Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

A Machine Learning Approach to Predict and Evaluate the Human Immune System Response to Intrusion with B –Cell Epitopes as Protein-Protein Interaction

[1] Pradeep Kumar H S, [2] Harsha S

[1] *Assistant Professor, The National Institute of Engineering, Mysuru, pradee.nie@gmail.com, ORCID: 0000-0002-7606-0005*
[2] *Associate Professor, Department of AI & ML, R N S Institute of Technology, Bengaluru, harshahassan@gmail.com, ORCID: 0000-0001-9075-2625)*

ARTICLE INFO

Received: 30 Dec 2024
Revised: 12 Feb 2025
Accepted: 26 Feb 2025

ABSTRACT

The immune system relies on its capacity to identify substances, for its functioning but this ability weakens over time. B cells and T cells the elements of the system actively search for antigens. An antigen typically carries a membrane with a sequence of amino acids on its surface. These amino acid clusters, known as epitopes act as identifiers for antigens. B Cell Epitopes are epitopes that B cells recognize. In this study we. Evaluate a model that distinguishes between epitopes and non epitopes offering insight into the systems recognition process involving antigenic components. Various machine learning models have been utilized in our analysis, such, as regression, random forests, decision trees, Bernoulli Navie Bayes and XGBoost.

Keywords: Immune system, linear B-cell epitopes, linear non B-cell epitopes, machine learning

INTRODUCTION

B-cell epitopes have a limited number of prediction models for linear or conformational fields, yet they are crucial for vaccine development, clinical diagnosis, and antibody production. There are developed B-cell epitopes [41, 42]. However, since linear epitopes operate easily in trials, it is ideal to create precise prediction models for linear B-cell epitopes. These models can be machine learning-based or based on traditional propensity scale-based techniques [43, 44]. B-cell epitopes are areas of an antigen's surface that particular antibodies detect, bind to, and cause an immune response to occur. The basis of the adaptive immune system, which in vertebrates is in charge of immunological memory and antigen-specific responses, is this interaction.

Identification of these binding sites in the sequence or structure of the antigen is critical for the development of immunological treatments [37, 38], diagnostic procedures [36], and synthetic vaccinations [34, 35]. Over time, there has been an increased focus on these applications via the lens of epitope identification, particularly with relation to the safety advantages of developing synthetic vaccines [39]. B-cell epitopes can be broadly classified into two types: conformational (discontinuous) epitopes, which are composed of residues that are brought together by the folded protein structure but are not contiguous in the primary protein sequence, and linear (continuous) epitopes, which are composed of a linear sequence of residues [8]. Furthermore, it has been estimated that only 10% of B-cell epitopes are linear and that 90% are conformational [9].

All aforementioned immunological applications share the need for discovery of all possible epitopes for any given antigen, a process called “Epitope mapping”. Although epitope mapping can be carried out using several experimental techniques [11], it is time consuming and expensive, especially on a genomic scale. To address this problem and tap into the evergrowing data on epitopes deposited in biological databases daily, several computational methods for predicting conformational or linear B-cell epitopes have been published over the last decades

RELATED WORK

B-cell epitopes are classified mainly into two groups namely continuous and discontinuous. The continuous or linear epitopes consists of successive amino acids. Discontinuous epitopes are made up of spatially folded amino acids.

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Linear B-cell epitope has various applications in the production of antibody, immune malfunction diagnosis, design of vaccine based on epitope, deimmunization of the proteins and in autoimmunity [1, 2]. Predicting B-Cell epitopes are costly and it consumes time for experimental methods. To overwhelm experimental methods limitations many algorithms have been proposed and evaluated to predict linear B-cell epitope and linear non B-cell epitope. Some the most widely used linear B-cell epitope predictors: BcePred [3], BepiPred [4], ABCpred [5], COBEpro [6], SVMTriP [7], LBtope [8], LBEEP [9]. Relatively small percentage of linear B-cell epitopes, most methods developed over the past few years focus on their prediction. This is mainly attributed to the requirement of an antigen's 3D structure when predicting its conformational epitopes [10]. few hybrid methods employ both approaches for better predictive performance [11, 12].

BcePred was published by Raghava *et al.* [3], and is based on a plethora of physicochemical propensity scales utilizing amino acid properties, such as hydrophilicity and antigenicity, either individually or in combination. Moreover, it achieved a reported 56% sensitivity, 61% specificity and its highest accuracy of 58.70%, on a data set obtained from the database Bcipep [13], using a combination of flexibility, hydrophilicity, polarity and surface accessibility propensity scales BepiPred was developed by Lund *et al.*[4], and it is the first ever method that utilizes an HMM. The HMM was trained using a data set derived from the database Antijen [14] and the Pellequer data set [24], and was then combined with Parker's hydrophilicity scale, resulting in the BepiPred method. This method managed to achieve an Area Under Curve (AUC) of the Receiver Operating Characteristic (ROC) curve of 0.671 ± 0.013 on the Pellequer data set. To improve the performance, some machine learning-based models have been developed [15–25].

Although these models have been developed, their performance is still away from satisfactory level. For example, the newly developed BepiPred2.0 [11] model and previous version Bepipred1.0 [8] were assessed on a test dataset, which contained 11,839 positive and 18,722 negative validated peptides obtained from the Immune Epitope Database (IEDB) [26]. The results indicated that the values of area under the receiver operating characteristic (ROC) curve (AUC) were only 57.40 and 54.80%, respectively. Rubinstein et al. [27] used two Naïve Bayesian classifiers to develop structure-based and sequence-based approaches. SEPPA 2.0 combines amino acid index (AAindex) characteristics in the SEPPA algorithm in the calculation of cluster coefficients [28]. AA index in SEPPA 2.0 is consolidated via Artificial Neural Networks (ANN). However, SEPPA 3.0 adds the glycosylation triangles and glycosylation-related AAindex to SEPPA 2.0 by Zhou et al., [29]. Glycosylation-related AAindex is consolidated to SEPPA 3.0 via ANN. Several researchers utilized the advantages of random forest by Dalkas & Rooman [30]. Galar et al., [31], studies focus on class imbalance using various approaches that are mainly divided into four, including data and algorithm levels, cost-sensitive, and ensemble. Gary [32], proposed data level approach, the resampling method is used to ensure a balanced distribution of data. Kavitha et al.,[45] used random forest and have got accuracy of 0.89. R. Liu et al. specify to include family and context information to improve the accuracy [46][47]. L. Zhao prioritize precision and recall as better indicators of model performance on imbalanced datasets [48]. Organism-specific training of machine learning models demonstrates superior performance in B-cell epitope prediction [49]. Yang et al. proposed a B-cell epitope prediction method based on diverse class-conditional data selection policies, achieving 92.88% accuracy by training complementary classifiers on strategically chosen subsets of data [51].

The main challenges in prediction of linear B-cell epitopes and linear non B-cell epitopes are class imbalance, sequence of patters and takes time for experimental methods. Therefore, it is necessary to develop more accurate models for predicting linear B-cell epitopes and linear non B-cell epitopes.

PROPOSED METHODOLOGY

In this paper, proposed methodology designed to analyse both amino acids given in sequence of patterns and binary values. It consists of steps such as acquisition and loading of data, pre-processing data, exploratory data analysis, tokenization and sequence of words, data segmentation to create training and testing data sets, development of machine learning models and predicting linear B-cell epitopes and linear non B-cell epitopes. The figure 1 shows the proposed methodology.

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

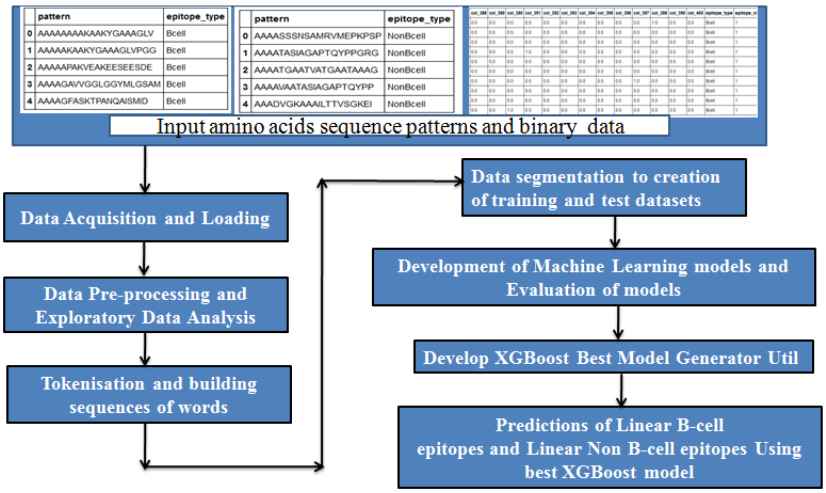

Figure 1: Proposed methodology.

Data Acquisition and Loading

Typically, the development of machine learning classifiers requires a training data set and a test data set. However, due to the fact that the individual training data sets for each predictor contained a significant number of overlapping sequences, gathered from a selected LBtope server discriminate the linear B cell epitopes and non-epitopes for a given protein sequence [33]. Downloaded all the fixed length peptides (epitopes & non-epitopes) of 20 residues. Lbtope_Fixed dataset contain 19803 unique positive patterns or B-cell epitopes and 28329 unique negative patterns or non B-cell epitopes, where each pattern has 20-residues. We also removed patterns common in both types of patterns. Our final Lbtope_fixed dataset contains 12063 B-cell epitopes and 20589 non-;epitopes. We also collected protein sequence in binary values it consists of 12,063 B-cell epitopes and 20,589 Non B-cell epitopes. Figure 2a and Figure 2b shows the sample linear B-cell epitopes and non B-cell epitopes sequence patterns with 20 residues respectively. Figure 3 and figure 4 shows binary sequence of amino linear B-cell epitopes and non B-cell epitopes sequence patterns with 20 residues respectively.

	pattern	epitope_type
0	AAAAAAAAKAAKYGAAAGLV	Bcell
1	AAAAAKAAKYGAAAGLVPGG	Bcell
2	AAAAAPAKVEAKEESEESDE	Bcell
3	AAAAGAVVGGLGGYMLGSAM	Bcell
4	AAAAGFASKTPANQAISMID	Bcell

Figure 2a. B-Cell epitopes with 20 residues

	pattern	epitope_type
0	AAAASSSNSAMRVMEPKPSP	NonBcell
1	AAAATASIAGAPTQYPPGRG	NonBcell
2	AAAATGAATVATGAATAAAG	NonBcell
3	AAAAVAATASIAGAPTQYPP	NonBcell
4	AAADVGKAAAILTTVSGKEI	NonBcell

Figure 2b. Non B-Cell epitopes with 20 residues

col_388	col_389	col_390	col_391	col_392	col_393	col_394	col_395	col_396	col_397	col_398	col_399	col_400	epitope_type	epitope_cl
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	Bcell	1
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	Bcell	1
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	Bcell	1
0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	Bcell	1
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	Bcell	1
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	Bcell	1
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	Bcell	1
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	Bcell	1
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	Bcell	1
0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	Bcell	1

Figure 3. Binary sequence of amino linear B-cell epitopes

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

col_389	col_390	col_391	col_392	col_393	col_394	col_395	col_396	col_397	col_398	col_399	col_400	epitope_type	epitope_class
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	NonBcell	0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	NonBcell	0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	NonBcell	0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	NonBcell	0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	NonBcell	0
1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	NonBcell	0
1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	NonBcell	0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	NonBcell	0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	NonBcell	0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	NonBcell	0

Figure 4. Binary sequence of amino linear non B-cell epitopes

Data Pre-processing and Exploratory Data Analysis

To understand the data distribution, Exploratory Data Analysis (EDA) is carried out; fist simple graph is plot on non B-Cell and B-cell epitopes to know the count. Figure 5 shows the 20,569 non B-cell epitopes and 12,059 B-cell epitopes. Also EDA is performed on average length of all the epitopes. Figure 6 shows the average length of all the epitopes. Some of the epitopes values in the dataset come out to be more. It requires data pre-processing on epitopes dataset. From the given data missing values are handled by the dropping the columns which contains missing values. Since missing values columns were few some of the columns are dropped. In case of amino acids given in sequence patterns then length is calculated, some of the amino acids sequence patterns were exceeding 20 lengths. Those epitopes sequence patterns are eliminated and epitopes with length 20 are retained in pre-processed dataset. EDA is performed on sequence patterns dataset to understand the frequency distribution of each amino acid. Figure 7 shows the EDA on amino residues distributed in sequence patterns dataset.

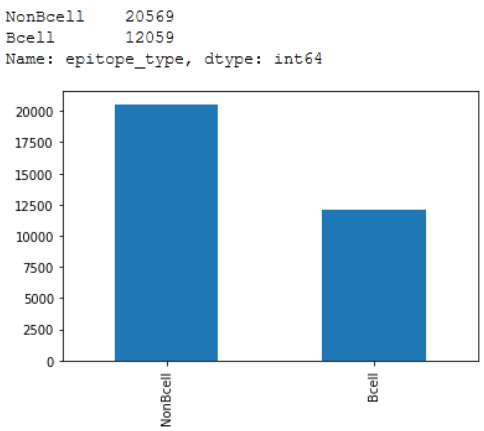

Figure 5. Count of B-cell and non B-cell epitopes

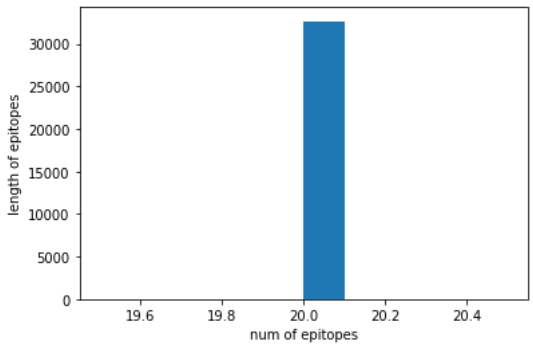

Figure 6. EDA on length of epitopes

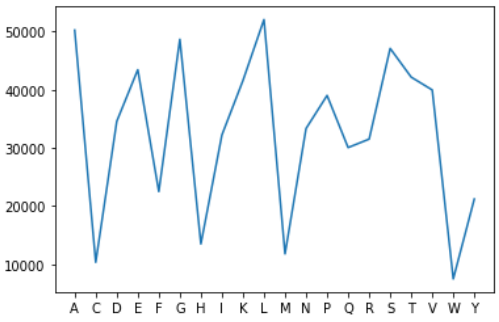

Figure 7. EDA on amino residues distributed in dataset

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Tokenisation and building sequences of words

The sequence patterns are tokenised in to characters and for each characters indexing is performed. Procedure to convert the sequence patterns into word with indexing is as follows:

Step1: Take input features, Example: Given input amino

acid="ACDEFGHIKLMNPQRSTVWY"

Step2: Traverse over the input feature and assign numbers to each characters and store in

character dictionary.

Step 3: Replace word index by using generated character dictionary. Here, Sequence patterns

are converted into numerical values

Output: {'A': 1, 'C': 2, 'D': 3, 'E': 4, 'F': 5, 'G': 6, 'H': 7, 'I': 8, 'K': 9, 'L': 10, 'M': 11, 'N':

12, 'P': 13, 'Q': 14, 'R': 15, 'S': 16, 'T': 17, 'V': 18, 'W': 19, 'Y': 20, 'UNK': 21}

Data segmentation to creation of training and test datasets

In this phase, target column is set for sequence pattern B-cell epitopes are labelled with class 1 and non B-cell epitopes labelled with class 0. Then dataset is split into training and testing dataset with 80:20 ratios.

XGBoost

XGBoost is an ensemble machine learning built based on decision tree that uses gradient boosting [20]. Ensemble machine learning combines the predictive output of multiple learned models. The aggregated models can be either same algorithm learnt or different learning algorithms. Bagging and boosting are the most commonly used in ensemble learning techniques. In bagging technique many decision trees are computed in parallel from the initial learners. Data patters with replacement are provided to the learners during the training. The average output will be the final prediction from all the learners. In boosting technique, the built trees are consecutively aims at reducing the errors from the previous built trees. Every tree receives from its predecessors and residual errors are updated. Initial learning in boosting may be weak learners and the bias are high and power of predictive can be better than guessing randomly.

In contrast to RF, where tress are grown to maximum length, boosting technique helps to use less splits in trees. Small trees can be highly interpretable because of not very deep in generating. Parameters such as iteration, number of trees, depth of trees and learning rate of gradient boosting can be optimised by validating through k-fold cross validation. Obtaining more number of tress may lead to overfitting. So, there is required to properly selecting the termination criteria in boosting. Boosting technique consists of three steps:

Initial built model Po is determined to predict target parameter‘t’. This model will be correlated with an residual (t-Po)

An new generated model m1 is fitted with residual in previous step.

Now, Po and m1 gives the P1 , the mean square error of P1 will be lesser than Po.

These steps can be made in ‘n’ iterations until the residual errors are minimised as shown in below equation.

$$P_n(x) < P_{n-1}(x) + m_n(x)$$

For gradient boosting following steps are followed.

Po (x) with initial model are determined and function to minimise the Mean Square error in this case is:

$$P_0(x) = \arg\min_{\phi} \sum_{i=1}^{n} S(\phi_i - \phi)^2$$

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

The loss function f_{in} in gradient are determined iteratively, where δ is an rate of learning :

$$f_{in} = -\delta\left[\frac{\partial(S(\phi_i, P(x_i))}{\partial P(x_i)}\right]_{P(x)=P_{n-1}(x)}$$

XGBoost Best Model Generator Util

Algorithm is proposed to build the best XGBoost model for predicting the B-cell and Non B-cell epitopes. The XGBoost Best Model Generator Util algorithm is as follows:

Algorithm: ***XGBoost Best Model Generator Util***

Input: x_train and y_train values, x – input features , y- target values

Method: Initialise with hyper parameters for XGBoost

n_estimators: 250, 1000, 100

learning rate: 0.01, 0.1

max_depth: 5, 15, 2

Use stratified k-fold validation

Perform grid search to find the best fit for the model

Select the best XGBoost model among many XGBoost models based on the model which gives less error rate

Output: Best values for hyper parameters to build the XGBoost model for a given input values and target values.

Implementation and Results

Utilizing Anaconda version 3, the work is implemented with Python 3.7. The development includes libraries like plotly, sklearn, pandas, and numpy. For numerical analysis, the Numpy library in the Python packages offers scientific computing capabilities. A Python machine learning package available as open source is called Scikit-Learn. You can get the 32,651 sequence pattern dataset from resource [33], which includes both B-cell and non-B-cell epitopes. Sequence patterns longer than twenty amino acids are deleted, and missing values are eliminated. Numerical values are generated from amino acid sequences using tokenization of sequence patterns. Model Generator XGBoost Is Superior To determine which hyperparameter settings provide the lowest error rate, the util algorithm is suggested.

Table 1: precision, recall and f1-score results from machine learning algorithms for binary amino 20 residues

Proposed ML Models	Class	Precision	Recall	F1-score	Accuracy (%)
DT	0	0.6311	0.9947	0.7722	63.01
	1	0.4762	0.0083	0.0163	
RF	0	0.6487	0.9818	0.7812	65.34
	1	0.7500	0.0933	0.1659	
BNB	0	0.6647	0.8340	0.7398	63.01
	1	0.4993	0.2823	0.3607	
LR	0	0.6527	0.9035	0.7579	63.61

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

	1	0.5223	0.1799	0.2677	
XGBoost	0	0.6406	0.9735	0.7727	63.90
	1	0.6022	0.0684	0.1229	
XGBoost Best Model Generator Util	0	0.6511	0.9273	0.7651	64.10
	1	0.5517	0.1526	0.2390	

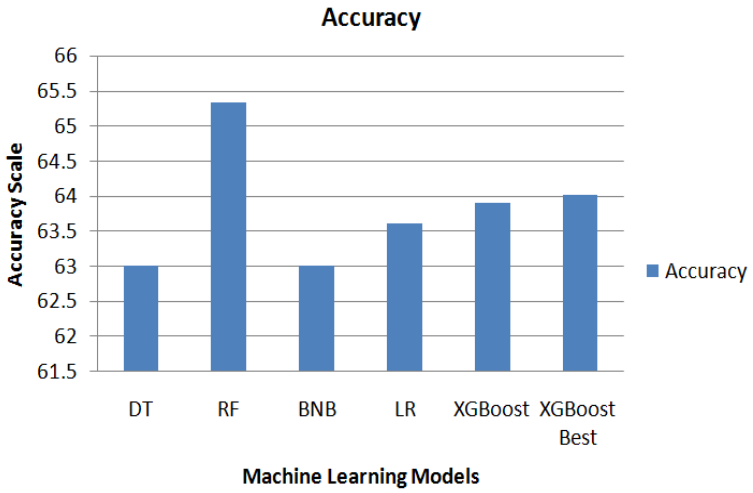

Figure 8: Obtained accuracy results from machine learning algorithms for binary amino 20 residues.

Table 2: precision, recall and f1-score results from machine learning algorithms for sequence patterns 20 residues

Proposed ML Models	Class	Precision	Recall	F1-score	Accuracy (%)
DT	Non B-cell	0.6346	0.9888	0.7730	63.39%
	B-cell	0.5965	0.0282	0.0538	
RF	Non B-cell	0.6529	0.9806	0.7838	65.90
	B-cell	0.7688	0.1102	0.1928	
BNB	Non B-cell	0.6305	1.0000	0.7734	63.05
	B-cell	0.4000	0.3000	0.4000	
LR	Non B-cell	0.6305	1.0000	0.7734	63.05
	B-cell	0.4000	0.3000	0.4000	
XGBoost	Non B-cell	0.6507	0.9301	0.7657	64.11
	B-cell	0.5535	0.1479	0.2335	
XGBoost Best Model Generator Util	Non B-cell	0.6507	0.9301	0.7657	64.11
	B-cell	0.5535	0.1479	0.2335	

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

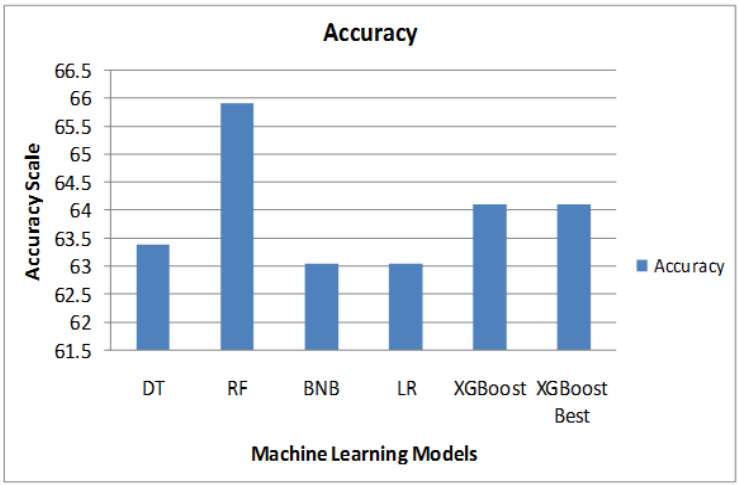

Figure 9: Obtained accuracy results from machine learning algorithms

for sequence patterns 20 residues

Conclusion

This research presents six machine learning models that are assessed using an amino acid dataset that is presented in binary values and sequence patterns. It was difficult to convert the unstructured sequence data into numerical values and remove missing values. Applied EDA to the provided dataset in order to determine the distribution of amino acids. A comparative analysis of machine learning models reveals that random forest produces more accurate outcomes. When compared to previous ML models that have been evaluated, the proposed XGBoost best model generator tool performs better in terms of recall and f1-score. Prediction accuracy declines as a result of data imbalance. We plan to investigate deep learning approaches using various amino acid datasets in the future.

References

1. Dudek NL, Perlmutter P, Aguilar MI, Croft NP, Purcell AW, Epitope discovery and their use in peptide based vaccines. Curr Pharm Des 16: 3149– 3157, 2010.
2. Bryson CJ, Jones TD, Baker MP, Prediction of immunogenicity of therapeutic proteins: validity of computational tools. BioDrugs 24: 1–8, 2010.
3. S. Saha, G.P.S. Raghava, BcePred: Prediction of Continuous B-Cell Epitopes in Antigenic Sequences Using Physico-chemical Properties, in, Springer Berlin Heidelberg, Berlin, Heidelberg, 2004, pp. 197-204.
4. J.E. Larsen, O. Lund, M. Nielsen. Improved method for predicting linear B-cell epitopes. Immunome Res 2006; 2:2.
5. S. Saha, G.P. Raghava. Prediction of continuous B-cell epitopes in an antigen using recurrent neural network. Proteins 2006; 65:40-48.
6. M.J. Sweredoski, P. Baldi. COBEpro: a novel system for predicting continuous B-cell epitopes. Protein Eng Des Sel 2009; 22:113-120.
7. B. Yao, L. Zhang, S. Liang, C. Zhang. SVMTriP: A Method to Predict Antigenic Epitopes Using Support Vector Machine to Integrate Tri-Peptide Similarity and Propensity. PLOS ONE 2012; 7:e45152.
8. H. Singh, H.R. Ansari, G.P. Raghava. Improved method for linear B-cell epitope prediction using antigen's primary sequence. PLoS One 2013; 8:e62216.
9. V. Saravanan, N. Gautham. Harnessing Computational Biology for Exact Linear B-Cell Epitope Prediction: A Novel Amino Acid Composition-Based Feature Descriptor. OMICS 2015; 19:648-658.
10. D.R. Flower. Immunoinformatics. Predicting immunogenicity in silico. Preface. Methods Mol Biol 2007; 409:v-vi.
11. M.J. Sweredoski, P. Baldi. COBEpro: a novel system for predicting continuous B-cell epitopes. Protein Eng Des Sel 2009; 22:113-120.

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

12. M.C. Jespersen, B. Peters, M. Nielsen, P. Marcatili. BepiPred-2.0: improving sequence based B-cell epitope prediction using conformational epitopes. Nucleic Acids Res 2017; 45:W24-W29.
13. S. Saha, M. Bhasin, G.P. Raghava. Bcipep: a database of B-cell epitopes. BMC Genomics 2005; 6:79.
14. C.P. Toseland, D.J. Clayton, H. McSparron, S.L. Hemsley, M.J. Blythe, K. Paine, I.A. Doytchinova, P. Guan, C.K. Hattotuwagama, D.R. Flower. AntiJen: a quantitative immunology database integrating functional, thermodynamic, kinetic, biophysical, and cellular data. Immunome Res 2005; 1:4.
15. Larsen JEP, Lund O, Nielsen M. Improved method for predicting linear B-cell epitopes. Immunome Res. 2006;2:2.
16. Singh H, Ansari HR, Raghava GPS. Improved method for linear B-cell epitope prediction using Antigen's primary sequence. PLoS One. 2013;8:e62216.
17. Shen W, Cao Y, Cha L, Zhang X, Ying X, Zhang W, et al. Predicting linear B-cell epitopes using amino acid anchoring pair composition. BioData Min. 2015;8:14.
18. Jespersen MC, Peters B, Nielsen M, Marcatili P. BepiPred-2.0: improving sequence-based B-cell epitope prediction using conformational epitopes. Nucleic Acids Res. 2017;45:W24–9.
19. Chen J, Liu H, Yang J, Chou K-C. Prediction of linear B-cell epitopes using amino acid pair antigenicity scale. Amino Acids. 2007;33:423–8.
20. EL-Manzalawy Y, Dobbs D, Honavar V. Predicting linear B-cell epitopes using string kernels. J Mol Recognit. 2008;21:243– 55.
21. Davydov II, Tonevitskiĭ AG. Linear B-cell epitope prediction. Mol. Biol. (Mosk). 2009;43:166–74.
22. Wee LJK, Simarmata D, Kam Y-W, Ng LFP, Tong JC. SVM-based prediction of linear B-cell epitopes using Bayes feature extraction. BMC Genomics. 2010;11:S21.
23. Wang H-W, Lin Y-C, Pai T-W, Chang H-T. Prediction of B-cell linear epitopes with a combination of support vector machine classification and amino acid propensity identification. J Biomed Biotechnol. 2011;2011:432830.
24. Yao B, Zhang L, Liang S, Zhang C. SVMTriP: a method to predict antigenic epitopes using support vector machine to integrate tri-peptide similarity and propensity. PLoS One. 2012;7:e45152.
25. Gao J, Faraggi E, Zhou Y, Ruan J, Kurgan L. BEST: improved prediction of B-cell epitopes from antigen sequences. PLoS One. 2012;7:e40104.
26. Vita R, Overton JA, Greenbaum JA, Ponomarenko J, Clark JD, Cantrell JR, et al. The immune epitope database (IEDB) 3.0. Nucleic Acids Res. 2015;43:D405–12.
27. Rubinstein ND, Mayrose I, Pupko T. 2009. A machine-learning approach for predicting B-cell epitopes. Molecular Immunology 46:840_847 DOI 10.1016/j.molimm.2008.09.009.
28. Qi T, Qiu T, Zhang Q, Tang K, Fan Y, Qiu J, Wu D, Zhang W, Chen Y, Gao J, Zhu R, Cao Z. 2014. SEPPA 2.0_more refined server to predict spatial epitope considering species of immune host and subcellular localization of protein antigen. Nucleic Acids Research 42(May):59_63
29. Zhou C, Chen Z, Zhang L, Zhang L, Yan D, Mao T, Tang K, Qiu T, Cao Z. 2019. SEPPA 3.0_enhanced spatial epitope prediction enabling glycoprotein antigens. Nucleic Acids Research 47(May):388_394.
30. Dalkas GA, Rooman M. 2017. SEPIa, a knowledge-driven algorithm for predicting conformational B-cell epitopes from the amino acid sequence. BMC Bioinformatics 18(95):1_12.
31. Galar M, Fern A, Barrenechea E, Bustince H. 2012. Hybrid-based approaches. IEEE Transactions on Systems Man and Cybernetics Part C (Applications and Reviews) 42(4):463_484
32. Gary MW. 2012. Foundation of imbalanced learning. In: He H, Ma Y, eds. Imbalanced learning: foundations, algorithms, and applications. Hoboken: John Wiley & Sons, Inc, 13_42.
33. Hifzur Rahman Ansari, Harinder Singh, G. P. S. Raghava, LBtope : Prediction of Linear B-cell Epitopes: https://webs.iiitd.edu.in/raghava/lbtope
34. E.E. Hughes, H.E. Gilleland, Jr. Ability of synthetic peptides representing epitopes of outer membrane protein F of Pseudomonas aeruginosa to afford protection against P.aeruginosa infection in a murine acute pneumonia model. Vaccine 1995; 13:1750-1753.
35. J.P. Tam, Y.A. Lu. Vaccine engineering: enhancement of immunogenicity of synthetic peptide vaccines related to hepatitis in chemically defined models consisting of T- and B-cell epitopes. Proc Natl Acad Sci U S A 1989; 86:9084-9088.

Journal of Information Systems Engineering and Management
2025, 10(61s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

36. R.C. Russi, E. Bourdin, M.I. Garcia, C.M.I. Veaute. In silico prediction of T- and B-cell epitopes in PmpD: First step towards to the design of a Chlamydia trachomatis vaccine. Biomed J 2018; 41:109-117.
37. G.A. Schellekens, H. Visser, B.A. de Jong, F.H. van den Hoogen, J.M. Hazes, F.C. Breedveld, W.J. van Venrooij. The diagnostic properties of rheumatoid arthritis antibodies recognizing a cyclic citrullinated peptide. Arthritis Rheum 2000; 43:155-163.
38. A.J. Chirino, M.L. Ary, S.A. Marshall. Minimizing the immunogenicity of protein therapeutics. Drug Discov Today 2004; 9:82-90.
39. H. Shirai, C. Prades, R. Vita, P. Marcatili, B. Popovic, J. Xu, J.P. Overington, K. Hirayama, S. Soga, K. Tsunoyama, D. Clark, M.P. Lefranc, K. Ikeda. Antibody informatics for drug discovery. Biochim Biophys Acta 2014; 1844:2002-2015.
40. E.H. Nardin, J.M. Calvo-Calle, G.A. Oliveira, R.S. Nussenzweig, M. Schneider, J.M. Tiercy, L. Loutan, D. Hochstrasser, K. Rose. A totally synthetic polyoxime malaria vaccine containing Plasmodium falciparum B cell and universal T cell epitopes elicits immune responses in volunteers of diverse HLA types. J Immunol 2001; 166:481-489.
41. Dhanda SK, Usmani SS, Agrawal P, Nagpal G, Gautam A, Raghava GPS. Novel in silico tools for designing peptide-based subunit vaccines and immunotherapeutics. Brief Bioinform. 2017;18:467–78.
42. Potocnakova L, Bhide M, Pulzova LB. An introduction to B-cell epitope mapping and in Silico epitope prediction. J Immunol Res. 2016;2016:6760830.
43. Parker JM, Guo D, Hodges RS. New hydrophilicity scale derived from high-performance liquid chromatography peptide retention data: correlation of predicted surface residues with antigenicity and X-ray-derived accessible sites. Biochemistry. 1986;25:5425–32.
44. Karplus PA, Schulz GE. Prediction of chain flexibility in proteins. Naturwissenschaften. 1985;72:212–3.
45. K. V. Kavitha, R. Saritha, and Vinod Chandra, "Computational prediction of continuous B-cell epitopes using random forest classifier," 2013 International Conference on Communication and Signal Processing*, India, 2013.
46. R. Liu et al., "Family-Specific Training Improves Linear B Cell Epitope Prediction for Emerging Viruses".
47. L. Liu, H. Yang and B. Cheng, "Prediction of Linear B-cell Epitopes Based on PCA and RNN Network".
48. L. Zhao, L. Wong and J. Li, "Antibody-Specified B-Cell Epitope Prediction in Line with the Principle of Context-Awareness".
49. A. Chharia and A. Narayan, "A novel fuzzy approach towards in silico B-cell epitope identification inducing antigen-specific immune response for Vaccine Design".
50. M. Alazmi and O. Motwalli, "Immuno-Informatics based Peptides: An Approach for Vaccine Development against Outer Membrane Proteins of Pseudomonas Genus".
51. H. Yang, Y. Zhou, B. Cheng and C. Li, "Prediction of B-cell epitopes using diverse class-conditional data selection policies".
52. J. Ashford, A. Ek´art and F. Campelo, "Estimating the Limits of Organism-Specific Training for Epitope Prediction".
53. K. V. Kavitha, R. Saritha and V. Chandra, "Computational prediction of continuous B-cell epitopes using random forest classifier".
54. R. Qiao, N. H. Tran, B. Shan, A. Ghodsi and M. Li, "Personalized workflow to identify optimal T-cell epitopes for peptide-based vaccines against COVID-19".
55. R. Reitmaier, "Review of Immunoinformatic approaches to in-silico B-cell epitope prediction".
56. J. Söllner, R. Grohmann, R. Rapberger, P. Perco, A. Lukas and B. Mayer, "Analysis and prediction of protective continuous B-cell epitopes on pathogen proteins".
57. J. K. Larsen, O. Lund and M. Nielsen, "Improved method for predicting linear B-cell epitopes.".
58. Y. EL-Manzalawy and V. Honavar, "Recent advances in B-cell epitope prediction methods".