Bioinformatics, 37(23), 2021, 4299–4306
doi: 10.1093/bioinformatics/btab439
Advance Access Publication Date: 22 June 2021
Original Paper

Genome analysis

Embeddings of genomic region sets capture rich biological associations in lower dimensions

Erfaneh Gharavi[1,2], Aaron Gu[1,3], Guangtao Zheng[3], Jason P. Smith[1,4], Hyun Jae Cho[1,3], Aidong Zhang[3], Donald E. Brown[2] and Nathan C. Sheffield [1,2,4,5,6,*]

[1]Center for Public Health Genomics, University of Virginia, Charlottesville, VA 22903, USA, [2]School of Data Science, University of Virginia, Charlottesville, VA 22903, USA, [3]Department of Computer Science, University of Virginia, Charlottesville, VA 22903, USA, [4]Department of Biochemistry and Molecular Genetics, University of Virginia, Charlottesville, VA 22903, USA, [5]Department of Public Health Sciences, University of Virginia, Charlottesville, VA 22903, USA and [6]Department of Biomedical Engineering, University of Virginia, Charlottesville, VA 22903, USA

*To whom correspondence should be addressed.

Associate Editor: Pier Luigi Martelli

Received on November 18, 2020; revised on June 7, 2021; editorial decision on June 9, 2021; accepted on June 15, 2021

Abstract

Motivation: Genomic region sets summarize functional genomics data and define locations of interest in the genome such as regulatory regions or transcription factor binding sites. The number of publicly available region sets has increased dramatically, leading to challenges in data analysis.

Results: We propose a new method to represent genomic region sets as vectors, or embeddings, using an adapted word2vec approach. We compared our approach to two simpler methods based on interval unions or term frequency-inverse document frequency and evaluated the methods in three ways: First, by classifying the cell line, antibody or tissue type of the region set; second, by assessing whether similarity among embeddings can reflect simulated random perturbations of genomic regions; and third, by testing robustness of the proposed representations to different signal thresholds for calling peaks. Our word2vec-based region set embeddings reduce dimensionality from more than a hundred thousand to 100 without significant loss in classification performance. The vector representation could identify cell line, antibody and tissue type with over 90% accuracy. We also found that the vectors could quantitatively summarize simulated random perturbations to region sets and are more robust to subsampling the data derived from different peak calling thresholds. Our evaluations demonstrate that the vectors retain useful biological information in relatively lower-dimensional spaces. We propose that vector representation of region sets is a promising approach for efficient analysis of genomic region data.

Availability and implementation: https://github.com/databio/regionset-embedding.

Contact: nsheffield@virginia.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

An epigenomics experiment is often represented as a region set, which is a collection of genomic intervals that identify the locations of interest produced by a biological experiment. Region sets are produced from various experiments, such as ChIP-Seq (Furey, 2012) or ATAC-Seq (Buenrostro *et al.*, 2013; Smith and Sheffield, 2020), and contain the locations of functional elements along the genome such as enhancers, promoters and transcription factor binding sites (Dunham *et al.*, 2012). Region sets are frequently stored as Browser Extensible Data (BED) files, which may contain up to millions of individual regions. As the amount of publicly available epigenome data has increased, the volume of data has led to challenges analyzing it.

In the past few years, many methods have been developed for processing, analyzing, and comparing genomic region sets. One key task has been finding connections between region sets, but like many other tasks, this is complicated by the volume and complexity of region set data (Dozmorov, 2017; Jalili *et al.*, 2019; Kanduri *et al.*, 2019; Layer *et al.*, 2018; Sheffield and Bock, 2016). Here, we address this problem by embedding region sets in a lower dimensional space that retains important biological information. Robust embeddings have potential to highlight important biological relationships among region sets and can lead to new ways to query and analyze data repositories. Importantly, robust embeddings can capture biological information that would be difficult to extract from raw data, and therefore the performance of downstream tasks, such as classification and clustering, can be improved (Woloszynek *et al.*, 2019).

One simple example of data representation is the bag-of-words method for textual data, which uses the existence of words to create a binary vector representation. More recently, new methods have invoked the distributional hypothesis, which states that words in similar context have similar meanings (Bengio *et al.*, 2013; Mikolov *et al.*, 2013; Pennington *et al.*, 2014). In this framework, embeddings learned from context have led to improved performance in a wide range of natural language processing (NLP) tasks (Young *et al.*, 2018), and distributed representations of data are now commonly used across many disciplines to reduce data dimensionality and learn relationships. In bioinformatics, embeddings have been used to represent DNA nucleotide sequences (Blackshields *et al.*, 2010; Dai *et al.*, 2017; Manavalan *et al.*, 2019; Wei *et al.*, 2012), protein sequences (Yang *et al.*, 2018), genes (Du *et al.*, 2019) and single-cell Hi-C data (Liu *et al.*, 2018), among others. Factorized tensor decomposition has also been used to achieve biologically meaningful representation for RNA-Seq data (Trofimov *et al.*, 2020). Most similar to our use case is the recently published Avocado method, which learns a dense representation for a region using a deep neural network trained on epigenome signal information (Schreiber *et al.*, 2020). Avocado uses metadata annotations during the training to develop the embeddings, which are then evaluated on other downstream tasks. Our approach differs in several ways: first, it is more general in that it trains on intervals only and does not require signal data, which makes it amenable to a larger class of data that may or may not have accompanying signal annotation (e.g. CpG island annotation, HMM chromatin states, k-mer locations, motif matches, etc.); second, training is based solely on co-occurrence of intervals without requiring metadata annotations; third, the underlying model uses the shallow word2vec neural network; and fourth, our primary goal is to create embeddings for *sets* of regions, whereas Avocado is geared toward representations of individual regions. Therefore, while there is clear value in methods that incorporate signal value and annotation information, there is also utility in the more general approach we present here.

We translate NLP techniques to genomic region set data by considering each region set as a text document, with each region as a word inside that document. We applied two NLP approaches in our work: a method based on term frequency-inverse document frequency (Salton and Buckley, 1988) for feature selection, and word2vec embeddings (introduced in Supplementary Materials) (Mikolov *et al.*, 2013). The proposed condensed representation for a region set is learned using only genomic coordinates. For comparison, we also tested a non-condensed representation based on a simple binary vector approach. We evaluate with three evaluation tasks: first, a classification task to predict and visualize biological characteristics of a region set, such as cell line, antibody and tissue type; second, a similarity detection task to assess if the models capture differences between a reference file and simulated perturbed ones; and third, a subsampling task to test robustness to data mimicking varying thresholds for calling peaks. Using these evaluations, we show that the proposed embeddings maintain high classification performance, quantitative reflection of simulated perturbation, and robustness to subsampling by different peak calling thresholds, despite multi-fold reduction in dimensionality.

2 Materials and methods

2.1 Overview of the approach

We used a dataset from the ChIP-Atlas database, which contains uniformly processed ChIP-seq data from the Sequence Read Archive (Fig. 1a) (Oki *et al.*, 2018). We applied three different approaches to represent these region sets: union representation, tf_idf-based representation, and region-set2vec embedding (Fig. 1b). To evaluate the embeddings, we employed three machine learning tasks: classification, similarity detection, and peak threshold robustness (Fig. 1c).

2.2 Dataset

Chromatin immunoprecipitation followed by sequencing (ChIP-seq) identifies the binding sites of DNA-associated proteins. We downloaded 12 731 BED files representing ChIP-Seq data from ChIP-Atlas and constructed 3 different test datasets: one annotated with antibody, one with cell line, and one with tissue type.

The antibodies are *h3k27ac*, *h3k27me3*, *h3k4me1*, *h3k4me2*, *h3k4me3*, *h3k36me3,* and *h3k9me3*. The cell lines are *MCF-7*, *HeLa*, *HEK293*, *A549*, *Hep G2*, *HCT116*, *LoVo*, *GM12878*, *LNCap,* and *K562*. The tissue types are *liver*, *peripheral blood*, *primary prostate cancer*, *blood*, *breast*, *bone marrow,* and *kidney*. For each of these three datasets, we divided the data into a training (80%) and test (20%) set (Table 1).

2.3 Representation methods

For each of our three test datasets, we represented the data in three different ways:

2.3.1 Union representation

The union representation represents a region set as a binary vector, with each position in the vector indicating whether a particular region is present in that set (Fig. 2a). The first step is to create a consensus set of regions across all region sets in a dataset, which we refer to as a *universe* of possible regions. We created a universe by first concatenating regions in 100 random training files from each dataset into one file, then merging any regions that were closer than 1000 base pairs into a larger region using the start position of the first region and the end position of the second region (Supplementary Table S1). To confirm that the random selection of 100 files did not have an impact on the universe, we created multiple union representations with different universes, which resulted in similar performance (Supplementary Table S2). The resulting n regions in the universe correspond to the vector of length n, with each region specifying a position, or feature, in the vector. The binary vector representation reflects, for a particular set, which of these universe regions is present in the set, which we evaluate with a simple interval overlap calculation.

2.3.2 Tf_idf-based representation

One problem with the union representation is that merging close regions creates larger regions that can obscure real biological differences in regions. To mitigate this, our second approach employs a feature selection and representation method from NLP to retain the regions that play an important role in distinction of the data (see Supplementary Material). Briefly, we consider each base pair location as a term and a region set as a document, and calculated the tf_idf score for each base pair across regions sets (Fig. 2b). Base pairs with higher scores are more informative for identifying relationships among region sets, as this approach will weight elements present in an intermediate number of region sets, while down-weighting elements that are either very common or very rare across region sets.

We selected the top 100k, 500k and 1000k scoring base pairs (kb) for each chromosome to use in our experiment, producing three tf_idf-based representations. This was also to show the effect of the number of selected base pairs on classification accuracy. After selecting the important base pairs, we merge any adjacent base pairs to build a region and remove regions with fewer than 100 base pairs. Like the union representation approach, these regions can then be features of a binary vector. The key difference between this approach and the union representation approach is that here we are creating vector features from merged base pairs that are considered as informative.

2.3.3 Region-set2vec representation

So far, these two approaches have represented the data as high dimensional vectors. These approaches consider each region as a separate feature and do not take into account relationships among the regions. Furthermore, high dimensionality increases time and space complexity for processing and analyzing data. To address these challenges, we adapted

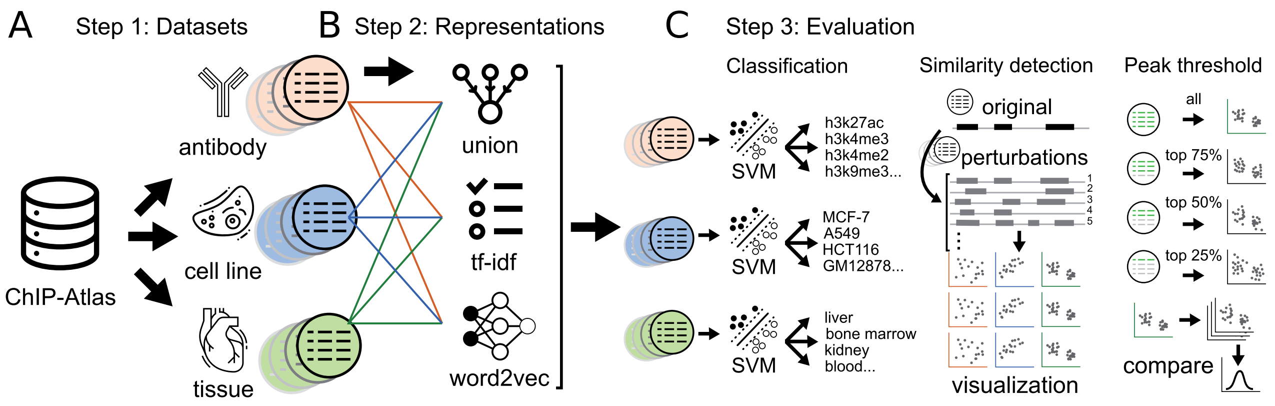

Fig. 1. Overview of methods. (**a**) Datasets were divided into three different experiment types from the ChIP-Atlas dataset. (**b**) Each file was converted into three vector representations. (**c**) We evaluated the representations with three evaluation tasks. First, classification of the region set representations into tissue type, cell line, and antibody. Second, evaluating sensitivity to known changes in the data using a simulated dataset. Third, testing the robustness of the embeddings to different subsets of the original region list

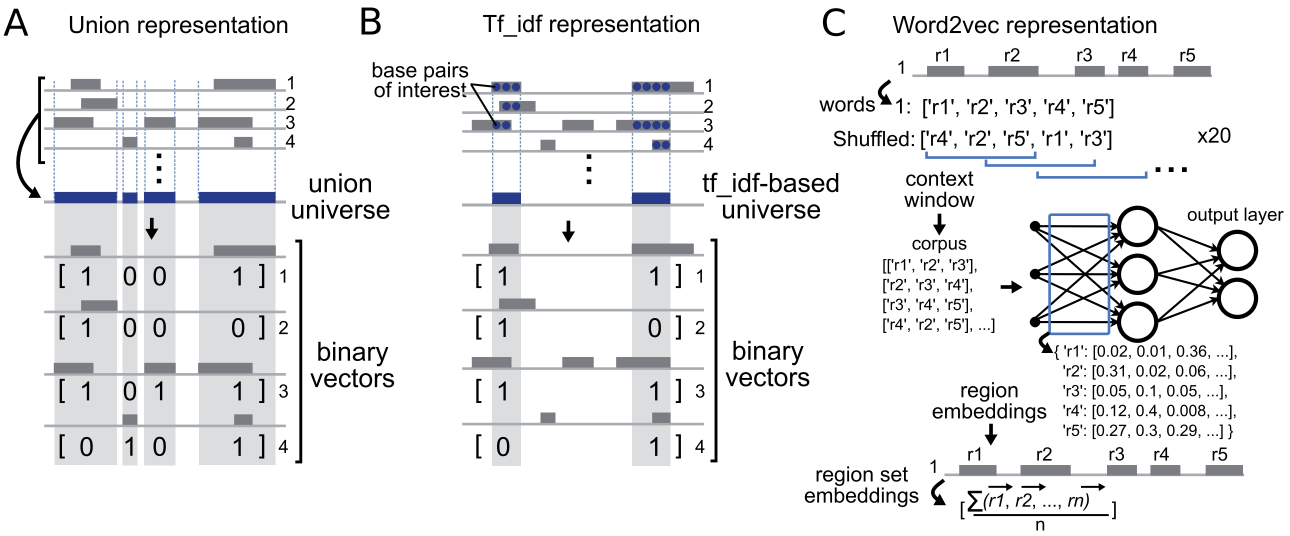

Fig. 2. Details of embedding methods. (**a**) The union representation simply merges the region sets to create a universe, then casts each region set as a binary vector reflecting presence or absence of the merged universe region. (**b**) The tf-idf approach builds a restricted universe based on nucleotides determined to be highly informative. (**c**) The word2-vec approach extracts embeddings as weights from a shallow neural network trained using co-occurrence frequencies

Table 1. Dataset statistics

	Classes	Training samples	Test samples
Antibody	7	2777	695
Cell line	10	5527	1368
Tissue	7	1788	440

the word2vec algorithm to train a distributed representation for each region in our training region sets (Fig. 2c). Similar to the pre-processing step of vocabulary building in NLP, we first map the regions from each file into a standard universe. We chose the union representation to be the universe because it did not use any feature selection, and is thus a fair comparison between the region-set2vec embedding and the tf_idf representation. During the training phase of word2vec, the task is to predict co-occurring regions given an input region in the same region set. After training, the trained weights in the neural network consist of 100-dimensional vector representations for each region, which we take as region embeddings.

Word2vec considers consecutive *words* in a context window (set to 100 words in our experiment) to train the embeddings. We want to capture the co-occurrence information for all the regions in a BED file. Ideally, the context window should have a length that could cover all the regions in a BED file. However, since a BED file could have thousands of regions or more, a very large context window would require very large memory and make it very hard to optimize the region embeddings. To circumvent these, we use a small constant window size at the cost of shuffling the regions multiple times. Since the order of the regions is not important, we can shuffle the regions in each region set before sampling the context window and passing them as the input of word2vec algorithm. With shuffling, all the regions that co-occur in a region set can appear in the same context window by random chance. With a smaller window size, we need less memory but a larger number of shuffling operations. Too small context window will limit the learning capability of region-set2vec, as very few regions are used at a time. In practice, we found that setting the context window size to 100 and shuffling each region set 20 times is a good choice for the current study.

After obtaining the vectors for each region, we used them to calculate a vector representing the region set by averaging all the vectors of the regions contained in the region set $\mathcal{R}$, shown in Equation 1.

$$RegionSet\_Embedding = \frac{\sum_{j=1}^{|\mathcal{R}|} \boldsymbol{r}_j}{|\mathcal{R}|} \qquad (1)$$

Here, $\mathcal{R}$ is the region set and $\boldsymbol{r}_j$ is the region embedding of the jth region in $\mathcal{R}$.

To get a document embedding from word embeddings, a commonly used method is doc2vec (Le and Mikolov, 2014), which extends word2vec by introducing an additional document embedding vector in the word2vec architecture such that it can be trained jointly with word embeddings. In the context of region embedding, since the order of regions in each region set is not important and we shuffle during training, the learned document embedding cannot catch meaningful information throughout the training. In fact, when we trained the doc2vec model with the same amount of data that was used for word2vec training, the performance was much lower than averaging the region embeddings (Supplementary Table S3). Therefore, we find that averaging over all the region embeddings obtained from a BED file is a simple and effective document embedding method.

2.4 Evaluation

Having created different vectors to represent region sets, we next sought to evaluate how well each proposed representation retains biological information. We designed three evaluation tasks: a classification task, a similarity detection task, and a peak threshold robustness task. The classification task asks how well the region set embeddings can reflect known biological relationships among region sets. The similarity detection task uses simulated random perturbations to assess how well the embeddings reflect known levels of mathematical difference among region sets. Finally, the peak threshold robustness task tests how much the embeddings change when the input data is a truncated subset of the regions, such as would happen if a different peak calling threshold were used.

2.4.1 Classification task

The classification task is as follows: given the region set vector representations as input, we trained classifiers to classify region sets either by tissue type, antibody, or cell line. Recall that this annotation information is not used to construct the original embeddings, which is unsupervised and relies on the regions themselves; we incorporate the annotation information in this classification task to evaluate whether the embeddings can capture the annotation information *de novo*. We employed a support vector machine (SVM) (Vapnik *et al.*, 1997) classifier for each of these tasks. An SVM is a supervised machine learning algorithm that creates a hyperplane that separates the data into sets. To classify multi-class data, we used one-versus-rest to split the data to a binary dataset for each class.

Evaluating classifier performance. To evaluate the performance of the classification algorithms, we used the micro-averaging of the F1 score (Yang and Liu, 1999) to account for the class imbalance. This F1 score is composed of micro-precision and micro-recall, which are defined as follows:

$$micro\_precision = \frac{tp}{tp + fp}, \qquad (2)$$

$$micro\_recall = \frac{tp}{tp + fn}, \qquad (3)$$

$$F1_{score} = \frac{2 * micro\_precision * micro\_recall}{micro\_precision + micro\_recall}, \qquad (4)$$

where the true positive $tp = \sum_{i=1}^{M} tp_i$ is the summation of all true positive numbers in each class, and fp, and fn are similarly defined as the summation of all false positive and all false negative numbers, respectively.

Visualising embeddings with UMAP. To further evaluate the representation approaches, we visualized the embeddings in 2 dimensions using uniform manifold approximation and projection (UMAP) (McInnes *et al.*, 2018). UMAP is a non-linear dimensionality reduction method designed to analyze high dimensional data. It has been proven as an effective tool to reveal meaningful structure in biological data (Becht *et al.*, 2019; Eng *et al.*, 2019).

2.4.2 Similarity detection task

Querying and retrieving similar region sets from huge datasets relies on detecting the similarity among region sets. As mentioned, representing data in lower dimensions facilitates this process. We employ the similarity detection task to further evaluate the enrichment of our representation approaches.

Creating a simulated dataset. We created a simulated dataset using bedshift, a tool that randomly perturbs BED files by adding, dropping or shifting regions to a parameterizable degree (Gu *et al.*, 2021). These perturbations create new files with a defined similarity that can be used as a benchmark for similarity scoring. For example, the similarity between the perturbed file and the original file should be greater if 10% of regions are dropped than if 20% of regions are dropped. The simulated dataset used all of the three perturbations, add, drop and shift, at percentage rates from 10% to 90% in increments of 10%. For each level of perturbation, we created 100 replicates, resulting in 900 total files. We then converted the files in all datasets to the numerical representation using three methods and visualized each representation on the perturbed datasets.

Visualizing perturbed embeddings with UMAP. We used UMAP to plot the simulated datasets to depict the effectiveness of each representation at detecting similarities. We also did a sensitivity analysis to test the effect of three major UMAP hyperparameters: the distance metric; the number of neighbors to consider; and the minimum distance allowed for points to be in the low dimensional representation.

2.4.3 Peak thresholds subsampling task

For our final evaluation task, we asked how much the embeddings would change if trained on only a subset of the data. Genomic regions are often generated by calling peaks from the ChIP-seq signals based on a threshold. We investigate the robustness of different representations to the threshold of peak calling from signals using the classification task. Using the cell line dataset, we binned the regions based on signal values into four quartiles. Four datasets from the cell line annotated dataset were generated by this approach. The first is the original data that contains all of the regions. We refer to this dataset as *All*. Three other datasets were generated by selecting the top 75%, top 50%, and top 25% of the regions in each BED file based on the signal values. We tested the resulting embeddings using both our classification approach, and our similarity detection approach. We calculated the cosine similarity between the representations of the original dataset, *All*, versus three other datasets and plotted the distribution of cosine similarities.

3 Results

3.1 Classification task

Our goal is to compare the three representation approaches. The number of features in each approach, which corresponds to the dimensionality of the representations, ranges from one hundred to over one hundred thousand (Table 2).

3.1.1 Classification performance

We evaluate the representations by the performance of the classifier trained to identify biological information for each region set. Therefore, for each representation method, we trained an SVM classifier on cell line, antibody, and tissue training sets and report the results on the test sets (Table 3; See Supplementary Material).

The classifier using the union representation performed well because all of the features in the universe were retained. On a similar

Table 2. Number of features for each representation

	Number of features		
Representation method	Antibody	Cell line	Tissue
Union representation	136 284	180 424	150 681
tf_idf—1000-kb	50 100	40 596	43 912
tf_idf—500-kb	28 527	22 783	25 238
tf_idf—100-kb	8262	7225	7791
region-set2vec embedding	100	100	100

Table 3. SVM classifier performance

	F1 score_SVM		
Representation method	Antibody	Cell line	Tissue
Union representation	0.9424	**0.9898**	**0.9591**
tf_idf—1000-kb	**0.9468**	0.9788	0.9523
tf_idf—500-kb	0.9439	0.9613	0.9500
tf_idf—100-kb	0.9022	0.8977	0.9568
region-set2vec embedding	0.9381	0.9605	0.9091

note, the representation using tf_idf with 1000-kb also performed well on the antibody classification task, due to the high number of base pairs selected. The region-set2vec embedding performed better than the tf_idf representation with 100-kb on the antibody and cell line classification, but performed the worst on the tissue classification. It seems that the tissue classification worked well even with 100-kb selected in the tf_idf, possibly indicating that the feature selection method found few significant regions signaling the tissue type. The region-set2vec embedding with 100 dimensions performed as well as the other high-dimensional representations on antibody and cell line datasets. In addition, it produces low-dimensional representations that cause the downstream run-time for classification to be much faster (Table 4). On average, training and testing the classifiers on region set embeddings are 2500 times faster than the union representation. We also compared the run-time of building the representation model and the required time to transform a new test dataset to the numerical representations. Despite higher run-time in training the region2vec model (Supplementary Table S4), transforming the test dataset occurred faster in region-set2vec representation methods (Supplementary Table S5). For union and tf_idf representations, a region-set needs to be mapped to the high dimensional space and then uses the saved dimension reduction model to reduce the dimension to 100 while region-set embedding is calculated in low dimension (see Supplementary Material).

Given the vast difference in number of features, we next sought to conduct a fairer performance comparison by restricting the dimensions of all methods to 100. Therefore, we selected the top 100 components using Principal Component Analysis (PCA) (Abdi and Williams, 2010). In addition to the linear PCA, we also tried two non-linear kernel PCA appraoches: *rbf* and *polynomial*. We selected the best PCA-kernel using cross-validation score in the same way as for the SVM kernel. The best results on each dataset were achieved by the *linear* kernel for PCA among three different kernels (Supplementary Tables S6–S8). Using these 100-dimensional features as inputs to the SVM classifier with linear kernel, the region-set2vec embedding performs the best for all three classification tasks (Table 5). The other methods, now with the same dimensions as the region-set2vec embedding, have reduced performance, showing that the region-set2vec embedding retained the most information in 100 dimensions.

Table 4. SVM training run-time (seconds)

Representation method	Antibody	Cell line	Tissue
Union representation	402.60	2480.97	348.38
tf_idf—1000-kb	129.12	540.65	101.28
tf_idf—500-kb	71.90	315.59	58.76
tf_idf—100-kb	21.52	92.99	17.64
region-set2vec embedding	0.21	0.78	0.17

Table 5. SVM-PCA classifier performance

	F1 score_SVM		
Representation method	Antibody	Cell line	Tissue
Union representation	0.9281	0.9539	0.8932
tf_idf—1000-kb	0.9223	0.9327	0.8932
tf_idf—500-kb	0.9108	0.9145	0.8727
tf_idf—100-kb	0.8993	0.8662	0.8886
region-set2vec embedding	**0.9381**	**0.9605**	**0.9091**

To evaluate the performance, we used 10-fold cross-validated paired t-tests. We first split the training data into 10 folds of equal size. In each cross-validation iteration, we compute the difference in performance between classifying the data represented by the union method and the same data represented by the regionset2vec approach. The test shows that the higher performance of region-set2vec is statistically significant for all datasets (P-values < 0.003 for antibody; 0.000036 for cell line; and 0.041 for tissue dataset).

We used the area under the Precision-Recall curve (AUPR), to visualize the performance classification of antibody, cell line, and tissue on different representations (Fig. 3a). Higher values indicate the trained classifier can better distinguish between classes. With region-set2vec embeddings, the trained classifiers achieve the highest area under the curve scores in all three datasets.

3.1.2 UMAP visualization

For each of our three datasets, we used UMAP to visualize the representation space. We selected 100 as the number of neighbors and Euclidean distance as the similarity metric. Since the three tf_idf representations had similar results, we only plot the 500-kb tf_idf representation. Union representation resulted in mapping most of the samples in the antibody dataset into a similar space on the UMAP plot (Fig. 3b). Using tf_idf representation, the samples are more distinctive due to the selection of discriminant features (Fig. 3c). The region-set2vec embeddings show more distinctive, tight clusters (Fig. 3d), and clearly distinguish antibodies associated with repression, activation, and transcription. Similarly, cell lines and tissue types are clustered effectively by the region-set2vec representation (Fig. 3e and f).

To represent each type of antibody with a numerical vector and as a single point in 2-d space, we merged all representations of the samples in the dataset using averaging as the combination function. The classes of antibody associated with active gene expression (*h3k4me1*, *h3k4me2*, *h3k4me3* and *h3k27ac*) are mapped closer to each other and further away from repressive marks (*h3k27me3* and *h3k9me3*), and transcription-associated (*h3k36me3*) antibody classes (Fig. 3g). This plot indicates that our learned representations retain biological information like antibody type without using this information in the training phase.

3.2 Similarity detection task

As a second independent evaluation of our region embeddings, we employed a similarity detection task. In this task, we simulated perturbations to a given BED file at a range of pre-specified rates, and then examined how the differences in embedding reflect the known perturbation rates.

3.2.1 UMAP plots

To visualize how the embeddings reflect the perturbation, we used UMAP plot with 100 neighbors, Cosine as the distance metric. The union and tf_idf representations are randomly distributed in the UMAP 2-dimensional space, despite the perturbation rate ranging

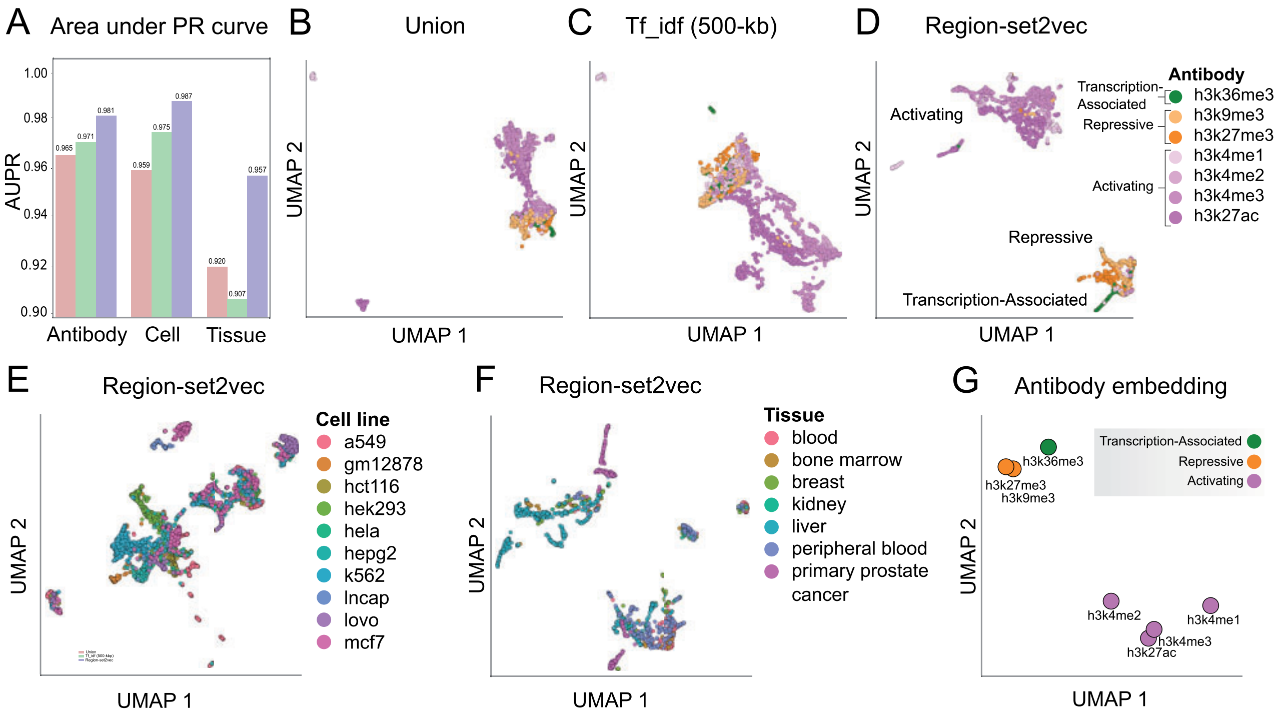

Fig. 3. Classification performance and UMAP visualizations of representations. (**a**) Area under the micro-average PR curve for each dataset. (**b**) Antibody dataset represented by union method. (**c**) Tf_idf representation using 500-kb of antibody dataset. Each point represents a region set. (**d**) Region-set2vec embedding of antibody dataset using combined region embeddings trained by word2vec algorithm. (**e**) Region-set2vec representation of cell line dataset. (**f**) Region-set2vec representation of tissue type datasets. Panels show the UMAP visualization of the original data, not the PCA-reduced data. (**g**) A UMAP projection of antibody embeddings annotated by class label. Each point is the combination of all the samples in each class

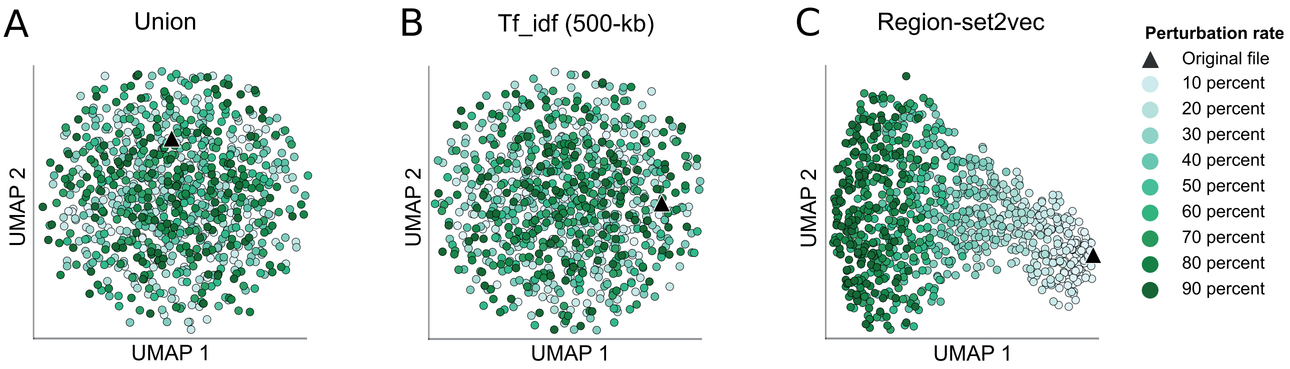

Fig. 4. UMAP visualization of the simulated dataset with different rates of perturbation using 3 representation methods. (**a**) Union representation (**b**) Tf_idf representation with 500k important base pairs (**c**) Region-set2vec representation

in increments of 10% (Fig. 4a and b). In contrast, the UMAP visualization of the region-set2vec embeddings showed a gradual deviation as the percentage of the perturbation increased (Fig. 4c). This result indicates that, at least for the purpose of UMAP visualization, the region-set2vec embeddings are able to reflect quantitative similarity among region sets, whereas the other methods are not. We also tested other UMAP parameters (Supplementary Table S9) with a sensitivity analysis and found that using 100 neighbors, 0.01 as the minimum distance and Cosine as the distance metric (Supplementary Fig. S1a–d) produced sensible and robust visualizations (see Supplementary Material). We also found that these results are consistent when perturbing files with *add*, *drop* and *shift* independently (Supplementary Fig. S2).

3.3 Peak thresholds subsampling task

To investigate the robustness of different representations, we ran the classification task on the original cell line dataset and three truncated datasets generated by selecting the top 75%, 50%, and 25% of the regions in each BED file based on the signal values. As expected, the performance of the classification task decreased as the files were more truncated; however, the region-set embedding representation method was least sensitive to the truncation (Fig. 5a). The region-set embedding representation performance is still >94% even when considering only the top 25% of peaks, compared to <84% for the Tf_idf approach. This indicates that this approach preserves the information of the missing regions by learning the embeddings based on the context.

We also evaluated the robustness of the representation by calculating the Cosine similarity between the representations of the original dataset and each truncated dataset. The distributions of Cosine similarities show the region-set2vec vector based on abbreviated data are most similar to the vectors based on the full data for all levels of truncation (Fig. 5b–d). This result again confirms that the region-set2vec approach retains the most information in the face of data subsets.

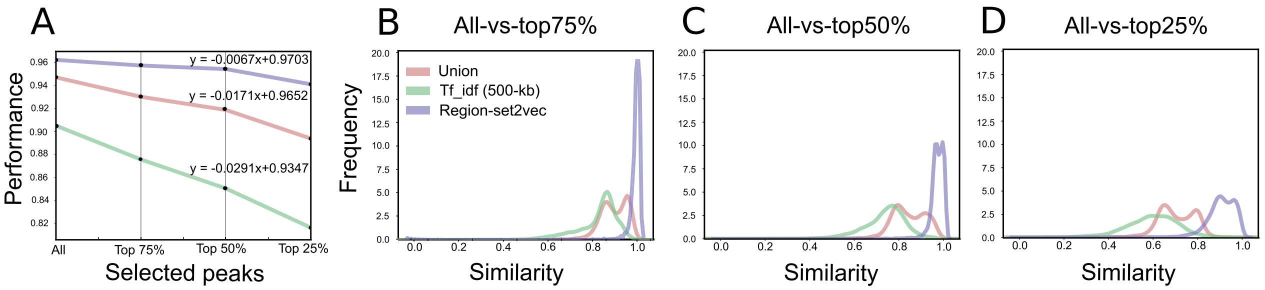

Fig. 5. Region set embeddings are more robust to peak calling threshold changes. (**a**) Sensitivity of different representation to selected peaks based on signal values. (**b-d**) Distribution of similarity between the original region sets and (**b**) top 75% of the selected peaks, (**c**) top 50% of the selected peaks and (**d**) top 25% of the selected peaks

3.4 Application of region-set2vec to single-cell data

Satisfied that region-set2vec could capture important relationships, we next sought to apply this approach as a proof-of-concept to discriminate cell types within single-cell ATAC-seq (scATAC) data. We used a simulated scATAC bone marrow dataset composed of six known tissues derived from an original set of bulk FACS-sorted data (Chen *et al.*, 2019). We processed this feature matrix and shuffled 25 times and evaluated with the following hyperparameters: 100 dimensions, a minimum count of 2, 12 neighbors and a window size of 250. Our adapted word2vec approach successfully separated each cell type into distinct clusters, similar to the previous best-performing candidates (Chen *et al.*, 2019) (Fig. 6a). We also applied the region-set2vec method on a single-cell ATAC-seq of LMPPs, monocytes, LSCs and leukemic blast cells (Corces *et al.*, 2016). The data is shuffled 50 times and 50-dimensional vectors are trained using region-set2vec method. The dissimilarity among monocytes and LMPPs (Xiong *et al.*, 2019) is clear here as the cells are mapped far from each other. LSCs are closer to LMPP cells and one class of the blast cell mapped adjacent to the monocytes (Fig. 6b). Together, these results indicate that region-set2vec is a promising approach for single-cell ATAC-seq data.

4 Discussion and conclusion

In this article, we proposed three feature selection methods to represent genomic regions sets: union representation, tf_idf representation, region-set2vec embedding, which we evaluated via classification, similarity detection and peak threshold robustness tasks. In the classification task, the region-set2vec method underperformed against other higher-dimensional vectors; however, after reducing the dimensionality of all methods to the same dimensions, the region-set2vec method outperformed all others significantly. Region-set2vec vectors also resulted in better visual class separation, and further showed distinction between, activating, repressive and transcription marks. This distinction was previously noted with the Avocado approach (Schreiber *et al.*, 2020), but Avocado considered the biological annotation in the training phase of the antibody embeddings; in contrast, our embeddings are learned solely from co-occurrences of regions within unannotated region sets.

In the similarity detection task, we showed that known interval-based similarity is better projected using region-set2vec representations. One possible explanation for this result is that the embedding vector for each region is trained in a way that preserves the information of the context (i.e. BED file). Because the vector embedding considers the co-occurrence of regions in the same file as a context, when some of the regions are dropped or shifted in a perturbed file, the information about these regions is conserved in the embedding of the remaining regions and consequently in the final representation. In contrast, the other methods consider regions as independent features, and the representation does not reflect the relationship among features. As a result, dropping, adding, or shifting the regions could cause abrupt transformation between the original file and the perturbed files. Embeddings that retain information across regions, therefore, appear to be less susceptible to such random perturbations. Along similar lines, our peak threshold robustness test showed that the region-set2vec vectors retained the most similarity to the original vectors, achieving high similar scores even when only 25% of the data is considered.

Overall, we demonstrated the feasibility of using NLP techniques to represent genomic region set data in new ways that will drive

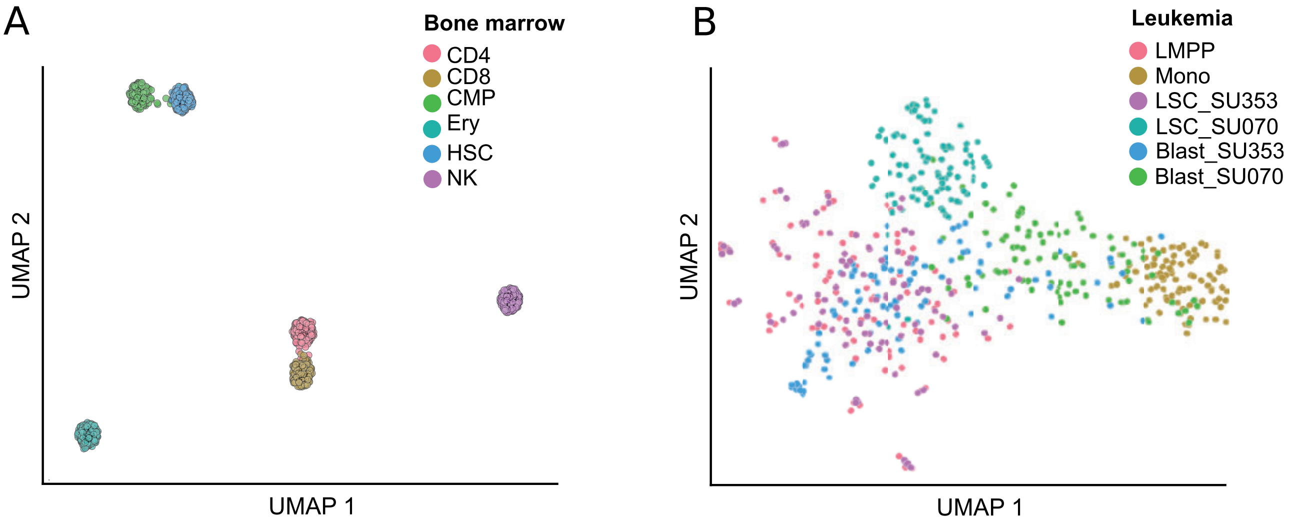

Fig. 6. UMAP visualization of region-set2vec representation on single cell datasets. (**a**) Simulated bone marrow dataset with a coverage of 2500 fragments per cell (Chen *et al.*, 2019). (**b**) Single-cell ATAC-seq of LMPPs, monocytes, LSCs, and leukemic blast cells dataset (Corces *et al.*, 2016). Individual cells are colored indicating the cell type label

analysis methods in the future. In the future, it may be possible to improve the quality of the region-set2vec embedding by optimizing the hyper-parameters. For example, here we arbitrarily chose 100-dimensional vectors, but it may be that a higher (or lower) number is optimal. In addition, we have several ideas for addressing the problem for building the universe of regions. We are now working to address this upstream problem and hope to have a general solution in the future. Furthermore, we have used a relatively limited collection of BED files, which can be extended with additional data sources. Applying the method on other source of data and mutated regions are the potential future directions. Altogether, our results indicate that low-dimensional representations of region sets built using nothing more than unsupervised collections of region set data can be an effective approach to build biologically meaningful vector representations.

Acknowledgements

The authors thank Tessa Danehy for her valuable comments.

Funding

This work was supported by the University of Virginia 3 Cavaliers funding initiative, by the National Science Foundation [IIS-1955151 and OAC-1934600] and by the National Institutes of Health [GM128636 and UL1TR003015].

Conflict of Interest: none declared.

References

Abdi,H. and Williams,L.J. (2010) Principal component analysis. *Wiley Interdiscip. Rev. Comput. Stat.*, **2**, 433–459.

Becht,E. *et al.* (2019) Dimensionality reduction for visualizing single-cell data using UMAP. *Nat. Biotechnol.*, **37**, 38–44.

Bengio,Y. *et al.* (2013) Representation learning: a review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.*, **35**, 1798–1828.

Blackshields,G. *et al.* (2010) Sequence embedding for fast construction of guide trees for multiple sequence alignment. *Algorithms Mol. Biol.*, **5**, 21.

Buenrostro,J.D. *et al.* (2013) Transposition of native chromatin for fast and sensitive epigenomic profiling of open chromatin, DNA-binding proteins and nucleosome position. *Nat. Methods*, **10**, 1213–1218.

Chen,H. *et al.* (2019) Assessment of computational methods for the analysis of single-cell ATAC-seq data. *Genome Biol.*, **20**, 241.

Corces,M.R. *et al.* (2016) Lineage-specific and single-cell chromatin accessibility charts human hematopoiesis and leukemia evolution. *Nat. Genet.*, **48**, 1193–1203.

Dai,H. *et al.* (2017) Sequence2vec: a novel embedding approach for modeling transcription factor binding affinity landscape. *Bioinformatics*, **33**, 3575–3583.

Dozmorov,M.G. (2017) Epigenomic annotation-based interpretation of genomic data: from enrichment analysis to machine learning. *Bioinformatics*, **33**, 3323–3330.

Du,J. *et al.* (2019) Gene2vec: distributed representation of genes based on co-expression. *BMC Genomics*, **20**, 82.

Dunham,I. *et al.* (2012) An integrated encyclopedia of DNA elements in the human genome. *Nature*, **489**, 57–74.

Eng,C.-H.L. *et al.* (2019) Transcriptome-scale super-resolved imaging in tissues by RNA seqFISH. *Nature*, **568**, 235–239. +.

Furey,T.S. (2012) ChIP-seq and beyond: new and improved methodologies to detect and characterize protein–DNA interactions. *Nat. Rev. Genet.*, **13**, 840–852.

Gu,A. *et al.* (2021) Bedshift: Perturbation of genomic interval sets. *bioRxiv*, doi:10.1101/2020.11.11.378554.

Jalili,V. *et al.* (2019) Next generation indexing for genomic intervals. *IEEE Trans. Knowledge Data Eng.*, **31**, 2008–2021.

Kanduri,C. *et al.* (2019) Colocalization analyses of genomic elements: approaches, recommendations and challenges. *Bioinformatics*, **35**, 1615–1624.

Layer,R.M. *et al.* (2018) GIGGLE: a search engine for large-scale integrated genome analysis. *Nat. Methods*, **15**, 123–126.

Le,Q. and Mikolov,T. (2014) Distributed representations of sentences and documents. In: *International Conference on Machine Learning,* pp. 1188–1196. PMLR, Beijing, China.

Liu,J. *et al.* (2018) Unsupervised embedding of single-cell Hi-C data. *Bioinformatics*, **34**, i96–i104.

Manavalan,B. *et al.* (2019) Meta-4mCpred: a sequence-based meta-predictor for accurate DNA 4mC site prediction using effective feature representation. *Mol. Ther. Nucleic Acids*, **16**, 733–744.

McInnes,L. *et al.* (2018) UMAP: uniform manifold approximation and projection. *J. Open Source Softw.*, **3**, 861.

Mikolov,T. *et al.* (2013) Distributed representations of words and phrases and their compositionality. In: *Advances in Neural Information Processing Systems*, pp. 3111–3119. Lake Tahoe, Nevada, USA.

Oki,S. *et al.* (2018) Ch IP-atlas: a data-mining suite powered by full integration of public ch IP-seq data. *EMBO Rep.*, **19**, 111.

Pennington,J. *et al.* (2014) GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543.

Salton,G. and Buckley,C. (1988) Term-weighting approaches in automatic text retrieval. *Inf. Process. Manag.*, **24**, 513–523.

Schreiber,J. *et al.* (2020) Avocado: a multi-scale deep tensor factorization method learns a latent representation of the human epigenome. *Genome Biol.*, **21**, 1–18.

Sheffield,N.C. and Bock,C. (2016) LOLA: enrichment analysis for genomic region sets and regulatory elements in R and bioconductor. *Bioinformatics*, **32**, 587–589.

Smith,J.P. and Sheffield,N.C. (2020) Analytical approaches for ATAC-seq data analysis. *Curr. Protoc. Hum. Genet.*, **106**, e101.

Trofimov,A. *et al.* (2020) Factorized embeddings learns rich and biologically meaningful embedding spaces using factorized tensor decomposition. *Bioinformatics*, **36**, i417–i426.

Vapnik,V. *et al.* (1997) Support vector method for function approximation, regression estimation and signal processing. In: *Advances in Neural Information Processing Systems*, pp. 281–287, Denver Colorado, USA.

Wei,D. *et al.* (2012) A novel hierarchical clustering algorithm for gene sequences. *BMC Bioinformatics*, **13**, 174.

Woloszynek,S. *et al.* (2019) 16S rRNA sequence embeddings: meaningful numeric feature representations of nucleotide sequences that are convenient for downstream analyses. *PLoS Comput. Biol.*, **15**, e1006721.

Xiong,L. *et al.* (2019) SCALE method for single-cell ATAC-seq analysis via latent feature extraction. *Nat. Commun.*, **10**, 1–10.

Yang,K.K. *et al.* (2018) Learned protein embeddings for machine learning. *Bioinformatics*, **34**, 2642–2648.

Yang,Y. and Liu,X. (1999) A re-examination of text categorization methods. In: *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.* pp. 42–49. Berkeley California, USA.

Young,T. *et al.* (2018) Recent trends in deep learning based natural language processing. *IEEE Comput. Intell. Mag.*, **13**, 55–75.