Improving Protein Sub-cellular Localization Prediction Through Semi-supervised Learning

Jorge Alberto Jaramillo-Garzón
Instituto Tecnológico Metropolitano
Medellín, Colombia
Email: jorgejaramillo@itm.edu.co

César Germán Castellanos-Domínguez
Universidad Nacional de Colombia
Manizales, Colombia
Email: cgcastellanosd@unal.edu.co

***Abstract*—Prediction of sub-cellular localization of proteins is a fundamental task in bioinformatics, since it can provide useful information to determine its function. Several prediction techniques have been proposed in the recent years and methods based on machine learning techniques have achieved state of the art classification, usually employing support vector machines and neural networks. However, those methods need high amounts of labeled samples (proteins with known function) in order to train accurate classifiers, and such information is not easily available for this task. In this paper, an alternative methodology that uses semi-supervised learning is proposed. This type of machine learning allows to use unlabeled samples (which are easily available) in order to improve the estimation of the classifiers. All the needed steps for using semi-supervised learning in the problem of predicting protein sub-cellular localizations are described in detail and the methodology is compared with the standard supervised alternative. The results show that using semi-supervised learning significantly improves the prediction performance of the classifier in several cases, proving to be a valuable tool in bioinformatics.**

Keywords-Sub-cellular localization, Gene Ontology, Semi-supervised, Support Vector Machines.

I. INTRODUCTION

One of the most important tasks in modern bioinformatics is to provide reliable functional annotations for gene products. Predicting protein sub-cellular localizations allows researchers to obtain useful information for revealing protein functions and helping to understand the pathways that regulate biological processes [1]. The localization of specific proteins can be experimentally determined by assays of expression of green fluorescent proteins in order to monitor its intrinsic fluorescence and subsequently locate it in the cell [2]. However, such procedures become expensive and highly time consuming when they have to be applied in high-throughput projects, which yields to the need of developing computational predictors able to identify the sub-cellular location of novel proteins based on its sequence information alone [3].

Several predictors have been proposed in the recent years (for full surveys, see [4, 5, 6]). In particular, most recent methods have used machine learning techniques trained over feature spaces of physical-chemical, statistical or locally-based attributes. Those methods employ techniques such as neural networks (ProtFun [7]), Bayesian multi-label classifiers 8]) and support vector machines (SVM-Prot [9], GOKey [10], PoGO 11]), obtaining high performance results in their own respective databases, mostly composed by model organisms such as bacteria and a few high order species [12].

One of the main limitations of machine learning methods, however, is that they need relatively high amounts of training data in order to learn reliable classification models. Such training data refers to "labeled instances", that is, enough protein sequences which function must be already known. It is a known fact, however, that only a small number of proteins have actually been annotated for certain functions [4]. Under such circumstances, semi-supervised learning methods provide an alternative approach to protein annotation. In semi-supervised learning methods, additionally to labeled data, the algorithm is provided with an amount of unlabeled data that can be used to improve the estimations of the classifier.

This work presents an implementation of semi-supervised learning using semi-supervised support vector machines (S^3VM) for predicting protein sub-cellular localizations. The results obtained with this approach show that using semi-supervised learning significantly improves the prediction performance of the standard support vector machine (SVM) in several cases, proving to be a valuable tool in bioinformatics.

The following section describes the theoretical background about SVM and S^3VM. Next, the "Experimental setup" section describes the database and all the components of the proposed methodology. The final two sections present the results and conclusions, respectively.

II. THEORETICAL BACKGROUND

A. Support vector machines

Support vector machines (SVM) are powerful tools for solving classification problems, designed over a strong theoretical background based on the idea of minimizing the structural risk [13]. For a non-linear SVM, the objective is to find a classification function of the form:

$$f_{(\boldsymbol{w},b)}(\boldsymbol{x})=\langle \boldsymbol{w},\boldsymbol{x}\rangle+b \qquad (1)$$

 ISBN: 978-1-61208-335-3

where $\langle\cdot,\cdot\rangle$ represents the dot product. A vector of parameters can be defined as $\boldsymbol{\theta} = [\boldsymbol{w},b]$, and the optimization problem can be stated as follows:

$$\boldsymbol{\theta}^{*}=\arg\min_{\boldsymbol{\theta}\in\mathcal{T}}\left\{\frac{1}{2}\|\boldsymbol{\theta}\|^{2}+C\sum_{i=1}^{L}\ell\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i)y_i\right)\right\} \qquad (2)$$

where $\ell(t)=\max(0,1-t)$ is the hinge loss function and C is a trade-off parameter regulating the complexity of the model. For the non-linear case, the data are first mapped in a high dimensional Hilbert space $\mathcal{H}$ through a mapping $\Phi:\mathcal{X}\mapsto\mathcal{H}$, and then a linear decision boundary is constructed in that space. The mapping Φ can be explicitly computed or only implicitly through the use of a kernel function K such that $K(\boldsymbol{x}_1,\boldsymbol{x}_2)=\langle\Phi(\boldsymbol{x}_1),\Phi(\boldsymbol{x}_2)\rangle$. The Representer Theorem can be used to show that the solution function has the form:

$$f_{\boldsymbol{\theta}^{*}}(\boldsymbol{x})=\sum_{i=1}^{L}\alpha_i K(\boldsymbol{x},\boldsymbol{x}_i) \qquad (3)$$

where the coefficients α_i can be found with a conventional quadratic optimization algorithm. The Gaussian kernel is the most commonly used because of its attractive features such as structure preservation [14]. This kernel is computed by:

$$K(\boldsymbol{x}_1,\boldsymbol{x}_2)=e^{-\frac{\|\boldsymbol{x}_1-\boldsymbol{x}_2\|_2}{2\sigma}} \qquad (4)$$

where σ is the dispersion parameter that must be properly chosen by the user. In this work, the SVM is trained with the '*kernlab*' package, available in R-CRAN [15].

B. Semi-supervised support vector machines

Semi-Supervised SVMs (S^3VMs) emerged as an extension to standard SVMs for semi-supervised learning. S^3VMs find a labeling for all the unlabeled data, and a separating hyperplane, such that maximum margin is achieved on both the labeled data and the (now labeled) unlabeled data. As a result, unlabeled data guides the decision boundary away from dense regions. The assumption of S^3VMs is that the classes are well-separated, such that the decision boundary falls into a low density region in the feature space, and does not cut through dense unlabeled data [16, chapter 6].

In a similar way than the conventional SVMs, the optimization problem for an S^3VMs can be stated as follows:

$$\boldsymbol{\theta}^{*}=\arg\min_{\boldsymbol{\theta}\in\mathcal{T}}\left\{\frac{1}{2}\|\boldsymbol{\theta}\|^{2}+C\sum_{i=1}^{L}\ell\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i)y_i\right)+\ldots + \lambda\sum_{i=L+1}^{L+U}\ell\left(|f_{\boldsymbol{\theta}}(\boldsymbol{x}_i)|\right)\right\} \qquad (5)$$

where $\ell(t)=\max(0,1-t)$ is the hinge loss function, C is the trade-off parameter and λ is a new regularization parameter. The first two terms in the above equation correspond to the traditional solution for the standard supervised SVM shown in equation (2), while the last term puts $f_{\boldsymbol{\theta}}(\boldsymbol{x}_i)$ of the unlabeled points $\boldsymbol{x}_i$ away from 0 (thereby implementing the low density assumption) [17].

Again, as in the supervised case, the kernel trick can be used for constructing non-linear S^3VMs. While the optimization in SVM is convex and can be solved with QP-hard complexity, optimization in S^3VM is a non-convex combinatorial task with NP-Hard complexity. Most of the recent work in S^3VM has been focused on the optimization procedure (a full survey in this matter can be found in [18]). Among the proposed methods for solving the non-convex optimization problem associated with S^3VMs, one of the first implementations is the S^3VM^{light} by Joachims [19], which is based on local combinatorial search guided by a label switching procedure. Chapelle et. al. [20] presented a method based on gradient descent on the primal, that performs significantly better than the optimization strategy pursued in S^3VM^{light}; the work by Chapelle et. al. [17] proposes the use of a global optimization technique known as "continuation", often leading to lower test errors than other optimization algorithms; Collobert et. al. [21] uses the Concave-Convex procedure, providing a highly scalable algorithm in the nonlinear case.

III. Experimental Setup

A. Database

This work uses the database designed in [12]. Such database comprises all the available *Embryophyta* proteins at UniProtKB/Swiss-Prot database [22], with at least one annotation in the Gene Ontology Annotation (GOA) project [23]. In order to avoid the presence of protein families that could bias the results, the dataset was filtered at an identity cutoff of 30%.

The main set comprises a total of 2210 sequences associated to 20 different sub-cellular localizations. Those localizations correspond to the Cellular component ontology defined by the plants GO slim [24]. Categories with less than 30 proteins were discarded because they did not have enough samples to train a statistically reliable classifier. All the available *Embryophyta* proteins at UniProtKB/Swiss-Prot database that has no entries in the GOA project were added as the core set of unlabeled instances. Proteins associated to the nodes in the functional path of each GO term were also left as unlabeled instances regarding that classifier. Finally, 22000 unlabeled instances were randomly chosen in order to accomplish an approximate relation of ten unlabeled instances per each labeled one.

 ISBN: 978-1-61208-335-3

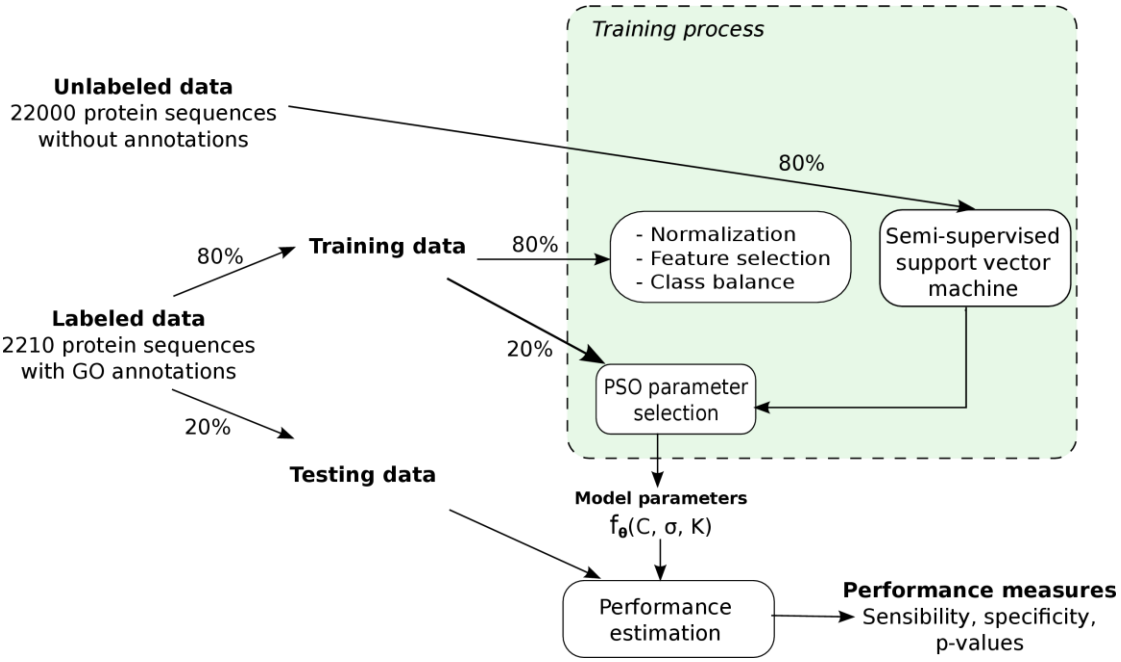

Figure 1: Main methodology

B. *Classification Methodology*

Figure 1 shows the main methodology for classification. The CCP - S^3VM [21] was used as base classifier, with the Gaussian kernel. All the parameters of the algorithm, including the dispersion of the kernels, the trade-off parameters of the SVMs, the regularization constants were tuned with a particle swarm optimization meta-heuristic [25].

In order to allow samples to be associated to multiple categories, decision making was implemented following the one-against-all strategy. The method produced a strong class imbalance that was tackled using the Synthetic Minority Over-sampling Technique (SMOTE) [26].

Feature selection was carried out before trying to induce any decision rule (classifier) because, having a limited number of training examples, excessive features would possibly overfit the training data. For this purpose, the *Fast Correlation-Based Filter* presented in [27] was used.

In order to estimate the performance of the predictive model, a 5-fold cross-validation strategy is implemented. In such strategy, the test procedure is repeated five times, and each time an 80% of the data is used for adjusting the SVM parameters and training the model, while the remaining 20% is used as testing samples.

IV. RESULTS

In order to analyze the results obtained with the proposed methodology, sensitivity and specificity for each GO term were computed. The obtained results are compared with the ones obtained in [12] with the commonly used BLASTp method (Figure 2), as well as with a standard SVM (Figure 3). Bars in the left plots show sensitivity and specificity of the Lap-S^3VM and lines depict geometric mean for S^3VM (orange), BLASTp (blue) and the classical supervised SVM (green). Right plots depict the p-values obtained by paired t-tests at a 95% significance level. Orange bars show the cases when the S^3VM significantly outperforms BLASTp and the supervised SVM, in Figures 2 and 3, respectively. On both figures, the best predicted categories are ordered from top to bottom.

While the comparison with BLASTp provides information about the applicability of the methodology compared with the alignment based methods, the main purpose of this comparing the SVM with the S3VM is to verify whether or not the inclusion of the additional cluster-based semi-supervised term in the training of the SVM improves the performance of the system. This can be understood as the accomplishment of the cluster assumption when the unlabeled data is incorporated to the training process.

Figure 2 shows that there are only two cellular components for which there is no statistically significant difference between BLASTp and the S3VM: *Perixosome* and *Endosome*. For all the remaining eighteen cellular components, the semi-supervised method obtained statistically significant superior performance.

Regarding Figure 3, it can be observed that eight cellular components were significantly improved, while another two (*Mitochondria* and *Cytoplasm**) also reached high p-values over 0.9.

 ISBN: 978-1-61208-335-3

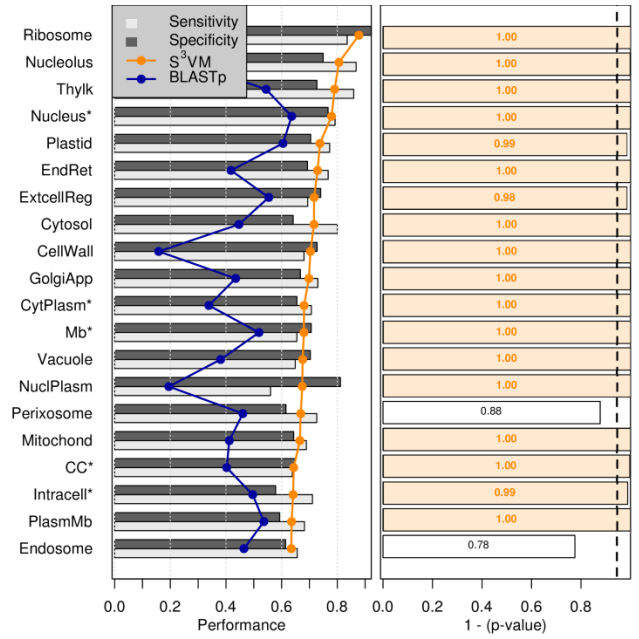

Figure 2: Comparison between the S3VM method and BLASTp

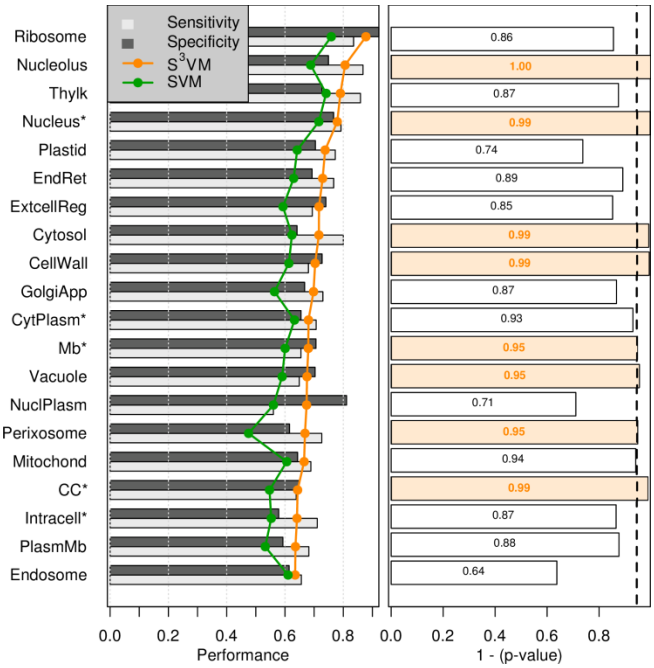

Figure 3: Comparison between the S3VM method and the supervised SVM

This results show that the inclusion of the additional information improves the estimation of the classification models and thus, provides an efficient way for alleviating the lack of labeled data in the field of bioinformatics. Although several localizations were not improved over a statistically significant threshold, no one of them degraded its performance with the inclusion of the additional data, proving to accomplish the underlying assumptions of semi-supervised learning.

Finally, in order to verify the influence of the number of unlabeled instances included in the training, process, several experiments were performed, varying the number of unlabeled instances from 0 to 2200. As exemplary cases, the results for six cellular components (*nucleus*, *cell wall*, *vacuole*, *cytosol*, *membrane* and the root node of the ontology, *cellular component*) are depicted on Figure 4. It is important to point out that these tests where done with the same SVM parameters across all the experiment and, consequently, the predictor is not optimized for each case. However it allows understanding the main influence of the unlabeled data.

From these results, it can be observed that the effect of progressively including unlabeled instances is reducing the specificity of the classifier, while increasing the sensitivity. In general terms, when no unlabeled instances are included, specificity is very high and sensitivity is almost zero. This means that the classifier is rejecting all the samples for that given GO term. The semi-supervised assumption allows the system to recognize the positive samples, thus increasing the overall performance of the predictor.

V. CONCLUSION

This work presented an experimental analysis of the suitability of semi-supervised methods for the prediction of protein sub-cellular localizations. The results show that semi-supervised learning applied to the prediction of GO terms, significantly outperforms the supervised learning approach in several cases. As future work another semi-supervised strategies must be explored in order to analyze if different assumptions (for example, graph-based methods) can be able to provide better results for the cases where this methodology was not significantly superior.

ACKNOWLEDGMENT

This work is within the framework of the Dirección de Investigaciones de Manizales (DIMA) of the Universidad Nacional de Colombia and the Centro de Investigación of the Instituto Tecnológico Metropolitano. The work has been partially founded by Colciencias grant 111952128388.

REFERENCES

[1] K. Chou, H. Shen, and E. Newbigin, “Plant-mPLoc: A Top-Down Strategy to Augment the Power for Predicting Plant Protein Subcellular Localization,” PloS one, vol. 5, no. 6, 2010, pp. 259–270.

[2] P. Baldi and S. Brunak, Bioinformatics: the machine learning approach. 1em plus 0.5em minus 0.4emThe MIT Press, 2001.

[3] K. Chou and H. Shen, “Recent progress in protein subcellular location prediction,” Analytical Biochemistry, vol. 370, no. 1, 2007, pp. 1–16.

[4] X. Zhao, L. Chen, and K. Aihara, “Protein function prediction with high-throughput data,” Amino Acids, vol. 35, no. 3, 2008, pp. 517–530.

[5] G. Pandey, V. Kumar, and M. Steinbach, “Computational approaches for protein function prediction: A survey,” Department of Computer

 ISBN: 978-1-61208-335-3

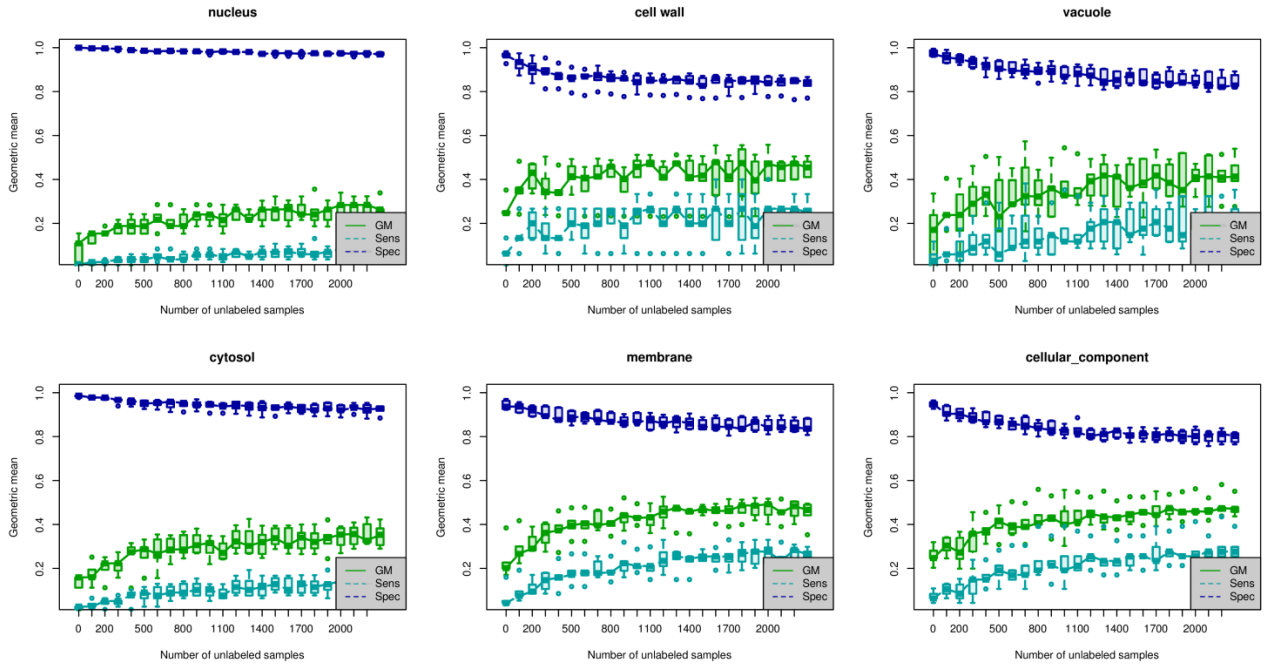

Figure 4: Variation of the number of unlabeled samples included in the training process

Science and Engineering, University of Minnesota, Twin Cities, Tech. Rep. 06-028, 2006.

[6] I. Friedberg, "Automated protein function prediction–the genomic challenge," Briefings in Bioinformatics, vol. 7, no. 3, 2006, p. 225.

[7] L. Jensen, R. Gupta, H. Staerfeldt, and S. Brunak, "Prediction of human protein function according to Gene Ontology categories," pp. 635–642, 2003.

[8] J. Jung and M. R. Thon, "Gene function prediction using protein domain probability and hierarchical Gene Ontology information," 2008 19th International Conference on Pattern Recognition, 2008, pp. 1–4.

[9] C. Z. Cai, "SVM-Prot: web-based support vector machine software for functional classification of a protein from its primary sequence," Nucleic Acids Research, vol. 31, no. 13, 2003, pp. 3692–3697.

[10] R. Bi, Y. Zhou, F. Lu, and W. Wang, "Predicting Gene Ontology functions based on support vector machines and statistical significance estimation," Neurocomputing, vol. 70, no. 4-6, 2007, pp. 718–725.

[11] J. Jung, G. Yi, S. a. Sukno, and M. R. Thon, "PoGO: Prediction of Gene Ontology terms for fungal proteins." BMC bioinformatics, vol. 11, 2010, p. 215.

[12] J. A. Jaramillo-Garzón, J. J. Gallardo-Chacón, C. G. Castellanos-Domínguez, and A. Perera-Lluna, "Predictability of gene ontology slim-terms from primary structure information in embryophyta plant proteins," BMC bioinformatics, vol. 14, no. 1, 2013, p. 68.

[13] V. Vapnik, Statistical learning theory. plus 0.5em minus 0.4emWiley New York, 1998.

[14] Z. Liu, M. J. Zuo, and H. Xu, "Parameter selection for Gaussian radial basis function in support vector machine classification," 2012 International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering, Jun. 2012, pp. 576–581.

[15] A. Karatzoglou, A. Smola, K. Hornik, and A. Zeileis, "kernlab – an S4 package for kernel methods in R," Journal of Statistical Software, vol. 11, no. 9, 2004, pp. 1–20.

[16] X. Zhu and A. B. Goldberg, "Introduction to semi-supervised learning," Synthesis lectures on artificial intelligence and machine learning, vol. 3, no. 1, 2009, pp. 1–130.

[17] O. Chapelle, M. Chi, and A. Zien, "A continuation method for semi-supervised SVMs," Proceedings of the 23rd international conference on Machine learning - ICML '06, 2006, pp. 185–192.

[18] O. Chapelle, V. Sindhwani, and S. S. Keerthi, "Optimization techniques for semi-supervised support vector machines," The Journal of Machine Learning Research, vol. 9, 2008, pp. 203–233.

[19] T. Joachims, "Transductive inference for text classification using support vector machines," in ICML, vol. 99, 1999, pp. 200–209.

[20] O. Chapelle and A. Zien, "Semi-supervised classification by low density separation," Proceedings of the tenth international workshop on, 2005.

[21] R. Collobert, F. Sinz, J. Weston, and L. Bottou, "Large scale transductive SVMs," The Journal of Machine …, vol. 1, 2006, pp. 1687–1712.

[22] E. Jain, A. Bairoch, S. Duvaud, I. Phan, N. Redaschi, B. Suzek, M. Martin, P. McGarvey, and E. Gasteiger, "Infrastructure for the life sciences: design and implementation of the UniProt website," BMC bioinformatics, vol. 10, no. 1, 2009, p. 136.

[23] D. Barrell, E. Dimmer, R. Huntley, D. Binns, C. O'Donovan, and R. Apweiler, "The GOA database in 2009–an integrated Gene Ontology Annotation resource," Nucleic Acids Research, 2008.

[24] T. Berardini, S. Mundodi, L. Reiser, E. Huala, M. Garcia-Hernandez, P. Zhang, L. Mueller, J. Yoon, A. Doyle, G. Lander et al., "Functional annotation of the Arabidopsis genome using controlled vocabularies," Plant Physiology, vol. 135, no. 2, 2004, p. 745.

[25] J. Kennedy and R. Eberhart, "Particle swarm optimization," Proceedings of ICNN'95 - International Conference on Neural Networks, vol. 4, 1995, pp. 1942–1948.

[26] N. Chawla, K. Bowyer, L. Hall, and W. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," Journal of Artificial Intelligence Research, vol. 16, no. 3, 2002, pp. 321–357.

[27] L. Yu and H. Liu, "Efficient feature selection via analysis of relevance and redundancy," The Journal of Machine Learning Research, vol. 5, 2004, pp. 1205–1224.

 ISBN: 978-1-61208-335-3