A Literature Review of Voting Ensemble and Semi-supervised Learning

Dong Cao
Computer Science Department at Earlham College
Richmond, Indiana, USA
dcaohuu18@earlham.edu

ACM Reference Format:
Dong Cao. 2021. A Literature Review of Voting Ensemble and Semi-supervised Learning. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Ensemble learning has received great attention from the Machine Learning community as the aggregated output of multiple learners is often better than that of any single one of them. For classification problems specifically, several methods have been proposed to combine multiple classifiers' predictions: Voting, Bagging, Boosting, Stacking, etc. Among these methods, Voting is very important not only because it is a simple, intuitive, and effective method in and of itself but also because it plays the role of determining the collective prediction in other ensemble frameworks like Bagging and Boosting. However, there has not been much discussion about leveraging the consensus of the base classifiers in unlabelled data in order to better inform the final prediction. My proposed method identifies the data points where the ensemble reaches consensus and where conflict arises in the unlabeled space. A meta weighted KNN model is trained upon this half-labeled set with the labels of the consensus and the conflict points marked as "Unknown", which is treated as a new, additional class. The predictions of the meta model are expected to better inform the decision of the ensemble in the case of conflict. This suggestion may be linked to Semi-supervised Learning. In fact, some Semi-supervised algorithms make use of the same concept. In this literature review, I will examine the relevant pieces in each of the two domains: Voting Ensemble and Semi-supervised Learning and how my proposed method is located between them and makes use of their advantages to solve a particular problem.

2 VOTING METHODS

In this section, I will present some of the most commonly used voting schemes as well as their strengths and weaknesses.

2.1 Majority Voting

Majority Voting or Plural Voting is probably the most well-known voting system due to its straightforwardness. A class is assigned to a sample if it receives a majority of votes from the base classifiers. However, there are different definitions of "majority." Depending on the situation, it can mean unanimity, simple majority (i.e. more than 50% of the base classifiers agree on a label). Yet, the most common approach has been that whichever class receiving the most votes "wins" and becomes the ultimate prediction. Simple Majority Voting makes a lot of assumptions about the relative accuracy of the classifiers and each classifiers' performance with respect to a particular class [9]. In reality, most of these assumptions are not accurate. However, the simplicity and efficiency of this method still makes it one of the favorite options of Machine Learning practitioners.

2.2 Weighted Voting

Another popular voting technique is Weighted Voting. Weighted Voting drops one of the assumptions made by Majority Voting [9]. Instead of considering the predictions of the base classifiers equally likely to be accurate, Weighted Voting attaches different weights to the classifiers' predictions based on their performance in the training set. One of the most well-known representative of this approach is the voting system of AdaBoost, which trains a number of weak learners, weights them differently based on their error rates, and aggregates their predictions by taking into account these weights [5]. Other techniques make use of fuzzy sets [3], particle swarm optimization [7], genetic algorithm [11], or weight assignment based on a classifier's relative performance with respect to others' [4].

2.3 Support Function

With a support function, instead of being confined to a single output class, a base classifier can provide their predictions in terms of the probabilities of a sample belonging to each of the available classes. The term "Support Function" is used by Woźniak et al. in their survey of classifiers combination [13].

One of the widely used type of probability is the a posteriori probability. Kittler et al. have proposed a variety of rules by which a class' a posteriori probabilities from different base classifiers can be combined [8]. Later work has focused on comparing the effectiveness of these rules under various conditions [1].

The value of the support function can also be the rankings of the classes. In this case, the method is known as Borda Count, which outperforms Majority Voting in some experiments [10].

2.4 Important Observations

While the aforementioned methods tackle the problem differently, they all agree on one point: it is critical for the ensemble to be diverse. This concept of diversity can be interpreted and measured in a variety of ways. Usually, it is important that the classifiers don't

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
Conference'17, July 2017, Washington, DC, USA
© 2021 Association for Computing Machinery.
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

make the same mistake together. Some diversification strategies are using different underlying algorithms for the base classifiers or different feature sets [8], and bootstrapping.

3 SEMI-SUPERVISED LEARNING

Semi-supervised learning is a broad field. Therefore, I will only examine the areas that are relevant to making use of the ensemble's consensus on unlabeled data. Before I go into the details of each area, let us quickly touch upon the rudiments of semi-supervised learning. The big problem that semi-supervised learning tries to solve is that labeled data for training is often insufficient and difficult to acquire while unlabeled data is abundant. Semi-supervised learning aims to fully exploit the few labeled samples available to extract patterns from the pool of ample unlabeled data [12]. The term "semi-supervised" suggests that many of the methods in this field are combinations of the approaches taken in supervised and unsupervised learning.

3.1 Inductive versus Transductive

The landscape of semi-supervised learning methods comprises of two major approaches: Inductive and Transductive [12]. The goal of an inductive framework is to build a mechanism that can independently predict unlabeled samples one by one. This goal is shared with most of the supervised algorithms but the training process of an inductive algorithm takes in both labeled and unlabeled data [12]. On the other hand, a transductive method seeks to optimize the predictions for each space of data. This space contains samples that are either labeled or unlabeled and a transductive algorithm attempts to use the distribution of the entire space to provide a set of predictions for all data points. In other words, the input for a transductive algorithm is the whole data space, not a single data point [12]. In this aspect, my method is similar to the transductive approach when the meta model needs to be fed the complete unlabeled set. However, while most transductive algorithms use graph theory to model the similarity among the data points [12], my method looks to draw a connection between the patterns in the training set and those in the unlabeled set via inspecting the consensus of the base classifiers.

3.2 Tri-Training

Introduced by Zhou and Li in 2005, Tri-Training is an inductive method designed to address the limitations of the Co-Training algorithm that had sparked widespread attention earlier [14]. For Co-Training to work, it is crucial that the dataset can be divided into two sufficient and independent views for simultaneous training of two separate learners [2], which is not always possible to satisfy. To bypass this hard requirement, Tri-Training uses three classifiers instead of two. All of them are trained upon the same complete dataset. When it comes to leveraging unlabeled data for "refinement", a classifier is given a sample to train with the label agreed upon by the other two [14]. A variation of Tri-Training is Multi-Train when more than three classifiers are used and a sample is accepted for the refinement of one classifier if a majority of the rest of the classifiers return the same label [6].

It is not difficult to point out the similarities between my idea and that of Tri-Training, when my meta KNN model learns from the data labeled based on the base classifiers' consensus. However, there are fundamental differences between my method and Tri-Training:

- As mentioned, Tri-Training falls under the inductive approach while my method is generally transductive.
- My meta KNN model does not learn from the labeled data in the training set.
- There is no co-training. In other words, my meta KNN model does not affect the base classifiers in any way. Hence, there is no refinement of the base classifiers using unlabeled data.
- My meta KNN model also takes into account the uncertainty of the nearby conflict points.

4 CONCLUSION

In this literature review, I have covered two major domains: Voting Ensemble and Semi-supervised Learning, and how my method stands between them. The theoretical motivation for my method is that it can take advantage of the collective decisions of a voting system and summarize how strong these collective patterns are in the unlabeled set to assist with the classification of the conflict points. The distinctions between my method and the techniques discussed in this review, specifically Tri-Training, reflect the different objectives that the two methods are pursuing. While Tri-Training, a representative of semi-supervised learning, tackles the lack of labeled data, my method aims at detecting fake patterns that exist in the training set but not in the unlabeled set, thereby reducing the chance of overfitting.

REFERENCES

[1] Luís A. Alexandre, Aurélio C. Campilho, and Mohamed Kamel. 2001. On combining classifiers using sum and product rules. *Pattern Recognition Letters* 22, 12 (2001), 1283–1289.
[2] Avrim Blum and Tom Mitchell. 1998. Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*. 92–100.
[3] Robert Burduk. 2012. Recognition task with feature selection and weighted majority voting based on interval-valued fuzzy sets. In *International Conference on Computational Collective Intelligence*. Springer, 204–209.
[4] Alican Dogan and Derya Birant. 2019. A weighted majority voting ensemble approach for classification. In *2019 4th International Conference on Computer Science and Engineering (UBMK)*. IEEE, 1–6.
[5] Yoav Freund and Robert E. Schapire. 1997. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences* 55, 1 (1997), 119–139.
[6] Shenkai Gu and Yaochu Jin. 2017. Multi-train: A semi-supervised heterogeneous ensemble classifier. *Neurocomputing* 249 (2017), 202–211.
[7] Asma Kausar, M. Ishtiaq, M. Arfan Jaffar, and Anwar M. Mirza. 2010. Optimization of ensemble based decision using PSO. In *Proceedings of the World Congress on Engineering*, Vol. 1. IAENG, 1–6.
[8] Josef Kittler, Mohamad Hatef, Robert PW Duin, and Jiri Matas. 1998. On combining classifiers. *IEEE transactions on pattern analysis and machine intelligence* 20, 3 (1998), 226–239.
[9] Ludmila I. Kuncheva and Juan J. Rodríguez. 2014. A weighted voting framework for classifiers ensembles. *Knowledge and Information Systems* 38, 2 (2014), 259–275.
[10] Florin Leon, Sabina-Adriana Floria, and Costin Bădică. 2017. Evaluating the effect of voting methods on ensemble-based classification. In *2017 IEEE International Conference on INnovations in Intelligent SysTems and Applications (INISTA)*. IEEE, 1–6.
[11] Yasir Mehmood, Muhammad Ishtiaq, Muhammad Tariq, and M. Arfan Jaffar. 2010. Classifier ensemble optimization for gender classification using genetic algorithm. In *2010 International Conference on Information and Emerging Technologies*. IEEE, 1–5.
[12] Jesper E. Van Engelen and Holger H. Hoos. 2020. A survey on semi-supervised learning. *Machine Learning* 109, 2 (2020), 373–440.
[13] Michał Woźniak, Manuel Grana, and Emilio Corchado. 2014. A survey of multiple classifier systems as hybrid systems. *Information Fusion* 16 (2014), 3–17.

[14] Zhi-Hua Zhou and Ming Li. 2005. Tri-training: Exploiting unlabeled data using three classifiers. *IEEE Transactions on knowledge and Data Engineering* 17, 11 (2005), 1529–1541.