COMPUTATIONAL APPROACHES FOR THE PREDICTION OF APICOPLAST-TARGETED PROTEINS

by

GOKCEN CILINGIR

A dissertation submitted in partial fulfillment of
the requirements for the degree of

DOCTOR OF PHILOSOPHY

WASHINGTON STATE UNIVERSITY
Department of Electrical Engineering and Computer Science

MAY 2013

To the Faculty of Washington State University:

The members of the Committee appointed to examine the dissertation of GOKCEN CILINGIR find it satisfactory and recommend that it be accepted.

Shira L. Broschat, Ph.D., Chair

Audrey O. T. Lau, Ph.D.

Lawrence Holder, Ph.D.

ACKNOWLEDGMENTS

I would like to state my sincere gratitude to my adviser and mentor Shira L. Broschat, for her irreplaceable guidance, everlasting support and invaluable suggestions throughout the development of this dissertation. I deeply thank Audrey O.T. Lau for the guidance she provided in Apicomplexan biology, and for believing in us in every step of the way. I would like to thank Lawrence Holder, for his invaluable suggestions and questions, that shaped many parts of this dissertation. I would like to thank Svetlana Lockwood, Roben (Yunbing) Tan, and Yunyun Zhou, whom I considered to be a part of my academic family, for the moral support they provided.

COMPUTATIONAL APPROACHES FOR THE PREDICTION OF APICOPLAST-TARGETED PROTEINS

Abstract

by Gokcen Cilingir, Ph.D.
Washington State University
May 2013

Chair: Shira L. Broschat

Motivation: The cells of eukaryotic organisms contain subunits called organelles. The apicoplast is a unique organelle found in a group of parasites, known as Apicomplexa, that are responsible for a wide range of serious diseases including malaria. The apicoplast is an ideal drug target because of its unique properties. Identifying apicoplast-targeted proteins (ATPs) is necessary for drug target identification and accurate *in silico* prediction methods are needed to accelerate this process. Current computational approaches concentrate on a single species of Apicomplexa and are capable of predicting only a subset of ATPs.

Methodology: We have developed two new computational approaches, ApicoAP and ApicoAMP, that concentrate on different types of ATPs and that are applicable to multiple species of Apicomplexa. ApicoAP is a generalized rule-based

classification model. In ApicoAP, we conduct a systematic search over a rule space using the expected prediction performance of a rule on a training set as the optimization criterion. The rule space is formalized by our parametric rule definition. We devised a genetic algorithm to perform the optimization that results in a classification rule. Performance of ApicoAP is evaluated for labeled datasets of proteins from 4 different apicomplexan species, and expected prediction accuracies range between 82%, and 87%. ApicoAMP is an ensemble classification model. In ApicoAMP, different algorithms and feature sets are used to train several classifiers that are evaluated and combined in an ensemble classification model to obtain the best expected performance. ApicoAMP is trained on a set of proteins from 11 apicomplexan species, and its expected prediction accuracy is found to be 91%. In addition, we developed ApicoAP Pipeline, where we introduced an automated training data gathering procedure. This pipeline works as an automated ApicoAP classifier generator that does not require training data to be provided, but instead is capable of generating a classifier from the information available from public resources at a given time.

Conclusions: Our work significantly broaden the set of apicoplast-targeted proteins that can be identified computationally. The ApicoAP and ApicoAMP prediction software and ApicoAP Pipeline client software are available for public use at http://bcb.eecs.wsu.edu.

TABLE OF CONTENTS

ACKNOWLEDGMENTS iii

ABSTRACT iv

LIST OF TABLES viii

LIST OF FIGURES ix

1. Introduction 1

    1.1 Background 1
    1.2 Motivation 2
    1.3 Problem Definition and Principal Findings 4
    1.4 Organization of the Dissertation 6

2. ApicoAP 7

    2.1 Introduction 7
    2.2 Materials and Methods 10
    2.3 Datasets 22
    2.4 Results 25
    2.5 Discussion 34

3. ApicoAMP 39

    3.1 Introduction 39
    3.2 Methods 44
    3.3 Results 59
    3.4 Discussion 62

4. ApicoAP Pipeline 69

    4.1 Introduction 69

4.2 ApicoAP Model 71
4.3 The ApicoAP Pipeline 72
4.4 ApicoAP-CS Results 79
4.5 Discussion 80

5. Conclusion 86

LIST OF TABLES

Table		Page
2.1	Breakdown of the labeled datasets into positive (ApicoTP) and negative (non-ApicoTP) classes.	24
2.2	Averaged expected prediction performance of ApicoAP (standard deviation (sd) in parentheses) for the labeled datasets.	26
2.3	ApicoAP classifier performance on the labeled datasets.	27
2.4	Comparison of ApicoAP and PlasmoAP for *P. falciparum* dataset of 78 positives and 27 negatives.	27
2.5	Comparison of ApicoAP model with various machine learning algorithms.	29
2.6	ApicoAP predictions for SP-containing *P. falciparum*, *B. bovis*, *T. gondii*, and *P. yoelii* proteins.	30
3.1	Label datasets used for ApicoTMP prediction.	46
3.2	Average expected accuracy of various classification models for the ApicoTMP prediction problem with different values of the *vote threshold*(vt) parameter.	66
3.3	Average expected accuracy of ApicoAMP for 11 apicomplexan species.	67
3.4	ApicoAMP predictions for 16 apicomplexan species.	68
4.1	Cardinalities of the positive interim training sets for the 13 apicomplexan species gathered by ApicoAP-CS.	82
4.2	Cardinalities of the negative interim training sets for the 13 apicomplexan species gathered by ApicoAP-CS.	83
4.3	Cardinalities of the final training sets for the 13 apicomplexan species.	84
4.4	ApicoTP classifier performances on the training sets.	85

LIST OF FIGURES

Figure Page

2.1 Schematic representation of a typical apicoplast-targeted protein (ApicoTP). 13

2.2 Averaged frequency distributions of preferred and avoided residues for the p regions of the training sequences. 37

2.3 Training data mapped onto the *PRSS-ARSS* plane using final ApicoAP classifiers. 38

3.1 Three subregions of a transmembrane domain (TMD). 51

4.1 ApicoAP Pipeline. 74

Dedication

This dissertation is dedicated to my parents Cemalettin Alay and Saniye Alay and to my husband Erdem Cilingir for their everlasting support and encouragement.

CHAPTER 1. INTRODUCTION

1.1 Background

In biology, all living organisms can be classified as eukaryotes or prokaryotes depending on the fundamental structure of their cells. Eukaryotes such as humans contain cells that have membrane-bound subunits with specialized functions, known as organelles. On the other hand, prokaryotes such as archaea and bacteria lack specialized compartments in the cell.

Proteins are biochemical compounds that are made up of amino acids, often called peptides when several are linked together. Gene sequences are translated into proteins via the genetic code. The code defines how a unit sequence, called a codon, will be translated into a single amino acid. Gene sequences are represented by an alphabet of four (A, C, G and T) and protein sequences are represented by an alphabet of 20.

1.2 Motivation

The apicoplast is a unique organelle found in a group of parasites known as Apicomplexa. Apicomplexa are responsible for a wide range of serious diseases of humans and livestock including the most deadly form of malaria, caused by the species called *Plasmodium falciparum*. As resistance to commonly used drugs is increasing in apicomplexan parasites, it is important to find new drug targets. The apicoplast is an essential organelle for the survival of these parasites [Fichera and Roos, 1997, He et al., 2001]. Moreover, many apicoplast proteins and pathways have prokaryotic characteristics due to the organelle's ancestral relationship to bacteria [McFadden, 1996, Ralph et al., 2004]. Because these proteins and pathways are either absent or divergent from those of its eukaryotic host (e.g., humans), they are seen as promising drug targets with minimum side effects to the infected host [McFadden and Roos, 1999, Ralph et al., 2004]. Understanding the metabolic activities performed in the apicoplast is essential for drug target identification, and this requires the ability to identify apicoplast-targeted proteins. Because experimental identification of these proteins is a costly and time-consuming task, accurate *in silico* prediction methods are needed to accelerate the drug target identification process.

The available computational approaches [Zuegge et al., 2001, Foth et al., 2003] for genome-wide apicoplast-targeted protein prediction are specifically designed for

P. falciparum species and their application to other Apicomplexa is considered to be unreliable. With the sequence completion of several apicomplexan genomes, there is a pressing need for computational methods to detect apicoplast-targeted proteins that are applicable to multiple species rather than to a single model species. Currently, genomes for 17 different apicomplexan species are available in EuPathDB [Aurrecoechea et al., 2010].

Available computational approaches [Zuegge et al., 2001, Foth et al., 2003] concentrate on predicting a subset of apicoplast-targeted proteins which contain a special segment called a bipartite signal. Recent experimental findings have confirmed many apicoplast-targeted membrane proteins, which have been found to lack a bipartite signal [Karnataki et al., 2007b, DeRocher et al., 2008, Sheiner et al., 2011]. Most of these findings apply to a subset of apicoplast membrane proteins that are called transmembrane proteins. These proteins contain transmembrane domains (TMDs) that reside in the membrane and function as membrane anchors. Although well-established prediction algorithms exist for TMD topology prediction, there is no computational approach in the literature that identifies transmembrane proteins targeted to the apicoplast. In fact, at present, only a handful of methods developed specifically for membrane localization prediction exist in the literature.

1.3 Problem Definition and Principal Findings

Apicoplast-targeted proteins can be classified into two main groups: those that are localized into the apicoplast lumen (ApicoTPs for *APICO*plast *T*argeted lumen *P*roteins) and those that reside on or between the four membranes of the apicoplast (ApicoTMPs for *APICO*plast *T*argeted trans*M*embrane *P*roteins). Since the properties of these two groups of proteins differ in many ways, we decided to divide the apicoplast-targeted protein identification task into two parts, developing two independent approaches for each sub-task. Each approach involved building a classification model and training it using labeled datasets. Trained models (i.e., classifiers) label a given protein as apicoplast targeted or not.

We developed ApicoAP for *APICO*mplexan *A*picoplast lumen *P*roteins, which is the first computational model for identifying ApicoTPs in multiple species of Apicomplexa [Cilingir et al., 2012]. ApicoAP is a generalized rule-based classification model. In ApicoAP, we conduct a systematic search over a rule space using the expected prediction performance of a rule on a training set as the optimization criterion. The rule space is formalized by our parametric rule definition. We devised a genetic algorithm to perform the optimization that results in a classification rule. Performance of ApicoAP is evaluated for labeled datasets of proteins from 4 different apicomplexan species, and expected prediction accuracies range between 82%, and

87%. The ApicoAP prediction software is available at http://bcb.eecs.wsu.edu.

We developed ApicoAMP for *APICO*mplexan *A*picoplast trans*M*embrane *P*roteins, which is the first computational model capable of identifying apicoplast-targeted transmembrane proteins in Apicomplexa. ApicoAMP is an ensemble classification model. In ApicoAMP, different algorithms and feature sets are used to train several classifiers that are evaluated and combined in an ensemble classification model to obtain the best expected performance. Hydrophobicity and composition characteristics of amino acids over TMDs are used as features in conjunction with the Support Vector Machine (SVM) classification model. In addition, we extended and employed the Projected Gene Ontology Score (PGOS) classification model which is a specialized model used with the Gene Ontology (GO) terms associated with proteins. ApicoAMP is trained on a set of proteins from 11 apicomplexan species, and its expected prediction accuracy is found to be 91%. The ApicoAMP prediction software is available at http://bcb.eecs.wsu.edu.

After publishing our paper on ApicoAP, we received many inquiries from researchers working on different apicomplexan species for which no ApicoAP classifier is provided by our software. In order to fulfill the current demand and any future demands, we developed the ApicoAP Pipeline that is comprised of an automated training data gathering procedure and the ApicoTP classifier training routine. This pipeline works as an automated ApicoTP classifier generator that does not require

training data to be provided, but instead is capable of generating a classifier from the information available from public resources at a given time. As the results from experimental confirmation of ApicoTPs are published, which is the main resource for obtaining training data, this pipeline will not only be useful for an apicomplexan species for which no ApicoAP classifier exists, but it will also provide ever-improving classifiers for apicomplexan species for which an ApicoAP classifier already exists. An implementation of this pipeline, ApicoAP-CS for *ApicoAP* *C*omplete *S*uite, is available as a collection of web services. ApicoAP-CS can be utilized to generate a species-specific ApicoAP classifier that can be easily integrated into the ApicoAP prediction software.

1.4 Organization of the Dissertation

This thesis is organized in five main chapters, including this introduction chapter as the first chapter. Second and third chapters discusses the ApicoAP and ApicoAMP models, respectively. Forth chapter discusses a new model of operation for specific supervised machine learning algorithms that learn from datasets extracted from dynamically changing public resources, such as genomic databases. In this chapter, ApicoAP Pipeline is discussed as a case study. The fifth and the last chapter contains the conclusion.

CHAPTER 2. APICOAP

2.1 Introduction

The apicoplast is a relict plastid that resides in most of the parasites of the phylum Apicomplexa [McFadden, 1996, Köhler et al., 1997]. Members of this phylum include *Plasmodium falciparum*, the causative agent of the most deadly form of malaria, *Plasmodium yoelii*, another malaria-causing agent, and *Toxoplasma gondii* and *Babesia bovis*, which cause toxoplasmosis and babesiosis, respectively. The apicoplast is an essential organelle for the survival of these parasites [Fichera and Roos, 1997, He et al., 2001]. Moreover, many apicoplast proteins and pathways have prokaryotic characteristics due to the organelle's ancestral relationship to bacteria [McFadden, 1996, Ralph et al., 2004]. Because these proteins and pathways are either absent or divergent from those of its eukaryotic host, they are seen as promising drug targets with minimum side effects to the infected host [McFadden and Roos, 1999, Ralph et al., 2004]. Most apicoplast proteins are nuclear-encoded and targeted post-translationally to the organellar lumen [Waller et al., 1998, Roos et al., 1999, Waller et al., 2000, Van Dooren et al., 2000]. Understanding the metabolic activities

performed in the apicoplast is essential for drug target identification, and this requires the ability to detect apicoplast targeting signals in proteins.

Protein import into the lumen of the apicoplast is facilitated by a bipartite signaling mechanism that requires an N-terminal signal peptide (SP) followed by a transit peptide (TP) [Waller et al., 2000]. Although other mechanisms may exist [Lim et al., 2009], the bipartite signaling mechanism is most easily recognized. Well-established prediction algorithms exist for determining the existence of an SP in a protein sequence independent of the organism to which it belongs [Petersen et al., 2011, Emanuelsson et al., 2007, Reynolds et al., 2008, Käll et al., 2007]. In contrast, there is no established computational method that determines the existence of a TP in multiple organisms. In fact, attempts to define a consensus motif that universally identifies apicoplast TPs have failed because preferred amino acids in TP regions are heavily influenced by the Adenine-Thymidine (AT) codon bias of parasitic genomes [Tonkin et al., 2008]. For example, the genome of *P. falciparum* is approximately 80% AT-enriched [Tonkin et al., 2008], and apicoplast TPs are dominated by amino acids such as asparagine (N) and lysine (K), which exclusively utilize codons lacking Guanine and Cytosine. PlasmoAP, a rule-based prediction method, makes use of this bias and suggests that the anticipated TP region (defined as the region that starts after the predicted SP-cleavage site with a cutoff of 80 amino acids) of apicoplast-targeted proteins (ApicoTPs) must contain an NK-enriched sub-region with a basic to

acidic amino acid ratio of at least 5 to 3 [Foth et al., 2003]. Application of this method to other Apicomplexa with more balanced AT content is not considered reliable. As a result, application of PlasmoAP to the *Babesia bovis* genome revealed only a handful of candidate ApicoTPs in comparison to more than 460 predicted ApicoTPs in *P. falciparum* [Brayton et al., 2007]. With the sequence completion of several apicomplexan genomes, there is a pressing need to have a computational method for detecting ApicoTPs that is applicable to different organisms rather than to a single model organism.

PATS [Zuegge et al., 2001] and PlasmoAP [Foth et al., 2003] are the only computational methods described in the literature that detect TP regions in protein sequences. These two methods are specifically designed for the *P. falciparum* proteome. PATS follows a black-box approach that is based on training a neural network over amino acid content-based features harvested from the anticipated TP region (defined as the region that starts after the predicted SP-cleavage site with a cutoff of 78 amino acids). Unlike PlasmoAP, PATS offers predictions only, without providing any understanding of the actual prediction mechanism. As a rule-based method, PlasmoAP holds an advantage over PATS in the sense that it offers insight into the underlying targeting mechanism and allows the formulation of testable hypotheses.

In this study, we propose a generalized rule-based classification model to identify ApicoTPs that use a bipartite signaling mechanism. Based only on the known

characteristics of ApicoTPs, a parametric model is constructed. Given a training set specific to an organism, our model, ApicoAP for *APICO*mplexan *A*picoplast *P*roteins, employs a procedure based on a genetic algorithm to tailor a discriminating rule that maximizes the prediction and generalization performance for the given set. An advantage of ApicoAP is that it is customizable to different organisms when training data are available.

2.2 Materials and Methods

2.2.1 Selection of a classification model

From a computational point of view, the prediction of a given protein as an ApicoTP or non-ApicoTP can be stated as a binary classification problem, for which we choose ApicoTP as the positive class. It is worth noting that we define the ApicoTP class such that proteins localizing to multiple organelles including the apicoplast are members of this class in addition to proteins localizing only to the apicoplast. In a typical supervised learning setting, a training set containing positive and negative labeled instances is used to learn a mapping from the input to the output. In our case, the goal is to learn a mapping from protein sequences to the binary class labels: ApicoTP and non-ApicoTP. Our machine learning approach towards this goal is to assume a parametric model to define this mapping and estimate model parameters

using a training set such that the error for parameter estimates is minimized. This estimation process is often called training. As a result of training, a model with specific parameters, in other words a classifier, is achieved, which can then be employed to predict the labels for new instances [Alpaydin, 2004].

After some consideration, we chose a rule-based approach, similar to the one used by the developers of PlasmoAP [Foth et al., 2003], as the basis for our classification model. Properties of ApicoTPs were used to construct a generalized rule defined by a set of parameters. After completion of training by means of a genetic algorithm, the resulting classifier was then used to predict a protein sequence as ApicoTP or non-ApicoTP. Before explaining the details of our generalized rule definition, we will discuss the known properties of ApicoTPs that underlie our model.

Properties of apicoplast-targeted proteins (ApicoTPs).

A typical nuclear-encoded ApicoTP contains an N-terminal signal peptide (SP) region followed by a transit peptide (TP) region and a mature protein. The SP is removed during co-translational import into the endoplasmic reticulum (ER) and the TP, which guides the protein into the apicoplast, is removed from the mature protein inside the lumen of the apicoplast [Waller et al., 2000, van Dooren et al., 2002].

Apicoplast TPs vary greatly in length and are biased towards polar (positive charge preferred), basic, and hydrophilic amino acids [Foth et al., 2003, Tonkin et al., 2006]. A recent study conducted by [Gallagher et al., 2011] indicates that TPs are

functionally disordered and therefore biased towards amino acids with low helical propensity as well. In addition, it has been shown that the absence of negative charge, in other words the depletion of acidic residues, is important for transit peptide fidelity [Foth et al., 2003, Tonkin et al., 2006].

Length variance among TP regions of known ApicoTPs points to the possibility that a smaller sub-region of a perhaps larger TP is used by the apicoplast for recognition. This smaller sub-region (hereafter referred to as the pattern p) can be expected to embody the aforementioned properties of TP regions. PlasmoAP makes use of this idea by searching for a stretch of 40 amino acids in the anticipated TP region (with a cutoff of 80 amino acids) that is enriched and depleted by certain amino acid groups. Selection of these amino acid groups and cutoff values was performed only for the model organism, *P. falciparum*, which is the main limitation of PlasmoAP for other organisms.

Generalized model for apicoplast-targeted proteins (ApicoTPs).

A schematic representation of a typical ApicoTP is given in Figure 2.1. Because the TP region can be variable in length and in most cases its exact length is unknown, the region r is introduced, which represents the anticipated TP region. The region r starts immediately after the predicted SP cleavage site and has a length of at most L_r. A pattern p with length L_p is assumed to exist in region r, which contains the core information that indicates whether the protein under consideration is an ApicoTP.

The pattern p is simply a contiguous sub-region of region r enriched by amino acids that have low helical propensity or are polar (positive charge preferred), basic, or hydrophilic and depleted of acidic and negative amino acids. H, K, R are the amino acids that are polar-positive, basic, and highly hydrophilic. N, Q are the amino acids that are polar-neutral and highly hydrophilic. S, P, Y are moderately hydrophilic amino acids that have low helical propensity. We refer to these eight amino acids as the *preferred residue set* (PRS). E, D are the amino acids that are polar-negative and acidic with high helical propensity. We refer to these as the *avoided residue set* (ARS). We determined these sets using Chou-Fasman [Fasman, 1989] helical propensity predictions and the Kyte-Doolittle [Kyte et al., 1982] hydropathy index.

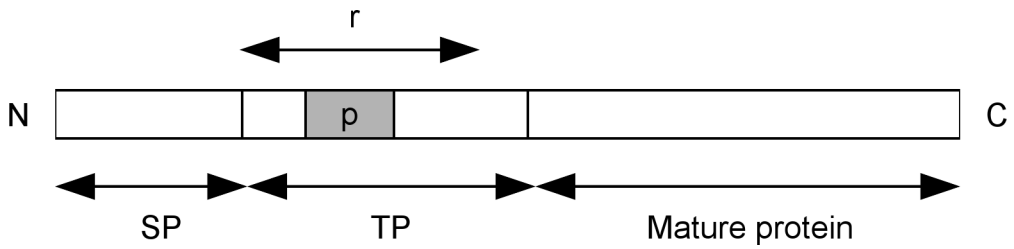

Figure 2.1: Schematic representation of a typical apicoplast-targeted protein (ApicoTP): A typical ApicoTP with defined regions r and p is shown, where r is the anticipated TP region that starts immediately after the predicted SP cleavage site and p is the pattern that contains the core information for predicting an ApicoTP. The pattern p is simply a contiguous sub-region of region r.

The *preferred residue set score* ($PRSS$) and *avoided residue set score* ($ARSS$) quantify the existence of PRS and ARS elements in an arbitrary region s. Equations

(2.1) and (2.2) give the functional forms of these quantities, where $f(x, s)$ is the frequency of an amino acid residue x in the region s. The *PRSS* and *ARSS* are simply the weighted sums of these frequencies. The weight sets $\mathbf{w_1}$ and $\mathbf{w_2}$ determine the relative influence of the residues in the scoring functions. When a weight is 0, the frequency of the corresponding residue will have no effect on the score, and when it is 1, it will have the maximum effect.

$$PRSS(s, \mathbf{w_1}) = \sum_{i=1}^{8} \mathbf{w_{1i}} * f(\mathbf{X_{1i}}, s),\ \mathbf{X_1} = \{H, K, R, N, Q, S, P, Y\} \tag{2.1}$$

$$ARSS(s, \mathbf{w_2}) = \sum_{i=1}^{2} \mathbf{w_{2i}} * f(\mathbf{X_{2i}}, s),\ \mathbf{X_2} = \{D, E\} \tag{2.2}$$

As stated earlier, the anticipated TP region r is assumed to contain a contiguous sub-region p with length L_p that embodies the core information for identifying an ApicoTP. We refer to the set containing all contiguous sub-regions with length L_p in r as S_p. In an ApicoTP, p should have a high *PRSS* and a relatively low *ARSS*. Assuming a linear relationship between the *PRSS* and *ARSS*, the *p-criterion* function given by Eq. (2.3) defines the criterion for selecting p from S_p. Essentially the sub-region with the highest ratio of preferred residues to avoided residues is the optimum choice.

$$p - criterion(r, lv, \mathbf{w_1}, \mathbf{w_2}) = \arg\max_{s \in S_p} \frac{PRSS(s, \mathbf{w_1}) - lv}{ARSS(s, \mathbf{w_2})} \tag{2.3}$$

The limiting value lv is an estimate of the $PRSS$ when e percent of the residues in a region s of length L_s are from the *preferred residue set* (PRS). The reason for including this limiting value is to ensure that a minimum number of elements from the PRS are present in the sub-region p. Sole absence of avoided residues is insufficient for a protein to be an ApicoTP; a minimum number of preferred residues are required as well. Equation (4) gives the functional form of lv.

$$lv(e, L_s, \mathbf{w_1}) = e * L_s * average(\mathbf{w_1}) \tag{2.4}$$

A rule-based classification model for ApicoTPs.

The generalized model for ApicoTPs discussed above defines a mapping from protein sequences to *p-criterion* values. In order to use this model as a classifier, a threshold value over *p-criterion* values that separates ApicoTPs from non-ApicoTPs must be determined. This is accomplished via feedback from the training set. We examine possible locations for the threshold and select the one that maximizes the prediction performance of the resulting classifier for the training set. The possible locations for the threshold are the midpoints of each adjacent pair of *p-criterion* values in sorted order. The resulting rule-based classifier classifies a protein sequence with a *p-criterion* value exceeding or equal to the threshold as an ApicoTP.

Geometric interpretation of the classification model for ApicoTPs.

The *PRSS* and *ARSS*, given by Eqs. (2.1) and (2.2), respectively, associated with the sub-region p for a given protein sequence map the sequence to a plane in which a discriminating line separates ApicoTPs and non-ApicoTPs. Protein sequences are mapped to a point in the *PRSS*-*ARSS* plane where the ones appearing on or above the discriminating line are predicted to be ApicoTPs. The limiting value *lv*, given by Eq. (2.4), determines the *PRSS*-intercept of the discriminating line. The threshold over *p-criterion* values, which is determined via feedback from the training set, gives the slope of this line.

If the *ARSS* is zero and the *PRSS* is greater than or equal to the limiting value *lv*, a sequence should be mapped to the ApicoTP region of the *PRSS*-*ARSS* plane, but the *p-criterion* value is undefined because the denominator in Eq. (2.3) is zero. For such cases, we set the *p-criterion* to be sufficiently large to ensure mapping of the sequence into the ApicoTP region. When the *PRSS* is smaller than *lv* and the *ARSS* is zero, the *p-criterion* is set sufficiently low to ensure mapping of the sequence into the non-ApicoTP region below the discriminating line.

The parameters for the rule-based classification model used in ApicoAP, including the weights, L_p, L_r, and e, are optimized using a genetic algorithm as described below, but before discussing our optimization method we discuss another requirement for identifying an ApicoTP with a bipartite signaling mechanism, the presence of a

signal peptide.

Signal peptide identification.

Implicit in our generalized model is that an ApicoTP contains an SP because the anticipated TP region r starts from the predicted SP cleavage site. We used SignalP 3.0 [Bendtsen et al., 2004] for SP cleavage site prediction, as it is the tool commonly reported in the literature for apicomplexan genomes. We considered using the most recent version of this tool, SignalP 4.0 [Petersen et al., 2011], which is believed to perform better at discriminating SP regions from transmembrane domains existing downstream from the N terminus of a sequence. However, we observed that SignalP 4.0 predicts significantly fewer SPs than SignalP 3.0 for apicomplexan genomes. For example, according to SignalP 3.0 the *P. falciparum* genome contains about 1100 SPs, but SignalP 4.0 identifies only about 600 SPs. Neither of these tools is trained or tested on apicomplexan genomes because no apicomplexan protein has been experimentally confirmed to contain an SP. Further study is needed on apicomplexan genomes to assess the possible causes for the difference in the number of predictions.

2.2.2 Optimizing model parameters

A prediction performance measure calculated with a given labeled dataset demonstrates how well the classification model performs on the available data, but it does

not predict how well a classifier can be expected to perform in practice. Instead, for our optimization criterion we use the *expected* prediction performance of a model, i.e., how well it is expected to generalize to new data instances; this can be estimated using a cross-validation procedure. In n-fold cross validation, a given dataset is randomly divided into n subsets of equal size. A classifier is trained n times by setting aside one distinct set for validation and using the remaining n-1 sets for training. The average prediction performance for the validation sets gives an estimate of the expected prediction performance of the classifier [Alpaydin, 2004].

We use Matthews Correlation Coefficient (MCC) as our performance measure; the MCC is known as a balanced measure because it weights a true positive prediction and a true negative prediction equally regardless of how imbalanced a test set might be [Baldi et al., 2000]. The more commonly used performance measure, accuracy, is biased toward classifiers that tend to do better on the majority class. The rule-based classification model used in ApicoAP requires several parameters: the weights that are used to calculate the $PRSS$ and $ARSS$, the region length L_r, the pattern length L_p, and the limiting percentage e from which the limiting value lv is determined. An optimization procedure based on a genetic algorithm is applied to determine the set of parameters that produces the model with the maximum expected prediction performance. The problem of choosing the best classification model parameters among all possibilities is characterized as a search problem in which the parameter space

is examined using the expected prediction performance as the objective function, calculated using the MCC measure.

A brief overview of genetic algorithms.

A genetic algorithm (GA) is a heuristic search method inspired by Darwinian evolution [Holland, 1992b]. Based on the principle of survival of the fittest, a GA maintains a set of candidate solutions called individuals, represented by a set of genes, and applies combination and transformation operations on individuals analogous to crossover and mutation operations in actual genes. A typical iteration for a GA involves selection of the fittest individuals (solutions with highest objective function values), application of the crossover operation to these individuals, generation of random mutations within the newly produced individuals (offspring), and replacement of a percentage of the total population by these offspring. This simulation of evolution on solution instances undergoes several iterations until the stop condition is reached. At this point, the algorithm returns the optimal solution achieved via the iterations.

The power of genetic algorithms comes from the employment of fitness-based selection and genetic operators (crossover and mutation) during reproduction [Kelly and Davis, 1991]. Fitness-based selection of individuals for reproduction enables the fittest ones to have offspring via the crossover operator, which enables the exchange of genetic information between parents. If we assume that each individual ideally captures different features of the global optima, combining subparts of these indi-

viduals from multiple parents on a single offspring greatly speeds up the process of reaching optima. This phenomenon is known as *implicit parallelism* in a GA [Holland, 1992a, Mitchell et al., 1994]. The mutation operator introduces localized changes in offspring, which is essential for sustaining exploration in the search space. Mutations introduce the genetic diversity that is not necessarily represented in a population but that may be needed to reach a global optimum.

Many variations of GAs exist in the literature. One can maintain a single population or multiple populations in parallel. If multiple populations are evolved in parallel, migration among them during each iteration can be allowed either for the fittest or for random individuals. At each iteration, the next population may or may not overlap with the previous one.

The genetic algorithm for ApicoAP.

In the genetic algorithm used in ApicoAP, an individual is represented by a real-valued parameter set containing ten weights, one region length L_r, one pattern length L_p, and one limiting percentage parameter e. To simplify the problem, we introduced constraints on the possible values of each parameter. Weight values can be 0, 0.5, or 1. Region length values can be between 60 and 90 with increments of 5. Pattern length values can be between 15 and 40 with increments of 1. Limiting percentage values can be between 0.2 and 0.4 with increments of 0.05. All ranges were determined by experimentation with the training portion of the available data.

Experiments conducted with longer region and pattern lengths did not result in significant differences in the rules or performance indicating that the lengths chosen are sufficient.

Uniform crossover and point mutation were defined, and the initial crossover and mutation probabilities were chosen to be 1.0 and 0.1, respectively. Four parallel populations containing 40 individuals were used, and migration was allowed (at each iteration) for the two fittest individuals. Populations were set to be overlapping where 15 individuals were replaced by the newly generated offspring at all iterations. A large number of populations with many individuals are desirable, but efficiency in the computational time required for optimization is also a concern. The replacement percentage and migration limit often determine how quickly population diversities converge to zero, but reaching this state too quickly is undesirable because a local optimum rather than a global optimum is likely to be reached. Maintenance of diverse populations is important for increasing the likelihood of reaching the global optimum of the search space. Thus, in determining parameters there is a tradeoff between time efficiency and maintenance of diverse populations.

To avoid local optimum traps, we implemented a mechanism to monitor population diversities and took preventive action when needed by gradually increasing the mutation rate and by changing the crossover selection criterion from fittest to random. When 30 generations had passed without achieving an improvement in the

optimal solution, we stopped the search. Although additional mechanisms were implemented to avoid local optimum traps, several runs were performed to insure an optimal solution had been reached.

2.3 Datasets

To evaluate the performance of ApicoAP, we used five labeled sets of protein sequences from *P. falciparum*, *P. yoelii*, *B. bovis*, and *T. gondii*, each containing sequences of a single organism. We used the published dataset employed in the development of PlasmoAP [Foth et al., 2003] for the sole purpose of comparing our method with theirs. In addition, we gathered a new training set for *P. falciparum* proteins that incorporates recent experimental findings. We also gathered novel training sets for *P. yoelii*, *B. bovis*, and *T. gondii*. ApiLoc [Woodcroft et al.] was used as the main resource for locating experimentally confirmed apicomplexan proteins.

We obtained experimentally-confirmed ApicoTP proteins from the ApiLoc database (version 3) and identified orthologs of these proteins from the OrthoMCL database (version 5) [Chen et al., 2006]. Proteins verified as having SPs by SignalP 3.0 were used in our positive training sets. Additional proteins were added to our training sets from references [Foth et al., 2003, Fleige et al., 2010, Kumar et al., 2010, Butzloff et al., 2010, Johnson et al., 2011, Caballero et al., 2011, Sheiner et al., 2011]. Because

of the scarcity of experimentally-confirmed *P. yoelii* and *B. bovis* ApicoTPs (only three proteins are confirmed to be ApicoTPs for each organism), we used homology transfer to establish reasonably sized training sets. CDART (Conserved Domain Architecture Retrieval Tool) [Geer et al., 2002] was employed to infer protein homology relationships by means of domain architecture similarity.

We obtained proteins tagged as non-Apicoplast from the ApiLoc database and found orthologs using the OrthoMCL database. The proteins predicted to have an SP region were used in our negative training sets. We also found proteins confirmed to localize to locations other than the apicoplast from the ApiLoc database. We manually eliminated proteins whose confirmed localization does not necessarily rule out apicoplast targeting. For example, we eliminated proteins confirmed to localize to mitochondria, food vacuoles, and the cytoplasm, as dual localization incidents have been reported in the literature involving apicoplasts and these locations. Because very few *P. yoelii* and *B. bovis* non-ApicoTPs have been experimentally confirmed, we added proteins annotated as variant erythrocyte surface antigen, merozoite surface antigen, and rhoptry related/associated to the negative training sets to increase their size.

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. Table 2.1 shows the breakdown of each training set by positive

(putative ApicoTPs) and negative (non-ApicoTPs) classes.

Dataset	**Number of putative ApicoTPs**	**Number of putative non-ApicoTPs**
*P. falciparum**	78	27
P. falciparum	47	41
B. bovis	28	29
T. gondii	35	33
P. yoelii	34	36

Table 2.1: Breakdown of the labeled datasets into positive (ApicoTP) and negative (non-ApicoTP) classes. *P. falciparum** refers to the published dataset used in the development of PlasmoAP. We used only the SP-containing portion of this set.

For ApicoAP, only proteins containing an SP were used for training. The published dataset of proteins for *P. falciparum* contains 102 non-ApicoTPs of which 75 lack SPs. As with ApicoAP, PlasmoAP requires a protein to contain an SP for prediction as an ApicoTP. Thus, exclusion of the 75 non-ApicoTPs will not affect comparison of the two methods. In fact, it is likely that a negative training set that includes proteins without SPs may well overstate the actual performance of a classifier given that the objective of such classifiers is to discriminate ApicoTPs from non-ApicoTPs when an SP is present.

2.4 Results

2.4.1 Evaluation of ApicoAP

ApicoAP was used with the five datasets described in the previous section. To estimate the expected prediction performance of ApicoAP, 35 cross validation was employed. A rule-based classifier is trained on a subset of a labeled dataset, which will be referred to as the training-validation set. As discussed earlier, this subset is further divided into training and validation sets, using 35 cross validation, to facilitate calculation of the objective function value during the parameter optimization phase. The parameters for our rule-based classifier are optimized in this phase, and the resulting classifier is applied to the remaining set (test set) to assess the performance of the model for unknown data. Fifteen test set samples were used to assess the model performance. The expected prediction performance of ApicoAP was calculated using Matthews Correlation Coefficient (MCC) by averaging the classifier MCCs over these samples.

During parameter optimization, often the parameter set found with the optimum objective value is not unique. Small perturbations of one or more parameters result in different parameter sets with the same optimum objective value. The trained classifiers with these parameter sets sometimes possess different expected prediction performances. In Table 2.2 we report the averages of minimum, maximum, and av-

erage accuracies observed together with the standard deviations. These reflect the worst-case, best-case, and the most-likely expected prediction performances, respectively.

Dataset	**Average accuracy (sd)**	**Minimum accuracy (sd)**	**Maximum accuracy (sd)**
*P. falciparum**	0.88 (0.08)	0.87 (0.09)	0.90 (0.07)
P. falciparum	0.87 (0.06)	0.84 (0.08)	0.91 (0.05)
B. bovis	0.82 (0.06)	0.76 (0.11)	0.87 (0.06)
T. gondii	0.83 (0.10)	0.8 (0.11)	0.86 (0.09)
P. yoelii	0.85 (0.07)	0.82 (0.09)	0.87 (0.06)

Table 2.2: Averaged expected prediction performance of ApicoAP (standard deviation (sd) in parentheses) for the labeled datasets.

The final classifier for each dataset uses a single parameter set. To form this parameter set we took the averages of the individual parameters obtained during the cross validation procedure. We then adjusted the threshold value taking into consideration the entire labeled dataset. Note that the performance measure used for threshold determination was also the MCC. The resulting classifiers for the four organisms were implemented in the ApicoAP software used for predicting putative ApicoTPs (discussed in detail in the next section). Table 2.3 lists the performance of ApicoAP for the different classifiers. In contrast to the values given in Table 2.2, the values in Table 2.3 do not estimate how well ApicoAP will perform for unknown

data but rather how well it performs for the available, labeled data.

Dataset	**True positive count (rate)**	**True negative count (rate)**	**Overall accuracy**
*P. falciparum**	73 (0.94)	26 (0.96)	0.94
P. falciparum	46 (0.98)	37 (0.9)	0.94
B. bovis	27 (0.96)	26 (0.9)	0.93
T. gondii	32 (0.91)	27 (0.82)	0.87
P. yoelii	32 (0.94)	33 (0.92)	0.93

Table 2.3: ApicoAP classifier performance on the labeled datasets.

A comparison between ApicoAP and PlasmoAP for the published *P. falciparum* dataset is given in Table 2.4. The values in Table 2.4 show that ApicoAP provides some improvement in both the true positive rate and the true negative rate, the latter implying fewer false positive predictions.

Classifier	**True positive count (rate)**	**True negative count (rate)**	**Overall accuracy**
ApicoAP	73 (0.94)	26 (0.96)	0.94
PlasmoAP	72 (0.92)	22 (0.81)	0.9

Table 2.4: Comparison of ApicoAP and PlasmoAP for *P. falciparum* dataset of 78 positives and 27 negatives.

2.4.2 *Comparison of ApicoAP model with various machine learning approaches*

A comparison between ApicoAP and various machine learning algorithms is given in Table 2.5 using 5-fold cross validation accuracy as the performance metric. Experiments are conducted with the help of the Weka tool [Hall et al., 2009]. Frequencies of the amino acids appeared in *PRS* and *ARS* are calculated for each protein sequence over the anticipated TP region by assuming a fixed length of 100. These frequencies are used to define the feature space. Several algorithms are used to train classifiers in this feature space. Table 2.5 lists the algorithms that provide the most promising results. When all the amino acids are used in feature extraction, instead of the ones listed in *PRS* and *ARS*, performances were poorer. As anticipated TP region length, values between 80 and 120 with increments of 10 are explored, and 100 is found to be the value that gave the best performance results. Several kernel choices and parameters are explored for the SVM classifier and the best results are achieved with the polynomial kernel with degree 1. The values in Table 2.5 show that ApicoAP performs significantly better than the other algorithms on the *B. bovis* dataset. Among the algorithms competed, ApicoAP is the top performer on the *P. falciparum* and *P. yoelii* datasets, and it performs slightly worse than the top performer, naïve Bayes algorithm, on the *T. gondii* dataset.

Dataset	Naïve Bayes	Logistic regression	SVM	ApicoAP
B. bovis	0.65	0.72	0.65	**0.82**
P. falciparum	0.80	0.76	0.84	**0.87**
P. yoelii	0.83	0.76	0.83	**0.85**
T. gondii	**0.84**	0.74	0.81	0.83

Table 2.5: Comparison of ApicoAP model with various machine learning algorithms: 5-fold cross-validated classification accuracy is used as the metric. Top performer for each dataset is shown in bold font.

2.4.3 ApicoAP predictions

After a given training set is used in the classification model, a rule-based classifier is obtained that predicts an ApicoTP when the following criteria are met:

- The protein sequence is predicted to contain an SP.
- The region of L_r amino acids following the SP cleavage site contains a pattern of L_p amino acids with a *p-criterion* value greater than or equal to the determined threshold.

The classifiers obtained using the training data available for *P. falciparum*, *P. yoelii*, *B. bovis*, and *T. gondii* are available in the ApicoAP software package. These classifiers were used to predict ApicoTPs as described in this section.

Many proteins expressed in the genomes of *P. falciparum*, *P. yoelii*, *B. bovis*, and *T. gondii* are predicted to contain SPs. The cardinality of these proteins for each organism, excluding the ones that are used for training and testing, is listed in Table 2.6. The number of proteins predicted to be ApicoTPs by ApicoAP is also listed in Table 2.6.

Organism	**SP-containing protein count***	**ApicoAP positive prediction count**
P. falciparum	1046	542
B. bovis	515	194
T. gondii	1037	417
P. yoelii	1049	285

Table 2.6: ApicoAP predictions for SP-containing *P. falciparum*, *B. bovis*, *T. gondii*, and *P. yoelii* proteins. *From all SP-containing protein sets, we excluded the training data.

Of the 1046 SP-containing *P. falciparum* proteins, 358 are predicted to be ApicoTPs by PlasmoAP. Of these 358, 261 (261/358 = 73%) are also predicted to be ApicoTPs by ApicoAP. The remaining SP-containing *P. falciparum* proteins (1046-358 = 688) are predicted to be non-ApicoTPs by PlasmoAP. Of these 688, 407 (407/688 = 60%) are also predicted to be non-ApicoTPs by ApicoAP. This leaves 281 (688-407 = 281) that are identified as additional putative ApicoTPs by ApicoAP.

Due to a lack of prediction tools in the literature for *B. bovis*, *P. yoelii*, and *T. gondii*, we were unable to compare our prediction results against a reference.

2.4.4 Optimized model parameters for ApicoAP classifiers

Figure 2.2 presents the frequency distributions for the preferred and avoided residues within the p regions of the training sequences for each organism. These regions are detected by applying the final ApicoAP classifiers to the sequences. In general, weight parameter estimates are found to be proportional to the differences between the frequency of residues for positive and negative sets. For *P. falciparum*, lysine (K) seems to have the greatest effect among the amino acids contributing to the *preferred residue set score* (*PRSS*). The greatest effect on the *PRSS* for the *P. yoelii* and *B. bovis* classifiers comes from Arginine (R) and for the *T. gondii* classifier it comes from Serine (S). All these estimates seem to be consistent with the given histograms.

The estimated region length parameter r was found to be 60, 62, 70, and 88 for *P. falciparum*, *P. yoelii*, *B. bovis*, and *T. gondii*, respectively. The estimated length of the p region was found to be 31, 36, 35, and 28 for *P. falciparum*, *P. yoelii*, *B. bovis*, and *T. gondii*, respectively.

Figure (2.3) shows how training data are mapped onto the *PRSS*-*ARSS* plane

when the final classifiers are applied. The discriminating line is shown, where the *PRSS*-intercept of this line corresponds to the estimated limiting value *lv*, given by Eq. (2.4), and the slope of the line corresponds to the estimated threshold value over the *p-criterion* value, given by Eq. (2.3). One interesting observation is that many of the *T. gondii* proteins contain p regions with no acidic residues, i.e. the *ARSS* is zero. Misclassifications of negative training data appear to be associated with this type of *p* region.

In addition to the content of the *p* regions presented in Figure 2.2 we analyzed the locations of these regions among our positive training data (with cardinality of 144). In about 55% of the sequences, the *p* region identified (with max *p-criterion* value) appears immediately after or within 5 residues of the predicted SP cleavage site. For the remaining sequences, the *p* region appears (on average) 20 residues away from the SP cleavage site. We analyzed the region between the predicted SP cleavage site and the start of the *p* region, which we refer to as the *pre-pattern* region. In order to account for SP cleavage site prediction errors, we assume a *pre-pattern* region exists when the *p* region appears 5 or more residues away from the predicted SP cleavage site. Our goal was to compare the acidic residue (D and E) frequencies of these two regions. Hypothesis testing was applied to confirm that the mean of the difference differs from zero. For this test and for all the interval estimates following, we used a p-value of 0.05. The acidic residue frequency in the *pre-pattern* region was

observed to be higher than in the p region by 8% to 11% in 78% of these proteins. The highest and lowest differences observed were 33% and 1%, respectively.

We repeated the same analysis on a subset of our positive training data containing only the experimentally confirmed ApicoTPs (with cardinality of 70). In 43% of these, a *pre-pattern* region existed. The acidic residue frequency in the *pre-pattern* region was observed to be higher than in the p region by 6% to 11% in 90% of these proteins. Similar tendencies were also observed among the ApicoTPs predicted by ApicoAP.

Experimental findings for *T. gondii* transit peptides (TP) indicate that the absence of acidic residues in the N-terminal portion of the TP is important for TP fidelity, even more important than the presence of positive charge [Tonkin et al., 2006]. Tonkin et al. used the acyl carrier protein (ACP) from *T. gondii* in these experiments. ApicoAP identifies no *pre-pattern* region in this particular protein, which means that the p region is located immediately after the predicted SP cleavage site. This indicates that the prediction mechanism of ApicoAP, based entirely on the p region, which does not necessarily appear on the N-terminal portion of a TP, does not contradict the experimental findings.

2.5 Discussion

The apicoplast is a unique organelle that resides in a group of eukaryotic parasites, known as Apicomplexa, which are responsible for a wide range of serious diseases among humans and livestock. As resistance to commonly used drugs increases in apicomplexan parasites, it is important to find new drug targets. The apicoplast is an essential organelle for the survival of these parasites and, with its prokaryotic origin, is viewed as a promising drug target. The majority of apicoplast proteins are nuclear-encoded and targeted post-translationally to the apicoplast organelle. Experimental identification of apicoplast-targeted proteins (ApicoTPs) is a costly and time-consuming task. Accurate *in silico* prediction methods are needed to accelerate the identification of promising drug targets.

The computational approach available for genome-wide ApicoTP prediction, known as PlasmoAP [Foth et al., 2003], was developed to identify ApicoTPs in *P. falciparum* and, as such, application to other Apicomplexa is considered to be unreliable. We have developed an alternative computational model ApicoAP. In ApicoAP, we conduct a systematic search over a rule space using the expected prediction performance of a rule on a training set as the optimization criterion. The rule space is formalized by our parametric rule definition, and optimization is performed using a genetic algorithm. A major advantage of our approach to the genome-wide Api-

coTP prediction task is that it is not restricted to a single organism but rather is customizable to different organisms for which training data are available.

Performance of ApicoAP is evaluated for labeled datasets of *P. falciparum*, *P. yoelii*, *B. bovis*, and *T. gondii* proteins, one of which is the dataset published in conjunction with PlasmoAP [Foth et al., 2003]. The evaluation utilizes cross validation, a common approach used to validate classification models. The cross-validation procedure provides an estimate of the prediction performance of a model by systematically retaining a portion of a labeled dataset and using this portion to test the model obtained using the remainder of the dataset. The expected prediction accuracies, i.e., the accuracy for unknown proteins rather than the accuracy for labeled data, for the current ApicoAP classifiers for *P. falciparum*, *P. yoelii*, *B. bovis*, and *T. gondii* are found to be 87%, 85%, 82%, and 83%, respectively. The best expected prediction accuracy is achieved using the *P. falciparum* training set, the largest of the four training sets. The larger the training data set, the more robust and accurate the resulting classifier is expected to be. With the addition of more training data, the classifiers can be updated to provide greater accuracy. While the four classifiers are specifically for use with the four species described, they may assist in the identification of potential ApicoTPs for related species when the AT-codon biases of the corresponding genomes are similar.

In this study we present ApicoAP, the first computational model capable of

identifying ApicoTPs in multiple species of Apicomplexa. In addition, we provide a user-friendly, Python-based program that includes the ApicoAP classifiers for *P. falciparum*, *P. yoelii*, *B. bovis*, and *T. gondii*. ApicoAP provides a learning framework for ApicoTP prediction based on a systematic approach to finding the rule-based classifier with the best expected prediction performance over a training set. This framework can be applied to other domains for which it is desirable to have a discriminating rule-finding process that is automated.

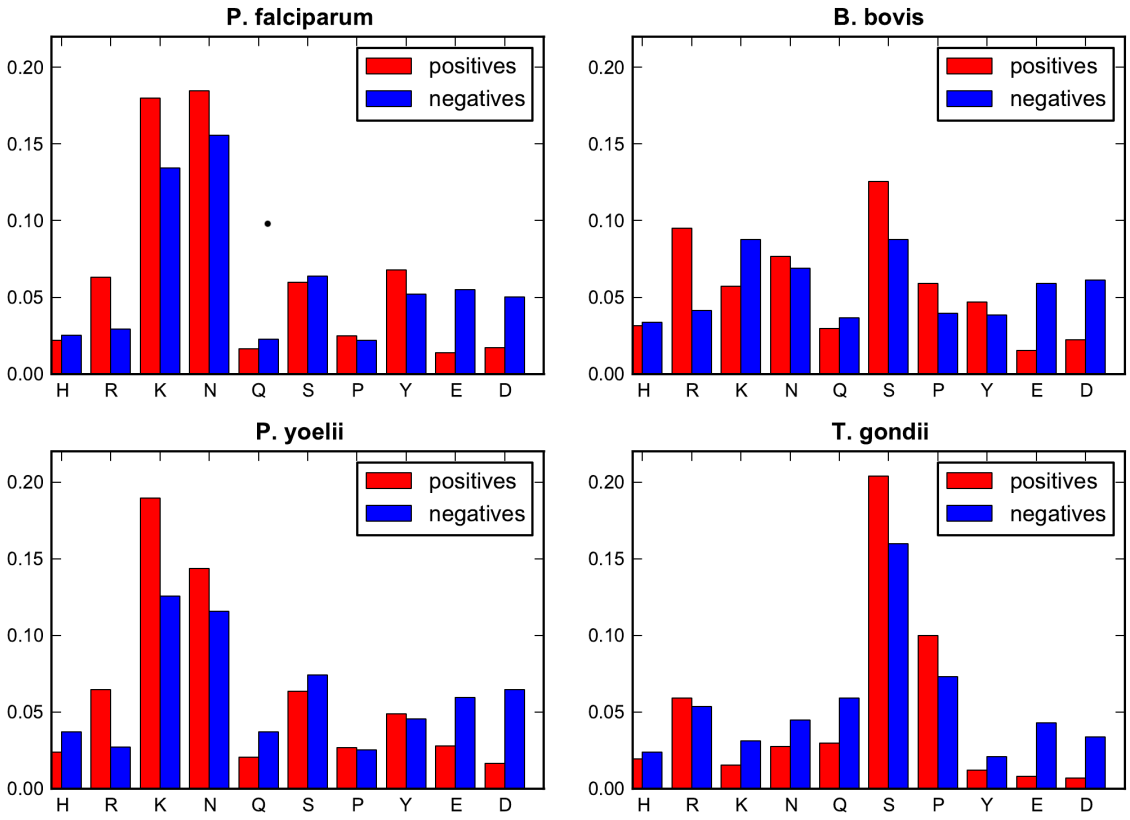

Figure 2.2: Averaged frequency distributions of preferred and avoided residues for the p regions of the training sequences: This figure presents the frequency distributions of preferred and avoided residues for the p regions of the training sequences for each organism. p is the contiguous sub-region with length Lp in the anticipated TP region r that has the maximum p-criterion value, given by Eq. (2.3). Final ApicoAP classifiers are used to identify p regions over each sequence. Residue counts over individual p regions are divided by the lengths of the p regions, and the resulting values are averaged over positive and negative training sets for each organism.

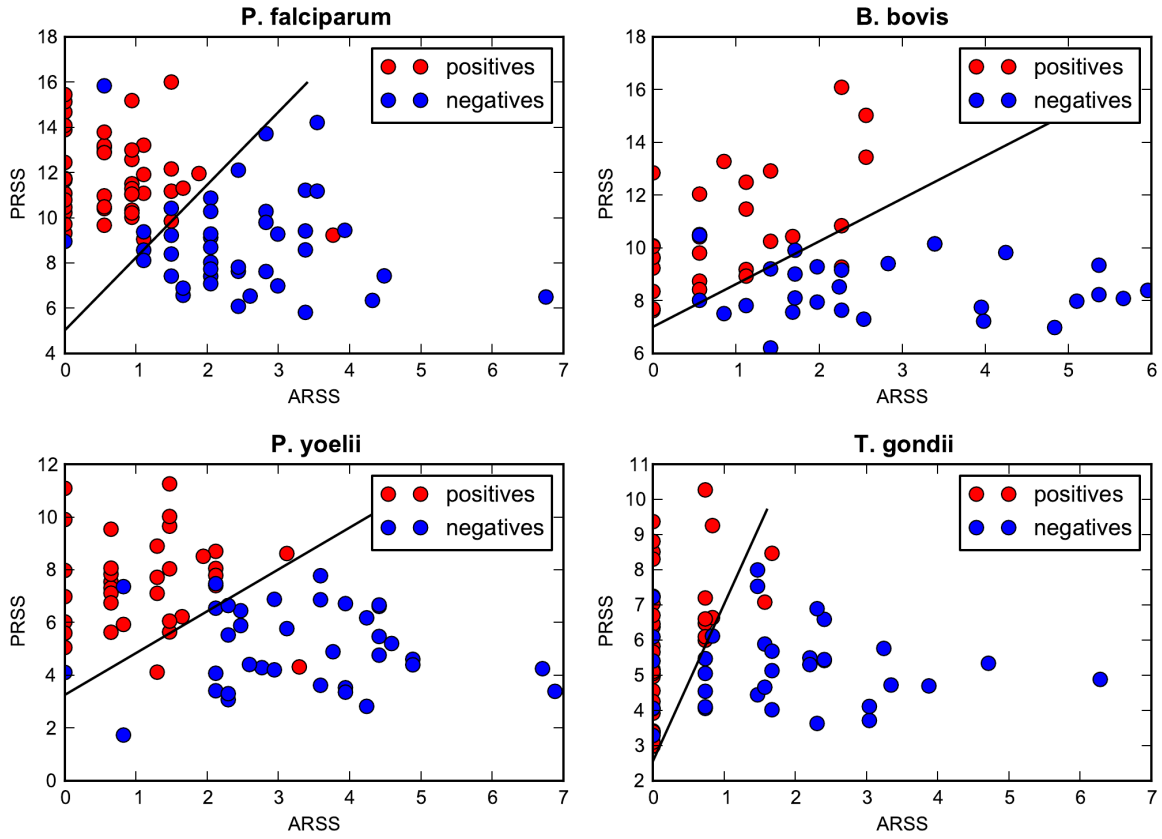

Figure 2.3: Training data mapped onto the PRSS-ARSS plane using final ApicoAP classifiers: This figure shows how training data are mapped onto the *PRSS-ARSS* plane when the final ApicoAP classifiers are applied. The *preferred residue set score* (*PRSS*) and *avoided residue set score* (*ARSS*) quantify the existence of *preferred residue set* (*PRS*) and *avoided residue set* (*ARS*) elements in the p regions of the training sequences for each organism. See Eqs. (2.1) and (2.2) for definitions. The discriminating lines are shown on each plot, where the *PRSS*-intercept of each line corresponds to the estimated limiting value *lv*, given by Eq. (2.4), and the slope of each line corresponds to the estimated threshold value over the *p-criterion* values, given by Eq. (2.3).

CHAPTER 3. APICOAMP

3.1 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) [Foth et al., 2003, Cilingir et al., 2012]. 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 [Karnataki et al., 2007b, DeRocher et al., 2008, 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 [Karnataki et al., 2007b, DeRocher et al., 2008, Karnataki et al., 2007a]. 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 [Krogh et al., 2001, HOFMANN, 1993, Von Heijne et al., 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 do-

main 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. 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., 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, Du et al. 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 [Li et al., 2012, Chi and Nam, 2012, Mei et al., 2011, Blum et al., 2009, Huang et al., 2008]. 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. used hydrophobicity and com-

position 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 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. 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 [Pierleoni et al., 2011, Sharpe et al., 2010, Du, 2012, Du et al., 2012] 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.

3.2 Methods

3.2.1 The dataset

We obtained experimentally-confirmed apicoplast-targeted proteins from the ApiLoc database (version 3) [Woodcroft et al.] and from recent references [Sheiner et al., 2011, Fleige et al., 2010]. 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. 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.

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 3.1 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.

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

Table 3.1: Breakdown of the labeled datasets into positive (ApicoTMP) and negative (non- ApicoTMP) classes for 11 species of Apicomplexa.

3.2.2 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.

3.2.3 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, 1999, Vapnick, 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 units 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.

3.2.4 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.

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. and Pierleoni et al., 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 3.1.

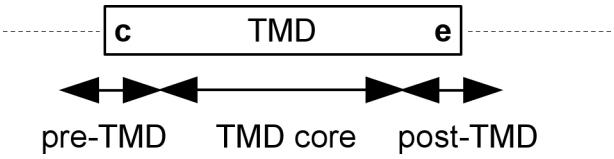

Figure 3.1: Three subregions of a transmembrane domain (TMD): A TMD region is composed of 3 sub-regions: a hydrophobic core and pre- and post-TMD sub-regions that are aligned with the inner and outer leaflets of the membrane. When TMDs are aligned from the cytoplasmic side to the exoplasmic side, rather 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.

Hydrophobic cores of TMDs were identified following a procedure similar to the one proposed by Sharpe et al.. 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.

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 *T. 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 F_P and 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 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 $width_{min}$ and $width_{max}$ 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 F_P, 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, $width_{min}$, $width_{max}$, and p were set to be 10, 3, 5 and 0.1, respectively.

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, Du et al. 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 in [Du, 2012] and [Du et al., 2012]. Differing from Du et al., 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 method 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. 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.

3.2.5 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.

3.3 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, 2004].

Table 3.2 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 3.3 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.

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 3.4 presents the prediction statistics for each apicomplexan species using 10 as the value of the *vote threshold*.

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 reason-

ably 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 [Pierleoni et al., 2011, Sharpe et al., 2010, Du, 2012, Du et al., 2012] 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. 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.

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. 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 work 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.

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

Table 3.2: Average expected accuracy of various classification models for the ApicoTMP prediction problem with different values of the *vote threshold*(vt) parameter. The table is sorted from best to worst performance. True-positive and false-positive rates are in parentheses.

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

Table 3.3: 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.

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

Table 3.4: ApicoAMP predictions for 16 apicomplexan species. The value of the *vote threshold* parameter is set to 10 for this analysis.

CHAPTER 4. APICOAP PIPELINE

4.1 Introduction

Public gene and protein databases such as GenBank [Benson et al., 1997], UniProt [Bairoch et al., 2005], and EuPathDB [Aurrecoechea et al., 2010] are major resources for gathering data to train supervised machine learning applications used by life scientists for a variety of objectives including the detection of targeting sequences and the prediction of transmembrane domain topology. While these data resources are quite dynamic in nature, that is to say they are continuously updated by the addition of new information, machine learning applications are often static and cannot incorporate the new information. When a supervised machine learning approach is introduced, by necessity the learning method is developed using the data available in public resources at the current time to train classifiers or predictors and is then provided for public use. It is not uncommon to find applications in prevalent use that are trained using data sets that are outdated not long after their introduction. Given that the vast majority of the procedures described for gathering training data can easily be automated, requiring very little, if any, human assistance, it makes sense

to transform valuable machine learning applications into self-evolving learners that adapt to the ever-changing data on genes and proteins and to develop new machine learning applications that are similarly capable.

In this study, we propose a new model of operation for specific supervised machine learning applications that is aligned with the needs of the fast changing nature of genomic data. We believe that applications that learn from genomic data should be defined in a pipeline in which the data gathering procedure for training data is automated and the learning process is as well. Such a pipeline would function as a classifier or predictor generator that does not require training data to be provided, but instead is capable of generating a model from the information available from public resources at a given time.

Because every learning problem has its own training data requirements and dataset curation procedures, the proposed model of operation is best explained using a case study. We have selected the apicoplast-targeted protein prediction problem for our case study and utilize an existing machine learning model, ApicoAP [Cilingir et al., 2012], in a pipeline that is comprised of an automated training data gathering procedure and the classifier training routine defined as part of the ApicoAP model.

4.2 ApicoAP Model

ApicoAP [Cilingir et al., 2012] is a generalized rule-based classification model that identifies apicoplast-targeted proteins that use a bipartite signaling mechanism (ApicoTPs). This model uses a training set containing known ApicoTPs and non-ApicoTPs for an apicomplexan species as input and outputs a classifier that identifies ApicoTPs from the proteins of this species. The ApicoAP prediction software currently offers service for 4 apicomplexan species. No other prediction method offers such a service for the remaining 13 apicomplexan species whose genomes have now been sequenced. ApicoAP is a generic model customizable to any apicomplexan species that has training data. If a training data gathering procedure can be systematically defined and automated, one can utilize the ApicoAP model as part of a pipeline to employ proteome information for an apicomplexan species to create a classifier for identifying ApicoTPs from the proteins of this species. As the results from experimental confirmation of ApicoTPs are published, the main resource for obtaining training data, this pipeline will not only be useful for an apicomplexan species for which no ApicoAP predictor exists, but it will also provide ever-improving classifiers for apicomplexan species for which an ApicoAP predictor already exists.

In the remainder of this chapter, we will define a generic pipeline for the purpose of ApicoTP prediction (hereafter referred to as the ApicoAP Pipeline) that consists of

an automated training data gathering procedure and the ApicoTP classifier training routine. In addition, we will discuss implementation of this pipeline, ApicoAP-CS for *Apico*AP *C*omplete *S*uite, which is a collection of web services. ApicoAP-CS can be utilized to generate a species-specific ApicoAP classifier that can be easily integrated into the ApicoAP prediction software. ApicoAP-CS utilizes public databases such as ApiLoc [Woodcroft et al.], EuPathDB [Aurrecoechea et al., 2010] and OrthoMCL [Chen et al., 2006], and public bioinformatics tools such as SignalP [Bendtsen et al., 2004], and BLAST [Camacho et al., 2009]. ApicoAP-CS client software is available at `http://bcb.eecs.wsu.edu`.

4.3 The ApicoAP Pipeline

Figure 1 shows the flowchart for the ApicoAP pipeline. The training data gathering procedure begins with the curation of a set of proteins whose subcellular localization has been experimentally confirmed. This set constitutes the seed training set and is used to identify the orthologs of the member proteins from the proteins of the apicomplexan species of interest. Known ApicoTPs and non-ApicoTPs for this apicomplexan species together with the orthologs of the seed training set make up the interim training set. A filtering step follows that eliminates proteins with no predicted signal peptide as the presence of a signal peptide is a requirement for ApicoTPs. The

resulting set is then screened for redundant sequences, which are removed, and the remaining elements form the final training set. This set is then fed into the ApicoTP classifier training routine which produces a species-specific ApicoTP classifier. Each of these steps is discussed in greater detail below.

4.3.1 Construction of seed training sets

Seed training sets contain proteins whose subcellular localization has been experimentally confirmed. Because the ApicoAP model requires two training sets, one containing ApicoTPs (positive set) and the other non-ApicoTPs (negative set), two disjoint seed sets are needed. These sets may contain proteins from multiple apicomplexan species. The positive seed set consists of proteins that are known to localize to the apicoplast. The negative seed set consists of proteins that are known to localize to organelles in the cell other than the apicoplast. Proteins whose confirmed localization does not necessarily rule out apicoplast targeting are not included in this set. For example, proteins confirmed to localize to mitochondria, food vacuoles, the endoplasmic reticulum, and the cytoplasm are eliminated because dual localization incidents have been reported in the literature involving apicoplasts and these other locations.

ApiLoc [Woodcroft et al.] is an expert-curated database which currently serves

as the main resource for curating apicomplexan proteins whose subcellular localization has been experimentally confirmed. Because at present this database is not updated often, it may be beneficial to perform a literature search to insure that all possible information is present in the seed training sets.

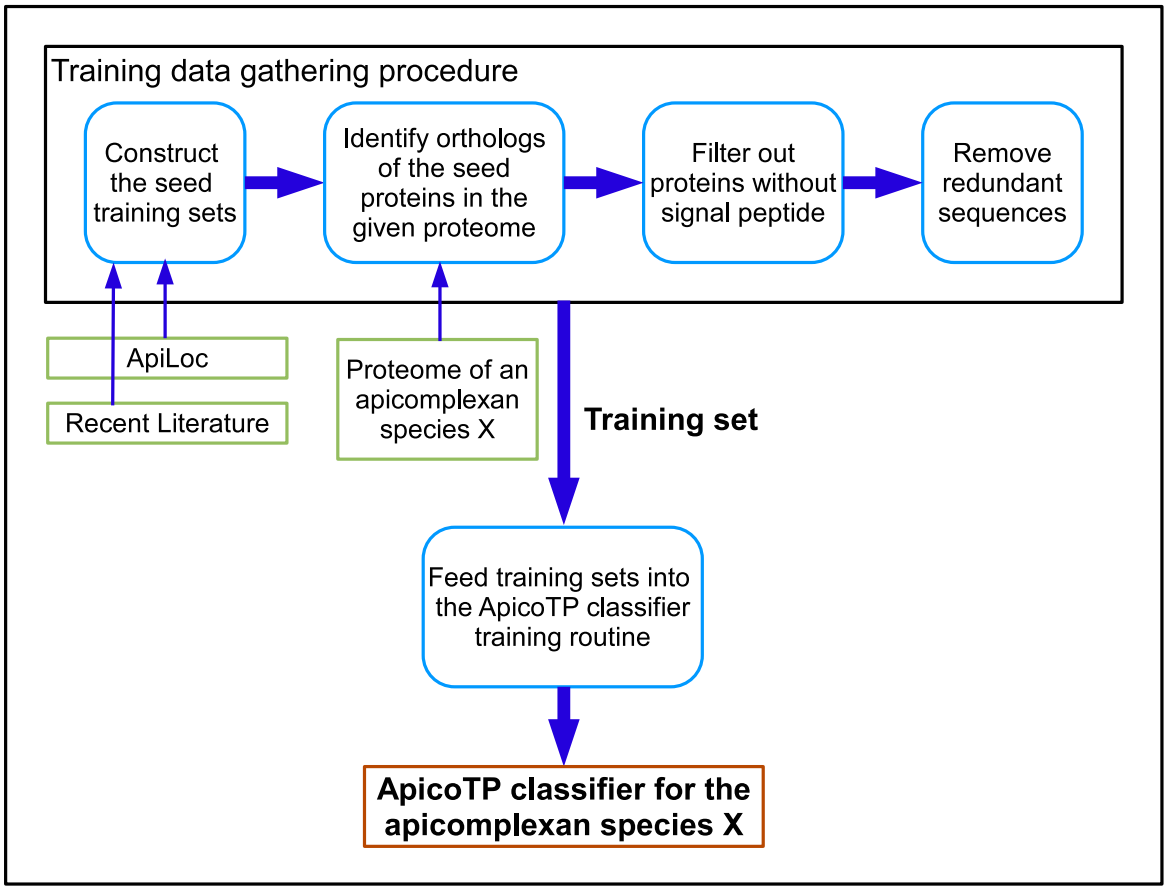

Figure 4.1: ApicoAP Pipeline.

ApicoAP-CS is capable of automatically parsing different ApiLoc versions to extract the information necessary to prepare seed training sets. ApicoAP-CS also

provides functionality for the user to submit additional proteins obtained from recent experimental studies.

4.3.2 Search for orthologs

Orthologs are defined as genes or gene products in different species which derive from a common ancestor [Fitch, 1970]. Orthologs are expected to retain the same function in different species, thus having a strong likelihood of localizing to the same organelle in a cell. Therefore, utilizing orthology search strategies in training data gathering procedures, especially for subcellular localization prediction tasks, is a common practice.

Several approaches have been developed to predict putative orthologous proteins on the basis of various information sources including phylogenic relationships, protein-protein interaction networks, and sequence similarity relationships. While a simple BLAST (Basic Local Alignment Search Tool) search with a stringent e-value cut-off may identify a sequentially conserved subset of orthologs of a protein in a database, other tools attempt to recognize orthology relationships in the event of low sequence conservation. Among other orthologous protein prediction tools, OrthoMCL [Chen et al., 2006] has a special focus on eukaryotic genomes, and it also has the most up-to-date support for apicomplexan species. EuPathDB [Aurrecoechea et al., 2010]

provides a user-friendly interface for orthology searches with OrthoMCL.

ApicoAP-CS utilizes the web service interface provided by EuPathDB to automatically identify orthologs of the seed training set members in an apicomplexan species. ApicoAP-CS also provides an alternative for newly sequenced genomes that may not yet have OrthoMCL support. It uses a BLAST-based algorithm with a stringent e-value cutoff of 1e-10, and the best hits for each protein in the seed training sets are retained as orthologs when present. ApicoAP-CS provides the option of using both methods and retaining the union set.

The orthology search results in two interim training sets, the positive and negative training sets. At the conclusion of this step, proteins appearing in both sets are assumed to be caused by annotation errors and are discarded.

4.3.3 Filtering out proteins with no signal peptide

ApicoTPs must contain a signal peptide. To insure this requirement is met, we apply a filtering stage in which proteins not predicted to contain a signal peptide are discarded from both interim training sets. SignalP 3.0 is used to identify proteins with putative signal peptides because it is the tool commonly reported in the literature for apicomplexan genomes. ApicoAP-CS utilizes the web service interface of SignalP to automate this filtering step in the ApicoAP Pipeline. Neto et al. hypothesized

that divergence in signal peptide predictions within orthologous groups is mainly due to N-terminal protein sequence misannotation and demonstrated that this is indeed the case. In addition to providing new gene models for certain proteins of *Plasmodium* spp, they suggested the use of thresholds differing from the default values for interpretation of SignalP [Bendtsen et al., 2004] results. We used their suggested threshold combination in ApicoAP-CS (D-Score=0.48; HMM probability=0.90).

4.3.4 Redundant sequence removal

It is known that a training set containing redundant members biases the learning process [Mitchell, 1997, Hobohm et al., 2008], especially when the learning involves an optimization procedure, as is the case for ApicoAP for which the optimization criterion is the expected performance of the candidate classifiers. The expected prediction performance of a classifier quantifies how well it is expected to generalize to new data instances. This performance metric can be estimated using statistical methods that involve retaining subsets of the training set from the training procedure and using these subsets to measure the prediction performance. These estimation strategies assume that the members of the training set are independently drawn from the main population. This assumption would be violated with the existence of redundant members causing either an overestimate or an underestimate of the estimated

performance.

ApicoAP-CS utilizes the CD-HIT method (version 4.6.1) [Li and Godzik, 2006] to eliminate redundant sequences that share more than 60% sequence similarity. The resulting training sets are subjected to user approval because expert knowledge may be required to identify unlikely ApicoTPs and non-ApicoTPs in the training sets. If the final training sets have cardinality of about or exceeding 20, the user is recommended to proceed with the ApicoAP training; for lower cardinality continuation to the next step is not recommended as the resulting classifier is not likely to have a high expected accuracy. As the cardinality and precision of the training sets increase, the expected accuracy of the resulting classifier improves.

4.3.5 Application of the ApicoTP classifier training routine

The ApicoAP model describes a parametric model of ApicoTPs and utilizes an optimization/training procedure in which the parameters that will lead to the classifier with the best expected accuracy are identified. For a detailed description of ApicoAP, the reader is referred to reference [Cilingir et al., 2012]. ApicoAP-CS utilizes this procedure with the training sets obtained in prior steps to generate an ApicoTP classifier that is specialized to the apicomplexan species of interest to the user. This classifier can be easily integrated into the ApicoAP prediction software. A

supplementary users manual includes detailed instructions on the integration procedure for the ApicoAP prediction software (version 3). The training procedure takes considerably more time relative to the other steps in the ApicoAP Pipeline due to the computationally intensive optimization procedure required by the ApicoAP model.

4.4 ApicoAP-CS Results

We applied ApicoAP-CS to the 13 apicomplexan species whose genomes are available in EuPathDB (version 2.17), namely *Babesia bovis*, *Babesia microti*,*Cryptosporidium hominis*,*Cryptosporidium muris*,*Cryptosporidium parvum*, *Eimeria tenella*, *Neospora caninum*, *Plasmodium berghei*, *Plasmodium chabaudi*, *Plasmodium cynomolgi*, *Plasmodium falciparum*, *Plasmodium knowlesi*, *Plasmodium vivax*, *Plasmodium yoelii*, *Theileria annulata*, *Theileria parva* and *Toxoplasma gondii*. The positive seed training set contains 75 known ApicoTPs extracted from ApiLoc (version 3) and 18 confirmed proteins curated from the recent literature. The current negative seed training set contains 400 known non-ApicoTPs extracted from ApiLoc.

We used both the OrthoMCL and BLAST-based algorithm (using Blast+ version 2.2.27) for the orthology search. Tables 4.1 and 4.2 show the cardinalities of positive and negative interim training sets that were automatically gathered by ApicoAP-CS. After the SignalP filtering step was applied, the resulting interim training sets

(shown as the last column in Tables 4.1 and 4.2) were subjected to the redundancy removal procedure, which produced the final training sets whose cardinalities are shown in Table 4.1.

For 10 of the 13 apicomplexan species, we were able to gather sufficiently large training sets to train specialized ApicoTP classifiers. These classifiers are included in the ApicoAP prediction software (version 3). Table 4.4) gives the prediction accuracies for the resulting ApicoTP classifiers obtained using each training set. These are not expected accuracy results, which estimate how well the resulting classifiers will perform with unknown data, but rather they indicate how well these classifiers perform with the available, labeled data.

All protein sequences were obtained from EuPathDB (version 2.17) except the ones whose gene models were proposed to be changed by [Neto et al., 2012].

4.5 Discussion

Supervised machine learning algorithms are used by life scientists for a variety of objectives including the detection of targeting sequences and the prediction of transmembrane domain topology. Expert-curated public gene and protein databases are major resources for gathering data to train these algorithms. While these data resources are continuously updated with the addition of new information, generally

this information is not incorporated into published machine learning algorithms which thereby can become outdated soon after their introduction.

In this study, we propose a new model of operation for supervised machine learning algorithms that learn from genomic data. By defining these algorithms in a pipeline in which the training data gathering procedure and the learning process are automated, one can create a system that generates a classifier or predictor using information available from public resources. Because data requirements and data set curation procedures vary, the proposed model of operation is explained using a case study in which an existing machine learning model, ApicoAP, is utilized in a pipeline. The ApicoAP Pipeline is capable of generating classifiers for different apicomplexan species without provision of training data.

Given that the vast majority of the procedures described for gathering training data can easily be automated, it is possible to transform valuable machine learning algorithms into self-evolving learners that benefit from the ever-changing data available for genes and proteins and to develop new machine learning algorithms that are similarly capable. This generic idea is applied to the apicoplast-targeted protein prediction problem to create the ApicoAP Pipeline. An implementation of this pipeline as a collection of web services is available. The client software can be found at http://bcb.eecs.wsu.edu.

Apicomplexan species	**Ortho-MCL**	**Blast**	**Confir-med**	**All combined**	**Conflicts removed**	**Non-SP filtered**
B. bovis	46	45	4	61	59	18
B. microti	51	50	0	61	58	23
C. hominis	17	24	0	28	25	1
C. muris	19	27	0	32	29	0
C. parvum	17	24	0	29	26	2
E. tenella	82	61	1	89	84	30
N. caninum	78	68	0	82	77	21
P. berghei	72	73	0	77	73	49
P. chabaudi	72	73	0	77	72	51
P. cynomolgi	70	72	0	77	73	31
P. falciparum	45	60	40	89	85	52
P. knowlesi	72	72	0	77	73	49
P. vivax	69	72	0	75	71	51
P. yoelii	70	68	3	77	73	41
T. annulata	45	47	0	56	54	23
T. parva	49	42	0	59	57	25
T. gondii	53	59	45	102	96	42

Table 4.1: Cardinalities of the positive interim training sets for the 13 apicomplexan species gathered by ApicoAP-CS: Interim set cardinality after orthology search with OrthoMCL and with Blast-based algorithm follow the cardinality of the set containing experimentally confirmed positive proteins. Column named "'All combined"' shows set cardinalities when the first three sets are merged. Before filtering out the proteins that are predicted to contain no SP (non-SP) from the resulting set, conflicts between the negative and the positive interim training sets are identified. The number of conflicts that are eliminated are shown at the column next to the last.

Apicomplexan species	**Ortho-MCL**	**Blast**	**Confir-med**	**All combined**	**Conflicts removed**	**Non-SP filtered**
B. bovis	144	136	8	161	159	33
B. microti	142	130	0	159	156	23
C. hominis	135	130	0	157	154	28
C. muris	143	137	0	163	160	34
C. parvum	130	129	10	164	161	33
E. tenella	400	175	8	443	438	169
N. caninum	254	220	15	288	283	81
P. berghei	222	212	28	260	256	101
P. chabaudi	238	223	2	258	253	108
P. cynomolgi	259	224	0	273	269	93
P. falciparum	284	173	156	443	439	138
P. knowlesi	236	227	6	258	254	91
P. vivax	261	227	13	281	277	103
P. yoelii	242	216	16	270	266	89
T. annulata	151	133	4	169	167	42
T. parva	186	128	4	204	202	71
T. gondii	194	198	131	333	327	92

Table 4.2: Cardinalities of the negative interim training sets for the 13 apicomplexan species gathered by ApicoAP-CS: Interim set cardinality after orthology search with OrthoMCL and with Blast-based algorithm follow the cardinality of the set containing experimentally confirmed negative proteins. Column named "'All combined"' shows set cardinalities when the first three sets are merged. Before filtering out the proteins that are predicted to contain no SP (non-SP) from the resulting set, conflicts between the negative and the positive interim training sets are identified. The number of conflicts that are eliminated are shown at the column next to the last.

Apicomplexan species	**Positive training set**	**Negative training set**
B. bovis	18	30
B. microti	23	22
C. hominis	1	28
C. muris	0	34
C. parvum	2	33
E. tenella	30	143
N. caninum	21	77
P. berghei	49	94
P. chabaudi	51	98
P. cynomolgi	31	90
P. falciparum	51	132
P. knowlesi	48	90
P. vivax	51	101
P. yoelii	41	87
T. annulata	23	41
T. parva	25	61
T. gondii	42	86

Table 4.3: Cardinalities of the final training sets for the 13 apicomplexan species.

Apicomplexan species	**True negative rate**	**True positive rate**	**Overall accuracy**
B. bovis	1.000	1.000	1.000
B. microti	0.909	1.000	0.956
E. tenella	0.951	0.800	0.925
N. caninum	1.000	0.857	0.969
P. berghei	0.936	0.959	0.944
P. chabaudi	0.959	0.902	0.940
P. cynomolgi	1.000	0.839	0.959
P. falciparum	0.924	0.843	0.902
P. knowlesi	0.922	0.958	0.935
P. vivax	0.901	0.980	0.928
P. yoelii	0.954	0.854	0.922
T. annulata	0.854	0.913	0.875
T. parva	0.820	0.960	0.860
T. gondii	0.977	0.905	0.953

Table 4.4: ApicoTP classifier performances on the training sets gathered by ApicoAP-CS.

CHAPTER 5. CONCLUSION

As resistance to commonly used drugs is increasing among apicomplexan parasites, it is important to find new drug targets. The apicoplast is an ideal drug target both because of its unique properties and because it is essential for the survival of the parasite. Understanding the metabolic activities performed in the apicoplast is necessary for drug target identification, and this requires the ability to identify apicoplast-targeted proteins. Because experimental identification of these proteins is a costly and time-consuming task, accurate *in silico* prediction methods are needed to accelerate the drug target identification process.

In this dissertation, we present two computational approaches, ApicoAP and ApicoAMP, that identifies two different types apicoplast-targeted proteins. ApicoAP is the first computational model capable of identifying ApicoTPs with bipartite signals in multiple species of Apicomplexa. ApicoAMP is the first computational model that identifies apicoplast-targeted transmembrane proteins.

In addition, we propose a new model of operation for specific supervised machine learning algorithms that learn from datasets curated from dynamically changing public resources, such as genomic databases. By employing these algorithms as part of a pipeline in which the training data gathering procedure as well as the learning

process is automated, one can have a system that functions as a classifier generator that does not require training data to be provided, but instead has the capability to utilize the information available in public resources at a given time for training. The proposed model of operation is explained using a case study where ApicoAP is utilized in such a pipeline. ApicoAP Pipeline is capable of generating classifiers for different apicomplexan species, without requiring training data to be provided. As the results from experimental confirmation of ApicoTPs are published, which is the main resource for obtaining training data, this pipeline will not only be useful for an apicomplexan species for which no ApicoAP classifier exists, but it will also provide ever-improving classifiers for apicomplexan species for which an ApicoAP classifier already exists. ApicoAP Pipeline is used to train classifiers for 10 more apicomplexan species in addition to the 4 existing ones. ApicoAP, ApicoAMP and ApicoAP Pipeline significantly broaden the set of apicoplast-targeted proteins that can be identified computationally.

Bibliography

Ethem Alpaydin. *Introduction to machine learning*. MIT press, second edition, 2004.

Cristina Aurrecoechea, John Brestelli, Brian P Brunk, Steve Fischer, Bindu Gajria, Xin Gao, Alan Gingle, Greg Grant, Omar S Harb, Mark Heiges, et al. Eupathdb: a portal to eukaryotic pathogen databases. *Nucleic acids research*, 38(suppl 1): D415–D419, 2010.

Timothy Bailey, Mikael Bodén, Tom Whitington, and Philip Machanick. The value of position-specific priors in motif discovery using meme. *BMC bioinformatics*, 11 (1):179, 2010.

Timothy L Bailey, Nadya Williams, Chris Misleh, and Wilfred W Li. Meme: discovering and analyzing dna and protein sequence motifs. *Nucleic acids research*, 34 (suppl 2):W369–W373, 2006.

Amos Bairoch, Rolf Apweiler, Cathy H Wu, Winona C Barker, Brigitte Boeckmann, Serenella Ferro, Elisabeth Gasteiger, Hongzhan Huang, Rodrigo Lopez, Michele Magrane, et al. The universal protein resource (uniprot). *Nucleic acids research*, 33(suppl 1):D154–D159, 2005.

Pierre Baldi, Søren Brunak, Yves Chauvin, Claus AF Andersen, and Henrik Nielsen.

Assessing the accuracy of prediction algorithms for classification: an overview. *Bioinformatics*, 16(5):412–424, 2000.

Asa Ben-Hur and Jason Weston. A users guide to support vector machines. *Methods in Molecular Biology*, 609:223–239, 2010.

J Dyrlov Bendtsen, Henrik Nielsen, Gunnar von Heijne, Søren Brunak, et al. Improved prediction of signal peptides: Signalp 3.0. *J Mol Biol*, 340(4):783–795, 2004.

Dennis A Benson, Mark S Boguski, David J Lipman, and James Ostell. Genbank. *Nucleic acids research*, 25(1):1–6, 1997.

Torsten Blum, Sebastian Briesemeister, and Oliver Kohlbacher. Multiloc2: integrating phylogeny and gene ontology terms improves subcellular protein localization prediction. *BMC bioinformatics*, 10(1):274, 2009.

Kelly A Brayton, Audrey OT Lau, David R Herndon, Linda Hannick, Lowell S Kappmeyer, Shawn J Berens, Shelby L Bidwell, Wendy C Brown, Jonathan Crabtree, Doug Fadrosh, et al. Genome sequence of Babesia bovis and comparative analysis of apicomplexan hemoprotozoa. *PLoS pathogens*, 3(10):e148, 2007.

Sabine Butzloff, Sandra M Cardoso, Rolf D Walter, Carsten Wrenger, and Ingrid B Muller. The iron-sulfur (fe-s) cluster assembly in Plasmodium falciparum. *Molecular Parasitology*, Abstract 256B, 2010.

Marina C Caballero, Monica J Pedroni, Guy H Palmer, Carlos E Suarez, Christine Davitt, and Audrey OT Lau. Characterization of acyl carrier protein and lytb in Babesia bovis apicoplast. *Molecular and biochemical parasitology*, 2011.

Christiam Camacho, George Coulouris, Vahram Avagyan, Ning Ma, Jason Papadopoulos, Kevin Bealer, and Thomas L Madden. Blast+: architecture and applications. *BMC bioinformatics*, 10(1):421, 2009.

Feng Chen, Aaron J Mackey, Christian J Stoeckert Jr, and David S Roos. Orthomcl-db: querying a comprehensive multi-species collection of ortholog groups. *Nucleic acids research*, 34(suppl 1):D363–D368, 2006.

Sang-Mun Chi and Dougu Nam. Wegoloc: accurate prediction of protein subcellular localization using weighted gene ontology terms. *Bioinformatics*, 28(7):1028–1030, 2012.

Gokcen Cilingir, Shira L Broschat, and Audrey OT Lau. Apicoap: The first computational model for identifying apicoplast-targeted proteins in multiple species of apicomplexa. *PloS one*, 7(5):e36598, 2012.

Amy E DeRocher, Isabelle Coppens, Anuradha Karnataki, Luke A Gilbert, Michael E Rome, Jean E Feagin, Peter J Bradley, and Marilyn Parsons. A thioredoxin family protein of the apicoplast periphery identifies abundant candidate transport vesicles in Toxoplasma gondii. *Eukaryotic cell*, 7(9):1518–1529, 2008.

Amy E DeRocher, Anuradha Karnataki, Pashmi Vaney, and Marilyn Parsons. Apicoplast targeting of a Toxoplasma gondii transmembrane protein requires a cytosolic tyrosine-based motif. *Traffic*, 2012.

Pufeng Du. Predicting subcellular localizations of membrane proteins in eukaryotes with weighted gene ontology scores. *Practical Applications of Intelligent Systems*, pages 191–195, 2012.

Pufeng Du, Yang Tian, and Yan Yan. Subcellular localization prediction for human internal and organelle membrane proteins with projected gene ontology scores. *Journal of Theoretical Biology*, 2012.

Olof Emanuelsson, Søren Brunak, Gunnar von Heijne, and Henrik Nielsen. Locating proteins in the cell using targetp, signalp and related tools. *Nature protocols*, 2(4): 953–971, 2007.

DM Engelman, TA Steitz, and A Goldman. Identifying nonpolar transbilayer helices in amino acid sequences of membrane proteins. *Annual review of biophysics and biophysical chemistry*, 15(1):321–353, 1986.

Gerald D Fasman. Chou-fasman prediction of secondary structure. In *Prediction of protein structure and the principles of protein conformation*. Springer, 1989.

Maria E Fichera and David S Roos. A plastid organelle as a drug target in apicomplexan parasites. *Nature*, 390(6658):407–409, 1997.

Walter M Fitch. Distinguishing homologous from analogous proteins. *Systematic Biology*, 19(2):99–113, 1970.

Tobias Fleige, Julien Limenitakis, and Dominique Soldati-Favre. Apicoplast: keep it or leave it. *Microbes and Infection*, 12(4):253–262, 2010.

Bernardo J Foth, Stuart A Ralph, Christopher J Tonkin, Nicole S Struck, Martin Fraunholz, David S Roos, Alan F Cowman, and Geoffrey I McFadden. Dissecting apicoplast targeting in the malaria parasite Plasmodium falciparum. *Science*, 299 (5607):705–708, 2003.

John R Gallagher, Krista A Matthews, and Sean T Prigge. Plasmodium falciparum apicoplast transit peptides are unstructured in vitro and during apicoplast import. *Traffic*, 12(9):1124–1138, 2011.

Lewis Y Geer, Michael Domrachev, David J Lipman, and Stephen H Bryant. Cdart: protein homology by domain architecture. *Genome research*, 12(10):1619–1623, 2002.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann,

and Ian H Witten. The weka data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18, 2009.

Cynthia Y He, Michael K Shaw, Charles H Pletcher, Boris Striepen, Lewis G Tilney, and David S Roos. A plastid segregation defect in the protozoan parasite Toxoplasma gondii. *The EMBO journal*, 20(3):330–339, 2001.

Uwe Hobohm, Michael Scharf, Reinhard Schneider, and Chris Sander. Selection of representative protein data sets. *Protein Science*, 1(3):409–417, 2008.

KAWS HOFMANN. Tmbase-a database of membrane spanning protein segments. *Biol. Chem. Hoppe-Seyler*, 374:166, 1993.

John H Holland. *Adaptation In Natural And Artificial Systems: An Introductory Analysis With Applications To Biology, Control, And Artificial Intelligence*. MIT Press, 1992a.

John H Holland. Genetic algorithms. *Scientific american*, 267(1):66–72, 1992b.

Wen-Lin Huang, Chun-Wei Tung, Shih-Wen Ho, Shiow-Fen Hwang, and Shinn-Ying Ho. Proloc-go: utilizing informative gene ontology terms for sequence-based prediction of protein subcellular localization. *Bmc Bioinformatics*, 9(1):80, 2008.

Thorsten Joachims. Making large scale svm learning practical. 1999.

Russell A Johnson, Geoffrey I McFadden, and Christopher D Goodman. Characterization of two malaria parasite organelle translation elongation factor g proteins: The likely targets of the anti-malarial fusidic acid. *PloS one*, 6(6):e20633, 2011.

Lukas Käll, Anders Krogh, and Erik LL Sonnhammer. Advantages of combined transmembrane topology and signal peptide predictionthe phobius web server. *Nucleic acids research*, 35(suppl 2):W429–W432, 2007.

Anuradha Karnataki, Amy DeRocher, Isabelle Coppens, Coral Nash, Jean E Feagin, and Marilyn Parsons. Cell cycle-regulated vesicular trafficking of Toxoplasma apt1, a protein localized to multiple apicoplast membranes. *Molecular microbiology*, 63 (6):1653–1668, 2007a.

Anuradha Karnataki, Amy E DeRocher, Isabelle Coppens, Jean E Feagin, and Marilyn Parsons. A membrane protease is targeted to the relict plastid of Toxoplasma via an internal signal sequence. *Traffic*, 8(11):1543–1553, 2007b.

James D Kelly and Lawrence Davis. A hybrid genetic algorithm for classification. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence*, pages 645–650, 1991.

Sabine Köhler, Charles F Delwiche, Paul W Denny, Lewis G Tilney, Paul Webster, RJM Wilson, Jeffrey D Palmer, and David S Roos. A plastid of probable green algal origin in apicomplexan parasites. *Science*, 275(5305):1485–1489, 1997.

Anders Krogh, BjoÈrn Larsson, Gunnar Von Heijne, Erik LL Sonnhammer, et al. Predicting transmembrane protein topology with a hidden markov model: application to complete genomes. *Journal of molecular biology*, 305(3):567–580, 2001.

Ambrish Kumar, Aiman Tanveer, Subir Biswas, Edupuganti VS Ram, Ankit Gupta, Bijay Kumar, and Saman Habib. Nuclear-encoded dnaj homologue of Plasmodium falciparum interacts with replication ori of the apicoplast genome. *Molecular microbiology*, 75(4):942–956, 2010.

Jack Kyte, Russell F Doolittle, et al. A simple method for displaying the hydropathic character of a protein. *Journal of molecular biology*, 157(1):105–132, 1982.

Liqi Li, Yuan Zhang, Lingyun Zou, Changqing Li, Bo Yu, Xiaoqi Zheng, and Yue Zhou. An ensemble classifier for eukaryotic protein subcellular location prediction using gene ontology categories and amino acid hydrophobicity. *PLoS One*, 7(1): e31057, 2012.

Weizhong Li and Adam Godzik. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics*, 22(13):1658–1659, 2006.

Liting Lim, Ming Kalanon, and Geoffrey I McFadden. New proteins in the apicoplast membranes: time to rethink apicoplast protein targeting. *Trends in parasitology*, 25(5):197–200, 2009.

Geoffrey I McFadden. Plastid in human parasites. *Nature*, 381:482, 1996.

Geoffrey I McFadden and David S Roos. Apicomplexan plastids as drug targets. *Trends in microbiology*, 7(8):328–333, 1999.

Suyu Mei, Wang Fei, and Shuigeng Zhou. Gene ontology based transfer learning for protein subcellular localization. *BMC bioinformatics*, 12(1):44, 2011.

Kai Michelsen, Hebao Yuan, and Blanche Schwappach. Hide and run. *EMBO reports*, 6(8):717–722, 2005.

Melanie Mitchell, John H Holland, Stephanie Forrest, et al. When will a genetic algorithm outperform hill climbing? *Advances in neural information processing systems*, pages 51–51, 1994.

Tom M Mitchell. Machine learning. wcb, 1997.

Armando M Neto, Denise A Alvarenga, Antônio M Rezende, Sarah S Resende, Ricardo S Ribeiro, Cor JF Fontes, Luzia H Carvalho, and Cristiana FA de Brito. Improving n-terminal protein annotation of Plasmodium species based on signal peptide prediction of orthologous proteins. *Malaria journal*, 11(1):375, 2012.

Thomas Nordahl Petersen, Søren Brunak, Gunnar von Heijne, and Henrik Nielsen. Signalp 4.0: discriminating signal peptides from transmembrane regions. *Nature Methods*, 8(10):785–786, 2011.

Andrea Pierleoni, Pier Luigi Martelli, and Rita Casadio. Memloci: predicting subcellular localization of membrane proteins in eukaryotes. *Bioinformatics*, 27(9): 1224–1230, 2011.

Foster Provost. Machine learning from imbalanced data sets 101. In *Proceedings of the AAAI2000 Workshop on Imbalanced Data Sets*, 2000.

Stuart A Ralph, Giel G van Dooren, Ross F Waller, Michael J Crawford, Martin J Fraunholz, Bernardo J Foth, Christopher J Tonkin, David S Roos, and Geoffrey I McFadden. Tropical infectious diseases: metabolic maps and functions of the Plasmodium falciparum apicoplast. *Nature Reviews Microbiology*, 2(3):203–216, 2004.

Sheila M Reynolds, Lukas Käll, Michael E Riffle, Jeff A Bilmes, and William Stafford Noble. Transmembrane topology and signal peptide prediction using dynamic bayesian networks. *PLoS computational biology*, 4(11):e1000213, 2008.

David S Roos, Michael J Crawford, Robert GK Donald, Jessica C Kissinger, Leszek J Klimczak, and Boris Striepen. Origin, targeting, and function of the apicomplexan plastid. *Current opinion in microbiology*, 2(4):426–432, 1999.

Ken Sato and Akihiko Nakano. Emp47p and its close homolog emp46p have a tyrosine-containing endoplasmic reticulum exit signal and function in glycoprotein secretion in Saccharomyces cerevisiae. *Molecular biology of the cell*, 13(7):2518–2532, 2002.

Hayley J Sharpe, Tim J Stevens, and Sean Munro. A comprehensive comparison of transmembrane domains reveals organelle-specific properties. *Cell*, 142(1):158–169, 2010.

Lilach Sheiner, Jessica L Demerly, Nicole Poulsen, Wandy L Beatty, Olivier Lucas, Michael S Behnke, Michael W White, and Boris Striepen. A systematic screen to discover and analyze apicoplast proteins identifies a conserved and essential protein import factor. *PLoS pathogens*, 7(12):e1002392, 2011.

Christopher J Tonkin, David S Roos, and Geoffrey I McFadden. N-terminal positively charged amino acids, but not their exact position, are important for apicoplast transit peptide fidelity in Toxoplasma gondii. *Molecular and biochemical parasitology*, 150(2):192–200, 2006.

Christopher J Tonkin, Ming Kalanon, and Geoffrey I McFadden. Protein targeting to the malaria parasite plastid. *Traffic*, 9(2):166–175, 2008.

GG Van Dooren, RF Waller, GI McFadden, KA Joiner, and DS Roos. Traffic jams: Protein transport in Plasmodium falciparum. *Parasitology Today*, 16(10):421–427, 2000.

Giel G van Dooren, Vanessa Su, Marthe C D'Ombrain, and Geoffrey I McFadden. Processing of an apicoplast leader sequence in Plasmodium falciparum and the

identification of a putative leader cleavage enzyme. *Journal of Biological Chemistry*, 277(26):23612–23619, 2002.

Vladimir N Vapnick. Statistical learning theory. *J. Wiley and Sons Inc. Nova York*, 1998.

Vladimir Vapnik. *The nature of statistical learning theory*. springer, 1999.

Celine Vens, Marie-Noëlle Rosso, and Etienne GJ Danchin. Identifying discriminative classification-based motifs in biological sequences. *Bioinformatics*, 27(9):1231–1238, 2011.

Gunnar Von Heijne et al. Membrane protein structure prediction. hydrophobicity analysis and the positive-inside rule. *Journal of molecular biology*, 225(2):487–494, 1992.

Ross F Waller, Patrick J Keeling, Robert GK Donald, Boris Striepen, Emanuela Handman, Naomi Lang-Unnasch, Alan F Cowman, Gurdyal S Besra, David S Roos, and Geoffrey I McFadden. Nuclear-encoded proteins target to the plastid in Toxoplasma gondii and Plasmodium falciparum. *Proceedings of the National Academy of Sciences*, 95(21):12352–12357, 1998.

Ross F Waller, Michael B Reed, Alan F Cowman, and Geoffrey I McFadden. Protein

trafficking to the plastid of Plasmodium falciparum is via the secretory pathway. *The EMBO journal*, 19(8):1794–1802, 2000.

Ben J. Woodcroft, Krystie-Lee Scanlon, Maria Doyle, Terry Speed, and Stuart A. Ralph. A database of published protein sub-cellular localisation in apicomplexa. http://apiloc.bio21.unimelb.edu.au. Accessed: version 3 (curated until May 28, 2011).

Jochen Zuegge, Stuart Ralph, Michael Schmuker, Geoffrey I McFadden, and Gisbert Schneider. Deciphering apicoplast targeting signals–feature extraction from nuclear-encoded precursors of Plasmodium falciparum apicoplast proteins. *Gene*, 280(1):19–26, 2001.