PAPER • OPEN ACCESS

Site2Vec: a reference frame invariant algorithm for vector embedding of protein–ligand binding sites

PAPER

Site2Vec: a reference frame invariant algorithm for vector embedding of protein–ligand binding sites

Arnab Bhadra[iD] and Kalidas Yeturu[iD]

Computer Science and Engineering, Indian Institute of Technology Tirupati, Tirupati, Andhra Pradesh, India

E-mail: cs18s501@iittp.ac.in and ykalidas@iittp.ac.in

Abstract

Protein–ligand interactions are one of the fundamental types of molecular interactions in living systems. Ligands are small molecules that interact with protein molecules at specific regions on their surfaces called binding sites. Binding sites would also determine ADMET properties of a drug molecule. Tasks such as assessment of protein functional similarity and detection of side effects of drugs need identification of similar binding sites of disparate proteins across diverse pathways. To this end, methods for computing similarities between binding sites are still evolving and is an active area of research even today. Machine learning methods for similarity assessment require feature descriptors of binding sites. Traditional methods based on hand engineered motifs and atomic configurations are not scalable across several thousands of sites. In this regard, deep neural network algorithms are now deployed which can capture very complex input feature space. However, one fundamental challenge in applying deep learning to structures of binding sites is the input representation and the reference frame. We report here a novel algorithm, Site2Vec, that derives reference frame invariant vector embedding of a protein–ligand binding site. The method is based on pairwise distances between representative points and chemical compositions in terms of constituent amino acids of a site. The vector embedding serves as a locality sensitive hash function for proximity queries and determining similar sites. The method has been the top performer with more than 95% quality scores in extensive benchmarking studies carried over 10 data sets and against 23 other site comparison methods in the field. The algorithm serves for high throughput processing and has been evaluated for stability with respect to reference frame shifts, coordinate perturbations and residue mutations. We also provide the method as a standalone executable and a web service hosted at (http://services.iittp.ac.in/bioinfo/home).

1. Introduction

A protein performs its function by interacting with ligands and other small molecules. Protein–ligand interaction plays an important role in biological systems. A binding site is a concavity on the surface of a protein, composed of several amino acids whose side chains interact with a ligand molecule. Electrostatic and chemical complimentarity, aromatic stacking and other forms of interactions between atoms of a concavity and a ligand molecule result in chemical binding. One protein may have a dozen concavities and some of which may be functional resulting in moonlighting behaviour. A major reason for functional similarity between proteins is the similarity of their binding sites and hence interaction patterns with ligand molecules. Binding site comparison is one of the major methods in the field of structural bioinformatics and drug discovery [1, 2].

A main source for structure of a binding site is the structure of its protein in terms of the 3D coordinates of all of its thousands of atoms. The structure of a protein is determined through a variety of means including experimental mechanisms such as x-ray crystallography or nuclear magnetic resonance and analytical methods such as sequence or structural alignment to known structures [3]. Today about 167 000-high-quality protein structures are available in the Protein Data Bank [4]. This volume of data has enabled

use of machine learning methodology for comparison of protein–ligand binding sites [5, 6]. Traditional machine learning methodologies over the last two decades require careful feature engineering and sometimes involve hand crafted features and atomic motifs. However for scaling to handle thousands of features originating in data such as medical texts, protein function, pathway information and other sources, deep learning methodologies have come into the picture [7–9].

There are two types of site comparison methods: alignment-dependent methods [10–14] and alignment-free methods [15–19]. Alignment dependent methods provide detailed information on the atomic mapping of protein structures where as alignment free methods output a similarity score. Several methods have come till date for the purpose of binding site comparison [10–28, 28–31]. A brief description of these methods is provided in (table 1).

Pairwise alignmentbased methods need a **pair of sites** as input. Representative points in 3D are mapped between the sites and a superimposition is carried out. PocketAlign [11] is an example of alignment based approach. Results of alignment provided detailed information on atomic mapping and can be visualized in standard applications. These algorithms also emit a matching score on number of atoms or residues matched between the sites.

While alignment is a detailed work for a given pair of sites, dealing with several million pairs is space and time complexity wise hard. A precomputed database of alignments and scores are stored to serve for a long list of known sites [31]. However this approach does not scale well for querying a large number of novel sites. Especially in protein design experiments for synthetic enzymes and pharmacophore based generation of binding sites [32]. Moreover in the context of pipelines of drug discovery processes involving screening against thousands of sites for side effect free ligand selection [33], more than a detailed alignment based approach, a quick and accurate filter is desirable.

The advantage of a vector descriptor based algorithm is its ability for use in diverse contexts. Queries for identification of similar binding sites reduces to proximity search in their vector descriptor space. Classifying a pair of binding sites into a category is achieved through concatenation of vectors and processing the fused vector through machine learning classifiers.

In the space of *pairwise methods*, PocketMatch [19] has been fastest [34] and competitively accurate across diverse data sets. PocketMatch considers a binding site structure as composed of amino acids represented by three point-types C_α, C_β and C_γ (centroid of side chain atoms). The 20 amino acids are grouped into five chemical categories. All pair distances between the points are then mapped into 120 bins corresponding to point-chemical-type pairs and sorted within each bin. This is the shape descriptor or finger print for a binding site.

However, PocketMatch bins are variable in length being proportional to the number of atoms and characteristic chemical composition of a site. Though the representation serves the purpose in pairwise scanario, the variable vector is not suitable for proximity queries. In this work Site2Vec, we mathematically extend the idea of PocketMatch to generate a fixed size vector which can serve as a locality-sensitive-hash descriptor.

In Site2Vec, a sliding window of numbers runs through each physico-chemical-bin of PocketMatch. The slides are all clustered for each physico-chemical-bin to form anchor points. For any new site, sliding window is run for all the bins and a histogram of cluster membership is derived. These histograms are then encoded using auto-encoder to form a 200-dimensional vector descriptor.

The algorithm has been benchmarked against 23 methods on 10 data sets. On average Site2Vec stood first in its performance for classification of sites provided in recent extensive benchmarking paper [34]. The algorithm is also robust to atomic coordinate perturbations characteristic of any site, single amino acid mutations and reference frame changes. The algorithm is fast which computes a vector descriptor at the rate of 3 to 5 sites per second, making it suitable for high throughput processing.

There has been recent work that employs deep learning for vectorization of binding sites [30]. In this method a voxel representation of 3D space featuring interaction energy of 14 atom types of a binding site is passed through a convolutional neural network (CNN) to obtain a phyisico-chemical shape descriptor. As CNNs are sensitive to reference frame, in this algorithm, a pre-processing step is carried out to determine the three axes using principal component analysis. However one major drawback of the algorithm is, as CNN is a supervised learning algorithm, it requires site categories beforehand and is not suitable in a large scale scenario across hundreds of categories and sub-categories. Moreover, the vector generation process is highly time consuming with processing abilities at the rate of 15 to 20 in one hour.

In summary, we report here a novel algorithm Site2Vec for vector representation of binding sites which is fast, reliable and robust, suitable for high throughput binding site analysis in structual bioinformatics and drug discovery fields.

Table 1. Brief descriptions of methods for binding site comparison.

Method	Description
Apoc [10]	Binding sites are aligned based on matching residue groups between sites and carrying out site alignment.
ASSIST [26]	It is a geometric hashing based comparison technique, hashes pairs of residues, taken from input structures into a 3D hash table based on residue types and distances from geometric centers. Similarity exercise is performed through alignment of hash tables.
DeepDrug3D [30]	A binding site is encapsulated in a 3D voxel-grid which is passed through a convolutional neural network to map to a vector descriptor. The method falls under supervised learning and requires categories of sites as input.
eMatchSite [25]	A sequence-order independent algorithm for binding site comparison. Physico-chemical and structural characteristics of protein residues and their interactions with small molecules are captured by the spatial distribution of residues and ligand binding probabilities.
FuzCav [15]	An alignment-free comparison method that represents a binding site as a 4833 integer vector of counts of the number of pharmacophoric triplets formed by C_α atoms.
IsoMIF [13]	A Molecular interaction-based binding site comparison method where a binding site is represented by a molecular interaction field descriptor. Interactions between atoms of residues of a site and a ligand in a protein ligand complex are input to the method.
KRIPO [18]	Quantifying the similarities at the level of sub-pockets in a binding site based on pharmacophore representation. The presence or absence of sub-pockets is then then translated to derive a fingerprint.
Morris and co-workers [28]	A binding site is represented by spherical harmonic coefficients about its centroid.
Nakumara and co-workers [16]	In their method, a binding site is represented by a 11-dimensional vector, reducing the fingerprint obtained by FuzCav [15] using principal component analysis on 4833 dimensional space.
PocketAlign [11]	A pair of binding sites are structurally superimposed based on matching residues for their local shape descriptors between sites.
PocketMatch [19]	Represents the binding site as lists containing distances between C_α, C_β and centroid of atoms of side chain and categorized into physico-chemical property bins. A pair of sites are compared for similarity by matching distance elements between corresponding bins.
Probis [12]	In this method a site is represented as a fully connected graph whose vertices are atoms. Two sites are compared based on maximal common sub-graph detection algorithms.
RAPMAD [17]	A histogram-based comparison method where several sets of pseudo centers represent a binding site. Distances between pseudo centers are calculated and mapped into histograms. These histograms represent binding sites which are compared for similarity.
SiteAlign [23]	A binding site is represented by a sphere containing physico-chemical properties on its surface. Site residues are projected on to this sphere. Two sites are compared by alignment of their sphere approximations.
SiteHopper [21]	It calculates a 3D shape representation of the active site colored by the chemical properties of the residues defining the active site. These active site representations are aligned and assessed for shape and chemical similarity.
SiteEngine [22]	Each amino acid in the site is represented by a pseudo-centre with physico-chemical label. two sites are superimposed for matching pseudo centers.
SitesBase [31]	Alignment based algorithm that superimposes atoms between a pair of sites. Alignments are pre-computed and served through a database.
SOIPPA [24]	Represents protein structures by Delaunay tessellation of C_α atoms characterized by their geometric potentials. Two structures are aligned by determining maximal weight common sub-graph.
TM-align [14]	An Alignment dependent approach where a protein structure is treated as position of C_α atoms of residues. Two structures are aligned by superimposing their backbone atoms.
VolSite [27]	A site is approximated by fitting a box into the pocket and lining the surface of the box with pharmacophoric properties. Two sites are compared through alignment of these box representations.

2. Methods and materials

Site2Vec is a mathematical enhancement to the PocketMatch [19] algorithm. In PocketMatch, pairwise distances are considered between representative points of a binding site. These distances are then put into bins corresponding to chemical types of pairs of site residues. The 20 amino acids are categorized into five group:s Group-0:(A,M,V,I,L,G,P); Group-1:(K,R,H); Group-2:(D,E,Q,N); Group-3:(Y,F,W); Group-4:(C,S,T) based on hydrophobic, hydrophilic and aromatic ring properties. Each amino acid of the

site is represented by points - C_α, C_β and C_γ for the centroid of the side-chain (including C_β). The PocketMatch algorithm takes as input *a pair of sites*. For a given site a variable dimensional descriptor is generated and for two sites, a similarity score is computed. As with any other pairwise method, limitations of PocketMatch include difficulty in serving proximity queries. For instance, if the query is to identify all sites that are proximal to a given novel nucleotide-binding site, it has to resort to several thousands of pairwise comparisons. The Site2Vec algorithm attacks the descriptor variability issue in PocketMatch. Vector generation in Site2Vec involves sliding window based processing of PocketMatch bins, clustering and auto-encoder based deep neural network derived embedding, the steps of which are described in the sections that follow.

2.1. Site2Vec: Sliding window-based encoding

Site2Vec takes as input 3D representation of a binding site and encodes it into a d-dimensional vector representation (figure 1).

(a) **Representation of atoms in a binding site**

- Let e be a binding site in dataset Γ and $A(e)$ be the atoms of e.
- $A(e) = \{(x_i, y_i, z_i, \phi_i, \psi_i) | i \in [1..N]\}$
- where,
  - x_i, y_i, z_i :3D coordinates of $A(e)$.
  - $\phi_i \in \alpha \, \beta \, \gamma$.
    - $\alpha : C_\alpha$
    - $\beta : C_\beta$
    - γ : Centroid of the side chain.
  - $\psi_i \in [1..k]$
    - where, $k = 5$ [19] 20 amino acids were considered in 5 groups – Group-0:(A,M,V,I,L,G,P); Group-1:(K,R,H); Group-2:(D,E,Q,N); Group-3:(Y,F,W); Group-4:(C,S,T).
  - N is number of atoms

(b) **Calculation of pairwise distances among the atoms of binding site**

- $D(e) = (dist(i,j), type(i,j) | \forall i,j \in [1..N]$
- where,
  - $dist(i,j) : \sqrt{\sum\limits_{p \in \{x,y,z\}} (A(e)[i].p - A(e)[j].p)^2}$
  - $type(i,j) : A(e)[i].\phi\_A(e)[j].\phi\_A(e)[i].\psi\_A(e)[j].\psi$
  - such that,
    - $\alpha\_\beta \equiv \beta\_\alpha$
    - $\alpha\_\gamma \equiv \gamma\_\alpha$
    - $\beta\_\gamma \equiv \gamma\_\beta$
    - $\psi_a\_\psi_b \equiv \psi_b\_\psi_a (\forall \psi_a \neq \psi_b)$

(c) **Grouping of distances**

- Let $|m|$ be the number of groups.
- $\forall m \in \{x[1] | x \in D\}$
  - $L_m(e) = [x[0] | x \in D, x[1] = m] \land (\forall a < b) \rightarrow L_m[a] \leq L_m[b]$

(d) **Sliding a sliding window of size ξ over the group of distances (calculated in the previous step)**

- $S_m^\xi(e) = [L_m(e)[i : i + \xi] | 1 \leq i \leq |L_m(e)| - \xi + 1]$

(e) **Creating a bag of features.**

- $S_m = \{S_m^\xi(e) | e \in \Gamma(Dataset)\}$

(f) **Quantization**

- $O_m = Kmeans(S_m, k)$ (where k is the number of clusters)

(g) **Mapping of each list (groups) to a histogram**

- $\forall e \in \Gamma$
  - $C_m(e) = [O_m.nearest(s) | s \in S_m^\xi(e)]$
  - $H_m(e) = hist(C_m(e))$
  - where,
    - $nearest(.)$ returns nearest cluster number [35].
    - $hist(.)$ function generates histogram of cluster numbers.

(h) **Feeding histogram to an auto-encoder [36], results in *d* dimensional vector encoding**

- $S_{m,d}^\xi(e) = \Psi(H_m(e), \theta^*)$
- where,
  - Ψ: is an auto-encoder
  - $\theta^* : \arg\min_\theta \sum_{e \in \Gamma} \|\Psi(H_m(e)) - H_m(e)\|^2$
  - $S_{m,d}^\xi(e)$: d-dimensional vector representation of binding site e using auto encoder Ψ.

2.1.1. Training process

In Site2Vec, a sliding window (refer step d) passes over the list representation of binding sites (refer step c) that generates equal length slices of distances of C_α, C_β and γ (refer step d) and the slices of distances are mapped to k clusters such that each slice maps exactly one cluster (refer step g). The default setting of this practice includes the size of sliding window $\xi = 10$ (refer step d) and the number of clusters $k = 10$ (refer step f). The default dimension of vector descriptor of binding site is $d = 200$ (refer step h) where a protein–ligand binding site is represented as m number of lists where $m = 120$ [19] (refer step d).

A 1D auto-encoder deep neural network is trained that results in d dimensional vector encoding of binding sites (refer step h). The auto-encoder consists of four hidden layers, so a total of six layers, including input and output layers followed by the ReLU activation function. The configuration of the auto-encoder is $1200 \times 1000 \times 800 \times 600 \times 400 \times 200$. The network is trained on a batch size of 16 with a learning rate of 0.0001. The setup mentioned above is the default configuration used in the study.

2.2. Simple mean-variance based vector representation

In the previous section 2.1, we have mentioned that each binding site is represented as a d-dimensional vector. In an intermediate step (step c) in section 2.1, a site is described a m groups of distances between atoms. The first simple approach is to compute mean and variance for each of the groups mentioned in step c and generate a m-dimensional vector. The pseudo-code of this mean-variance based method is described in Algorithm 1.

Algorithm 1 Mean-variance method

Input: e be a binding site.
Output: e_v: Vector representation of the binding site.
1: $e_\mu = [MEAN(L_i(e))]\big|_{i=1}^{|m|}$ (L_i, same as step c in section 2.1)
2: $e_\sigma = [VARIANCE(L_i(e))]\big|_{i=1}^{|m|}$ (L_i, same as step c in section 2.1)
3: $e_v = e_\mu \cdot e_\sigma$

2.3. Uniform-Site2Vec: Uniform distribution of centroids based binding site encoding method

In section 2.1, Site2Vec method is discussed and in step f , $K - means$ clustering algorithm [37] is used for grouping of S_m (refer step e in 2.1) in k clusters. Another simple and time-effective approach is to generate uniformly k pseudo centers in ξ space that represent k clusters. The rest of the steps as well as the training process (2.1.1) are same as in 2.1. This version of the method we call *Uniform-Site2Vec*.

2.4. Discretized distance and histogram-based binding site representation method

In this approach, distances between C_α, C_β and γ (Step b) are discretized into seven intervals (table 2).

Table 2. Discretization of distances between atoms into seven intervals [15]

Interval	Distance range(in Å)
1	$[0, 7.6)$
2	$[7.6, 10.1)$
3	$[10.1, 12.3)$
4	$[12.3, 14.3)$
5	$[14.3, 16.8)$
6	$[16.8, 20.0)$
7	$[20.0, \infty)$

Table 3. Description of data sets used in [34] for large scale evaluation of methods for binding site comparison.

Data set	Description
D1	Structures having identical sequences
D1.2	Structures with identical sequences and similar ligands
D2	NMR structures.
D3	Discrimination between binding sites having different physicochemical properties
D4	Discrimination between binding sites having different physicochemical and geometrical properties
D5	Kahraman dataset [43]
D6	Barelier dataset [44]
D7	Binding site within known set of proteins

The distance elements within each PocketMatch based bin (refer step b and c)) are mapped to these intervals and histogram of counts of number of elements in each interval (table 2) are computed. And finally, for each group ($L_m(e)$), the histogram of the interval number is generated. Concatenation of these histograms for all the groups is considered a vector representation (Algorithm 2).

Algorithm 2 Discretized distance and histogram-based method

Input: e be a binding site.
Output: $e_v^{\mathbf{B}}$: Vector representation of binding site.
1: $\mathbf{B} = [\beta_1, \beta_2, \ldots, \beta_7]$
2: $X_i(e) = [\beta | \exists \beta \in \mathbf{B}: \beta_k \le L_i(e)[j] < \beta_{k+1}, \text{ where } k \in (1,6)]_{j=1}^{|L_i(e)|}$
3: $X(e) = [X_i(e)]_{i=1}^{|m|}$
4: $e_v^{\mathbf{B}} = [hist(X_i(e))]_{i=1}^{|m|} \backslash (hist(.) \text{ returns histogram of bin numbers.})$

2.5. Benchmarking datasets and methods

Site2Vec algorithm has been evaluated on 10 different datasets and 23 existing methods for binding site comparison. The first study involves benchmarking of Site2Vec against 21 methods and seven datasets reported in [34]. The second study involves the evaluation of the algorithm on ApocS3 Dataset [10]. The third study involves the evaluation of Site2Vec on a data set of 2249 binding sites covering a class of five types of sites [42]. The fourth study involves the evaluation of the algorithm against another deep-learning based method in the literature [30].

2.5.1. Benchmarking on ProSPECCTs datasets and methods

Site2Vec has been evaluated against 21 methods on seven datasets based on [34].
Each of the methods are described in (table 1). The datasets are described in (table 3).

2.5.2. PLIC: protein–ligand interaction clusters

PLIC stands for Protein–ligand interaction clusters. It is a relational database that categorizes the ligand-binding sites based on different attributes [42]. The database mentioned above contains 67 550 unique protein–ligand binding sites which are divided into 10 854 groups based on pocket shape, types of residues, atomic contacts, and binding energy. Among these binding site groups, five groups that have more than 300 binding sites have been extracted for evaluating this method as any machine learning model requires a sufficient amount of data to learn each class properly. The first group has 537 binding sites. The second, third, fourth, and fifth have 513, 473, 384 and 342 binding sites respectively.

2.5.3. ApocS3 Dataset

ApocS3 dataset is composed of subject and control data which consists of sites from 2090 and 21, 660 unique protein chains from PDB, respectively [10]. There are 38 066 pairs of binding sites in subject set (positive
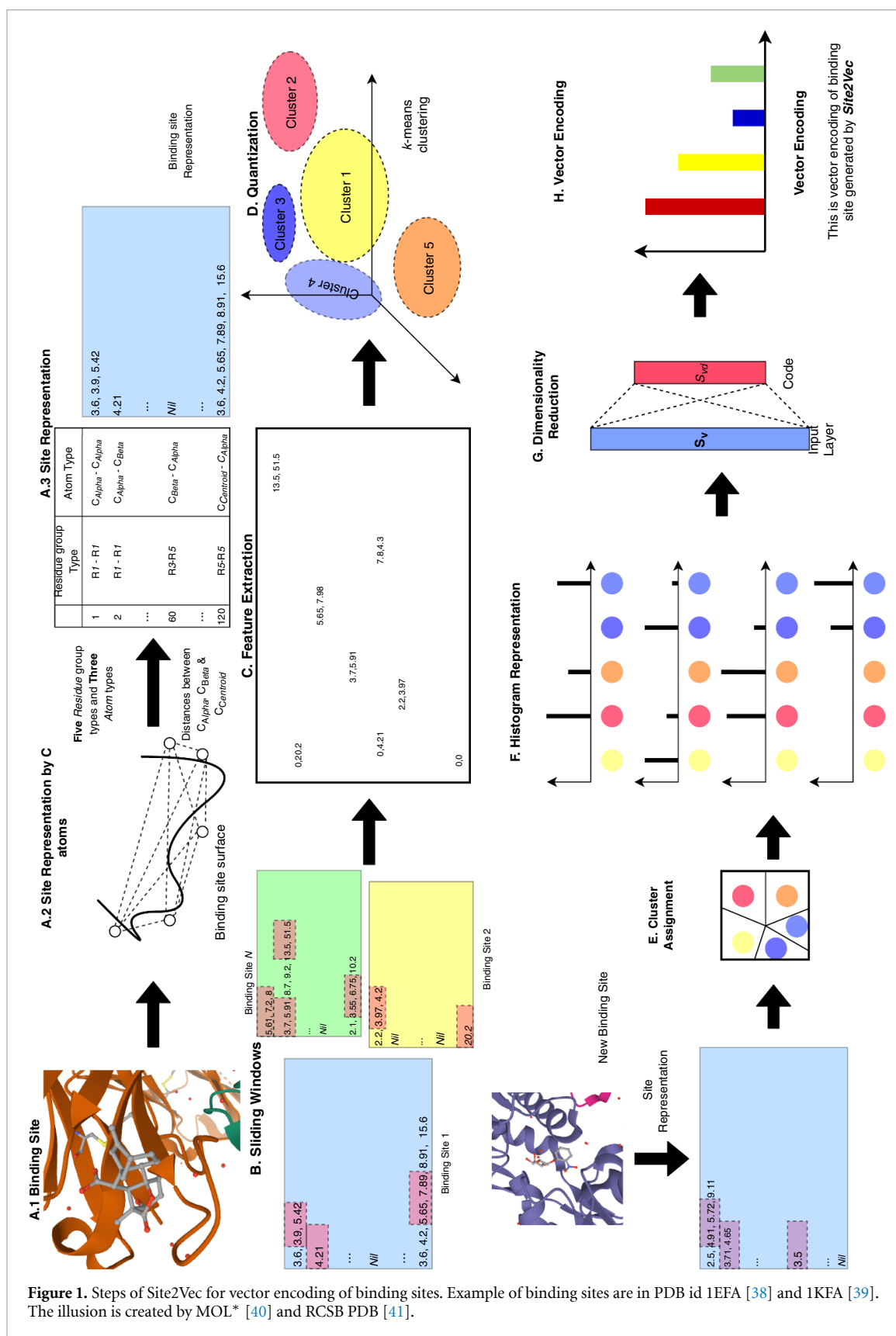
Figure 1. Steps of Site2Vec for vector encoding of binding sites. Example of binding sites are in PDB id 1EFA [38] and 1KFA [39]. The illusion is created by MOL* [40] and RCSB PDB [41].

pairs), and 38 066 pairs of binding sites in the control set (negative pairs). Subject set corresponds to known similar sites and control set corresponds to known dissimilar sites.

2.5.4. Comparison against nucleotide and heme binding sites

TOUGH-C1 dataset compiled by [45] consists of nucleotide and heme binding sites. This data set consists of 1556 nucleotide pockets and 556 heme-binding sites. We have used this data set for benchmarking of Site2Vec with respect to [30] which is another recent study on using convolutional neural networks for binding site classification.

2.6. Evaluation technique

A two class clarification algorithm can be evaluated on standard metrics such as precision, recall, accuracy and AUC [46]. AUC stands for area under the curve of a graph plotted against true positive and false positive rates of prediction of a classifier at various thresholds. The performance of binding site comparison and classification methods is represented by the receiver operating characteristic (ROC) curve [47]. Area under the curve (AUC) of ROC is a standard quality assessment metric. ROC curve plots true positive rate (TPR) with respect to false positive rate (FPR) of a classifier about several threshold values on its confidence score. In the ROC curve, TPR is plotted on the *Y*-axis and FPR is plotted on the *X*-axis. The TPR and FPR are defined below:

$$TPR = \frac{TP}{TP + FN}$$

and

$$FPR = \frac{FP}{FP + TN}.$$

Besides the ROC curve, there are other measuring tools like Accuracy, Precision and Recall that are also used to evaluate the performance of protein–ligand binding site comparison methods. These metrics are described as

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

where TP is the number of true positives, FP is the number of false positives, TN is the number of true negatives and FN is the number of false negatives. In this exercise of benchmarking against other methods [15, 16, 30] we have used AUC as a quality assessment metric. The state of the art methods given in [34] provide a similarity score between a pair of sites which is used to compute ROC curve.

2.7. Run time evaluation

In this study, we have computed the run time performance of Site2Vec and compared it against [30]. The run time is computed on an Intel Xeon workstation, having CPU E5-2640 and with 2.40 GHz, 40 cores and 125GB RAM and Ubuntu 16.04.5 LTS is installed. The site comparisons have happened in a serial manner.

3. Results

Site2Vec algorithm has been evaluated on 10 different datasets and several methods for binding site comparison. The first study involves benchmarking of the method against 21 methods and seven datasets reported in [34]. The second study involves evaluation of the algorithm on ApocS3 dataset[10]. The third study involves the evaluation of the method on a data set of 67 550 binding sites covering a class of five types of sites [42].

The fourth study involves the evaluation of the algorithm against another deep-learning based method in the literature [30].

3.1. Benchmarking against existing binding site comparison methods on ProSPECCTs datasets

The Site2Vec model is trained on whole PLIC dataset [42] and represents a binding site as a 200-dimensional vector. For each of the ProSPECCTs Datasets [34], Site2Vec vector is generated for each site. For a pair of sites, the corresponding vectors are fused to result in a concatenated vector. The fused vector is then used in classification tasks. For finding similarity, each dataset is divided into two halves randomly. A data-driven classifier is trained by one half. The classifier is tested by the remaining half of the data and AUC of ROC curve measures performance. Our approach is compared to 21 different binding site comparison methods mentioned in ProSPECCTs. For all the previous methods, the AUC scores are directly obtained from [34] without having to run the methods locally again. Given that Site2Vec is a vector embedding generator, it needs to have train and test sets for application in a classification scenario. To this end, we have considered each data set provided in [34] and split it into train and test sets. A classifier (random forest [48] having 100 trees) is trained on the train set and evaluated for AUC value on the test set. Instead of single random split of train and test sets, we have randomly split the data set into two halves 50 times and computed the average AUC value.

Dataset D1 consists of structures having identical sequences that bind to different ligands. It has 13 430 similar pairs and 92 846 dissimilar pairs. Dataset D1.2 evaluates sensitivity to the binding sites with identical sequences and similar ligands. Our proposed approach achieved AUC scores of 1 and 0.94 on D1 and D1.2, respectively. Probis [12], SMAP [49], TM-align [14] have also performed very well and got AUC of 1 on both datasets. From the AUC scores of 1 and AUC 0.94, we can conclude that Site2Vec is robust method for the structural definition of binding sites.

Dataset D2 measures the sensitivity with respect to the pocket flexibility of Nuclear Magnetic Resonance (NMR) structures. Site2Vec has scored AUC of 1 which concludes that it is sensitive to binding site flexibility.

Two decoy datasets D3 & D4 are compiled to measure the difference between nearly similar sites differing by physicochemical properties (D3) and both physicochemical and geometrical properties by performing five mutations as mentioned in ProSPECCTs [34]. Site2Vec scores better (AUC 0.99) than the other 21 methods. Hence, we conclude that our Site2Vec is robust in handling mutation as well.

Next, dataset D5 is compiled by [43] and has 10 000 pairs of binding sites for comparison. On this data set, Site2Vec achieves an AUC of 0.86 which is highest among other methods (table 4). Dataset D6, compiled by [44], comprises of protein–ligand binding sites with identical ligands. On dataset D6, using default random forest classifier with 100 trees and unlimited depth, our approach performs moderately by scoring AUC of 0.53 where KRIPO [18] outperforms all other methods, including Site2Vec with an AUC of 0.73. However using Gradient Boosting Classifier on this data set, we obtain an AUC value of about 0.75. These experiments clearly indicate that by doing a hyper parameter grid search, Site2Vec vectors can be used to improve classifier accuracy and AUC metrics.

Lastly, dataset D7 is compiled to measure the recovery of known binding site similarities within a set of diverse proteins. There are only 115 positives and 56 284 negatives in this data set. It has heavy bias towards negative data set. Site2Vec achieves an AUC of 0.66, which is lower than most pocket matching algorithms when default random forest classifier is used. However, when Ada Boost Classifier is used with 100 trees, the AUC value boosts up to 0.80. We can therefore infer that Site2Vec vectors can be considered along with machine learning model hyper parameters to result in high quality scores.

On most datasets, Site2Vec is outperforming other existing methods. The benchmarking study supports the claim that vectorization approach is robust and reliable vectors are generated out of binding sites in Site2Vec. In cases where a random forest classifier was not suitable, an alternative ensembling method such as Ada Boost or Gradient Boost methods with appropriate parameter settings have resulted in high AUC scores. This exercise infers that Site2Vec vectors are of good quality and can be used in classification scenarios with appropriate machine learning model selection through hyper parameter search.
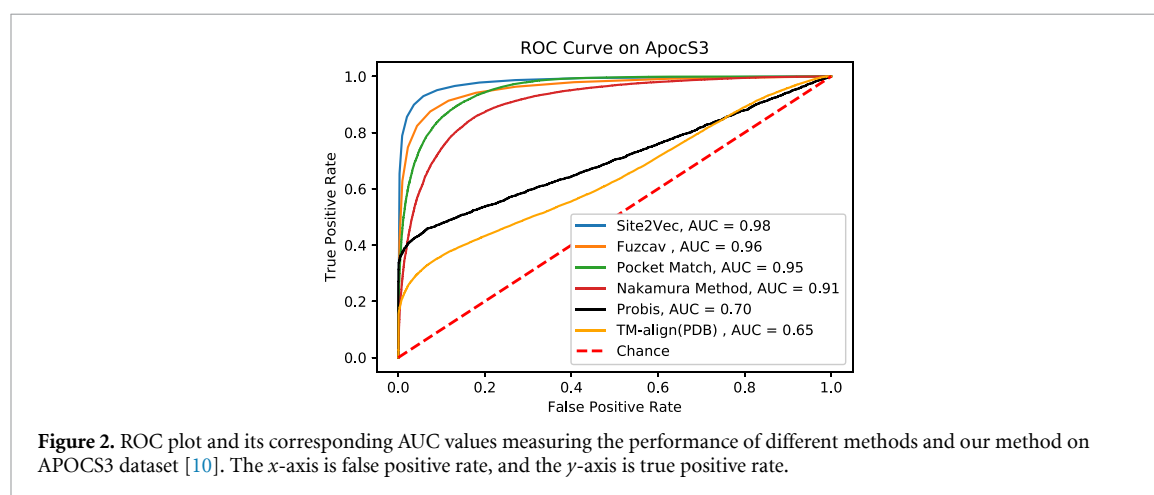
3.2. Evaluation of quality of vector representation on APOCS3 Dataset

ApocS3 dataset [10] consists of 38,066 pairs of binding sides in subject (positive pairs) as well as 38,066 pairs of sites in control (negative pairs) set. The positive data denotes known set of similar sites and the negative data denotes known set of dissimilar sites.

Using Site2Vec, vector fingerprints of all 23 750 binding sites are generated. To find a similarity between two sites, the whole dataset (Subject and Control dataset) is divided into two halves. One half is used as a training data, and the other half is treated as test data such that both train and test sets have an equivalent amount of subject and control data. The pair of binding sites in the training data are fed to a classifier (random forest classifier containing 1000 trees) to train the similarity between two ligand-binding sites, and test data is used to predict the similarity score. The performance is measured by receiver operating characteristic (ROC) curve and its corresponding area under the curve (AUC) score.

Table 4. AUC values for 21 existing site similarity methods and Site2Vec on ProSPECCTs datasets.

Method	D1	D1.2	D2	D3	D4	D5	D6	D7	Average AUC across data sets
Site2Vec*	1.00	0.94	1.00	0.99	0.99	0.86	0.53	0.66	0.87
SiteHopper [21]	0.98	0.94	1.00	0.75	0.75	0.81	0.56	0.77	0.82
SiteEngine [22]	0.96	1.00	1.00	0.82	0.79	0.57	0.55	0.86	0.82
SiteAlign [23]	0.97	1.00	1.00	0.85	0.8	0.57	0.44	0.87	0.81
SMAP [49]	1.00	1.00	1.00	0.76	0.65	0.54	0.68	0.86	0.81
KRIPO [18]	0.91	1.00	0.96	0.6	0.61	0.77	0.73	0.85	0.80
Shaper [27]	0.96	0.93	0.93	0.71	0.76	0.65	0.54	0.75	0.78
Shaper(PDB) [27]	0.96	0.93	0.93	0.71	0.76	0.64	0.54	0.75	0.78
VolSite/Shaper [27]	0.93	0.99	0.78	0.68	0.76	0.58	0.71	0.77	0.78
FuzCav [15]	0.94	0.99	0.99	0.69	0.58	0.54	0.67	0.77	0.78
FuzCav(PDB) [15]	0.94	0.99	0.98	0.69	0.58	0.54	0.65	0.77	0.77
TM-align [14]	1.00	1.00	1.00	0.49	0.49	0.62	0.59	0.88	0.76
Cavbase [50]	0.98	0.91	0.87	0.65	0.64	0.57	0.55	0.82	0.75
VolSite/Shaper(PDB) [27]	0.94	1.00	0.76	0.68	0.76	0.56	0.5	0.72	0.74
IsoMIF [13]	0.77	0.97	0.7	0.59	0.59	0.81	0.62	0.87	0.74
PocketMatch [19]	0.82	0.98	0.96	0.59	0.57	0.6	0.51	0.82	0.73
ProBiS [12]	1.00	1.00	1.00	0.47	0.46	0.55	0.5	0.85	0.73
TIFP [20]	0.66	0.9	0.91	0.66	0.66	0.63	0.55	0.71	0.71
RAMPAD [17]	0.85	0.83	0.82	0.61	0.63	0.52	0.6	0.74	0.70
Grim [20]	0.69	0.97	0.92	0.55	0.56	0.61	0.45	0.7	0.69
Grim(PDB) [20]	0.62	0.83	0.85	0.57	0.56	0.58	0.45	0.64	0.64
TIFP(PDB) [20]	0.55	0.74	0.78	0.56	0.57	0.53	0.56	0.66	0.62

Figure 2. ROC plot and its corresponding AUC values measuring the performance of different methods and our method on APOCS3 dataset [10]. The *x*-axis is false positive rate, and the *y*-axis is true positive rate.

We compared our proposed method with alignment dependent methods (i) TM-align [14] and Probis [12] and (ii) alignment independent methods: Fuzcav [15], a method by Nakumara and co-workers [16] and PocketMatch [19]. Brief description of these methods can be found in (table 1).

The measured ROC curves are displayed in (figure 2). Our proposed method achieved an area under the curve of 0.983 and is outperforming all other methods discussed in this study. FuzCav (AUC 0.96), PocketMatch (AUC of 0.95), Nakamura method (AUC of 0.91) are also performing better on the ApocS3 dataset. Alignment dependent binding site similarity methods Probis has AUC of 0.70 and TM-align achieves AUC of 0.65. This result provides an insight that the quality of vector encoding of binding sites computed by Site2Vec is acceptable and it is noted that our method is capable of doing multiple comparisons of binding sites at a single point of time. The proposed method converts binding sites to vectors in *d*-dimensional space and compares vector representations in that multidimensional space.

3.3. Benchmarking against DeepDrug3D

Another method available today that employs deep neural networks for binding site embedding is DeepDrug3D [30]. This is a classification algorithm that uses convolutional neural networks (CNN) to classify a given data set of categories of binding sites. In the process, it generates a vector embedding once the training is complete for a given binding site. In this method, a binding site is represented as a voxel representation with a 3D grid. For each grid point, interaction energy is calculated for 14 atom types. The voxel-grid is then passed through a convolutional neural network which regresses over a loss function based

Table 5. Performance measure of Site2Vec and other binding site comparison methods on the ApocS3 dataset.

Method	Accuracy (threshold)	AUC	Precision	Recall
Site2Vec	**0.93(0.5)**	**0.98**	**0.94**	**0.93**
FuzCav	0.90(0.5)	0.96	0.88	0.91
PocketMatch	0.86(0.1)	0.95	0.92	0.79
Nakamura method	0.84(0.5)	0.91	0.80	0.89
Probis	0.61(0)	0.70	0.90	0.59
TM-align	0.62(0.4)	0.65	0.89	0.28

Table 6. Performance analysis of Site2Vec and Deepdrug3D to classify the binding sites on PLIC data set.

Method	Accuracy	AUC
K-NN on Site2Vec	0.97	0.990
Decision tree on Site2Vec	0.95	0.976
Random forest on Site2Vec	**0.98**	**0.992**
DeepDrug3D	0.96	0.991

on category labels of the input sites. As with any CNN, the method is sensitive to input representation and reference frame. To address this, the authors use a pre-processing step where principal component analysis (PCA) is carried out to determine X, Y and Z axes and align the site and translate the origin to the centroid of the site.

However, fundamental difference between DeepDrug3D and Site2Vec is, the former is fundamentally a classification technique that requires categories of binding sites in the data set. Site2Vec is a vector embedding technique that does not require categories of binding sites to generate vector representation. A plain vector embedding generator has clear advantages over fixed classifiers as it scales to several thousands of sites across diverse categories in dynamic pipeline processing scenarios.

The first benchmarking we did was on PLIC [42] data set for both AUC scores and the second exercise was on TOUGH-C1 data set [45]. We can very clearly see that Site2Vec is faster at scale (about 3400 times) than DeepDrug3D and performs better or at par with the latter method.

3.3.1. Performance on PLIC data set
Comparison studies are carried out against another deep learning based method Deepdrug3D [30] on the same data set. figure3 indicates the ROC curve for the PLIC dataset for both the methods. We can observe that both the methods have obtained very high AUC of above 0.99.

In addition to a random forest classifier, other classifiers are also considered while performing the evaluation process. K-nearest neighbor classifier and Decision tree [51] are used to assess the performance of vector representation of Site2Vec. Table 6 shows the accuracy, area under the curve of ROC plot of Site2Vec using different classifiers and Deepdrug3D on the PLIC dataset. We can observe that Site2Vec has slightly out performed [30] on this data set. The Site2Vec algorithm comes with a very clear advantage of ultra fast execution speed compared with [30]. It is at least *3400 times faster* for vector generation. The main reason why [30] is slow compared to Site2Vec is, it is a CNN based method which is fundamentally a classification approach. It requires full level of training to be able to generate a vector embedding. Whereas in Site2Vec, absolute representation of site as in [19] is used followed by a normalization step of clustering and auto-encoder to generate generate embedding which is faster at scale.

3.3.2. Performance on TOUGH-C1
The TOUGH-C1 dataset [45] has four sub-sets: nucleotide-, heme-, steroid- and control-binding pockets.

In this benchmarking exercise, heme-binding sites are treated as positive class data, and nucleotide sites are considered a negative class. The whole dataset of TOUGH-C1 is divided into 70% of training data and 30% test data. Site2Vec is used to generate vector fingerprint from the binding site structure, and a classifier (random forest; number of trees is 100) is used to classify protein–ligand binding sites using their vector form. The performance is measured using the ROC curve. The resultant ROC is shown in figure 4. Site2Vec has got AUC of 0.92, and DeepDrug3D achieved an AUC of 0.89. Table 7 gives more insight into the performance of Site2Vec and Deepdrug3D on the TOUGH-C1 dataset from which we can infer that Site2Vec has performed well heme and nucleotide binding sites of the TOUGH-C1 dataset. The TOUGH-C1 dataset [45] has two other sub-data sets, called control and steroid sets on which [30] has been tested. We chose to not evaluate against these sub-data sets as we felt it is not required in the context of a large scale evaluation already happened against [34]. In [34], Site2Vec, has out performed several other algorithms and has been
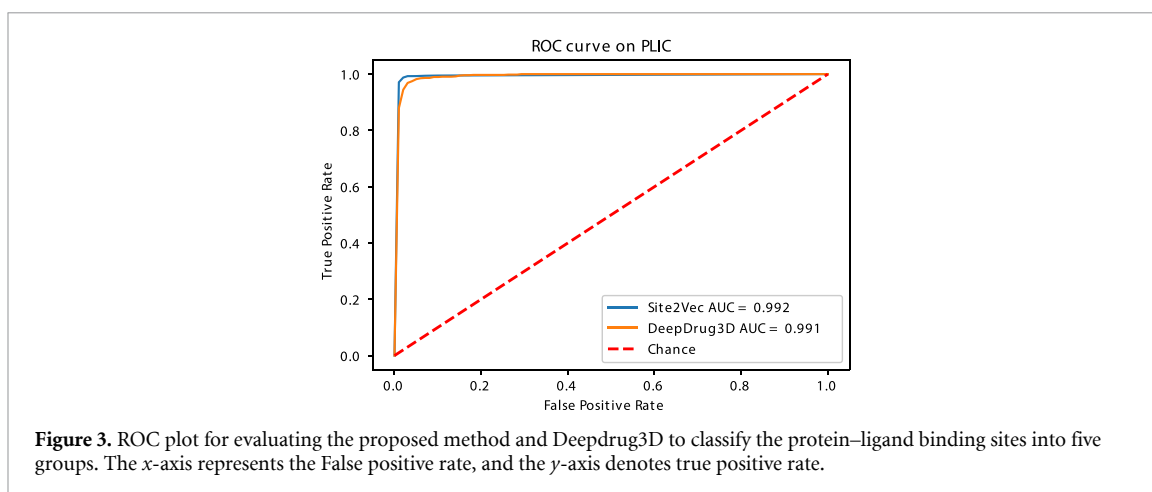
Figure 3. ROC plot for evaluating the proposed method and Deepdrug3D to classify the protein–ligand binding sites into five groups. The *x*-axis represents the False positive rate, and the *y*-axis denotes true positive rate.
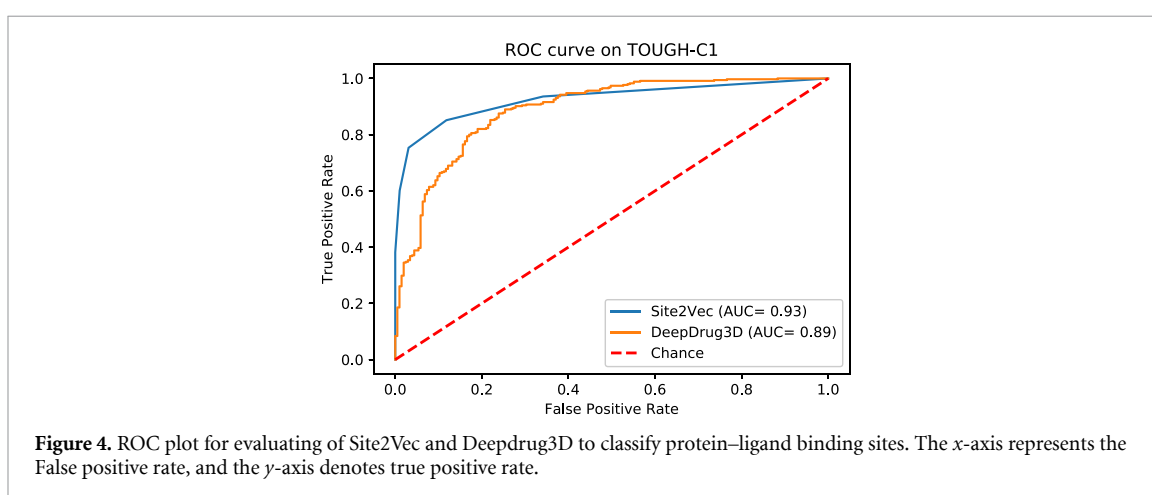
Figure 4. ROC plot for evaluating of Site2Vec and Deepdrug3D to classify protein–ligand binding sites. The *x*-axis represents the False positive rate, and the *y*-axis denotes true positive rate.

Table 7. Performance measure of Site2Vec and DeepDrug3D on the classification of nucleotide- and heme-binding sites.

Method	Accuracy	AUC	Precision	Recall
Site2Vec method	**0.90**	**0.92**	**0.88**	0.71
DeepDrug3D	0.84	0.89	0.85	**0.89**

the top performing algorithm. Moreover, the amount of time requirement by DeepDrug3D [30] takes several days to weeks on [34] benchmarking data sets to carry out any rigorous large scale study.

3.4. Comparison of variant of alternative vectorizations of Site2Vec on TOUGH-C1 Dataset

The idea of this exercise is to assess different vectorization alternatives under Site2Vec problem formulation. Once distance elements are available in 120 bins, vectorization can happen using any of the approaches as in section 2. We have used TOUGH-C1 data set [45] to compare within Site2Vec formulation and decide on which vector generation variant is promising. We have used nucleotide and heme binding sites as two categories against which tested Site2Vec internal variations.

Here, heme-binding sites are treated as positive class data, and nucleotide sites are considered as negative class. The whole dataset is divided into 70% of train data and 30% of test data. From the binding site structures, methods described in section 2, are used to generate vector signatures and then classifiers are used to classify the sites using the vector representations of ligand-binding sites. Table 8 reports the performance of binding site vector encoding methods, and the performance is measured using accuracy and area under ROC. Apart from the performance measure, the time required to encode binding sites by different proposed methods is also calculated and the average time taken by each method is listed in table 8. It is observed that Site2Vec using PLIC [42] as clustering data set has out performed other variants. Based on this exercise we have set default Site2Vec configuration to use clustering model for sliding window vectors from PLIC data set.

Table 8. Performance measure of different encoding methods on the classification of nucleotide- and heme-binding sites.

Method	Accuracy	AUC	Precision	Recall	Avg. Time
Site2Vec method	**0.90**	**0.92**	**0.88**	**0.71**	0.32 s
Mean variance method	0.81	0.83	0.67	0.61	0.09 s
Uniform-Site2Vec	0.88	0.90	0.84	**0.71**	0.39 s
Discretized distance method	0.86	0.90	0.87	0.62	**0.08 s**

Table 9. Run Time analyses of Site2Vec and other binding site comparison algorithms for comparison binding sites.

Method	Avg. Comparison time(s)
Pocket Match	0.009 s
Site2Vec*	0.03 s
TM-Align	0.33 s
Probis	1.99 s

Table 10. Run Time analyses of Site2Vec and DeepDrug3D.

Method	Avg. run time(s)
Site2Vec*	0.34 s
DeepDrug3D	1 228.24 s

3.5. Run Time analyses

In this exercise we have estimated time of execution of the Site2Vec algorithm and compared against other methods in the state of the art. Apart from site vector quality benchmarking exercises, In this exercise, 30 structures were randomly chosen from a data set by Kahraman and co-workers [43] and running times were calculated on Site2Vec PocketMatch [19], TM-align [14] and Probis [12]. Brief descriptions of the above methods can be found in (table 1). After performing this experiment (table 9), it is found that among all the methods studied here as well as Site2Vec, PocketMatch is the most computational cost-efficient method. On average, PocketMatch takes 0.009 s to compare pairs of binding sites where Site2Vec consumes average 0.03 s for comparison. TM-align and Probis compare two bind sites, on average 0.33 s and 1.99 s, respectively. We can observe that Site2Vec is fastest in single site vector embedding generation methods in these methods and data set.

Deepdrug3D [30] follows a different approach for classifying the binding sites. It uses 3D voxel to represent a binding site and this voxel is fed into CNN for classification. Another benchmarking exercise is carried out to measure the time taken to convert binding sites into vector forms by Site2Vec and voxels form by Deedrug3D. In this study, twenty binding site is taken from [45]. Deepdrug3D is computationally very expensive method with respect to run time. It takes on an average of 1 228.24 s and Site2Vec is taking 0.34 s to encode a binding site. Site2Vec is about 3400 times faster than [30] on this data set. Both computational cost measure exercises are performed on Intel Xeon workstation, details mentioned in section 2.7.

4. Discussion

4.1. Data visualization

Site2Vec is an algorithm that generates vector embedding for a binding site. It is different from any pairwise comparison method which takes two sites as input and compute a score. The advantage of a pure vector embedding generator method is its usefulness as a locality sensitive hashing function. Proximity queries for binding sites in the vector embedded dimensional space would correspond to similar binding sites. To evaluate the quality of the vector representation, we have considered PLIC data set [42] and generated vectors for each of the sites. The data set has 5 categories of ligands. The 200 dimensional vectors hence generated using Site2Vec are then projected onto a 2-dimensional space and plotted for visual inspection. We have used t-distributed stochastic neighbor embedding (t-SNE) [52] for the projection purpose.

We can observe (figure 5) that similar sites have occured in proximal locations. As can be seen from the colour of the points. Sites belonging in same class have same colour. This experiment supports the hypothesis that the vector embeddings generated by Site2Vec capture spatial proximity well. It is the first novel locality sensitive hashing function in the state of the art in the space of binding site vectorization algorithms.
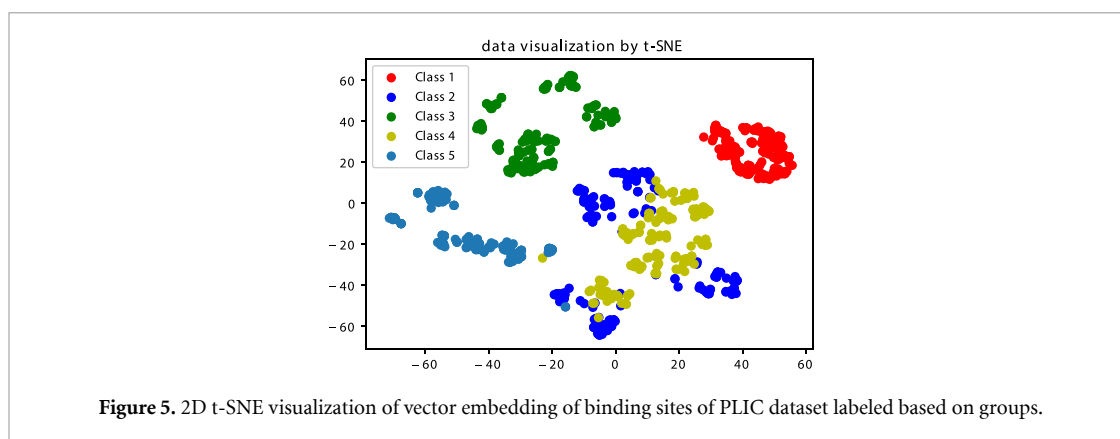
Figure 5. 2D t-SNE visualization of vector embedding of binding sites of PLIC dataset labeled based on groups.

4.2. Metrics for comparison of binding site pair

Protein–ligand binding site comparison methods use different techniques to compare sites. The metric used for computing similarity is closely tied to the way data is represented. For instance alignment based method such as PocketAlign [11] uses the number of mapped residues between sites and SitesBase [31] uses matching atoms between sites. In case of alignment free methods, PocketMatch [19] uses number of matching distance elements between sites, RAPMAD [53] uses the Jensen–Shannon divergence between histograms of distance elements between sites and FuzCav [15] uses intersection of fingerprint vectors.

In Site2Vec method, cosine similarity (equation (1)) and Euclidean distance (equation (2)) between vector embeddings are used metrics for similarity and dissimilarity respectively. Cosine similarity captures directionality of the sites where as Euclidean distance only looks at magnitude of separation between two sites and algebraically both are related quantities given angle between the two vectors. These metrics are used in sensitivity analysis of the method:

$$cosineSimilarity = \frac{\mathbf{A}.\mathbf{B}}{\| \mathbf{A} \| \| \mathbf{B} \|} = \frac{\sum_{i=0}^{d-1} A_i B_i}{\sqrt{\sum_{i=0}^{d-1} A_i^2} \sqrt{\sum_{i=0}^{d-1} B_i^2}} \tag{1}$$

where **A** and **B** are the vector signatures of two binding sites; a(A,B) are the Euclidean distance between A and B; and d is the dimension dimension of vector embedding of the binding site.

The cosine similarity ranges between -1 and 1. The value -1 implies that two vectors are opposite, and 1 indicates that the two vectors are the same and similarity value of 0 indicates orthogonality. The intermediate values indicate intermediate similarity.

Euclidean distance between the pair of vector signatures of binding sites is used to calculate the similarity between two binding sites:

$$e(\mathbf{A}, \mathbf{B}) = \sqrt{\sum_{i=0}^{d-1} (A_i - B_i)^2} \tag{2}$$

where **A** and **B**: Vector signatures of two binding sites; $e(\mathbf{A},\mathbf{B})$ are the Euclidean distance between **A** and **B**; and d is the dimension of vector representation of the binding site.

These two comparison techniques are used in the sensitivity analysis of this method.

4.3. Sensitivity Studies

The Site2Vec algorithm is evaluated on three types of perturbations of sites for assessing its sensitivity. The first evaluation is to perform sanity test with respect to rotation. The second study is is to induce perturbations in atomic coordinates and evaluate robustness. The third study is to assess the robustness of vectorization with respect to minor changes in residue compositions. These three types of perturbations can be expected in any naturally occuring binding sites across protein families and to account for effects of molecular dynamics in protein structures in a living system.

4.3.1. Sensitivity analysis based on reference frame rotations

Translation and rotation of coordinates of atoms in a binding sites do not affect the relative positions and topology of atoms. Any site representation algorithm should be robust to these types of trivial reference frame changes. Any algorithm that directly consumes a binding site as input, faces two challenges. The first
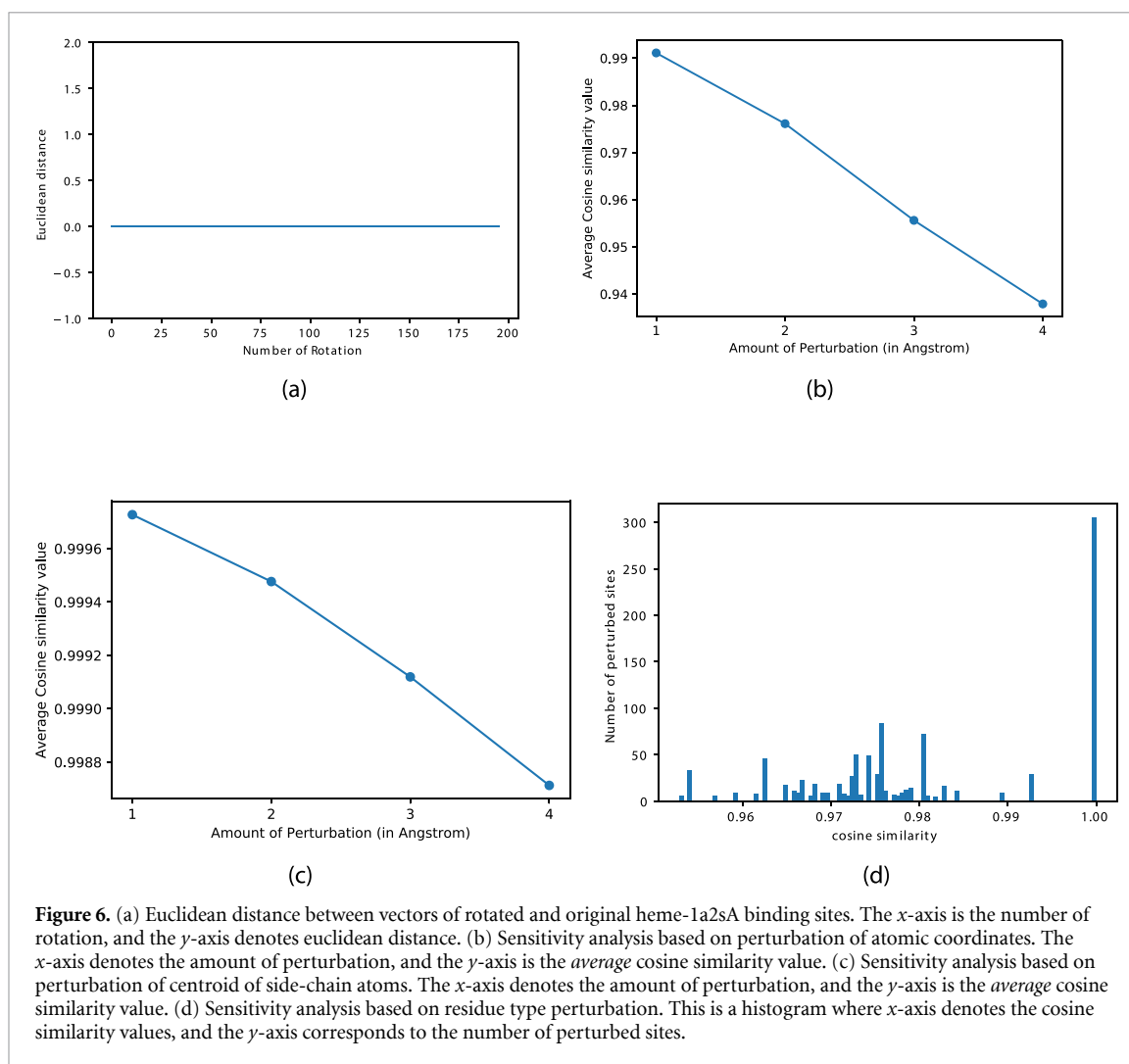
Figure 6. (a) Euclidean distance between vectors of rotated and original heme-1a2sA binding sites. The *x*-axis is the number of rotation, and the *y*-axis denotes euclidean distance. (b) Sensitivity analysis based on perturbation of atomic coordinates. The *x*-axis denotes the amount of perturbation, and the *y*-axis is the *average* cosine similarity value. (c) Sensitivity analysis based on perturbation of centroid of side-chain atoms. The *x*-axis denotes the amount of perturbation, and the *y*-axis is the *average* cosine similarity value. (d) Sensitivity analysis based on residue type perturbation. This is a histogram where *x*-axis denotes the cosine similarity values, and the *y*-axis corresponds to the number of perturbed sites.

challenge is the values of coordinates based on the reference frame. The second challenge is input order of atoms.

A deep learning-based method, DeepDrug3D [30], uses PCA [54] to first determine three axes and then re-orient and re-position points about the axes and their center. This step is performed as a pre-processing step to *normalize* the input.

Methods based on pairwise distances between atoms do not get affected by reference frame changes in the input. Fuzcav [15] is one method that enumerates triplets of C_α atoms and computes the histogram of number of such triplets resulting in a 4833-dimensional vector. This method is robust to reference frame variations. The method by Nakumara and co-workers [16] improves on top of Fuzcav by dimensionality reduction using PCA.

The Site2Vec algorithm is also a pairwise distance-based method there by inherently robust to reference frame changes in the input. We have considered this as a first and sanity test to compute differences between vectors generated for various rotations of a given site. As expected, the vectors are identical irrespective of changes in reference frame for a given site. We chose the heme-1a2sA binding site and generated 250 random rotations of the same. Euclidean distances between vectors of the original site before rotation and those of rotated versions are all zero as desired (figure 6(a)). This experiment demonstrates reference frame invariance of Site2Vec as required.

4.3.2. Sensitivity analysis based on perturbation of coordinate of atoms of the binding site

Any vectorization method should be robust to minor perturbations in coordinates of atoms of a binding site. In any crystal structure, it is not uncommon to see minor changes in side-chain conformations of the same types of sites across different proteins. In this regard, we have carried out a sensitivity analysis of Site2Vec with respect to perturbations to atomic coordinates of the site.

Natural sites are not expected to be perturbed much due to constraints on bond lengths and atomic forces. This poses a challenge in obtaining a natural dataset of sites with atomic perturbations. However,

synthetically introducing atomic perturbations introduces violations of bond lengths and atomic radii constraints. Using molecular dynamics solvers to generate perturbations is also a costly process when whole protein needs to be considered for structural constraints. In order to overcome these challenges and introduce a practically feasible way of studying robustness of vectorization to atomic perturbations, we have considered C_α, C_β and γ coordinates directly (Algorithm 3). Vectors are computed for each of the perturbed versions of a given binding site and are compared against the vector of the unperturbed site. It is expected that the cosine similarities are all high between the vectors and only decrease slowly with respect to the amount of perturbation.

We have considered a Heam binding site (1a00-HEM) as an example to assess sensitivity to perturbations from the PLIC dataset. As many as 1000 random perturbations are generated using (Algorithm 3) between 1 Å to 4 Å uniformly in steps of 1 Å. It has been observed that most of the perturbed sites demonstrated very high cosine similarity to the unperturbed sites figure 6(b).

It is also interesting to see how the side chain orientations alone would affect the similarity of vectors between unperturbed and perturbed variations. However, molecular dynamics simulation of the entire protein to generate valid side-chain orientations is a costly process and also does not generate many variations. We have devised an equivalent experiment to fix C_α and C_β and varied only C_γ to roughly capture movements of side-chain atoms. Given that Site2Vec considers centroid of the side chain atoms, much of the variations in atomic coordinates would be averaged out and the movement of centroid is only expected to be very small. We have carried out perturbation studies of C_γ coordinates by generating as many as 1000 sites for the Heam binding sites (1a00-HEM). The results of the comparison of the unperturbed vector to the perturbed variations resulted in very high cosine similarity values as desired (figure 6(c)). While only C_γ points are considered, the N will be $\approx N/3$ in (Algorithm 3). This experiment demonstrates that Site2Vec is tolerant to positional perturbations of site atoms under natural dynamic conditions of a living cell by capturing similarities between sites even after mild conformational changes.

Algorithm 3 Algorithm for perturbation of coordinates

1: Given a net amount of perturbation ρ
2: Let N be total number of points in a site (including C_α, C_β and C_γ)
3: $\rho^2 = \frac{\sum_{i=1}^{i=N} |s_i - s_i'|^2}{N}$ (where s_i and s_i' are i^{th} coordinates of original and perturbed site respectively)
4: $|s_i - s_i'| = \sqrt{\rho^2/3}$
5: Let $p = s_i$ and $q = s_i'$, then, $q_x \approx p_x \pm \rho/\sqrt{3}$ where $(p_x, q_x$ denotes X coordinates)
6: Similarly for other coordinates as well

4.3.3. Sensitivity analysis based on perturbation of residue types of the binding site
This is the third type of perturbation study where the robustness of site vectorization is studied with respect to minor changes in the chemical composition of a binding site. To this extent, we have considered single residue type perturbation as a way of inducing chemical perturbation. First, we have fixed the coordinates of atoms and randomly picked a residue in a binding site and changed its type to generate a perturbed site. From the perturbed site vector descriptor is freshly computed based on Site2Vec algorithm. Cosine similarity is computed between the vectors of the unperturbed site and each of the perturbed sites. For each residue, one of the 20 amino acids is chosen as its new type. The base work for Site2Vec algorithm uses [19] to fundamentally group residues into five groups. To arrive at a ballpark number on how many perturbations would result in a distinct chemical type, we can assume each group to represent ≈ 4 amino acids. A random type perturbation falls in one of these five groups and in a number of 1000 perturbations, we can expect $\approx$ 200 distinct sites. We have observed that most of the type perturbed sites possessed high cosine similarity to the original site as desired (figure 6(d)).

4.4. Site2Vec as an online Web Service
The Site2Vec algorithm is provisioned as both a stand-alone version and as a web service at (http://services.iittp.ac.in/bioinfo/home). The stand-alone version can be downloaded and locally installed to run in a python environment. The web services accepts one or more PDB structures as inputs and computes binding sites for all the groups of HETATM records corresponding to ligands in the file. Alternatively the user may also specify a PDB id which is internally downloaded or reused on the server side. The service maintains a session for which a user waits and downloads the site vectors. The web service provides for search option to retrieve similar binding site for a given site, more details can be seen from the manual in the web portal. Backend of web ported is developed by Python3 and numpy, Keras, Tensorflow packages are used and Python Flask is used as a web framework in the backend, and the front end uses bootstrap.js libraries to facilitate the interactive interface.

5. Conclusion

In this work, we have presented Site2Vec, a machine learning-based, reference frame invariant, ligand-independent method for vector embedding of protein–ligand binding sites. This method encodes a protein–ligand binding site into d-dimensional vector. The vector representation of the site captures both the chemical and structural properties of binding sites. The quality of vector representations of protein–ligand binding sites is measured in terms of robustness to physical and chemical perturbations of a site and reference frame changes. The vector embedding mechanism serves as a locality sensitive hashing function where proximal vectors have similar meaning. We have demonstrated the ability of the vector embeddings to capture site similarity as proximity in their vector dimensional space on the PLIC data set. The algorithm has been rigorously benchmarked against on 23 methods on 10 different data sets on which Site2Vec stood the top performer in terms of average AUC quality scores. Site2Vec is a highly configurable algorithm on how it generates vector embeddings and we have evaluated configurations on TOUGH-C1 data set and benchmarked for determining best working default parameters. The algorithm is evaluated on TOUGH-C1 nucleotide and heme sites, ProSPECCTs and ApocS3 and it performed among the top in terms of accuracy and AUC scores. The Site2Vec algorithm also out performed another deep learning alternative today [30] in terms of speed by about 3000 times and out performed or matched its performance on reported data sets. The Site2Vec algorithm is an unsupervised methodology which does not require a pair of sites as it is an computes encoding for a given site *in-silos*. Such an approach is desirable today for *site arithmetic* where proximity of sites in vector space and relative positioning is novel, with immense applications in structual bioinformatics and drug discovery. We report here one of the very first deep learning-based vector embedding methods, Site2Vec, in the state of the art backed by thorough benchmarking exercises and quality assessments.

5. Acknowledgement

ORCID iDs

Arnab Bhadra ⓘ https://orcid.org/0000-0002-4059-6414
Kalidas Yeturu ⓘ https://orcid.org/0000-0002-9237-5453

References

[1] Xie Li, Jerry Li, Xie L and Bourne P E 2009 Drug discovery using chemical systems biology: identification of the protein–ligand binding network to explain the side effects of cetp inhibitors *PLoS Comput. Biol.* **5** e1000387
[2] Goh G B, Hodas N O and Vishnu A 2017 Deep learning for computational chemistry *J. Comput. Chem.* **38** 1291–307
[3] Kuhlman B and Bradley P 2019 Advances in protein structure prediction and design *Nat. Rev. Mol. Cell Biol.* **20** 681–97
[4] Berman H M, Bourne P E, Westbrook J and Zardecki C 2003 The protein data bank *Protein Structure* (Boca Raton, FL: CRC Press) pp 394–410
[5] Krivák R and Hoksza D 2018 P2rank: machine learning based tool for rapid and accurate prediction of ligand binding sites from protein structure *J. Cheminformatics* **10** 39
[6] Hassan Md M, Mogollón D Cñeda, Fuentes O and Sirimulla S 2018 Dlscore: A deep learning model for predicting protein–ligand binding affinities *ChemRxiv*
[7] Chen H, Engkvist O, Wang Y, Olivecrona M and Blaschke T 2018 The rise of deep learning in drug discovery *Drug Discov. Today* **23** 1241–50
[8] Mayr A, Klambauer Gunter, Unterthiner T and Hochreiter S 2016 Deeptox: toxicity prediction using deep learning *Front. Environ. Sci.* **3** 80
[9] Ragoza M, Hochuli J, Idrobo E, Sunseri J and Koes D R 2017 Protein–ligand scoring with convolutional neural networks *J. Chem. Inf. Model.* **57** 942–57
[10] Gao M and Skolnick J 2013 Apoc: large-scale identification of similar protein pockets *Bioinformatics* **29** 597–604
[11] Yeturu K and Chandra N 2011 Pocketalign: a novel algorithm for aligning binding sites in protein structures *J. Chem. Inf. Model.* **51** 1725–36
[12] Konc J and Janežič Dšanka 2010 Probis algorithm for detection of structurally similar protein binding sites by local structural alignment *Bioinformatics* **26** 1160–8
[13] Chartier M and Najmanovich R 2015 Detection of binding site molecular interaction field similarities *J. Chem. Inf. Model.* **55** 1600–15
[14] Zhang Y and Skolnick J 2005 Tm-align: a protein structure alignment algorithm based on the tm-score *Nucleic Acids Res.* **33** 2302–9
[15] Weill Nel and Rognan D 2010 Alignment-free ultra-high-throughput comparison of druggable protein- ligand binding sites *J. Chem. Inf. Model.* **50** 123–35
[16] Nakamura T and Tomii K 2016 Protein ligand-binding site comparison by a reduced vector representation derived from multidimensional scaling of generalized description of binding sites *Methods* **93** 35–40

[17] Krotzky T, Grunwald C, Egerland U and Klebe G 2015 Large-scale mining for similar protein binding pockets: with rapmad retrieval on the fly becomes real *J. Chem. Inf. Model.* **55** 165–79

[18] Wood D J, de Vlieg J, Wagener M and Ritschel T 2012 Pharmacophore fingerprint-based approach to binding site subpocket similarity and its application to bioisostere replacement *J. Chem. Inf. Model.* **52** 2031–43

[19] Yeturu K and Chandra N 2008 Pocketmatch: a new algorithm to compare binding sites in protein structures *BMC Bioinform.* **9** 543

[20] Desaphy J, Raimbaud E, Ducrot P and Rognan D 2013 Encoding protein–ligand interaction patterns in fingerprints and graphs *J. Chem. Inf. Model.* **53** 623–37

[21] Batista J, Hawkins P C D, Tolbert R and Geballe M T 2014 Sitehopper-a unique tool for binding site comparison *J. Cheminformatics* **6** 1–1

[22] Shulman-Peleg A, Nussinov R and Wolfson H J 2005 Siteengines: recognition and comparison of binding sites and protein–protein interfaces *Nucleic Acids Res.* **33** W337–W341

[23] Schalon C, Surgand J-Sebastien, Kellenberger E and Rognan D 2008 A simple and fuzzy method to align and compare druggable ligand-binding sites *Proteins* **71** 1755–78

[24] Xie L and Bourne P E 2008 Detecting evolutionary relationships across existing fold space, using sequence order-independent profile–profile alignments *Proc. Natl Acad. Sci.* **105** 5441–6

[25] Brylinski M 2014 e matchsite: Sequence order-independent structure alignments of ligand binding pockets in protein models *PLoS Comput. Biol.* **10** e1003829

[26] Caprari S, Toti D, Hung L V, Stefano M Di and Polticelli F 2014 Assist: a fast versatile local structural comparison tool *Bioinformatics* **30** 1022–4

[27] Desaphy Jérémy, Azdimousa K, Kellenberger E and Rognan D 2012 Comparison and druggability prediction of protein–ligand binding sites from pharmacophore-annotated cavity shapes *J. Chem. Inf. Model.* **52** 2287–99

[28] Morris R J, Najmanovich R J, Kahraman A and Thornton J M 2005 Real spherical harmonic expansion coefficients as 3d shape descriptors for protein binding pocket and ligand comparisons *Bioinformatics* **21** 2347–55

[29] Jiménez Je, Doerr S, Martínez-Rosell G, Rose A S and De Fabritiis G 2017 Deepsite: protein-binding site predictor using 3d-convolutional neural networks *Bioinformatics* **33** 3036–42

[30] Limeng P, Govindaraj R G, Lemoine J M, Hsiao-Chun W and Brylinski M 2019 Deepdrug3d: Classification of ligand-binding pockets in proteins with a convolutional neural network *PLoS Comput. Biol.* **15** e1006718 (https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1006718)

[31] Gold N D and Jackson R M 2006 Sitesbase: a database for structure-based protein–ligand binding site comparisons *Nucleic Acids Res.* **34** D231–D234

[32] Welborn V V and Head-Gordon T 2019 Computational design of synthetic enzymes *Chem. Rev.* **119** 6613–30

[33] Raman K, Yeturu K and Chandra N 2008 targetTB: a target identification pipeline for mycobacterium tuberculosis through an interactome, reactome and genome-scale structural analysis *BMC Syst. Biol.* **2** 109+

[34] Ehrt C, Brinkjost T and Koch O 2018 A benchmark driven guide to binding site comparison: An exhaustive evaluation using tailor-made data sets (prospeccts) *PLoS Comput. Biol.* **14** e1006483

[35] Beis J S and Lowe D G 1997 Shape indexing using approximate nearest-neighbour search in high-dimensional spaces *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition* IEEE pp 1000–6

[36] Guo Y, Liu Y, Oerlemans A, Lao S, Wu S and Lew M S 2016 Deep learning for visual understanding: A review *Neurocomputing* **187** 27–48

[37] Jain A K and Dubes R C 1988 *Algorithms for Clustering Data* (Englewood Cliffs, NJ: Prentice-Hall, Inc.) (https://dl.acm.org/doi/book/10.5555/46712)

[38] Bell C E and Lewis M 2000 A closer view of the conformation of the lac repressor bound to operator *Nat. Struct. Mol. Biol.* **7** 209

[39] Murata T, Fushinobu S, Nakajima M, Asami O, Sassa T, Wakagi T and Yamaguchi I 2002 Crystal structure of the liganded anti-gibberellin a4 antibody 4-b8 (8)/e9 fab fragment *Biochem. Biophys. Res. Commun.* **293** 489–96

[40] Sehnal D, Rose A S, Koča J, Burley S K and Velankar S 2018 Mol*: towards a common library and tools for web molecular graphics *Proc. Workshop on Molecular Graphics and Visual Analysis of Molecular Data* Eurographics Association pp 29–33

[41] Berman H M *et al et al* 2002 The protein data bank *Acta Crystallogr.* D **58** 899–907

[42] Anand P, Nagarajan D, Mukherjee S and Chandra N 2014 PLIC: protein–ligand interaction clusters *Database* (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3998096/)

[43] Kahraman A, Morris R J, Laskowski R A, Favia A D and Thornton J M 2010 On the diversity of physicochemical environments experienced by identical ligands in binding pockets of unrelated proteins *Proteins* **78** 1120–36

[44] Barelier S, Sterling T, O'Meara M J and Shoichet B K 2015 The recognition of identical ligands by unrelated proteins *ACS Chem. Biol.* **10** 2772–84

[45] Brylinski M 2018 Tough-C1 (https://osf.io/enz69/)

[46] Yeturu Kalidas 2020 Chapter 3 - Machine learning algorithms, applications, and practices in data science *Hand Book of Statistics: Principles and Methods for Data Science* eds A S R Srinivasa Rao and C R Rao (Amsterdam: Elsevier) (https://doi.org/10.1016/bs.host.2020.01.002)

[47] Bradley A P 1997 The use of the area under the ROC curve in the evaluation of machine learning algorithms *Pattern Recognit.* **30** 1145–59

[48] Brieman L 2001 Random forests *Machine Learning* **45** 5–32

[49] Xie L, Xie Li and Bourne P E 2009 A unified statistical model to support local sequence order independent similarity searching for ligand-binding sites and its application to genome-based drug discovery *Bioinformatics* **25** i305–i312

[50] Schmitt S, Kuhn D and Klebe G 2002 A new method to detect related function among proteins independent of sequence and fold homology *J. Mol. Biol.* **323** 387–406

[51] S B Kotsiantis, I Zaharakis and P Pintelas 2007 Supervised machine learning: A review of classification techniques *Emerging Artificial Intelligence Applications Computer Engineering* **160** 3–24

[52] van der Maaten L and Hinton G 2008 Visualizing data using t-sne *J. Mach. Learn. Res.* **9** 2579–605 (https://www.jmlr.org/papers/volume9/vandermaaten08a/vandermaaten08a.pdf)

[53] Krotzky T, Grunwald C, Egerland U and Klebe G 2014 Large-scale mining for similar protein binding pockets: with rapmad retrieval on the fly becomes real *J. Chem. Inf. Model.* **55** 165–79

[54] Abdi He and Williams L J 2010 Principal component analysis *Wiley Interdiscip. Rev.: Comput. Stat.* **2** 433–59