On a scalable problem transformation method for multi-label learning

Dora Jambor
Shopify Inc.
dora.jambor@shopify.com

Peng Yu
Shopify Inc.
peng.yu@shopify.com

Abstract

Binary relevance is a simple approach to solve multi-label learning problems where an independent binary classifier is built per each label. A common challenge with this in real-world applications is that the label space can be very large, making it difficult to use binary relevance to larger scale problems. In this paper, we propose a scalable alternative to this, via transforming the multi-label problem into a single binary classification. We experiment with a few variations of our method and show that our method achieves higher precision than binary relevance and faster execution times on a top-K recommender system task.

1 Introduction

A simple and performant approach commonly used in real-world multi-label learning applications is Binary Relevance (BR) [Tsoumakas and Katakis (2007)]. BR is a decomposition method that trains a single binary classifier for each label to classify input instances as relevant or irrelevant for the given label.

As the number of label sets grows exponentially with increases in the number of class labels, a key challenge in multi-label learning is scalability. In the BR framework, when training individual binary classifiers for each label sequentially prohibitively takes a long time, we might resort to parallelising our models across workers. However, this approach increases the I/O cost of reading the input data and transferring the data across the network. In this work, we propose a set of transformations which allows us to solve a multi-label learning problem by solving one binary classification problem. We demonstrate the effectiveness of our proposed method via a real-world top-K recommendation task and show that we are able to solve our problem much faster, and with higher precision.

2 Background and Methodology

As discussed by Tsoumakas et al. (2009), there are two main paradigms to solve multi-label problems: (i) *problem transformation*, (ii) *algorithm adaptation*. The former transforms the learning task into one or more single-label classification tasks, whereas the latter extends specific learning algorithms in order to handle multi-label data directly. Similar to BR, our proposed method is an instance of (i).

Formally, let $X \in \mathbb{R}^{m \times n}$ be a matrix of m instances where each instance is an n-dimensional feature vector, and let $Y \in \{0, 1\}^{m \times k}$ be a matrix of m responses where each response if a k-dimensional label vector.

Inspired by Kesler's construction [Nilsson (1965), Duda and Hart (1973)], an approach to extend learning algorithms for binary classification to the multiclass case [Har-Peled et al. (2003)], we propose a set of transformations on X and Y to convert the multi-label problem into a single binary classification problem. We also introduce a way to map the solutions of the binary classification problem back to the original multi-label setting. These transformations are defined as:

$$X' = diag(\underbrace{X, \dots, X}_{k}) \qquad\qquad Y' = \begin{bmatrix} Y_1 \\ \cdots \\ Y_k \end{bmatrix}$$

where $Y_1, \dots, Y_k$ are $m \times 1$-dimensional vectors corresponding to m responses for each label, i.e. $Y = \begin{bmatrix} Y_1 & \cdots & Y_k \end{bmatrix}$.

Using the transformed $X' \in \mathbb{R}^{mk \times nk}$ and $Y' \in \{0,1\}^{mk \times 1}$, we solve a single binary classification with X' as instances and Y' as responses. After obtaining $\widehat{Y'} = \begin{bmatrix} \widehat{Y'}_1 \\ \cdots \\ \widehat{Y'}_k \end{bmatrix}$, our estimates of Y', we assign $\widehat{Y} = \begin{bmatrix} \widehat{Y'}_1 & \cdots & \widehat{Y'}_k \end{bmatrix}$ as the predicted label scores of the original multi-label problem.

3 Experiments and discussion

We conduct our experiments on an internal dataset containing 705,093 users' app installations for the top 100 most popular apps [1] on Shopify App Store [2]. In this task, both X and Y are the binary user-item matrix composed of each user's historical app installations. We then perform a three fold time-series based split to obtain three pairs of train-test dataset. [3]

We compare our method (termed DiagT) against BR as a baseline. We seek to answer if our approach is amenable to the application of dimensionality reduction techniques due to the sparsity and large size of X' by investigating the performance of a few variations of DiagT, utilizing the hashing trick [Langford et al. (2007)] and random undersampling [4].

We show in Table 1 that DiagT and its variations obtain higher precisions in models DiagT, DiagT-hb0.9, and DiagT-rus and have faster execution time compared to BR. Model DiagT-rus-hb0.9, which employs both the hashing trick and undersampling suffers from a high bias. The hashing bucket ratio 0.9 is chosen after performing hyperparameter tuning.

4 Conclusion

We proposed a problem transformation method to solve multi-label learning via a single binary classification that is shown to have clear improvements in execution time and precision compared to the binary relevance method. In future work, we intend to perform a more extensive hyperparameter search, experiment with different dimensionality reduction techniques, and compare our method against other popular multi-label learning algorithms.

[1] Items in a recommendation task constitute as the label set in a multi-label learning problem setting. Top-K recommendations for every user is obtained by picking the top K label estimates per user.

[2] Shopify App Store: http://apps.shopify.com

[3] We performed the temporal split such that all user-item interactions in the training set were interactions that happened before interactions contained in the test set.

[4] https://imbalanced-learn.readthedocs.io

Table 1: Summary of experiments

models	**# nnz**	**density (%)**	**speed (s)**	**p@1 (%)**	**p@5 (%)**	**p@10 (%)**
BR	2,594,150	5	216.4	17.9 $\pm$1.0	18.2 $\pm$1.4	20.2 $\pm$1.2
DiagT	273,942,306	0.05	**172.3**	**21.7** $\pm$1.0	**21.4** $\pm$1.6	**23.6** $\pm$1.2
DiagT-hb0.9	259,377,381	0.056	**175.6**	**20.3** $\pm$0.7	**20.3** $\pm$0.6	**22.4** $\pm$0.6
DiagT-rus-hb0.9	259,399,448	0.056	**40.1**	17.0 $\pm$1.0	17.7 $\pm$0.9	19.8 $\pm$0.8
DiagT-rus	256,820,916	0.197	**13.86**	**22.6** $\pm$2.3	**21.2** $\pm$1.05	**23.3** $\pm$0.8

Bold entries are DiagT-based results that are better than BR.

Abbreviations: BR - binary relevance, DiagT - our proposed method, hb - hashing bucket ratio, rus - random under sampling, nnz - of nonzeros. Precision metrics are calculated using the three-fold evaluation with a 95% confidence interval.

References

Richard O Duda and Peter E Hart. 1973. Pattern classification and scene analysis. *A Wiley-Interscience Publication, New York: Wiley, 1973* (1973).

Sariel Har-Peled, Dan Roth, and Dav Zimak. 2003. Constraint classification for multiclass classification and ranking. In *Advances in neural information processing systems*. 809–816.

John Langford, Lihong Li, and Alex Strehl. 2007. Vowpal wabbit online learning project. (2007).

Nils J Nilsson. 1965. *Learning machines: foundations of trainable pattern-classifying systems*. McGraw-Hill.

Grigorios Tsoumakas and Ioannis Katakis. 2007. Multi-label classification: An overview. *International Journal of Data Warehousing and Mining (IJDWM)* 3, 3 (2007), 1–13.

Grigorios Tsoumakas, Ioannis Katakis, and Ioannis Vlahavas. 2009. Mining multi-label data. In *Data mining and knowledge discovery handbook*. Springer, 667–685.