VideoSense at TRECVID 2011 : Semantic Indexing from Light Similarity Functions-based Domain Adaptation with Stacking

Emilie Morvant[1], Stéphane Ayache[1], Amaury Habrard[2], Miriam Redi[3], Claudiu Tanase[3],
Bernard Merialdo[3], Bahjat Safadi[4], Franck Thollard[4], Nadia Derbas[4], Georges Quenot[4]
[1] Aix-Marseille Univ, LIF-QARMA, CNRS UMR 7279, F-13013, Marseille, France
[2] University of St-Etienne, Lab. Hubert Curien, CNRS UMR 5516, F-42000, St-Etienne, France
[3] EURECOM, Sophia Antipolis, 2229 route des cretes, Sophia-Antipolis, France
[4] UJF-Grenoble 1 / UPMF-Grenoble 2 / Grenoble INP / CNRS, LIG UMR 5217, Grenoble, F-38041, France

March 21, 2012

Abstract

This paper describes our participation to the TRECVID 2011 challenge [1]. This year, we focused on a stacking fusion with Domain Adaptation algorithm. In machine learning, Domain Adaptation deals with learning tasks where the train and the test distributions are supposed related but different. We have implemented a classical approach for concept detection using individual features (low-level and intermediate features) and supervised classifiers. Then we combine the various classifiers with a second layer of classifier (stacking) which was specifically designed for Domain Adaptation. We show that, empirically, Domain Adaptation can improve concept detection by considering test information during the learning process.

1 Introduction

The High-Level semantic retrieval task concerns features or concepts such as "Indoor/Outdoor", "People", "Speech" etc., that occur in video databases. The TRECVID SIN task [1] contributes to work on a benchmark for evaluating the effectiveness of detection methods for semantic concepts. The task is as follows: given the feature test collection composed of hundred of hours of videos, the common shot boundary reference for the feature extraction test collection, and the list of feature definitions, participants return for each feature the list of at most 2000 shots from the test collection, ranked according to the highest possibility of detecting the presence of the feature. Each feature is assumed to be binary, *i.e.*, it is either present or absent in the given reference shot.

The VideoSense[1] project aims at automatic video tagging by high level concepts, including static concepts (*e.g.* object, scene, people, etc.), events, and emotions, while targeting two applications, namely video recommendation and ads monetization. The innovations targeted by the project include video content description by low-level features, emotional video content recognition, cross-concept detection and multimodal fusion, and the use of a pivot language for dealing with multilingual textual resources associated with video data.

[1] The french project VideoSense ANR-09-CORD-026 of the ANR and the IST Programme of the European Community.

The first participation of the project to the TRECVID'11 SIN task is based on a stacking fusion with Domain Adaptation. Domain adaptation is a machine learning task consisting in adapting a classifier on new data that come from a distribution different but related to the distribution of the train examples. Since the TRECVID'11 corpus is a real corpus with amount data, the train data could not be representative of all the test data. Our aim is to test a Domain Adaptation algorithm that combines various classifier outputs from individual feature in order to adapt the learned classifier on the test data for improving the performances.

In the following, section 2 lists the features and classifiers we used. Section 3 presents our algorithm for Domain Adaptation namely DASF. Section 4 describes our fusion approach with DASF. Then, in section 5, we show the runs we submitted and discuss about their relative performance.

2 Feature extraction and individual classifiers

Since the VideoSense members are also members of the IRIM project which participate to TRECVID'11, features and individual classifiers are described in the paper [2].

We used the following features from LIG, EURECOM and LIF teams:

Global features: LIG/hg104 (Color histogram + Gabor filters)

Local features: LIG/opp_sift, LIG/stip (SIFT and STIP)

Shape features: EUR/sm462 (Saliency Moments)

Intermediate features: LIF/percepts (Mid-level concepts based on 15 concepts).

We used two types of classifiers from LIF team:

KNN: LIG_KNNB (multiclass KNN)

SVM: LIG_MSVM (for imbalanced data).

3 DASF: Domain Adaptation with Similarity Functions

Domain Adaptation arises when learning and test data are generated according to two different probability distributions: the first one generating training data is often referred to as the *source domain*, while the second one for test data corresponds to the *target domain*. According to the existing theoretical frameworks of Domain Adaptation [3, 4, 7] a classifier can perform well on the target domain if its error relatively to the source distribution and the divergence (or distance) between the source and target distributions are together low. One possible solution to learn a performing classifier on the target domain is to find a projection space in which the source and target distributions are close while keeping a low error on the source domain.

In order to illustrate quickly this idea in a formal manner, if err_T denotes the error on the target domain (ie our goal here) and err_S the error on the source domain where labeled data are available, then for any classifier h belonging to an hypothesis space $\mathcal{H}$, we have from [3]:

$$\forall h \in \mathcal{H},\ err_T(h) \ \leq\ err_S(h) + \frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(D_S, D_T) + \nu. \tag{1}$$

The term ν can be seen as a kind of adaptation ability measure of $\mathcal{H}$ for the Domain Adaptation problem considered and corresponds to the error of the best joint hypothesis over the two domains: $\nu = \arg\min_{h\in\mathcal{H}} err_S(h) + err_T(h)$. The second term $d_{\mathcal{H}\Delta\mathcal{H}}(D_S, D_T)$ is called the distance between the two domain marginal probability distributions: D_S for the source and D_T for the target. Note that both $err_S(h)$ and $d_{\mathcal{H}\Delta\mathcal{H}}$ can be estimated from finite samples.

Our algorithm addresses Domain Adaptation for binary classification in the challenging case where no target label is available. Following the Domain Adaptation framework of Ben-David *et al.* [3], our method looks for a relevant projection space where the source and target distributions tend to be close. Our approach is formulated as a linear program with a 1-norm regularization leading to sparse models. To improve the efficiency of the method we propose an iterative version based on a reweighting scheme of the similarities to move closer the distributions in a new projection space. Hyperparameters and reweighting quality are controlled by a reverse validation process.

Furthermore, our approach is based on a recent framework of Balcan *et al.* [5] allowing to learn linear classifiers in an explicit projection space based on *good similarity functions* defined as follows. Roughly speaking, under a criterion of goodness introduced in [5], a good similarity function ensures that a low error linear classifier exists in a space made of similarity scores to a set of prototype examples called reasonable points. The learned linear classifier is of the form:

$$h(\cdot) = \sum_{i=1}^{d_u} \alpha_i K(\cdot, \mathbf{x}'_i)$$

where the examples $\mathbf{x}_i$ correspond to this so called set of reasonable points. The formulation is close to the one of SVM, except that the framework allows one to use similarity functions K that are more general than kernels in the sense K is not required to be symmetric nor positive semi-definite. The main idea of our Domain Adaptation algorithm, called DASF, consists in automatically modifying the projection space to the similarities (ie $\phi^R(\cdot) =< K(\cdot,\mathbf{x}'_1),\ldots,K(\cdot,\mathbf{x}'_i),\ldots,K(\cdot,\mathbf{x}'_{d_u}) >$) for moving closer source and target points. For this purpose, we proposed a general method based on the optimization of a regularized convex objective function trying to find reasonable points close both to source and target examples. The objective function proposed in our algorithm is defined as follows:

$$\left\{\begin{array}{rcl} \min\limits_{\boldsymbol{\alpha}} \;\; F(\boldsymbol{\alpha}) & = & \dfrac{1}{d_l}\displaystyle\sum_{i=1}^{d_l} L\big(h,(\mathbf{x}_i,y_i)\big) + \lambda\|\boldsymbol{\alpha}\|_1 \\ & & +\beta \displaystyle\sum_{(\mathbf{x}_s,\mathbf{x}_t)\in\mathcal{C}_{ST}} \Big\|\big({}^t\phi^R(\mathbf{x}_s) - {}^t\phi^R(\mathbf{x}_t)\big)\,\mathrm{diag}(\boldsymbol{\alpha})\Big\|_1, \\ \multicolumn{3}{l}{\text{with } L\big(h,(\mathbf{x}_i,y_i)\big) = \Big[1 - y_i h(\mathbf{x}_i)\Big]_+ \text{ and } h(\mathbf{x}_i) = \displaystyle\sum_{j=1}^{d_u} \alpha_j K(\mathbf{x}_i,\mathbf{x}'_j).} \end{array}\right. \qquad (DASF_{opt})$$

Examples $(\mathbf{x}_i, y_i)$ are labeled examples from the source domain, $L(\cdot)$ is the loss function optimized for learning classifiers, $\mathbf{x}'_j$ being the "reasonable" points considered. We have two regularizers, one is a classical $L1$ norm over the weights α_j of the linear classifier $\boldsymbol{\alpha}$. The complex last one, weighted by a parameter β, corresponds to the term trying to move closer some source instances $(\mathbf{x}_s)$ with target examples $(\mathbf{x}_t)$ belonging to a set chosen by reverse validation.

More details can be found in [6]. Note that we have provided theoretical results with DASF and that our algorithm has been evaluated on a toy problem and on two real image annotation tasks.

4 Classifier fusion with DASF

We used DASF algorithm as a stacking classifier using outputs from individual classifier on the considered features. Stacking provides a way of combining classifiers together to find an overall system with usually improved generalization performance [8]. In the context of SIN task, we considered the training set as source domain and the test set as target domain, although they are supposed to be already closed. However, as the challenge is based on real data, the training set could not be representative of the test set. We thus expect DASF to improve further the performance of video indexing.

In order to make good use of the similarity function framework of Balcan *et al.* we proposed to compare two different similarity function. The first one is a usual Gaussian kernel (which is symmetric and PSD). For the second one we build a new similarity function by normalizing the Gaussian kernel. In order to link the two domains, we consider the information of both of them at the same time by actually renormalizing the Gaussian kernel for all the source and target points relatively to the set of similarities to the reasonable points. By construction, the similarity is then non-symmetric and non-PSD. Our choice is clearly heuristic and our aim is just to evaluate the interest of renormalizing a similarity for domain adaptation problems. Finally, we made a light search of the hyperparameters to accelerate our approach.

5 Results

In the following, we report our results (infAP) on the full SIN task. We compare DASF with SF as our baseline, SF-classifier is a linear classifier without Domain Adaptation, proposed for leaning with good similarity function ([5]). Both SF and DASF have been submitted with and without normalization. We can observe that:

- A poor performance of our runs compare to the best run and the median. The stacking process as we implemented seems to overfit data ;
- Normalization was not successful, probably because source and target domains was actually closed: in such a particular case, we lost information by making use of the normalization ;
- As expected Domain Adaptation runs both outperformed our baseline. Even if train and test set are similar, we can still take advantage of Domain Adaptation approach by considering (test) target information during the learning process.

Videosense 1	0.067	DASF with normalization
Videosense 2	0.040	SF with normalization
Videosense 3	0.080	DASF
Videosense 4	0.072	SF
Best run	0.1731	-
Median	0.1083	-

6 Conclusion

Our main focus for TRECVID'11 SIN task is on the exploration of Stacking with Domain Adaptation to combine individual classifiers. Taking into account target information by using Domain Adaptation has allowed us to improve baseline results which was our main objective. However, we suspect that we actually have overfitted data and unfortunately was not able to generalize as we expected. This can be explained by the following reasons:

- We did not consider the goodness of the similarity function used (ie the Gaussian kernel) according to the outputs of the individual classifiers. In particular we are not sure that such a similarity is the best choice with our framework. Optimizing the similarity with some metric learning approaches, for example, would help us to better adapt our framework to the stacking problem considered.
- The renormalization of the kernel function used in our approach is simple but we know that it works well when the two domains are very different. The fact that this renormalization does not increase the performance shows that the train and test data are not that different here. By combining an information on the distance domains and some metric learning approaches, as evoked before, we may drastically improve the results.
- In domain adaptation, the estimation of the hyperparameters is difficult and costly. In our runs, we simplified a bit the search of the parameters which may explain why we tend to overfit.
- We did not take into account some second order (ie variance/covariance) information from the outputs of the classifiers and we think that it may be useful to find a better combination.

Our first attempt shows that Domain Adaptation could bring some improvement. However, to be competitive with the state of the art, we have to take into account more accurately the diversity of the data and classifiers. It appears especially important in the case of multiple source of annotations. Under this condition, we think that Domain Adaptation approaches can lead to significant improvement of the results. Using some correlations between labels is in particular an interesting direction.

Acknowledgment

This work was supported in part by the french project VideoSense ANR-09-CORD-026 of the ANR.

References

[1] Paul Over, George Awad, Martial Michel, Jonathan Fiscus, Wessel Kraaij, Alan F. Smeaton, and Georges Quenot. Trecvid 2011 – an overview of the goals, tasks, data, evaluation mechanisms and metrics. In *Proceedings of TRECVID 2011*. NIST, USA, 2011.

[2] Bertrand Delezoide, Frédéric Precioso, Philippe Gosselin, Miriam Redi, Bernard Mérialdo, Lionel Granjon, Denis Pellerin, Michèle Rombaut, Hervé Jégou, Rémi Vieux, Boris Mansencal, Jenny Benois-Pineau, Stéphane Ayache, Bahjat Safadi, Franck Thollard, Georges Quénot, Hervé Bredin, Matthieu Cord, Alexandre Benoit, Patrick Lambert, Tiberius Strat, Joseph Razik, Sébastion Paris, and Hervé Glotin. Irim at trecvid 2011: Semantic indexing and instance search. In *Proceedings of TRECVID 2011*. NIST, USA, 2011.

[3] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J.W. Vaughan. A theory of learning from different domains. *Machine Learning Journal*, 79(1-2):151–175, 2010.

[4] Y. Mansour, M. Mohri, and A. Rostamizadeh. Domain adaptation: Learning bounds and algorithms. In *Proceedings of COLT*, pages 19–30, 2009.

[5] M.-F. Balcan, A. Blum, and N. Srebro. Improved guarantees for learning via similarity functions. In *Proceedings of COLT*, pages 287–298, 2008.

[6] Emilie Morvant, Amaury Habrard, and Stéphane Ayache. Sparse Domain Adaptation in Projection Spaces based on Good Similarity Functions. In *11th IEEE International Conference on Data Mining (ICDM)*, pages 457–466. IEEE Computer Society, 2011.

[7] S. Ben-David, T. Lu, T. Luu, and D. Pal. Impossibility theorems for domain adaptation. *JMLR W&CP*, 9:129–136, 2010.

[8] Padhraic Smyth and David Wolpert. Linearly combining density estimators via stacking. *Machine Learning*, 36(1-2):59–83, 1999.