Representation Learning for Sparse, High Dimensional Multi-Label Classification

Ryan Kiros[1], Axel J. Soto[2], Evangelos Milios[2], and Vlado Keselj[2]

[1] Department of Computing Science, University of Alberta
Edmonton, Canada - rkiros@ualberta.ca
[2] Faculty of Computer Science, Dalhousie University
Halifax, Canada - {soto,eem,vlado}@cs.dal.ca

Abstract. In this article we describe the approach we applied for the JRS 2012 Data Mining Competition. The task of the competition was the multi-labelled classification of biomedical documents. Our method is motivated by recent work in the machine learning and computer vision communities that highlights the usefulness of feature learning for classification tasks. Our approach uses orthogonal matching persuit to learn a dictionary from PCA-transformed features. Binary relevance with logistic regression is applied to the encoded representations, leading to a fifth place performance in the competition. In order to show the suitability of our approach outside the competition task we also report a state-of-the-art classification performance on the multi-label ASRS dataset.

Keywords: Multi-label classification, feature learning, text mining

1 Introduction

Different representations for text corpora have been extensively studied, being TF-IDF and Okapi BM25 two of the most common ways of representing data [9]. The choice of document representation has a major incidence in the performance for tasks such as document classification or retrieval. Particularly, multi-labelled document classification is a research problem that received much less attention in the literature than the single-labelled counterpart. Yet multi-labelled classification is in many cases a more natural approach for document classification tasks [13].

The machine learning community, specially in the area of computer vision, has witnessed the importance of learning feature representations as an alternative to manually configuring the best data representation to feed a prediction method. Learned representations coupled with simple classification methods usually tend to have similar classification accuracy and even overcome other more complex classification methods [3, 4, 14].

The JRS 2012 Data Mining Competition represented a great opportunity to benchmark and evaluate our hypothesis of whether a learned representation of the data combined with a standard classification approach could have a competitive performance against other approaches for multi-label text classification.

The learned representation can be thought as a means to both reduce and enhance the original data representation to facilitate learning of the classifier that is used *a posteriori*.

The present paper reports the details of the method that got our highest preliminary score on the competition. Our proposed method is based on a two step feature learning procedure that was motivated by recent work in object recognition [4] and adapted to be used for sparse, high dimensional data. We first embed the inputs using Principal Component Analysis (PCA) and then learn a sparse dictionary that we concatenate with the PCA embeddings for classification. Our final predictions were obtained using binary relevance with a linear logistic regression classifier. In order to further illustrate the benefits of our approach we also show the results on the multi-label SIAM 2007 competition dataset for text mining.

2 Method

Our approach can be divided in three main parts, namely: preprocessing, representation learning and classification.

2.1 Data Preprocessing

Let $X = \{x^{(1)}, \ldots, x^{(m)}\}$ be the set of m training documents where the j-th feature of $x^{(i)}$ is $x_j^{(i)}$, $j = 1 \ldots n$. We first process the data by normalizing each $x^{(i)}$ to $[0, 1]$, where $i = 1 \ldots m$. This is done by dividing each feature by its range (i.e. the difference between the maximum and the minimum value). The data is then rescaled to $[-1, 1]$. Finally, we use a 'regularized' mean centering and variance normalization for each feature j:

$$x_j^{(i)} = \frac{x_j^{(i)} - \mu_j}{\sqrt{\sigma_j^2 + \epsilon}} \tag{1}$$

where the feature means (μ_j) and standard deviations (σ_j) are preserved for use with the test set. For our experiments we use $\epsilon = 0.01$.

2.2 Unsupervised Representation Learning

Given the pre-processed data, we first apply PCA and extract the first k principal components. This calculation is the most expensive procedure for our method, both memory and time wise, due to the size of the covariance matrix. Let $S = \{s^{(1)}, \ldots, s^{(m)}\}$ represent the k-dimensional outputs. We perform one additional processing step of S by centering and normalizing the variances of each individual datapoint $s^{(i)}$. We use the same 'regularized' normalization as in Equation 1 but applied to documents as opposed to features. This step, when applied to images, can be seen as a form of brightness and contrast normalization. Although less motivated for our PCA-transformed data, we found it to be a useful addition to the pipeline.

After this normalization step, the data are now ready to be used for constructing a dictionary (also called prototypes or codebook). Dictionaries have been traditionally used in the area of signal processing for data compression. Common examples to this kind of techniques are vector quantization or k-means [5]. We use orthogonal matching pursuit (OMP) [11], which aims to solve the following optimization problem:

$$\begin{aligned} \underset{D, \hat{s}^{(i)}}{\text{minimize}} \quad & \sum_{i=1}^{m} ||D\hat{s}^{(i)} - s^{(i)}||_2^2 \\ \text{subject to} \quad & ||D^{(j)}||_2^2 = 1 \\ & ||\hat{s}^{(i)}||_0 \leq q \end{aligned} \tag{2}$$

where $D \in \mathbb{R}^{k,d}$ is the dictionary to be learned. The first constraint enforces that the dictionary elements remain normalized (to avoid degeneracy) while the second constraint enforces sparsity, allowing at most q elements of $\hat{s}^{(i)}$ to be nonzero. Our objective is minimized using alternation: first fixing D and minimizing $\hat{s}^{(i)}$, then fixing $\hat{s}^{(i)}$ and minimizing D. We set $q = 1$ for all of our experiments. We chose to use OMP over other methods due to the speed of training, taking only a few minutes to train the dictionary.

As opposed to using the proper representations $\hat{s}^{(i)}$, Coates et al. [4] showed that in the presence of enough labelled data a simple soft activation may perform equally, if not better when applied in an object recognition setting. We follow this approach by encoding that data with a simple rectification unit:

$$\check{s}^{(i)} = \max\{D^T s^{(i)}, 0\}, \tag{3}$$

where the max function is applied componentwise. Finally, we apply another form of 'contrast normalization' over individual datapoints as was previously done with the PCA transformed data. Given the encoded data $\check{S}$, we concatenate the learned features with the PCA embeddings and we then normalize the data once more using Equation 1. These datapoints, $Z = \{z^{(1)}, \ldots, z^{(m)}\}$, are now ready for training our classifier.

2.3 Classification

To train our model, we used binary relevance with logistic regression incorporating weight decay for regularization. This means that for each single label l we perform the following optimization:

$$\underset{\theta^{(l)}}{\text{minimize}} \quad -\frac{1}{m}\big[\sum_{i=1}^{m} y^{(i)}\text{log } h_{\theta^{(l)}}(z^{(i)}) + (1 - y^{(i)})\text{log}(1 - h_{\theta^{(l)}}(z^{(i)}))\big] + \lambda\|\theta^{(l)}\|_F^2 \tag{4}$$

where $h_\theta(z) = 1/(1+\exp(-\theta^T z))$, $y^{(i)}$ is the label of the i-th document (for class l) and $\|\cdot\|_F^2$ stands for the square of the Frobenius norm. Optimization was done using L-BFGS [8] with minFunc[3], often converging in less than 30 iterations per class.

We also experimented using a multi-label SVM with a linear kernel combined with an adaptive thresholding scheme used to maximize F-measure. We found that although the latter was faster to train, it was less stable for parameter selection and often led to an imbalance of either precision or recall.

3 Results

3.1 JRS 2012 data set

The competition dataset comprises of 20000 documents out of which half of them were held out for testing (whose labels were not disclosed during the competition time). Each document is represented by a 25640-dimensional vector, where each component represents the association strength to a Medical Subject Heading (MeSH) term.

For our best preliminary result, we used a total of $k = 1600$ principal components and a dictionary size of $d = 3200$. This lead to a final feature vector of size 4800. We used the same parameter λ for each of the classifiers. To obtain λ, we split the training set into 8000 samples for training and 2000 for validation. We then performed a grid search across powers of 2, followed by a more fine-grained search, leading to a chosen parameter of $\lambda = 0.5$. We obtained an F-measure on the validation set of 0.522 and with the same model a final preliminary score of 0.523. We also attempted to find a separate λ_j for each class j, leading to a validation score of 0.527. Unfortunately, this did not generalize to the preliminary result. Our final performance on the test data was **0.53**, resulting in a 5th place finish.

We used the same pipeline as described above throughout the whole competition with all of our improvements coming from modifying the choice of dimensionality reduction, number of bases and normalization steps. Combining the PCA embeddings with the encodings performs much better than either one or the other alone. We found that performance consistently improved by using more bases and principal components.

To further motivate our basis learning approach, we trained the described multi-label SVM on the normalized input data, receiving a validation score of 0.5. When combined with the encodings learned from the PCA embeddings, this increased the score to 0.52. We opted out from further pursuing this approach in the competition due to the high dimensionality of the final feature vectors.

Finally, we experimented with the usefulness of all of our normalization steps and found the most important being the initial regularized normalization as well as the same normalization before training the classifier. In this sense, all of the 'contrast normalization' steps can be safely removed and still result in good performance.

[3] `http://www.di.ens.fr/~mschmidt/Software/minFunc.html`

3.2 Additional results: SIAM 2007 data set

In the domain of text mining, a common approach to representing documents is through a bag-of-words (or the more general n-gram) representation. Here, a weight is given for each document-term pair, such as frequency of TF-IDF. Such a representation is often of high dimension with only a few non-zero entries corresponding to the n-grams which are present in the corresponding document. Thus we further evaluate our proposed approach on a text classification task to show that our approach can be utilized in other domains aside from the biomedical classification task from this competition.

We consider the ASRS (Aviation Safety Reporting System) dataset, which was used for the SIAM 2007 competition [10]. ASRS is a collection of roughly 21000 training documents and 7000 testing documents whose labels are 22 dimensional binary vectors indicating the presence of one or more aircraft security issues. Sample classes include weather, fuel emergencies, passenger disruptions, pilot attentiveness and runway obstructions to name a few. The SIAM competition data was sampled from the publicly available ASRS database maintained by NASA[4]. Below is an example of a typical document from the dataset:

> “UPON TOUCHDOWN AT NIGHT ON runway _ AT BID THE right land GEAR STRUCK A DEER ON THE runway.I DID NOT SEE THE DEER.THE result DAMAGE WAS THE remove OF THE right GEAR AND THE aircraft settle ON right WING skid OFF THE right SIDE OF runway.”

To obtain the initial representation of a document, we first pre-process the data by lowering case and performing stopword removal. We then obtain the 5000 most frequent words for which a document-term matrix of unnormalized TF-IDF values is constructed. The IDF values are obtained from the training set and applied in conjunction with the frequencies calculated on each test point. Model selection was performed in the same way as was done on the JRS competition data using the same number of principal components (1600) and bases (3200).

Method	Precision	Recall	F-Measure	Error
SIAM Winner (Score) [6]	61.53	62.37	61.95	6.80
SIAM Winner (F-Measure) [6]	53.30	**78.55**	63.51	8.01
Normalized Baseline (this paper)	63.15	70.30	66.40	6.31
PCA + OMP (this paper)	**64.25**	71.68	**67.76**	**6.05**

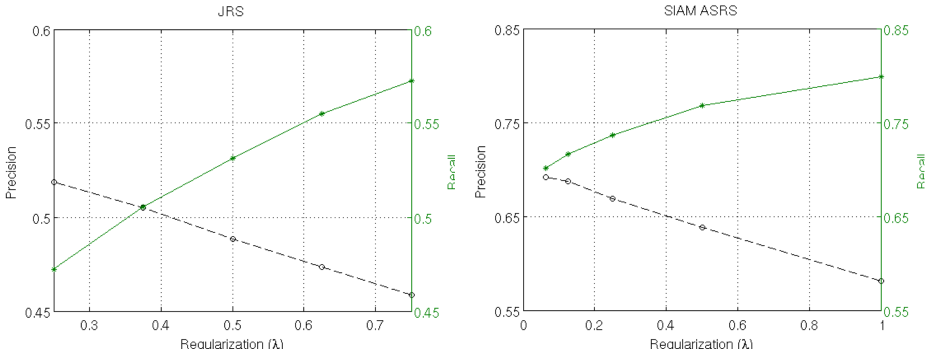
Table 1: A comparison of (micro-averaged) classification performance on the SIAM ASRS dataset. The first two methods are the results of the competition winner, the first being where the competition score is maximized while the second being where F-measure is maximized.

[4] `http://asrs.arc.nasa.gov/`

To test the effectiveness of our approach, we compared our results to the performance of the first place finisher of the SIAM 2007 competition [6]. Table 1 shows our result in comparison to the winner's best approach with respect to the competition evaluation as well as the approach that maximized F-measure. We also included a baseline showing the result when binary relevance is applied directly to the normalized document-term matrix. Our result outperforms the competition winner yielding state-of-the-art results on this dataset[5]. Surprisingly, the baseline without any representation learning is able to also outperform the existing result. We attribute this to the regularized normalization of the features. In particular, result are over 2% worse across all metrics when no regularization ($\epsilon \approx 0$) is used.

3.3 Adapting Precision and Recall

One further observation that was made on both datasets was that the precision and recall can be directly tuned through the logistic regression regularization parameter (λ). More specifically, increasing λ leads to an increase of recall but decrease of precision and equivalently in the opposite when decreasing λ. Figure 1 illustrates this effect on the validation sets of both the JRS competition and ASRS datasets. Moreover, the parameter that led to the best model for the competition was that which had slightly higher recall on the validation set.



(a) JRS (b) SIAM ASRS

Fig. 1: Graphs illustrating the effect of the logistic regression regularization on precision in black and recall in red (best seen in color).

We note that it is often the case that specific applications may be more interested in maximizing either precision or recall. As a related example to aircraft

[5] We note that other proposed methods have been evaluated from taking random samples from the full ASRS database.

security, it is much more important for text retrieval systems to have high recall rather than high precision so that all security incidents can be retrieved, even if this may lead to a high false positive rate. Being able to control this effect gives our approach potential use in these types of domains.

4 Conclusions

In this work we described an unsupervised feature learning methodology in the context of multi-label document classification. Our method obtained the 5th position in the JRS 2012 Data Mining Competition. We also showed the results of applying our method on another challenging and multi-labelled dataset. In this last data set we obtained a state-of-the-art performance, hence indicating the suitability of our approach for multi-label text classification.

There are several possible avenues for future research. One such approach is testing the effectiveness of our method in a semi-supervised setting. More specifically, a dictionary could be trained using both sets of labelled and unlabelled data. Related to this is the use of self-taught learning [12] also known as transfer learning from unlabelled data. One could train a large dictionary on a corpus that have come from a different distribution than the target dataset intended for classification. Such an approach may be effective in tasks where only small amounts of labelled data exist. Our approach could also be used for domain adaptation in situations where two or more datasets have the same label distribution but the target dataset of interest is unlabelled. These situations often occur in sentiment classification tasks when labels may correspond to binary positive and negative opinions or 5-star ratings. Finally, there has been much work on learning deep representations [1, 7]. The output encoding features could be used as input to another layer of dictionary training with the first layer bases frozen. Such greedy layer-by-layer training could be used as many times as desired. It is still an open problem what the best approaches to training such architectures are [2].

Throughout our procedure almost all the focus was made on the representation learning and normalization phases, with little effort put towards classification. Binary relevance is flawed in that it does not take into account correlations between labels. It is worth exploring whether improvements can be made by adapting more sophisticated multi-label classification approaches. It also remains as an interesting research question whether the knowledge of a domain ontology, such as the disclosure of the MeSH ontology for the competition dataset, can be used to further improved the classification accuracy.

Acknowledgments

Authors thank The Boeing Company and NSERC for funding of this work.

References

1. Bengio, Y., Lamblin, P., Popovici, D., Larochelle, H.: Greedy layer-wise training of deep networks. Advances in neural information processing systems 19, 153 (2007)

2. Bengio, Y.: Learning deep architectures for AI. Foundations and Trends in Machine Learning 2(1), 1–127 (2009)
3. Boureau, Y.L., Bach, F., LeCun, Y., Ponce, J.: Learning mid-level features for recognition. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 2559–2556 (2010)
4. Coates, A., Ng, A.Y.: The importance of encoding versus training with sparse coding and vector quantization. In: Getoor, L., Scheffer, T. (eds.) International Conference on Machine Learning. pp. 921–928. Omnipress (2011)
5. Gersho, A., Gray, R.M.: Vector quantization and signal compression. Kluwer Academic Publishers, Norwell, MA, USA (1991)
6. Goutte, C.: A probabilistic model for fast and confident categorization of textual documents. In: Castellanos, M.W.B., Malu (eds.) Survey of Text Mining II, vol. 4, pp. 187–202. Springer (2008)
7. Hinton, G., Osindero, S., Teh, Y.: A fast learning algorithm for deep belief nets. Neural Computation 18(7), 1527–1554 (2006)
8. Liu, D.C., Nocedal, J.: On the limited memory method for large scale optimization. Mathematical Programming 45(3), 503–528 (1989)
9. Manning, C.D., Raghavan, P., Schütze, H.: An Introduction to Information Retrieval. Cambridge University Press (2008)
10. NASA: SIAM 2007 – Aviation Safety Reporting System (ASRS) Challenge Dataset (2007), `http://web.eecs.utk.edu/events/tmw07/`
11. Pati, Y., Rezaiifar, R., Krishnaprasad, P.: Orthogonal matching pursuit : recursive function approximation with application to wavelet decomposition. In: Asilomar Conference on Signals, Systems and Computers. pp. 40–44. Pacific Grove, CA (1993)
12. Raina, R., Battle, A., Lee, H., Packer, B., Ng, A.Y.: Self-taught learning, pp. 759–766. ACM Press (2007)
13. Tsoumakas, G., Katakis, I.: Multi-label classification: An overview. International Journal of Data Warehousing & Mining 3(3), 1–13 (2007)
14. Yang, J., Yu, K., Gong, Y., Huang, T.: Linear spatial pyramid matching using sparse coding for image classification. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 1794–1801 (2009)