A Novel Metagenomic Binning Framework Using NLP Techniques in Feature Extraction

VIET TOAN TRAN\textsuperscript{1} HOANG D. QUACH\textsuperscript{2} PHUONG V. D. VAN\textsuperscript{1,3,a} VAN HOAI TRAN\textsuperscript{1,b}

Received: June 17, 2021, Accepted: October 27, 2021

Abstract: Without traditional cultures, metagenomics studies the microorganisms sampled from the environment. In those studies, the binning step results serve as an input for the next step of metagenomic projects such as assembly and annotation. The main challenging issue of this process is due to the lack of explicit features of metagenomic reads, especially in the case of short-read datasets. There are two approaches, namely, supervised and unsupervised learning. Unfortunately, only about 1% of microorganisms in nature is annotated. That can cause problems for supervised approaches when an under-study dataset contains unknown species. It is well-known that the main challenging issue of this process is due to the lack of explicit features of metagenomic reads, especially in the case of short-read datasets. Previous studies usually assumed that reads in a taxonomic label have similar k-mer distributions. Our new method is to use Natural Language Processing (NLP) techniques in generating feature vectors. Additionally, the paper presents a comprehensive unsupervised framework in order to apply different embeddings categorized as notable NLP techniques in topic modeling and sentence embedding. The experimental results present our proposed approach’s comparative performance with other previous studies on simulated datasets, showing the feasibility of applying NLP for metagenomic binning. The program can be found at https://github.com/vandinhvyphuong/NLPBimeta.

Keywords: Metagenomic binning, topic modelling, NLP, short reads

1. Introduction

The main purpose of microbiological community research is to understand the composition, diversity, and function of microorganisms. Differentiating from genomics whose target is a single DNA sequence, metagenomics is to study many organisms living within the same sample, which are not easily separated or further explored due to their interaction and mutual impact. A sequencing-based metagenomics project normally involves several steps: DNA extraction, library preparation, sequencing, clustering, assembly, annotation, and statistical analysis.

Recently, advanced sequencing techniques have enabled to study of whole metagenomes sampled directly from the environment. These techniques (in DNA extraction, library preparation, and sequencing) are able to reproduce in-depth metagenomic data which consists of genomes of microorganisms. However, the outputs of these techniques (specifically shotgun metagenomic sequencing) are possible to generate a mixture of short DNA fragments (also called reads) of sampled genomes. Certainly, some sequencing techniques could generate long fragments making later steps (clustering, assembly, and annotation) more efficient, along with certain disadvantages such as high cost, higher error rates.

The aim of the paper is on the clustering step working on datasets of short reads (∼80 bp). In biology, this problem targeted by this paper is named metagenomic binning (or taxonomic binning). In detail, such DNA fragments should be separated into taxonomy-related groups. After that, the output of binning will be fed to the next phases (e.g., assembly or annotation), therefore, clustering quality will have a strong impact on the final result of the whole project.

Supervised approach for taxonomy classification requires a comprehensive reference database, which is very difficult due to the fast evolution of viruses and bacterial in nature. Therefore, unsupervised one is a more reasonable solution. To solve this problem, previous studies [1], [2] proposed two-phase algorithms which assumed that reads of a gene family should have similar k-mer distributions. Based on this idea, those studies apply the first phase to connect all similar reads together, creating a so-called overlapping graph. A certain graph partition algorithm is involved to divide a read set (i.e., a vertex set) into groups which will be represented by certain feature extraction methods. The BiMeta [2] employs k-mer distribution as in Fig. 1 to count only the frequency of k-mers and other global and local statistics of k-mer in a single read and in the whole dataset. Therefore, MetaProb [1] proposed an improved version in feature extraction. Our main idea as Fig. 2 is to explore the capability of NLP (Natural Language Processing) embedding (aka feature) techniques in a process of automatic feature extraction. As discussed in NLP studies, NLP techniques can extract both frequent and order information of DNA sequences. The research try notable methods of NLP techniques including:

• Topic modeling using Latent Dirichlet Allocation (LDA) [3],
frequency (like ically, in feature extraction, NLP models do not only count word NLP which outperforms human capacity in various tasks. Specifically, during the past five years, there are many breakthroughs in understanding human capability of text understanding by that of computers. Moreover, with the evolution of viruses and bacteria, there are many new genomes that no reference database can be kept up-to-date. Consequently, the second approach, called taxonomy-independent, is of greater interest in which no reference database is required. Without any external information, this approach depends entirely on internal taxonomic information of reads. In this section, the paper will describe notable studies relevant to our work.

Feature extraction is a main common sub-problem that must be solved by all taxonomy-independent methods, which based on unsupervised learning. The paper only discuss biological issues of feature extraction. Two feature types of metagenomic reads are sequence composition and abundance (or coverage). Categorized by the feature types, taxonomy-independent methods are classified into 3 different mechanisms: (i) sequence composition-based methods, (ii) abundance-based methods, and (iii) hybrid methods.

A crucial step in composition-based methods is to transform nucleotide sequences to a numeric feature vector, also called genomic signature or composition. A basic example of this transformation is GC-content. Note that it is assumed that a signature of a genome is unique for each taxon, especially in the context of composition-based methods. An advanced, effective and widely-used way to create the signature is to use k-mer frequency distribution, with a given k. Figure 1 expresses with k = 2, and a vector of length 16 (2^4 = 16) be generated. This vector shows the frequency of “words” (AA, AT, .., TT). In the experiments, words are formed with the value k = 4, a value commonly used when analyzing metagenomic data. The words will have 256 (2^8 = 256) words (AAAA, AAAC, ..., TTTT). In-depth analysis and application of this composition have been proposed in different studies, along with additional similarity measures. It is easy to see that the vector dimension of k-mer distribution should be large enough to highly represent certain taxonomic features, leading to the limitation of composition-based methods to large datasets. Therefore, some studies focused on reducing the dimension of k-mer distribution. The list of well-known composition-based methods could be TETRA [10], LikelyBin [11], SCIMM [12], VizBin [13], BiMeta [2], MetaProb[1]. Some of them do not base on k-mer distribution; instead, they introduced some sophisticated sequence signature which proves to have a better representation of insight taxonomic information (e.g., MetaProb [1]).

A weakness of composition-based methods is the ineffectiveness of clustering species with a low abundance. In other words, binning algorithms can wrongly assign reads belonging to small groups to ones of other species with higher abundance. The abundance-based approach is designed to address this problem by
assuming that the distribution of sequences is modeled as a mixture of Poisson distributions. Instead of working on small \( k \) as in composition-based methods, abundance-based methods set up high values of \( k (k \geq 20) \) to form a large distribution, effectively separating clusters by analyzing the abundance of \( k \)-mers. Another advantage of abundance-based methods (e.g., AbundanceBin[14]) is that they deliver higher binning accuracy than that of composition-based methods with short-read datasets (75 bp long). Note that the abundance-based approach is not comparative with the composition-based approach for long-read datasets (\( \geq 800 \) bp).

Hybrid methods orchestrate composition-based features and abundance-based features into a unified framework to obtain higher accuracy. CompostBin[15] firstly reduces the dimension of 6-mer distribution by principal component analysis (PCA) and then assigns to every frequency a weight valued by the inverse of its abundance. CONCOCT[16] leverages the PCA analysis for a combined vector of both \( k \)-mer frequency and coverage. Instead of using Euclidean measure to compare combined features, COCACOLA[17] uses \( L_1 \) to avoid over-penalizing the large difference of some frequency. MaxBin[18], MetaBAT[19], MyCC[20] are counted as hybrid methods.

Natural language process (NLP) is an emerging area in theory and practice, as mentioned in Section 1. A blooming domain of applying NLP in text classification strongly relates to our binning problem. Reference[21] is one of the good surveys in this domain. Four nucleotides (A, C, G, T) construct an alphabet that generates words (\( k \)-mer) and further documents as in NLP. Therefore, researchers believe NLP techniques would be helpful in DNA analysis in general and metagenomics individually. Topic modeling is a vital tool in understanding natural language texts, and Liu et al. [22] cover an overview on applications of topic modeling in bioinformatics. For the metagenomic binning, Chen et al.[23], [24], [25] used frequency vectors of \( k \)-mers (large \( k \)) to represent DNA sequences and applied the Latent Dirichlet Allocation (LDA) model to infer hidden topics in the expectation that each hidden topic represents a particular genome. Similar studies using LDA were proposed in Refs. [26], [27], and achieved promising performance compared to other state-of-the-art methods such as AbundanceBin[14], MetaCluster[28], and MCluster[29].

As discussed before, feature extraction by topic modeling is still an exciting research field. To our knowledge, all other studies primarily apply LDA and its variants to transform the \( k \)-mer distribution of each single DNA sequence into a hidden space. They presume that the feature vector in that space summarizes better the internal taxonomic information of the sequence, finally leading to good binning performance. However, it is all-known that LDA does not work well on short texts (see Ref. [30]) when considering a sequence as a single document. Therefore, a comprehensive mechanism to solve the binning problem for short-read metagenomic datasets are still open, especially by topic modeling techniques. The following section will present a framework to address this challenge.

3. Method

3.1 BiMeta: A Two-phase Binning Algorithm

BiMeta proposes a two-phase binning algorithm in which a main idea is to transform the binning problem into a graph partitioning problem. This idea of BiMeta has been employed in other studies such as MetaProb, CLAME, MetaBCC-LR. In the first phase, reads are partitioned into groups based on overlapping information between them. The partitioning process is performed on a graph modeling a close biological relationship between reads. In other words, instead of considering a single read as a unit for clustering problem, a group of reads will be the smallest the clustering unit.

In the second phase, BiMeta embeds groups of reads by using \( k \)-mer distribution, then it clusters the groups into species using \( k \)-means algorithm. Meanwhile, MetaProb improves \( k \)-mer embedding by using a metagenomic-customized probabilistic sequence signature. With such the embedding improvement in the second phase, MetaProb could improve the overall performance in most datasets.

To cluster well the groups, a good embedding method that can represent groups of reads of species into similar/near vectors in the embedding space and far vectors for groups of different species. \( k \)-mer omits sequential data of reads, so this paper propose to use NLP techniques to exploit this data.

3.2 NLP Embedding

Figure 3 signifies a new comprehensive framework for a two-stage algorithm using NLP techniques for feature extraction. Not as BiMeta and MetaProb, our proposed method executes two branches independently on the input (i.e., read set). On one side, NLP model is executed to capture the embedding of every single read (in this case, a read is preprocessed into a document as discussed in the previous section, using \( k \)-mer based vocabulary). it is called per-read embedding. On another side, the graph-based first phase as used in BiMeta and MetaProb is employed to divide the read set into groups and their corresponding seeds (seed is a subset of a group, representing the group). Then, at the end of this first phase, the per-read embedding of reads in the same group are merged (by mean computation) to create per-group embedding. NLP techniques when working on long documents will try to understand high levels (paragraph, document) of embedding which contains positional information of words and sentences within paragraph/document. It is meaningless in metagenomic binning, especially in the case of short reads. In the clustering step, assembly is yet executed, therefore, grouping reads to create a document (e.g., by concatenating them) will mislead the feature extraction by NLP techniques. That is the main reason why our method extracts only per-read NLP embeddings and then combines them later.

In the second phase, this paper reuses the same idea in BiMeta furthermore does not attempt to use other clustering methods and even different clustering parameter settings to have a fair comparison. The main purpose of the study is to investigate the capabilities of the state-of-the-art NLP embeddings for the metagenomic data.
3.3 NLP Embedding - Topic Modeling

In this step, the paper conduct three methods (Latent Dirichlet Allocation (LDA), Non-negative Matrix Factorization (NMF), and Latent Semantic Analysis (LSA)) to discover hidden topics. We assume that there are $r$ topics across all groups of reads.

3.3.1 Using Latent Dirichlet Allocation (LDA) [3]

Similar to $k$-mer, LDA considers documents as bags of words (i.e., LDA does not extract information on the order of words). The main goal of LDA is to represent topics by word probabilities. LDA assumes that each document $d$ can be represented as a Dirichlet distribution $\theta_d$ of topics, this distribution contains parameter $\alpha$. Also, each topic $k$ can be represented as a Dirichlet distribution $\phi_k$ of words with parameter $\beta$. After training, the probability of a collection of documents $D$ is as Eq. (1).

$$p(D|\alpha,\beta) = \frac{1}{M} \prod_{d=1}^{M} \int p(\theta_d|\alpha) \left( \prod_{n=1}^{N_d} p(z_{dn}|\theta_d) p(w_{dn}|z_{dn},\beta) d\theta_d \right)$$  

(1)

3.3.2 Using Non-negative Matrix Factorization (NMF) [4]

Firstly, we build a term frequency-inverse document frequency (TF-IDF) matrix from groups denoted by $V$, which is $n \times 4^k$; $n$ is the number of groups; $4^k$ is the number of words ($k$-mers). NMF tries to represent $V$ by two smaller matrices as $V \approx WH$ – Eq. (2).

$$V = \begin{bmatrix} \text{k-mer}_1 & \cdots & \text{k-mer}_r \\ \vdots & \ddots & \vdots \\ \text{k-mer}_{4^k} \end{bmatrix}$$

$$W = \begin{bmatrix} \text{Group}_1 & \cdots & \text{Group}_n \\ \vdots & \ddots & \vdots \\ \text{Group}_{4^k} \end{bmatrix}$$

$$H = \begin{bmatrix} \text{Topic}_1 & \cdots & \text{Topic}_r \\ \vdots & \ddots & \vdots \\ \text{Topic}_{4^k} \end{bmatrix}$$

(2)

3.3.3 Using Latent Semantic Analysis (LSA) [5]

LSA tries to represent the input matrix into three smaller matrices as $A = U \ast \Sigma \ast V$ – Eq. (3). In the Equation, $U$ is the word assignment to topics; $\Sigma$ presents topic importance; $V$ is the topic distribution, also is the result of embedding the groups.

$$A = \begin{bmatrix} \text{Groups} \\ \text{k-mers} \end{bmatrix} = \begin{bmatrix} \text{Topics} \\ \text{U} \end{bmatrix}$$

$$V = \begin{bmatrix} \text{Topics} \\ \Sigma \end{bmatrix} = \begin{bmatrix} \text{Groups} \\ \text{V} \end{bmatrix}$$

(3)
3.4 NLP Embedding - Sentence Embedding

In this approach, we apply two notable methods including Doc2Vec and the state-of-the-art method – BERT. Note that, in our paper, words are k-mers and documents are reads. Other natural language components of a text such as sentence and paragraph are not investigated explicitly in this study.

3.4.1 Doc2Vec [6]

The goal of Doc2Vec is to create a numeric representation of a document, regardless of its length. Doc2Vec introduced two models including Distributed Bag Of Words (DBOW) and Distributed Memory (DM). In Doc2Vec, every paragraph (document) is mapped to a unique vector (paragraph id), represented by a column in matrix $D$, and every word is also mapped to a unique vector, represented by a column in matrix $W$.

Distributed Memory version of Paragraph Vector (PV-DM) uses words and a document-unique vector to predict the next word as Fig. 4. This model includes word vectors $W$, paragraph vectors $D$, and softmax weights $U, b$. To train them, at every step of stochastic gradient descent, one can sample a fixed-length context from a random paragraph, compute the error gradient from the network and use the gradient to update parameters of the model.

Distributed Bag of Words version of Paragraph Vector (PV-DDBOW) uses only a document-unique vector to predict all surrounding words as Fig. 5. DBOW does not consider the order and local-context/neighboring words. Similarly, PV-DM, the paragraph vectors, and word vectors are trained using stochastic gradient descent and the gradient is obtained via back-propagation.

After being trained, the paragraph vectors in $D$ can be used as a representation for the paragraph/document. When we face new paragraphs/documents (never seen before), we add more columns to $D$, gradient descending on $D$ while holding $W, U, b$. PV-DM alone usually works well for most tasks (with state-of-art performances) [6], so in this paper, we just apply PV-DM.

3.4.2 Bidirectional Encoder Representations from Transformers (BERT) [7]

BERT is undoubtedly a breakthrough in the use of Deep Learning for Natural Language Processing. There are many pre-trained models of BERT supporting many languages. Using BERT includes two stages: pre-training and fine-tuning. With the human-language task, we only need the second stage because pre-trained models were conducted pre-training on huge datasets of human languages. However, our biological task is different due to no biological datasets for pre-training, so the pre-trained models may not work. Therefore, we need to conduct pre-training from scratch, but this approach is relatively computing expensive. In this paper, we train a reasonably small-sized BERT model with less than 100 thousand parameters to verify the benefits of the NLP embeddings.

Instead of predicting the next word as Doc2Vec, BERT proposed a novel technique call Masked-Language Modeling (MLM). The technique randomly masks words in the sentence then the model tries to predict these masked words. Doc2Vec uses only previous words and paragraph id to predict the next words, while BERT uses both left and right surroundings to predict. This approach can help BERT to represent a word using both its previous and next context. Also, BERT based on a very advanced layer named Transformer [31] that is an attention mechanism learning contextual relationships between words in a text.

4. Experiment Results

4.1 Datasets

Our proposed framework aims at clustering short-read datasets, which should be chosen as experimental datasets. As discussed in methodological Section 3, the main focus of the study is on feature extraction by topic modeling. It replaces the feature ideas of $k$-mer distribution of BiMeta [2], and the probabilistic sequence signature of MetaProb [1]. Therefore, we will reuse synthetic datasets mentioned in those papers to compare the experimental results with previous methods. Datasets are generated from the Metasim application [32]. Note that MetaSim generates the datasets with an error-permitted Illumina profile (1% error rate). The paired-end reads are approximately 80 bp. The problems $S^*$ are controlled by a wide spectrum of several parameters (number of species, coverage ratio, phylogenetic distance), therefore, they should be more challenging in binning. The problems $L^*$ only have two species (Eubacterium eligens and Lactobacillus amylovorus) with different abundances. More details on the datasets are in Table 1.

4.2 Evaluation Measures

Denote $\{i\}_{i=1}^{\#R}$ be the index set of reads in a dataset $R$. $y_i \in \{1, 2, \cdots , p\}$ is the ground-truth labels (families) for read $i$. Note that $p$ is the number of clusters (and also number of species to be clustered). $T = \{T_1, T_2, \cdots , T_p\}$, in which cluster $T_j$ consists reads which have label $j$, i.e., $T_j = \{r \in R|y_r = j\}$. $C = \{C_1, C_2, \cdots , C_p\}$ denotes a collection of clusters of a dataset, which is the results of the clustering algorithm, and $\hat{y}_i \in \{1, 2, \cdots , p\}$ is the cluster label of $r_i$. $T$ is the ground-truth partitioning, and to each $T_i$ as a partition.

Fig. 4 PV-DM model [6].

Fig. 5 PV-DDBOW model [6].
The clustering validation measures try to capture the extent to which reads from the same partion (species) appear in the same cluster and the extent to which reads from different partitions are grouped in different clusters. These measures rely on the $p \times p$ contingency table $N$ (see Table 2) that is induced by a clustering $C$ and the ground-truth partitioning $T$, defined as follows:

$$N(i, j) = n_{ij} = |C_i \cap T_j|$$

To evaluate methods, we calculate three metrics including precision, recall, and F-measure. These measures are calculated from Table 2 following below formulas.

### Table 1 Short-reads metagenomic datasets (adapted from Refs. [2] and [1]).

| Dataset | No. of species | Phyllogenetic distance | No. of reads (ratio) |
|---------|----------------|------------------------|----------------------|
| S1      | 2              | Species                | 963867 (1:1)         |
| S2      | 2              | Species                | 195339 (1:1)         |
| S3      | 2              | Order                  | 338725 (1:1)         |
| S4      | 4              | Phylum                 | 375302 (1:1)         |
| S5      | 3              | Species and Family     | 325400 (1:1:1)       |
| S6      | 3              | Phylum and Kingdom     | 713388 (3:2:1)       |
| S7      | 5              | Order, Order, Genus, Order | 1653550 (1:1:1:4:4) |
| S8      | 5              | Genus, Order, Order, Order | 456224 (3:5:7:9:11) |
| S9      | 15             | various distances     | 2254168 (1:1:1:1:1:2:2:1:2:3:3:3:3) |
| S10S    | 30             | various distances     | 1500000 (4:4:4:4:4:4:6:6:6:6:6:7:7:7:7:8:8:8:8:9:9:9:9:9:9:9:10:10:10:10:10:10:10) |
| L1      | 2              | Class                  | 176688 (1:1)         |
| L2      | 2              | Class                  | 2595608 (1:2)        |
| L3      | 2              | Class                  | 342348 (1:3)         |
| L4      | 2              | Class                  | 425328 (1:4)         |
| L5      | 2              | Class                  | 508209 (1:5)         |
| L6      | 2              | Class                  | 591089 (1:6)         |

### Table 2 Contingency table of clustering results.

$$\begin{array}{ccccccc}
C_1 & n_{11} & n_{12} & \cdots & n_{1p} \\
C_2 & n_{21} & n_{22} & \cdots & n_{2p} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
C_p & n_{p1} & n_{p2} & \cdots & n_{pp} \\
\end{array}$$

### 4.3 Experiments

The paper conducts various NLP techniques including LDA, NMF, LSA for topic modeling, and Doc2Vec, BERT for sentence embedding. Details of the processing steps are described in Section 3 and the process diagram is described in Fig. 3. Due to the limitation of processing resources and memory, the hyperparameters in the models of NLP embedding have been changed accordingly.

- With Topic modeling methods, the number of the topic is less than 30 with an iteration count of 200.
- With Doc2Vec method, the experiment is used with parameters including: vector-size = 10, window-size = 5, dm-concat = 0 (If 1, use concatenation of context vectors rather than sum/average), dm = 1 (distributed memory' (PV-DM)), dm-mean = None (use the sum of the context word vectors. If 1, use the mean), dbow-words = 0, epochs = 2, dm-tag-count = 5000 and max-vocab-size = 500
- With BERT method, results obtained from experiments with the parameters: hidden = 128, layer = 4, attr-heads = 4, seq-len = 128, batch-size = 64, epochs = 2.

The experiments results in $F$-measure are shown in Table 3, including results extracted from other papers. Note that BiMeta’s performance is re-evaluated in our experiments for direct comparison. A notable thing in our study is that LDA outperforms on S2-S7 datasets. Noted that LDA is quite classical, which has not been used much in NLP. Our LDA approach also works in a stable manner with datasets $S^*$, which have only 2 species, but various abundance ratios. The performance is also quite stable when

### Table 3 $F$-measure of AbundanceBin, MetaCluster, BiMeta, MetaProb (adapted from [1], [2]), and NLP Embedding methods on short-read datasets.

| Dataset | AbundanceBin | MetaCluster | BiMeta | MetaProb | LDA | NMF | LSA | Doc2vec | BERT |
|---------|--------------|-------------|--------|----------|-----|-----|-----|---------|------|
| S1      | 0.883        | 0.617       | 0.990  | 0.991    | 0.99| 0.88| 0.66| 0.7     | 0.90 |
| S2      | 0.713        | 0.631       | 0.738  | 0.901    | 0.74| 0.65| 0.61| 0.74    | 0.66 |
| S3      | 0.824        | 0.415       | 0.909  | 0.928    | 0.99| 0.92| 0.67| 0.79    | 0.96 |
| S4      | 0.885        | 0.460       | 0.994  | 0.908    | 1.00| 0.94| 0.97| 0.99    | 0.97 |
| S5      | 0.552        | 0.643       | 0.803  | 0.832    | 0.89| 0.60| 0.55| 0.57    | 0.67 |
| S6      | 0.692        | 0.492       | 0.996  | 0.970    | 1.00| 0.94| 0.88| 0.69    | 0.97 |
| S7      | 0.606        | 0.652       | 0.814  | 0.782    | 0.83| 0.56| 0.57| 0.81    | 0.68 |
| S8      | 0.528        | 0.529       | 0.683  | 0.169    | 0.57| 0.46| 0.45| 0.63    | 0.81 |
| S9      | -            | 0.639       | 0.697  | 0.719    | 0.68| 0.34| 0.33| 0.51    | 0.42 |
| S10S    | 0.137        | 0.052       | 0.378  | 0.495    | 0.28| -   | -   | 0.16    | 0.23 |
| L1      | 0.625        | 0.549       | 0.974  | 0.984    | 0.98| 0.87| 0.66| 0.68    | 0.96 |
| L2      | 0.793        | 0.675       | 0.802  | 0.992    | 0.80| 0.89| 0.64| 0.80    | 0.97 |
| L3      | 0.900        | 0.667       | 0.824  | 0.933    | 0.95| 0.91| 0.71| 0.85    | 0.84 |
| L4      | 0.979        | 0.703       | 0.977  | 0.956    | 0.89| 0.94| 0.77| 0.70    | 0.75 |
| L5      | 0.977        | 0.612       | 0.982  | 0.983    | 0.98| 0.92| 0.81| 0.89    | 0.74 |
| L6      | 0.984        | 0.649       | 0.983  | 0.984    | 0.98| 0.94| 0.83| 0.91    | 0.73 |
| Avg     | 0.725        | 0.565       | 0.852  | 0.888    | 0.85| 0.84| 0.72| 0.72    | 0.77 |

Precision = \[
\frac{\sum_{j=1}^{p} \max_{i \in \{1, \ldots, p\}} n_{ij}}{\sum_{j=1}^{p} \sum_{i=1}^{p} n_{ij}}
\]

Recall = \[
\frac{\sum_{i=1}^{p} \max_{j \in \{1, \ldots, p\}} n_{ij}}{\sum_{j=1}^{p} \sum_{i=1}^{p} n_{ij}}
\]

F-measure = \[
\frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}
\]
NMF applied to L*. In our implementation, employing other advanced techniques (e.g., TF-IDF) in LDA and NMF improves their performance. Table 3, overall performance is quite close to MetaProb and comparative with those of BiMeta, and overall NLP techniques are more viable than AbundanceBin, MetaCluster. It is quite interesting that Doc2Vec seems to have between |

| Table 4 | Precision, Recall of NLP embedding methods on short-read datasets. |
|---------|-------------|----------------|-------------|----------------|-------------|
|         | LDA         | Doc2Vec       | BERT        |               |             |
| Dataset | Prec        | Recall        | Prec        | Recall        | Prec        | Recall        |
| S1      | 1.00        | 0.84          | 1.00        | 0.56          | 0.96        | 0.96          |
| S2      | 0.99        | 0.99          | 1.00        | 0.50          | 0.96        | 0.96          |
| S3      | 0.99        | 0.99          | 0.99        | 0.49          | 0.97        | 0.97          |
| S4      | 0.99        | 0.99          | 0.99        | 0.49          | 0.97        | 0.97          |
| S5      | 0.99        | 0.99          | 0.99        | 0.49          | 0.97        | 0.97          |
| S6      | 0.99        | 0.99          | 0.99        | 0.49          | 0.97        | 0.97          |
| S7      | 0.99        | 0.99          | 0.99        | 0.49          | 0.97        | 0.97          |
| S8      | 0.99        | 0.99          | 0.99        | 0.49          | 0.97        | 0.97          |
| S9      | 0.99        | 0.99          | 0.99        | 0.49          | 0.97        | 0.97          |
| S10      | 0.99        | 0.99          | 0.99        | 0.49          | 0.97        | 0.97          |
| S11      | 0.99        | 0.99          | 0.99        | 0.49          | 0.97        | 0.97          |
| S12      | 0.99        | 0.99          | 0.99        | 0.49          | 0.97        | 0.97          |
| S13      | 0.99        | 0.99          | 0.99        | 0.49          | 0.97        | 0.97          |
| S14      | 0.99        | 0.99          | 0.99        | 0.49          | 0.97        | 0.97          |
| S15      | 0.99        | 0.99          | 0.99        | 0.49          | 0.97        | 0.97          |
| S16      | 0.99        | 0.99          | 0.99        | 0.49          | 0.97        | 0.97          |
| S17      | 0.99        | 0.99          | 0.99        | 0.49          | 0.97        | 0.97          |
| S18      | 0.99        | 0.99          | 0.99        | 0.49          | 0.97        | 0.97          |
| S19      | 0.99        | 0.99          | 0.99        | 0.49          | 0.97        | 0.97          |
| S20      | 0.99        | 0.99          | 0.99        | 0.49          | 0.97        | 0.97          |

5. Conclusion

The embedding process is one of the key factors affecting the overall performance of unsupervised binning algorithms. The paper proposed a novel framework using NLP techniques for this process. We conducted experiments on various NLP techniques, belonging to topic modeling sentence embedding. The results showed that LDA and NMF of topic modeling worked quite efficiently and achieved good results on simulated short-read datasets. Sentence embedding methods (Doc2Vec and BERT) did not provide better performances than BiMeta and MetaProb. It is mainly because vocabulary and dataset are not large compared with datasets on human languages. However, it is quite encouraged to continue the application of them if more metagenomic datasets are collected and much biological knowledge is used in building feature extraction. Our proposed framework could be useful to exploit further embedding techniques not only in NLP but also by the combination of metagenomic embeddings and NLP embeddings.

Acknowledgments This research is funded by Vietnam National University Ho Chi Minh City (VNU-HCM) under grant number 2B2019-20-06.

References

[1] Girotto, S., Pizzi, C. and Comin, M.: MetaProb: Accurate metagenomic reads binning based on probabilistic sequence signatures, Bioinformatics, Vol.32, No.17, pp.567–575 (2016).
[2] Vinh, L., Lang, T., Binh, L. and Hoai, T.: A two-phase binning algorithm using 1-mer frequency on groups of non-overlapping reads, Algorithms Mol. Biol., Vol.10, No.2 (2015).
[3] Blei, D., Ng, A. and Jordan, M.: Latent dirichlet allocation, Journal of Machine Learning Research, Vol.3, pp.993–1022 (2003).
[4] Lee, D.D. and Seung, H.S.: Learning the parts of objects by non-negative matrix factorization, Nature, Vol.401, No.6755, pp.788–791 (1999).
[5] Deerwester, S., Dumais, S., Landauer, T., Furnas, G. and Harshman, R.: Indexing by latent semantic analysis, J. Am. Soc. Inf. Sci., Vol.41, pp.391–407 (1990).
[6] Le, Q. and Mikolov, T.: Distributed representations of sentences and documents, Proc. 31st International Conference on International Conference on Machine Learning, ICML'14, Vol.32, pp.II-1188-II-1196, JMLR.org (2014).
[7] Juric, I., Chang, M.-W., Lee, K. and Toutanova, K.: Well-read students learn better: On the importance of pre-training compact models, arXiv preprint arXiv:1908.08962v2 (2019).
[8] Mende, S.S., Mohammed, M.H. and Ghosh, T.S.: Classification of metagenomic sequences: Methods and challenges, Briefings in Bioinformatics, Vol.13, No.6, pp.669–681 (2012).
[9] Breitwieser, F.P., Lu, J. and Salzberg, S.L.: A view of methods and databases for metagenomic classification and assembly, Briefings in Bioinformatics, Vol.20, No.4, pp.1125–1136 (2017).
[10] Teeling, H., Waldmann, J., Lombardot, T., Bauer, M. and Glöckner, F.O.: Tetra: A web-service and a stand-alone program for the analysis and comparison of tetranucleotide usage patterns in dna sequences, BMC Bioinformatics, Vol.5, No.163 (2004).
[11] Kishyuk, A., Bhatnagar, S., Dushoff, J. and Weitz, J.S.: Unsupervised statistical clustering of environmental shotgun sequences, BMC Bioinformatics, Vol.10, No.1, (2009).
[12] Kelley, D. and Salzberg, S.: Clustering metagenomic sequences with interopolated markov models, BMC Bioinformatics, Vol.11, No.544 (2010).
[13] Laczný, C.C., Sternal, T., Plugar, V., Gavron, P., Atashpazgarad, A. and Margossian, H.: Vizbin - An application for reference-independent visualization and human-augmented binning of metagenomic data, Microbiome, Vol.3, No.1 (2015).
[14] Wu, Y.-W. and Ye, Y.: A novel abundance-based algorithm for binning metagenomic sequences using 1-tupleless, Journal of Computational Biology, Vol.18, No.3, pp.523–534 (2011).
[15] Chatterji, S., Yamazaki, I., Bai, Z. and Eisen, J.: Composbin: A DNA composition-based algorithm for binning environmental shotgun reads, Lecture Notes in Computer Science, Vol.4955, pp.17-28 (2008).
[16] Alneberg, J., Bjarnason, B.S., De Bruijn, I., Schirmer, M., Quick, J. and IJaz, U.Z.: Binning metagenomic contigs by coverage and com-
position, Nature Methods, Vol.11, No.11, pp.1144–5 (2014).

[17] Lu, Y.Y., Chen, T., Fuhrman, J.A. and Sun, F.: Cocacola: Binning metagenomic contigs using sequence composition, read coverage, co-alignment and paired-end read linkage, Bioinformatics, Vol.33, No.6, pp.791–798 (2017).

[18] Wu, Y.-W., Tang, Y.-H., Tringe, S.G., Simmons, B.A. and Singer, S.W.: Maxbin: An automated binning method to recover individual genomes from metagenomes using an expectation-maximization algorithm, Microbiome, Vol.2, No.1, (2014).

[19] Kang, D.D., Froula, J., Egan, R. and Wang, Z.: Metabat - An efficient tool for accurately reconstructing single genomes from complex microbial communities, PeerJ, Vol.3, No.e1165 (2015).

[20] Lin, H.-H. and Liao, Y.-C.: Accurate binning of metagenomic contigs via automated clustering sequences using information of genomic signatures and marker genes, Sci. Rep. (2016).

[21] Li, Q., Peng, H., Li, J., Xia, C., Yang, R., Sun, L., Yu, P.S. and He, L.: A survey on text classification: From shallow to deep learning, IEEE Trans. Neural Networks and Learning Systems, Vol.31, No.11, pp.1–21 (2020).

[22] Liu, L., Tang, L., Dong, W., YaoEmail, S. and ZhouEmail, W.: An overview of topic modeling and its current applications in bioinformatics, SpringerPlus, Vol.5, No.1 (2016).

[23] Chen, X., Hu, X., Shen, X. and Rosen, G.: Probabilistic topic modeling for genomic data interpretation, IEEE International Conference on Bioinformatics and Biomedicine (BIBM) (2011).

[24] Chen, X., He, T., Hu, X., Zhou, Y. and An, Y., et al.: Estimating functional groups in human gut microbiome with probabilistic topic models, IEEE Trans. NanoBioscience, Vol.11, No.3, pp.203–215 (2012).

[25] Chen, X., Hu, X., Lim, Y.Y. and Shen, X.: Exploiting the functional and taxonomic structure of genomic data by probabilistic topic modeling, IEEE/ACM Trans. Computational Biology and Bioinformatics, Vol.9, No.4, pp.980–991 (2012).

[26] La Rosa, M., Fiannaca, A., Rizzo, R. and Urso, A.: Probabilistic topic modeling for the analysis and classification of genomic sequences, BMC Bioinformatics, Vol.16, No.S2 (2015).

[27] Zhang, R., Cheng, Z., Guan, J. and Zhou, S.: Exploiting topic modeling to boost metagenomic reads binning, BMC Bioinformatics, Vol.16, No.5, pp.1–10 (2015).

[28] Wang, Y., Leung, H.C.M., Yiu, S.M. and Chin, F.Y.L.: Metacluster 5.0: A two-round binning approach for metagenomic data for low-abundance species in a noisy sample, Bioinformatics, Vol.28, No.18, pp.356–362 (2012).

[29] Liao, R., Zhang, R., Guan, J. and Zhou, S.: A new unsupervised binning approach for metagenomic sequences based on n-grams and automatic feature weighting, IEEE/ACM Trans. Computational Biology and Bioinformatics (TCBB), Vol.11, No.1, pp.42–54 (2014).

[30] Albalawi, R., Yeap, T.H. and Benyoucef, M.: Using topic modeling methods for short-text data: A comparative analysis, Frontiers in Artificial Intelligence, Vol.3, No.42 (2020).

[31] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L. and Polosukhin, I.: Attention is all you need, NIPS 2017 (2017).

[32] Richter, D.C., Ott, F., Auch, A.F., Schmid, R. and Husein, D.H.: Metasim - A sequencing simulator for genomics and metagenomics, PLoS ON, Vol.3, No.10, (2008).

Viet Toan Tran received his bachelor degree in Computer Engineering from Ho Chi Minh University of Technology, VNU-HCM in 2020. His research interests include data mining, solving real-world problems in the urban environment.

Phuong V.D. Van is currently a lecturer at Lac Hong University, Vietnam. She is also a Ph.D. student at Ho Chi Minh City University of Technology, VNU-HCM. She graduated with a Master of Information Technology in 2011. She is interested in studying bioinformatics, the theory of database optimization. In addition, she is working in Quality Assurance in Higher Education.

Van Hoai Tran is an associate professor at Ho Chi Minh University of Technology - VNU-HCM. He received his Ph.D. degree of applied mathematics from Heidelberg University in 2005. His theoretical background is combinatorial optimization. After graduation, he has initiated and joined different projects to apply his knowledge on optimization to various fields. His research interests are metagenomic binning in bioinformatics, convexity in computational geometry and industrial scheduling.

(Communicated by Susumu Goto)