A Structure Based Multiple Instance Learning Approach for Bacterial Ionizing Radiation Resistance Prediction

Manel Zoghlami, Sabeur Aridhi, Mondher Maddouri, Engelbert Nguifo

Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 00 (2019) 000–000

Procedia Computer Science

www.elsevier.com/locate/procedia

23rd International Conference on Knowledge-Based and Intelligent Information & Engineering Systems

A Structure Based Multiple Instance Learning Approach for Bacterial Ionizing Radiation Resistance Prediction

Manel Zoghlami[a,b,*], Sabeur Aridhi[c], Mondher Maddouri[d], Engelbert Mephu Nguifo[b]

[a] *University Clermont Auvergne, CNRS, LIMOS, BP 10125, 63173 Clermont Ferrand, France*
[b] *University of Tunis El Manar, Faculty of sciences of Tunis, LIPAH, Tunis, Tunisia*
[c] *University of Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France*
[d] *University of Jeddah, College of Business, PB 80327, 21589 Jeddah, KSA*

Abstract

Ionizing-radiation-resistant bacteria (IRRB) could be used for bioremediation of radioactive wastes and in the therapeutic industry. Limited computational works are available for the prediction of bacterial ionizing radiation resistance (IRR). In this work, we present ABClass, an *in silico* approach that predicts if an unknown bacterium belongs to IRRB or ionizing-radiation-sensitive bacteria (IRSB). This approach is based on a multiple instance learning (MIL) formulation of the IRR prediction problem. It takes into account the relation between semantically related instances across bags. In ABClass, a preprocessing step is performed in order to extract substructures/motifs from each set of related sequences. These motifs are then used as attributes to construct a vector representation for each set of sequences. In order to compute partial prediction results, a discriminative classifier is applied to each sequence of the unknown bag and its correspondent related sequences in the learning dataset. Finally, an aggregation method is applied to generate the final result. The algorithm provides good overall accuracy rates. ABClass can be downloaded at the following link: `http://homepages.loria.fr/SAridhi/software/MIL/`.

Keywords: Ionizing-radiation-resistant bacteria; Multiple instance learning; Bacterial phenotype prediction

1. Introduction

Ionizing-radiation-resistant bacteria (IRRB) are important in biotechnology. These organisms are being engineered for *in situ* bioremediation of radioactive wastes [1] and could also be used in the therapeutic industry [2]. Several *in vitro* and *in silico* works studied the causes of the high resistance of IRRB to ionizing radiation and tried to determine

* Corresponding author. Tel.: +33 (0)3 54 95 86 46.
E-mail address: manel.zoghlami@etu.uca.fr

peculiar features in the IRRB genomes. However, limited computational works are provided for the prediction of bacterial onizing radiation resistance (IRR) [3] [4].

Predicting if a bacterium belongs to IRRB using *in vitro* experiments is not an easy task since it requires a big effort and a time consuming lab work. In this work, we aim to use machine learning in order to perform the bacterial IRR prediction task. This approach formulates the problem of predicting bacterial IRR as a Multiple Instance Learning (MIL) problem. The standard supervised learning task deals with data that consist on a set of objects/examples, where each object is associated with a label. However, an MIL task deals with data that consists of a set of *bags* where each bag is an unordered set of examples. In an MIL context, each example is called an *instance*. Labels are assigned to bags rather than individual instances, i.e., we do not know the label of each instance inside a bag. Several MIL algorithms have been proposed. A review of MIL approaches and a comparative study could be found in [5] and [6]. According to the proposed MIL formalization, bacteria represent the bags and protein sequences represent the instances. In particular, each protein sequence may differ from a bacterium to another, e.g., each bag contains the protein named *Endonuclease III*, but it is expressed differently from one bag to another : these are called orthologous proteins [7]. To learn the label of an unknown bacterium, comparing a random couple of sequences makes no sense, it is rather better to compare the protein sequences that have a functional relationship/dependency: the orthologous proteins. Hence, this work deals with an MIL problem in sequence data that present structural dependencies between instances of different bags. The problem has the following three criteria: (1) the instances inside the bags are sequences so we have to deal with data representation, (2) the instances may have dependencies across the bags and (3) all the instances inside a bag contribute to define the bag's label. The standard MIL assumption states that a bag is positive if at least one of its instances is positive while in every negative bag all of the instances are negative. This is not guaranteed to hold in some domains so alternative assumptions are considered [8]. Particularly, the standard assumption is not suitable for the above presented bacterial IRR prediction problem since one positive instance is not sufficient to classify a bag as positive. We opt for *the collective assumption* that considers that all of the instances contribute to the bag's label [5] [8].

In a previous work [4], we proposed MIL-ALIGN, a tool that aims to solve the IRR prediction problem based on an MIL formulation. As far as we know, MIL-ALIGN is the only bioinformatics tool that has been proposed to predict if a bacterium belongs to IRRB or ionizing-radiation-sensitive bacteria (IRSB). MIL-ALIGN computes the alignment scores between instances and then apply an aggregation method to compute a final prediction result. In this work, we further explain the formulation of the investigated MIL problem and we present the ABClass tool that uses a motif-based approach to predict the label of an unknown bacterium. ABClass performs first a preprocessing step of the input sequences. This step consists in extracting motifs from the set of sequences. These motifs will be used as attributes to construct a binary table where each row corresponds to a sequence. Then a discriminative classifier is applied to the sequences of an unknown bag in order to assign its label. We applied the proposed approach to the problem of prediction of IRR in bacteria using MIL techniques introduced and described in [4]. We describe an implementation of our algorithm and we present an experimental study of the proposed approach.

The remainder of this paper is organized as follows. Section 2 defines the problem of MIL for sequence data and provides a running example followed by a short presentation of MIL-ALIGN. In Section 3, we describe the proposed MIL-based approach for IRR prediction named ABClass. In Section 4, we describe our experimental environment and we discuss the obtained results. Concluding points make the body of Section 5.

2. Background

In this section, we describe the terminology and the MIL formulation of the problem. Then, we introduce a simple use case that serves as a running example throughout this paper. Finally, we provide a short presentation of MIL-ALIGN.

2.1. Problem Formulation

We denote Σ an *alphabet* defined as a finite set of characters or symbols. A simple symbolic sequence is defined as an ordered list of symbols from Σ. Let DB be a learning database that contains a set of n labeled bags $DB = \{(B_i, Y_i), i = 1, 2 \ldots, n\}$ where $Y_i = \{-1, 1\}$ is the label of the bag B_i. Instances in B_i are sequences and are denoted

by B_{ij}. Formally $B_i = \{B_{ij}, j = 1, 2 \ldots, m_{Bi}\}$, where m_{Bi} is the total number of instances in the bag B_i. We note that the bags do not contain the same number of instances. The problem investigated in this work is to learn a multiple instance classifier from DB. Given a query bag $Q = \{Q_k, k = 1, 2 \ldots, q\}$, where q is the total number of instances in Q, the classifier should use sequential data in this bag and in each bag of DB in order to predict the label of Q.

We note that there is a relation $\mathfrak{R}$ between instances of different bags denoted *the across bag sequences relation* and it is defined according to the application domain. To represent this relation, we opt for an index representation. We note that this notation does not mean that instances are ordered. In fact, a preprocessing step assigns an index number to the instances inside each bag according to the following notation: each instance B_{ij} of a bag B_i is related by $\mathfrak{R}$ to the instance B_{hj} of another bag B_h in DB. An instance may not have any corresponding related instance in some bags, i.e., a sequence is related to zero or one sequence per bag. We do not have necessarily the exact number of instances in each bag.

2.2. Running Example

In order to illustrate our proposed approach, we rely on the following running example. Let $\Sigma = \{A, B, \ldots, Z\}$ be an alphabet. Let $DB = \{(B_1, +1), (B_2, +1), (B_3, -1), (B_4, -1), (B_5, -1)\}$ a learning database that contains five bags (B_1 and B_2 are positive bags, B_3, B_4 and B_5 are negative bags). Initially, the bags contain the following sequences:
B_1 = {ABMSCD, EFNOGH, RUVR}
B_2 = {CEFCGHDD, EABZQCD}
B_3 = {YGHWM, XMCDSYZ}
B_4 = {ABIJYZ, KLSSO, EFYRTAB}
B_5 = {EFIVGH, KLSNAB}
We first use the across bag relation $\mathfrak{R}$ to represent the related instances using the index notation as described previously.

$$B_1 = \begin{cases} B_{11} = \mathbf{AB}MS\mathbf{CD} \\ B_{12} = \mathbf{EF}NO\mathbf{GH} \\ B_{13} = RUVR \end{cases} \quad B_2 = \begin{cases} B_{21} = E\mathbf{AB}ZQ\mathbf{CD} \\ B_{22} = C\mathbf{EF}C\mathbf{GH}DD \end{cases}$$

$$B_3 = \begin{cases} B_{31} = XM\mathbf{CD}S\mathbf{YZ} \\ B_{32} = Y\mathbf{GH}WM \end{cases} \quad B_4 = \begin{cases} B_{41} = \mathbf{AB}IJ\mathbf{YZ} \\ B_{42} = \mathbf{EF}YRTAB \\ B_{43} = \mathbf{KL}SSO \end{cases}$$

$$B_5 = \begin{cases} B_{52} = \mathbf{EF}IV\mathbf{GH} \\ B_{53} = \mathbf{KL}SN\mathbf{AB} \end{cases}$$

The goal here is to predict the class label of an unknown bag $Q = \{Q_1, Q_2, Q_3\}$ where:

$$Q = \begin{cases} Q_1 = \mathbf{AB}WX\mathbf{CD} \\ Q_2 = \mathbf{EF}XYNI\mathbf{GH} \\ Q_3 = \mathbf{KL}OF \end{cases}$$

2.3. MIL-ALIGN Approach

MIL-ALIGN [4] uses protein sequences of bacteria in order to predict whether a bacterium belongs to IRRB or IRSB. It formulates the IRR prediction problem as an MIL problem. The key idea of MIL-ALIGN is to discriminate bags by the use of local alignment technique to measure the similarity between each protein sequence in the query bag and corresponding protein sequences in the different bags of the learning database. Informally, the algorithm works as follows:

1. For each protein sequence in the query bag, MIL-ALIGN computes the corresponding alignment scores.
2. Alignment scores of all protein sequences of query bacterium into a matrix.
3. An aggregation method is applied in order to compute the final prediction result. Two aggregation methods have been proposed: (1) *Sum of Maximum Scores (SMS)* and (2) *Weighted Average of Maximum Scores (WAMS)*. More details can be found in [4].

In order to apply MIL-ALIGN to our running example, we use a simple similarity measure as an alignment score. This measure consists in the number of common symbols between the sequences. The first iteration computes the common symbols between the instance Q_1 of the query bag and the four related instances B_{11}, B_{21}, B_{31} and B_{41} (there is no related instance in the bag B_5). The results are stored in the first column of the matrix M. The second iteration and the third one compute the second and the third column of the matrix M.

$$M = \begin{pmatrix} 4 & 5 & 0 \\ 4 & 4 & - \\ 3 & 3 & - \\ 2 & 3 & 3 \\ - & 5 & 2 \end{pmatrix} \begin{matrix} B_1 \\ B_2 \\ B_3 \\ B_4 \\ B_5 \end{matrix}$$

Finally, an aggregation algorithm is applied to M in order to generate the final prediction result.

3. *ABClass*: Across Bag Sequences Classification

Figure 1 presents the system overview of ABClass, a structure based MIL approach.

As shown in Figure 1, ABClass starts by an encoding step which transforms sequences to an attribute-value format. It extracts motifs that serve as attributes/features. It is worthwhile to mention that if we extract motifs from all sequences of all bags (without taking into account the across bag relation) and use them as attributes, only a subset of the used attributes will be representative for each processed sequence. As for MIL-ALIGN, ABClass takes advantage of the across bag relationship between sequences in the classification process. Each set of related instances is presented by it own vector of motifs. Every vector will be used to generate a classification model. In the prediction step, a partial prediction result is produced from every model. These results are then aggregated to compute the final result. Based on the formalization, the algorithm discriminates bags by applying a classification model to each instance of the query bag.

The execution workfow of ABClass is described in Algorithm 1.

The acrossBagSeq function groups the related instances among bags into a list. During the execution of the algorithm, we will use the following variables:

- A matrix M to store the encoded data of the learning database.
- A vector QV to store the encoded data of the query bag.
- A vector PV to store the partial prediction results.

As illustrated in Algorithm 1, the main steps of *ABClass* are:

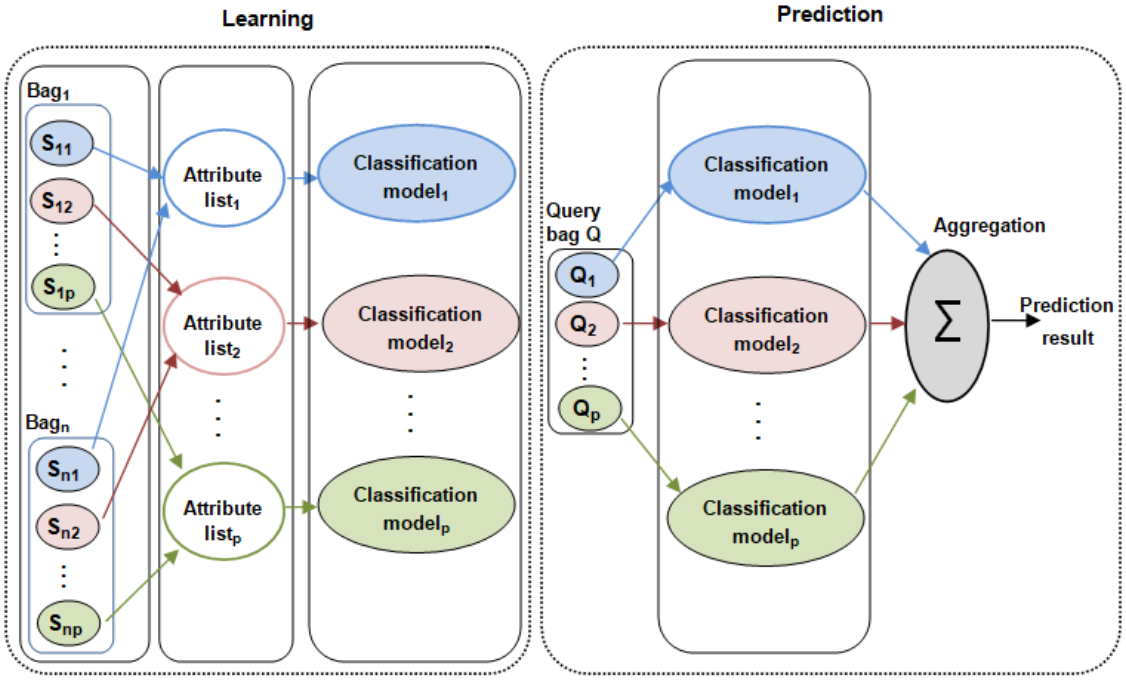


Fig. 1. System overview of the *ABClass* approach

Algorithm 1 *ABClass* algorithm

Input: Learning database $DB = \{(B_i, Y_i)|i = 1, 2, \ldots, n\}$, Query bag $Q = \{Q_k|k = 1, 2, \ldots, q\}$
Output: Prediction result P

1: **for all** $Q_k \in Q$ **do**
2: $AcrossBagSeqList_k \leftarrow AcrossBagSeq(k, DB)$
3: $MotifList_k \leftarrow MotifExtractor(AcrossBagsList_k)$
4: $M_k \leftarrow EncodeData(MotifList_k, AcrossBagsList_k)$
5: $Model_k \leftarrow GenerateModel(M_k)$
6: $QV_k \leftarrow EncodeData(MotifList_k, Q_k)$
7: $PV_k \leftarrow ApplyModel(QV_k, Model_k)$
8: **end for**
9: $P \leftarrow Aggregate(PV)$
10: **return** P

1. For each instance sequence Q_k in the query bag Q, the related instances among bags of the learning database are grouped into a list (lines 1 and 2).
2. The algorithm extracts motifs from the list of grouped instances. These motifs are used to encode instances in order to create a discriminative model (lines 3 to 5).
3. *ABClass* uses the extracted motifs to represent the instance Q_k of the unknown bag into a vector QV_k, then it compares it with the corresponding model. The comparison result is stored in the k^{th} element of a vector PV (lines 6 and 7).
4. An aggregation method is applied to PV in order to compute the final prediction result P (line 9), which consists in a positive or a negative class label.

3.1. Running Example

We apply the *ABClass* approach to our running example. Since the query bag contains 3 instances Q_1, Q_2 and Q_3, we need 3 iterations followed by an aggregation step.

Iteration 1: The algorithm groups the set of bags that are related and extracts the corresponding motifs.

$$AcrossBagsList_1 = \{B_{11}, B_{21}, B_{31}, B_{41}\}$$
$$MotifList_1 = \{AB, CD, YZ\}$$

Then, it generates the attribute-value matrix M_1 describing the sequences related to Q_1.

$$M_1 = \begin{array}{c} \begin{matrix} AB & CD & YZ \end{matrix} \\ \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix} \end{array} \begin{matrix} \\ B_{11} \\ B_{21} \\ B_{31} \\ B_{41} \end{matrix}$$

A model $Model_1$ is created from the encoded data. Then, a vector QV_1 is generated to describe Q_1.

$$QV_1 = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}$$

By applying the model to the vector QV_1, we obtain the first partial prediction result and we store it into the vector PV.

$$PV_1 \leftarrow ApplyModel(QV_1, Model_1)$$

Iteration 2: The second iteration concerns the second instance Q_2 of the query bag. We do the same instructions described in the first iteration.

$$AcrossBagsList_2 = \{B_{21}, B_{22}, B_{32}, B_{42}, B_{52}\}$$
$$MotifList_2 = \{EF, GH\}$$

$$M_2 = \begin{array}{c} \begin{matrix} EF & GH \end{matrix} \\ \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 0 & 1 \\ 1 & 0 \\ 1 & 1 \end{pmatrix} \end{array} \begin{matrix} \\ B_{12} \\ B_{22} \\ B_{32} \\ B_{42} \\ B_{52} \end{matrix}$$

$$QV_2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

$$PV_2 \leftarrow ApplyModel(QV_2, Model_2)$$

Iteration 3: Only B_1, B_4 and B_5 have related instances to Q_3.

$$AcrossBagsList_3 = \{B_{13}, B_{43}, B_{53}\}$$
$$MotifList_3 = \{KL\}$$

$$M_3 = \begin{array}{c} KL \\ \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} \end{array} \begin{array}{l} \\ B_{13} \\ B_{43} \\ B_{53} \end{array}$$

$$QV_3 = \begin{pmatrix} 1 \end{pmatrix}$$

$$PV_3 \leftarrow ApplyModel(QV_3, Model_3)$$

The aggregation step is finally used to generate the final prediction decision using the partial prediction results stored in the vector PV. We opt for the majority vote.

4. Experiments

We apply ABClass and MIL-ALIGN to the problem of bacterial IRR prediction. For our tests, we used the dataset described in [4] which consists of 28 bags (14 IRRB and 14 IRSB). Each bacterium/bag contains 25 to 31 instances that correspond to proteins implicated in basal DNA repair in IRRB. We used WEKA [9] data mining tool in order to apply existing well known classifiers to test ABClass.

4.1. Experimental Protocol

In order to evaluate the ABClass approach, we first encode the protein sequences of each bag using a set of motifs. Then, we apply an existing classifier to the encoded data. We used the Leave-One-Out (LOO) evaluation technique. In our tests, we used DMS [10] as a motif extraction method. The minimum motif length is fixed to 3. DMS allows building motifs that can discriminate a family of proteins from other ones. It first identifies motifs in the protein sequences. Then, the extracted motifs are filtered in order to keep only the discriminative and minimal ones. A substring is considered to be discriminative between the family F and the other families if it appears in F significantly more than in the other families. DMS extracts discriminative motifs according to α and β thresholds where α is the minimum rate of motif occurrences in the sequences of a family F and β is the maximum rate of motif occurrences in all sequences except those of the family F. In the following, we present the used motif extraction settings according to the values of α and β:

- **S1** ($\alpha = 1$ and $\beta = 0.5$): used to extract frequent motifs with medium discrimination.
- **S2** ($\alpha = 1$ and $\beta = 1$): used to extract frequent motifs without discrimination.
- **S3** ($\alpha = 0.5$ and $\beta = 1$): used to extract motifs having medium frequencies without discrimination.
- **S4** ($\alpha = 0$ and $\beta = 1$): used to extract infrequent and non discriminative motifs.
- **S5** ($\alpha = 1$ and $\beta = 0$): used to extract frequent and strictly discriminative motifs.

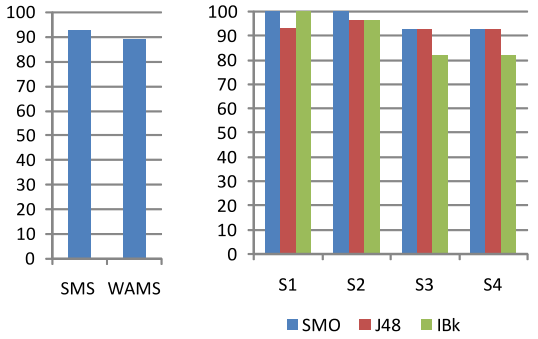
4.2. Experimental Results

Table 1 presents for each extraction setting the number of extracted motifs from each set of orthologous protein sequences. For the setting S5 ($\alpha = 1$ and $\beta = 0$), there is no frequent and strictly discriminative motifs for most proteins. This is why we will not use these values of α and β for our next experiments. We note that the number of extracted motifs increases for high values of β and low values of α.

Table 1. Number of extracted motifs for each set of orthologous protein sequences using a minimum motif length = 3.

Protein ID	Motif extraction setting				
	S1	S2	S3	S4	S5
P1	348	352	612	2226	229
P2	15	75	1139	5152	0
P3	6	41	681	4361	0
P4	2	21	446	3751	0
P5	1	1	119	1698	0
P6	11	29	349	3379	0
P7	5	18	371	3907	1
P8	3	62	484	3910	0
P9	7	42	780	4211	0
P10	25	90	719	3830	0
P11	3	7	200	2769	0
P12	4	17	144	1871	0
P13	0	1	111	1542	0
P14	2	12	133	2444	0
P15	3	50	303	2071	0
P16	0	1	187	2659	0
P17	3	27	349	2712	0
P18	0	1	81	1752	0
P19	7	14	427	3800	0
P20	2	20	343	3218	0
P21	21	79	882	4581	1
P22	18	173	785	3910	1
P23	5	43	524	4152	0
P24	5	48	520	3861	0
P25	1	5	264	2563	0
P26	22	72	778	3355	2
P27	5	9	162	1667	0
P28	16	111	572	3308	1
P29	2	11	189	2729	0
P30	9	65	281	1852	0
P31	0	0	92	2061	0
Total	551	1497	13072	95302	235

As shown in Figure 2, ABClass provides good overall accuracy results close to those provided by MIL-ALIGN. The best accuracy percentage using ABClass (100%) exceeds the highest one obtained using MIL-ALIGN (92.8%). It is reached using ABClass with the motif extraction settings S1 and S2. Using these settings, a minimum threshold of frequency and discrimination should be reached when extracting motifs. The least obtained accuracy percentage using ABCLass is 82.1%. This shows that our proposed approach is efficient. Table 2 presents the rate of classification models that contribute to predict the true class of each bacterium using ABClass approach. We present this rate for the motif extraction setting S1 that provides the best accuracy rates using SMO and IBk and the setting S4 that provides low accuracy percentages. The rate of successful classification models that does not exceed 50% are marked with bold



(a) MIL-ALIGN (b) ABClass approach

Fig. 2. Accuracy results.

text. The two bacteria B11 (*Methylobacterium radiotolerans*) and B15 (*Brucella abortus*) often generate low rates. We note that results are similar to those found in [4] using MIL-ALIGN. These results may help to understand some characteristics of the studied bacteria. A possible biological explanation is provided in [4] and [3]. It notes that this could be explained by the increased rate of sequence evolution in endosymbiotic bacteria.

5. Conclusion

In this paper, we focused on the problem of bacterial IRR prediction. The goal was to propose an algorithm that affiliated a bacterium to either IRRB or IRSB using a set of protein sequences. We described ABClass, a novel structure based MIL approach for bacterial IRR prediction. By running experiments, we have shown that the proposed approach is efficient. In the future work, we will study how to use the *a priori* knowledge in order to improve the efficiency of our algorithm. We specifically want to define weights for sequences using *a priori* knowledge in the learning phase.

Acknowledgments

This work was partially supported by the French-Tunisian project: General Directorate of Scientific Research in Tunisia (DGRST)/National Center for Scientific Research in France (CNRS) [IRRB11/R-14- 09], by the French Region of Auvergne, and by the Fédération de Recherche en Environnement [UBP/CNRSFR- 3467].

References

[1] Brim H, Venkateswaran A, Kostandarithes HM, Fredrickson JK, Daly MJ. Engineering Deinococcus geothermalis for bioremediation of high-temperature radioactive waste environments. Appl Environ Microbiol. 2003;69(1):4575-4582.
[2] Gabani P, Singh OV. Radiation-resistant extremophiles and their potential in biotechnology and therapeutics. Appl Microbiol Biotechnol. 2013;97(3):993-1004.
[3] Zoghlami M, Aridhi S, Maddouri M, Nguifo EM. An overview of in silico methods for the prediction of ionizing radiation resistance in bacteria. In: Reeve T, editor. Ionizing Radiation: Advances in Research and Applications, Nova Science Publishers Inc;2018. p. 241-256.
[4] Aridhi S, Sghaier H, Zoghlami M, Maddouri M, Nguifo EM. Prediction of ionizing radiation resistance in bacteria using a multiple instance learning model. J Comput Biol. 2016;23(1):10-20.
[5] Amores J. Multiple instance classification: Review, taxonomy and comparative study. Artif Intell. 2013;201:81-105.
[6] Alpaydn E, Cheplygina V, Loog M, Tax DM. Single-vs. multiple-instance classification. Pattern Recognit. 2015;48(9):2831-2838.
[7] Fang G, Bhardwaj N, Robilotto R, Gerstein MB. Getting started in gene orthology and functional analysis. PLOS Comput Biol. 2010;6(3):e1000703.
[8] Foulds J, Frank E. A review of multi-instance learning assumptions. The Knowledge Engineering Review. 2010;25(1):1-25.
[9] Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH. The weka data mining software: an update. SIGKDD explor. 2009;11(1):10-18.

Table 2. Rate of successful classification models using ABClass approach and LOO evaluation method

Bacterium ID	S1 motif extraction setting			S4 motif extraction setting		
	SMO	J48	IBk	SMO	J48	IBk
B1	86.3	81.8	90.9	68	60	**44**
B2	96.2	96.2	96.2	100	96.7	100
B3	92.5	92.5	92.5	100	90.3	100
B4	96.1	92.3	96.1	100	93.3	100
B5	100	92.3	100	100	86.6	100
B6	100	92.3	100	100	86.6	100
B7	88.8	88.8	92.5	100	93.5	100
B8	92	92	92	100	93.1	96.5
B9	92	86.9	91.3	100	84	96
B10	100	84	88	100	82.1	92.8
B11	62.5	**41.6**	**45.8**	**17.8**	**46.4**	**10.7**
B12	91.6	91.6	91.6	100	92.8	100
B13	95.6	82.6	95.6	92.5	66.6	**18.5**
B14	80.7	61.5	84.6	96.6	70	**43.3**
B15	83.3	79.1	87.5	**10.7**	**28.5**	**10.7**
B16	100	100	100	100	100	100
B17	95.8	95.8	95.8	96.2	96.2	100
B18	100	100	100	100	100	100
B19	100	100	100	100	100	100
B20	96	88	92	86.2	58.6	93.1
B21	100	100	100	93.1	82.7	93.1
B22	96.1	96.1	96.1	100	100	100
B23	92.5	92.5	96.2	100	96.7	100
B24	92.5	92.5	96.2	100	100	100
B25	92.5	92.5	96.2	100	96.7	100
B26	92.5	92.5	96.2	100	100	100
B27	100	100	100	100	96.2	100
B28	96.1	96.2	96.1	96.6	96.6	96.6

[10] Maddouri M, Elloumi M. Encoding of primary structures of biological macromolecules within a data mining perspective. J Comput Sci Technol. 2004;19(1):78-88.