Sequence-Based Nanobody-Antigen Binding Prediction

Usama Sardar[1]*, Sarwan Ali[2]*, Muhammad Sohaib Ayub[1], Muhammad Shoaib[1], Khurram Bashir[1], Imdad Ullah Khan[1], and Murray Patterson[2]

[1] Lahore University of Management Sciences, Lahore, Pakistan
usamasardar2022@gmail.com,
{15030039,mshoaib,khurram.bashir,imdad.khan}@lums.edu.pk
[2] Georgia State University, Atlanta GA, USA
{sali85,mpatterson30}@gsu.edu
* Equal Contribution

Abstract. Nanobodies (Nb) are monomeric heavy-chain fragments derived from heavy-chain only antibodies naturally found in Camelids and Sharks. Their considerably small size (∼3-4 nm; 13 kDa) and favorable biophysical properties make them attractive targets for recombinant production. Furthermore, their unique ability to bind selectively to specific antigens, such as toxins, chemicals, bacteria, and viruses, makes them powerful tools in cell biology, structural biology, medical diagnostics, and future therapeutic agents in treating cancer and other serious illnesses. However, a critical challenge in nanobodies production is the unavailability of nanobodies for a majority of antigens. Although some computational methods have been proposed to screen potential nanobodies for given target antigens, their practical application is highly restricted due to their reliance on 3D structures. Moreover, predicting nanobody-antigen interactions (binding) is a time-consuming and labor-intensive task. This study aims to develop a machine-learning method to predict Nanobody-Antigen binding solely based on the sequence data. We curated a comprehensive dataset of Nanobody-Antigen binding and nonbinding data and devised an embedding method based on gapped k-mers to predict binding based only on sequences of nanobody and antigen. Our approach achieves up to 90% accuracy in binding prediction and is significantly more efficient compared to the widely-used computational docking technique.

1 Introduction

Nanobodies (Nbs) are single-domain antibodies (sdAb), derived from heavy-chain only antibodies naturally occurring in Camelids and Sharks. They represent a unique class of proteins/antibodies having a molecular weight of 12–15 kDa, that combine the advantageous characteristics of conventional antibodies with desirable attributes of small-molecule drugs. Nbs are remarkably adaptable to various applications and offer several advantages over conventional antibodies [6]. Every Nb contains the distinct structural and functional properties found in naturally-occurring heavy-chain antibodies. They have a naturally low potential for causing immune responses and exhibit high similarity to variable region of heavy chain (VH) in human antibodies, making them excellently suited for therapeutic and diagnostic applications. Due to their small size, unique structure and high stability, Nbs can access targets that are beyond the reach of conventional antibodies and small-molecule drugs [24,7]. Nbs Structure prediction and modeling are still challenging tasks [5,30]. Hundreds of Nb crystallographic structures have been deposited in the Protein Data Bank (PDB) [3,4]. Despite this, the current representation falls short of capturing the vast structural and sequence diversity observed in Nb hypervariable loops. Moreover, Nbs display a greater range of conformational variations, lengths, and sequence variability in their CDR3 compared to antibodies [18]. This makes modeling and prediction of their 3D structure more complex.

Machine Learning (ML) plays a crucial role in predicting nanobody-antigen(Nb-Ag) interactions. ML offers a powerful and effective approach to analyzing and comprehending complex patterns within extensive datasets [8]. Traditional non-computational methods for determining Nb-Ag interactions can be both costly and time-consuming. ML provides a faster and more cost-effective alternative, enabling scientists to prioritize potential nanobodies candidates for further research [29]. The large amount of training data that is easily accessible when utilizing sequence-based ML techniques is advantageous. Even though there is an increasing amount of data on protein structures, most Nb-Ag sequences still lack validated structural details, even though the number of protein sequence entries is still rising quickly [27,12].

ML algorithms are capable of handling a vast amount of data, encompassing nanobodies and antigen sequences, structural information, and experimental binding data. Through the examination of this data, ML algorithms can find complex relationships and patterns that might be hard for people to see on their own. These methods can automatically extract relevant details from raw data, such as structural information or amino acid sequences. This feature extraction method makes it easier to spot important molecular traits that influence antibody and Nb-Ag interaction [21,31]. ML models can gain knowledge from large databases of training samples and produce precise predictions. By being trained on known Nb-Ag binding data, these models may understand the underlying principles and patterns regulating binding interactions. This allows them to make accurate predictions for previously undiscovered Nb-Ag combinations. Traditional experimental techniques, on the other hand, need a lot of time and money to determine Nb-Ag binding. To arrange possible antibody candidates for further investigation, ML offers a quicker and more economical alternative [23,32,31,17].

Nanobody-antigen binding play a significant role in the immune response. By predicting and studying these bindings, researchers can gain insights into how nanobodies recognize and neutralize specific antigens. This understanding is fundamental for elucidating immune mechanisms and developing strategies for diagnostics to combat infectious diseases, autoimmune diseases, and cancer.

Predicting Nb-ag binding can aid in discovering and engineering nanobodies with desired properties. Binding helps identify the most essential antigens for vaccine development, effective vaccine formulations, and understanding the mechanisms of immune protection.

Predicting binding interactions enables the selection of highly specific nanobodies and antigens for accurate and sensitive diagnostic tests. Binding can guide the rational design of therapeutic nanobodies or nanobodies-based drugs and diagnostic tests. Scientists can modify or engineer nanobodies to improve their affinity, selectivity, and therapeutic potential by understanding the binding interactions between nanobodies and their target antigens. This approach can be applied in areas such as diagnostic tests and cancer immunotherapy, where nanobodies are designed to target specific tumor antigens [28].

The input in our study is nanobody sequences and antigen sequences and the output is binding/docking score (Yes/No).In this paper, we trained ML on nanobodies and antigen sequences extracted from the single-domain antibody (sdAb) database to determine binding. We make the following contributions.

- We have performed the comparison of various ML approaches for predicting nanobody-antigen binding from sequences only.
- We evaluated the impact of various sequence features (e.g., isoelectric point, hydrophilicity) on the prediction accuracy of ML models.
- We have curated a dataset of nanobody-antigen pairs for training and testing machine-learning models and made it publicly available for further research.

The rest of the paper is organized as follows: Section 2 explores the existing research and highlights the research gaps. Section 3 explains the data collection, feature extraction, and data visualization. The proposed embedding is discussed in Section 4. Section 5 describes the ML models and evaluation metrics. We discuss our results in Section 6. Finally, we provide the conclusion and future directions in Section 7.

2 Related Work

Antibodies (Abs) are crucial tools in biological research and the biopharmaceutical industry due to their exceptional binding specificity and strong affinity for target antigens. The effectiveness of the immune system directly reflects the diversity of antigens against which specific tightly binding ‘B-lymphocyte antigen receptors (BCRs) can be generated. The vast range of binding specificity is achieved through sequence variations in the heavy chain (VH) and light chain (VL), resulting in an estimated diversity of BCRs in humans [19] that surpasses the population size of B-lymphocytes in an individual. However, it is still unclear how this immense sequence diversity translates into antigen specificity. Although not every unique combination of VH-VL sequences leads to a distinct binding specificity, predicting the number and positions of amino acid mutations required to change binding specificity has proven challenging [22].

A more manageable system is offered by heavy-chain antibodies found in camelid species like camels, llamas, and alpacas, where the light chain is absent, as shown in Figure 1. These antibodies, known as nanobodies (Nbs), consist of an isolated variable VHH domain that is about ten times smaller than conventional Abs but retains comparable binding specificity [20].

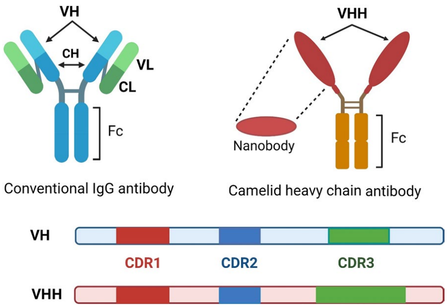


Fig. 1: Nanobody vs Antibody (Figure taken from [11])

Both Nbs and Abs face the fundamental challenge of deciphering the molecular code that links amino acid sequence, particularly the choice of paratope residues, to the binding specificity of the folded molecule. In regular Abs, the paratope is situated at the interface of the VH and VL domains, typically comprising residues from up to six distinct hypervariable loop regions. The VH and VL domains can dock together in various ways, allowing the antibody to maximize the diversity of potential antigen-binding surfaces. In contrast, the Nb paratope is entirely contained within the VHH domain, significantly limiting the range of possible antigen-binding surfaces without seemingly affecting the diversity of resulting binding specificities. Indeed, Nbs typically bind their target antigens with affinities comparable to classical monoclonal Abs [19].

Several studies have been conducted to generate antibodies/nanobodies using non-computational methods. Experimental techniques such as hybridoma technology and phage display [26] are used to generate specific antibodies/nanobodies, but these have limitations and challenges. Hybridoma technology involves immunizing animals with an antigen and fusing B cells from the immunized animal with cancer cells to create hybridoma cells that produce specific antibodies. However, this method raises ethical concerns due to animal cruelty and is time-consuming, labor-intensive, and limited in antibody/nanobody diversity. On the other hand, phage display utilizes bacteriophages to display antibody fragments, but it also has time-consuming rounds of selection and amplification, labor-intensive requirements, and high costs.

Several ML methods are used to predict nanobody-antigen binding. Sequence-based methods utilize amino acid sequences, extracting features such as physiochemical properties, sequence motifs, and sequence profiles [29]. These features are then used as input for machine learning algorithms like support vector machines (SVM), random forests, or neural networks. Structure-based methods employ three-dimensional structures obtained from experimental techniques like X-ray crystallography or homology modeling. Structural features like solvent accessibility, electrostatic potential, or shape complementarity are extracted and fed into machine-learning models. Hybrid methods combine sequence-based and structure-based features, integrating both sequence and structural information to capture a broader range of characteristics. Deep learning methods, such as convolutional neural networks (CNN) and recurrent neural networks (RNN), learn complex patterns and relationships from large datasets, including sequence and structural information, for accurate predictions [5]. Docking-based methods use molecular docking algorithms to predict the binding orientation and affinity by calculating a binding score based on the optimal spatial arrangement of the interacting molecules [23].

3 Proposed Approach

The proposed pipeline comprised different steps including data collection, Nb-Ag sequence analysis, numerical embedding generation, and optimal feature extraction from the Nb and Ag sequences. We discuss each step in detail.

3.1 Data Collection

We collected 47 Ag sequences from UniProt [3], and for each we collected all binding Nbs from Single Domain Antibody Database[4], which are total 365, as shown in Table 1 along with the basic statistics for the length of the antigen sequences including average, minimum, and maximum lengths, etc. Basica summary of the number of nanobodies binding to antigens is given in Table 2

		Sequence Length Statistics				
Type	Count	Mean	Min	Max	Std. Dev.	Median
Antigens	47	671.51	158	1816	421.24	480
Nanobodies	365	122.84	104	175	8.87	123

Table 1: Sequence length statistics for antigen and nanobody sequences.

3.2 Features Extracted From Sequences

We performed basic protein sequence analysis using the 'bioPython' package[5] on each nanobody and antigen sequence to determine their features. These fea-

[3] https://www.uniprot.org/
[4] http://www.sdab-db.ca/
[5] https://biopython.org/docs/dev/api/Bio.SeqUtils.ProtParam.html

Type	Mean	Min	Max	Std. Dev.	Median
Nanobodies in each antigen	7.77	1	36	9.28	4

Table 2: Statistics for nanobody sequences binding to each antigen.

tures include charge at pH, Grand Average of Hydropathy (GRAVY), molecular weight, aromaticity, instability index, isoelectric point, secondary structure fraction (helix, turn, and sheet), and molar extinction coefficient (reduced and oxidized).

Charge at pH: The charge of a protein at a pH is determined by presence of charged amino acids (aspartic acid, glutamic acid, lysine, arginine, histidine, and cysteine) and their ionization state. These amino acids gain or lose protons at different pH values, resulting in a net charge. The charge at pH affects the protein's solubility, interaction with other molecules, and biological function.

Grand Average of Hydropathy (GRAVY): GRAVY [15] measures the overall hydrophobicity or hydrophilicity of a protein, calculated by averaging the hydropathy values of its amino acids. Positive GRAVY values indicate hydrophobic, while negative values represent hydrophilic regions and provide insights into protein stability, membrane interactions, and protein-protein interactions.

Molecular weight: Molecular weight refers to the sum of the atomic weights of all atoms in a protein molecule. It is calculated based on the amino acid composition of the protein sequence. Molecular weight impacts various protein properties, such as protein folding, thermal stability, and mobility, and is crucial for protein identification, characterization, and quantification.

Instability index: The instability index [9] is a measure of the propensity of a protein to undergo degradation or unfold. Higher instability index values indicate increased susceptibility to degradation and decreased protein stability. The index is useful for evaluating protein expression, protein engineering, and predicting potential regions of protein instability.

Isoelectric point: The isoelectric point (pI) is the pH at which a protein has a net charge of zero. It is determined by the presence of charged amino acids and their ionization states. The pI influences protein solubility, crystallization, and electrophoretic mobility. Knowledge of the pI is crucial for protein purification, protein characterization, and protein separation techniques based on charge.

Secondary structure fraction (helix, turn, and sheet): The secondary structure fraction [10,13,14] refers to the proportions or percentages of different secondary structure elements (helices, turns, and sheets) in a protein sequence. These elements are determined by the pattern of hydrogen bonds between amino acids. The secondary structure fraction provides insights into the protein's folding, stability, and functional properties. Different secondary structure fractions contribute to the unique 3D structure and biological function of the protein.

Molar extinction coefficient (reduced and oxidized): The molar extinction coefficient is a measure of the ability of a molecule to absorb light at a specific wavelength. It quantifies the efficiency of light absorption by the molecule. The molar extinction coefficients can be different for reduced and oxidized forms of a protein due to changes in the chromophores. These coefficients are useful for protein quantification, monitoring protein folding/unfolding, and studying protein-protein interactions.

Remark 1. We compute classification results with and without these features for all embedding methods, including the proposed and baseline methods.

3.3 Obtaining Non-Binding Nb-Ag Pairs

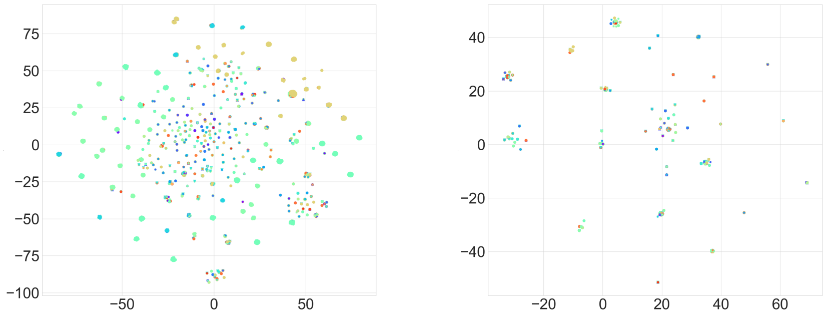
We created the proximity matrix of these sequences using Clustal Omega [6] to evaluate the pairwise edit distance between antigens and nanobodies. This pairwise distance is used to identify further binding pairs and non-binding pairs of antigens and nanobodies. There are three pairs of antigens with very high similarity (distance $< .25$), namely, antigen green fluorescent protein (GFP) and Superfolder green fluorescent protein (sfGFP), (pairwise distance = 0.05042), RAC-gamma serine/threonine-protein kinase pleckstrin homology domain (Akt3PH) and RAC-alpha serine/threonine-protein kinase pleckstrin homology domain (Akt1PH) (pairwise distance = 0.19833), Glioblastoma multiforme dihydropyrimidinase-related protein 2 (DPYSl2) and/or methylenetetrahydrofolate dehydrogenase 1 (MTHFD1) and Glioblastoma multiforme collapsin response mediator protein 1 (CRMP1) (pairwise distance = 0.0.236014). For these pairs, if a nanobody binds with one of them, we assume it also binds to the other. Thus, we add 1388 of additional binding pairs that bind with each other.

The non-binding pairs are obtained as follows: Suppose we have two binding Nb-Ag (n_i, g_j) and (n_ℓ, g_k). Then both the pairs (n_i, g_k) and (n_ℓ, g_j) are candidates for being declared as non-binding pairs if the distance between g_i and g_j is more than a certain threshold (we set the threshold $\in \{.8, .85, .9\}$. A random sample of such candidate pairs is added to the dataset as non-binding pairs, which consist of 1728 such pairs in total.

Data Visualization: In order to visually assess the proximity of similar points in the Spike2Vec-based embeddings, we employ the t-Distributed Stochastic Neighbor Embedding (t-SNE) technique to obtain two-dimensional representations of the embeddings. These representations are then plotted using a scatterplot [16]. The t-SNE plots for the Nanobody and Antigen-based embeddings are depicted in Figure 2. The colored data points show the different antigen

[6] https://www.ebi.ac.uk/Tools/msa/

categories (47 in total). Although the data points are scattered in the whole plot, we can observe small grouping for different labels, which shows that the embedding captures the hidden hierarchical and structural information inherent in the protein sequences.



(a) Nanobody

(b) Antigen

Fig. 2: t-SNE plots for Nanobody and Antigen Embeddings using Spike2Vec approach. The figure is best seen in color.

4 Representation Learning For Nb-Ag Binding Prediction

We learn various machine learning models on the training data of binding and non-binding pairs. To train the classifiers for the binary classification problem, we generate the feature vector (also called embeddings) for the nanobody and antibody in pairs. The corresponding feature vectors are a concatenation of the features extracted from the sequence as outlined above and feature vector embedding of the whole nanobody and antigen sequences, using state-of-the-art sequence2vec models discussed below.

Embedding Generation: To generate the fixed length numerical embedding from the variable length protein sequences, we use the idea of gapped k-mers. The gapped k-mers are a representation of biological sequences, such as DNA or protein sequences, that capture patterns while allowing for flexibility in the positioning of the k-mer elements (i.e., nucleotides or amino acids). In gapped k-mers, there are gaps or missing positions between the elements of the k-mers. These gaps introduce variability and enable the capture of more diverse and complex patterns compared to traditional contiguous k-mers.

The spectrum generated from gapped k-mers refers to the collection of all possible (unique) gapped k-mers (along with their count) that can be formed from a given protein sequence. These k-mers counts are then added into the numerical vector, which is generated based on all possible k-mers for the given alphabet Σ that corresponds to $ACDEFGHIKLMNPQRSTVWXY$-. Note that in Σ, the character '-' is used to include the gap in the k-mers. For example, for a k-mer 'ACD', the gapped k-mers will comprise of '-CD', 'A-D', 'AC-', and 'ACD'. An important point to note here is that we generated 4 k-mers from just 1 original k-mer. This extra information helps us to preserve more information, which eventually helps in the downstream supervised analysis. The pseudocode for generating gapped k-mers-based spectrum is given in Algorithm 1. The spectrum provides a comprehensive representation of the sequence, taking into account both conserved elements and the flexibility introduced by the gaps. Each gapped k-mer in the spectrum represents a distinct pattern or subsequence present in the sequence, capturing variations and relationships between elements that may be important for various biological processes. Using gapped k-mers offers several advantages compared to other typical biological methods:

Algorithm 1 Gapped k-mers Spectrum

```
Input: Set of sequences $\mathcal{S}$, alphabet $\Sigma$, $k$ is size of $k$-mer
Output: Spectrum $V$
combos = GenerateAllCombinations($\Sigma$, k)        ▷ all possible combinations of $k$-mers
$V$ = [0] * $|\Sigma|^k$
for s in $\mathcal{S}$ do
    $Gkmers$ = GappedKmers(s,k)                        ▷ compute all possible gapped $k$-mers
    for $i \leftarrow 1$ to $|Gkmers|$ do
        idx = combos.index($Gkmers$[i])
        $V$[idx] $\leftarrow$ $V$[idx] + 1
return($V$)
```

Increased sensitivity: Gapped k-mers can capture more complex patterns and relationships compared to traditional contiguous k-mers. Gapped k-mers can capture conserved elements that are not necessarily adjacent in the sequence by allowing gaps. This increased sensitivity can be crucial for identifying motifs or regions of interest but may exhibit variable spacing.

Enhanced flexibility: Gapped k-mers offer flexibility in terms of the spacing between elements. This flexibility allows for the inclusion of different variations and insertions, providing a more comprehensive representation of the sequence. Gapped k-mers can accommodate diverse patterns and handle insertions or deletions more effectively than traditional contiguous k-mers.

Comprehensive motif representation: The spectrum generated from gapped k-mers provides a comprehensive representation of the sequence by capturing a wide range of conserved patterns and variations. This allows for a more detailed analysis of complex motifs or functional regions that involve specific arrangements of elements.

Improved specificity: Gapped k-mers can help improve specificity by reducing false-positive matches. By considering both conserved elements and gaps, gapped k-mers can differentiate between true motifs and random matches that may occur by chance in traditional k-mers.

After generating the embeddings using gapped k-mers spectrum, we use those embeddings as input to the machine learning classifiers for binary classification to predict the binding of Nb-Ag pairs binding.

5 Experimental Setup

In this section, we discuss the dataset statistics and evaluation metrics along with baseline models. All experiments are carried out using Python on a system equipped with a 2.4 GHz Core i5 processor, 32 GB of memory, and the Windows 10 operating system. For experiments, we randomly divide the data into $70-30\%$ training and testing set and use 10% of the data from the training set as the validation data. The experiments are repeated 5 times and we show average results to eliminate any biases in the random splits. For the sake of reproducibility, our code and preprocessed datasets can be accessed online [7].

Baseline Models: To evaluate the proposed embedding, we compare their results with several popular baseline models from the literature. The baselines includes Spike2Vec [2], Minimizers [25], and PWM2Vec [1]. A detailed description of the baseline embedding models is given in Section A in the appendix.

Evaluation Metrics: To assess the performance of embeddings, we employ several evaluation metrics, including accuracy, precision, recall, F1 score (weighted), F1 score (macro), Receiver Operating Characteristic (ROC) curve, Area Under the Curve (AUC), and training runtime. For metrics designed for binary classification tasks, we adopt the one-vs-rest approach for multi-class classification.

Machine Learning Classifiers: Supervised analysis entails the utilization of diverse linear and non-linear classifiers, such as Support Vector Machine (SVM), Naive Bayes (NB), Multi-Layer Perceptron (MLP), K-Nearest Neighbors (KNN), Random Forest (RF), Logistic Regression (LR), and Decision Tree (DT).

6 Results And Discussion

The classification results for different evaluation metrics with and without the sequence features are reported in Table 3. We observe (bold values) that the gapped k-mers spectrum outperforms all embeddings in the case of average accuracy, precision, recall, and ROC-AUC using the random forest classifier. For Weighted and Macro F1, the baseline Spike2Vec performs better than other embeddings using the random forest classifier. One interesting observation is that

	Embeddings	Algo.	Acc. ↑	Prec. ↑	Recall ↑	F1 (Weig.) ↑	F1 (Macro) ↑	ROC AUC ↑	Train Time (sec.) ↓
Without Sequence Features	Spike2Vec	SVM	0.818	0.824	0.818	0.818	0.818	0.819	5.662
		NB	0.813	0.815	0.813	0.813	0.813	0.813	0.103
		MLP	0.844	0.846	0.844	0.844	0.844	0.844	4.075
		KNN	0.892	0.893	0.892	0.892	0.892	0.892	1.290
		RF	0.906	0.911	0.906	**0.906**	**0.906**	0.906	3.725
		LR	0.813	0.815	0.813	0.813	0.813	0.814	2.417
		DT	0.878	0.878	0.878	0.878	0.877	0.878	1.293
	Minimizers	SVM	0.824	0.826	0.824	0.823	0.823	0.823	5.444
		NB	0.791	0.792	0.791	0.790	0.790	0.790	0.091
		MLP	0.844	0.845	0.844	0.844	0.844	0.844	2.997
		KNN	0.880	0.880	0.880	0.880	0.880	0.880	1.257
		RF	0.892	0.898	0.892	0.892	0.892	0.893	4.000
		LR	0.811	0.812	0.811	0.811	0.811	0.811	1.343
		DT	0.851	0.851	0.851	0.850	0.850	0.850	1.677
	PWM2Vec	SVM	0.810	0.812	0.810	0.809	0.809	0.809	5.732
		NB	0.792	0.793	0.792	0.792	0.792	0.792	0.095
		MLP	0.820	0.821	0.820	0.820	0.819	0.820	3.730
		KNN	0.875	0.875	0.875	0.875	0.875	0.875	1.232
		RF	0.892	0.899	0.892	0.891	0.891	0.892	3.746
		LR	0.804	0.805	0.804	0.804	0.804	0.804	7.137
		DT	0.866	0.866	0.866	0.866	0.866	0.866	1.692
	Gapped *k*-mers	SVM	0.814	0.816	0.814	0.813	0.813	0.812	5.740
		NB	0.798	0.798	0.798	0.797	0.797	0.796	0.087
		MLP	0.824	0.825	0.824	0.824	0.824	0.824	2.886
		KNN	0.885	0.886	0.885	0.885	0.885	0.885	0.995
		RF	**0.907**	**0.912**	**0.907**	0.894	0.894	**0.908**	3.755
		LR	0.812	0.813	0.812	0.812	0.812	0.812	4.395
		DT	0.872	0.872	0.872	0.872	0.871	0.872	1.777
With Sequence Features	Spike2Vec	SVM	0.791	0.796	0.791	0.790	0.790	0.790	8.804
		NB	0.695	0.737	0.695	0.680	0.678	0.691	0.085
		MLP	0.811	0.814	0.811	0.811	0.811	0.811	2.326
		KNN	0.844	0.845	0.844	0.844	0.844	0.844	0.953
		RF	0.897	0.903	0.897	0.896	0.896	0.898	3.890
		LR	0.827	0.827	0.827	0.827	0.826	0.827	1.183
		DT	0.847	0.848	0.847	0.847	0.847	0.847	1.246
	Minimizers	SVM	0.778	0.783	0.778	0.777	0.777	0.777	10.938
		NB	0.674	0.736	0.674	0.649	0.647	0.670	0.094
		MLP	0.801	0.806	0.801	0.800	0.800	0.800	3.228
		KNN	0.842	0.842	0.842	0.842	0.842	0.842	0.827
		RF	0.896	0.902	0.896	0.896	0.896	0.897	3.801
		LR	0.823	0.823	0.823	0.823	0.823	0.823	1.167
		DT	0.846	0.846	0.846	0.845	0.845	0.845	1.297
	PWM2Vec	SVM	0.766	0.770	0.766	0.765	0.765	0.766	9.569
		NB	0.679	0.726	0.679	0.659	0.657	0.674	0.087
		MLP	0.811	0.813	0.811	0.811	0.811	0.811	2.889
		KNN	0.828	0.828	0.828	0.827	0.827	0.827	0.768
		RF	0.893	0.901	0.893	0.892	0.892	0.894	3.765
		LR	0.819	0.819	0.819	0.819	0.819	0.819	1.495
		DT	0.851	0.851	0.851	0.851	0.851	0.850	1.279
	Gapped *k*-mers	SVM	0.785	0.792	0.785	0.784	0.783	0.784	9.270
		NB	0.720	0.745	0.720	0.712	0.711	0.718	0.086
		MLP	0.807	0.810	0.807	0.806	0.806	0.806	2.432
		KNN	0.839	0.839	0.839	0.839	0.838	0.838	0.753
		RF	0.895	0.901	0.895	0.894	0.894	0.895	3.468
		LR	0.823	0.823	0.823	0.823	0.823	0.823	1.123
		DT	0.860	0.861	0.860	0.860	0.860	0.860	0.955

Table 3: Classification results (averaged over 5 runs) for different evaluation metrics. The best values for each embedding are underlined while overall best values among all embeddings for different evaluation metrics are shown in bold.

the random forest classifier consistently outperforms other classifiers for all embeddings and evaluation metrics, as shown with underlined values in Table 3.

[7] https://github.com/sarwanpasha/Nanobody_Antigen

Comparing embeddings with and without sequence features, we observe that using these features degrades classifiers' accuracy. This degradation is due to redundancy and feature dimensionality as some of the features might capture similar information as the k-mers spectrum. For example, the charge at pH, aromaticity, or GRAVY may already encode certain aspects of protein sequence patterns. Including redundant features can lead to multicollinearity, where the features are highly correlated, making it difficult for the classifier to distinguish their contributions. In the case of feature dimensionality, the addition of new features increases the dimensionality of the input space. With a higher number of features, the classifier faces the curse of dimensionality. Insufficient training data or a limited number of samples in each class relative to the feature space can result in overfitting, reduced generalization performance, and decreased accuracy.

6.1 Statistical Significance

Considering the random splitting of the data into training and testing sets during the computation of the results, the statistical significance of the obtained results becomes a significant concern. To address this, we conducted a student t-test to evaluate the significance of the results. The p-values were determined using the averages (as shown in the results table) and standard deviations (SD) derived from five different runs. It is worth noting that the SD values obtained from these five runs, which involved different random splits, were consistently small. In the majority of cases, they were less than 0.002. Consequently, we observed that the p-values were predominantly less than 0.05 due to the low SD values. This confirms the statistical significance of the obtained results. Due to page limitation, we did not include the SD values and p-values. However, it is important to highlight that the small SD values and the majority of p-values being less than 0.05 further support the statistical significance of our results.

7 Conclusion

This study aimed to develop an ML approach to predict Nb-Ag binding solely based on sequences, thereby reducing the need for computationally intensive techniques such as docking. The proposed method utilized an embedding approach using gapped k-mers to generate a spectrum, which was then used for supervised analysis. Experimental evaluation of our approach demonstrates that the gapped k-mers spectrum outperformed competing embeddings. Our approach offers a more efficient and cost-effective alternative for screening potential Nbs and holds promise for facilitating the development of Nb-based diagnostics and therapeutics for various diseases, including cancer and other serious illnesses. Future research involves evaluating the proposed model on bigger datasets and also working on the generalizability and robustness of the model. Additionally, exploring the integration of additional features and considering other machine learning algorithms could further enhance predictive performance.

References

1. Ali, S., Bello, B., Chourasia, P., Punathil, R.T., Zhou, Y., Patterson, M.: PWM2Vec: An efficient embedding approach for viral host specification from coronavirus spike sequences. Biology **11**(3), 418 (2022)
2. Ali, S., Patterson, M.: Spike2vec: An efficient and scalable embedding approach for covid-19 spike sequences. In: IEEE International Conference on Big Data (Big Data). pp. 1533–1540 (2021)
3. Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N., Bourne, P.E.: The protein data bank. Nucleic acids research **28**(1), 235–242 (2000)
4. Burley, S.K., Berman, H.M., Bhikadiya, C., Bi, C., Chen, L., Di Costanzo, L., Christie, C., Dalenberg, K., Duarte, J.M., Dutta, S., et al.: Rcsb protein data bank: biological macromolecular structures enabling research and education in fundamental biology, biomedicine, biotechnology and energy. Nucleic acids research **47**(D1), D464–D474 (2019)
5. Cohen, T., Halfon, M., Schneidman-Duhovny, D.: Nanonet: Rapid and accurate end-to-end nanobody modeling by deep learning. Frontiers in immunology **13**, 958584 (2022)
6. Cortez-Retamozo, V., Backmann, N., Senter, P.D., Wernery, U., De Baetselier, P., Muyldermans, S., Revets, H.: Efficient cancer therapy with a nanobody-based conjugate. Cancer research **64**(8), 2853–2857 (2004)
7. Deffar, K., Shi, H., Li, L., Wang, X., Zhu, X.: Nanobodies-the new concept in antibody engineering. African Journal of Biotechnology **8**(12) (2009)
8. Farhan, M., Tariq, J., Zaman, A., Shabbir, M., Khan, I.: Efficient approximation algorithms for strings kernel based sequence classification. In: Advances in neural information processing systems (NeurIPS). pp. 6935–6945 (2017)
9. Guruprasad, K., Reddy, B.B., Pandit, M.W.: Correlation between stability of a protein and its dipeptide composition: a novel approach for predicting in vivo stability of a protein from its primary sequence. Protein Engineering, Design and Selection **4**(2), 155–161 (1990)
10. Haimov, B., Srebnik, S.: A closer look into the α-helix basin. Scientific reports **6**(1), 38341 (2016)
11. Hosseindokht, M., Bakherad, H., Zare, H.: Nanobodies: A tool to open new horizons in diagnosis and treatment of prostate cancer. Cancer Cell International **21**(1), 1–9 (2021)
12. Hou, Q., Stringer, B., Waury, K., Capel, H., Haydarlou, R., Xue, F., Abeln, S., Heringa, J., Feenstra, K.A.: Serendip-ce: sequence-based interface prediction for conformational epitopes. Bioinformatics **37**(20), 3421–3427 (2021)
13. Hutchinson, E.G., Thornton, J.M.: A revised set of potentials for β-turn formation in proteins. Protein Science **3**(12), 2207–2216 (1994)
14. Kim, C.A., Berg, J.M.: Thermodynamic β-sheet propensities measured using a zinc-finger host peptide. Nature **362**(6417), 267–270 (1993)
15. Kyte, J., Doolittle, R.F.: A simple method for displaying the hydropathic character of a protein. Journal of molecular biology **157**(1), 105–132 (1982)
16. Van der M., L., Hinton, G.: Visualizing data using t-SNE. Journal of Machine Learning Research (JMLR) **9**(11) (2008)
17. Miller, N.L., Clark, T., Raman, R., Sasisekharan, R.: Learned features of antibody-antigen binding affinity. Frontiers in Molecular Biosciences **10**, 1112738 (2023)

18. Mitchell, L.S., Colwell, L.J.: Analysis of nanobody paratopes reveals greater diversity than classical antibodies. Protein Engineering, Design and Selection **31**(7-8), 267–275 (2018)
19. Mitchell, L.S., Colwell, L.J.: Comparative analysis of nanobody sequence and structure data. Proteins: Structure, Function, and Bioinformatics **86**(7), 697–706 (2018)
20. Muyldermans, S.: Nanobodies: natural single-domain antibodies. Annual review of biochemistry **82**, 775–797 (2013)
21. Myung, Y., Pires, D.E., Ascher, D.B.: Csm-ab: Graph-based antibody–antigen binding affinity prediction and docking scoring function. Bioinformatics **38**(4), 1141–1143 (2022)
22. Peng, H.P., Lee, K.H., Jian, J.W., Yang, A.S.: Origins of specificity and affinity in antibody–protein interactions. Proceedings of the National Academy of Sciences **111**(26), E2656–E2665 (2014)
23. Ramon, A., Saturnino, A., Didi, K., Greenig, M., Sormanni, P.: Abnativ: Vq-vae-based assessment of antibody and nanobody nativeness for engineering, selection, and computational design. bioRxiv pp. 2023–04 (2023)
24. Revets, H., De Baetselier, P., Muyldermans, S.: Nanobodies as novel agents for cancer therapy. Expert opinion on biological therapy **5**(1), 111–124 (2005)
25. Roberts, M., Hayes, W., Hunt, B.R., Mount, S.M., Yorke, J.A.: Reducing storage requirements for biological sequence comparison. Bioinformatics **20**(18), 3363–3369 (2004)
26. Rossant, C.J., Carroll, D., Huang, L., Elvin, J., Neal, F., Walker, E., Benschop, J.J., Kim, E.E., Barry, S.T., Vaughan, T.J.: Phage display and hybridoma generation of antibodies to human cxcr2 yields antibodies with distinct mechanisms and epitopes. MAbs **6**(6), 1425–1438 (2014)
27. Schwede, T.: Protein modeling: what happened to the "protein structure gap"? Structure **21**(9), 1531–1540 (2013)
28. Sormanni, P., Aprile, F.A., Vendruscolo, M.: Rational design of antibodies targeting specific epitopes within intrinsically disordered proteins. Proceedings of the National Academy of Sciences **112**(32), 9902–9907 (2015)
29. Tam, C., Kumar, A., Zhang, K.Y.: Nbx: machine learning-guided re-ranking of nanobody–antigen binding poses. Pharmaceuticals **14**(10), 968 (2021)
30. Valdés-Tresanco, M.S., Valdés-Tresanco, M.E., Jiménez-Gutiérrez, D.E., Moreno, E.: Structural modeling of nanobodies: A benchmark of state-of-the-art artificial intelligence programs. Molecules **28**(10), 3991 (2023)
31. Yang, Y.X., Huang, J.Y., Wang, P., Zhu, B.T.: Area-affinity: A web server for machine learning-based prediction of protein–protein and antibody–protein antigen binding affinities. Journal of Chemical Information and Modeling (2023)
32. Ye, C., Hu, W., Gaeta, B.: Prediction of antibody-antigen binding via machine learning: Development of data sets and evaluation of methods. JMIR Bioinformatics and Biotechnology **3**(1), e29404 (2022)

Appendix

A Baseline Models

***Spike2Vec* [2]:** The objective of this approach is to generate numerical embeddings for input protein sequences, facilitating the utilization of machine learning models. To achieve this, the method begins by creating k-mers from the given spike sequence. k-mers are employed due to their ability to retain the sequential information of the protein sequence.

Definition 1 (k-mers). *K-mers refer to sets of consecutive amino acids (called mers) of length k in a given sequence. The consecutive k-mers are computed using the sliding window approach, where the next k-mer is 1 window to the right of the previous k-mer. In the domain of natural language processing (NLP), they are referred to as n-grams.*

All possible k-mers that can be generated for a given sequence of length N are $N - k + 1$. Spike2Vec calculates the frequency vector based on k-mers to convert the alphabetical information of k-mers into a numerical representation. This vector captures the occurrence counts of each k-mer in the sequence, also called k-mers spectrum.

***Minimizers* [25]:** The Minimizer-based feature vector is a method that involves computing a *minimizer* of length m (also called m-mer) for a given k-mer. In the case of m-mer, we have $m < k$. The m-mer is determined as the lexicographically smallest sequence in both the forward and reverse order of the k-mer. A fixed-length frequency vector is constructed from the set of minimizers, where each bin in the frequency vector represents the count of a specific minimizer. This method is also referred to as m-mers spectrum. The length of each vector is determined by the alphabet size Σ (where Σ contains all possible characters or amino acids in protein sequence i.e. $ACDEFGHIKLMNPQRSTVWXY$) and the length of the minimizers denoted as m (we set m to 3, which is decided using the standard validation set approach). Hence, the length of the vector is $|\Sigma|^m$.

***PWM2Vec* [1]:** The PWM2Vec is a feature embedding method that transforms protein sequences into numerical representations. Instead of relying on the frequency of k-mers, PWM2Vec assigns weights to each amino acid within a k-mer. These weights are determined using a position weight matrix (PWM) associated with the k-mer. By considering the position-specific importance of amino acids, PWM2Vec captures both the relative significance of amino acids and preserves the ordering information within the sequences.