Linear B-cell epitope prediction: a performance review of currently available methods

*Kosmas A. Galanis[1], Katerina C. Nastou[1], Nikos C. Papandreou[1], Georgios N. Petichakis[1] and Vassiliki A. Iconomidou[1,]**

[1]Section of Cell Biology and Biophysics, Department of Biology, School of Sciences,

National and Kapodistrian University of Athens, Panepistimiopolis, Athens 15701, Greece

Corresponding Author

*Assist. Prof. Vassiliki A. Iconomidou

Section of Cell Biology and Biophysics,

Department of Biology,

National and Kapodistrian University of Athens,

Panepistimiopolis, Athens 15701, Greece

Phone: +30 210 727 4871

Fax: +30 210 7274254

e-mail: veconom@biol.uoa.gr

ABSTRACT: Linear B-cell epitope prediction research has received a steadily growing interest ever since the first method was developed in 1981. B-cell epitope identification with the help of an accurate prediction method can lead to an overall faster and significantly cheaper vaccine design process. Consequently, several B-cell epitope prediction methods have been developed over the past few decades, but without significant success. In this study, we review the current performance and methodology, of some the most widely used linear B-cell epitope predictors: BcePred, BepiPred, ABCpred, COBEpro, SVMTriP, LBtope and LBEEP. Additionally, we attempt to remedy performance issues of the individual methods by developing a consensus classifier, that combines the separate predictions of these methods into a single output. The performance of these methods was evaluated using a large unbiased data set. All methods performed worse than documented in the original manuscripts, since all predictors performed marginally better than random classification against the test data set. While the method comparison was performed with some necessary caveats, we hope that this update in performance can aid researchers towards the choice of a predictor, whilst conducting their research. The necessary files for the execution of the consensus method that we developed can be found at http://thalis.biol.uoa.gr/BCEconsensus/ .

KEYWORDS: B-cell epitope, linear epitope, consensus prediction method, immunotherapy, vaccine development

KEY POINTS

- Review of the performance and methodology of currently available BCE predictors
- Design and development of consensus predictor
- Comparison of consensus with state-of-the-art BCE prediction methods
- Consensus method available at http://thalis.biol.uoa.gr/BCEconsensus/

Kosmas A. Galanis has a BSc in Biology and has performed his undergrad thesis in Bioinformatics. He is interested in the development of computational methods for protein function prediction.

Katerina C. Nastou is a Biologist with a PhD in Bioinformatics. Her research focuses on the study of biological networks, the computational prediction of protein function and biological database development.

Nikos C. Papandreou has a PhD in Biophysics and works as Special Laboratory Teaching Staff in "Bioinformatics-Biophysics" at the Department of Biology, National & Kapodistrian University of Athens.

Georgios N. Petichakis is a Computer Scientist with an MSc in Bioinformatics. His research focuses on the development of computational methods for the functional annotation of proteomes.

Vassiliki A. Iconomidou is an Assistant Professor of Molecular Biophysics and the group leader of the Biophysics and Bioinformatics Lab at the Department of Biology, National and Kapodistrian University of Athens.

Introduction

B-cell epitopes are regions on the surface of an antigen, where specific antibodies recognize and bind to, triggering the immune response. This interaction is at the core of the adaptive immune system, where among others is responsible for immunological memory and antigen-specific responses in vertebrates. The ability to identify these binding areas in the antigen's sequence or structure is important for the development of synthetic vaccines [1-3], diagnostic tests [4] and immunotherapeutics [5, 6]. Focus on these applications, through the lens of epitope discovery, has gained attention over the years, especially in regard to the safety benefits of synthetic vaccine development [7].

Generally, B-cell epitopes are divided into two categories: linear (continuous) epitopes, that consist of a linear sequence of residues and conformational (discontinuous) epitopes, that consist of residues that are not contiguous in the primary protein sequence, but are brought together by the folded protein structure [8]. Moreover, about 90% of B-cell epitopes have been estimated to be conformational and only about 10% to be linear [9]. Nonetheless, it has been shown that many discontinuous epitopes contain several groups of continuous residues that are also contiguous in the tertiary structure of the protein [10], making the distinction between them unclear.

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 ever-growing 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 [12-14] (Table 1). Despite the 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 [15]. Thus, in this review we will discuss solely the performance of linear B-cell epitope (BCE) predictors.

In most cases, the algorithms that predict BCEs can either be sequence-based and/or structure-based. Most predictors utilize only data derived from the protein sequence of the antigen and thus are sequence-based, while structure-based predictors utilize only an

antigen's 3D structure. Furthermore, some hybrid methods employ both approaches for better predictive performance [16, 17]. Historically, initial attempts at predicting epitopes made use of a single amino acid propensity scale, assigning each amino acid a numerical value, followed by a local averaging of these values along the peptide chain. The first method, implementing this approach, was published by Hopp and Woods [18] in 1981, and it utilized Levitt's hydrophilicity scale [19]. Aside from hydrophilicity, which was utilized again in another scale by Parker *et al.* [20], other amino acid properties were explored in later methods, such as antigenicity [21], flexibility [22], surface accessibility [23], and turns [24]. The next wave of predictors built upon this development, when methods like PREDITOP [25], PEOPLE [26], BEPITOPE [27] and BcePred [28], combined multiple physicochemical properties. Although these methods represented the best attempts yet at predicting epitopes, Blythe and Flower [29] demonstrated that the performance of such methods was overstated. They did a thorough assessment of 484 amino acid propensity scales in combination with information on the location of epitopes for 50 known proteins and found that even the best possible combination of scales performed only slightly better than random [29]. In their work they also correctly suggested, that more advanced approaches for predicting linear B-cell epitopes needed to be developed, such as methods that employ artificial intelligence technology.

As anticipated, given the booming of available biological data, the entire next generation of methods utilized some form of machine learning models. One of the first such approaches was BepiPred [30], that combined a Hidden Markov Model (HMM) with an amino acid propensity scale. Additionally, other machine learning models were used in methods developed afterwards, including Neural Networks in ABCpred [31], a Naïve Bayes classifier in Epitopia [32] and Support Vector Machines (SVMs) in most of the recent predictors. SVM-based predictors dominated the machine learning approaches used in BCE prediction, each one differing from the other on feature selection, data set curation and SVM specific parameters (Table 1). The BCPred [33] and FBCPred [34] methods published in 2008, predict fixed linear B-cell epitopes and flexible length linear B-cell epitopes respectively, utilizing SVM models with the subsequence kernel. The AAPPRED [35] method also utilizes SVM models trained on the frequency of Amino Acid Pairs (AAP), a scale first developed by Chen *et al.* [36]. Other notable approaches include: BayesB [37], LEPS [38] and BEORACLE [39]. A new machine learning approach that was developed in 2014, called EPMLR [40], utilizes multiple linear regression for epitope classification. Another recent

novel approach is the DMN-LBE [41] method, that was developed using deep maxout networks, a type of deep neural network with a different activation layer called maxout. The DRREP [42] method was published in 2016, and it also utilizes deep neural network technology to extrapolate structural features related to epitopes from protein sequences. One of the latest additions is the second version of the BepiPred method, BepiPred-2.0 [17], that was developed in 2017. This method is based on a random forest algorithm and differs from its predecessor in that it was trained only on epitope data derived from crystal structures. Another promising algorithm is iBCE-EL [43], which is an ensemble learning framework combining Extremely Randomized Tree (ERT) and Gradient Boosting (GB) classifiers.

Table 1. Linear B-cell epitope predictors in chronological order, alongside a short description of their methodology, their current status and their web page. After researching the relevant publications, we gathered up all the linear B-cell epitopes predictors we could find in this fairly complete, but not exhaustive catalogue. For every method we reference the source material to determine their methodology, which we have summed up for each predictor in a short description. For every predictor we also checked their availability status, as of writing this review, and categorized them regarding their general and current availability online as tools, as well as their obtainability as standalone software packages. In the last column, we provide the website links for each method, when available.

Predictor	***Description***	***Status***	***Link***
Antigenic [21]	Physico-chemical propensity scales, occurrence of residues	Not currently available online	http://www.emboss.bioinformatics.nl/cgi-bin/emboss/antigenic
PEOPLE [26]	Physico-chemical propensity scales	Not available online	-
BEPITOPE [27]	Physico-chemical propensity scales	Freely available online	http://bepitope.ibs.fr/
BcePred [28]	Physico-chemical propensity scales	Freely available online and downloadable	http://crdd.osdd.net/raghava/bcepred/index.html
BepiPred-1.0 [30]	HMM & Parker hydrophilicity scale	Freely available online and downloadable	http://www.cbs.dtu.dk/services/BepiPred-1.0/
Söllner [44]	Physicochemical Properties, MOE, KNN, Decision Tree	Not available online	-
Chen [36]	SVM & AAP	Not available online	-
ABCpred [31]	Neural networks (feed forward & reccurent)	Freely available online and downloadable	http://crdd.osdd.net/raghava/abcpred/index.html

Predictor	*Description*	*Status*	*Link*
BCPREDS [33, 34]	SVM	Not currently available online	http://ailab.ist.psu.edu/bcpred/
AAPPred [35]	SVM & AAP	Freely available online and downloadable	http://www.bioinf.ru/aappred/predict
Epitopia [32]	ML algorithm trained to discern antigenic features	Freely available online and downloadable	http://epitopia.tau.ac.il/index.html
COBEpro [16]	SVM	Freely available online and downloadable	http://scratch.proteomics.ics.uci.edu/
BayesB [37]	SVM	Not currently available online	http://immunopred.org/bayesb/index.html
LEPS[38]	SVM & Physicochemical propensity scales & AAS	Not currently available online	http://leps.cs.ntou.edu.tw/
BEOracle [39]	SVM	Not available online	-
BEST [45]	SVM	Not available online	-
SVMTriP [46]	SVM	Freely available online and downloadable	http://sysbio.unl.edu/SVMTriP/
BEEPro [47]	SVM & Physicochemical propensity scales & PSSM	Not available online	-
LBtope [48]	SVM & Physicochemical propensity scales & AAP	Freely available online and downloadable	http://crdd.osdd.net/raghava/lbtope/protein.php
Random Forest [49]	Amino acid descriptors & Random Forest	Not currently available online	http://sysbio.yznu.cn/Research/Epitopesprediction.aspx
EPMLR [40]	Multiple Linear Regression	Not currently available online	http://www.bioinfo.tsinghua.edu.cn/epitope/EPMLR/
DMN-LBE [41]	Deep Maxout Networks	Not currently available online	http://bioinfo.tsinghua.edu.cn/epitope/DMNLBE/
LBEEP [50]	DDE - SVM	Freely available download	https://github.com/brsaran/LBEEP
APCpred [51]	APC & SVM	Not currently available online	http://ccb.bmi.ac.cn/APCpred/
DRREP [42]	Deep Ridge Neural Network	Not currently available online	https://github.com/gsher1/DRREP
BepiPred-2.0 [17]	Random forest algorithm trained on epitopes derived from crystal structures	Freely available online and downloadable	http://www.cbs.dtu.dk/services/BepiPred/
iBCE-EL [43]	Ensemble framework combining ERT & GB	Freely available online	http://thegleelab.org/iBCE-EL/

Here, we review the performance of some of the most widely used linear B-cell epitope predictors currently available, namely BcePred, BepiPred, ABCpred, COBEpro, SVMTriP, LBtope and LBEEP. We also examine the performance of a consensus classifier combining

these methods, to test whether a consensus approach can boost predictive performance[52-54]. Finally, we compare the performance of all these classifiers and the consensus method we developed against one of the most recently published BCE predictors, BepiPred-2.0. This review aims to give non-expert researchers an overview of available linear BCE predictors, as well as an update in their current performance and availability, which they can use to quickly locate them and choose the appropriate tools for their research work. Moreover, we have created contemporary non-redundant datasets of linear BCEs that could aid both experimental researchers as well as bioinformaticians actively working in the field of algorithm development.

Materials and Methods

Selection of suitable linear B-cell epitope predictors

The first priority of this work was to gather and test as many individual predictors as possible. However, the scope of methods that were to be tested could not be limitless, and thus some criteria for their selection were applied. At first, we decided to catalogue all available B-cell epitope predictors (Table 1). This is when we first noticed an alarming trend; where many of the online tools of the predictors that we looked up were either offline for some hours during the day or – even worse –completely unreachable. Furthermore, even when operational, most prediction servers have limitations on the amount of sequences and the workload they can process. Considering the present issues and the future problems that might arise, we decided to resort only to methods that were available as standalone software, that became our main criterion. The second criterion was that methods should be usable via the command line and not only through a Graphical User Interface (GUI) and the third criterion was that each method's way of operation should be somewhat comparable and in tune with the rest of the available predictors. Out of the many methods that have been developed through the years (Table 1), seven were selected for testing: **BcePred** [28], **BepiPred** [30], **ABCpred** [31], **COBEpro** [16], **SVMTriP** [46], **LBtope** [48] and **LBEEP** [50] (Table 1). During our study the second version of BepiPred was released, and its comparison with the rest of the methods and our decision not to utilize it in the development of the consensus method is discussed later in this article.

BcePred was published in 2004 by Raghava *et al*. [28], 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 [55], using a combination of flexibility, hydrophilicity, polarity and surface accessibility propensity scales.

BepiPred was developed in 2006 by Lund *et al*.[30], and it is the first ever method that utilizes an HMM. The HMM was trained using a data set derived from the database Antijen [56] 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.

ABCpred was created in 2006 [31], again by the Raghava group and it was the first test case of a more sophisticated machine learning model. It is based on a Recurrent Neural Network (RNN), that was trained using a variety of different window sizes and hidden units. The window sizes that were tested, were 10, 12, 14, 16, 18 and 20. Thus six models were developed in total, with the window size of 16 amino acid residues achieving the highest accuracy of 65.93% and a Matthews Correlation Coefficient (MCC) of 0.3187, after five-fold cross-validation on a data set derived from Bcipep [55].

COBEpro was published in 2009 by Baldi *et al. [16]* at the University of California. This method utilizes a novel two-step system for the prediction of both linear and discontinuous B-cell epitopes. Firstly, it utilizes an SVM model to assign an epitopic propensity score to fragments within the given peptide sequence. Additionally, COBEpro is able to incorporate into the SVM model the provided or predicted secondary structure and solvent accessibility of the given sequence, that are predicted by SSpro [57] and ACCpro [58] respectively. During the second stage, the method calculates an epitopic propensity score for each amino acid, based on the previous scores assigned by the model in the first stage. Among others, this predictor was tested on the fragmented version of Chen's [36] data set, achieving an AUC of 0.829 and an accuracy of 78%.

SVMTriP was developed in 2012 [46] and it is an application of an SVM model that employs tri-peptide similarity calculated through the Blosum62 matrix in combination with amino acid propensity scales. Its prediction suite comes with six different models corresponding to window sizes of 10, 12, 14, 16, 18 and 20 of which the 20 amino acid residue model performed the best with a reported 80.10% sensitivity and 55.20% precision on a data set gathered from the Immune Epitope Data Base (IEDB) [59].

LBtope was the most recent effort, out of our selected predictors,on epitope prediction published by Raghava's lab in 2013. This method uses, among other previously used types of features, a modified AAP profile from Chen's method [36]. These profiles are used to convert the input sequence into numerical features that are then used as input for an SVM model that predicts epitopes. LBtope was trained and tested on a data set collected from IEDB, which comprised of experimentally verified epitopes and non-epitopes, in contrast to

previous methods that used random peptides as non-epitopes. Its reported performance on different data sets varied significantly, with an accuracy ranging from 51.57% to 85.74%.

LBEEP was developed in 2015 by Saravan *et al. [50]* from the University of Madras in India. In this work, a novel amino acid feature descriptor called Dipeptide Deviation from Expected Mean (DDE) was developed, in an attempt to distinguish linear epitopes from non-epitopes. This new descriptor was then implemented with both SVM and AdaBoost-Random Forest machine learning techniques. The method was trained for window sizes of 5-15 amino acids and it achieved an accuracy between 61% and 73%, after five-fold cross-validation, on a data set derived from IEDB [60].

Once all methods were installed in a local unix-based machine, their output was validated by comparing example sequences of the local versions of software with the corresponding online tools. Additionally, all methods used in this analysis, had their threshold set on its default value except for BcePred and COBEpro (Table 2). In the case of BcePred the default threshold value of the method used, which combined the results of four different propensity scales, was decreased from 2.38 to 2. This decrease was decided after extensive testing, because the default threshold value proved to be extremely high. Nevertheless, it should be noted that the new value used agreed with the default threshold currently used by both the online and the local version of the method, in contrast with the one reported in the initial publication. COBEpro on the other hand didn't have a default threshold value, since its results are printed out in a graph where epitopic propensity is given a relative positive or negative score for each position of the query protein. The threshold value that was chosen for this method was that of four positive votes above the baseline score of zero, because it yielded the best results during testing.

Table 2. A summary of methods, threshold values and modifications applied to each predictor. Each predictor first had its best performing mode selected and its threshold value set to a specific value shown in the table, using the criteria described in the manuscript.

Predictor	***Threshold***	***Mode***	***Threshold Type***
BcePred	2	Combined	Not Default
BepiPred-1.0	0.35	BepiPred	Default
ABCpred	0.51	20	Default
COBEpro	4	-	Not Default
SVMTriP	0.2	20	Default

LBtope	0.6	-	Default
LBEEP	0.6	Balanced	Default

Development of the consensus method

A consensus method was developed to incorporate all available methods that were selected in the first stage and is available at http://thalis.biol.uoa.gr/BCEconsensus/ as a standalone application. Instructions on how to install and use the consensus method can be found both in the web page where it is hosted and in **Supplementary File 1**. All sequence-based methods can be divided into two categories based on their classification approach. The first category comprises of the methods that assign an epitopic propensity score to each residue of the provided sequence. Four methods are included in it; BcePred, BepiPred, COBEpro and LBtope. The second category comprises of the methods that classify peptides within certain length sizes as epitopes or non-epitopes and such methods are ABCpred, SVMTriP and LBEEP. The two categories are summarized in Table 3.

Table 3. Input window sizes and prediction approach of each method. The classification of query proteins as epitopes can generally be performed in either a "per residue" or a "per peptide" basis. In the "per residue" methods each separate residue of a protein is assigned an antigenicity score, while in the "per peptide" methods, a prediction is limited within fixed windows sizes.

Predictor	***Prediction***	***Window Size***
ABCpred	Per peptide	10, 12, 14, 16, 18, 20
SVMTriP	Per peptide	10, 12, 14, 16, 18, 20
LBEEP	Per peptide	5 - 15
BcePred	Per residue	-
BepiPred-1.0	Per residue	-
COBEpro	Per residue	-
LBtope	Per residue	-

The methods that predict per peptide, ABCpred and SVMTriP, use predetermined fixed window sizes. Thus, it was necessary to choose a window size where these methods would operate sufficiently well, both in individual testing and as part of the consensus classifier.

The window size chosen for these methods after initial testing was that of 20 residues. The main reasons were the better reported performance of SVMTriP at that window size and the lack of any default threshold values for the rest of the models in the documentation. As far as ABCpred is concerned, the performance penalty of selecting a window size of 20 instead of the reported best of 16 residues was minor. It should also be noted that initial testing for LBEEP at a window size of 20 was experimental, since the method was trained using only epitopes of lengths between 5 and 15, and thus any results outside that range were unreliable.

Once a window size of 20 was selected for the "per peptide" methods, an effective strategy had to be formulated where the two different categories of output would produce a single consensus result. The solution was a consensus voting system that classifies a residue as belonging to an epitope when a predetermined threshold of votes has been achieved. When a "per residue" method classifies a residue of the query sequence as "epitopic" it counts as one positive vote, while when a "per peptide" method classifies a fragment of a protein as an epitope each amino acid of that peptide is classified as "epitopic". So, when the sum of positive votes for a given position of a query sequence surpasses the threshold of the consensus classifier, that residue is marked as part of an epitope. The consensus threshold chosen, after testing, is defined as the hit overlap of at least half out of "n" selected methods, where "n" is the number of methods embedded in the algorithm [61]. The consensus method accepts protein sequences, of a length of 20 amino acid residues or higher, in FASTA format as input. The workflow of the consensus method is shown in Supplementary File 2.

For testing purposes, a slightly different architecture of the consensus method was implemented, that specialized in rapid consensus output on our fixed length data sets (Figure 1). All methods – including the consensus – were mainly tested on a data set consisting of peptides with a length of 20. To resolve this issue, two parallel approaches were explored. In the first approach, all methods were included, and each method predicted whether an entire peptide is an epitope or not. However, in order for the results between the "per peptide" and "per residue" methods to be comparable, since only "per peptide" methods classify protein fragments, it was accepted that when "per residue" methods have predicted half or more of a peptide's fragments as "epitopic", then the whole peptide too is a predicted epitope. Such caveats are generally found in other forms of predictors of biological nature[62, 63], and thus were chosen in our evaluation approach, as well. In the second approach, only "per residue" methods were included, and the consensus result was simply, a combination of only those predictions.

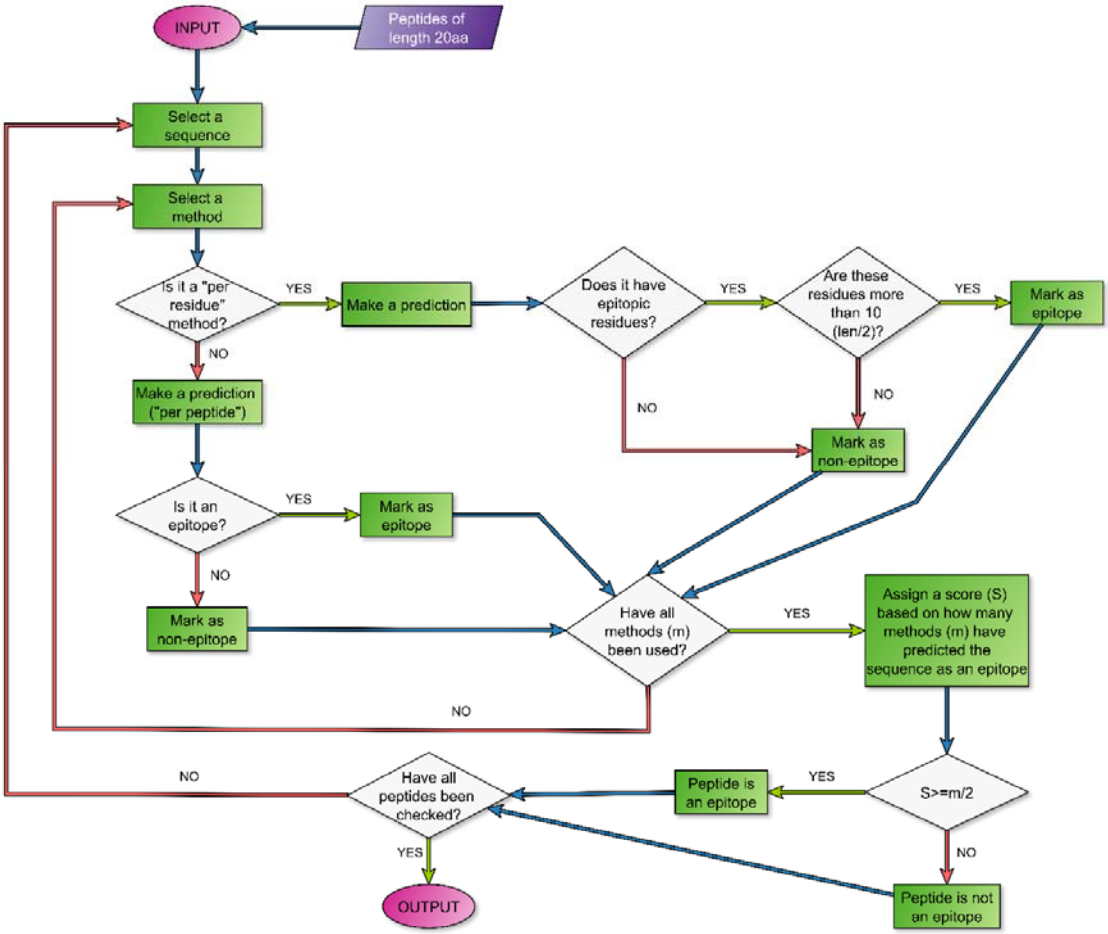


Figure 1. The workflow of the validation experiment performed to assess the performance of the consensus predictor during testing. A data set of multiple fixed length peptides of 20 amino acid residues in FASTA format is used as input. For each peptide a separate prediction is made by every individual method. "Per residue" methods predict a peptide as an epitope, when 10 more of its residues are predicted as epitopic, by classifying the entire peptide as an epitope or a non-epitope. On the other hand, "per peptide" methods classify the whole peptide as either an epitope or a non-epitope. The scores of these predictions are then summed using a vote system and the algorithm checks the score against the consensus threshold. Each separate peptide can be ranked from a minimum of 0 votes to a maximum of "m" votes, where "m" is the number of predictors embedded in our consensus method. If the score of a peptide is greater or equal than the number of half the methods used "m", then the peptide is classified as an epitope, otherwise the peptide is classified as a non-epitope. Once all peptides of a given data set have been parsed, the prediction results for all of them are printed out in a single text file.

Data sets

Typically, the development of machine learning classifiers requires a training data set and a test data set, but since all the predictors tested in this work were previously developed, only the latter was necessary. However, due to the fact that the individual training data sets for each predictor contained a significant number of overlapping sequences, gathered from a select few databases (like IEDB and Bcipep), their inclusion in our test data set would introduce bias in the results. So, in order to test all the different methods in an unbiased manner, the positive and negative training data sets for each method were gathered from their respective publications and webpages. As shown in Table 4, the positive training data set for the majority of predictors comprises of all available BCEs from a given database, while the negative set contains random amino acid sequences from Swiss-Prot [64]. The way the negative set of control data is constructed, changed in algorithms developed after 2012 to include only sequences from confirmed non-epitopes, as is the case for SVMTriP, LBtope and LBEEP. This change was introduced in order to improve the ability of prediction algorithms to effectively distinguish "epitopic" from random sequences, as it had been previously proposed [65].

Table 4. A summary of the source of positive and negative data sets for each predictor. For every predictor a database had to be used to construct its training data sets, which are comprised of a positive and a negative subset of data. In this table, we outline the database or curated data set from which each method sourced its training data set, along with the date that the data was obtained. The date could be used to determine the snapshot of the data, which could have been obtained for each predictor's training, allowing us to determine possible overlaps of our testing data set with the relevant training data.

Predictor	***Positive***	***Negative***
BcePred	BCIPEP (2004)	1029 random sequences
BepiPred-1.0	HIV/PELLEQUER/ANTIJEN	Not described in the original publication
ABCpred	BCIPEP (2006)	700 random sequences
COBEpro	HIV/PELLEQUER	HIV/Pellequer non-Epitopes
SVMTriP	IEDB (2012)	4925 IEDB non-epitopes
LBtope	IEDB (2012)	IEDB (2012) non-epitopes
LBEEP	IEDB (2015)	IEDB (2015) non-epitopes

While developing the consensus algorithm, a new version of BepiPred was published called BepiPred-2.0 [17]. Even though the method itself wasn't utilized in the development of the consensus method, its curated publicly available data set of linear epitopes was used as the source for this work's data sets. This data set represents the biggest collection of linear epitope and non-epitope data used for the development of a prediction method to date, as IEDB is the largest and most frequently updated epitope database [60]. The BepiPred-2.0 data set was created by procuring from this database, all available epitopes (positive assay results) and non-epitopes (negative assay results), which were confirmed as such from two or more separate experiments. Afterwards, all peptides with a length smaller than 5 and longer than 25 residues were removed from the data set, because epitopes are rarely found outside this range [66]. Any epitopes that were found both in the positive and negative subsets were also removed. The resulting data set contains 11834 epitopes in the positive subset and 18722 non-epitopes in the negative subset. Aside from its curation, a useful feature of this data set was the mapping of all epitopes and non-epitopes on their respective parent protein sequence. This made extending each epitope to a desired length much easier.

The predictors that used IEDB as their source of epitope data are SVMTriP, LBtope and LBEEP (Table 4). In order to produce an unbiased data set, their data sets were compared with BepiPred-2.0's data set and all the matching peptides were removed. This resulted in our first data set, named Consensus_Redundant (Consensus_R) which comprises of 7675 epitopes and 15617 non-epitopes. Using this data set as the source, a second non-redundant data set was constructed, by clustering peptides with the online tool CD-HIT [67]. All parameters were set to default and the sequence identity cut-off was set to 0.6 or 60%, as previously done in LBEEP's data set creation [50]. The resulting data set was named Consensus_Non_Redundant (Consensus_NR) and it includes 4286 epitopes and 5266 non-epitopes. By creating the Consensus_NR data set in this manner, we essentially made the largest non-redundant data set possible, which contained known sequences that none of predictors had previously "seen". Additionally, from the Consensus_NR data set a subset was extracted, containing 552 epitopes and 480 non-epitopes with a peptide length of exactly 20 amino acids, which was named Consensus_NR_exact. This subset was used to test the performance of predictors using only true epitopes and not epitope containing regions that result from the extension-truncation technique. A summary of all data sets used in this study is presented in Table 5, while the complete data sets are provided in Supplementary Table 1.

Table 5. A summary of test data sets utilized in this study. The counts of positive and negative subsets of data used in each of the three data sets developed for method testing is shown.

Data set	*Epitopes*	*Non-Epitopes*
BepiPred-2.0*	11814	18689
Consensus_R	7675	15617
Consensus_NR	4286	5266

*A slightly modified version of BepiPred-2.0's data set was used, which had a few epitopes removed because their sequence of origin was shorter than 20 amino acid residues and thus the epitope couldn't be extended to the desired length.

Each data set used for testing contained peptides modified beforehand into fixed length patterns using the technique of sequence extension and truncation, employed in previous methods [12, 16, 31, 41]. This was done to accommodate the fixed size input methods and thus included only their corresponding input lengths, namely 10, 12, 14, 16, 18 and 20 residues. For example, for a window size of 20, any epitopes or non-epitopes that were longer than 20 amino acids were shortened from both sides to have the desired length. Moreover, peptides with a length shorter than 20 residues were extended sideways on their parent sequence up to the desired length. The primary input size that was tested in this study was that of 20 residues for performance reasons as described in the development of the consensus algorithm. However, preliminary testing was also performed on a length of 16 residues, after analyzing the distribution of epitope lengths in the BepiPred-2.0 data set (Figure 2). The mean peptide length of the data set was about 14 and the median value 15, which coincides with previous research on the characteristics of epitopes [66].

The workflow used to create the non-redundant data sets is shown in **Supplementary File 2** and all data sets referenced in this section can be downloaded from this web page http://thalis.biol.uoa.gr/BCEconsensus/

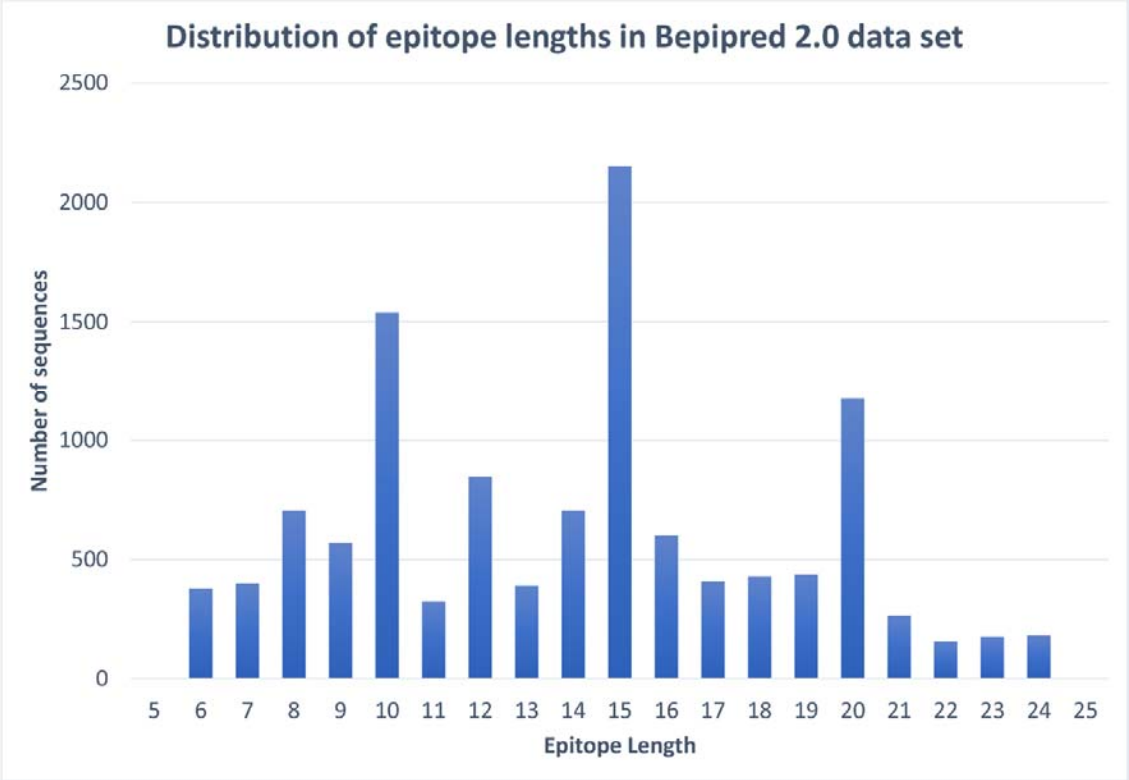


Figure 2. Distribution of epitope lengths in BepiPred-2.0 data set. The number of sequences is shown in the vertical axis, while the epitope length is shown in the horizontal axis. The most frequent epitope lengths are 10, 15 and 20 amino acid residues. The mean peptide length is 14, while the median value is ca. 15 residues.

Performance measures

Generally, to evaluate a classifier's performance both threshold dependent and independent metrics are used. The main threshold independent metric used in such cases is the AUC of the ROC curve. This metric was suggested as the preferred metric for benchmarking epitope prediction performance at a workshop by Greenbaum *et al.* [65] and thus, it grew to become a standard in the BCE prediction field. However, because all the predictors that we examined were already fully developed and their optimal thresholds set, it didn't make sense to use such a metric in our testing, since no model training was performed. For that reason, only threshold dependent metrics were employed, namely Sensitivity (SN), Specificity (SP), Accuracy (ACC) and MCC. Out of these metrics, significant attention was given to MCC, since it is generally regarded as the best performance metric for binary classifiers [68, 69]. The coefficient's value can range from -1 to +1, were the maximum value represents a perfect

prediction and the minimum a total disagreement between predictions and observations. When the coefficient's value is zero it indicates a prediction that is no better than random. Aside from the known value in accessing performance utilizing the MCC and accuracy metrics, regarding the other metrics, more importance was attached to sensitivity rather than specificity. Sensitivity indicates how effectively a predictive method manages to successfully locate areas that are actual epitopes, in contrast to specificity, which measures how effectively a predictive method manages to locate the sites that are not epitopes. In this study, the correctly predicted epitopes or "epitopic" residues were considered True Positive (TP), whereas the correctly predicted non-epitopes or "non-epitopic" residues were characterized as True Negative (TN). Conversely, the respective false predictions were defined as False Positive (FP) and False Negative (FN), respectively.

RESULTS AND DISCUSSION

As mentioned in the *Methods* section, two approaches are followed to evaluate all predictions made by the consensus algorithm. In the first approach results from all methods are incorporated in the consensus method — both those predicting in a "per residue" and in a "per peptide" manner — while in the second approach the consensus prediction only utilizes the "per residue" methods. Two different versions of the consensus algorithm were created in the "per peptide" mode, as seen in Table 6; one which includes all predictors and one which utilizes all of them except for LBEEP. This was done after noticing that LBEEP performs much worse, compared to the rest of the predictors. This performance issue can be partly attributed to the fact that the optimal prediction window of 5-15 residues for LBEEP is different than the 20-residue length that was used for our testing purposes (Supplementary File 2).

The evaluation of the predictors' performance was done primarily by measuring their MCC values, while secondary importance was assigned to achieving higher accuracy, and sensitivity. Sensitivity was considered more important than specificity for this particular application, since a BCE predictor's primary goal is to find possible BCEs in unknown sequences. Naturally, sensitivity and specificity are not mutually exclusive entities, yet in this study optimal sensitivity is preferred to optimal specificity.

Performance of all predictors on Consensus_NR

The results regarding the "per peptide" approach (Table 6) show that the highest MCC value was achieved by the BepiPred method with 0.0778, followed by our Consensus_NoLBEEP algorithm — the one without LBEEP — that achieved an MCC of 0.0721. Moreover, LBEEP had the lowest MCC (-0.0103), while BcePred and SVMTriP also scored low (0.0251 and 0.0290, respectively). The highest accuracy was achieved by our Consensus_ALL method with 55.59%, which was marginally better than those of SVMTriP and BcePred. SVMTriP had the best specificity out of all the methods (85.87%), followed by LBEEP and BcePred. Additionally, the ABCpred method achieved the greatest sensitivity with 66.44%, and COBEpro achieved the second highest with 58.63%. The Consensus_NoLBEEP algorithm achieved values close to the best for both MCC and accuracy, and also had a relatively improved MCC and a significantly increased sensitivity compared to its first version.

Table 6. Performance of all predictors in "per peptide" mode. The methods are tested against the Consensus_NR data set.

Predictor	*SN%*	*SP%*	*ACC%*	*MCC*
Consensus_noLBEEP	48.39	58.81	54.14	**0.0721**
Consensus_ALL	27.15	78.73	55.59	0.0687
BcePred	22.21	79.85	53.99	0.0251
ABCpred	66.44	36.9	50.16	0.0348
LBtope	45.91	58.94	53.1	0.0488
BepiPred-1.0	49.95	57.84	54.3	**0.0778**
COBEpro	58.63	45.67	51.49	0.0431
SVMTriP	16.21	85.87	54.62	0.0290
LBEEP	19.06	80.12	52.72	-0.0103

In the case of the "per residue" approach (Table 7), the consensus method (Consensus_RES) achieved the best MCC with 0.489, while BepiPred scored marginally worse with 0.0488. The same pattern was also observed for accuracy, where the Consensus_RES method scored 53.04% and BepiPred 52.88%. The greatest sensitivity was achieved by COBEpro with 49.27%, while BepiPred was again second best with 48.12%. The worst performance regarding MCC was attained by BcePred and COBEpro with scores of 0.0154 and 0.0175 respectively. Overall, despite the slight improvement in MCC and accuracy, the performance of the consensus algorithm was not significantly better in any of the statistical measures examined in the second part of the results.

Table 7. Performance of "per residue" predictors. The methods are tested against the Consensus_NR data set.

Predictor	*SN%*	*SP%*	*ACC%*	*MCC*
Consensus_RES	46.64	58.24	53.04	0.0489
BcePred	29.18	72.21	52.9	0.0154
LBtope	45.56	57.47	52.13	0.0304
BepiPred-1.0	48.12	56.76	52.88	0.0488
COBEpro	49.27	52.49	51.05	0.0175

When comparing the results of the two approaches only minor differences in performance are observed between the two modes of prediction for the four "per residue" methods. Generally, we notice a slight decrease in MCC from a maximum of 0.0778 in the first approach to a

maximum of 0.0489 in the second, while accuracy is comparatively worse on average. Out of the "per residue" methods BepiPred comes on top in both approaches in MCC and accuracy. The Bcepred method appears to perform relatively worse than the rest in both groups with the lowest MCC in both cases, whereas the COBEpro method performs relatively better in its "per peptide" iteration, with an average MCC score in the first part but a poor score in the second segment of the results. Moreover, in both approaches our consensus algorithm doesn't significantly outperform the rest of the predictors and only achieves a performance which is quite similar to that of BepiPred.

In summary, we observe that in all cases: MCC values are less than 0.1, accuracy is ranging from 50% to 55%, there are relatively high specificity values in certain cases such as SVMTriP and BcePred, and sensitivity values are low. Aside from our consensus methods, the best performers were LBtope and BepiPred and the worst ABCpred and LBEEP, which also displayed the lowest MCC scores.

Using the Consensus_NR data set we implemented many iterations of the consensus method utilizing many different method combinations, in order to find the optimum. As expected, LBEEP's presence undermined the consensus predictor's performance and it was therefore omitted from the final version (Consensus_NoLBEEP) and any further testing in the 20-residue window size. It was also observed that ABCpred overestimated the presence of epitopes in their respective peptides, which led to reduced accuracy and increased sensitivity. Nevertheless, it remained part of the final consensus algorithm to improve its overall sensitivity.

At this point it should be noted that LBEEP was also tested on a peptide length of 14-residues, since the method was reported to perform optimally when a window size between 5-15 residues is used for prediction. Results showed that the method indeed performs better at this window size, but it is still marginally better than a random prediction according to its MCC (**Supplementary Table 2**). Even though, the results were better for LBEEP the rest of the methods either cannot be used at that window size or perform way worse than what we had already seen and so we opted to not use the 14-residue window any further.

Performance of all predictors except LBEEP on Consensus_NR_exact

Further performance benchmarking was done on the Consensus_NR_exact data set, this time utilizing only the Consensus_NoLBEEP method, and all of the predictors except LBEEP. In

the "per peptide" approach (Table 8), a similar pattern with the Consensus_NR results emerges. The highest MCC score is achieved by BepiPred, with our Consensus_NoLBEEP method coming in second. BcePred is still characterized by high specificity, while ABCpred shows the best sensitivity. A slight shift in performance for the worst is observed by COBEpro and SVMTriP, where both methods score below 50% in accuracy and especially COBEpro, which also has a negative MCC.

Table 8. Performance of predictors in "per peptide" mode. The methods are tested against the Consensus_NR_exact data set.

Predictor	*SN%*	*SP%*	*ACC%*	*MCC*
Consensus_NoLBEEP	50.18	58.54	54.07	0.0873
BcePred	22.46	83.33	50.78	0.0726
ABCpred	67.93	37.08	53.59	0.0527
LBtope	46.2	59.58	52.42	0.0581
BepiPred-1.0	52.72	56.88	54.65	0.0957
COBEpro	33.15	64.79	47.87	-0.0216
SVMTriP	15.4	88.54	49.42	0.0574

The results obtained from the "per residue" approach (Table 9) are quite similar with those reported on the Consensus_NR data set. Furthermore, the BepiPred and Consensus_RES classifiers still perform the best and COBEpro performs the poorest, with a negative MCC score of -0.0107. LBtope seems to perform a bit worse, with a lower MCC value, in contrast with BcePred which showed an increase in MCC value.

Table 9. Performance of "per residue" predictors. The methods are tested against the Consensus_NR_exact data set.

Predictor	*SN%*	*SP%*	*ACC%*	*MCC*
Consensus_RES	47.42	59.64	53.1	0.0709
BcePred	28.5	75.01	50.13	0.0395
LBtope	45.82	57.13	51.08	0.0296
BepiPred-1.0	50.91	56.47	53.49	0.0737
COBEpro	37.09	61.86	48.61	-0.0107

Performance of predictors on BepiPred-2.0's data set at window size of 16

All predictors were also tested using a fixed window size of 16 residues, except for SVMTriP which unfortunately had no default threshold value set for the corresponding model. After initially obtaining poor results on the non-redundant data set, we also opted to use the entire BepiPred-2.0's data set modified, so that all peptides had a fixed length of 16 residues, in order to ascertain the best possible performance for all methods.

Firstly, in the "per peptide" mode (Table 10), the highest MCC score — by a wide margin — was achieved by LBtope, as well as the highest sensitivity, closely followed by BepiPred. The Consensus_noSVMTrip algorithm achieved the highest accuracy with 61.44%, but also exhibited an extremely low sensitivity and a specificity of nearly 100%, indicating that it rejected almost all sequences, perhaps as a result of a very high value in the consensus threshold. High specificity values were also observed for LBEEP, COBEpro and BcePred. Compared to the previous results on Consensus_NR with the window size of 20 amino acid residues, we observe a performance boost, especially in accuracy.

Table 10. Performance of predictors in "per peptide" mode. The methods are tested against BepiPred-2.0's data set at fixed length of 16.

Predictor	*SN%*	*SP%*	*ACC%*	*MCC*
Consensus_noSVMTrip	1.39	99.41	61.44	0.0409
BcePred	27.37	75.48	56.81	0.0319
ABCpred	40.96	63.5	52.24	0.0459
LBtope	57.7	56.06	56.69	0.134
BepiPred-1.0	53.98	53.86	53.91	0.0765
COBEpro	23.06	79.13	57.41	0.0259
LBEEP	23.14	79.73	57.81	0.0341

In the case of the "per residue" predictors (Table 11), LBtope had once again the highest MCC, followed by our consensus method. Regarding the other metrics we see similar performance rankings with our previous test on the 20-residue peptides in MCC and accuracy. However, there is an overwhelming difference between the method's sensitivity and specificity, rendering the method incapable of performing predictions in that window size.

Table 11. Performance of "per residue" predictors. The methods are tested against BepiPred-2.0's data set at fixed peptide length of 16.

Predictor	*SN%*	*SP%*	*ACC%*	*MCC*
Consensus_RES	26.7	79.9	59.3	0.0769
BcePred	31.42	70.74	55.52	0.0231
LBtope	50.09	58.57	55.29	0.0849
BepiPred-1.0	51.45	54.36	53.23	0.0566
COBEpro	36.19	64.86	53.75	0.0107

In short, changing the window size to 16 and including all available peptides of BepiPred-2.0's data set, improved performance as expected, but led to a huge gap between – the previously balanced – sensitivity and specificity metrics. However, even with this advantage for the methods that were trained using peptides from IEDB (LBtope, LBEEP), the results remain unimpressive. Accuracy may have increased to ~58% for LBEEP and COBEpro, but their MCC values are still near zero indicating random classification. The performance improvement observed for LBEEP and LBtope is biased and probably associated with the inclusion of sequences already present in the training data sets of the two methods. On the other hand, ABCpred and COBEpro showed relatively improved accuracy without benefitting from the biased data set, which probably indicates their preference for shorter epitope segments.

Overall method performance and comparison with BepiPred-2.0

The performance of the linear B-cell epitope predictors examined was found to be poor in the data sets and window sizes used during testing (Figure 3). Additionally, despite our optimization, our consensus method performed only marginally better than the rest of the methods, thus nullifying its usefulness. We believe that the problems which may explain these results can be divided into two categories; those concerning the individual methods and those of the consensus approach.

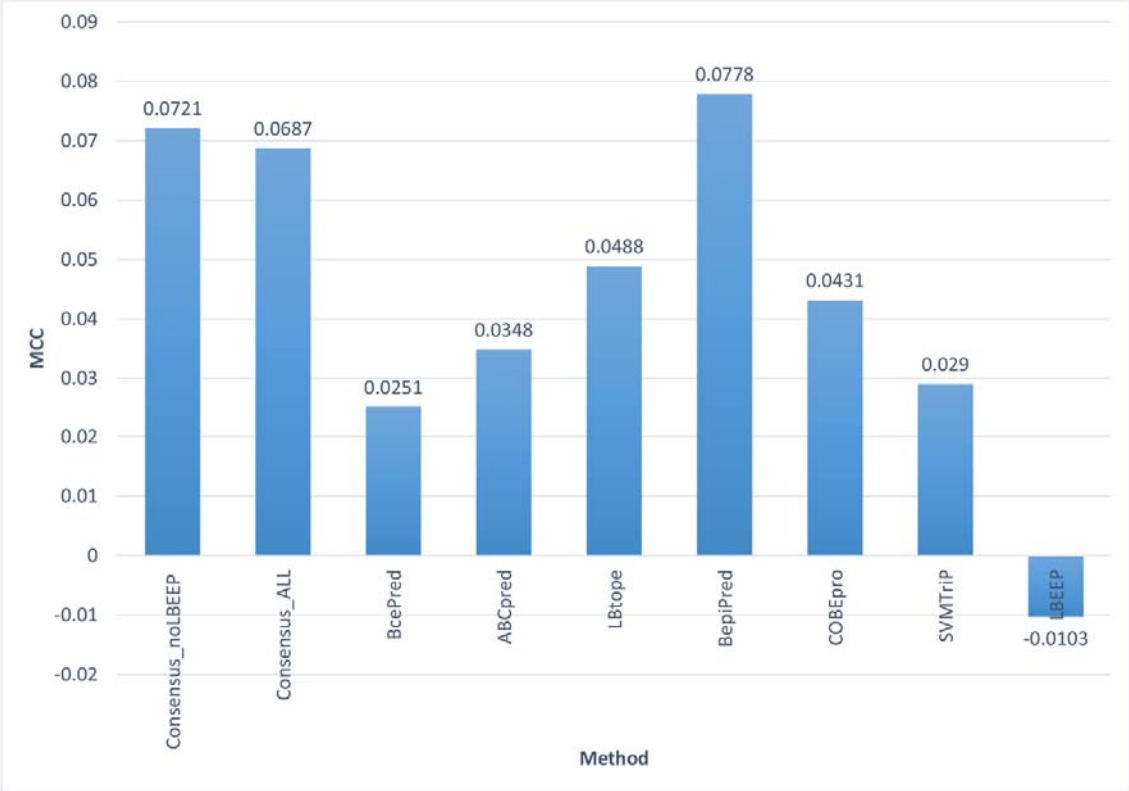


Figure 3. MCC values achieved by all methods tested on the Consensus_NR data set at 20 amino acid residues in "per peptide" mode. The vertical axis represents the MCC value for all the methods and the horizontal axis the names of these methods. The best MCC is achieved by the BepiPred method, followed closely by our Consensus methods, while the worst performers are the LBEEP, SVMTriP and BcePred methods.

The first problem regarding the prediction methods is that the epitope data used to train and test them, and as a result the methods themselves, is outdated. This probably is what caused their significantly reduced performance in our contemporary and considerably larger set of data. Furthermore, the general difficulty of creating a relatively reliable sequence-based predictor is well known, in contrast with those available for example in the prognosis of T-cell epitopes [70]. This is mainly due to the 3D nature of all B-cell epitopes, which consist of seemingly unrelated residue patterns of the antigen. Their emergence is also subject to multiple factors, such as antigen concentration and the type of chemical test [65].

In our attempt to create a consensus predictor, the first problem we encountered was the different mode of operation of the individual prediction methods, namely their distinction into "per peptide" and "per residue" predictors. To effectively compare the two modes, "per

residue" predictor outputs were converted to "per peptide" by using a percentage cut-off to classify peptides as epitopes and non-epitopes. This, however, is not their intended operation mode, which certainly influences the performance of these methods and thus the performance of the consensus method.

Another obstacle in this effort was time and complexity. The prediction and evaluation process for all possible windows (10, 12, 14, 16, 18 and 20) is very time-consuming. This also had to be performed for as many predictors as possible to make the consensus classifier more effective, leading to a significant increase in software development complexity as the number of incorporated predictors grew. In addition, accurate assessment of the viability of such an effort is very difficult, due to the inability to accurately compare them beforehand using the results presented in the corresponding publications, as there is no single set of evaluation data or metrics [12]. Finally, there was a lack of variety in the methods utilized in our selected predictors, where most of them were based on SVM models, which may have negatively affected the performance of our consensus predictor [71].

When comparing all of the methods we tested, with some of the newer methods such as BepiPred-2.0 and iBCE-EL, which were tested on large non-redundant data sets much like the ones we used, their reported superiority is apparent. Out of the two, BepiPred-2.0 was released during the initial part of testing in our research, and as such it was a likely candidate for our consensus method. However, after observing the poor performance of all the different methods tested against its data set, we decided to not include it in our consensus approach, but simply to use it as a reference for what a modern predictor can achieve versus the older ones. Unlike its predecessor, BepiPred-1.0, and most other sequence based predictors, BepiPred-2.0 is trained exclusively on epitope data derived from antigen-antibody crystal structure complexes obtained from the Protein Data Bank [72]. This was done in order to combat the generally poor performance of predictors, which can be partly attributed to poorly annotated and noisy training data, in comparison with data derived from crystal structures which is presumed to be of higher quality and indeed resulted in a significantly improved predictive power [17]. From these complexes all antigenic residues close enough to their respective antibody were gathered. These residues became the positive subset of the training data set, while the negative subset was constructed from randomly selected non-epitope residues.

While, BepiPred-2.0 was trained using epitope data derived only from 3D structures, its performance on linear BCEs was also benchmarked on one such data set. We compared the performance of BepiPred-2.0 against our Consensus_noLBEEP predictor using the Consensus_NR dataset at a window size of 20 amino acid residues. When compared to our consensus method, BepiPred-2.0 has a similar performance in accuracy and MCC, but exhibits higher sensitivity and lower specificity, as shown in the comparison performed in Table 12. However, the results for both methods are far from optimal, and a lot of work still remains to be done in order to create a predictor that will perform optimally during linear BCE detection.

Table 12. Comparison of the performance of our consensus predictor and BepiPred-2.0 against the Consensus_NR data set.

Predictor	***SN%***	***SP%***	***ACC%***	***MCC***
Consensus_NoLBEEP	50.18	58.54	54.07	0.0873
BepiPred-2.0	63.35	42.63	51.93	0.0607

CONCLUSIONS

In summary, in this paper we independently evaluated the performance of seven of the most popular linear B-cell epitope predictors on the largest unbiased data set possible. In the process, we also presented the course of design, development and evaluation of a consensus prediction algorithm for linear B-cell epitopes. The performance of all predictors, except for LBEEP on whom testing was exploratory, was found marginally better than random classification. Additionally, our Consensus classifier failed to significantly outperform its constituent methods. While the method comparison was performed with some necessary compromises, we believe that this update in performance can help to better inform researchers that wish to consult some of these tools to facilitate and direct their research. Instead, we should also like to suggest that researchers opt for some of the newer predictors referenced in this work, like BepiPred-2.0. Also, due to the apparent difficulty of constructing an accurate general-purpose linear BCE predictor, we believe that software development should instead be focused to the creation of more specialized predictors for specific antigenic systems, such as known viruses or viral families of high interest. This could lead to optimization in the feature selection process during classifier training and better predictive performance within that limited scope.

Supporting Information

Supplementary Table 1: Evaluation Datasets

Supplementary Table 2: LBEEP's performance on peptides of 14-amino acid residues

Supplementary File 1: Instructions for the installation of the Virtual Machine containing the BCE predictors and for the execution of the consensus method.

Supplementary File 2: Workflow (A) of the consensus method and (B) of the creation of the non-redundant data sets.

Author Contributions

Study design: KCN, NCP, VAI; Conceptualization: KCN, NCP, VAI; Retrieval of standalones and installation: KAG, GNP, KCN; Development of consensus method: KAG; Data set acquisition and curation: KAG, KCN; Writing - original draft: KAG; Writing - review and editing: KCN, KAG, NCP, GNP, VAI; Supervision: VAI. The manuscript was written through contributions of all authors. All authors have given approval to the final version of the manuscript.

Acknowledgement

We thank the National and Kapodistrian University of Athens for use of premises and equipment.

Abbreviations

Linear B-cell epitope, BCE; Hidden Markov Model, HMM; Support Vector Machine, SVM; Amino Acid Pair, AAP; Molecular Operating Environment, MOE; K-Nearest Neighbor, KNN; Position-Specific Scoring Matrix, PSSM; Extremely Randomized Tree, ERT; Gradient Booosting, GB; Graphical User Interface, GUI; Area Under Curve, AUC; Receiver Operating Characteristic, ROC; Recurrent Neural Network, RNN; Matthews Correlation Coefficient, MCC; Immune Epitope Data Base, IEDB; Dipeptide Deviation from Expected Mean, DDE; Consensus_Redundant, Consensus_R; Consensus_Non_Redundant, Consensus_NR; Sensitivity, SN; Specificity, SP; Accuracy, ACC; True Positive, TP; True Negative, TN; False Positive, FP; False Negative, FN

Conflict of Interest: None declared.

References

[1] 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.

[2] 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.

[3] 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.

[4] 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.

[5] A.J. Chirino, M.L. Ary, S.A. Marshall. Minimizing the immunogenicity of protein therapeutics. Drug Discov Today 2004; 9:82-90.

[6] 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.

[7] 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.

[8] D.J. Barlow, M.S. Edwards, J.M. Thornton. Continuous and discontinuous protein antigenic determinants. Nature 1986; 322:747-748.

[9] J.L. Pellequer, E. Westhof, M.H. Van Regenmortel. Predicting location of continuous epitopes in proteins from their primary structures. Methods Enzymol 1991; 203:176-201.

[10] M.H. Van Regenmortel. Immunoinformatics may lead to a reappraisal of the nature of B cell epitopes and of the feasibility of synthetic peptide vaccines. J Mol Recognit 2006; 19:183-187.

[11] U. Reineke, M. Schutkowski. Epitope mapping protocols. Preface. Methods Mol Biol 2009; 524:v-vi.

[12] Y. El-Manzalawy, V. Honavar. Recent advances in B-cell epitope prediction methods. Immunome Res 2010; 6 Suppl 2:S2.

[13] J.L. Sanchez-Trincado, M. Gomez-Perosanz, P.A. Reche. Fundamentals and Methods for T- and B-Cell Epitope Prediction. J Immunol Res 2017; 2017:2680160.

[14] N. Tomar, R.K. De. Immunoinformatics: a brief review. Methods Mol Biol 2014; 1184:23-55.

[15] D.R. Flower. Immunoinformatics. Predicting immunogenicity in silico. Preface. Methods Mol Biol 2007; 409:v-vi.

[16] M.J. Sweredoski, P. Baldi. COBEpro: a novel system for predicting continuous B-cell epitopes. Protein Eng Des Sel 2009; 22:113-120.

[17] 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.

[18] T.P. Hopp, K.R. Woods. Prediction of protein antigenic determinants from amino acid sequences. Proc Natl Acad Sci U S A 1981; 78:3824-3828.

[19] M. Levitt. A simplified representation of protein conformations for rapid simulation of protein folding. J Mol Biol 1976; 104:59-107.

[20] J.M. Parker, D. Guo, R.S. Hodges. 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; 25:5425-5432.

[21] A.S. Kolaskar, P.C. Tongaonkar. A semi-empirical method for prediction of antigenic determinants on protein antigens. FEBS Lett 1990; 276:172-174.

[22] P. Karplus, G. Schulz. Prediction of chain flexibility in proteins. Naturwissenschaften 1985; 72:212-213.

[23] E.A. Emini, J.V. Hughes, D.S. Perlow, J. Boger. Induction of hepatitis A virus-neutralizing antibody by a virus-specific synthetic peptide. J Virol 1985; 55:836-839.

[24] J.L. Pellequer, E. Westhof, M.H. Van Regenmortel. Correlation between the location of antigenic sites and the prediction of turns in proteins. Immunol Lett 1993; 36:83-99.

[25] J.L. Pellequer, E. Westhof. PREDITOP: a program for antigenicity prediction. J Mol Graph 1993; 11:204-210, 191-202.

[26] A.J. Alix. Predictive estimation of protein linear epitopes by using the program PEOPLE. Vaccine 1999; 18:311-314.

[27] M. Odorico, J.L. Pellequer. BEPITOPE: predicting the location of continuous epitopes and patterns in proteins. J Mol Recognit 2003; 16:20-22.

[28] 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.

[29] M.J. Blythe, D.R. Flower. Benchmarking B cell epitope prediction: underperformance of existing methods. Protein Sci 2005; 14:246-248.

[30] J.E. Larsen, O. Lund, M. Nielsen. Improved method for predicting linear B-cell epitopes. Immunome Res 2006; 2:2.

[31] S. Saha, G.P. Raghava. Prediction of continuous B-cell epitopes in an antigen using recurrent neural network. Proteins 2006; 65:40-48.

[32] N.D. Rubinstein, I. Mayrose, E. Martz, T. Pupko. Epitopia: a web-server for predicting B-cell epitopes. BMC Bioinformatics 2009; 10:287.

[33] Y. El-Manzalawy, D. Dobbs, V. Honavar. Predicting linear B-cell epitopes using string kernels. J Mol Recognit 2008; 21:243-255.

[34] Y. El-Manzalawy, D. Dobbs, V. Honavar. Predicting flexible length linear B-cell epitopes. Comput Syst Bioinformatics Conf 2008; 7:121-132.

[35] Y.I. Davydov, A.G. Tonevitsky. Prediction of linear B-cell epitopes. Molecular Biology 2009; 43:150-158.

[36] J. Chen, H. Liu, J. Yang, K.C. Chou. Prediction of linear B-cell epitopes using amino acid pair antigenicity scale. Amino Acids 2007; 33:423-428.

[37] L.J. Wee, D. Simarmata, Y.W. Kam, L.F. Ng, J.C. Tong. SVM-based prediction of linear B-cell epitopes using Bayes Feature Extraction. BMC Genomics 2010; 11 Suppl 4:S21.

[38] H.W. Wang, Y.C. Lin, T.W. Pai, H.T. Chang. 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.

[39] Y. Wang, W. Wu, N.N. Negre, K.P. White, C. Li, P.K. Shah. Determinants of antigenicity and specificity in immune response for protein sequences. BMC Bioinformatics 2011; 12:251.

[40] Y. Lian, M. Ge, X.M. Pan. EPMLR: sequence-based linear B-cell epitope prediction method using multiple linear regression. BMC Bioinformatics 2014; 15:414.

[41] Y. Lian, Z.C. Huang, M. Ge, X.M. Pan. An Improved Method for Predicting Linear B-cell Epitope Using Deep Maxout Networks. Biomed Environ Sci 2015; 28:460-463.

[42] G. Sher, D. Zhi, S. Zhang. DRREP: deep ridge regressed epitope predictor. BMC Genomics 2017; 18:676.

[43] B. Manavalan, R.G. Govindaraj, T.H. Shin, M.O. Kim, G. Lee. iBCE-EL: A New Ensemble Learning Framework for Improved Linear B-Cell Epitope Prediction. Front Immunol 2018; 9:1695.

[44] J. Sollner, B. Mayer. Machine learning approaches for prediction of linear B-cell epitopes on proteins. J Mol Recognit 2006; 19:200-208.

[45] J. Gao, E. Faraggi, Y. Zhou, J. Ruan, L. Kurgan. BEST: improved prediction of B-cell epitopes from antigen sequences. PLoS One 2012; 7:e40104.

[46] 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.

[47] S.Y. Lin, C.W. Cheng, E.C. Su. Prediction of B-cell epitopes using evolutionary information and propensity scales. BMC Bioinformatics 2013; 14 Suppl 2:S10.

[48] 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.

[49] J.H. Huang, M. Wen, L.J. Tang, H.L. Xie, L. Fu, Y.Z. Liang, H.M. Lu. Using random forest to classify linear B-cell epitopes based on amino acid properties and molecular features. Biochimie 2014; 103:1-6.

[50] 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.

[51] W. Shen, Y. Cao, L. Cha, X. Zhang, X. Ying, W. Zhang, K. Ge, W. Li, L. Zhong. Predicting linear B-cell epitopes using amino acid anchoring pair composition. BioData Min 2015; 8:14.

[52] A.C. Tsolis, N.C. Papandreou, V.A. Iconomidou, S.J. Hamodrakas. A consensus method for the prediction of 'aggregation-prone' peptides in globular proteins. PLoS One 2013; 8:e54175.

[53] C. Ji, S. Ma. Combinations of weak classifiers. IEEE Trans Neural Netw 1997; 8:32-42.

[54] M. Moutaftsi, B. Peters, V. Pasquetto, D.C. Tscharke, J. Sidney, H.H. Bui, H. Grey, A. Sette. A consensus epitope prediction approach identifies the breadth of murine T(CD8+)-cell responses to vaccinia virus. Nat Biotechnol 2006; 24:817-819.

[55] S. Saha, M. Bhasin, G.P. Raghava. Bcipep: a database of B-cell epitopes. BMC Genomics 2005; 6:79.

[56] 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.

[57] G. Pollastri, P. Baldi, P. Fariselli, R. Casadio. Prediction of coordination number and relative solvent accessibility in proteins. Proteins 2002; 47:142-153.

[58] J. Cheng, A.Z. Randall, M.J. Sweredoski, P. Baldi. SCRATCH: a protein structure and structural feature prediction server. Nucleic Acids Res 2005; 33:W72-76.

[59] R. Vita, L. Zarebski, J.A. Greenbaum, H. Emami, I. Hoof, N. Salimi, R. Damle, A. Sette, B. Peters. The immune epitope database 2.0. Nucleic Acids Res 2010; 38:D854-862.

[60] R. Vita, J.A. Overton, J.A. Greenbaum, J. Ponomarenko, J.D. Clark, J.R. Cantrell, D.K. Wheeler, J.L. Gabbard, D. Hix, A. Sette, B. Peters. The immune epitope database (IEDB) 3.0. Nucleic Acids Res 2015; 43:D405-412.

[61] K.K. Frousios, V.A. Iconomidou, C.M. Karletidi, S.J. Hamodrakas. Amyloidogenic determinants are usually not buried. BMC Struct Biol 2009; 9:44.

[62] C. Peters, K.D. Tsirigos, N. Shu, A. Elofsson. Improved topology prediction using the terminal hydrophobic helices rule. Bioinformatics 2016; 32:1158-1162.

[63] L. Kall, A. Krogh, E.L. Sonnhammer. A combined transmembrane topology and signal peptide prediction method. J Mol Biol 2004; 338:1027-1036.

[64] C. The UniProt. UniProt: the universal protein knowledgebase. Nucleic Acids Res 2017; 45:D158-D169.

[65] J.A. Greenbaum, P.H. Andersen, M. Blythe, H.H. Bui, R.E. Cachau, J. Crowe, M. Davies, A.S. Kolaskar, O. Lund, S. Morrison, B. Mumey, Y. Ofran, J.L. Pellequer, C. Pinilla, J.V. Ponomarenko, G.P. Raghava, M.H. van Regenmortel, E.L. Roggen, A. Sette, A. Schlessinger, J. Sollner, M. Zand, B. Peters. Towards a consensus on datasets and evaluation metrics for developing B-cell epitope prediction tools. J Mol Recognit 2007; 20:75-82.

[66] J.V. Kringelum, M. Nielsen, S.B. Padkjaer, O. Lund. Structural analysis of B-cell epitopes in antibody:protein complexes. Mol Immunol 2013; 53:24-34.

[67] Y. Huang, B. Niu, Y. Gao, L. Fu, W. Li. CD-HIT Suite: a web server for clustering and comparing biological sequences. Bioinformatics 2010; 26:680-682.

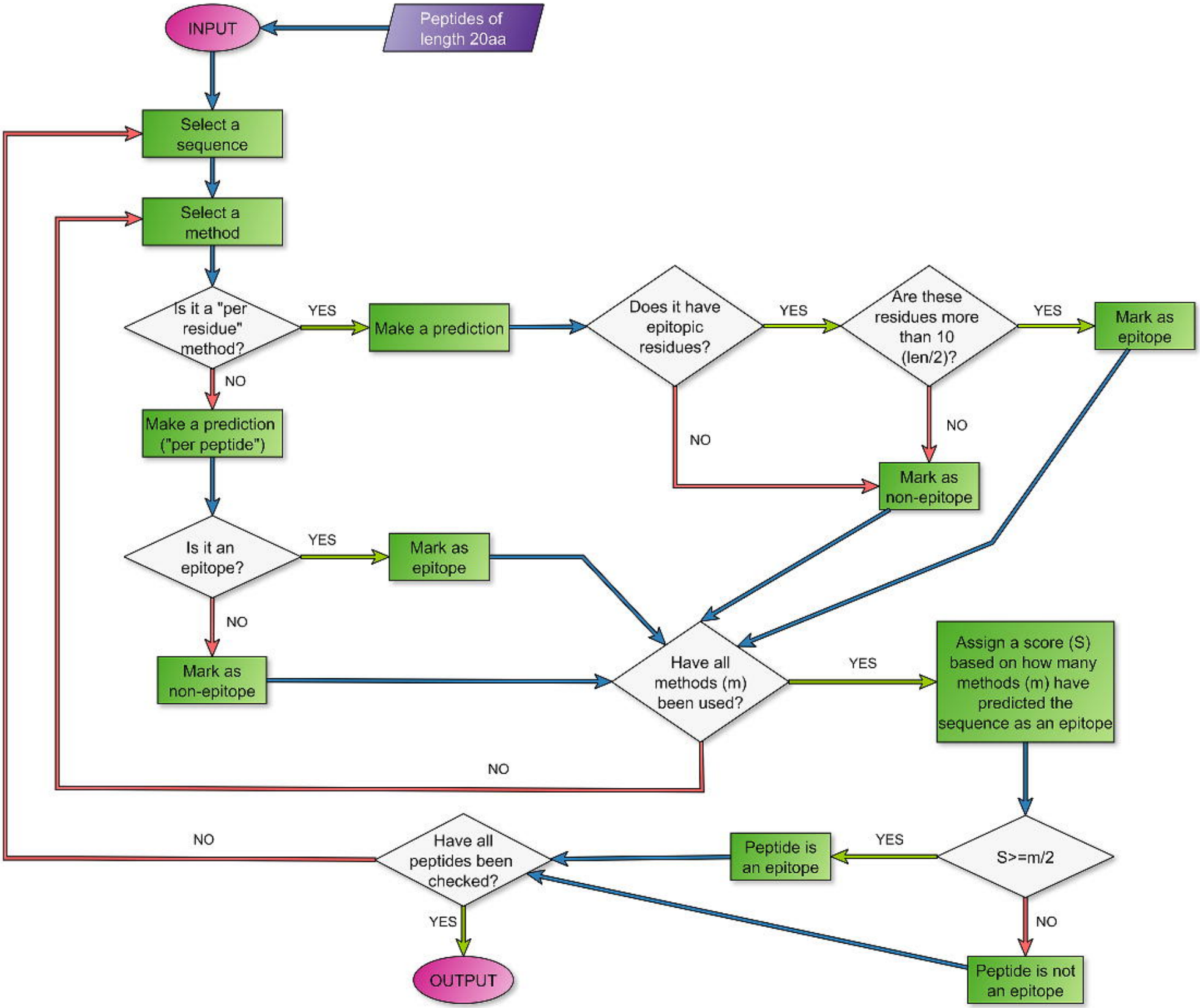
[68] D. Chicco. Ten quick tips for machine learning in computational biology. BioData Min 2017; 10:35.

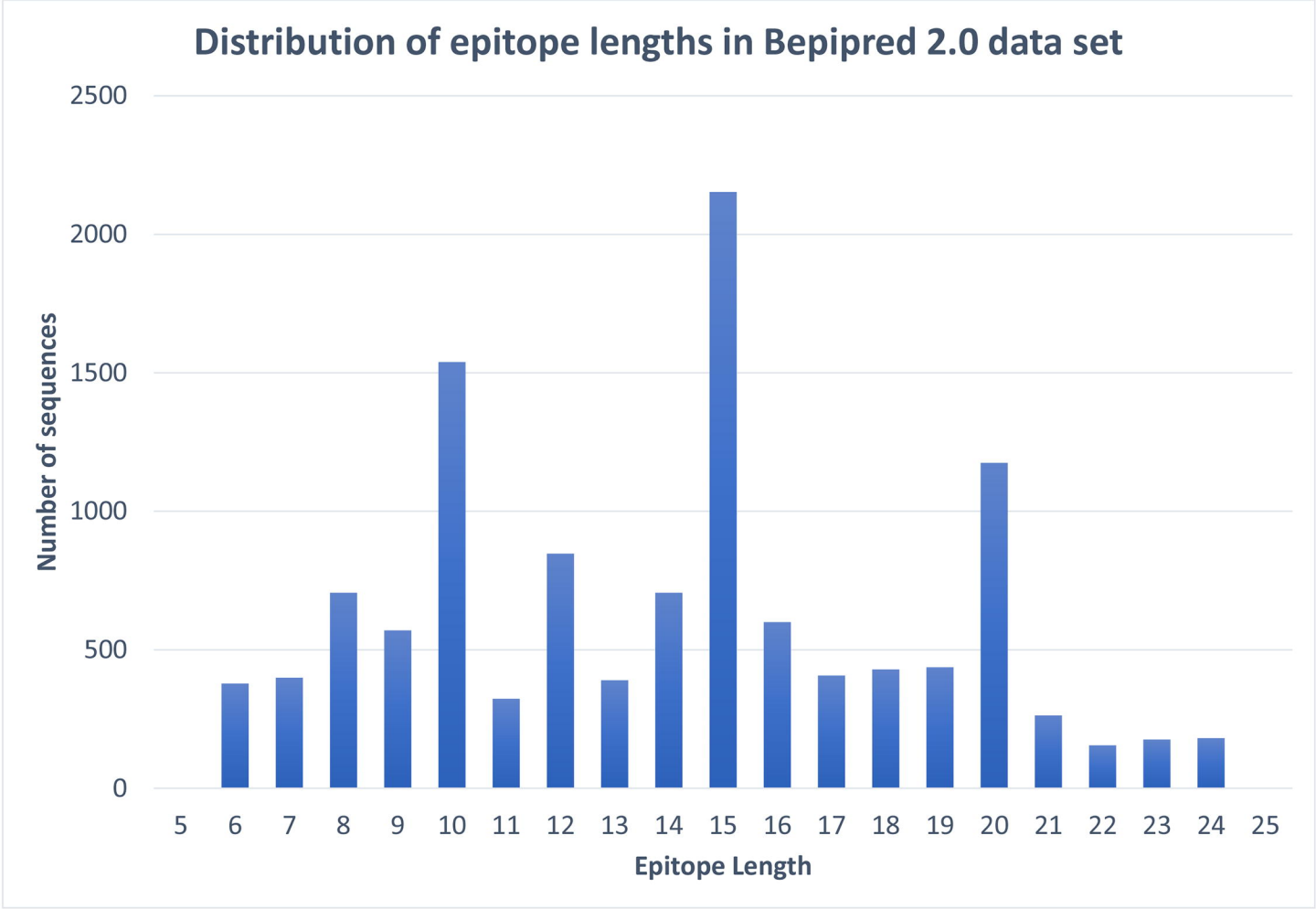
[69] S. Boughorbel, F. Jarray, M. El-Anbari. Optimal classifier for imbalanced data using Matthews Correlation Coefficient metric. PLoS One 2017; 12:e0177678.

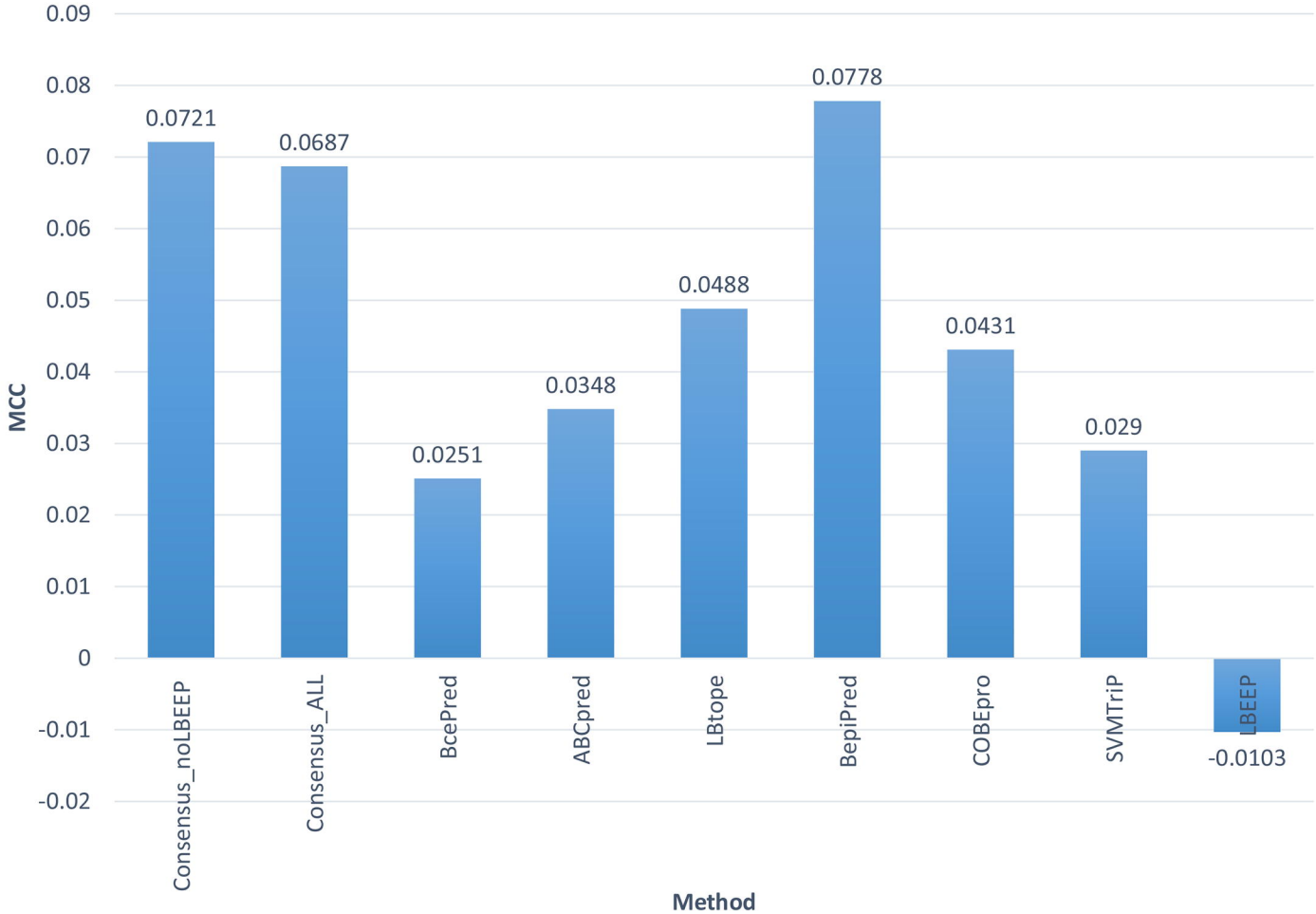
[70] G.N. Konstantinou. T-Cell Epitope Prediction. Methods Mol Biol 2017; 1592:211-222.

[71] D.R. Flower. Immunoinformatics and the in silico prediction of immunogenicity. An introduction. Methods Mol Biol 2007; 409:1-15.

[72] H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I.N. Shindyalov, P.E. Bourne. The Protein Data Bank. Nucleic Acids Res 2000; 28:235-242.


INPUT
Peptides of length 20aa
Select a sequence
Select a method
Is it a "per residue" method?
YES
NO
Make a prediction
Does it have epitopic residues?
YES
NO
Are these residues more than 10 (len/2)?
YES
NO
Mark as epitope
Mark as non-epitope
Make a prediction ("per peptide")
Is it an epitope?
YES
NO
Mark as epitope
Mark as non-epitope
Have all methods (m) been used?
YES
NO
Assign a score (S) based on how many methods (m) have predicted the sequence as an epitope
S>=m/2
YES
NO
Peptide is an epitope
Peptide is not an epitope
Have all peptides been checked?
NO
YES
OUTPUT


Distribution of epitope lengths in Bepipred 2.0 data set
Number of sequences
2500
2000
1500
1000
500
0
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
Epitope Length


0.09
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0
-0.01
-0.02
MCC
0.0721
0.0687
0.0251
0.0348
0.0488
0.0778
0.0431
0.029
-0.0103
Consensus_noLBEEP
Consensus_ALL
BcePred
ABCpred
LBtope
BepiPred
COBEpro
SVMTriP
LBEEP
Method