Knowledge Extraction with Non-Negative Matrix Factorization for Text Classification

Catarina Silva[1,2] and Bernardete Ribeiro[2]

[1] School of Technology and Management of the Polytechnic Institute of Leiria, Morro do Lena - Alto do Vieiro, Portugal, P-2411-901 Leiria, Portugal
[2] Department of Informatics Engineering, Center for Informatics and Systems (CISUC), University of Coimbra, Polo II, P-3030-290 Coimbra, Portugal
catarina@dei.uc.pt, bribeiro@dei.uc.pt

Abstract. Text classification has received increasing interest over the past decades for its wide range of applications driven by the ubiquity of textual information. The high dimensionality of those applications led to pervasive use of dimensionality reduction methods, often black-box feature extraction non-linear techniques.

We show how Non-Negative Matrix Factorization (NMF), an algorithm able to learn a parts-based representation of data by imposing non-negativity constraints, can be used to represent and extract knowledge from a text classification problem. The resulting reduced set of features is tested with kernel-based machines on Reuters-21578 benchmark showing the method's performance competitiveness.

1 Introduction

The non-negative matrix factorization (NMF) [1] is an algorithm which is able to learn a parts-based representation of data by imposing non-negativity constraints that allow only non-subtractive combinations. Similarly to principal component analysis (PCA) [2] that is based on finding a new representation (eigenspace) of the original data, NMF is also a projection method since the original data is projected onto the new space. In contrast to PCA, the projected coefficients that are obtained using the NMF method are only positive. Furthermore, some of the basis components for PCA are distorted versions of the original data. The NMF basis is radically different: it is possible to extract localized features that correspond better with intuitive notions of the parts of the original data [1]. This correspondence is in fact knowledge about the underlying problem, which can be used not only to guide learning procedures, but also to provide interpretability to users.

Although the concept is not new, since it has been investigated in linear algebra [3] where it was called positive matrix factorization (PMF), the last ten years have witnessed a large amount of research on NMF since the seminal work [1,4] was presented. Several variants of NMF have been proposed by researchers. Hoyer [5] proposed a method of non-negative sparse coding which minimizes a new cost function containing a positive regularization parameter. NMF has also been applied to many areas such as bioinformatics [6,7,8], molecular pattern discovery [9], chemometrics [10], pattern recognition [1] and risk analysis [11].

E. Corchado and H. Yin (Eds.): IDEAL 2009, LNCS 5788, pp. 300–308, 2009.

In text classification, the vectors represent or identify semantic features, i.e., a set of words denoting a particular concept or topic. If a document is viewed as a combination of basis vectors, then it can be categorized as belonging to the topic represented by its principal vector. Thus, NMF can be used to organize text collections into partitional structures or clusters directly derived from the nonnegative factors. Therefore in the text mining and classification area, NMF has been essentially applied in unsupervised settings.

In [12] a methodology for automatically identifying and clustering semantic features or topics in a heterogeneous text collection is presented. A hybrid NMF algorithm is proposed that can be used to construct a parts-based representation of the text data, in which the localization of the parts or features can be regularized to create a balance between computational cost and accuracy. In [13] a document clustering method is proposed based on the non-negative factorization of the term-document matrix of the given document corpus. The authors have demonstrated that NMF outperforms methods such as singular value decomposition and is comparable to graph partitioning methods that are widely used in clustering text documents. In [14] a version of NMF with an extended cost function and with smoothness constraints is proposed and applied to email surveillance again in an unsupervised setting.

In our work, we propose to use NMF in a supervised setting. We use NMF as preprocessing dimensionality reduction step, obtaining a set of interpretable extracted semantic features that can next be used by a standard learning algorithm, namely support vector machines. This strategy intends to retrieve the best of both approaches, namely provide the user with an interpretation of semantic features and maintain a competitive edge on classification performance.

This paper is organized as follows. In Section 2 a brief review of non-negative matrix factorization is presented. The proposed approach for knowledge extraction for text classification is discussed in Section 3. Experimental setup is introduced in Section 4 and results are presented and discussed in Section 5. Finally in Section 6 we conclude the paper along with further lines of future research.

2 Background on Non-negative Matrix Factorization

Non-negative matrix factorization (NMF) is an algorithm to obtain a linear representation of data under non-negativity constraints. These constraints lead to a parts-based representation because they allow additive, not subtractive, combinations of the original data [1]. The basic idea is below.

First, represent a corpus as an $n \times m$ matrix V, where each column corresponding to an initial document, includes n non-negative elements characterizing the term values and m is the number of training documents. Then, we can find two new non-negative matrices (W and H) to approximate the original matrix:

$$V_{ij} \simeq (WH)_{ij} = \sum_{a=1}^{r} W_{ia} H_{aj}, W \in \mathbb{R}^{n \times r}, H \in \mathbb{R}^{r \times m}, \quad (1)$$

where matrix W consists of r non-negative basis vectors and the rank r is usually chosen to be as small as possible for dimensionality reduction, while column

vectors of H represent the weights in the approximation of the corresponding columns in V using the bases from W. From the original definition, we know, in contrast to the PCA approach, no subtractions can occur in the above NMF procedure, so the non-negativity constraints are compatible with the intuitive idea of combining parts to form a whole document.

The constrained minimization problem can be put as minimizing the difference between V and WH by [15]:

$$\min_{W,H} f(W,H) \equiv \frac{1}{2}\sum_{i}^{n}\sum_{j}^{m}(V_{ij} - (WH)_{ij})^2 \quad \text{subject to } W_{ia} \geq 0, H_{bj} \geq 0 \ \forall i,j \tag{2}$$

The most popular approach [1] to solve this problem seeks to iteratively update the factorization based on a given objective function. This approach is similar to that used in Expectation-Maximization (EM) algorithms and is known as the multiplicative algorithm given below:

Algorithm 1. NMF algorithm [1]

Input: $V \in \mathbb{R}^{n\times m}$ and $r = rank$
Step 1. Randomize W and H with positive numbers in [0, 1].
Select a cost function to be minimized.
Step 2. With W fixed, update H, then update W for the updated H.
Iterate until the process converges.
Return: $W \in \mathbb{R}^{n\times r}$ and $H \in \mathbb{R}^{r\times m}$.

In the above algorithm, the cost function is either $C_1(V,WH) = \| V - WH \|_F^2$ (where $\| \ . \ \|_F$ is the Frobenius norm) or the generalized Kullback-Leibler divergence $C_2(V,WH) = \sum_{i,j}(V_{ij} log V_{ij}/(WH)_{ij} - V_{ij} + (WH)_{ij})$. When cost function C_1 is used, the formulæ for updating H and W are:

$$W_{ia} := W_{ia}\frac{(VH^T)_{ia}}{(WHH^T)_{ia}}, \tag{3}$$

$$H_{bj} := H_{bj}\frac{(W^TV)_{bj}}{(W^TWV)_{bj}}, \tag{4}$$

whereas if cost function C_2 is used, the updating formulæ for H and W are:

$$W_{ia} := W_{ia}\sum_{j}\frac{V_{ij}}{(WH)_{ij}}H_{bj}, \tag{5}$$

$$W_{ia} := \frac{W_{ia}}{\sum_j W_{ja}}, \tag{6}$$

$$H_{bj} := H_{bj}\sum_{i} W_{ia}\frac{V_{ij}}{(WH)_{ij}}. \tag{7}$$

The above equations are obtained by minimization using non-linear programming methods such as the gradient descent [15]. Several bound-constrained optimization techniques have been used [5,18] to solve the problem. Since factors W and H are nonconvex only local minimum is guaranteed to be obtained [4]. An implementation in MatLab is provided in [15] where a systematical experimentation of CPU run time is exploited in an image benchmark data. However, the NMF algorithms based on gradient descent method exhibit slow convergence.

A simpler yet efficient way is to use multiplicative update algorithms. A convergence proof of Lee-Seungs multiplicative update algorithm for nonnegative matrix factorization is given in [17]. Alternating Least Square (ALS) is considered an alternative to the traditional multiplicative update algorithms and gradient descent (additive) algorithms. In this algorithm, a least square step is followed by another least square step in an alternating manner. The NMF/ALS algorithm [14] sets the negative components in the unconstrained least squares solution to zeros. According to the authors while the optimization problem is not convex in both W and H, it is convex in either W or H. Hence, given one matrix it is possible to find the other using the least squares computation. A simple implementation of the algorithm is proposed in [14]. Although this method may solve the subproblems faster, its convergence behavior is problematic.

There are other known decomposition techniques but only NMF implies that the nonnegative constraints are obeyed which, with texts, is specially important since any term that composes a text has a non-negative value. The fact that numerical methods used for NMF are able to extract underlying patterns or features as basis vectors can be used for identification and classification [19].

3 Knowledge Extraction for Text Classification

Text classification can be defined as the automatic assignment of semantic categories to natural language texts or documents. Documents are represented by vectors of numeric values, with one value for each word that appears in any training document, making it a very high dimensional representation. Moreover, the number of available text documents is increasing exponentially [20], lending a tremendous importance to the dimensionality of the problem.

Common feature selection and dimension reduction methods used in this work to reduce the number of features (words or terms) are stopword removal, Porter stemming and removing less frequent words.

Beyond these general dimension reduction methods, we propose to apply nonnegative matrix factorization (NMF) as knowledge extraction method, giving text classification users a reliable insight into the dimensionality reduction step.

3.1 Semantic Features

NMF, as described in the last section, can be applied to supervised learning settings as a pre-processing feature extraction technique. In this case, the new

set of extracted semantic features consists of the matrix H, that represents each of the m training documents with r new features, instead of the initial n terms.

On the other hand, matrix W incorporates the information of how the r new semantic features are constructed from the initial n terms. Therefore, it can be used to find an interpretation of the NMF feature extraction. Given that the number of initial features, as well as the number of training documents, are usually very high, some form of selection of more relevant term contributions has to be found. Each semantic feature we extract is represented by the terms with highest frequency in that feature. This procedure will allow for a perception of which initial terms were considered in the extraction of the semantic features.

3.2 Learning Approach

Support vector machines (SVM) are state-of-the-art learning algorithms for text classification [20]. SVM are a learning method introduced by Vapnik [21] based on Statistical Learning Theory and Structural Minimization Principle. When using SVM for classification, the basic idea is to find the optimal separating hyperplane between the positive and negative examples. The optimal hyperplane is defined as the one giving the maximum margin between the training examples that are closest to it. Support vectors are the examples that lie closest to the separating hyperplane. Once this hyperplane is found, new examples can be classified simply by determining on which side of the hyperplane they fall.

To test the effectiveness of the semantic feature extraction classification performance we used SVM in the learning step and for baseline comparison. Fig. 1 represents the proposed approach. The initial corpus representation is reduced by extracting features with NMF and the interpretation of the semantic features is retrieved to the user. The reduced representation is then used to infer a SVM model and results are also presented to the user.

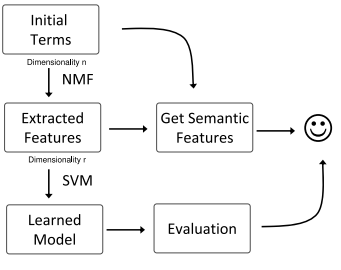

Fig. 1. Proposed approach: NMF-SVM

4 Experimental Setup

4.1 Benchmark: Reuters-21578

Reuters-21578 is a financial corpus with news articles averaging 200 words each[1]. In this corpus there are 21578 classified stories into 118 possible categories.

[1] Reuters-21578 is publicly available at http://kdd.ics.uci.edu/databases/reuters21578/reuters21578.html

Table 1. Contingency table for binary classification

	Class Positive	Class Negative
Assigned Positive	a	b
	(True Positives)	(False Positives)
Assigned Negative	c	d
	(False Negatives)	(True Negatives)

However, the corpus comes with a set of predefined training and testing splits. The commonly used ModApte split [22] filters out duplicate articles and those without a labeled topic, and then uses earlier articles as the training set (9603 items) and later articles as the testing set (3299 items).

4.2 Classification Metrics

The performance will be evaluated using the testing sets defined for each category, with several metrics to determine the learning ability. In order to assess a binary decision task, we first define a contingency matrix representing the possible outcomes of the classification, as shown in Table 1. Several measures have been defined based on this contingency table, such as, Error Rate ($\frac{b+c}{a+b+c+d}$), Recall ($\frac{a}{a+c}$) and Precision ($\frac{a}{a+b}$). Measures that combine recall and precision have been defined, such as, the van Rijsbergen's F_β measure [23], which combines recall and precision in a single score, $F_\beta = \frac{(\beta^2+1)P \times R}{\beta^2 P+R}$ and is one of the best suited measures for text classification [24]. Thus results reported in this paper are macro-averaged F1 values.

5 Results

This section presents and analyzes both semantic and classification results. Interpretability, given by the analysis of extracted feature representation is always subjective. In fact, each user in each application is in the best place to evaluate it. In Reuters-21578, categories are highly correlated, as can be inferred just by their names. Therefore semantic differences will be necessarily scarce.

To capture the semantic information in each extracted feature, we first applied stopword removal, Porter stemming and removing words that appeared in less than 190 documents, resulting in an initial dimensionality of $n = 497$. Then, the proposed NMF-SVM approach was carried out with a rank $r = 200$ of embedded semantic features. Thus we obtain the W matrix ($n \times r$) resulting from the NMF representation step, i.e. 497 initial features $\times$ 200 semantic features. As the semantic features are very high-dimensional vectors, each one is represented by a list of the words (initial features) with highest frequency in that specific semantic feature, as shown in Fig. 2. For Reuters-21578 five words were heuristically considered sufficient to provide some interpretability (other thresholds can be found for other data sets). We can glean a coherence from these semantic features, despite the inherent difficulty in distinguishing some of the Reuters-21578 classes:

Table 2. F_1 performances for Reuters-21578

Category	SVM	NMF-SVM
Earn	97.06%	97.20% ± 0.22%
Acq	93.78%	93.65% ± 0.49%
Money-fx	64.57%	73.66% ± 3.64%
Grain	71.63%	75.40% ± 1.76%
Crude	74.83%	79.89% ± 1.01%
Trade	70.53%	73.16% ± 1.31%
Interest	69.01%	72.55% ± 1.91%
Ship	31.37%	62.00% ± 10.03%
Wheat	70.27%	68.11% ± 6.37%
Corn	52.17%	58.77% ± 5.25%
Average	69.52%	75.44% ± 1.31%

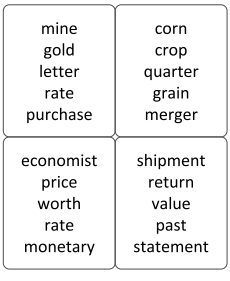

Fig. 2. Examples of semantic features derived with NMF from Reuters-21578. Each feature is represented by the five words with highest frequency.

the top left is related to trade, the top right to wheat/grain/corn, the bottom left to interest/money-fx and the bottom right to ship.

Table 2 presents the F_1 performances for the Reuters-21578 categories with and without NMF feature extraction. The SVM software used was libsvm interface to Matlab[2]. NMF includes a random initialization of the W matrix, thus the procedure was carried out in ten runs to achieve statistical significance (mean and standard deviation values are presented). Concerning SVM settings, both results from baseline (SVM) and our approach (NMF+SVM) were achieved with a linear kernel, the C parameter (balance between error and generalization) set to 100 and $w1$ set to two (errors on positive examples, i.e. false negatives, are twice as important as false positives). The baseline of comparison was inferred using the initial already reduced representation of the corpus, with 497 terms, and the NMF+SVM strategy uses only the 200 semantic features extracted with NMF. Comparing both approaches, we can conclude that the new proposed tech-

[2] http://www.csie.ntu.edu.tw/∼cjlin/libsvm/#matlab

nique presents an overall improvement of around 6% within acceptable standard deviations.

6 Conclusions

In this paper we show how non-negative matrix factorization (NMF) can be used as a knowledge extraction technique for text classification, improving both interpretability and performances.

NMF has the property of intuitive parts-based representation of the original features. This unique ability can be extremely helpful to aid users gain knowledge about text classification systems. Usually complex text classification tasks, i.e. tasks that do not fully rely on words on users' queries, but rather use more elaborate learning mechanisms, appear suspicious to users. If researchers can provide interpretable knowledge, still using cutting-edge learning, more users will rely on them, expanding even more the field of applications of text classification.

Further work along these lines will include a more elaborate model of interpretability and more effort on the learning of class-specific semantic features.

References

1. Lee, D.D., Seung, H.S.: Learning the Parts of Objects by Non-negative Matrix Factorization. Nature 401, 788–791 (1999)
2. Jolliffe, I.T.: Principal Component Analysis. Springer, New York (1986)
3. Paatero, P., Tapper, U.: Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. Environmetrics 5, 111–126 (1994)
4. Lee, D.D., Seung, H.S.: Algorithms for non-negative matrix factorization. In: Advances in Neural Information Processing 13 (Proc. NIPS 2000). MIT Press, Cambridge (2000)
5. Hoyer, P.O.: Non-negative matrix factorization with sparseness constraints. Journal of Machine Learning Research 5, 1457–1469 (2004)
6. Zhang, Z.Y., Zhang, X.S.: Two improvements of NMF used for tumor clustering. In: 1st Int. Symposium on Optimization and Systems Biology, pp. 242–249 (2007)
7. Carmona-Saez, P., Pascual-Marqui, R.D., Tirado, F., Carazo, J., Pascual-Montano, A.: Biclustering of gene expression data by non-smooth non-negative matrix factorization. BMC Bioinf. (2006)
8. Fogel, P., Young, S., Hawkins, D., Ledirac, N.: Inferential, robust non-negative matrix factorization analysis of microarray data. BMC Bioinf. 23(1) (2007)
9. Brunet, J.P., Tamayo, P., Golub, T.R., Mesirov, J.P.: Metagenes and molecular pattern discovery using matrix factorization. National Academy of Science 101 (2004)
10. Guimet, F., Boque, R., Ferre, J.: Application of non-negative matrix factorization combined with fishers linear discriminant analysis for classification of olive oil excitation emission fluorescence spectra. Chemometrics and Intelligent Laboratory Systems 81, 94–106 (2006)
11. Ribeiro, B., Silva, C., Vieira, A., Neves, J.: Extracting Discriminative Features Using Non-Negative Matrix Factorization in Financial Distress Data. In: Kolehmainen, V., Toivanen, P., Beliczynski, B. (eds.) ICANNGA 2009. LNCS, vol. 5495. Springer, Heidelberg (2009)

12. Shahnaz, F., Berry, M., Pauca, V., Plemmons, R.: Document clustering using nonnegative matrix factorization. Information Processing and Management: an International Journal 42(2), 373–386 (2006)
13. Xu, W., Liu, X., Gong, Y.: Document clustering based on non-negative matrix factorization. In: ACM SIGIR 2003, pp. 267–273 (2003)
14. Berry, M., Browne, M., Langville, A., Pauca, V., Plemmons, R.: Algorithms and applications for approximate nonnegative matrix factorization. Computational Statistics & Data Analysis 52(1), 155–173 (2007)
15. Lin, C.J.: Projected gradient methods for nonnegative matrix factorization. Neural Computation 19(10), 2756–2779 (2007)
16. Hofmann, T.: Probabilistic latent semantic indexing. In: Twenty-Second Annual International SIGIR Conference on Research and Development in Information Retrieval (1999)
17. Lin, C.J.: On the convergence of multiplicative update algorithms for nonnegative matrix factorization. IEEE Tran. on Neural Networks 6(18), 1589–1596 (2007)
18. Chu, M., Plemmons, R.J.: Nonnegative matrix factorization and applications. IMAGE 34, 1–25 (2005)
19. Almeida, A., Júdice, J., Fernandes, L., Patrício, J.: On the computation of a nonnegative matrix factorization and its application in telecommunications. In: 7th Conference on Telecommunications (2009)
20. Sebastiani, F.: Classification of Text, Automatic. In: Brown, K. (ed.) The Encyclopedia of Language and Linguistics, 2nd edn., vol. 14. Elsevier, Amsterdam (2006)
21. Vapnik, V.: The Nature of Statistical Learning Theory, 2nd edn. Springer, Heidelberg (1999)
22. Apté, C., Damerau, F., Weiss, S.: Automated Learning of Decision Rules for Text Categorization. ACM Trans. for Information Sys. 12, 233–251 (1994)
23. van Rijsbergen, C.: Information Retrieval. Butterworths (1979)
24. Ruiz, M., Srinivasan, P.: Automatic Text Categorization and Its Application to Text Retrieval. IEEE Tran. Know. Data Eng. 11(6), 865–879 (1999)