Dynamic Contrastive Learning with Pretrained Deep Language Model Enhances Metagenome Binning for Contigs

Bohao Zou[1†], Zhenmiao Zhang[1,2†], Rong Tao[1], Nianzhen Gu[3]
Lu Zhang[1 *]
[1]Dept of Computer Science, Hong Kong Baptist University, Kowloon Tong, Hong Kong
[2]Dept of Computer Science & Engineering, UC San Diego, CA 92122, USA
[3]Dept of Biosystems Science & Engineering, ETH Zurich,
Klingelbergstrasse 48, 4056 Basel, Switzerland

Abstract

The recovery of microbial genomes from large metagenomic datasets is a critical step for identifying uncultivated microbial populations and explaining their functional roles. This process requires metagenomic binning, which involves clustering assembled contigs into metagenome-assembled genomes (MAGs). However, existing computational binning methods often fail to leverage contigs' sequence context effectively. Most of them cannot incorporate taxonomic knowledge from external databases, and insufficient training data frequently limits the deep learning-based binning methods. To address these challenges, we propose CompleteBin, an innovative pretrained deep language model enhanced by dynamic contrastive learning. CompleteBin trains a pretrained deep language model using both long and short contigs and clusters the contigs based on their embeddings generated by the language model. This approach uniquely integrates the sequence context of contigs, incorporates taxonomic knowledge from external databases, and leverages unlimited data during training to significantly improve binning accuracy. Compared to state-of-the-art binning tools such as CONCOCT, MetaBAT2, SemiBin2, and COMEBin, CompleteBin achieves higher performance across both simulated datasets (CAMI I and II) and diverse real-world datasets, including ocean, plant, freshwater, and human fecal samples. On average, CompleteBin achieves a 39.7% improvement on simulated datasets and a 58.9% improvement on real-world datasets in terms of the number of near-complete MAGs when compared to the best-performing results of other binning methods. These results highlight CompleteBin's robust capability to recover high-quality MAGs from long and short contigs, establishing it as a powerful tool for advancing metagenomic research. CompleteBin is open-source and available at https://github.com/zoubohao/CompleteBin.

*Corresponding Author. Email: ericluzhang@hkbu.edu.hk

1 Introduction

Metagenomic sequencing has emerged as a widely adopted technique to overcome the challenges associated with isolating and cultivating certain microorganisms in laboratory settings [1, 2, 3]. This approach leverages whole environmental samples as sequencing inputs, enabling a comprehensive understanding of both cultivable and uncultivable microbial communities [4]. The standard workflow for reconstructing metagenome-assembled genomes (MAGs) from sequencing reads comprises two primary computational steps: (1) metagenomic assembly, which generates non-overlapping continuous sequences (contigs) from raw reads [5], and (2) metagenomic binning, which groups contigs that originated from the same or closely related taxonomic groups, ultimately forming MAGs [5]. Achieving reliable and high-performance binning results for metagenomic research has become a critical area of investigation.

Numerous statistical binning methods like MaxBin [6], MetaBAT2 [7], and MetaDecoder [8] have been developed by utilizing read depths and tetranucleotide frequencies (TNFs) as sequence composition features. Read depth is obtained by aligning the reads to the contigs and calculating the average depth of each contig, while TNFs represent the frequencies of all 4-mers (subsequences of length 4) for each contig. MaxBin employs expectation maximization for probabilistic modeling of read depths and TNFs. MetaBAT2 constructs a similarity graph based on scores derived from these features, followed by graph partitioning via label propagation. The recently introduced MetaDecoder applies a modified Dirichlet-Gaussian mixture model to generate preliminary clusters based on read depths and TNFs. It then refines these clusters using a semi-supervised probabilistic model and a modified Gaussian mixture model for TNFs and read depths.

In addition to statistical methods, deep learning-based approaches like VAMB [9], SemiBin2 [10], and COMEBin [11] have also been developed for metagenomic binning with read depths and TNFs. VAMB reduces the dimensionality of read depths and TNFs to obtain lower-dimensional latent embeddings through a variational autoencoder, followed by a density-based latent clustering algorithm to group contigs. However, VAMB does not incorporate single-copy genes (SCGs) to guide clustering. SemiBin2 constructs pairwise must-link constraints by splitting long contigs into equal-length segments and introduces cannot-link constraints via random sampling of contig pairs. It employs a semi-supervised autoencoder with contrastive learning to extract constraint information and obtain contig embeddings. SemiBin2 employs the Infomap algorithm for clustering and incorporates SCGs to guide the clustering process. COMEBin is a greatly improved version of SemiBin2. It generates multiple must-link constraints for each contig and applies contrastive multi-view representation learning with a combined network to obtain contig embeddings. COMEBin employs the Leiden algorithm [12] for clustering and incorporates SCGs to guide the clustering step.

Although deep learning-based methods have shown more promising performance compared to traditional statistical methods due to their ability to learn complex data patterns that are challenging to model manually [13], several challenges continue to hinder their widespread adoption across various applications. Most existing binning methods, such as MetaBAT2 and VAMB, rely on TNF as the primary sequence feature for contigs. However, TNF fails to capture the sequence context of contigs, which can be significantly altered by genomic variations, changes in gene order, and genome rearrangements. Incorporating these genomic alterations into the sequence embedding could enhance the resolution and reliability of distinguishing genomic sequences from different species [14, 15, 16]. Furthermore, despite the growing availability of microbial reference genomes [17, 18, 19], most binning methods do not leverage this external knowledge to improve contig binning accuracy. Instead, they depend exclusively on raw sequence data and read depths from metagenomic sequencing

samples. As a result, these methods fail to utilize valuable reference genomes and their taxonomic lineages, which could substantially improve clustering accuracy. Finally, a significant limitation of deep learning-based binning methods is their reliance on the volume of training data. These methods are constrained to using raw contigs from individual samples for training. Only a few approaches, such as SemiBin2 and COMEBin, generate limited pairwise training data. Expanding the volume and diversity of training data for deep learning-based binning methods could mitigate overfitting and improve the quality of contig embeddings.

In this study, we present CompleteBin, a novel pretrained deep language model enhanced with a dynamic contrastive learning algorithm for metagenomic binning. CompleteBin trains a pretrained deep language model with dynamic contrastive learning and then clusters the contigs with their embeddings through the Leiden and FLSpp [20] algorithms. During contrastive training, CompleteBin leverages both TNF and the sequence context of contigs as the inputs of the model. CompleteBin extracts the sequence context of contigs by sequence patch embedding (**Methods**), which is inspired by the patch embedding in the vision Transformer (ViT) [21]. To incorporate knowledge in external reference genomes, CompleteBin pretrains half of the model layers with reference genomes and their taxonomic lineages (**Methods**). These layers remain fixed during contrastive training, generating taxonomic embeddings for simultaneous use in the contrastive training procedure. To address the limitation of training data volume, CompleteBin employs dynamic data augmentation during contrastive training (**Methods**) rather than relying on a fixed set of augmentations performed beforehand. This approach effectively increases the training data volume from finite to infinite and enhances the diversity of the training data. Thereby enhancing the model's representation capabilities and mitigating overfitting. During the clustering, CompleteBin initially applies the Leiden algorithm with varying hyperparameter settings to cluster contigs based on SCGs and their embeddings (**Methods**). The FLSpp algorithm is then used to refine these clusters further (**Methods**). Finally, clustering results from different hyperparameter settings are ensembled, and Galah [22] is employed to identify and remove duplicate MAGs (**Methods**).

We extensively benchmarked CompleteBin against leading binning tools, including CONCOCT, MetaBAT2, SemiBin2, and COMEBin. The evaluation was conducted on the CAMI I high-complexity, CAMI II marine, and plant simulation datasets, as well as various real-world environmental datasets (human feces, freshwater, marine, and plant). CompleteBin achieves the highest number of near-complete MAGs (NCMAGs; completeness ≥ 90 and contamination ≤ 5; **Methods**) compared to state-of-the-art (SOTA) metagenome binning tools. When compared to the best-performing results of other binning methods (the top results from different tools across datasets), CompleteBin demonstrates an average improvement of 39.7% and 58.9% on simulated and real-world datasets, respectively. On the CAMI II marine datasets with multi-sample binning, CompleteBin achieves an average improvement of 31.0% over the best-performing alternative methods. Additionally, CompleteBin achieves the highest reference genome recovery rate compared with other binning tools on CAMI-simulated datasets.

2 Results

2.1 Architecture of CompleteBin and binning workflow

CompleteBin is a pretrained deep language model designed to reconstruct MAGs from both long and short contigs (≥ 900 base pairs). The overview framework of CompleteBin is illustrated in **Figure**

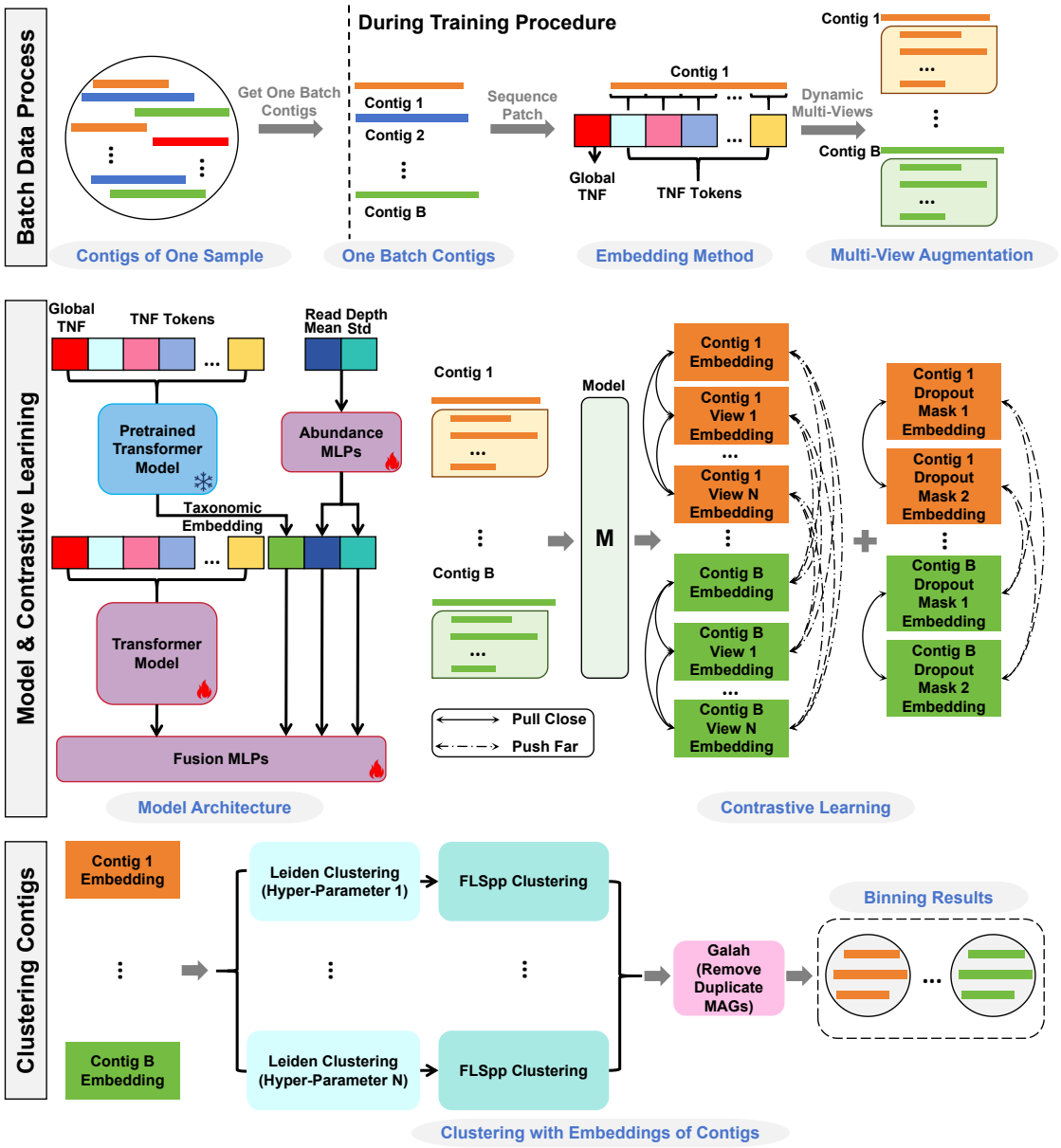

Figure 1: The Overview framework of CompleteBin. CompleteBin dynamically generates multi-views of contigs during the training procedure and extracts the TNF tokens using sequence patch embedding. Taxonomic embeddings for contigs are generated using a pretrained model and concatenated with sequence composition features and read depth statistics, such as mean and standard deviation (Std), into fusion multi-layer perceptrons (MLPs). During contrastive training, the parameters of the pretrained model remain fixed, while the other parameters are updated. After training, CompleteBin applies the Leiden and FLSpp clustering methods to group contigs. Then, CompleteBin uses Galah to ensemble the results and remove duplicate MAGs.

1. Its primary network architecture is based on the Transformer model [23]. Unlike other binning tools that solely rely on TNFs as sequence features for contigs, CompleteBin introduces a novel approach by dividing contigs of any length into a fixed number of equally sized sequence patches and calculating the TNFs for each patch. Subsequently, CompleteBin concatenates these local TNF features in sequential order with the global TNFs to construct the sequence features of contigs. This sequence patch embedding method integrates the global nucleotide composition, local nucleotide patterns, and sequence context of the contigs. Combining these elements ensures a comprehensive representation of the sequence, capturing both the overall structure and local variations within the contig. Furthermore, half of the layers in CompleteBin are pretrained using reference genomes and their taxonomic annotations. These pretrained layers remain fixed during contrastive training and generate taxonomic embeddings for contigs based on their sequence features. CompleteBin explicitly integrates taxonomic knowledge from external databases into the contrastive training process by incorporating these embeddings. Additionally, CompleteBin employs multi-layer perceptrons (MLPs) to extract statistical features, such as mean and standard deviation (Std), from read depths of contigs. Finally, the model concatenates the sequence features, taxonomic embeddings, and extracted read depth features, which serve as the input for further training.

During training, CompleteBin employs dynamic multi-view and self-dropout [24] contrastive learning strategies. Instead of relying on a fixed set of predefined data augmentations for contigs, CompleteBin dynamically generates multiple views of each contig during contrastive learning. This dynamic multi-view approach allows the model to learn almost infinite subsequences for each contig, enhancing its representation capabilities and reducing the risk of overfitting. The multiple views of a contig are treated as positive pairs, while other contigs within the same batch are treated as negative pairs. Furthermore, CompleteBin considers embeddings of a contig generated with different dropout masks as positive pairs, while other contigs in the batch are treated as negative pairs. CompleteBin encourages higher similarity between the embedding of positive pairs compared to negative pairs.

After training, CompleteBin performs a series of clustering steps to group the contigs. First, it applies the Leiden community detection algorithm with varying hyperparameter settings to generate initial clusters. The quality of these clusters is evaluated using an approach similar to that described in MetaBinner [25]. Next, CompleteBin refines highly contaminated clusters (with contamination levels exceeding 5%) using the FLSpp algorithm. Finally, CompleteBin aggregates the clustering results obtained from different parameter settings and employs Galah to identify and remove replicated MAGs.

2.2 Superior performance of CompleteBin on CAMI simulated datasets

We applied CONCOCT, MetaBAT2, SemiBin2, COMEBin, and CompleteBin on CAMI I high-complexity [26], CAMI II marine (10 samples) [27], and CAMI II plant (21 samples) [27] datasets. We evaluated the binning performance by the number of recovered NCMAGs as in the previous works [28].

For the CAMI I high-complexity dataset, CompleteBin significantly outperforms other binning methods. It increases over 182 (568.7%), 88 (69.8%), 80 (59.7%), and 32 (20.2%) NCMAGs more than CONCOCT, MetaBAT2, SemiBin2, and COMEBin, respectively (**Figure 2a**). Furthermore, CompleteBin consistently produces the highest number of MAGs with low contamination ($\leq$ 5%) compared to other tools (**Figure 2a**). Additionally, CompleteBin generates 457 MAGs that are above medium-quality criteria (completeness $\geq$ 50% and contamination $\leq$ 10%; **Methods**), while CONCOCT, MetaBAT2, SemiBin2, and COMEBin only produce 67, 257, 282, and 324 MAGs above

medium-quality criteria, respectively (**Figure 2d**). This observation highlights the efficiency of CompleteBin in recovering NCMAGs from high-complexity samples.

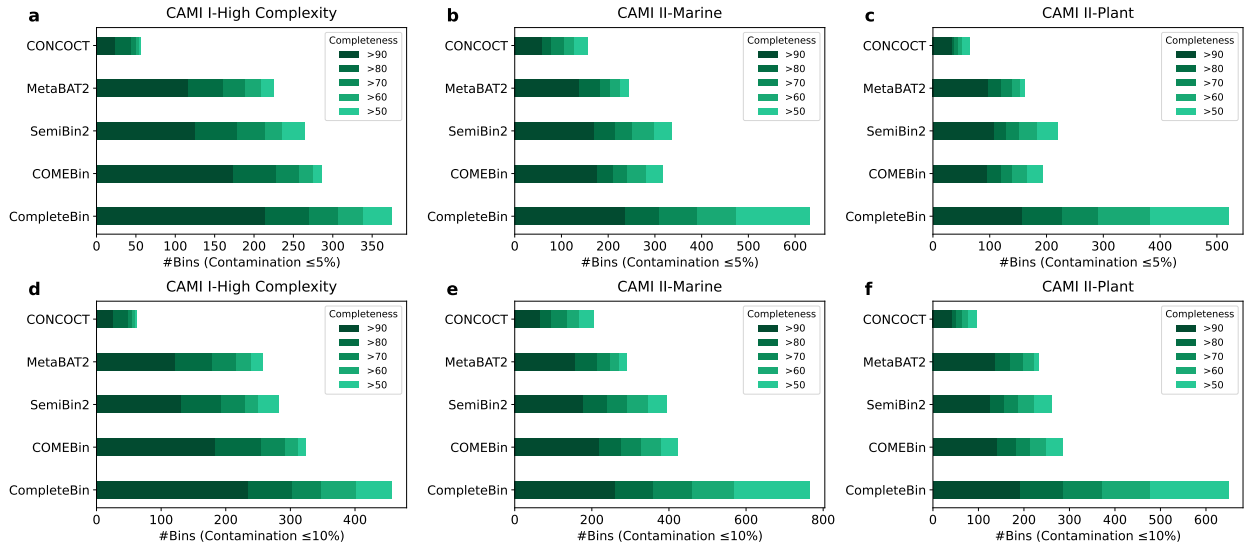

Figure 2: Comparison of binning methods on the CAMI I high-complexity (a, d), CAMI II marine (b, e), and CAMI II plant (c, f) simulated datasets. The number of recovered bins with contamination thresholds of $\leq 5\%$ and $\leq 10\%$ is shown for varying completeness thresholds.

For the CAMI II challenge datasets, CompleteBin demonstrates superior performance on both the marine and plant datasets, as shown in **Figure 2b** and **c**. On the marine dataset, it increases over 178 (306.8%), 98 (71.0%), 72 (43.9%), and 61 (34.8%) NCMAGs more than CONCOCT, MetaBAT2, SemiBin2, and COMEBin, respectively (**Figure 2b**). Similarly, on the plant dataset, CompleteBin increases over 101 (303.1%), 37 (37.2%), 28 (25.6%), and 40 (41.6%) NCMAGs more than CONCOCT, MetaBAT2, SemiBin2, and COMEBin, respectively (**Figure 2c**). Furthermore, CompleteBin consistently produces the highest number of MAGs with low contamination ($\leq 5\%$) across both datasets (**Figure 2b** and **c**). Additionally, CompleteBin deeply generates 766 and 650 MAGs that are above medium quality from the marine and plant datasets, respectively. In contrast, CONCOCT, MetaBAT2, SemiBin2, and COMEBin produce only 206 and 96, 292 and 233, 394 and 260, 423 and 284 MAGs above medium-quality from marine and plant datasets, respectively (**Figure 2e** and **f**). These results highlight CompleteBin's superior ability to recover MAGs from complex environmental samples compared to other binning methods.

2.3 CompleteBin demonstrates effectiveness with different binning modes

To further evaluate the effectiveness of CompleteBin with multi-sample binning mode (**Supplementary Note 2.2**), we conducted a comparative analysis of its performance against other binning methods on the CAMI II marine dataset, which includes 10 samples. The results indicate that CompleteBin outperforms all other methods in the multi-sample binning mode (**Figure 3**). Specifically, CompleteBin generates 81 (49.3%), 125 (104.2%), 57 (30.3%), 93 (61.1%), and 58 (31.0%) more NCMAGs compared to VAMB, CONCOCT, MetaBAT2, SemiBin2, and COMEBin, respectively (**Figure 3a**). This improvement demonstrates the substantial advantage of CompleteBin in accurately recovering

high-quality MAGs with multi-sample binning mode. Furthermore, CompleteBin consistently produces the highest number of MAGs with low contamination ($\leq 5\%$) across the dataset, significantly outperforming its competitors (**Figure 3a**). Additionally, CompleteBin generates 736 MAGs above medium-quality criteria, whereas VAMB, CONCOCT, MetaBAT2, SemiBin2, and COMEBin produce only 315, 389, 413, 442, and 507 MAGs above medium-quality criteria, respectively (**Figure 3b**). This remarkable improvement highlights the advanced capabilities of CompleteBin in recovering both near-complete and medium-quality MAGs with multi-sample binning mode.

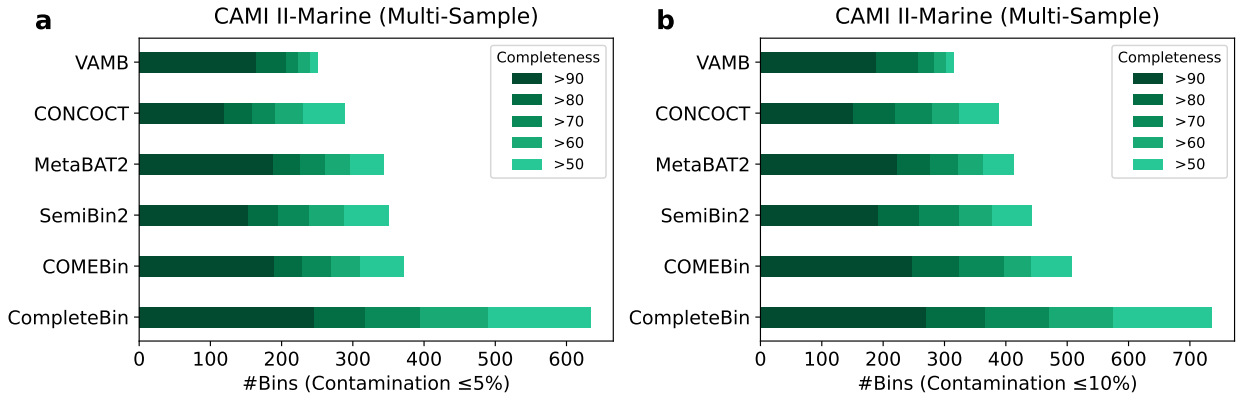

Figure 3: Comparison of binning methods on the CAMI II marine simulated datasets with multi-sample binning mode. The number of recovered bins with contamination thresholds of $\leq 5\%$ and $\leq 10\%$ is shown for varying completeness thresholds.

2.4 CompleteBin outperforms other binning methods on complex real datasets

In this study, we comprehensively evaluated the performance of CompleteBin on highly complex and diverse real-world metagenomic sequencing datasets, with the primary objective of assessing its effectiveness in enhancing the number of NCMAGs. The datasets utilized in this analysis are derived from a range of complex environmental contexts, including ocean ecosystems (three samples) [29], plant environments (one sample) [30], freshwater habitats (three samples) [31, 32, 33], and human fecal samples (one sample) [34]. CompleteBin consistently outperforms other binning methods across all datasets, as demonstrated in **Figure 4**. Across all datasets, CompleteBin demonstrated a substantial advantage in recovering NCMAGs, outperforming CONCOCT, MetaBAT2, SemiBin2, and COMEBin by 287 (1,594.4%), 197 (182.4%), 192 (169.9%), and 113 (58.8%) additional NCMAGs, respectively. (**Figure 4**). These results highlight the substantial advantage of CompleteBin in recovering NCMAGs from highly complex metagenomic datasets derived from various environmental conditions. Moreover, CompleteBin consistently produced the highest number of MAGs with low contamination ($\leq 5\%$) across all the datasets, significantly outperforming competing methods (**Figure 4**). In addition, CompleteBin successfully generated a total of 1415 MAGs above medium-quality criteria, a markedly superior performance compared to the 134, 430, 465, and 414 MAGs above medium-quality criteria, which were produced by CONCOCT, MetaBAT2, SemiBin2, and COMEBin, respectively (**Figure 5**). This significant improvement highlights the enhanced capabilities of CompleteBin in accurately recovering both near-complete and medium-quality MAGs from highly complex environmental datasets.

CompleteBin has the capacity to effectively capture the sequence context of short contigs (900 bps - 1500 bps). Such contigs are typically challenging for other binning methods to process since their TNF features are highly variable and difficult to accurately cluster. In contrast, CompleteBin demonstrates a remarkable capacity to overcome these difficulties by leveraging its advanced mechanisms. CompleteBin can perform more precise, reliable, and robust binning for these short contigs, which may contain biologically significant evidence.

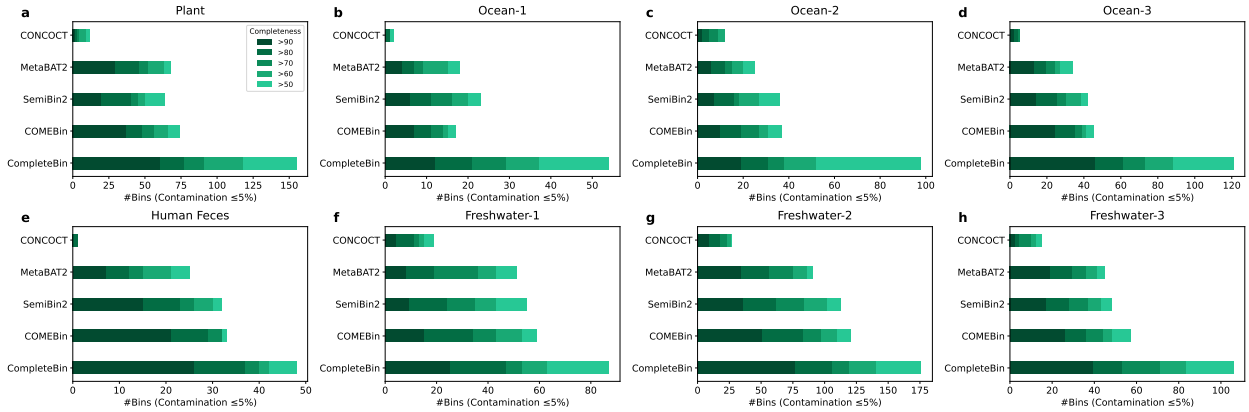

Figure 4: Comparison of binning methods on the real-world datasets. The number of recovered bins with contamination thresholds of $\leq 5\%$ is shown for varying completeness thresholds.

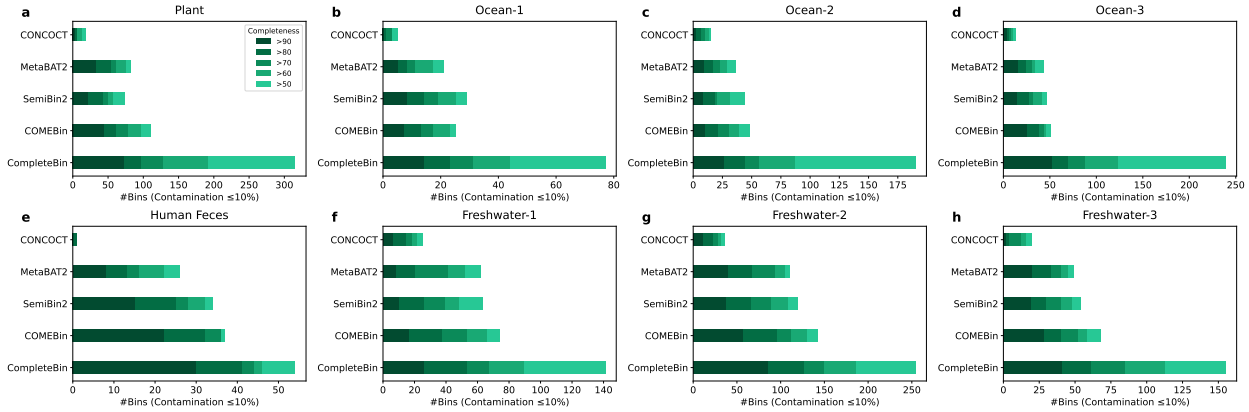

Figure 5: Comparison of binning methods on the real-world datasets. The number of recovered bins with contamination thresholds of $\leq 10\%$ is shown for varying completeness thresholds.

3 Discussion

CompleteBin is a pretrained deep language model that incorporates dynamic contrastive learning for the task of metagenomic binning. In contrast to traditional binning methods, CompleteBin leverages the global and local TNF as model input, which can effectively capture the sequence context and the gene or genome variation within contigs. Moreover, the sequence patch embedding allows

CompleteBin to effectively process very short contigs (900 bps - 1000 bps), which are challenging for other binning methods to process. These short contigs may contain genes associated with specific diseases, and they are frequently overlooked by conventional binning methods due to their size and the inherent complexity of accurately clustering them. In addition, sequence patch embedding effectively addresses the challenge of embedding contigs of varying lengths. An interesting property of the sequence patch embedding is that, when the patch size equals the length of the contig, the resulting embedding becomes equivalent to a one-hot encoding of the sequence.

Leveraging the advantages of the sequence patch embedding, CompleteBin is the first binning method to learn representations of contigs in a manner of natural language processing, utilizing a Transformer-based deep language model (DLM) for metagenomic binning, whereas other binning tools rely on MLPs. DLMs outperform MLPs in tasks involving sequence data, long-range dependencies, large datasets, and complex relationships. The attention architecture in DLM offers excellent flexibility, scalability, and the ability to learn more expressive representations compared to the fixed architecture of MLPs.

CompleteBin pretrains the Transformer model using external reference genomes and their taxonomic annotations. This pretrained model generates taxonomic embeddings for contigs and explicitly incorporates them during contrastive learning. Notably, traditional binning methods require contig alignment to reference databases to incorporate the external knowledge of reference genomes. This process is both time-consuming and inflexible. In contrast, CompleteBin's pretrained model eliminates the need for alignment, enabling the integration of external reference genome knowledge into the binning process more efficiently and with greater flexibility.

CompleteBin employs dynamic multi-view and self-dropout contrastive learning strategies for contrastive training. These two strategies align with two fundamental principles of self-supervised learning: *Alignment* and *Uniformity*. Alignment refers to the mapping of positive pairs to similar feature representations, while Uniformity refers to a feature distribution that maximizes information preservation, specifically the uniform distribution on the unit hypersphere. The dynamic multi-view strategy primarily aims to enhance the alignment between positive pairs. It is based on the assumption that contigs and their subsequences originate from the same genome, and their embeddings must be similar. Furthermore, the self-dropout strategy focuses on improving uniformity within the embedding space to maximize information preservation. This strategy is based on the assumption that the same contigs with different dropout masks must be derived from the same genome, and their embeddings must be similar.

CompleteBin employs multiple clustering algorithms to effectively group contigs. First, the Leiden algorithm is applied to cluster contigs using their embeddings, lengths, and SCGs. Subsequently, the FLSpp algorithm refines clusters with high contamination levels, as estimated based on SCGs. Finally, CompleteBin aggregates clustering results from different hyperparameter settings and removes duplicate MAGs using Galah. This ensemble binning strategy integrates clustering results from diverse perspectives, improving the resolution and accuracy of metagenomic binning.

Throughout the CompleteBin pipeline, four distinct modalities are utilized: contig sequences, read depths, taxonomic embeddings, and single-copy genes. By integrating these diverse data types, CompleteBin enhances its ability to recover metagenome-assembled genomes with greater accuracy and precision, significantly advancing metagenomic research.

4 Methods

4.1 Generate tetra nucleotide frequencies (TNFs) vector

A sliding window of length k with a stride of one was applied to traverse the sequence, generating multiple k-mer fragments. The frequency distribution of these k-mers was then calculated, which resulted in a vector with T dimensions. In our binning algorithm, each k-mer and its reverse complement were treated as equivalent, and we set k to 4 in this study. The specific formulas used in the calculation are provided below.

$$F' = (f_1, ..., f_j, ..., f_{136}) \tag{1}$$

where f_j represents the frequency of the j-th k-mer feature. We further normalized the frequency distribution F' with:

$$q_j = \frac{f_j}{\sum_{i=1}^{136} f_i} \tag{2}$$

Finally, the TNF vector could be calculated as:

$$F = (q_1, ..., q_j, ...q_{136}) \tag{3}$$

4.2 Construct sequence patch embedding

The sequence was partitioned into multiple fixed patches of equivalent lengths, and a TNF vector was calculated independently for each patch to capture local compositional patterns. The TNF vectors corresponding to these patches were then concatenated in their original sequence order, preserving both the local variations and the structural context of the sequence. Furthermore, a global TNF vector, representing the overall trinucleotide composition of the entire sequence, was prepended to the ordered TNF vectors to provide a comprehensive representation of the sequence's compositional features. The sequence patch embedding for a contig could be calculated as:

$$S = [F_{global}, F_{p_1}, ..., F_{p_j}, ..., F_{p_l}] \tag{4}$$

where F_{p_j} represents the TNF vector of the j-th patch, l is the number of patches. We set l as 16 in our study.

4.3 Prepare and process reference microbial genomes

Reference genomes were downloaded from the proGenomes v3.0 database [35], and their complete taxonomic lineages were retrieved from NCBI. To ensure data quality, we initially counted the number of species in each phylum and applied a filter to exclude phyla containing fewer than 10 species. Genomes with identical taxonomy IDs (txid) were removed, retaining only the genome with the largest size. Genomes from the same species but different strains were also filtered, preserving only the largest genome for each species. After this processing, we collected 15,780 distinct representative genomes from different species and ensured the inclusion of comprehensive genetic information.

4.4 Pretrain Transformer model

We pretrained half of the layers of the Transformer model using reference microbial genomes and their taxonomic annotations. During pretraining, sequences were randomly sampled from reference genomes. The length ranges were from 768 to 60,000 base pairs with a uniform distribution. The sampled sequences were used with the sequence patch embedding and fed them into the model for training. Each reference genome was treated as a unique class, resulting in 15,780 classes for classification. The Transformer model was pretrained over 128 epochs with a batch size of 256 and a learning rate $1e^{-5}$ with a cosine annealing schedule.

4.5 Contrastive Learning

Dynamic multi-view contrastive learning

CompleteBin generated multi-view (subsequences) for each contig in one batch of data during the contrastive training. This optimization process pulled the representations of a contig and its generated subsequences closer together while pushing the representations of other contigs and their subsequences in the same batch size far away. The specific loss for two subsequences (L_{ts}) is detailed below.

$$\ell(\theta_{i,s_1}, \theta_{i,s_2}) = \frac{exp(cos(\theta_{i,s_1}, \theta_{i,s_2})/\tau)}{\sum_{b=1}^{B} 1_{b\neq i}[\sum_{j=1}^{2} exp(cos(\theta_{i,s_1}, \theta_{b,s_j})/\tau)] + exp(cos(\theta_{i,s_1}, \theta_{i,s_2})/\tau)} \tag{5}$$

$$L_{ts} = -log(\ell(\theta_{i,s_1}, \theta_{i,s_2})) \tag{6}$$

where θ_{i,s_1} and θ_{i,s_2} are the representations of the two subsequences of the i-th contig. B represents the batch size. $cos(\cdot,\cdot)$ is the cosine similarity between two vectors, and τ is the temperature. We expanded the formula from two subsequences to multiple subsequences. The final dynamic contrastive loss (L_{dmv}) is below.

$$L_{dmv} = \frac{1}{BS(S-1)} \sum_{b=1}^{B} \sum_{s_1=1}^{S} \sum_{s_2=s_1+1}^{S} L_{ts} \tag{7}$$

where S represents the number of subsequences generated in each training step. During our training process, we set $S = 6$.

Self-dropout contrastive learning

The concept of self-dropout was straightforward: the embeddings of a contig with different dropout masks were treated as positive pairs, while the embeddings of other contigs within the same batch of data were treated as negative pairs. The Transformer model was fed the same contig twice, generating two embeddings with different dropout masks. During optimization, the representations of a contig with different dropout masks were pulled closer together, while the representations of other contigs within the batch were pushed far away. The objective function for self-dropout contrastive learning is defined as:

$$\ell(\theta_{i,z_1}, \theta_{i,z_2}) = \frac{exp(cos(\theta_{i,z_1}, \theta_{i,z_2})/\tau)}{\sum_{b=1}^{B} 1_{b\neq i}[\sum_{j=1}^{2} exp(cos(\theta_{i,z_1}, \theta_{b,z_j})/\tau)] + exp(cos(\theta_{i,z_1}, \theta_{i,z_2})/\tau)} \tag{8}$$

$$L_{sd} = -log(\ell(\theta_{i,z_1}, \theta_{i,z_2})) \tag{9}$$

where θ_{i,z_1} and θ_{i,z_2} are the representations of the two embeddings of the i-th contig with z_1 and z_2 dropout masks. The final training objection is: $L = \alpha L_{dmv} + \beta L_{sd}$. We set $\alpha = 1.8$ and $\beta = 1.0$ during our contrastive training.

4.6 Cluster contigs

After contrastive learning, CompleteBin generated embedding for each contig. CompleteBin applied different clustering algorithms to group the contigs based on their embeddings.

Initial clustering of contigs using the Leiden algorithm

We utilized the same Leiden algorithm as COMEBin but adjusted its hyperparameters to account for the sensitivity of this advanced community detection algorithm. The resolution parameters were set to 1, 2, 4, 6, 8, 10, and 12, while the edge ratios (proportions of edges retained for clustering) were set to 50%, 75%, and 100%. The maximum number of edges was set to 75 and 100. Other parameters were configured similarly to those used in COMEBin. Additionally, SCGs were incorporated using the Leiden algorithm, following the approach of COMEBin. The contigs containing the same SCGs were treated as fixed members, ensuring that contigs with the same SCG were grouped into separate clusters.

Refine the bins with high contamination

Following the initial clustering using the Leiden algorithm, we evaluated the quality of the clustering results by a method similar to that used in MetaBinner. Based on these evaluations, the clustering results were refined as follows: bins with contamination levels exceeding 10% were re-clustered using FLSpp to improve purity. For bins with moderate contamination (5% $\leq$ contamination $\leq$ 10%), shorter contigs containing redundant single-copy genes were removed. Bins with low contamination (contamination $\leq$ 5%) were left unchanged. This refinement process was repeated until the contamination level was reduced to less than 5% or a maximum of three iterations was reached.

Ensemble the clustering results and remove duplicated MAGs

We first selected the clustering results with various hyperparameter configurations for ensembling, followed by the application of Galah to identify and remove duplicate MAGs. We divided the clustering results into distinct groups based on three key hyperparameters: resolution, edge ratios, and the maximum number of edges. From each group, we selected the clustering result that achieved the highest number of bins with contamination below 5% and completeness above 90%. Similarly, we also selected another clustering result with contamination below 10% and completeness above 50%. Subsequently, Galah was utilized to compare MAGs, detect duplicates, and eliminate them from the selected clustering results to ensure the retention of unique and highly accurate MAGs.

4.7 Metrics for performance evaluation

We evaluated completeness and contamination scores using CheckM2 [36]. MAGs with completeness ≥50% and contamination ≤10% were classified as medium-quality MAGs, while those with completeness ≥90% and contamination ≤5% were classified as near-complete MAGs. For the MAGs generated from both simulated and real-world datasets, we evaluated the numbers of high- and medium-quality MAGs after binning.

4.8 Benchmark datasets

For simulated datasets, the CAMI I short reads were downloaded from the '1st CAMI Challenge Dataset 3 CAMI high' from https://data.cami-challenge.org/participate/. The CAMI II short reads were downloaded from https://cami-challenge.org/datasets/.

For real-world datasets, the NCBI SRA accessions of the 3 freshwater samples were SRR26420192 (Freshwater-1), ERR9631077 (Freshwater-2), and ERR4195020 (Freshwater-3); the plant sample was SRR10968246. The 3 ocean samples were from [37] and their IDs were 1021526 (Ocean-1), 1125694 (Ocean-2), and 1102218 (Ocean-3). The human fecal metagenomic sequencing reads of the IBS-D cohort were downloaded from China National GeneBank (CNGB) with accession number CNPO000334, and its ID is K0268.

4.9 Code availability

The source code is freely available at 'https://github.com/zoubohao/CompleteBin' under an MIT license.

References

[1] Berg, G. *et al.* Microbiome definition re-visited: old concepts and new challenges. *Microbiome* **8**, 1–22 (2020).

[2] Bharti, R. & Grimm, D. G. Current challenges and best-practice protocols for microbiome analysis. *Briefings in bioinformatics* **22**, 178–193 (2021).

[3] Quince, C., Walker, A. W., Simpson, J. T., Loman, N. J. & Segata, N. Shotgun metagenomics, from sampling to analysis. *Nature biotechnology* **35**, 833–844 (2017).

[4] Yang, C. *et al.* A review of computational tools for generating metagenome-assembled genomes from metagenomic sequencing data. *Computational and Structural Biotechnology Journal* **19**, 6301–6314 (2021).

[5] Ayling, M., Clark, M. D. & Leggett, R. M. New approaches for metagenome assembly with short reads. *Briefings in bioinformatics* **21**, 584–594 (2020).

[6] Wu, Y.-W., Tang, Y.-H., Tringe, S. G., Simmons, B. A. & Singer, S. W. Maxbin: an automated binning method to recover individual genomes from metagenomes using an expectation-maximization algorithm. *Microbiome* **2**, 1–18 (2014).

[7] Kang, D. D. *et al.* Metabat 2: an adaptive binning algorithm for robust and efficient genome reconstruction from metagenome assemblies. *PeerJ* **7**, e7359 (2019).

[8] Liu, C.-C. *et al.* Metadecoder: a novel method for clustering metagenomic contigs. *Microbiome* **10**, 46 (2022).

[9] Nissen, J. N. *et al.* Improved metagenome binning and assembly using deep variational autoencoders. *Nature biotechnology* **39**, 555–560 (2021).

[10] Pan, S., Zhao, X.-M. & Coelho, L. P. Semibin2: self-supervised contrastive learning leads to better mags for short-and long-read sequencing. *Bioinformatics* **39**, i21–i29 (2023).

[11] Wang, Z. *et al.* Effective binning of metagenomic contigs using contrastive multi-view representation learning. *Nature Communications* **15**, 585 (2024).

[12] Traag, V. A., Waltman, L. & Van Eck, N. J. From louvain to leiden: guaranteeing well-connected communities. *Scientific reports* **9**, 1–12 (2019).

[13] Lamurias, A., Sereika, M., Albertsen, M., Hose, K. & Nielsen, T. D. Metagenomic binning with assembly graph embeddings. *Bioinformatics* **38**, 4481–4487 (2022).

[14] Periwal, V. & Scaria, V. Insights into structural variations and genome rearrangements in prokaryotic genomes. *Bioinformatics* **31**, 1–9 (2015).

[15] Zou, B. *et al.* A multi-modal deep language model for contaminant removal from metagenome-assembled genomes. *Nature Machine Intelligence* 1–11 (2024).

[16] Nguyen, E. *et al.* Sequence modeling and design from molecular to genome scale with evo. *Science* **386**, eado9336 (2024).

[17] Zhou, Z., Luhmann, N., Alikhan, N.-F., Quince, C. & Achtman, M. Accurate reconstruction of microbial strains from metagenomic sequencing using representative reference genomes. In *Research in Computational Molecular Biology: 22nd Annual International Conference, RECOMB 2018, Paris, France, April 21-24, 2018, Proceedings 22*, 225–240 (Springer, 2018).

[18] Parks, D. H. *et al.* Recovery of nearly 8,000 metagenome-assembled genomes substantially expands the tree of life. *Nature microbiology* **2**, 1533–1542 (2017).

[19] Mukherjee, S. *et al.* 1,003 reference genomes of bacterial and archaeal isolates expand coverage of the tree of life. *Nature biotechnology* **35**, 676–683 (2017).

[20] Conrads, T., Drexler, L., Könen, J., Schmidt, D. R. & Schmidt, M. Local search k-means++ with foresight. *arXiv preprint arXiv:2406.02739* (2024).

[21] Alexey, D. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv: 2010.11929* (2020).

[22] J., W. B. Galah: More scalable dereplication for metagenome assembled genomes. *(Github)* (2024).

[23] Vaswani, A. Attention is all you need. *Advances in Neural Information Processing Systems* (2017).

[24] Gao, T., Yao, X. & Chen, D. Simcse: Simple contrastive learning of sentence embeddings (2022). URL `https://arxiv.org/abs/2104.08821. 2104.08821`.

[25] Wang, Z., Huang, P., You, R., Sun, F. & Zhu, S. Metabinner: a high-performance and stand-alone ensemble binning method to recover individual genomes from complex microbial communities. *Genome Biology* **24**, 1 (2023).

[26] Sczyrba, A. *et al.* Critical assessment of metagenome interpretation—a benchmark of metagenomics software. *Nature methods* **14**, 1063–1071 (2017).

[27] Meyer, F. *et al.* Critical assessment of metagenome interpretation: the second round of challenges. *Nature methods* **19**, 429–440 (2022).

[28] Bowers, R. M. *et al.* Minimum information about a single amplified genome (misag) and a metagenome-assembled genome (mimag) of bacteria and archaea. *Nature biotechnology* **35**, 725–731 (2017).

[29] Duncan, A. *et al.* Metagenome-assembled genomes of phytoplankton microbiomes from the arctic and atlantic oceans. *Microbiome* **10**, 67 (2022).

[30] Tláskal, V. *et al.* Metagenomes, metatranscriptomes and microbiomes of naturally decomposing deadwood. *Scientific data* **8**, 198 (2021).

[31] Buck, M. *et al.* Comprehensive dataset of shotgun metagenomes from oxygen stratified freshwater lakes and ponds. *Scientific Data* **8**, 131 (2021).

[32] Kavagutti, V. S. *et al.* High-resolution metagenomic reconstruction of the freshwater spring bloom. *Microbiome* **11**, 15 (2023).

[33] Maestre-Carballa, L., Navarro-López, V. & Martinez-Garcia, M. City-scale monitoring of antibiotic resistance genes by digital pcr and metagenomics. *Environmental Microbiome* **19**, 16 (2024).

[34] Zhao, L. *et al.* A clostridia-rich microbiota enhances bile acid excretion in diarrhea-predominant irritable bowel syndrome. *The Journal of clinical investigation* **130**, 438–450 (2024).

[35] Mende, D. R. *et al.* progenomes: a resource for consistent functional and taxonomic annotations of prokaryotic genomes. *Nucleic acids research* gkw989 (2016).

[36] Chklovski, A., Parks, D. H., Woodcroft, B. J. & Tyson, G. W. Checkm2: a rapid, scalable and accurate tool for assessing microbial genome quality using machine learning. *Nature Methods* **20**, 1203–1212 (2023).

[37] Duncan, A. *et al.* Metagenome-assembled genomes of phytoplankton microbiomes from the arctic and atlantic oceans. *Microbiome* **10**, 67 (2022).