Bacteriocin detection with distributed biological sequence representation

Md Nafiz Hamid and Iddo Friedberg

Department of Veterinary Microbiology and Preventive Medicine

Iowa State University Ames, IA, USA

{nafizh, idoerg}@iastate.edu

ABSTRACT

Antibiotic resistance is a major public health crisis, and finding new sources of antimicrobial drugs is crucial to solving this crisis. Bacteriocins, antimicrobial peptide products, have a narrow killing spectrum which leads to reduced pressure on selection for resistance. We propose a method that uses support vector machines to predict novel bacteriocins from primary protein sequences. To that end we use the word2vec method, widely used in natural language processing, to represent amino acid sequences. We use the Uniprot TrEMBL database taking advantage of this huge unlabeled data. Our method predicts multiple putative bacteriocins in *Lactobacillus*. This method will also help in detecting putative bacteriocins from rapidly generating newly sequenced bacterial data.

1. INTRODUCTION

The discovery and application of antibiotics rank among the greatest achievements of modern medicine. Antibiotics have eradicated many infectious diseases and enabled many medical procedures that would have otherwise been fatal, including modern surgery, organ transplants, and immunosupressive treatment such as radiation and chemotherapy. However, due to the massive use of antibiotics in healthcare and agriculture, antibiotic resistant bacteria have been emerging in unprecedented scales. Each year, 23,000 people in the US alone die from infections caused by antibiotic resistant bacteria.[1] One strategy to combat antibiotic resistance is to search for other antimicrobial drugs that are not classic antibiotics, and which may not be as prone to resistance.

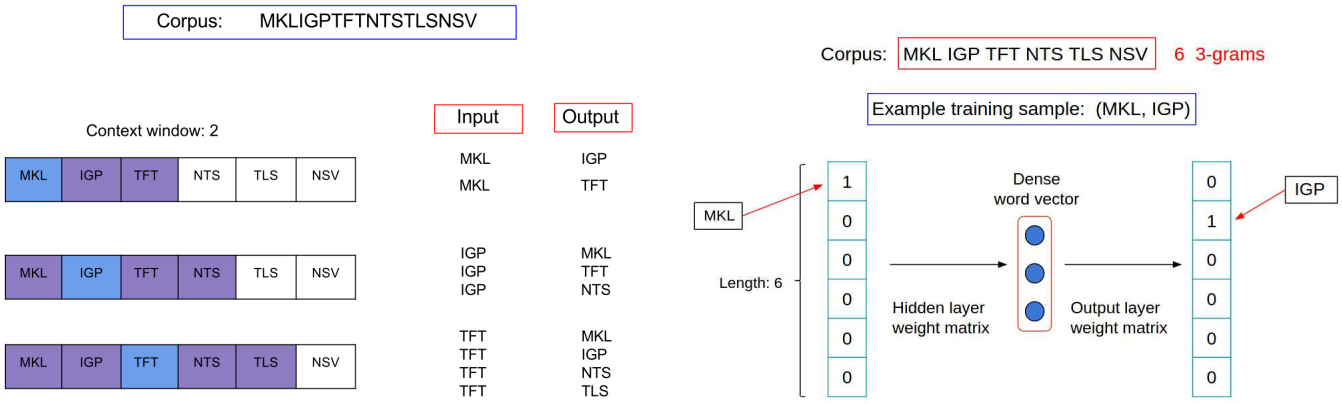
A promising class of compounds are the peptide-based bacteriocins, which are synthesized by many bacteria as antimicrobials. With the increased sequencing of genomes and metagenomes, we are presented with a wealth of data that also include genes encoding bacteriocins. Bacteriocins can also have a narrow killing spectrum leading to a hypothesis that they can be applied as 'designer drugs'.[2] Because of this selectivity in virulence, antiobiotics synthesized from bacteriocins will lead to a slow proliferation of resistance increasing the shelf life of new drugs.

Several computational tools and databases have been developed to aid in the discovery and identification of bacteriocins. BAGEL[3] is a database and a homology-based mining tool that includes a large number of annotated bacteriocin sequences. BACTIBASE[4] is a similar tool, which also contains predicted sequences. AntiSMASH[5] is a platform for genome mining for secondary metabolite producers, which also includes bacteriocin discovery. Recently, we introduced BOA,[6] a standalone genome-mining software that identifies possible bacteriocins by searching for homologs of *context genes*: genes that are associated with the transport, immunity, regulation, and post-translational modification of bacteriocins. However, due to their unorthodox structure, bacteriocins are hard to find using standard bioinformatics methods.

Here we present a novel method to identify bacteriocins from protein sequence alone. The main challenge in detecting bacteriocins is having a small number of positive examples of known bacteriocin sequences. Therefore, the requirement for a representation of the sequences that captures its unique intricacies is more so crucial. Towards that end, we represent protein sequences using word2vec.[7] After an initial unsupervised learning step, we use a supervised learning algorithm to detect novel putative bacteriocins.

2. METHODS

Word2vec is a technique used in natural language processing that uses a large corpus of text as its input and provides a vector representation for the words as output. The goal is to give words that appear in similar contexts similar representations. This type of representation usually leads towards better results in various types of subsequent classification problems. The training can be done in two ways: the continuous bag of words(CboW) model, or the skip-gram model.

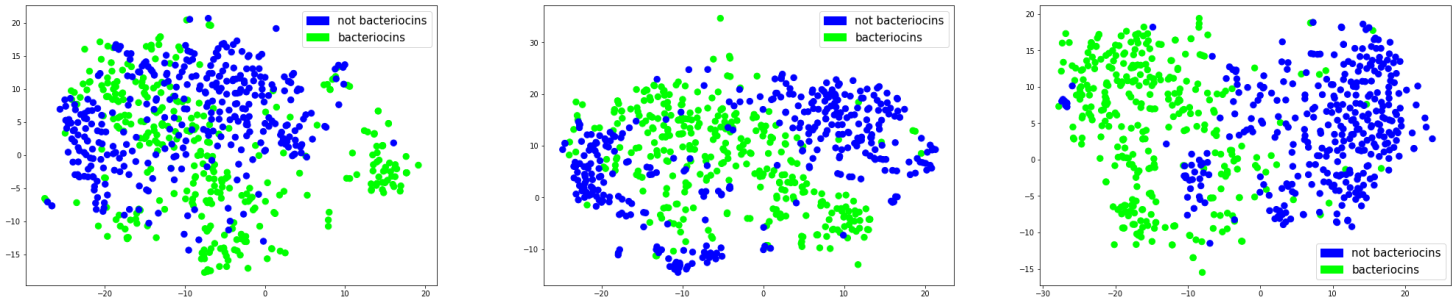


(a) Generating training instances from an amino acid sequence

(b) Neural network architecture for a single training instance

Figure 1: Representation learning for 3-grams with skip-gram training. (a) **Training**: a protein sequence (top left), is broken into 3-grams with each 3-gram associated with the previous and following two 3-grams. (b) the neural network architecture for MKL as input, and IGP as output.

We used bacterial sequences from the Uniprot TrEMBL database[8] as our unsupervised training corpus and the skip-gram model as the training method to get a representation for each 3-gram. The skip-gram model is a neural network where the inputs and outputs of the network are one hot vectors with our training instance input word and output word. The size of the one hot vector is the size of our whole vocabulary which is all the 3-grams. We generated the training instances with a context window of size 3, where we took a word as input and used all of its surrounding words within the context window as outputs. The process is explained in Figure 1 for a corpus with a single amino acid sequence. Unsupervised training, generated a 100 dimensional vector for each of the 3-grams. We represented each sequence as the summation of vectors of all overlapping 3-grams which was shown to be effective for protein sequecne representation by Asgari *et al.*[9]



(a) with primary negative dataset **(b)** with second negative dataset **(c)** with third negative dataset

Figure 2: t-sne visualization of word vector representations for positive and negative bacteriocin amino acid sequences

We used 346 sequences from the BAGEL database as our positive bacteriocin training samples. For the negative training set, we used the Uniprot Swissprot database which is manually reviewed and more reliable. We took all the bacterial amino acid sequences from this database and used CD-HIT with a 50% identity threshold to reduce redundancies. Then, for the primary negative training set, we took 346 sequences

Table 1: Comparison between word2vec and k-mer representation. Bold numbers are the best results between these two representations. Blue color areas represent the best classification method for both types of representation

	word2vec			k-mer		
Methods	Mean Precision	Mean Recall	Mean F_1	Mean precision	Mean Recall	Mean F_1
SVM	0.844	**0.843**	**0.842**	**0.867**	0.796	0.827
Logistic Regression	0.845	**0.831**	0.837	**0.864**	0.831	**0.842**
Decision Tree	0.759	**0.759**	0.757	**0.767**	0.759	**0.761**
Random Forest	0.811	**0.820**	0.813	**0.834**	0.803	**0.817**

Table 2: SVM with word2vec representation on three different negative set

	Mean Precision	Mean Recall	Mean F_1
primary negative dataset	0.844	0.843	0.842
second negative dataset	0.916	0.892	0.902
third negative dataset	0.955	0.927	0.940

that had the keywords 'not anti-microbial', 'not antibiotic', 'not in plasmid', and that had the same length distribution as our positive bacteriocin sequences. For consistency, we also created two additional negative datasets following the same steps as above, but with different sequences and no overlap between the three sets. Figure 2 shows the t-sne visualizations of all the datasets where each point is a sequence that was represented by a 100 dimensional vector. We then applied several supervised classification methods to classify positive and negative bacteriocin sequences.

To search for genomic regions with better chances of containing novel bacteriocins, we took advantage of the biological knowledge of context genes which assist in the transport, modification, and regulation of bacteriocins. Usually, many bacteriocins have some or all of four types of context genes in proximity. Having an experimentally verified set of 54 context genes, we collected the annotation keywords for these 54 context genes from the Refseq database, and BLASTed the BAGEL bacteriocins against the non-redundant protein database. We removed the top hits from the result which are essentially the bacteriocins themselves. We then took all the genes with similar keywords from our experimentally verified context gene set surrounding these bacteriocins within 25kb. After doing CD-HIT to get rid of redundancy, we had 1240 new putative context genes.

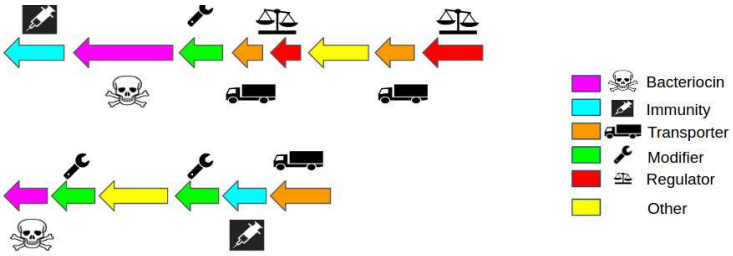


Figure 3: Bacteriocins with context genes

We BLASTed all 1294 putative context genes against the whole bacteria refseq database and we collected hits with an e-value $\leq 10^{-6}$. We then identified 50kb regions of contiguous hits. Figure 4 provides a graphical view of our approach. We applied our trained machine learning model on these 50kb regions to predict putative bacteriocins.

3. RESULTS

We performed a 10-fold nested cross-validation on the set of positive bacteriocins and the primary negative bacteriocins. Table 1 compares the word2vec approach and a k-mer based representation with various classification algorithms. For the k-mer based approach we created an 8000 size vector for each sequence where the indices had counts for each occurrence of a 3-gram in that sequence. As this is hugely sparse, we used truncated Singular Value Decomposition(SVD) to reduce the dimension to 100, then we applied the classification algorithms.

Table 1 shows that a support vector machine (SVM) gives us the best F_1 score for word2vec while for k-mer it is the logistic regression. While the k-mer based approach had a better precision overall, word2vec approach provides a greater balance between precision and recall. Specially, for our purpose, which is to retrieve as many bacteriocins as we can, having a good

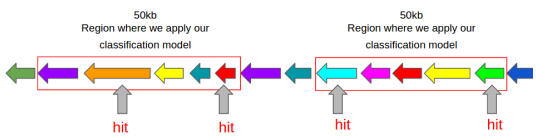


Figure 4: Potential areas for putative bacteriocins

recall is essential. Consequently, we used our trained SVM to find new bacteriocins in the Refseq[10] database where the sequences are represented with our word2vec model.

Table 2 shows the performance of SVM with word2vec representation for different negative bacteriocin datasets. The performance gain is quite huge which is somewhat anticipated as the second and third negative set has slightly different length distributions for the negative set than the positive bacteriocin set due to lack of sequences of certain lengths. The t-sne representations with these two sets also show a more clear separation between the positive points and the negative points.

We applied our trained SVM model on the identified 50kb regions to predict putative bacteriocins. The model predicted a total of 1186 putative bacteriocins in *Lactobacillus* with varying degrees of probability. We predicted 11 putative bacteriocins with a probability of 0.95 or more. While work on experimentally verifying them is ongoing, we also computationally analyzed their potential of being bacteriocins. Previously, we mentioned that many bacteriocins have some or all of four types of neighboring context genes. Therefore, we looked for these context genes in the vicinity of our predicted bacteriocins.

Figure 5 shows three examples of context genes found in genomic stretches next to bacteriocin genes. Our predicted bacteriocin in *Lactobacillus acidophilus La-14* has regulator and transporter genes in its vicinity. The predicted bacteriocin in *Lactobacillus acidophilus 30SC* also has regulator and transporter genes surrounding it. Similarly, the predicted bacteriocin in *Lactobacillus acidophilus NCFM* has regulator, transporter, and immunity genes in its vicinity. Class I bacteriocins are heavily post-translationally modified with the help of modifier genes while class II bacteriocins remain unmodified. Our predictions seem to indicate that they are of class II.

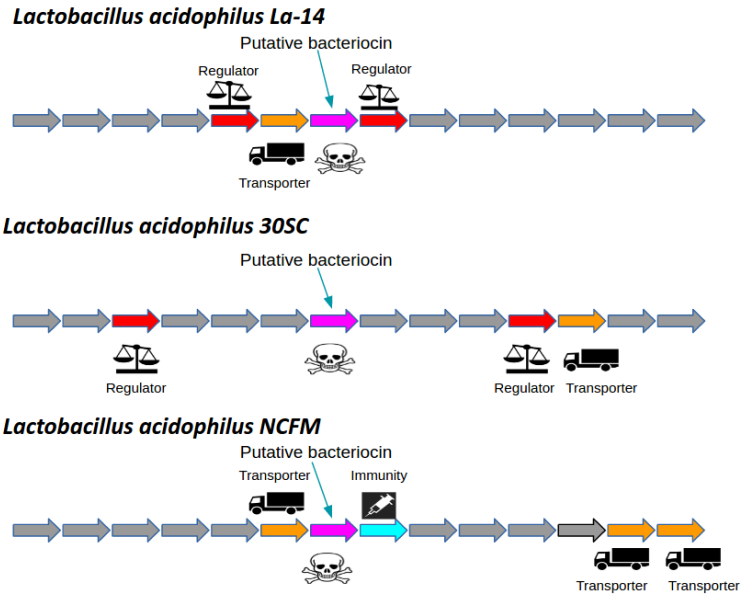


Figure 5: Context genes found surrounding the predicted bacteriocins wihin +/- 25kb range

4. DISCUSSION

We developed a machine learning approach for predicting bacteriocins from protein sequence that takes advantage of the large volume of unlabeled data. Our application of word2vec provides us with a representation of amino acid sequences that helps us use the modest size of our positive training set. The 3-gram representation we generated can be used in other machine learning tasks across computational microbiology without requiring any more expensive training. For example, these representations can work as a pre-trained layer in a different deep learning task. We hypothesize that changing hyperparameters for our skip-gram training to do more extensive training might improve the results which will improve upon the best case of a k-mer based approach. For example, in our training, we used a context window of 3 and a vector size of 100. Increasing the context window size might enable the neural network to incorporate more information into the dense vector though it requires more training time. Our machine learning approach does not need hand-engineered features. Using machine learning also provides us with an associated confidence score, which is useful for experimentalists who wish to apply this method towards genome mining of bacteriocins.

References

[1] Centres for Disease Control and Prevention (US). *Antibiotic resistance threats in the United States, 2013*. Centres for Disease Control and Prevention, US Department of Health and Human Services, 2013.

[2] Margaret A Riley and John E Wertz. Bacteriocins: evolution, ecology, and application. *Annual Reviews in Microbiology*, 56(1):117–137, 2002.

[3] Auke J van Heel, Anne de Jong, Manuel Montalban-Lopez, Jan Kok, and Oscar P Kuipers. Bagel3: automated identification of genes encoding bacteriocins and (non-) bactericidal posttranslationally modified peptides. *Nucleic acids research*, 41(W1):W448–W453, 2013.

[4] Riadh Hammami, Abdelmajid Zouhir, Christophe Le Lay, Jeannette Ben Hamida, and Ismail Fliss. Bactibase second release: a database and tool platform for bacteriocin characterization. *Bmc Microbiology*, 10(1):22, 2010.

[5] Tilmann Weber, Kai Blin, Srikanth Duddela, Daniel Krug, Hyun Uk Kim, Robert Bruccoleri, Sang Yup Lee, Michael A Fischbach, Rolf Müller, Wolfgang Wohlleben, et al. antismash 3.0a comprehensive resource for the genome mining of biosynthetic gene clusters. *Nucleic acids research*, 43(W1):W237–W243, 2015.

[6] James T Morton, Stefan D Freed, Shaun W Lee, and Iddo Friedberg. A large scale prediction of bacteriocin gene blocks suggests a wide functional spectrum for bacteriocins. *BMC bioinformatics*, 16(1):381, 2015.

[7] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[8] Rolf Apweiler, Amos Bairoch, Cathy H Wu, Winona C Barker, Brigitte Boeckmann, Serenella Ferro, Elisabeth Gasteiger, Hongzhan Huang, Rodrigo Lopez, Michele Magrane, et al. Uniprot: the universal protein knowledgebase. *Nucleic acids research*, 32(suppl 1):D115–D119, 2004.

[9] Ehsaneddin Asgari and Mohammad RK Mofrad. Continuous distributed representation of biological sequences for deep proteomics and genomics. *PloS one*, 10(11):e0141287, 2015.

[10] Nuala A O'Leary, Mathew W Wright, J Rodney Brister, Stacy Ciufo, Diana Haddad, Rich McVeigh, Bhanu Rajput, Barbara Robbertse, Brian Smith-White, Danso Ako-Adjei, et al. Reference sequence (refseq) database at ncbi: current status, taxonomic expansion, and functional annotation. *Nucleic acids research*, 44(D1):D733–D745, 2016.

[11] Antimicrobial Resistance. Tackling a crisis for the health and wealth of nations. review on antimicrobial resistance, chaired by jim oneill. december 2014, 2015.

[12] Radim Řehůřek and Petr Sojka. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May 2010. ELRA.

[13] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.