## Title

ApicoAMP: The first computational model for identifying apicoplast-targeted transmembrane proteins in Apicomplexa

## Author Names and Affiliations

Gokcen Cilingir[1], Audrey O.T. Lau[2,3], and Shira L. Broschat[1,2,3*]

[1]School of Electrical Engineering and Computer Science, Washington State University
Pullman, WA 99164, USA

[2]Department of Veterinary Microbiology and Pathology, Washington State University
Pullman, WA 99164, USA

[3]Paul G. Allen School for Global Animal Health, Washington State University
Pullman, WA 99164, USA

[*]Corresponding author: shira@eecs.wsu.edu, 1-509-335-5693

## Abstract

**Background:** Apicomplexan parasites contain a relict chloroplast known as the apicoplast. This organelle is essential for parasite survival and thus serves as a promising target for drug treatment. As the gatekeepers of this important organelle, apicoplast membrane proteins are potentially excellent drug target candidates and therefore their identification is important. A limited number of apicoplast membrane proteins have been identified experimentally, but it is impractical to identify them all *in vitro*. Thus, there is a strong need for identification of apicoplast membrane proteins by means of a computational approach. Unfortunately, no such computational method exists.

**Methodology/Principal Findings:** In this work, we develop a method for predicting apicoplast-targeted transmembrane proteins for multiple species of Apicomplexa, whereby several classifiers trained on different feature sets and based on different algorithms are evaluated and combined in an ensemble classification model to obtain the best expected performance. The feature sets considered are the hydrophobicity and composition characteristics of amino acids over transmembrane domains, the existence of short sequence motifs over cytosolically disposed regions, and Gene Ontology (GO) terms associated with given proteins. Our model, ApicoAMP, is an ensemble classification model that combines decisions of classifiers following the majority vote principle. ApicoAMP is trained on a set of proteins from 11 apicomplexan species and achieves 91% overall expected accuracy.

**Conclusions/Significance:** ApicoAMP is the first computational model capable of identifying apicoplast-targeted transmembrane proteins in Apicomplexa. The ApicoAMP prediction software is available at http://code.google.com/p/apicoamp/ and http://bcb.eecs.wsu.edu.

**Keywords:** apicoplast, apicoplast-targeted membrane proteins, transmembrane proteins, Apicomplexa, machine learning, ensemble classification models, gene ontology annotation, protein motifs

## 1. Introduction

Apicomplexan parasites, including the causative agent of the most deadly form of malaria, *Plasmodium falciparum*, contain a relict prokaryotic-derived plastid known as the apicoplast. This organelle is essential for parasite survival and thus is a promising drug target. Most apicoplast proteins are nuclear-encoded and targeted post-translationally to the organelle. *In silico* prediction of proteins that are destined to the apicoplast lumen can be reliably performed for multiple species of Apicomplexa because of the known bipartite signaling mechanism that requires an N-terminal signal peptide (SP) followed by a transit peptide (TP) (Cilingir et al., 2012; Foth et al., 2003). However, we have limited understanding of the signaling mechanism for proteins that reside in the four membranes surrounding the apicoplast.

Recent experimental findings have confirmed many apicoplast-targeted membrane proteins which have been found to lack a bipartite signal (DeRocher et al., 2008; Karnataki et al., 2007; Sheiner et al., 2011). These findings have revealed a trafficking mechanism that occurs via the endoplasmic reticulum (ER) whereby an internal signal sequence anchors the protein on the ER membrane (Lim et al., 2009). The remainder of the trafficking, explaining the transport of proteins from the ER to apicoplasts, has not been dissected yet, but studies have confirmed the involvement of vesicles for some apicoplast membrane proteins (DeRocher et al., 2008;

Karnataki et al., 2007a, 2007b). Vesicular transport is not uncommon for other cellular destinations by which membrane-bound proteins traffic through the ER en route to an organelle. Transportation of such membrane proteins within the secretory system involves short sequence based sorting signals that appear on the cytosolically disposed regions of membrane proteins (Michelsen et al., 2005; Sato and Nakano, 2002).

Most of the recent findings on apicoplast membrane proteins apply to a subset of membrane proteins that are called transmembrane proteins. These proteins contain transmembrane domains (TMDs) that function as membrane anchors. The topology of TMDs, i.e., the location and orientation of the membrane spanning regions, can be reliably identified by well-established prediction algorithms (Hofmann and Stoffel, 1993; Krogh et al., 2001; von Heijne, 1992). These methods provide location as well as direction information for each predicted TMD, indicating whether the non-TMD regions of a protein reside in the cytosolic side or in the exoplasmic side of the membrane.

Although well-established prediction algorithms exist for transmembrane domain topology prediction, there is no computational approach in the literature that identifies transmembrane proteins targeted to the apicoplast. In fact, prediction of subcellular localization of membrane proteins had not been studied separately from globular proteins until recent years. At present, only a handful of methods developed specifically for membrane localization prediction exist in the literature. Pierleoni et al. (2011) have described the shortcomings of not studying membrane proteins separately from globular proteins, providing evidence that popular predictors mostly trained on globular proteins fail to classify membrane proteins accurately. They developed the predictor called MemLoci, which is trained on membrane proteins. MemLoci greatly outperforms some popular general-purpose predictors on an independent set of eukaryotic membrane proteins.

The MemLoci algorithm was highly influenced by the work of Sharpe et al. (2010), in which an original hypothesis regarding membrane protein localization prediction was developed and tested. It is known that various membranes of eukaryotic cells differ in composition. Sharpe et al. hypothesized that the sequences of TMDs should reflect this compositional difference and should have different physical properties because TMDs are the regions of transmembrane proteins that reside in the membrane. Through extensive analysis their work clearly demonstrated that there are in fact identifiable differences in TMDs of known ER, Golgi, and plasma membrane proteins in both vertebrates and fungi. Pierleoni et al. extended this idea and applied it on a larger scale to discriminate plasma membrane, internal membrane, and organelle membrane proteins of eukaryotes.

In contrast to these two sequence-based methods, Du (2012) and Du et al. (2012) demonstrated how the use of external information such as Gene Ontology (GO) annotations might improve prediction of membrane protein localization. Prediction through annotation transfer is a common methodology in subcellular localization prediction (Blum et al., 2009; Chi and Nam, 2012; Huang et al., 2008; Li et al., 2012; Mei et al., 2011). A downside of this approach is that one cannot predict the subcellular localization if no annotation is available for a given protein. One generally overcomes this disadvantage by combining annotation transfer-based predictors with other types of predictors. This has the advantage of using existing knowledge on a class of proteins, while still allowing prediction in cases where no prior knowledge exists. Recent studies on subcellular localization prediction of membrane proteins have demonstrated the utilization of an array of different feature sets as well as different machine learning approaches. Sharpe et al. developed a neural network classifier that predicts localization from amino acid composition, hydrophobicity characteristics, and the length of membrane spanning regions of single-pass transmembrane proteins (proteins with a single TMD). This method achieved a mean accuracy of 76% over 3 classes (ER, Golgi, and plasma membrane) for which the highest accuracy achieved was 39% by other popular localization predictors. Pierleoni et al. (2011) used hydrophobicity and composition characteristics of amino acids over highly hydrophobic stretches, as well as the N and C sequence termini of proteins, to train Support Vector Machine (SVM) classifiers. Du (2012) determined the prospective localization of a given protein solely by looking at the GO terms associated with a protein. Each GO term was assigned a likelihood score during training which was then used to quantify the likelihood of a given protein belonging to a particular localization class. Du et al. (2012) improved this approach by introducing the use of a sequence similarity search to enrich the set of GO terms of a protein with the GO terms of proteins that share sequence similarity with the given protein.

The trafficking of membrane proteins from ribosomes to their final destinations is a process that involves diverse molecular mechanisms which have been only partially unraveled (Pierleoni et al., 2011). The strength of the four prediction approaches described above (Du, 2012; Du et al., 2012; Pierleoni et al., 2011; Sharpe et al., 2010) is their ability to discriminate membrane proteins by classes *independent* of the trafficking mechanisms involved. Experimental verification of their success indicates that emergent properties, in fact, do exist that are specific to membrane classes and, importantly, these properties can be utilized by machine learning approaches to predict membrane localization of proteins.

In this study, we have developed a method for predicting apicoplast-targeted transmembrane proteins (ApicoTMPs) over multiple species of Apicomplexa, whereby several classifiers based on different algorithms and trained on different feature sets are evaluated and combined in an ensemble classification model to get the best expected performance. Hydrophobicity and composition characteristics of amino acids over transmembrane domains, existence of short sequence motifs over cytosolically disposed regions, and Gene Ontology (GO) terms associated with given proteins are the feature sets considered. Our model, ApicoAMP, is an ensemble classification model that combines decisions of classifiers following the majority vote principle. ApicoAMP, is trained on a set of proteins from 11 apicomplexan species and achieves 91% overall expected accuracy.

# 2. Methods

## 2.1. The dataset

We obtained experimentally-confirmed apicoplast-targeted proteins from the ApiLoc database (version 3, http://apiloc.bio21.unimelb.edu.au) and from recent references (Fleige et al., 2010; Sheiner et al., 2011). Additionally, we identified orthologs of these proteins from the OrthoMCL database (version 5) (Chen et al., 2006). Proteins predicted to contain transmembrane domains are used as the positive training set in the training of ApicoAMP [see Supplementary data 1]. The transmembrane Hidden Markov Model (TMHMM) (Krogh et al., 2001) is used for transmembrane domain prediction.

We obtained proteins from the ApiLoc database tagged as non-Apicoplast or confirmed to localize to a parasitophorous vacuole, plasma membrane, rhoptry, microneme, Golgi, endosome, erythrocyte, dense granule, or host cell plasma membrane. Additionally, we identified orthologs of these proteins from the OrthoMCL database (version 5) (Chen et al., 2006). Proteins predicted to contain transmembrane domains are used as the negative training set in the training of ApicoAMP [see Supplementary data 2].

All protein sequences were obtained from EuPathDB (version 2.13) (Aurrecoechea et al., 2010), which is the main biological sequence repository for eukaryotic pathogens such as Apicomplexa. Redundant sequences that share more than 70% sequence similarity were eliminated from both negative and positive sets using the CD-HIT method (Li and Godzik, 2006).

Proteins from 11 apicomplexan species exist in the resulting sets, namely *Plasmodium knowlesi*, *Plasmodium berghei*, *Neospora caninum*, *Toxoplasma gondii*, *Plasmodium yoelii*, *Plasmodium chabaudi*, *Plasmodium falciparum*, *Babesia bovis*, *Theileria annulata*, *Plasmodium vivax*, and *Theileria parva*. Table 1 shows the breakdown of the training set by positive (ApicoTMP) and negative (non-ApicoTMP) classes for the 11 species. Overall, positive and negative training sets contain 56 and 154 proteins, respectively.

## 2.2. Computational problem definition

From a computational point of view, the prediction of a given protein as an ApicoTMP or non-ApicoTMP can be stated as a binary classification problem, for which we choose ApicoTMP as the positive class. A typical supervised learning strategy utilizes a training set containing positive and negative labeled instances to learn a mapping from the input space to the output space. In our case, the input space is defined as the set of all apicomplexan protein sequences, and the output space contains two class label values: ApicoTMP and non-ApicoTMP. When applied to a classification model, the training procedure produces a classifier instance, which can then be employed to predict the status of unlabeled proteins.

Devising a typical supervised classification model requires a decision of how to encode inputs—i.e., how to map them into a given feature space—whereby positive and negative classes can be reliably separated. Another important decision is the choice of a classification algorithm to actually separate positive and negative classes in the feature space. In the next sections, we discuss the different classification algorithms and feature extraction strategies we evaluated to develop nine different classification models, each a candidate solution for the ApicoTMP prediction problem. The performance of the different classification models is compared in the results section, and the model with the best performance is identified. Rather than presenting only the best model, we present all the candidate models we considered. Because at present no established computational approaches to our problem exist in the literature, we believe that including this information will be useful for future development. In addition, it demonstrates the merits of our choice in comparison to the other viable candidate models.

## 2.3. Classification algorithm selection

After considering a number of different classification algorithms, including naïve Bayes, logistic regression, and neural network algorithms, we chose to use the support vector machine (SVM) as the main classification algorithm for our experiments. SVM is a popular classification algorithm (Vapnik, 1995, 1998), which has been successfully applied in many problem domains including the subcellular localization prediction of proteins. SVM is a supervised learning algorithm that produces a classifier by constructing an optimal hyperplane dividing the positive and negative classes with a maximum margin of separation. The SVM-light classifier (Joachims, 1999) was used with the radial basis function kernel. Gamma and C parameters were set to 1 and 4, respectively, based on a grid search in parameter space. In a grid search, one defines ranges and increments for all parameters and evaluates possible combinations in the resulting n-dimensional parameter grid space to find the best parameter combination. We utilized this approach to determine all the parameters used in this work. Initially we used relatively large ranges and increment values which we then gradually reduced. More is said about parameter optimization in the results section.

For our candidate models, we utilized the SVM classification algorithm with different feature sets. In addition to the use of SVMs, we evaluated the Projected Gene Ontology Score (PGOS) (Du et al., 2012) classification algorithm. Given a protein associated with a number of GO terms, the PGOS algorithm uses the training set to calculate the prospect of each GO term being associated with both positive and negative instances. Scores associated with each GO term are then added over each class and the one with the maximum score is chosen as the class of the given protein.

As described earlier in the dataset section, our training set consists of 56 positives and 154 negatives, which means that our training set is imbalanced. Training a classifier on an imbalanced dataset is often problematic and this is true for SVMs (Ben-Hur and Weston, 2010; Provost, 2000). Two common ways of overcoming this problem are by using separate soft-margin constants for positive and negative classes and by altering the training balance. From our experiments we found that the latter approach works best for our training set.

To address the imbalanced training data problem, we evaluate each of the nine classification models with an ensemble classification architecture consisting of classification units that are independently trained on balanced subsets of the training data. Each balanced subset contains all positive instances and the same number of negative instances, which are drawn randomly from the negative training set. Each classification unit is trained using a different training subset but the same classification model. Because having 10 classification units guarantees that almost every negative instance appears at least once in one of the training subsets, we use 10 units. Given a protein sequence, each classification unit's decision is obtained, which can be either positive or negative. For a protein to be labeled as positive, at least *n* out of 10 classification units should give a positive class label. Here, *n* or the *vote threshold* is treated as a parameter in our classification architecture and is set by the user. We evaluate the use of different classification models assuming this standard ensemble architecture, and we report performance for several values of the *vote threshold* parameter.

## 2.4. Extracting features from proteins

As stated earlier, development of a classification model requires both a classification algorithm and a method for mapping input protein sequences into feature space. We described candidates for classification algorithms in the previous section, and in this section we discuss the different feature sets we extract from the training data for use in differentiating between ApicoAMPs and non-ApicoAMPs.

### 2.4.1. Feature extraction from transmembrane domains

The sequences of transmembrane domains (TMD) reflect the different physical properties of various membranes of eukaryotic cells. As demonstrated by Sharpe et al. (2010) and Pierleoni et al. (2011), one can exploit this difference for transmembrane protein classification.

We identified TMDs in protein sequences using the transmembrane Hidden Markov Model (TMHMM) (Krogh et al., 2001). Since N-terminal transmembrane domains are often confused with signal peptide (SP) regions, we crosschecked predictions of TMHMM with SignalP 3.0 (Bendtsen et al., 2004) predictions to eliminate proteins with SPs rather than a single transmembrane domain (TMD) at the N-terminal. A TMD region is composed of 3 sub-regions: a hydrophobic core and pre-TMD and post-TMD sub-regions that are aligned with the inner and outer leaflets of the membrane. When TMDs are aligned from the cytoplasmic side to the exoplasmic side rather than from N terminus to C terminus, pre-TMD and post-TMD regions are found on the cytoplasmic and exoplasmic end of the TMD region, respectively. A schematic representation of a typical TMD region is given in Figure 1.

Hydrophobic cores of TMDs were identified following a procedure similar to the one proposed by Sharpe et al. (2010). The approximate TMD edges identified by TMHMM were used as guides and these edges were indented by ***i*** amino acids at each end. Then the resulting region was scanned through a window of ***w*** residues centered on the measured residue. For each measured residue, a decision for involvement in a hydrophobic core was reached by comparing the average hydrophobicity over the window against a threshold (-0.94 kcal/mol) and by comparing the hydrophobicity of the measured residue against another threshold (8 kcal/mol). If one of these measurements exceeded the given thresholds for a residue, it was set as the edge of the hydrophobic core. Scanning was performed from each end toward the other. Thresholds involved in this procedure were taken directly from Sharpe et al. (2010). The hydrophobicity scale of Goldman, Engelman, and Steitz (GES) (Engelman et al., 1986) was used.

Once the hydrophobic core of a TMD was identified, pre-TMD and post-TMD regions were found to be the regions of length ***p*** that start immediately before and immediately after the TMD core. The following features were extracted from the TMDs of a protein:

- Frequency of each amino acid in the identified hydrophobic core of a TMD, recorded in a 20-valued vector with elements ranging between 0 and 1. An element-wise average is taken over all TMDs in a protein sequence.

- Average length of the hydrophobic cores of a TMD.

- Average hydrophobicity of the hydrophobic cores of a TMD as well as the average hydrophobicity of fractions of the cores such as each half, each one third, and up to each one eighth of the cores.

- Average hydrophobicity of the pre-core and post-core regions of a TMD.

The parameters used during this feature extraction procedure, namely the indentation amount ***i***, window size ***w*****,** and pre-core and post-core region lengths ***p***, were determined via a grid search in parameter space and set to be 5, 5, and 4, respectively.

### 2.4.2. Feature extraction based on short sequence motifs

Transportation of membrane protein targeting within the secretory system is known to involve short sequence motifs that appear on the cytosolically disposed regions of these proteins. A recent study confirmed that a cytosolic tyrosine-based motif is required but not sufficient for apicoplast targeting of a *Toxoplasma gondii* protein, apicoplast phosphate transporter 1 (APT1) (DeRocher et al., 2012). The sequence motif identified was Y[GE], and it was observed in the N-terminal region prior to the first TM domain. Although this motif does not appear with significant frequency in our training set, this finding motivated our use of motif discovery algorithms to identify a set of short sequence motifs for feature encoding. We used TMHMM (Krogh et al., 2001) to identify the regions of transmembrane proteins predicted to reside on the cytoplasmic side of the membrane in our training data. Next two different motif discovery algorithms, MERCI (Vens et al., 2011) and MEME (Bailey et al., 2006), were used to perform motif discovery over the cytosolically disposed regions of the proteins.

MERCI uses a consensus string model as the motif model, which essentially expresses motifs as regular expressions. This method identifies the top ***k*** motifs that are most frequent in a positive training set and absent from a negative training set. The MERCI algorithm requires two parameters $\boldsymbol{F_P}$ and $\boldsymbol{F_N}$, which denote the minimal frequency threshold for the positive sequences and the maximal frequency threshold for the negative sequences, respectively. MERCI performs level-wise search over the motif space, modifying the basic AprioriAll algorithm, such that motifs that occur frequently in positive sequences are searched for compliance with the maximal frequency threshold $\boldsymbol{F_N}$.

MEME uses a position weight matrix model as the motif model, which describes the probability of each possible letter at each position in a motif. The original algorithm only uses positive training data to determine the set of overrepresented motifs, but the use of position-specific priors allows the algorithm to make use of negative training data (Bailey et al., 2010). MEME applies an expectation maximization algorithm to fit a mixture of motif models. It identifies ***k*** motifs with widths between $\boldsymbol{width_{min}}$ and $\boldsymbol{width_{max}}$ and uses a p-value threshold ***p*** while quantifying the existence of a motif in a sequence.

The motifs identified by MERCI were used as features to encode the proteins where feature quantification was performed as follows: if a protein contains a motif, its corresponding feature value is taken as 1; otherwise it is taken as 0. Because it is a probabilistic model, MEME associates p-values with motif occurrences. When MEME was utilized, feature quantification involved the use of these p-values. The parameters required by the MERCI algorithm, namely $\boldsymbol{F_P}$, $\boldsymbol{F_N}$**,** and ***k***, were determined via a grid search in parameter space and set to be 5, 2, and 20, respectively. The same strategy was used with MEME, where ***k***, $\boldsymbol{width_{min}}$, $\boldsymbol{width_{max}}$, and ***p*** were set to be 10, 3, 5 and 0.1, respectively.

### 2.4.3. Feature extraction based on GO annotations

The goal of the Gene Ontology (GO) project is to provide a controlled vocabulary for gene and gene product attributes. Ontology covers 3 domains: cellular component, molecular function, and biological process. GO terms associated with a protein can be used as descriptors of the protein. Du (2012) and Du et al. (2012) demonstrated the use of this approach in subcellular localization prediction of eukaryotic membrane proteins. In their initial work, they determined the prospective localization of a protein solely by looking at the GO terms associated with the given protein. They improved this approach by introducing the use of a sequence similarity search to the model. A sequence similarity search is used to identify proteins that are similar to a given protein. The GO terms of the similar proteins are then utilized to enrich the set of GO terms for the given protein.

We evaluated both feature extraction strategies used by Du (2012) and Du et al. (2012). Differing from Du et al. (2012), however, we used an e-value threshold of 1e-05 to ensure that only sufficiently similar sequences were used in the GO term set enrichment process. This, in fact, improved the performance. We built a custom database with Blast+ (Camacho et al., 2009) for our sequence similarity search, using all apicomplexan proteins that share no more than 70% sequence similarity in the creation of this database. We used the CD-HIT (Li and Godzik, 2006) program to identify the clusters of proteins whose sequences are sufficiently similar to each other. CD-HIT selects a representative protein from each cluster. If a protein was not the only one in its cluster, we enriched the GO term list of the representative proteins with the GO terms of the other proteins in the cluster. Du et al. (2012) did not discuss this sort of enrichment process in the preparation of the database to be used in the sequence similarity search, but we think it is a crucial step. The only parameter in this feature extraction method is the number of similar sequences that need to be found in the database. Because of the e-value threshold we introduced, this parameter indicates the maximum number of similar sequences to be found. The actual number of similar sequences to be used for a particular protein varies due to the e-value threshold. The maximum number of sequences parameter was determined via a grid search and set to be 25. We observed that as the value of this parameter is increased, the performance improves, but after it reaches 25 there is no substantial increase in the performance. In our training set, the average actual number of similar sequences used for a protein was observed to be 11. EuPathDB (version 2.13) (Aurrecoechea et al., 2010) was used to obtain the GO terms associated with all apicomplexan proteins. Both the official GO annotations and the predicted ones listed in EuPathDB were used in feature encoding.

Often a protein is not associated with any GO term even following application of the GO term enrichment process, as was observed in about 15% of the proteins in our training set. The presence of a GO term provides useful information regarding the prospect of a protein belonging in a localization class. However, the absence of a GO term is indeterminate because the GO annotation process only evolves as our knowledge of genes and gene products grows. Because of this limitation, a binary classification model using GO terms to encode a protein does not work because there are 3 possible outcomes: positive, negative, or *no-prediction* where the *no-prediction* outcome indicates the absence of known GO terms. A model has to be designed to handle this latter outcome.

### 2.5. Classification models

The two classification algorithms and the various feature extraction methods were used in combination to create nine candidate classification models for ApicoTMP prediction. Three of the classification models use the SVM classification algorithm, two use the PGOS classification algorithm, and the remaining four are ensemble models that use both algorithms. The SVM-based models are trained on features extracted from transmembrane domains and on motif features identified by the MERCI and MEME motif discovery algorithms and are called the SVM-TM Classifier, SVM-MERCI Classifier, and SVM-MEME Classifier, respectively. The PGOS-based models are trained using GO terms and enriched GO terms obtained via sequence similarity searches. These are called the PGOS Classifier and the PGOS-enriched Classifier, respectively.

Our ensemble models consist of two or more of the classifiers described in the previous paragraph. The decisions of the individual classifiers are combined following a majority vote principle, i.e., the final decision is based on the majority vote. For cases when an even number of votes results in a tie, we optimistically choose the protein to be a positive instance.

All the trained classifiers except the ones trained on GO terms label a given protein as either positive or negative. The classifiers that are trained on GO terms do not make a prediction if no GO term is associated with the given protein. When this is the case, the decisions of the rest of the classifiers in the ensemble are combined following the majority vote principle, ignoring the existence of the classifier trained on GO terms.

## 3. Results

Our nine classification model candidates were evaluated using an expected prediction accuracy metric obtained via 5-fold cross validation. Earlier we described the method we employ to balance our training set, which consists of 56 positives and 154 negatives. To implement this balancing approach for 5-fold cross validation, we randomly divided our positive set into 5 groups, each group containing approximately 11 positive instances, and our negative set into 14 groups, each containing 11 instances. These groups of positive and negative instances were used first to determine the optimum parameters for a classification model, next to determine the accuracy of the classification model with the given parameters, and finally to train the classification model found to be most accurate in the previous step to serve as ApicoAMP. These steps are described in the following paragraphs.

From the 5 groups of positives and 14 groups of negatives, two groups from each were placed in reserve. The remaining 3 groups of positives and 3 groups randomly selected from the 12 remaining groups of negatives were used for training each classification unit during the parameter optimization step. As we previously described, our classification architecture consists of 10 classification units. Thus, training was performed 10 times with the same 3 groups of positives but 3 different groups of negatives, randomly chosen, for each classification model. One of the reserved groups was used to test the classification accuracy for a given set of parameters. The procedure was repeated with a different set of parameters until the results converged to the optimum parameter set, i.e., the set that produced the best classification accuracy. The parameter test set was then merged with the parameter training set, and the resulting set comprised of 4 groups was used to train each of the ten classification units constituting each classification model.

The remaining reserved group, the validation set, was used to determine the accuracy of each classification model. To insure that each positive and negative group was used at least once in the validation set, we conducted 70 (14x5 = 70) training sessions for each classification model, and the prediction performance for each validation set was noted. The average prediction accuracy for the validation sets, i.e., the average of 70 different values, gives an estimate of the expected prediction accuracy of a classification model (Alpaydin, 2010).

Table 2 presents the average expected accuracies of the classification models for several values of the *vote threshold* parameter. The PGOS-enriched and SVM-TM Classifiers both did quite well, and the ensemble classifier combining their decisions was found to give the best performance compared to the other models. This classifier achieved 91% expected prediction accuracy with a *vote threshold* of 10. Because it gave the best performance, we chose this ensemble model to serve as ApicoAMP. Our experiments demonstrated that the use of the GO term enrichment process in feature encoding results in significantly better performance compared to the approach described in (Du, 2012). We attribute the poor performance of the motif classifiers to the cardinality of our training set. *Ab initio* motif discovery algorithms like MERCI and MEME tend to require a substantial amount of training data to avoid overfitting, i.e., to be capable of identifying motifs that are generalizable.

Table 3 lists the average expected accuracy of ApicoAMP for the 11 apicomplexan species that appear in our training sets along with their appearance rate in the test sets. One can observe that the appearance rate of a species in the training set is not correlated with the estimated prediction performance of ApicoAMP on the proteomes of these species, which indicates that ApicoAMP does not favor the most frequently appearing species in the training set, but instead it is able to capture the general characteristics of ApicoTMPs for multiple species. This is important because a bias in the results would indicate the possibility that using positive and negative training data from different species is insufficient for developing a prediction model.

All available apicomplexan proteins from 16 apicomplexan species were downloaded from EupathDB (version 2.16) (Aurrecoechea et al., 2010) and subjected to TMHMM and SignalP 3.0 to identify 16914 transmembrane proteins. ApicoAMP was used to predict putative ApicoTMPs from these apicomplexan proteins. This final ApicoAMP classifier was trained using all 5 groups of positive instances and 14 groups of negative instances, i.e., all the available training data. Following the same architectural principle we used in performance estimations, we trained 10 classification units using training subsets, each containing 56 positives and 56 negatives randomly selected from the set of 154. Table 4 presents the prediction statistics for each apicomplexan species using 10 as the value of the vote threshold. An additional spreadsheet shows the predicted ApicoTMPs in detail [see Supplementary data 3].

## 4. Discussion

The apicoplast is an essential organelle for a group of eukaryotic parasites known as Apicomplexa, which includes *Plasmodium falciparum*, the causative agent of the most deadly form of malaria. This organelle is important not only for the survival of the parasite, but its prokaryotic origin makes it an ideal drug target. As the gatekeepers of this important organelle, apicoplast membrane proteins are potentially excellent drug target candidates and, as such, their identification is important. Experimental identification of apicoplast membrane proteins is a costly and time-consuming task. Accurate *in silico* prediction methods are needed to accelerate the identification of promising drug targets. Unfortunately, no such prediction method exists.

With the publication of recent experimental findings on a subset of apicoplast membrane proteins, called transmembrane proteins, we were able to gather a reasonably sized training set that we utilized to develop a computational approach capable of identifying apicoplast-targeted transmembrane proteins (ApicoTMP). ApicoAMP is the first computational model that identifies ApicoTMPs in multiple species of Apicomplexa. Although the trafficking mechanisms involved in apicoplast membrane protein targeting have not been fully dissected, existing research on membrane localization prediction demonstrates the feasibility of finding emergent properties for specific membrane classes in a group of proteins regardless of the trafficking mechanisms used to reach their destinations. Such emergent properties have been utilized by existing machine learning approaches (Du, 2012; Du et al., 2012; Pierleoni et al., 2011; Sharpe et al., 2010) to successfully predict membrane localization of proteins. Moreover, several of these approaches used heterogeneous training sets for the destination membrane. For example, Pierleoni et al. (2011) combined proteins known to localize to either mitochondria or plastids in one training set that was used to predict proteins that localize to a class they defined as the organelle membrane class. Our treatment of the apicoplast membrane as a single class rather than as four separate classes, one for each of the four membrane layers, adheres to existing approaches reported in the literature. When a sufficient number of apicoplast membrane proteins localizing to a specific membrane layer have been identified, it will be possible to develop prediction methods with greater granularity.

In the development of ApicoAMP, we exploited the discovery by Sharpe et al. (2010) that the sequences of transmembrane domains (TMDs) reflect the different physical properties of various membranes of eukaryotic cells. The SVM-TM classifier trained using features extracted from the TMDs of apicomplexan proteins achieved 82% overall expected accuracy in the ApicoTMP prediction task, providing supporting evidence for this finding.

Du et al. (2012) demonstrated the merits of using Gene Ontology (GO) terms as descriptors of proteins with their classification algorithm PGOS. Their feature extraction strategy included an enrichment process of the GO term set of a given protein with the help of a sequence similarity search. We revised their method by introducing an e-value threshold in the sequence similarity search to ensure that only sufficiently similar sequences are used in the GO term set enrichment process. We also applied an additional GO term enrichment process to the database that is used in the sequence similarity search. The PGOS-enriched classifier trained using features calculated by our revised GO term enrichment procedure achieved 88% overall expected accuracy in the ApicoTMP prediction task.

ApicoAMP is an ensemble classification model that combines the decisions of the SVM-TM and PGOS-enriched classifiers. ApicoAMP is trained on a set of proteins from 11 apicomplexan species and achieves 91% overall expected accuracy. By design, ApicoAMP uses 10 classification units, each containing one SVM-TM and one PGOS-enriched classifier. Each unit has a single vote, which can either be ApicoTMP or non-ApicoTMP. If one of the classifiers indicates that a given protein is an ApicoTMP, the vote is given as ApicoTMP. If *n* of the 10 classification units vote for ApicoTMP, ApicoTMP is predicted as the label for a given protein. Here *n*, the *vote threshold*, is treated as a parameter in ApicoAMP and is set by the user.

ApicoAMP software allows users to set the *vote threshold* parameter during prediction. If a user wants to obtain minimal false positive predictions, this parameter should be set to a high value such as 9 or 10. If a user wants to obtain minimal false negative predictions, this parameter should be set to a low value such as 6 or 7.

In this paper we presented ApicoAMP, the first computational model capable of identifying ApicoTMPs in multiple species of Apicomplexa. In addition, we provide a user-friendly, Python-based program of the ApicoAMP classifier. We developed ApicoAMP with the idea of providing assistance to researchers in narrowing the number of candidates for laboratory validation with the expectation that they will choose the most likely candidates based on their expertise. Eventually it is hoped that there will be a sufficient number of known ApicoTMPs for a particular species to permit the development of a more robust prediction tool for the given species.

## 5. References

Alpaydin E., 2010. Introduction to Machine Learning. The MIT Press. pp 9.

Aurrecoechea C., Brestelli J., Brunk B.P., Fischer S., Gajria B., Gao, X., Gingle A., Grant G., Harb O.S., Heiges M., Innamorato F., Iodice J., Kissinger J.C., Kraemer E.T., Li W., Miller J.A., Nayak V., Pennington C., Pinney D.F., Roos D.S., Ross C., Srinivasamoorthy G., Stoeckert C.J., Thibodeau R., Treatman C., Wang H., 2010. EuPathDB: a portal to eukaryotic pathogen databases. Nucleic Acids Res. 38(Database issue), D415–9.

Bailey T.L., Bodén M., Whitington T., Machanick P., 2010. The value of position-specific priors in motif discovery using MEME. BMC Bioinformatics. 9(11), 179.

Bailey T.L., Williams N., Misleh C., Li W.W., 2006. MEME: discovering and analyzing DNA and protein sequence motifs. Nucleic Acids Res. 1(34), W369–73.

Bendtsen J.D., Nielsen H., von Heijne G., Brunak S., 2004. Improved prediction of signal peptides: SignalP 3.0. J Mol Biol. 340(4), 783–95.

Ben-Hur A., Weston J., 2010. A user's guide to support vector machines. Methods Mol Biol. 609. 223–39.

Blum T., Briesemeister S., Kohlbacher O., 2009. MultiLoc2: integrating phylogeny and Gene Ontology terms improves subcellular protein localization prediction. BMC Bioinformatics. 1(10), 274.

Camacho C., Coulouris G., Avagyan V., Ma N., Papadopoulos J., Bealer, K., Madden, T., 2009. BLAST+: architecture and applications. BMC Bioinformatics. 15(10), 421.

Chen F., Mackey A.J., Stoeckert C.J., Roos D.S., 2006. OrthoMCL-DB: querying a comprehensive multi-species collection of ortholog groups. Nucleic Acids Res. 34, D363–8.

Chi S.M., Nam D., 2012. WegoLoc: accurate prediction of protein subcellular localization using weighted Gene Ontology terms. Bioinformatics. 28(7), 1028–30.

Cilingir G., Broschat S.L., Lau A.O.T., 2012. ApicoAP: The First Computational Model for Identifying Apicoplast-Targeted Proteins in Multiple Species of Apicomplexa. PLoS ONE. 7(5).

DeRocher A.E., Coppens I., Karnataki A., Gilbert L.A., Rome M.E., Feagin, J.E., Bradley P.J., Parsons, M., 2008. A thioredoxin family protein of the apicoplast periphery identifies abundant candidate transport vesicles in Toxoplasma gondii. Eukaryot Cell. 7(9), 1518–29.

DeRocher A.E., Karnataki A., Vaney P., Parsons M., 2012. Apicoplast Targeting of a Toxoplasma gondii Transmembrane Protein Requires a Cytosolic Tyrosine-Based Motif. Traffic. 15(5), 694–704

Du P., 2012. Predicting Subcellular Localizations of Membrane Proteins in Eukaryotes with Weighted Gene Ontology Scores. Advances in Intelligent and Soft Computing. 124, 191–195

Du P., Tian Y., Yan Y., 2012. Subcellular localization prediction for human internal and organelle membrane proteins with projected gene ontology scores. J Theor Biol. 313, 61–7.

Engelman D.M., Steitz T.A., Goldman A., 1986. Identifying nonpolar transbilayer helices in amino acid sequences of membrane proteins. Annu. Rev. Biophys. Biophys. Chem. 15, 321–353.

Fleige T., Limenitakis J., Soldati-Favre D., 2010. Apicoplast: keep it or leave it. Microbes Infect. 12(4), 253–62.

Foth, B. J., Ralph, S. A., Tonkin, C. J., Struck, N. S., Fraunholz, M., Roos, D. S., Cowman A. F., McFadden, G. I., 2003. Dissecting Apicoplast Targeting in the Malaria Parasite Plasmodium falciparum. Science. 299, 705–708.

Hofmann K., Stoffel W., 1993. TMbase - A database of membrane spanning proteins segments. Biol Chem Hoppe-Seyler. 374, 166.

Huang W.L., Tung C.W., Ho S.W., Hwang S.F., Ho S.Y., 2008. ProLoc-GO: utilizing informative Gene Ontology terms for sequence-based prediction of protein subcellular localization. BMC Bioinformatics. 1(9), 80.

Joachims T., 1999. Making large-scale support vector machine learning practical. In Advances in kernel methods, B. Schölkopf and C. Burges and A. Smola (Eds.). MIT Press. pp. 169–184.

Karnataki A., DeRocher A.E., Coppens I., Feagin J.E., Parsons M., 2007a. A membrane protease is targeted to the relict plastid of toxoplasma via an internal signal sequence. Traffic. 8(11), 1543–53.

Karnataki A., DeRocher A.E., Coppens I., Nash C., Feagin J.E., Parsons, M., 2007b. Cell cycle-regulated vesicular trafficking of Toxoplasma APT1, a protein localized to multiple apicoplast membranes. Mol Microbiol. 63(6), 1653–68.

Krogh A., Larsson B., von Heijne G., Sonnhammer E.L., 2001. Predicting transmembrane protein topology with a hidden Markov model: application to complete genomes. J Mol Biol. 305(3), 567–80.

Li L., Zhang Y., Zou L., Li C., Yu B., Zheng, X., Zhou, Y., 2012. An ensemble classifier for eukaryotic protein subcellular location prediction using gene ontology categories and amino acid hydrophobicity. PLoS One. 7(1).

Li W., Godzik A., 2006. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. Bioinformatics. 22, 1658–1659.

Lim L., Kalanon M., McFadden G.I., 2009. New proteins in the apicoplast membranes: time to rethink apicoplast protein targeting. Trends Parasitol. 25(5), 197–200.

Mei S., Fei W., Zhou S., 2011. Gene ontology based transfer learning for protein subcellular localization. BMC Bioinformatics. 2(12), 44.

Michelsen K., Yuan H., Schwappach B., 2005. Hide and run. Arginine-based endoplasmic-reticulum-sorting motifs in the assembly of heteromultimeric membrane proteins. EMBO Rep. 6(8), 717–22.

Pierleoni A., Martelli P.L., Casadio R., 2011. MemLoci: predicting subcellular localization of membrane proteins in Eukaryotes. Bioinformatics. 27(9), 1224–1230.

Provost F., 2000. Machine learning from imbalanced data sets 101. Proceedings of the AAAI-2000 Workshop on Imbalanced Data Sets.

Sato K., Nakano A., 2002. Emp47p and its close homolog Emp46p have a tyrosine-containing endoplasmic reticulum exit signal and function in glycoprotein secretion in Saccharomyces cerevisiae. Mol Biol Cell. 13(7), 2518–32.

Sharpe H.J., Stevens T.J., Munro S., 2010. A comprehensive comparison of transmembrane domains reveals organelle-specific properties. Cell. 142(1), 158–69.

Sheiner L., Demerly J.L., Poulsen N., Beatty W.L., Lucas O., Behnke, M.S., White M.W., Striepen, B., 2011. A Systematic Screen to Discover and Analyze Apicoplast Proteins Identifies a Conserved and Essential Protein Import Factor. PLoS Pathog. 7(12).

Vapnik V., 1995. The Nature of Statistical Learning Theory. Springer, New York.

Vapnik V., 1998. Statistical Learning Theory. Wiley, New York.

Vens C., Rosso M.N., Danchin E.G., 2011. Identifying discriminative classification-based motifs in biological sequences. Bioinformatics. 27(9), 1231–1238.

von Heijne G., 1992. Membrane protein structure prediction. Hydrophobicity analysis and the positive-inside rule. J Mol Biol. 225(2), 487–94.

## Figure Legends

**Figure 1. Three subregions of a transmembrane domain (TMD).**
A TMD region is composed of 3 sub-regions: a hydrophobic core and pre- and post-TMD sub-regions that are aligned with the inner and outer leaflets of the membrane. When TMDs are aligned from the cytoplasmic side to the exoplasmic side, rather than N terminus to C terminus, pre-TMD and post-TMD regions are found on the cytoplasmic (c) and exoplasmic (e) end of the TMD region, respectively.

# Tables

**Table 1. Labeled datasets used for ApicoTMP prediction.**[1]

| **Apicomplexan Species** | **Putative ApicoTMPs** | **Putative non-ApicoTMPs** |
|---|---|---|
| *N. caninum* | 2 | 5 |
| *P. vivax* | 4 | 7 |
| *B. bovis* | 5 | 5 |
| *P. yoelii* | 4 | 3 |
| *T. parva* | 3 | 4 |
| *P. berghei* | 4 | 9 |
| *P. chabaudi* | 5 | 7 |
| *P. falciparum* | 13 | 75 |
| *P. knowlesi* | 5 | 7 |
| *T. gondii* | 8 | 28 |
| *T. annulata* | 3 | 4 |
| **Total** | 56 | 154 |

[1] Breakdown of the labeled datasets into positive (ApicoTMP) and negative (non- ApicoTMP) classes for 11 species of Apicomplexa.

**Table 2. Average expected accuracy of various classification models for the ApicoTMP prediction problem.**[2]

| **Vote Threshold/ Classifier** | **6** | **7** | **8** | **9** | **10** |
|---|---|---|---|---|---|
| PGOS-enriched & SVM-TM Ensemble Classifier | 0.868<br>(0.98, 0.76) | 0.888<br>(0.98, 0.80) | 0.903<br>(0.97, 0.84) | 0.903<br>(0.95, 0.86) | 0.911<br>(0.92, 0.90) |
| PGOS & SVM-TM Ensemble Classifier | 0.842<br>(0.94, 0.74) | 0.855<br>(0.92, 0.79) | 0.862<br>(0.89, 0.83) | 0.856<br>(0.85, 0.86) | 0.858<br>(0.82, 0.90) |
| PGOS-enriched Classifier | 0.875<br>(0.80, 0.95) | 0.876<br>(0.80, 0.95) | 0.873<br>(0.79, 0.95) | 0.866<br>(0.78, 0.95) | 0.860<br>(0.76, 0.96) |
| SVM-TM Classifier | 0.814<br>(0.83, 0.79) | 0.824<br>(0.81, 0.84) | 0.822<br>(0.76, 0.88) | 0.793<br>(0.68, 0.90) | 0.758<br>(0.58, 0.94) |
| PGOS-enriched, SVM-TM, & SVM-MERCI Ensemble Classifier | 0.849<br>(0.82, 0.88) | 0.841<br>(0.78, 0.90) | 0.827<br>(0.73, 0.92) | 0.809<br>(0.68, 0.94) | 0.767<br>(0.58, 0.96) |
| PGOS-enriched, SVM-TM, & SVM-MEME Ensemble Classifier | 0.834<br>(0.82, 0.85) | 0.834<br>(0.79, 0.88) | 0.831<br>(0.76, 0.90) | 0.814<br>(0.72, 0.91) | 0.789<br>(0.64, 0.94) |
| PGOS Classifier | 0.701<br>(0.47, 0.93) | 0.699<br>(0.46, 0.94) | 0.702<br>(0.46, 0.94) | 0.7<br>(0.45, 0.95) | 0.69<br>(0.42, 0.96) |
| SVM-MERCI Classifier | 0.63<br>(0.57,0.68) | 0.615<br>(0.51, 0.72) | 0.605<br>(0.44, 0.77) | 0.588<br>(0.37, 0.81) | 0.559<br>(0.27, 0.84) |
| SVM-MEME Classifier | 0.59<br>(0.58, 0.60) | 0.599<br>(0.57, 0.63) | 0.602<br>(0.54, 0.66) | 0.607<br>(0.53, 0.68) | 0.610<br>(0.50, 0.72) |

[2] Average expected accuracy of various classification models for the ApicoTMP prediction problem (true-positive and false-positive rates in parentheses) with different values of the vote threshold parameter. The table is sorted from best to worst performance.

**Table 3. Average expected accuracy of ApicoAMP for 11 apicomplexan species.**[3]

| **Apicomplexan Species** | **Average Expected Accuracy** | **Appearance Rate in Test Sets** |
|---|---:|---:|
| *P. falciparum* | 0.833 | 0.421 |
| *T. gondii* | 0.925 | 0.172 |
| *P. berghei* | 0.980 | 0.062 |
| *P. chabaudi* | 1.000 | 0.057 |
| *P. knowlesi* | 1.000 | 0.057 |
| *P. vivax* | 0.978 | 0.053 |
| *B. bovis* | 0.895 | 0.048 |
| *N. caninum* | 0.906 | 0.033 |
| *T. parva* | 0.935 | 0.033 |
| *T. annulata* | 0.774 | 0.033 |
| *P. yoelii* | 0.982 | 0.029 |

[3] Average expected accuracy of ApicoAMP for 11 apicomplexan species that appear in our training set together with their appearance rate. The value of the vote threshold parameter is set to 10 for this analysis.

**Table 4. ApicoAMP predictions for 16 apicomplexan species.** [4]

| **Apicomplexan Species** | **Total Transmembrane Proteins** | **ApicoAMP Positive Predictions** |
|---|---:|---:|
| *T. gondii* | 1441 | 378 |
| *P. chabaudi* | 1178 | 376 |
| *P. berghei* | 1178 | 365 |
| *B. bovis* | 591 | 111 |
| *P. falciparum* | 1400 | 536 |
| *C. muris* | 694 | 154 |
| *T. parva* | 624 | 159 |
| *T. annulata* | 714 | 195 |
| *N. caninum* | 1188 | 265 |
| *P. knowlesi* | 1018 | 318 |
| *P. yoelii* | 2099 | 634 |
| *E. tenella* | 1261 | 295 |
| *C. parvum* | 660 | 139 |
| *C. hominis* | 619 | 132 |
| *P. cynomolgi* | 1118 | 319 |
| *P. vivax* | 1131 | 292 |
| **Total** | 16914 | 4668 |

[4] ApicoAMP predictions for 16 apicomplexan species. The value of the vote threshold parameter is set to 10 for this analysis.

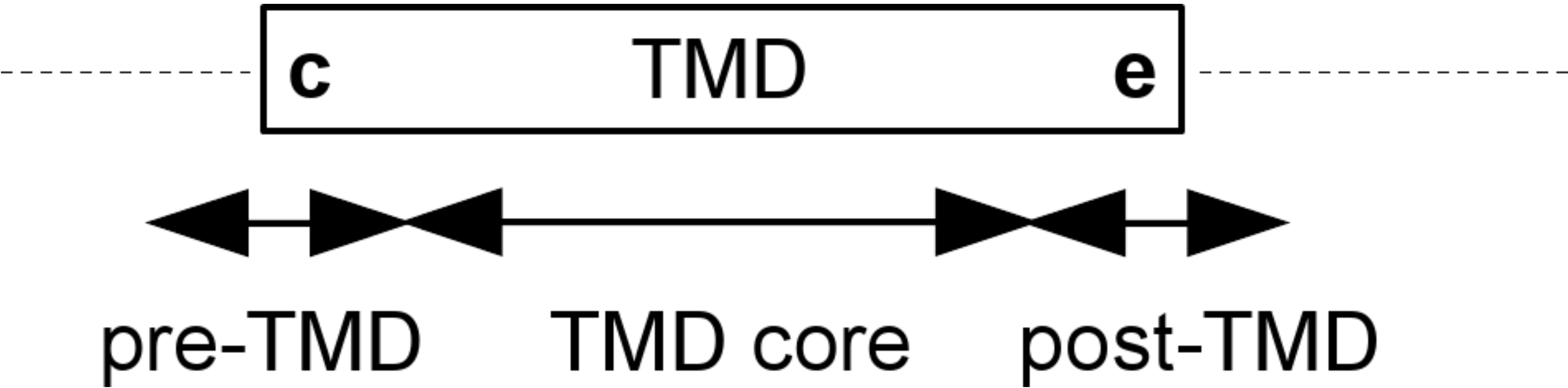
## Supplementary data

**Supplementary data 1:** Positive training set used for developing ApicoAMP.

**Supplementary data 2:** Negative training set used for developing ApicoAMP.

**Supplementary data 3:** List of putative ApicoTMPs for 16 apicomplexan species based on ApicoAMP predictions.

Figure(s)


c
TMD
e
pre-TMD
TMD core
post-TMD