significantly less than the processing time of BLAST method, for example, which is proportional to product of the length of sequence that has to be classified and the total length of the set of already classified sequences.

Table 2: The input data sets (trening and testing)

Species of bacteria class Chlamydiales			
NC_000117	Chlamydia trachomatis D/UW-3/CX	41.3	1042519
NC_000922	Chlamydophila pneumoniae CWL029	40.57	1230230
NC_002179	Chlamydophila pneumoniae AR39	40.57	1229853
NC_002182	Chlamydia muridarum str. Nigg	40.3	7501
NC_002491	Chlamydophila pneumoniae J138	40.58	1226565
NC_002620	Chlamydia muridarum str. Nigg	40.3	1072950
NC_003361	Chlamydophila caviae GPIC	39.18	1173390
NC_004552	Chlamydophila abortus S26/3	39.86	1144377
NC_004720	Chlamydophila caviae GPIC	39.18	7966
NC_005043	Chlamydophila pneumoniae TW-183	40.57	1225935
NC_005861	Candidatus Protochlamydia amoebophila UWE25	34.71	2414465
NC_007429	Chlamydia trachomatis A/HAR-13	41.26	1044459
NC_007430	Chlamydia trachomatis A/HAR-13	41.26	7510
NC_007899	Chlamydophila felis Fe/C-56	39.34	1166239
NC_007900	Chlamydophila felis Fe/C-56	39.34	7552
NC_010280	Chlamydia trachomatis L2b/UCH-1/proctitis	41.32	1038863
NC_010287	Chlamydia trachomatis 434/Bu	41.32	1038842
NC_012686	Chlamydia trachomatis B/Jali20/OT	41.29	1044352
NC_012687	Chlamydia trachomatis B/TZ1A828/OT	41.3	1044282
NC_014225	Waddlia chondrophila WSU 86-1044	43.73	2116312
Test set			
NCBI ID	Species	content	
NC_014226	Waddlia chondrophila WSU 86-1044	43.73	15593
NC_015217	Chlamydia psittaci 6BC	39.02	7553
NC_015408	Chlamydophila pecorum E58	41.07	1106197
NC_015470	Chlamydia psittaci 6BC	39.02	1171660
NC_015702	Parachlamydia acanthamoebae UV-7	39.02	3072383
NC_015710	Simkania negevensis Z	41.6	132038
NC_015713	Simkania negevensis Z	41.6	2496337
NC_015744	Chlamydia trachomatis L2c	41.32	1038313

Besides the advantages, the model for functional classification also has some disadvantages. The accuracy of models depends on the set of input data. For increased accuracy and reaching precision of widely used methods is generally necessary to take the data from each family (organisms). This process requires the approach to huge data bases and it is not easy to be efficiently carried out in the local environment due to restricted computer resources. For that reason, the percentage of unclassified proteins that could be classified with the current version of the method is still unsatisfactory and without optimization the proposed method could not be used as the basic, but could be used as an additional method of protein classification according to COG categories.

3 Results and Discussion

In this section we evaluated the performances of the proposed approach. The method was applied on the proteins with determined COG categories from the set of 28 organisms from *Chlamydiales* phylum. The data were downloaded from http://ftp.ncbi.nlm.nih.gov/genomes/Bacteria/ (state on 1 October 2014). The input data were divided into two sets for the classification (see Table 2): the training data used for a model construction (20 organisms) and the testing data which were used to verify the correctness of the model (8 organisms). The number of protein by COGs in the model and test set are shown in Table 3 and Table 4.

Table 3: Distribution of proteins per COG category in the training set

COG category	A	B	C	D	E	F	G	H	I	J	K	L
The number of proteins	8	11	499	167	600	218	402	417	371	1199	278	761
COG category	M	N	O	P	Q	R	S	T	U	V	W	
The number of proteins	581	145	409	321	75	825	505	211	391	49	2	

Table 4: Distribution of proteins per COG category in the test set

COG category	A	B	C	D	E	F	G	H	I	J	K	L
The number of proteins	0	6	311	71	390	143	294	251	221	631	177	405
COG category	M	N	O	P	Q	R	S	T	U	V	W	
The number of proteins	398	84	241	198	84	623	347	152	209	65	398	

Applying the algorithm described in the previous section we successfully extracted *descriptors* of *n*-grams of length $n \in \{5, \ldots, 10\}$ and *approximate descriptors* of *n*-grams of length $n \in \{7, \ldots, 10\}$ for each COG category. For shorter *n*-grams, there was no BAA sequence that belonged to only one COG category.

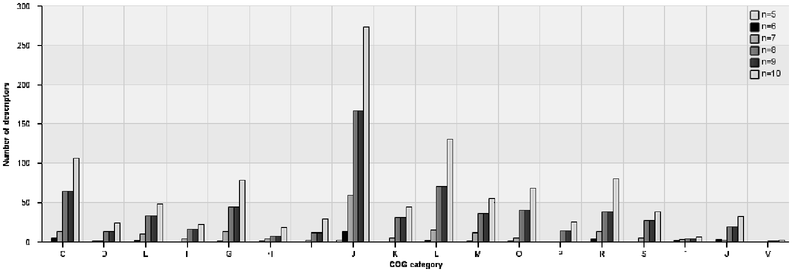

Fig 1. The number of extracted descriptors per COG category

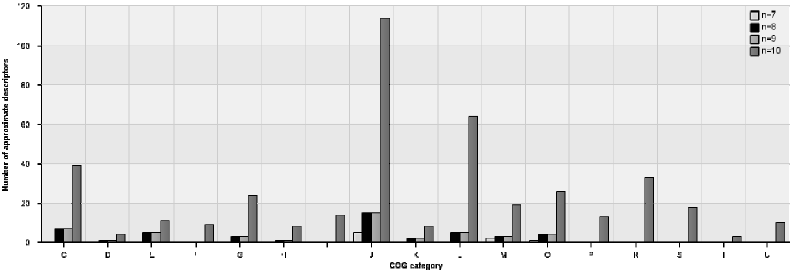

Fig 2. The number of extracted approximated descriptors per COG category

The number of selected descriptors increased with the increase of the length of *n*-grams which mapped to appropriate descriptor (see Fig. 1 and Fig. 2). This evidence confirmed the expectations, because it experimentally showed that longer patterns produced more relevant results of COG assignment.

Table 5: The precision of protein classification in test set based on descriptors

n-gram length	*precision* (%)	#characteristic *n*-grams in proteins from an improper category	#characteristic *n*-grams in proteins from the appropriate category
n=5	77.77	4	14
n=6	74.77	56	166
n=7	75.77	245	766
n=8	80.81	406	1710
n=9	80.72	672	2814
n=10	81.36	1108	4835

Table 6: The precision of protein classification in test set based on approximate descriptors

n-gram length	*precision* (%)	#characteristic *n*-grams in proteins from an improper category	#characteristic *n*-grams in proteins from the appropriate category
n=7	79.16	10	38
n=8	71.95	99	254
n=9	72.73	303	808
n=10	72.83	676	1812

The model constructed on a determined set of descriptors was applied on the proteins from test set. The precision of protein classification to the COG category, which was associated with the appropriate descriptor (approximate descriptor), was greater than 71% for all lengths of *n*-grams (see Table 5 and Table 6).

Table 7 shows some of the descriptors and approximate descriptors of *n*-grams of length 10 of training set, which were also found to be descriptors of test set.

Table 7: Some of descriptors and approximate descriptors of *n*-grams for both training and test sets

cog category	Descriptors	Test set			Training set		
		#*n*-grams	#	% of genes	#*n*-grams	#genes	%genes
C	AQSVCH	13	5	2.0%	44	16	3.2%
C	ANLGPMFYC	12	4	1.0%	58	15	3.0%
E	AEDNVICW	12	3	1.0%	56	14	2.3%
F	AENLGYCWH	14	5	1.2%	42	14	6.4%
G	AGFYW	12	3	0.4%	29	10	2.5%
J	ADGVPYCH	20	5	3.4%	60	15	1.3%
J	ADGVPIFYW	14	5	1.2%	45	16	1.3%
J	NLGKVRMFY	13	6	2.7%	39	12	1.0%
R	ADGTPFYC	12	4	1.2%	46	16	1.9%
T	AQDGSTYH	12	4	1.2%	48	16	

category	approximate descriptors	Test set			Training set		
		#*n*-grams	#genes	% of genes	#*n*-grams	#genes	%genes
	ANLGPMFYC,*	12	4	0.7%	62	16	
E	AERPIFYWH,*	8	4	1.2%	32	16	2.7%
F	AENLGYCWH,*	14	5	0.7%	42	14	6.4%
F	QNGKRPIMW,*	10	4	0.9%	42	14	6.4%
G	ENGRPIYCW,*	9	3	0.7%	52	13	3.2%
L	AEQDNLGMW,*	10	5	3.4%	32	16	2.1%
L	AELGVRMWH,*	9	5	0.6%	36	21	2.8%
L	AQKVRTCWH,*	8	5	1.2%	24	12	1.6%
O	AKSTPIYCW,*	10	5	1.6%	32	16	3.9%
R	QLGKSRMWH,*	9	3	1.0%	30	10	1.2%

It may be noted that some of the descriptors in Table 7 are subsets of approximate descriptors. For example, d_1 = 'AENLGYCWH' is contained in an approximate descriptor d_2 = 'AENLGYCWH, *'. The descriptor d_1 describes all *n*-grams of length 10 which contains all the amino acids from the group A_1 = {'A', 'E', 'S', 'L', 'G', 'Y', 'C', 'W', 'H'}, and no other amino acids. The approximate descriptor d_2 describes all *n*-grams of length 10 which contain all the amino acid from the group A_1 = {'A', 'E', 'S', 'L', 'G', 'Y', 'C', 'W', 'H'}, and only one position in *n*-grams that can include some another amino acid. This is important and shows the existence of characteristic COG patterns with specific amino acids that need not necessarily be continuous and that are resistant to insertions/deletions of amino acids.

The set of discovered predictors are used for the classification of proteins form test set. Table 8 shows the results obtained from a confusion matrix for each COG category for which the set of descriptors is determined. It can be seen that accuracy and precision of prediction are very high, and that the recall is higher in COG categories with a large number of proteins. This can be a consequence of an insufficient number of proteins in some COGs in the training set to generate classification rules in model set.

Table 8: The results obtained from a confusion matrix for each COG category in the test set

COG category	Accuracy	Precision	Recall
C	95%	86%	20%
D	99%	100%	14%
E	93%	93%	7%
F	98%	100%	10%
G	95%	93%	14%
H	95%	100%	2%
I	96%	100%	5%
J	91%	97%	27%
K	97%	100%	12%
L	94%	93%	17%
M	93%	93%	6%
O	96%	98%	20%
P	96%	100%	4%
R	89%	95%	6%
S	94%	100%	7%
T	97%	100%	6%
U	96%	100%	9%

The accuracy of prediction is directly related to the threshold *h* (the minimum number of characteristic *n*-grams that occur in a protein that is necessary to classify the protein in the appropriate COG category). Table 9 shows the results of the prediction cumulatively for all COG categories depending on the threshold. It can be seen that the precision increases with the threshold while the recall decreases.

Table 9: Results prediction of COG categories depending on the set threshold

Threshold	#matches with biologically determined category class	#mismatches with biologically determined category class	Precision(%)	Recall(%)
5	596	28	96%	11%
6	507	15	97%	10%
7	449	15	97%	8%
8	390	6	98%	7%
9	346	5	99%	7%
10	269	0	100%	5%

For all threshold values precision is greater than 96%. Precision reaches 100% when the threshold is 10, but the number of classified proteins is twice smaller.

4 Conclusion

The classification of proteins in the COG groups is important for better understanding of biological processes. The traditional approach for solving the problem is based on extensive search for in-advance known similarity patterns. In this paper we have introduced the novel method for extracting a set of *n*-grams whose occurrence is specific for proteins which belong to a certain COG categories, and the model for classification of proteins based on extracted *n*-grams. The main advantages are that the novel method requires less memory resources and processing time than classical ones, and that the model for classification of proteins determines the COG category of a protein without its comparison with other proteins. Also, by using the proposed method, potentially new and previously unknown characteristics patterns can be discovered. According to COG categories, the proposed model for classification of a protein has high precision and accuracy. The presented results can be used as an additional approach for classification of new proteins according COG categories. Our future plans are to apply this approach to larger set of organisms in order to increase support and prove hypothesis that characteristic *n*-grams of a certain COG category are independent of the organism phylum.

Acknowledgements: The work presented has been financially supported by the Ministry of Education, Science and Technological Development, Republic of Serbia, Projects No. III-44007 and 174021.

References

[1] R. L. Tatusov, M. Y. Galperin, D. A. Natale, E. V. Koonin, The COG database: a tool for genome-scale analysis of protein functions and evolution, *Nucleic Acids Res.* **28** (2000) 33–36.

[2] E. V. Koonin, Orthologs, paralogs and evolutionary genomics, *Ann. Rev. Gen.* **39** (2005) 309–338.

[3] C. A. Ouzounis, R. M. Coulson, A. J. Enright, E. V. Koonin, J. B. Pereira–Leal, Classification schemes for protein structure and function, *Nature Rev. Gen.* **4** (2003) 508–519.

[4] R. L. Tatusov, E. V. Koonin, D. J. Lipman, A genomic perspective on protein families, *Science* **278** (1997) 631–637.

[5] E. V. Koonin, The clusters of orthologous groups (COGs) database: Phylogenetic classification of proteins from complete genomes, in: J. McEntyre, J. Ostell (Eds.), *The NCBI Handbook*, National Center for Biotechnology Information, 2002, http://www.ncbi.nlm.nih.gov/books/NBK21090/.

[6] R. L. Tatusov, N. D. Fedorova, J. D. Jackson, A. R. Jacobs, B. Kiryutin, E. V. Koonin, D. M. Krylov, R. Mazumder, S. L. Mekhedov, A. N. Nikolskaya, B. S. Rao, S. Smirnov, A. V. Sverdlov, S. Vasudevan, Y. I. Wolf, J. J. Yin, D. A. Natale, The COG database: an updated version includes eukaryotes, *BMC Bioinformatics*, **4** (2003) #41 (14 pp.).

[7] C. A. Shannon, Mathematical theory of communication, *Bell Syst. Tech. J.* **27** (1948) 379–423.

[8] J. Wisniewski, Effective text compression with simultaneous digram and trigram encoding, *J. Inf. Sci.* **13** (1987) 159–164.

[9] W. B. Cavnar, J. M. Trenkle, n-Gram-based text categorization, Symposium on Document Analysis and Information Retrieval, Univ. Nevada, Las Vegas, 1994.

[10] J. C. Schmitt, Trigram–based method of language identification, *US Patent* **5** (1991) 62–143.

[11] V. Kešelj, F. Peng, N. Cercone, C. Thomas, n-Gram-based author profiles for authorship attribution, Proceedings of the Conference Pacific Association for Computational Linguistics, Halifax, Canada, 2003.

[12] M. Faouzi, R. Ricco, E. A. Mourad, A hierarchical n-grams extraction approach for classification problem, in: E. Damian, K. Yetongnon, R. Chbeir, A. Dipanda (Eds.), *Advanced Internet Based Systems and Applications*, Springer, Berlin, 2009, pp. 211-222.

[13] H. U. Osmanbeyoglu, M. K. Ganapathiraju, n-Gram analysis of 970 microbial organisms reveals presence of biological language models, *Bioinformatics* **12** (2011) #12 (12 pp.).

[14] M. Ganapathiraju, N. Balakrishnan, R. Reddy, J. Klein–Seetharaman, Computational biology and language, in: Y. Cai (Ed.), *Ambient Intelligence for Scientific Discovery*, Springer, Berlin, 2005, pp. 25-47.

[15] A. Poddar, N. Chandra, M. Ganapathiraju, K. Sekar, J. Klein–Seetharaman, R. Reddy, N. Balakrishnan, Evolutionary insights from suffix array–based genome sequence analysis, *J. Biosci.* **32** (2007) 871-881.

[16] M. Ganapathiraju, A. Mitchell, M. Thahir, K. Motwani, S. Ananthasubramanian, Suite of tools for statistical n-gram language modeling for pattern mining in whole genome sequences, *J. Bioinform. Comput. Biol.* **10** (2012) #1250016 (22 pp.).

[17] E. Cohen, B. Chor, Detecting phylogenetic signals in eukaryotic whole genome sequences, *J. Comput. Biol.* **19** (2012) 945-956.

[18] B. Cheng, J. Carbonell, J. Klein-Seetharama, Protein classification based on text document classification techniques, *Proteins* **58** (2005) 955–970.

[19] R. N. Mantegna, S. V. Buldyrev, A. L. Goldberger, S. Havlin, C. K. Peng, M. Simons, H. E. Stanley, Systematic analysis of coding and noncoding DNA sequences using methods of statistical linguistics, *Phys. Rev. E* **52** (1995) 2939–2950.

[20] O. G. Troyanskaya, O. Arbell, Y. Koren, G. M. Landau, A. Bolshoy, Sequence complexity profiles of prokaryotic genomic sequences: a fast algorithm for calculating linguistic complexity, *Bioinformatics* **18** (2002) 679-688.

[21] A. Tomović, P. Janičić, V. Kešelj, n-Gram-based classification and unsupervised hierarchical clustering of genome sequences, *Comput. Meth. Programs Biomed.* **81** (2006) 137-153.

[22] T. Rani, S. Bhavani, R. Bapi, Analysis of E.coli promoter recognition problem in dinucleotide feature space, *Bioinformatics* **23** (2007) 582-588.

[23] N. Mitić, G. Pavlović-Lazetić, M. Beljanski, Could n-gram analysis contribute to genomic island determination, *J. Biomed. Inf.* **41** (2008) 936-943.

[24] G. Pavlović-Lazetić, N. Mitic, M. Beljanski, n-Gram characterization of genomic islands in bacterial genomes, *Comput. Meth. Programs Biomed.* **93** (2009) 241-256.

[25] W. Wang, J. Yang, *Mining Sequential Patterns from Large Data Sets*, Springer, Berlin, 2005.

[26] S. Rudeanu, *Boolean Functions and Equations*, North-Holland, Amsterdam, 1974.

[27] S. Rudeanu, *Lattice Functions and Equations*, Springer, Berlin, 2001.

[28] D. Banković, Boolean inequations, *Discr. Math.* **307** (2007) 750-755.

[29] D. Banković, Boolean equations and Boolean inequations, *J. Multiple-Val. Logic Soft Comput.* **16** (2010) 189-196.

[30] D. Banković, U. Marovac, System of two Boolean inequations, *J. Multiple-Val. Logic Soft Comput.* **24** (2015) 521-528.

[31] U. Marovac, Systems of k Boolean inequations, *J. Multiple-Val. Logic Soft Comput.* **25** (2015) in press.

[32] U. Marovac, Disjunction of Boolean equations, *Publ. Inst. Math. (Beograd)* accepted.

[33] P. N. Tan, M. Steinbach, V. Kumar, *Introduction to Data Mining,* Pearson Education, 2006.