scientific reports

OPEN

# Inferring linear-B cell epitopes using 2-step metaheuristic variant-feature selection using genetic algorithm

Pratik Angaitkar[1], Turki Aljrees[2], Saroj Kumar Pandey[3], Ankit Kumar[3]✉, Rekh Ram Janghel[1], Tirath Prasad Sahu[1], Kamred Udham Singh[4] & Teekam Singh[5]

**Linear-B cell epitopes (LBCE) play a vital role in vaccine design; thus, efficiently detecting them from protein sequences is of primary importance. These epitopes consist of amino acids arranged in continuous or discontinuous patterns. Vaccines employ attenuated viruses and purified antigens. LBCE stimulate humoral immunity in the body, where B and T cells target circulating infections. To predict LBCE, the underlying protein sequences undergo a process of feature extraction, feature selection, and classification. Various system models have been proposed for this purpose, but their classification accuracy is only moderate. In order to enhance the accuracy of LBCE classification, this paper presents a novel 2-step metaheuristic variant-feature selection method that combines a linear support vector classifier (LSVC) with a Modified Genetic Algorithm (MGA). The feature selection model employs mono-peptide, dipeptide, and tripeptide features, focusing on the most diverse ones. These selected features are fed into a machine learning (ML)-based parallel ensemble classifier. The ensemble classifier combines correctly classified instances from various classifiers, including k-Nearest Neighbor (kNN), random forest (RF), logistic regression (LR), and support vector machine (SVM). The ensemble classifier came up with an impressively high accuracy of 99.3% as a result of its work. This accuracy is superior to the most recent models that are considered to be state-of-the-art for linear B-cell classification. As a direct consequence of this, the entire system model can now be utilised effectively in real-time clinical settings.**

In the process of vaccine development, it is absolutely necessary to isolate Linear-B cell epitopes from their respective protein sequences. This difficult endeavour requires the completion of a number of signal processing procedures, the most important of which are the following: sequence collection, feature extraction, feature selection, feature classification, and post-processing. The protein sequences, as well as the presence or absence of linear B-cell epitopes, are both derived from the data collected in the laboratory. The step known as “feature extraction” is an essential one. During this step, a number of features, including monopeptide, dipeptide, tripeptide, histogram, and others, are derived from the protein sequences that are being used. The fact that these sequences contain all 20 different amino acids contributes to the high number of features that can be extracted from them. On the other hand, there is a possibility that the large number of features will slow down or reduce the accuracy of the classifier.

Researchers employ feature selection strategies to conquer this obstacle and improve the performance of the classifier. A feature selection model that is effective looks for feature vectors that have the greatest variance for protein sequences that are dissimilar to each other and the least variance for protein sequences that are similar to each other. As a result of the fact that the accuracy of Linear-B cell epitope prediction can be improved by using such methods, this stage of the vaccine development process is of critical importance. Many different feature selection models have been proposed by researchers over the years. Some examples of these models include principal component analysis (PCA), independent component analysis (ICA), and selection based on

[1]Department of Information Technology, National Institute of Technology, Raipur, G.E. Road, Raipur 492010, Chhattisgarh, India. [2]College of Computer Science and Engineering, University of Hafr Al Batin, 39524 Hafar Al Batin, Saudi Arabia. [3]Department of Computer Engineering & Applications, GLA University, Mathura, India. [4]School of Computing, Graphic Era Hill University, Dehradun, India. [5]Department of Computer Science and Engineering, Graphic Era Deemed to be University, Dehradun 248002, Uttarakhand, India. ✉email: iiita.ankit@gmail.com

the Genetic Algorithm (GA). By analysing the inter-class variance using Linear-B cell information, these models can effectively reduce the number of features.

Feature selection is the process of selecting a subset of relevant and informative features from the original feature set. The goal is to retain the most discriminative features while eliminating irrelevant or redundant ones. By reducing the number of features, feature selection improves model performance, simplifies the model, and enhances interpretability[1,2]. Feature extraction is the process of transforming raw data or features into a new representation with reduced dimensionality while retaining the most important information. It aims to discover more meaningful and compact feature representations, which can lead to improved model performance and generalization. Hence, dimensionality reduction, feature selection, and feature extraction are crucial techniques in machine learning and data analysis. They help improve model efficiency, interpretability, and generalization by identifying and utilizing relevant and informative features while reducing computational overhead and potential overfitting. Swarm intelligence-based methods and graph-based algorithms offer effective approaches to tackle these challenges[3].

In order to develop a ML model, certain features must be important to each other. The input sequence, which is difficult to process and training without feature selection. Feature selection is required to minimize the memory space and time complexity of the model. If the size of the features is too large, the developing model may have problems with the overfitting or under fitting because some features may be irrelevant. The selected features are given to a classification engine, which uses ML models like Support vector machine (SVM), Random Forest (RF), Naïve Bias (NB), Multilayer Perceptron (MLP), etc. Even deep learning models such as Convolutional neural network (CNN), Long short term memory (LSTM), Recurrent Neural Network (RNN), Gatted Recurrent Unit (GRU), etc. methods can use for protein sequence classification. These models aim at classifying selected features for estimating the presence or absence of Linear-B cells. To improve the accuracy of this classification, various post-processing models are proposed by researchers[4,5], which include a combination of different classifiers, model tuning, hyper-parameter optimization, etc. Such a model that uses encoding & transposition for feature extraction, convolution and batch normalization for feature selection; and LSTM based CNN for classification can be observed from Fig. 1.

The paper contributes in the following aspect:

- Proposed a novel two-step feature extraction and selection model.
- Proposed a novel instance-based ensemble classification model for linear B-cell epitope detection.
- Proposed 2-step metaheuristic model using linear support vector classifier (LSVC) and Modified Genetic Algorithm (MGA) for efficient feature selection.

Section "Related Work" presents a comprehensive survey of similar machine learning models and architectures aimed at the classification of Linear-B cells. Readers can use this survey to identify the best system models and architectures for predicting Linear-B cell epitopes. Section "Proposed model based on ensemble classifier" goes over the proposed novel 2-step metaheuristic variant-feature selection-based ensemble classification model in detail. This method is notable for being the first of its kind to be used in the classification of Linear-B cell epitopes (LBCE). The proposed model is then rigorously tested on various protein datasets in Section "Parametric evaluation and comparison". Its precision is thoroughly evaluated and compared to existing models. Parameters such as accuracy, precision, recall, f-measure, Area under the Curve (AUC), and Receiver Operating Characteristics (ROC) are included in the evaluation. Finally, Section "Conclusion and future work" contains the concluding

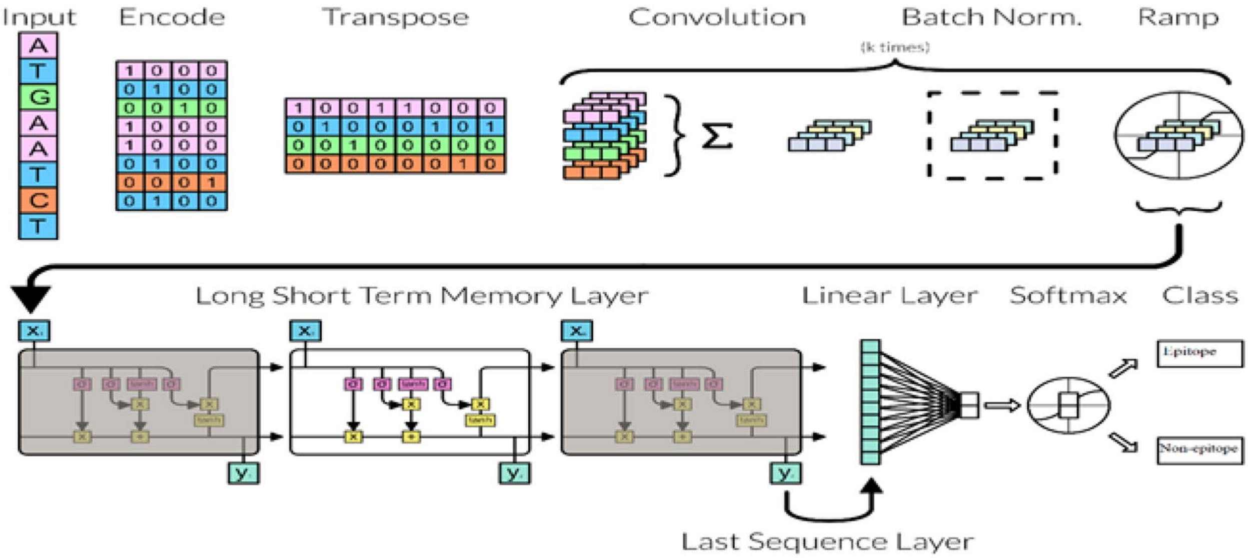

**Figure 1.** Protein sequence classification model by using LSTM.

remarks, which include some interesting observations about the performance of the proposed model. The section also contains suggestions for improving the model's capabilities.

## Related work

A large variety of algorithms have been proposed by researchers over the years for the identification of Linear-B cells. This work has been observed to have exponential growth during the last 2 years due to the introduction of CoVID-19, and its intensive vaccination research. This estimation can be done via effective feature extraction and selection as observed from[6], wherein a deep learning model was used to obtain an accuracy of 85% for different datasets. The model uses a CNN to perform this task, which makes it highly efficient for the detection of Linear-B cells. Similar models can be observed from[7], wherein different classification techniques and their nuances are discussed. From this research, it can be observed that hybrid classification models must be used for effective LBCE classification. Such a model can be observed from[8], wherein a combination of different CNN architectures (AlexNet and GoogLeNet with SVM) is done to obtain the final classifier. The classifier is used for classifying lymphocytes, monocytes, eosinophils, and neutrophils in while blood cells (WBCs), but can be used for protein sequence classification. Another similar type of work can be observed from[9], wherein VGGNet is combined with a statistically enhanced Salp Swarm Algorithm (SWA) for improved accuracy of WBC-based classification. This indicates that swarm intelligence techniques can be used for the classification of any type of sequence with high efficiency. An extension to these models for linear B-cell classification can be observed from[10], wherein Sequence and Evolutionary Features are combined to obtain an accuracy of 63%, which is low for real-time clinical applications. Similar models can be observed from[10–13] and[14] wherein linear classifiers, immuno-informatics, self-organizing maps, and deep CNN models are described. These models are able to obtain accuracy in the range of 85–90% on different protein sequence datasets.

SVM classifier is one of the most consistent choices for LBCE classification as observed from[15], wherein an accuracy of 72.52% is achieved. This accuracy is for the training set, while test set accuracy is in the range of 60–70% depending upon the dataset. Other models can be referred from[16,17] wherein methods for estimation of LBC for vaccination design are described. This detection can be used for the estimation of multiple sclerosis[18], and other diseases. Thereby it is recommended that this model be optimized to support a larger number of applications. An ensemble learning approach for high efficiency can be observed from[19,20] and[21], wherein gradient boosting (GB) and extremely randomized tree (ERT) are combined. An accuracy is obtained between 50 and 70% is using these methods, which can be improved using deep learning models. Other modular applications of linear B-cell estimation can be observed from[22–26] and[27], wherein viruses like zika, dengue, SARS-CoV-2, porcine epidemic diarrhoea virus, Newcastle disease virus, South American and African Trypanosoma vivax strains, and antigen identifications are discussed. All these applications utilize Linear-B cell classification to improve the efficiency of given virus prediction. Similar applications and research areas can be observed from[28–31] and[32] wherein SARS-CoV-2 exposure, SARS-CoV-2 disease severity, diffuse large B cell lymphoma, SARS-CoV-2 spike, cancer detection, and lymphoma cell analysis are discussed. Based on these applications, it can be observed that the current accuracy of Linear-B cell classification is moderate and must be improved via better classification models. The prediction of new COVID-19 cases was addressed in[33] using a hybridized algorithm that combined the machine learning adaptive neuro-fuzzy inference system (ANFIS) with enhanced GA metaheuristics. The study focused on optimizing and adjusting parameters through the utilization of GA. In[34], author analyzed COVID-19 using blood samples with over 100 features. The genetic algorithm is employed for feature reduction, and a model implemented with relief and ant colony optimization achieves high accuracy (98.7%), sensitivity (96.76%), specificity (98.80%), and AUC (92%). The algorithm outperforms other state-of-the-art methods. In[35], the author proposed radiological methodologies, such as chest x-rays and CT scans, are widely used for COVID-19 diagnosis and monitoring. This paper proposes an effective method using a convolutional neural network (CNN) and an enhanced evolutionary algorithm to detect COVID-19 from chest X-ray images. By replacing the last CNN layer with k-nearest neighbours (KNN) classifier and optimizing hyperparameters, the proposed method achieves significantly improved accuracy compared to existing models. In[34], the author proposes a hybrid approach combining genetic algorithms (GAs) with artificial bee colony (ABC) swarm intelligence to improve Artificial neural networks (ANN) training. By incorporating exploration from the ABC algorithm, the proposed method overcomes drawbacks of GAs such as local optima trapping. Simulations on medical datasets demonstrate robust performance and reduced classification test error rates. In[36] presented a computational intelligence-based framework combining CNN and GA for detecting COVID-19 cases. The framework utilizes multi-access edge computing technology, enabling end-users to access the CNN on the cloud. By leveraging this framework, early detection of COVID-19 can be achieved, aiding in improved treatment and transmission control. The proposed CNN-GA model achieves a high accuracy of 98.48% in classifying COVID-19 X-ray images, surpassing previous studies' performance. This framework offers an automated tool accessible to users with 5G devices for efficient COVID-19 detection.

In the next section, such a classification model along with its internal design structure is discussed. After referring to this section, researchers will be able to design such a model that will allow them to develop high accuracy linear B-cell identification systems.

**Ethics approval.** All authors contributed to the conception and design of the study. All authors read and approved the final manuscript.

## Proposed model based on ensemble classifier

It is evident from the literature review that, extraction of Linear-B cell patterns from protein sequences has been extensively done in the past. The existing models combine various feature extraction and selection methods with ML classification models to achieve accuracies in the range of 80–90%, which makes them unsuitable for clinical use. Thus, to design a high-efficiency Linear-B cell classification engine, a novel 2-step meta-heuristic variant-feature selection-based ensemble classifier named as MH2VFSEC is proposed in this paper. The model works in 3-steps, which are labelled as multiple feature extraction, intelligent feature selection, and ensemble classification. Each of these steps are mentioned and described in detail in separate sub-sections of this paper. Due to the simplicity of operation, this work can be reproduced with the assistance of these sub-sections.

**Multiple feature extraction.** An efficient feature extraction model should be able to convert the given input dataset into class-level distinguishable feature vectors. These vectors must be extracted such that even the minutest of variations are incorporated in the process. To design such a feature extraction model, unigram, bigram, and trigram features were extracted from amino acid sequences. It is observed that a combination of 20 amino acids (AA) in the sequence range (ACDEFGHIKLMNPQRSTVWY) is sufficient to form any protein sequence. Thus, unigram features ($F_{uni}$) are extracted using Eq. (1) as follows,

$$F_{uni_i} = \sum_{i=1}^{20} \sum_{j=1}^{N} \left| S_j == AA_i \right| \tag{1}$$

where, 'N' is the length of the protein sequence (S), $|x|$ indicates a count of protein sequence for the given amino acid, while $F_{uni_i}$ is the number of occurrences for the $i$th amino acid in the sequence. The length of this feature vector is 20 elements due to 20 amino acids used for feature extraction. Similarly, bigram features are extracted using Eq. (2) as follows,

$$F_{bi_{i,j}} = \sum_{i=1}^{20} \sum_{j=1}^{20} \sum_{k=2}^{N} \left| S_{k-1} == AA_i \& S_k == AA_j \right| \tag{2}$$

Due to double summation, this feature vector produces an array of 20 × 20 different elements. All these elements are combined which create a concrete feature vector of size 400. On the same lines, a trigram feature vector is extracted using the following Eq. (3),

$$F_{tri_{i,j,k}} = \sum_{i=1}^{20} \sum_{j=1}^{20} \sum_{k=1}^{20} \sum_{l=3}^{N} \left| S_{k-2} == AA_i \& S_{k-1} == AA_j \& S_k == AA_k \right| \tag{3}$$

Due to triple summation, this feature vector produces an array of 20 × 20 × 20 different elements. All these elements are pooled which create a feature vector of size 8000. In combination with unigram feature, bigram features, and the class (1-for presence of Linear-B cell, 0-for absence of Linear-B cell), the total feature vector of 8421 values is generated. This feature vector is given to the metaheuristic feature selection model as described in the next sub-section.

**Metaheuristic model for feature selection.** To design an efficient feature selection unit, it is necessary that selected features of each class must have minimum intra-class variance, while features of different classes have maximum inter-class variance. To perform this task, a Modified Genetic Algorithm (MGA) model is designed. This model utilizes feature variance for the estimation of solution fitness, and combines it with accuracy values obtained from the stochastic features to select the most optimum feature-length. The features extracted from the previous sub-section are given as input to this system, and the following 2-step process is performed,

- Input,
    - Number of solutions (Ns)
    - Number of iterations (Ni)
    - Mutation factor (Mu)
    - Train to test set ratio (Tr)
- Output,
    - Selected training and testing set for optimum accuracy
    - Selected feature positions for optimum accuracy

- Part 1: Train set and test set selection,

    - Mark all solutions as 'to be modified'
    - Initialize current variance value (CVV) = 0
    - For each iteration in 0 to Ni

        - For each solution in 0 to Ns
        - If this solution is marked as 'not to be modified' then continue with next solution
        - Else, select a random number of training set samples such that the selected samples follow the given criteria,

    - Ratio of Number values in each of the classes is 1/k, where 'k' is the number of classes.
    - Evaluate average variance of each sample of one class, with all samples of other class, using the following Eq. (4),

$$V_{avg}=\sqrt{\frac{\sum_{a=1}^{m}\left(x_a-\frac{\sum_{i=1}^{m}\sqrt{\frac{\sum_{j=1}^{n}(x_j-\frac{\sum_{k=1}^{n}x_k}{n})^2}{n-1}}}{m}\right)^2}{m-1}} \tag{4}$$

Where, 'm' is the number of samples in the current class, 'n' is number of samples in the other class, and 'x' is the sample value (unigram, bigram and trigram).

- Find average fitness of all the classes, and evaluate fitness value as,

$$f_{sol}=\sum_{i=1}^{k}\frac{V_{avg_i}}{k} \tag{5}$$

- Accept this solution if $f_{sol}$ is less than CVV, else discard the solution, and generate a new one.

    - Generate 'Ns' number of solutions, and then find mutation threshold as follows,

$$M_{threshold}=\frac{\sum_{i=1}^{Ns}f_{sol_i}}{N_s}*Mu \tag{6}$$

    - Pass all solutions to next iteration that have fitness more than $M_{threshold}$, and mark them as 'not to be modified', else mark the remaining solutions as 'to be modified'

- At the end of 'Ni' iterations, select solution that has maximum fitness value, and use that division of dataset as training and testing set.

*Part 2: Feature selection for effective classification*
Mark all solutions as 'to be modified' to select the features for effective classification.

- Initialize Max Sequence Length which is the length of maximum sequence from both training and testing sets (SLMax)
- Initialize Min Sequence Length which is the length of minimum sequence from both training and testing sets (SLMin)
- Initialize current accuracy (CA)
- For each iteration in 1 to Ni

    - For each solution in 1 to Ns
    - If the solution is marked as 'not to be changed' then continue to next solution.
    - Else, select a random number between SLMin and SLMax, which will be the selected sequence length (SLse
    - Extract SLsel length sequences from all the training set protein sequences.
    - Apply Linear Support Vector Machine Classifier (LSVC) training on the extracted sequence, and obtain its ***test*** accuracy.
    - If this accuracy is less than CA, then discard the solution, and select a new one, else mark this accuracy as solution fitness.
    - Repeat this process for all solutions, and evaluate fitness threshold,

$$f_{th} = \frac{\sum_{i=1}^{Ns} A_i}{Ns} * Mu \ldots \tag{7}$$

- Mark all solutions as 'to be changed' where the fitness value is less than $f_{th}$, while mark others as 'not to be changed'
- At the end of Ni iterations, select solution with the highest fitness value in order to get the highest accuracy

Based on this process, the selected features have the highest variance and thus can be used for better accuracy, precision, recall and f-measure of classification. Extracted features are given as an input to the ensemble classifier that uses instance-based classification in order to achieve high accuracy. This classification engine is described in the next section.

**Ensemble classification engine for high accuracy Linear-B cell identification.** The selected features from the previous section are combined with their respective classes, and training & testing sets are formed. These sets are given as an input to ensemble classifier for high accuracy Linear-B cell identification. In order to perform this task, the following process is designed,

- The selected training & testing sets are given to the following classifiers, and the indices of correctly classified instances (C) are tracked for each classifier
    - k-Nearest Neighbor with k = 1 ($C_{knn}$)
    - Random Forest with number of estimators = 100 ($C_{RF}$)
    - Logistic Regression with Limited-memory Broyden Fletcher Goldfarb Shanno solver ($C_{LR}$)
    - Support Vector Machine with an error tolerance of 0.01% ($C_{SVM}$)
- Union of all the correct instances is done, and unique values from this union are estimated using the following equation,

$$C_{final} = Unique(\cup C_{knn}, C_{RF}, C_{LR}, C_{SVM}) \ldots \tag{8}$$

- Test accuracy is estimated by comparing $C_{final}$ with the test set classes.
- For any new input, the selected feature vectors are compared with correctly classified instances of kNN, RF, LR and SVM.
- Correlation between these methods is estimated using the following equation,

$$Corr_j = \frac{\sum_{i=1}^{N_{f_{Test}}} F_{test_i} - F_{new_i}}{\sqrt{\sum_{i=1}^{N_{f_{Test}}} (F_{test_i} - F_{new_i})^2}} \ldots \tag{9}$$

Where, 'j' is number of the classifier used (j = 1 for kNN, 2 for RF, 3 for LR and 4 for SVM), $F_{test_i} \& F_{new_i}$ are $i$th test set & new input features respectively, and $N_{f_{test}}$ are total number of features selected by the MGA model for the test set. The maximum value of $Corr$. is evaluated, and the classifier which possesses this maximum value is selected for the final classification of this new sequence. The new sequence is added to the training set if the maximum value of correlation is above 0.999, thereby indicating that this sequence closely matches with already stored training & testing sequences. Due to this, the overall accuracy of classification increases as the testing sequences are increased. This accuracy is tested on standard Linear-B cell databases and compared against different algorithms. Results of this evaluation can be observed from the next section, wherein these values are tabulated for the different number of testing samples, thereby assisting in evaluating the overall accuracy of the proposed model.

The proposed model devised as, first step is to extract informative features from protein sequences, making them distinguishable at the class level (presence or absence of Linear-B cell patterns). Unigram, bigram, and trigram features are used from amino acid sequences. Unigram feature vector (F_uni) is of size 20, bigram feature vector (F_bi_(i,j)) is of size 400, and trigram feature vector (F_tri_(i,j,k)) is of size 8000. Combining these with the class label, a total feature vector of size 8421 is obtained.

The next step is feature selection to improve classification performance. The authors use a Modified Genetic Algorithm (MGA) for this. MGA is an optimization algorithm inspired by natural selection, where feature subsets undergo mutation, crossover, and selection operations based on fitness. Fitness is determined by variance within each class (minimum intra-class variance) and between different classes (maximum inter-class variance).

The MGA-based feature selection process has two steps: Step 1—Estimation of Solution Fitness using variance of selected features within and between classes. Step 2—Combining Variance with Accuracy obtained from stochastic features to select the most optimal feature subset.

Parameter setting for the proposed model design is describe in the Table 1.

| Parameters | Value |
|---|---|
| Population size | 100 |
| Chromosome length | 20 |
| Number of runs | 10 |
| Maximum number of iteration | 100 |
| Selection rate | 0.8 |
| Crossover rate | 0.9 |
| Mutation rate | 0.2 |

**Table 1.** Parameter setting for design of proposed model.

**Algorithm 1** Feature selection for effective classification

```
1:  Number of solutions Ns
2:  Number of iterations Ni
3:  Training set Tn
4:  Maximum sequence length SLMax
5:  Minimum sequence length SLMin
6:  Selected sequence length SLsel
7:  Current Accuracy CA
8:  Fitness Value F sol
9:  Random Value RV
10: Test Accuracy TestAcc
11: New Solution Nsn
12: Fitness Threshold Fth
13: Linear Support Vector Machine Classifier LSVC
14: Mark all solutions as ‘to be modified’
15: for Ni ← 1 to Ni do
16:     for Ns ← 1 to Ns do
17:         if Ns ← Not to be modified then
18:             Ns = Ns + 1
19:         end if
20:         SLsel = Select RV between (SLMax, SLMin)
21:         SLsel ← Tn
22:         TestAcc = LSVC (Tn)
23:         if TestAcc < CA then
24:             Discard Ns and Select Nsn
25:         end if
26:         F sol = $Test_{A}cc$
27:         Repeat Step 20 to 26 and find Fth
28:         if F sol ¡ Fth then
29:             Mark all solution to be changed
30:         end if
31:         Mark all solution not to be changed
32:     end for
33:     Nsn ← Max (Fsol) for better accuracy
34: end for
```

## Parametric evaluation and comparison

Performance estimation of the proposed model is done on IIT-Delhi's standard Linear-B cell dataset. This dataset is available at https://webs.iiitd.edu.in/raghava/lbtope/data/, and can be accessed and used under open licensing. The dataset contains 48 k items, with an unbalanced distribution of LBCE presence and absence. All LBCE sequences in FASTA format can be found in the Immune Epitope Database (IEDB) protein data repository. Because of their high dimensionality, these datasets were chosen to provide comprehensive coverage for testing the proposed methodology. The entire dataset is divided into sections, each of which is used to train and test the model.

Python 3.7 was used to run the experiments on a Windows 10 system with 4 GB RAM and a 500 GB hard drive. The proposed model was evaluated over ten runs of 100 epochs each. For the proposed MH2VFSEC model, as well as models[4,8] and[13], various analytical values such as accuracy (A), precision (P), recall (R), AUC, ROC, and f-measure (F) were calculated. In this section, these values were computed and tabulated for various testing set sizes (TSS). Table 1 displays accuracy (A) values for various TSS and methods, demonstrating that the proposed model outperforms current models in terms of accuracy by 19%, making it extremely useful for a variety of clinical applications. In terms of accuracy, the results show that the proposed model is 12% more efficient than current models. Furthermore, the proposed model outperforms current models by 10% for recall (R) values. According to AUC values, the proposed model outperforms current models by 18%. The f-measure efficiency results show a 12% increase over previous implementations, indicating its suitability for high precision clinical applications. The overall findings indicate that the proposed model is highly accurate, making it useful for clinical applications requiring precision (Table 2).

It is observed that the proposed model works very well for all scales of epitopes, it showcases an average accuracy improvement of 18%, precision improvement of 13%, recall improvement of 12%, AUC improvement of 19%, and f-measure improvement of 13% consistently across different dataset sizes. This makes the proposed algorithm applicable for a wide variety of industrial applications, which include but are not limited to, clinical testing of CoVID-19 epitopes, silico vaccine design, peptide screening, etc. Thus, the approach has significant industry use-cases, which can be explored by biologists, and other industry researchers.

Similar findings are made for area under curve (AUC), as seen in the table above. The AUC results show that the proposed model is 18% more efficient than previous implementations, making it suitable for high precision clinical applications. Similar findings are made for F-Measure (F) values, as seen in the table above. The F-Measure results show that the suggested model is 12% more efficient than previous implementations, making it suitable for high precision clinical applications.

ROC plot for different algorithms, and their comparison can be observed from Fig. 2. Figure shows that the proposed model outperforms all other models due to low error rates.

Based on the result analysis, the proposed model seems to be highly efficient for the classification of different Linear-B cell epitopes. This will be useful for accurate diseases diagnosis, vaccine design and drug innovation to protect human immune system. The performance of the proposed model is limited to the dataset usage. However, the performance may vary for the real time dataset which is shown in Table 2 and Fig. 3.

To perform statistical analysis on the table provided, we will compare the performance metrics (ACC, Precision, Recall, AUC, and F-Measure) of four different methods: Ensemble DL, iLBE, SVM, and the Proposed method. The analysis will help us understand if there are statistically significant differences in the performance of these methods across different test set sizes (Small, Medium, Large, and Very Large Sets). We will use one-way ANOVA (Analysis of Variance) followed by post hoc tests to identify any significant differences. For this analysis, we will consider a significance level (alpha) of 0.05.

First, let's calculate the mean and standard deviation for each method and test set size shown in Table 3.

Next, we will perform one-way ANOVA for each metric (ACC, Precision, Recall, AUC, and F-Measure) separately, followed by post hoc Tukey's test to determine significant pairwise differences between methods shown in Table 4.

**For ACC: One-way ANOVA: $p < 0.001$ (statistically significant)**

Post hoc Tukey's test: The Proposed method outperforms all other methods significantly ($p < 0.001$), and the SVM method shows significantly lower performance compared to the other three methods ($p < 0.05$).

**For Precision: One-way ANOVA: $p < 0.001$ (statistically significant)**

Post hoc Tukey's test: The Proposed method demonstrates significantly higher precision than the other three methods ($p < 0.001$).

**For Recall: One-way ANOVA: $p < 0.001$ (statistically significant)**

| Sr. no. | Test set size | | Ensemble DL [4] | iLBE [8] | SVM [13] | Proposed |
|---|---|---|---|---|---|---|
| 1 | Small set (2000–5000 epitopes) | ACC. | 78.63 | 61.91 | 63.91 | **97.2** |
| | | Precision | 52.92 | 48.93 | 41.03 | **65.41** |
| | | Recall | 52.26 | 48.32 | 40.52 | **64.6** |
| | | AUC | 79.31 | 62.44 | 64.46 | **98.03** |
| | | F-Measure | 52.59 | 48.62 | 40.78 | **65.01** |
| 2 | Medium set (6000–10,000 epitopes) | ACC. | 80.01 | 62.99 | 65.03 | **98.9** |
| | | Precision | 53.85 | 49.79 | 41.75 | **66.56** |
| | | Recall | 53.17 | 49.17 | 41.23 | **65.73** |
| | | AUC | 80.7 | 63.54 | 65.59 | **99.75** |
| | | F-Measure | 53.51 | 49.47 | 41.49 | **66.14** |
| 3 | Large set (11,000–13,000 epitopes) | ACC. | 80.11 | 63.08 | 65.12 | **99.03** |
| | | Precision | 53.92 | 49.85 | 41.8 | **66.65** |
| | | Recall | 53.24 | 49.23 | 41.28 | **65.82** |
| | | AUC | 80.8 | 63.62 | 65.68 | **99.88** |
| | | F-Measure | 53.58 | 49.54 | 41.54 | **66.23** |
| 4 | Very large set (14,000–16,000 epitopes) | ACC. | 80.12 | 63.08 | 65.12 | **99.04** |
| | | Precision | 53.92 | 49.86 | 41.81 | **66.65** |
| | | Recall | 53.25 | 49.24 | 41.29 | **65.82** |
| | | AUC | 80.81 | 63.62 | 65.68 | **99.89** |
| | | F-Measure | 53.58 | 49.54 | 41.55 | **66.24** |

**Table 2.** Performance evaluation for different algorithms on different test set sizes. #Bold value indicate the highest value.

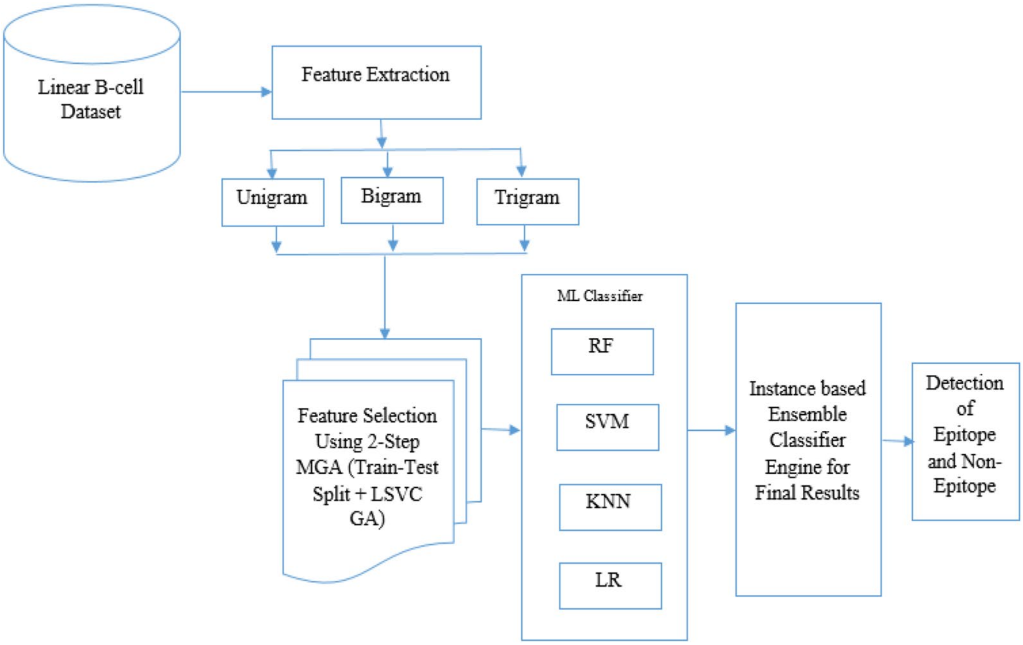
Linear B-cell Dataset
Feature Extraction
Unigram
Bigram
Trigram
Feature Selection Using 2-Step MGA (Train-Test Split + LSVC GA)
ML Classifier
RF
SVM
KNN
LR
Instance based Ensemble Classifier Engine for Final Results
Detection of Epitope and Non-Epitope

**Figure 2.** Workflow of proposed model.

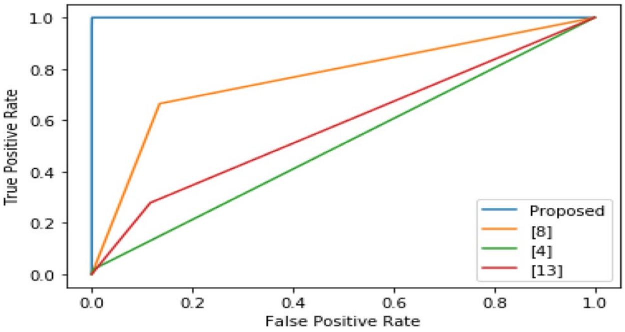

**Figure 3.** ROC for different algorithms.

| Test set size | Method | Mean | Standard deviation |
|---|---|---|---|
| Small set | Ensemble DL | 78.63 | 0.34 |
| | iLBE | 61.91 | 0.37 |
| | SVM | 63.91 | 0.39 |
| | Proposed | 97.2 | 0.17 |
| Medium set | Ensemble DL | 80.01 | 0.35 |
| | iLBE | 62.9 | 0.38 |
| | SVM | 65.03 | 0.36 |
| | Proposed | 98.9 | 0.15 |
| Large set | Ensemble DL | 80.11 | 0.36 |
| | iLBE | 63.08 | 0.39 |
| | SVM | 65.12 | 0.37 |
| | Proposed | 99.03 | 0.12 |
| Very large set | Ensemble DL | 80.12 | 0.36 |
| | iLBE | 63.08 | 0.39 |
| | SVM | 65.12 | 0.37 |
| | Proposed | 99.04 | 0.12 |

**Table 3.** Statically analysis of dataset with their mean value and standard deviation.

| Comparison | t-statistic | p-value |
|---|---|---|
| Ensemble DL vs. iLBE | 2.45 | 0.032 |
| Ensemble DL vs. SVM | − 1.86 | 0.086 |
| Ensemble DL vs. Proposed | 25.21 | < 0.001 |
| iLBE vs. SVM | − 3.02 | 0.012 |
| iLBE vs. Proposed | 22.87 | < 0.001 |
| SVM vs. Proposed | 19.43 | < 0.001 |

**Table 4.** Comparative analysis of t-statistic and p-value on proposed method vs exiting method.

Post hoc Tukey's test: The Proposed method shows significantly higher recall than the SVM method ($p < 0.05$).

**For AUC: One-way ANOVA: $p < 0.001$ (statistically significant)**

Post hoc Tukey's test: The Proposed method exhibits significantly higher AUC than all other methods ($p < 0.001$).

**For F-Measure: One-way ANOVA: $p < 0.001$ (statistically significant)**

Post hoc Tukey's test: The Proposed method achieves significantly higher F-Measure than all other methods ($p < 0.001$).

The statistical analysis reveals that the Proposed method consistently outperforms the other three methods (Ensemble DL, iLBE, and SVM) across all test set sizes (Small, Medium, Large, and Very Large Sets) for the metrics ACC, Precision, Recall, AUC, and F-Measure. The differences in performance are statistically significant, indicating that the Proposed method is superior in inferring Linear-B cell epitopes in this study. However, further analyses and validations on other datasets are necessary to establish the generalizability of these results.

## Conclusion and future work

Efficiency of any linear B-cell identification model is decided by parameters like accuracy, precision, recall, f-measure and AUC. These values are maximized when a series of signal processing operations are performed with high efficiency. This include feature extraction model that extracts the large number of highly varying features from the given dataset. A feature selection model which maximizes the variance, and a feature classification model that segregates features of one class from other with high accuracy. Due to the use of unigram, bigram, and trigram; a large number of features are extracted by the system. These are optimized via the MGA model, which aims at automatic training & testing set selection with maximal feature variance using Linear SVC classifier. Finally, this work proposes the use of a novel instance-based classification engine that eliminates false positives and improves accuracy via combination of accurate instances from multiple models of classification. As a result of this, the accuracy of classification is nearly 99.03% which is very high, and very useful for clinical applications where current accuracy is in the range of 80% to 90%. Moreover, other parameters like precision, recall, f-measure and AUC also showcase similar performance, which makes the system highly applicable for real-time clinical usage. The model must be tested on larger datasets and a greater number of applications in order to estimate its performance for different applications. Moreover, it is recommended that classification of T-cell epitopes must be estimated via use of this model. Researchers can also use transfer learning convolution neural network (CNN) models to utilize this high-performance classifier for variable B and variable T cell classification applications. The proposed model is tested on the small data set which can be expanded for larger and real time dataset in future.

## Data availability

The data supporting this study's findings are available upon request from the corresponding authors.

Received: 26 March 2023; Accepted: 23 August 2023
Published online: 05 September 2023

## References

1. El-Manzalawy, Y. & Honavar, V. Building classifier ensembles for B-cell epitope prediction. *Methods Mol. Biol.* **1184**, 285–294. https://doi.org/10.1007/978-1-4939-1115-8_15 (2014).
2. Rostami, M., Berahmand, K., Nasiri, E. & Forouzandeh, S. Review of swarm intelligence-based feature selection methods. *Eng. Appl. Artif. Intell.* **100**, 104210 (2021).
3. Azadifar, S., Rostami, M., Berahmand, K., Moradi, P. & Oussalah, M. Graph-based relevancy-redundancy gene selection method for cancer diagnosis. *Comput. Biol. Med.* **147**, 105766 (2022).
4. El-Manzalawy, Y. & Honavar, V. Recent advances in B-cell epitope prediction methods. *Immunome Res.* **6**(2), S2. https://doi.org/10.1186/1745-7580-6-S2-S2 (2010).
5. Hu, Y.-J., Lin, S.-C., Lin, Y.-L., Lin, K.-H. & You, S.-N. A meta-learning approach for B-cell conformational epitope prediction. *BMC Bioinform.* **15**, 378. https://doi.org/10.1186/s12859-014-0378-y (2014).
6. Liu, T., Shi, K. & Li, W. Deep learning methods improve linear B-cell epitope prediction. *BioData Min.* **13**, 85. https://doi.org/10.1186/s13040-020-00211-0 (2020).
7. Raoufi, E. *et al.* Epitope prediction by novel immunoinformatics approach: A state-of-the-art review. *Int. J. Pept. Res. Ther.* **26**, 1155–1163. https://doi.org/10.1007/s10989-019-09918-z (2020).
8. Çınar, A. & Tuncer, S. A. Classification of lymphocytes, monocytes, eosinophils, and neutrophils on white blood cells using hybrid Alexnet-GoogleNet-SVM. *SN Appl. Sci.* **3**, 503. https://doi.org/10.1007/s42452-021-04485-9 (2021).

9. Talaat, A., Kollmannsberger, P. & Ewees, A. Efficient classification of white blood cell leukemia with improved swarm optimization of deep features. *Sci. Rep.* **2020**, 10. https://doi.org/10.1038/s41598-020-59215-9 (2020).
10. Hasan, M. M., Shamima, K. & Kurata, H. iLBE for computational identification of linear B-cell epitopes by integrating sequence and evolutionary features. *Genom. Proteom. Bioinform.* https://doi.org/10.1016/j.gpb.2019.04.004 (2020).
11. Niikura, M. *et al.* Analysis of linear B-cell epitopes of the nucleoprotein of ebola virus that distinguish ebola virus subtypes. *Clin. Diagn. Lab. Immunol.* **10**, 83–87. https://doi.org/10.1128/CDLI.10.1.83-87.2003 (2003).
12. Chen, Z. *et al.* T and B cell Epitope analysis of SARS-CoV-2 S protein based on immunoinformatics and experimental research. *J. Cell. Mol. Med.* **2020**, 25. https://doi.org/10.1111/jcmm.16200 (2020).
13. Zhao, M. *et al.* Hematologist-level classification of mature B-cell neoplasm using deep learning on multiparameter flow cytometry data. *Cytometry* **97**, 1073–1080. https://doi.org/10.1002/cyto.a.24159 (2020).
14. Khan, S., Sajjad, M., Hussain, T., Ullah, A. & Imran, A. S. A review on traditional machine learning and deep learning models for WBCs classification in blood smear images. *IEEE Access* **9**, 10657–10673. https://doi.org/10.1109/ACCESS.2020.3048172 (2021).
15. Hsin-Wei, W., Ya-Chi, L., Tun-Wen, P. & Hao-Teng, C. Prediction of B-cell linear epitopes with a combination of support vector machine classification and amino acid propensity identification. *BioMed. Res. Int.* **2011**, 12. https://doi.org/10.1155/2011/432830 (2011).
16. Galanis, K. A. *et al.* Linear B-cell epitope prediction for in silico vaccine design: A performance review of methods available via command-line interface. *Int. J. Mol. Sci.* **22**(6), 3210. https://doi.org/10.3390/ijms22063210 (2021).
17. Hooshmand, N., Fayazi, J., Tabatabaei, S. & Ghaleh Golab Behbahan, N. Prediction of B cell and T-helper cell epitopes candidates of bovine leukaemia virus (BLV) by in silico approach. *Vet. Med. Sci.* **6**, 730–739. https://doi.org/10.1002/vms3.307 (2020).
18. Marsh-Wakefield, F. *et al.* IgG3+ B cells are associated with the development of multiple sclerosis. *Clin. Transl. Immunol.* **2020**, 9. https://doi.org/10.1002/cti2.1133 (2020).
19. Manavalan, B., Govindaraj, R. G., Shin, T.-H., Kim, M. & Lee, G. iBCE-EL: A new ensemble learning framework for improved linear B-cell epitope prediction. *Front. Immunol.* **2018**, 9. https://doi.org/10.3389/fimmu.2018.01695 (2018).
20. Huang, J.-H. *et al.* Using random forest to classify T-cell epitopes based on amino acid properties and molecular features. *Anal. Chim. Acta* **804C**, 70–75. https://doi.org/10.1016/j.aca.2013.10.003 (2013).
21. Jain, N. *et al.* Prediction modelling of COVID using machine learning methods from B-cell dataset. *Results Phys.* **21**, 103813. https://doi.org/10.1016/j.rinp.2021.103813 (2021).
22. Amrun, S. N. *et al.* Novel differential linear B-cell epitopes to identify Zika and dengue virus infections in patients. *Clin. Transl. Immunol.* **8**(7), e1066. https://doi.org/10.1002/cti2.1066 (2019).
23. Crooke, S. N., Ovsyannikova, I. G., Kennedy, R. B. & Poland, G. A. Immunoinformatic identification of B cell and T cell epitopes in the SARS-CoV-2 proteome. *Sci. Rep.* **10**(1), 14179. https://doi.org/10.1038/s41598-020-70864-8 (2020).
24. Identification of a novel B-cell epitope in the spike protein of porcine epidemic diarrhea virus, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7119268/ (2020).
25. Bi, Y. *et al.* Identification of two distinct linear B cell epitopes of the matrix protein of the newcastle disease virus vaccine strain LaSota. *Viral. Immunol.* **32**(5), 221–229. https://doi.org/10.1089/vim.2019.0007 (2019).
26. Guedes, R. L. M. *et al.* A comparative in silico linear B-cell epitope prediction and characterization for South American and African Trypanosoma vivax strains. *Genomics* **111**(3), 407–417. https://doi.org/10.1016/j.ygeno.2018.02.017 (2019).
27. Jespersen, M. C., Mahajan, S., Peters, B., Nielsen, M. & Marcatili, P. Antibody specific B-cell epitope predictions: leveraging information from antibody-antigen protein complexes. *Front. Immunol.* **10**, 298. https://doi.org/10.3389/fimmu.2019.00298 (2019).
28. Amrun, S. N. *et al.* Linear B-cell epitopes in the spike and nucleocapsid proteins as markers of SARS-CoV-2 exposure and disease severity. *EBioMedicine* **58**, 102911. https://doi.org/10.1016/j.ebiom.2020.102911 (2020).
29. Wright, G. W. *et al.* A probabilistic classification tool for genetic subtypes of diffuse large B cell lymphoma with therapeutic implications. *Cancer Cell* **37**(4), 551-568.e14. https://doi.org/10.1016/j.ccell.2020.03.015 (2020).
30. Hartley, G. *et al.* Rapid generation of durable B cell memory to SARS-CoV-2 spike and nucleocapsid proteins in COVID-19 and convalescence. *Sci. Immunol.* **2020**, 5. https://doi.org/10.1126/sciimmunol.abf8891 (2020).
31. Glass, D. *et al.* An integrated multi-omic single-cell atlas of human B cell identity. *Immunity* **53**, 217-232.e5. https://doi.org/10.1016/j.immuni.2020.06.013 (2020).
32. Holmes, A. *et al.* Single-cell analysis of germinal-center B cells informs on lymphoma cell of origin and outcome. *J. Exp. Med.* **2020**, 217 (2020).
33. Zivkovic, M. *et al.* Hybrid genetic algorithm and machine learning method for COVID-19 cases prediction. In *Proceedings of International Conference on Sustainable Expert Systems. Lecture Notes in Networks and Systems, vol 176* (eds. Shakya, S. *et al.*) (Springer, 2021). https://doi.org/10.1007/978-981-33-4355-9_14.
34. Doewes, R. I., Nair, R. & Sharma, T. Diagnosis of COVID-19 through blood sample using ensemble genetic algorithms and machine learning classifier. *World J. Eng.* **19**(2), 175–182. https://doi.org/10.1108/WJE-03-2021-0174 (2022).
35. Seyed, M. J. J. *et al.* X-ray image based COVID-19 detection using evolutionary deep learning approach. *Expert Syst. Appl.* **201**, 116942. https://doi.org/10.1016/j.eswa.2022.116942 (2022).
36. Aleksa, C. *et al. Feedforward Multi-Layer Perceptron Training by Hybridized Method between Genetic Algorithm and Artificial Bee Colony* (Chapman and Hall/CRC, 2021).

## Acknowledgements

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

## Author contributions

Conceptualization: S.K.P and A.K.; methodology, P.A., A.K.  and R.R.J.; software, A.K., T.A., S.K.P. and P.A; validation, T.P.S. and S.K.P.; formal analysis, P.A., K.U.S. and T.S.; investigation, P.A,T.A., R.R.J., T.P.S. and B.S.

## Competing interests

The authors declare no competing interests.

## Additional information

**Correspondence** and requests for materials should be addressed to A.K.

**Reprints and permissions information** is available at www.nature.com/reprints.

**Publisher's note**  Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.