IDIAP RESEARCH REPORT

A Multitask Learning Approach to Document Representation using Unlabeled Data

Mikaela Keller [a] Samy Bengio [a]

IDIAP–RR 06-44

September 14, 2006

[a] IDIAP Research Institute, CP 592, 1920 Martigny, Switzerland, firstname.name@idiap.ch

IDIAP Research Institute **www.idiap.ch**

Rue du Simplon 4 P.O. Box 592 1920 Martigny – Switzerland

Tel: +41 27 721 77 11 Fax: +41 27 721 77 12 Email: info@idiap.ch

IDIAP Research Report 06-44

A Multitask Learning Approach to Document Representation using Unlabeled Data

Mikaela Keller Samy Bengio

September 14, 2006

Abstract. Text categorization is intrinsically a supervised learning task, which aims at relating a given text document to one or more predefined categories. Unfortunately, labeling such databases of documents is a painful task. We present in this paper a method that takes advantage of huge amounts of unlabeled text documents available in digital format, to counter balance the relatively smaller available amount of labeled text documents. A Siamese MLP is trained in a multi-task framework in order to solve two concurrent tasks: using the unlabeled data, we search for a mapping from the documents' bag-of-word representation to a new feature space emphasizing similarities and dissimilarities among documents; simultaneously, this mapping is constrained to also give good text categorization performance over the labeled dataset. Experimental results on Reuters RCV1 suggest that, as expected, performance over the labeled task increases as the amount of unlabeled data increases.

Contents

1 Introduction

In any Information Retrieval (IR) task, such as Document Retrieval or Text Categorization, the starting point is the preprocessing and representation of the documents, in order for them to be automatically processed. The most common representation of these documents used in IR is the so-called *bag-of-words* in which the document is seen as a vector of the size of the dictionary, each component w_i of the vector indicating the weighted presence or the absence of the ith dictionary word in the document.

Most machine learning approaches to IR tasks propose to learn a useful representation from the basic *bag-of-words*, using either labeled documents (and a supervised learning approach) or unlabeled documents (and an unsupervised learning approach). While the supervised learning approaches are in general better to solve a given task, the small amount of available labeled data is often problematic; on the other hand, unsupervised approaches can use large amounts of data but the obtained representation is not specifically chosen to solve a real task, rather to optimize some form of data likelihood (which may or may not be appropriate for a given task [13]).

In this paper, we propose a method that would jointly make use of unlabeled and labeled documents in order to solve one or more supervised learning tasks.

Let us consider a raw unlabeled text document. If we split it into two parts we can reasonably assume that these two parts are related somehow to each other, otherwise the author of the document would not have put them together. This *a priori* expected relatedness between two parts of the same document could help us in finding what makes two documents related to each other, based on the words they contain. We would like to incorporate this information in the representation of documents, using a large unlabeled corpus of documents, while constraining the obtained representation to also solve (at least) one supervised IR task on some other, labeled, documents.

For that we have chosen to learn a mapping from the *bag-of-words* representation to a more compact and useful representation. We learn this mapping in a Multi-task learning framework, where one task relies on the labeled training data available for the IR task, while the second task is to learn a representation in which related documents are close to each other and unrelated documents are far from each other.

The paper is organized as follows. In Section 2, we describe our proposed approach. Section 3 shows how this approach is related to various other models in the literature. Section 4 presents some experimental comparison between our approach and two competing approaches on a text categorization task over the Reuters RCV1 database. Finally, Section 5 concludes the paper.

2 Our Approach

As explained in the introduction, we would like to learn a mapping from a bag-of-words representation of documents to a richer and more informative representation. Let $\phi(.)$ be the mapping we are looking for. We would like that, in this new representation, topic related documents be more similar to each others than to documents discussing different subjects.

More formally, given a triplet (x, x^+, x^-) such that x is a document, x^+ is a document similar to x and x^- a document dissimilar to x, we would like the scalar product of the similar ones to be higher than that of the dissimilar ones:

$$\phi(x) \cdot \phi(x^+) > \phi(x) \cdot \phi(x^-). \tag{1}$$

Let us consider a large unlabeled corpus of documents $\mathcal{D}$, and in particular two documents $d, d' \in \mathcal{D}$. Let us divide d and d' in reasonably sized sub-parts, such as paragraphs. We stated in the introduction that we make the *a priori* assumption that any two paragraphs of d should by more related to each other than one paragraph of d and one of d'. We can then create a triplet (x, x^+, x^-) from $(d,\ d')$ by sampling any two paragraphs x, x^+ of d, and one paragraph x^- from d'. Let us call $\mathcal{D}_{unlab}$ a dataset of such triplets created from $\mathcal{D}$.

While $\mathcal{D}$ could be used to infer an interesting mapping $\phi(.)$, we would like this mapping to also be constrained to obtain good performance on some specific IR task for which we have access to labels. We will concentrate in this paper on a text categorization task, but the ideas presented here may also apply to other IR tasks.

The goal of text categorization is to assign automatically categories, among a predefined set, to documents. In this framework, we have access to a set of labeled training documents $\mathcal{D}_{lab}$, in which each document d has an associated vector $y = (y_1, \cdots, y_K)$, $y_j \in \{-1, 1\}$ indicating the membership of d to category j, for each j among the K predefined categories. A usual supervised approach is to train a function $f(x) = (f_1(d), \cdots, f_K(d))$ in order to maximize the micro-averaged F_1 score at the break-even point. The F_1 score is the compound of two measures used in the IR community, Precision and Recall, as follows:

$$\text{Precision} = \frac{N_{tp}}{N_{tp} + N_{fp}}, \quad \text{Recall} = \frac{N_{tp}}{N_{tp} + N_{fn}} \quad \text{and} \quad F_1 = \left(\frac{1}{2}\left[\frac{1}{\text{Recall}} + \frac{1}{\text{Precision}}\right]\right)^{-1}$$

where N_{tp} is the number of true positives (documents belonging to a category that were classified as such), N_{fp} the number of false positives (documents out of a particular category but classified as being part of it) and N_{fn} the number of false negatives (documents from a category classified as out of it). Precision and Recall are effectiveness measures, *i.e.* inside $[0, 1]$, the closer to 1 the better. Precision and Recall values may be tuned through the choice of a specific decision function threshold, $f_j(d) \lessgtr \theta_j$. The break-even point corresponds to a θ for which Precision is as close as possible to Recall. In micro-average mode $\theta_1 = \ldots = \theta_K$ and Precision and Recall are computed globally across categories.

Hence, we would like our mapping $\phi(.)$ to be selected in order to perform well on two separate tasks, as illustrated in Figure 1: we want to learn jointly a function $\psi_1(x) = f(\phi(x))$ using examples in $\mathcal{D}_{lab}$ and a function $\psi_2(x, x^+, x^-) = \phi(x) \cdot \phi(x^+) - \phi(x) \cdot \phi(x^-)$ with examples in $\mathcal{D}_{unlab}$. This is a *multi-task* framework [2] which can be reformulated by saying that we want to minimize jointly the expected risk of failing in one of the two tasks, hence to minimize the following risk for all documents:

$$R = \alpha_1 \int_{z=(d,y)} L_1(\psi_1(d), y)dz + \alpha_2 \int_{z=(x,x^+,x^-)} L_2(\psi_2(z))dz, \tag{2}$$

where L_1 and L_2 are the losses associated with each task, while α_1 and α_2 represent the relative weights we put in each task. Indeed, in our case, learning a function to categorize texts is our priority, while learning a representation of documents through unlabeled data is an auxiliary task.

A function such as $\psi_2(x, x^+, x^-)$ can be learned using a model similar to the *Siamese* neural network proposed in [4] and more recently explored in [6] (see Section 3.2). In our case $\phi(.)$ is an MLP, replicated three times for x, x^+ and x^-, and learned by the optimization of a ranking criterion with proximity constraints as in [15, 5, 9]:

$$L_2(\psi_2(x, x^+, x^-)) = |1 - \left[\phi(x) \cdot \phi(x^+) - \phi(x) \cdot \phi(x^-)\right]|_+$$

where $|z|_+ = \max(0, z)$. We chose to infer $\psi_1(d)$ with a K-output MLP. As illustrated in Figure 1, the first layer of ψ_1 is the same as the first layer of $\psi_2(x, x^+, x^-)$ and encodes $\phi(x)$. The parameters of $\psi_1(d)$ are trained by minimizing the following loss function:

$$L_1(\psi_1(d), y) = \frac{1}{K} \sum_{k=1}^{K} |1 - y_k \cdot [\psi_1(d)]_k|_+$$

where $[z]_k$ represents the zth component of vector z. The empirical risk corresponding to (2) is thus:

$$\hat{R} = \alpha_1 \cdot \sum_{(d,y) \in \mathcal{D}_{lab}} L_1(\psi_1(d), y) + \alpha_2 \cdot \sum_{(x,x^+,x^-) \in \mathcal{D}_{unlab}} L_2(\psi_2(x, x^+, x^-)).$$

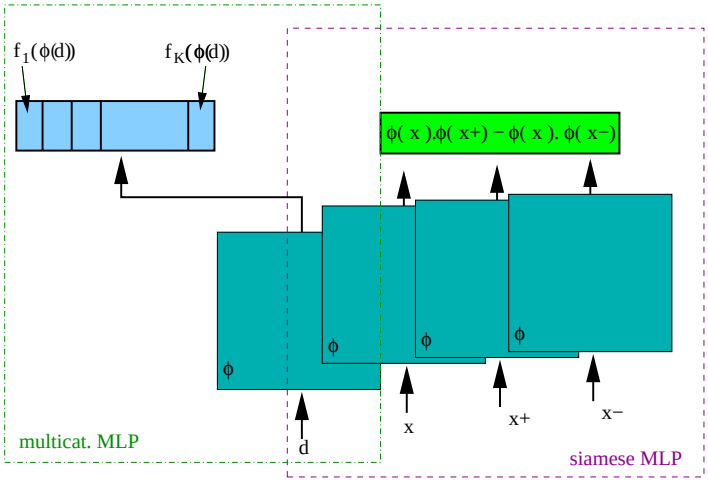

Figure 1: Neural Network Structure

The whole model is implemented as a neural network that can be trained by stochastic gradient descent, using both labeled and unlabeled examples. In fact, instead of fixing α_1 and α_2, we alternatively select randomly l labeled documents from $\mathcal{D}_{lab}$ and u unlabeled triplets in $\mathcal{D}_{unlab}$, for which we train the corresponding parameters of the model, and repeat these random selections over several epoch on $\mathcal{D}_{lab}$. The values of α_1 and α_2 are then not easy to explicit. Their values must take into account the ratio of learning rates for ψ_1 and ψ_2, the ratio of number of unlabeled vs labeled examples seen during training and the ratio of sparseness of the documents vs paragraphs. In the experiments reported in Section 4, a rough approximation is that we give three times more importance to the text categorization task. Furthermore, as we want to bias the model towards better solving the supervised task, we select the various hyper-parameters of $\phi(.)$ on a validation set composed of only labeled documents.

3 Related Work

Various approaches have been presented in the machine learning literature to learn a better text representation than *bag-of-words*. In order to present a simple typology of this domain we can consider two characteristics of these approaches. First, whether learning of the text representation takes into account some world knowledge through the use of auxiliary data; Second, whether it is related or specific to the supervised task to solve. We will present in the following some approaches related to our work according to this second dichotomy.

3.1 Unsupervised Learning of the Mapping

We briefly present here models performing an unsupervised learning of the mapping $\phi(.)$, by which we mean that the text representation is not trained directly in relation to a specific IR task.

The first group of approaches is not designed to take into account any auxiliary source of information, and thus the concerned approaches concentrate their efforts on the available labeled dataset.

The main drawback of the *bag-of-words* representation is the lack of information about the relations between the words composing it. The Latent Semantic Analysis (LSA) [7] approach thus tries to link words together according to their co-occurrences in a database of documents by performing a Singular Value Decomposition. The more recent Probabilistic Latent Semantic Analysis (PLSA) model [10] seeks a generative model for word/document co-occurrences. It makes the assumption that each word w_j in a given document d_i of the corpus is generated from a latent aspect t among a finite set of latent aspects. The introduction of the latent aspects allows to capture relations between words through the

estimation of $P(w_j|t)$ for all j. One weakness of these approaches is that they model each document in the corpus. This implies that they do not scale well for increasing databases, and thus they tend to be used only on the labeled data.

A second group of unsupervised approaches attempts at including information from auxiliary databases. The idea behind the Latent Dirichet Allocation (LDA) model [3] is close to PLSA. However, documents in the model are seen as sets of words and not as entities. Another difference is that LDA has a continuous latent space. In spite of the fact that LDA is not limited by the number of training examples as for PLSA, as far as we know LDA has never been used to take advantage of huge unlabeled databases.

The two following approaches attempt at including information extracted from expert annotated hierarchical semantic databases. The Semantic kernel proposed in [17] is based on a similarity between documents constructed with the help of the *Wordnet* database containing links between words. The Feature Generator proposed in [8] allows to enrich the *bag-of-words* with new words extracted from the *Open Directory Project* (Internet based URL repositories maintained by volunteers) taking advantage of the expert knowledge encoded there.

3.2 (Semi-)Supervised Learning of the Mapping

Another set of approaches, more similar to ours, try to learn the mapping $\phi(.)$ with the help of auxiliary data while being optimized for a specific supervised task.

The idea of the Structural Learning framework presented in [1], very close to the multi-tasks framework, is to solve several related tasks simultaneously in order to improve the learning of the structure that their solution may share. To encode this assumption the authors propose to model the predictors of their tasks as follow: $f_\Theta(x; w, v) =< [w^T, v^T], [x, \Theta x] >$, w and v being task specific, while the matrix Θ is common to all tasks. Experiments with tasks involving auxiliary unlabeled data have been conducted, leading to significant improvement over the single task performance. While giving a nice generic framework, this approach does not tackle the problem of learning a document representation directly and thus cannot target desired properties of this representation such as similarity between documents.

Another approach related to ours is the LinkLearn algorithm presented in [9]. The idea is to learn the optimal term weighting of the *bag-of-words* representation for a document retrieval task. This is done by optimizing a ranking criterion similar to the one we use in Section 2, over a huge hyperlinked corpus of document, *Wikipedia*. It is assumed that documents linked should be more similar than documents not linked, in the same way as a query and its relevant documents with respect to a query and any irrelevant document. Because the task solved to learn the document representation is very close to the targeted task we may say that the representation is optimized for the document retrieval task. On the other hand, the unlabeled data is used prior to solving the real task and not jointly, as in our approach.

In order to close this list of related works one has to cite the Trasductive SVM proposed in [].

4 Experiments

In this section, experiments comparing our approach to two other methods on the RCV1 (Reuters Corpus Volume 1) database [14] are reported. We have chosen to compare the multi-task learning algorithm to its single task counterpart (using only the labeled data), considering the latter as a baseline to see if the unsupervised data helps to better solve the supervised task. We further compare our model to Support Vector Machines which are considered to be the state-of-the-art approach in text categorization as reported in [16]. Details of the implementation of the compared models are as follows:

- **State-of-the-Art:** One Support Vector Machine per category with a linear kernel trained in a *one-versus-all* scheme. We used the SVMlight implementation [11] of SVMs with the parameter

C responsible for the trade-off between the training error and the margin is kept at its default value, that is $\left[\sum_{i=1}^{N} \|d_i\|^2/N\right]^{-1}$. Thus SVMs results are suboptimal.

- **Baseline:** An MLP with M inputs, 2 hidden layers of respectively h_1 and h_2 units and K outputs, M being the size of the vocabulary, and K the number of categories. h_1 and h_2 are chosen on a validation set among $\{50, 100, 200, 300, 500\}$. We refer to this model as a *multicat MLP*.
- **Two-Tasks:** A Siamese MLP with 1 hidden layer learned jointly with a multicat MLP. The mapping $\phi(.)$, common to the Siamese MLP and the multicat MLP is composed of the input layer and the first hidden layer of the multicat MLP. h_1 and h_2 are kept as optimally chosen for the baseline.

The RCV1 database is a corpus of $806,791$ news stories written over one year and labeled. There is a total of 101 categories and each document is labeled with one or more of these categories. For the experiments presented in the following, we have preserved the $6,945$ documents of the 4 last days as a test set, which we will note $\mathcal{D}_{test}$. From the remaining $799,846$ documents, noted $\mathcal{D}_{dev}$, we have sampled randomly sets of several sizes to simulate various $\mathcal{D}_{lab}$ and $\mathcal{D}_{unlab}$. In other words, each point in Figures 2, 3 and 4 simulates a scenario where we would be given a little set $\mathcal{D}_{lab}$ of labeled documents, and a big set $\mathcal{D}$ of unlabeled documents from which a set $\mathcal{D}_{unlab}$ of triplets is extracted. The dictionary is extracted from the biggest of the two sets, $\mathcal{D}$. This dictionary covers almost entirely $\mathcal{D}_{lab}$ one's. As can be seen in Table 1 the number M of words in the dictionary in our experiments is contained between 25K and 113K depending on the size of $\mathcal{D}$. It means that our MLPs have an input layer with a huge dimension. However, given that the bag-of-word representation is sparse, only a few of these dimensions are active any time an example is presented to the network and thus the propagation of the gradient, during the training, is also sparse. This allows our networks to scale to relatively huge amount of data despite the high dimension of their input layers.

4.1 Varying the Labeled Set Size

In this first experiment, we have first sampled from $\mathcal{D}_{dev}$ a set $\mathcal{D}$ of $10,000$ documents, from which we have extracted a set $\mathcal{D}_{unlab}$ of $116,032$ unlabeled triplets. After stopping and stemming the words present in $\mathcal{D}$, which is a standard preprocessing step of textual databases (see [16]), we obtained a vocabulary of $M = 25,138$ words. For each $s \in \{100, 200, 500, 1000\}$, we have then sampled a training set and a validation set of size s. The training set together with its corresponding validation set form a set $\mathcal{D}^s_{lab}$ of $2 \times s$ labeled documents. The documents in $\mathcal{D}^s_{lab}$ are transformed in their *bag-of-words* representation using the vocabulary extracted from $\mathcal{D}$. The documents in $\mathcal{D}^s_{lab}$ were used to train the state-of-the-art model, the baseline model and, with the help of $\mathcal{D}_{unlab}$, the two-tasks model. The selection of the labeled training set for each s was repeated 5 times in order to obtain an estimate of the variance in the performance due to the choice of the training set. The mean and standard deviation of the F_1 score at the break-even point (BEP) over the $\mathcal{D}^s_{lab}$s for each size s and each model, are reported in Figure 2 and analysis of the results is provided in Section 4.4.

4.2 Varying the Unlabeled Set Size

For this experiment we have sampled a set $\mathcal{D}_{lab}$ of $1,000$ labeled documents. For each $s \in \{10^4, 5 \times 10^4, 10^5, 2 \times 10^5\}$ we have sampled a set $\mathcal{D}^s$ of s documents. After stopping and stemming the words in each $\mathcal{D}^s$ we have extracted vocabularies of increasing size and unlabeled sets $\mathcal{D}^s_{unlab}$ of triplets as reported in Table 1. The state-of-the-art model was trained on $\mathcal{D}_{lab}$ while the two-tasks models used $\mathcal{D}_{lab}$ and $\mathcal{D}^s_{unlab}$, in order to see the effect of increasing the unlabeled material. Note however, that the hyper-parameters of the two-tasks models were kept as optimally chosen for the baseline model. Two separate training sets $\mathcal{D}_{lab}$ were used to train the models, and the corresponding results are reported in Figure 3 with more analysis given in Section 4.4.

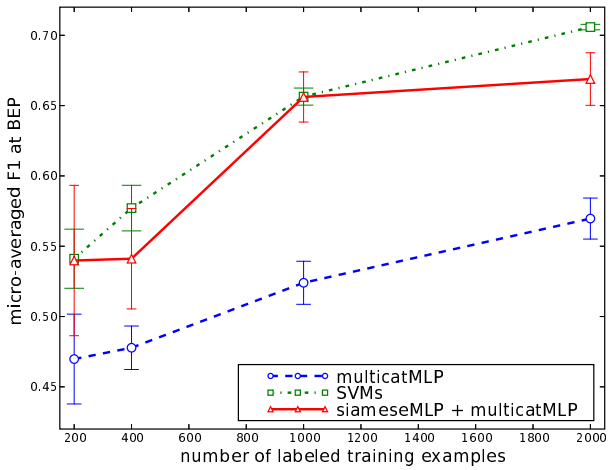

Figure 2: F_1 at BEP versus the size of the labeled training set. Each point and bar represent the mean and standard deviation of the performance obtained over the test set for 5 models, each trained on a different training set of the same size.

	Number s of documents in $\mathcal{D}^s$			
	10^4	5×10^4	10^5	2×10^5
M	$25,138$	$57,649$	$81,603$	$113,621$
$\lvert\mathcal{D}^s_{unlab}\rvert$	$116,032$	$590,043$	$1,179,164$	$2,343,769$

Table 1: Number of words in the vocabularies and number of triplets extracted from $\mathcal{D}^s$.

4.3 Verifying the Usefulness of the Multi-Task Framework

This experiment is a kind of sanity check, in order to see if the multi-task setup was really necessary or whether we could have learned separately the document representation on unlabeled data and then train a K outputs perceptron on the labeled set to obtain similar results. This experiment has been conducted using the same setup as described in experiment 4.2. The function $\psi_2(.)$ of Section 2 is trained alone on $\mathcal{D}^s_{unlab}$ with hyperparameters tuned on a validation set of triplets extracted from 100 documents. The resulting $\phi(.)$ mapping is used to project the documents $d \in \mathcal{D}_{lab}$ into a new representation, and the function $f(.)$ of Section 2 is then trained on this transformed data. Figure 4 compares the various settings.

4.4 Analysis

We infer from experiment 4.1 that the use of unlabeled data in the two-tasks model provides a considerable improvement with respect to the baseline performance. This improvement is emphasized by the results of experiment 4.3, which show that if we learn the document representation mapping as a separate task and then learn a categorization model over this new representation, performance improves when the unlabeled data increases, but it remains significantly lower than with the two-tasks approach. We can also see in Figure 2 that the performance of the two-tasks approach is close even if a little inferior to the state-of-the-art approach. However, we can infer from experiment 4.2 that the increase of unlabeled material yields an improvement of the two-tasks models performance, opening the possibility of even better performance with very large unlabeled datasets, possibility that the standard SVM does not have. Note finally that the difference in performance observed in Figure 3

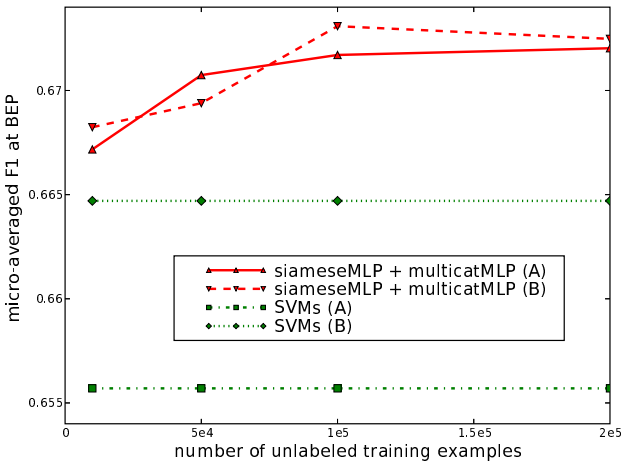

Figure 3: F_1 at BEP versus the size of the unlabeled training set for a fixed size of labeled training set (1000 documents). Compared models: state-of-the-art against two-tasks. Results for models trained over two training sets (A and B).

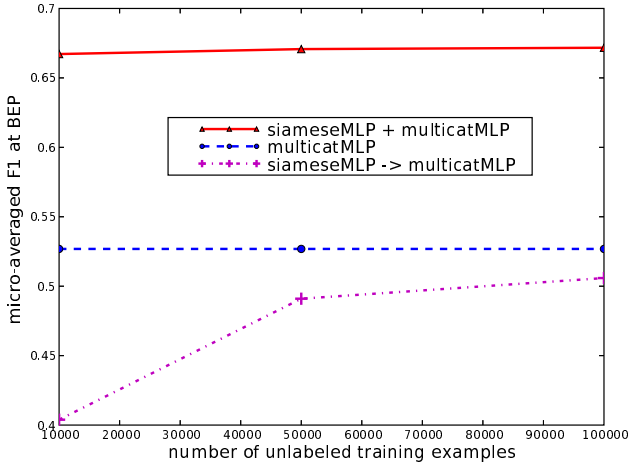

Figure 4: F_1 at BEP versus the size of the unlabeled training set for a fixed size of labeled training set (1000 documents). Compared models: baseline, two-tasks and successive learning of the mapping and the classifier.

between the state-of-the-art model and the two-tasks approach is statistically significant according to the bootstrap percentile test for F_1 score as defined in [12]. However, this significance test measure the significance with respect to the variability of the evaluation set, not to training set one, which at we can see in Figures 2 and 2 is quite high in our experiments.

5 Conclusion

Many information retrieval tasks are based on the careful selection (or estimation) of an appropriate representation space for documents. Estimating this representation through supervised learning approaches is limited by the scarce amount of available labeled documents. On the other hand, the use of (more easily available) unlabeled documents often leads to optimize a generic criterion, such as data likelihood, which may not necessarily be related to the information retrieval task. In this paper, we have proposed a model, based on the Siamese neural network and the *a priori* knowledge that similarity among documents should play an important role in the definition of the representation space. This model learns a representation space using simultaneously labeled and unlabeled documents by the optimization of a multi-task criterion. We have shown empirically, on the Reuters RCV1 database, that increasing the amount of unlabeled documents during training yielded better performance on a separate, supervised, task, and was able to reach, and sometimes surpass, the state-of-the-art approach based on SVMs. These preliminary results need to be confirmed on other, more standard databases, such as Reuters 21578, and compared against other unsupervised approaches.

Thanks

We would like to thank Johnny Mariéthoz for valuable discussions. This work was supported in part by the Swiss NSF through the NCCR on IM2 and in part by the European PASCAL Network of Excellence, IST-2002-506778, through the Swiss OFES.

References

[1] R.K. Ando and T. Zhang. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6:1817–1853, Nov 2005.

[2] J. Baxter. A model of inductive bias learning. *Journal of Artificial Intelligence Research*, 12:149–198, 2000.

[3] D. Blei, A. Ng, and M. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022, January 2003.

[4] J. Bromley, I. Guyon, Y. LeCun, E. Sackinger, and R. Shah. Signature verification using a siamese time delay neural network. In *Advances in Neural Information Processing Systems 6*, 1993.

[5] C.J.C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G.N. Hullender. Learning to rank using gradient descent. In *ICML*, pages 89–96, 2005.

[6] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *Proc. of Computer Vision and Pattern Recognition Conference*, 2005.

[7] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman. Indexing by Latent Semantic Analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.

[8] E. Gabrilovich and S. Markovitch. Feature generation for text categorization using world knowledge. In *Proceedings of The Nineteenth International Joint Conference for Artificial Intelligence*, Edinburgh, Scotland, 2005.

[9] D. Grangier and S. Bengio. Exploiting hyperlinks to learn a retrieval model. In *NIPS Workshop on Learning to Rank*, 2005.

[10] T. Hofmann. Unsupervised learning by Probabilistic Latent Semantic Analysis. *Machine Learning*, 42:177–196, 2001.

[11] T. Joachims. *Learning to Classify Text Using Support Vector Machines*. Kluwer, 2002.

[12] M. Keller, S. Bengio, and S.Y. Wong. Benchmarking non-parametric statistical tests. In *Advances in Neural Information Processing Systems (NIPS) 18*. MIT Press, 2005.

[13] Y. LeCun and F. J. Huang. Loss functions for discriminative training of energy-based models. In *Proc. of AIStats*, 2005.

[14] T.G. Rose, M. Stevenson, and M. Whitehead. The Reuters Corpus Volume 1 - from yesterday's news to tomorrow's language resources. In *Proceedings of the 3rd Int. Conf. on Language Resources and Evaluation*, 2002.

[15] M. Schultz and T. Joachims. Learning a distance metric from relative comparison. In *Advances in Neural Information Processing Systems 16*, 2003.

[16] F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.

[17] G. Siolas and F. d'Alché Buc. Support vectors machines based on a semantic kernel for text categorization. In *IEEE-ICANN-IJCNN*, 2000.