Multi-View Learning with Dependent Views

Ulf Brefeld
Knowledge Mining & Assessment Group
TU Darmstadt and DIPF
brefeld@cs.tu-darmstadt.de

ABSTRACT

Multi-view algorithms, such as co-training and co-EM, utilize unlabeled data when the available attributes can be split into independent and compatible subsets. Experiments have shown that multi-view learning is *sometimes* beneficial for problems for which the independence assumption is not satisfied. In practice, unfortunately, it is not possible to measure the dependency between two attribute sets; hence, there is no criterion which allows to decide whether multi-view learning is applicable. We conduct experiments with various text classification problems and investigate on the effectiveness of the co-trained SVM and the co-EM SVM under various conditions, including violations of the independence assumption. We identify the error correlation coefficient of the initial classifiers as an elaborate indicator of the expected benefit of multi-view learning.

Categories and Subject Descriptors

I.2.6 [**Artificial Intelligence**]: Parameter learning

General Terms

Algorithms, Experimentation

Keywords

Multi-view learning

1. INTRODUCTION

Multi-view algorithms – such as co-training [1] – split the attributes into two independent subsets, each of which has to be sufficient for learning. An example of a domain that is suitable for multi-view learning is web page classification: a page can be classified based on its content as well as based on the anchor texts of its inbound hyperlinks.

Multi-view algorithms learn two independent classifiers based on independent attribute subsets. These classifiers then provide each other with labels for the unlabeled data. Because of the independence between the views, an example that has been labeled by one classifier behaves much like a randomly drawn (slightly noisily) labeled example for the second classifier. Practical applications, however, hardly ever possess two views that are perfectly independent. Still, multi-view learning frequently improves classification results. In fact, experiments with text classification problems show that multi-view learning with *random* partitions of the attributes into two views sometimes lead to more accurate results than the corresponding single-view classifier [4, 12]. These observations raise the question which property of a given learning problem is indicative on the suitability of multi-view learning. Investigating on this question, we identify the error correlation coefficient Φ^2 of the initial classifiers as a measure for the potential benefit of multi-view learning.

The contribution of this paper is twofold. Firstly, we report on many experiments with co-trained Support Vector Machines (SVMs) and co-EM Support Vector Machines [4]. We experiment with semi-artificial text classification problems that allow us to introduce controlled violations of the independence assumption, with the Reuters-21578 data set, and with the course web page classification data set. We find that – even with random attribute splits – multi-view learning is often very beneficial for text classification. Secondly, we report on experiments that reveal the relevance of the error correlation coefficient Φ^2 for the potential benefit of multi-view learning.

The rest of our paper is organized as follows. We discuss related work in Section 2. We formulate the problem setting in Section 3. In Section 4, we review the co-trained and co-EM Support Vector Machine. We introduce the error correlation coefficient Φ^2 statistics in Section 5 and report on our experimental results in Section 6. Section 7 concludes.

2. RELATED WORK

Multi-view learning algorithms address the problem of *semi-supervised learning* ([6, 20]), where a learning algorithm has access to a limited amount of labeled data and (typically many) unlabeled examples. The Expectation Maximization (EM) algorithm [7] is probably the most prominent approach for learning from labeled and unlabeled data [13, 17]. It is wrapped around learning algorithms that fit model parameters to probabilistically labeled data.

Co-training and the *co-EM* algorithm are *multi-view learning methods*; that is, they utilize unlabeled data, provided that the attributes can be split into two subsets that are both sufficient for learning. The co-training algorithm [1]

SAC'15, April 13-17, 2015, Salamanca, Spain.

ACM 978-1-4503-3196-8/15/04 ...$15.00.
http://dx.doi.org/10.1145/2695664.2695829.

learns two independent decision functions which bootstrap each other (see Section 4). Co-training is easily applicable for support vector learning; in the context of text classification, the co-trained Support Vector Machine has shown to outperform a co-trained naive Bayesian classifier [11, 12].

The co-EM algorithm [17, 9] combines multi-view learning and the EM algorithm [7]. Co-EM (with naive Bayes as underlying classifier) has been found to outperform co-training in some cases [17]; in particular, when the compatibility and independence assumptions (see Section 4) are not violated [15]. A meta-learning approach to selection of single- or multi-view learners has been studied [16]. A co-EM version of the Support Vector Machine has been developed [4] which has in turn shown to outperform co-EM with naive Bayes for text classification problems.

Applications of co-training that have been studied include classification of web pages [1], named entity recognition [5], text classification [8], wrapper induction [16], classification of emails [11, 12], and word form normalization [14].

3. PROBLEM SETTING

We focus on the *semi-supervised learning* setting in which *labeled data* $D_l = \langle (x_1, y_1), \ldots, (x_{m_l}, y_{m_l}) \rangle$, $y_i \in \{+1, -1\}$ and *unlabeled data* $D_u = \langle x_1^*, \ldots, x_{m_u}^* \rangle$ are available. Our goal is to learn a *decision function* $f(x)$ which assigns high values to positive and low values to negative examples. The ability of a decision function to discriminate positives against negatives is naturally characterized by the *receiver operating characteristic (ROC)* analysis [2, 19].

In the multi-view setting that we discuss, the available attributes V are split into disjoint sets V_1 and V_2. A labeled instance (x, y) is decomposed and viewed as (x_1, x_2, y), where x_1 and x_2 are vectors over the attributes V_1 and V_2, respectively. These views have to be *independent* and *compatible*.

DEFINITION 1. *Views V_1 and V_2 are* independent *when* $\forall x_1 \in V_1, x_2 \in V_2 : p(x_1, x_2|y) = p(x_1|y)p(x_2|y)$.

DEFINITION 2. *Views V_1 and V_2 are* compatible *with target concept $t : x \mapsto y$ when there are hypotheses $h_1 : V_1 \to \{-1, +1\}$ and $h_2 : V_2 \to \{-1, +1\}$ such that, for all $x = (x_1, x_2)$, $f_1(x_1) = f_2(x_2) = t(x)$.*

4. MULTI-VIEW LEARNING

In this section, we review the co-training and co-EM Support Vector Learning algorithms. In each iteration of the *co-training algorithm* (Table 1), each of the two decision functions commits to class labels for (at least) one positive and one negative example – the ones that are most confidently rated positive and negative. In contrast to co-EM, co-training never revises conjectured labels for unlabeled data. Since co-training only requires the underlying learning algorithm to return values of a (possibly uncalibrated) decision function, the Support Vector Machine can be integrated as easily as the naive Bayes algorithm. Since the results of experiments with text data have clearly favored the co-trained SVM (*e.g.,* [11, 12]), we focus on the SVM.

The co-training algorithm has a favorable theoretical property: because of their independence, the two decision functions can provide each other with labels for the unlabeled data in a way that is essentially equivalent to drawing (slightly noisy) labeled examples at random [1]. A co-training step improves the classifier performance when one classifier errs for an unlabeled instance, whereas the peer classifier is very confident and adds the correct class label to the labeled data. The independence of the views reduces the chance of both hypotheses agreeing on an erroneous label of an unlabeled instance. We now review the co-EM algorithm. In order to be able to better compare the results of co-training and co-EM – and because the multi-view SVM has exhibited a more favorable performance than multi-view naive Bayes – we discuss how the SVM can be cast into the co-EM framework. In order to implement a co-EM version of the Support Vector Machine, we have to address two principal difficulties [4]: The co-EM algorithm requires each classifier to yield class probability estimates for the unlabeled data. Additionally, we have to construct a learning algorithm that utilizes data which have been labeled with class probabilities for training.

Table 1: The co-training algorithm.

Co-training. Input: Labeled data D_l, unlabeled data D_u, parameters.

1. Train f_0^1 and f_0^2 on D_l using attribute sets V_1 and V_2, respectively.
2. **For** $i = 1 \ldots T$ until $D_u = \emptyset$:
   (a) **For** $v = 1 \ldots 2$: Remove n_p elements with greatest $f_{i-1}^v(x_j^*)$, from D_u and add $(x_j^*, +1)$ to D_l.
   (b) **For** $v = 1 \ldots 2$: Remove n_n elements with smallest $f_{i-1}^v(x_j^*)$, from D_u, add $(x_j^*, -1)$ to D_l.
   (c) Train f_i^1 and f_i^2 on D_l using attribute sets V_1 and V_2, respectively.
3. **Return:** $\frac{1}{2}(f_T^1 + f_T^2)$.

A linear classifier f gives us an uncalibrated decision function $f(x) = w^\top x$, but we need an estimate of the class posterior $p(y|x)$. We assume a parametric model: the decision function values for a class, $p(f(x)|y)$, are assumed to be governed by a normal distribution $N[\mu, \sigma^2]$. We estimate the parameters μ and σ^2 during training from the given labeled and unlabeled training data. Firstly, we estimate the prior probabilities $\hat{p}(y)$ from the *labeled* data. We split the unlabeled data into positives and negatives according to the fixed ratio $\hat{p}(y)$; the unlabeled instances x_j^* with highest $f(x_j^*)$ are rated positive. Secondly, we estimate the mean decision function values μ_+ and μ_- and corresponding variances σ_+^2 and σ_-^2 from this data. From the priors $\hat{p}(y)$ and Gaussian likelihoods with parameters μ_+, μ_-, σ_+^2, and σ_-^2, we can infer the desired class probabilities $\hat{p}(y|x_j^*)$ (Eq. 1).

We need to address a second problem: Given labeled data D_l, and unlabeled data D_u with class probability estimates $\hat{p}(y|x_j^*)$, how can we train a support vector classifier? Intuitively, if $\hat{p}(y|x_j^*) = 1$ for some instance x, then that instance is essentially a labeled example and should contribute to the optimization criterion accordingly. On the other hand, $\hat{p}(y|x_j^*) = 1/2$ indicates a lack of information about the class label of x_j^*; the optimization criterion should not be influenced by the class label it assigns to such an x_j^*.

We introduce an individual weight for each example into the optimization criterion analogously to [3]; we define the weight such that we achieve a smooth transition from full contribution for $\hat{p}(y|x_j^*) = 1$ to no contribution for

Table 2: The co-EM SVM algorithm.

Co-EM SVM. Input: Labeled data D_l, unlabeled data D_u, slack parameter C, number of iterations T.

1. Initialize smoothing factor $C_S = \frac{1}{2^T}$
2. Train initial support vector machine f_0^2 on labeled data D_l using the view V_2.
3. Estimate $\hat{p}(y)$ using the labeled data D_l.
4. **For** $i = 1 \ldots T$: **For** $v = 1 \ldots 2$:
   (a) **Let** D_u^+ be the $\hat{p}(y=1)|D_u|$ many unlabeled examples with highest decision function values $f_{i-1}^{\bar{v}}(x_j^*)$ (use decision function with complementary view $\bar{v}$); **Let** $D_u^- = D_u \setminus D_u^+$.
   (b) Estimate (μ_+, σ_+^2) from D_l^+ and D_u^+, and (μ_-, σ_-^2) from D_l^- and D_u^-.
   (c) For all unlabeled data x_j^*, estimate $\hat{p}(y|x_j^*)$ (Eq. 1), based on $f_{i-1}^{\bar{v}}$.

$$\hat{p}(y|x_j^*) = \frac{N[\mu_y, \sigma_y^2](f(x_j^*))\hat{p}(y)}{\sum_{z\in\{+1,-1\}} N[\mu_z, \sigma_z^2](f(x_j^*))\hat{p}(z)} \quad (1)$$

   (d) Train f_i^v by solving the co-EM SVM optimization problem with smoothing factor C_S; that is, let $c_{x_{j^*}} = (\max_y \hat{p}(y|x_j^*) - \min_y \hat{p}(y|x_j^*))$ and minimize Eq. 2 subject to the constraints $\forall_{j=1}^{m_l} y_j(wx_j + b) \geq 1 - \xi_j$, $\forall_{j=1}^{m_u}(\text{argmax}_y \hat{p}(y|x_j^*))(wx_j^* + b) \geq 1 - \xi_j^*$, and $\forall_{j=1}^{m_l}\xi_j > 0$, $\forall_{j=1}^{m_u}\xi_j^* > 0$, using the attributes in view V_v.

$$\min_{w,b,\xi,\xi^*} \frac{1}{2}|w|^2 + C\left(\sum_{j=1}^{m_l}\xi_j + C_S\sum_{j=1}^{m_u} c_{x_j^*}\xi_j^*\right) \quad (2)$$

   (e) **End For** v; **Let** $C_S = 2C_S$; **End For** i.
5. **Return** the combined function $\frac{1}{2}(f_T^1 + f_T^2)$.

$\hat{p}(y|x_j^*) = 1/2$. We label an unlabeled instance x_j^* with $y = \text{argmax}_y \hat{p}(y|x_j^*)$ and define its weight to be $c_{x_j^*} = \max_{y'} \hat{p}(y'|x_j^*) - \min_{y'} \hat{p}(y'|x_j^*)$. In order to reduce the risk of finding local minima, we copy the smoothing strategy of the transductive SVM [10] and multiply the contributions of the unlabeled data by an initially small number C_S which is doubled in each iteration until it reaches one (Eq. 2). The resulting co-EM SVM algorithm is shown in Table 2.

Intuitively, when x is a large margin example for f^1, then f^1 has a small error probability for x. When V_1 and V_2 are truly independent, then the projection of x into V_2 is a randomly drawn instance in V_2; x may be a support vector in V_2 even though it is a large-margin example in V_1. The co-EM SVM labels each unlabeled example in V_2 with the class label assigned by f^1. Unlike co-training which assigns class labels greedily to unlabeled data, co-EM can revise the assigned class labels in each step. Note that we can trivially extend the co-EM SVM to non-linear functions by moving from the primal to the dual representation of the optimization criterion and replacing the inner products by kernel functions.

5. ERROR CORRELATION COEFFICIENT Φ^2

The previous section points out that multi-view learning works most effectively when the views are independent. Unfortunately, given a data set with two *sets* of *continuous valued* attributes, it is not possible to quantify the dependency between them; hence there is no measurable criterion that determines the potential benefit of multi-view learning for any given learning problem. Note that, given two sets of continuous attributes, it is possible to measure the *linear correlation* between them. The lack of a linear correlation, however, is fundamentally weaker than the *statistical independence* that is required to prove that the labels that are generated by multi-view learning behave much like randomly drawn labeled data.

Two individual, discrete random variables can be tested for statistical independence based on the χ^2 statistics that is derived from their contingency table. How can we infer two discrete random variables from two sets of possibly continuous attributes without losing any relevant information about dependencies? Our observation now is that the independence of the attributes is actually required to prove that the misclassification risk of one resulting classifier is independent of the misclassification risk of the second classifier. Therefore, dependencies between the views are not harmful, unless they result in dependencies between the initial classifiers in the two views.

We formalize this by introducing two binary random variables E_1 and E_2 that indicate whether either machine errs on a random example. On a test set, we can determine a 2×2 contingency table of the joint frequencies $P(E_1 = i, E_2 = j)$ $(i, j \in \{0, 1\})$. The correlation of E_1 and E_2 is given by r_{E_1,E_2} (Eq. 3).

$$r_{E_1,E_2} = \frac{P(E_1=1,E_2=1) - P(E_1=1)P(E_2=1)}{\sqrt{P(E_1=0)P(E_1=1)}\sqrt{P(E_2=0)P(E_2=1)}} \quad (3)$$

Due to binary random variables, the sign of r_{E_1,E_2} is determined by the arbitrary mapping of observations to the realizations 0 and 1, respectively. Thus, the correlation of two binary random variables $\bar{E}_1$ and $\bar{E}_2$, indicating whether a machine is correct on a random example, is $r_{\bar{E}_1,\bar{E}_2} = -r_{E_1,E_2}$.

Therefore, the intrinsic dependency of E_1 and E_2 is measured by the unsigned, squared *error correlation coefficient* Φ^2 (Eq. 4). By means of simple algebraic transformations we derive the equivalent notation of Eq. 5.

$$\Phi^2 = r_{E_1,E_2}^2 = r_{\bar{E}_1,\bar{E}_2}^2 \quad (4)$$

$$= \sum_{i,j=0}^{1} \frac{(P(E_1=i,E_2=j) - P(E_1=i)P(E_2=j))^2}{P(E_1=i)P(E_2=j)} \quad (5)$$

In order to get a better feeling for the properties of Φ^2, let us prove the following propositions.

PROPOSITION 1. *$\Phi^2 = 0$ if and only if E_1 and E_2 are independent.*

Proof. If E_1 and E_2 are independent each numerator of the sums equals zero since – by the definition of independence – $P(E_1 = i, E_2 = j) = P(E_1 = i)P(E_2 = j)$ holds for all i, j; therefore $\Phi^2 = 0$. Φ^2 can only become zero if the numerator becomes zero, this is equivalent to $P(E_1, E_2) = P(E_1)P(E_2)$

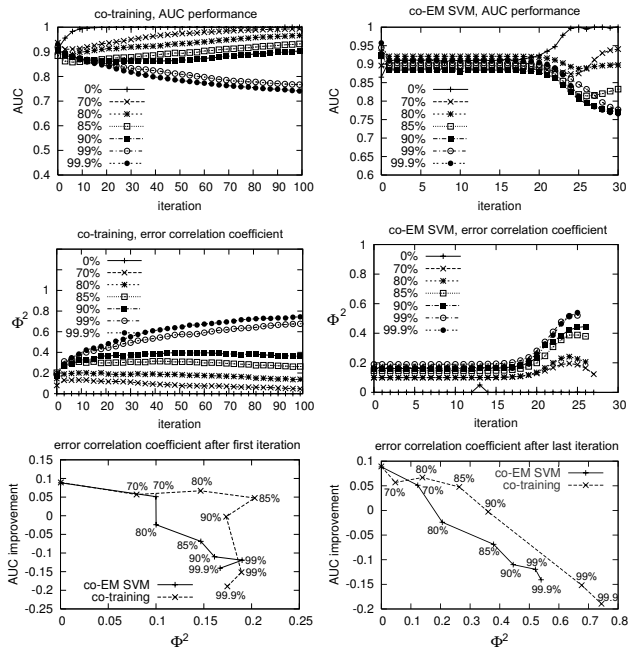

Figure 1: AUC/AUC improvements and Φ^2 values for the artificial data with varying dependencies p_{dep}.

which is just the definition of statistical independence again. □

PROPOSITION 2. *If $E_1 = E_2$ holds for all examples ($P(E_1 = E_2) = 1$), then $\Phi^2 = 1$.*

Proof. If $E_1 = E_2$ holds for all examples, it follows that $P(E_1 = i, E_2 = j) = 0$ for $i \neq j$. As a consequence the marginal distributions $P(E_k = i)$ are determined by the joint distribution $P(E_1 = i) = P(E_2 = i) = P(E_1 = i, E_2 = i)$. Using the substitutes $a = P(E_1 = 0)$ and $b = 1 - a = P(E_1 = 1)$, this allows us to rewrite Eq. 5 as Eq. 6, which immediately takes us to Eq. 7.

$$\Phi^2 = \frac{(a-a^2)^2}{a^2} + 2\frac{(0-ab)^2}{ab} + \frac{(b-b^2)^2}{b^2} \quad (6)$$

$$= (1-a^2) + 2ab + (1-b)^2 = 1 \quad (7)$$

□

Note that, if we do not restrict E_1 and E_2 to binary random variables, $m \cdot \Phi^2$ equals the well known χ^2 statistics, where m denotes the test sample size. In contrast to χ^2, the scale of Φ^2 is independent of the test set size; its range of between 0 and 1 allows an intuitive interpretation as a measure of dependency.

6. EXPERIMENTS

Our experiments are based on the well known 20-newsgroups, Reuters-21578, and WebKB course data sets. Our implementations of co-training and the co-EM SVM are built into SVM^{light} [10]. We use linear kernels for all experiments. We address the following questions.

What is the relationship between the independence of the views, the error correlation coefficient, and their performance? In order to answer this question, we construct a data set with completely independent and compatible views. We use four of the 20 newsgroups: rec.auto, comp.graphics, sci.space, and talk.politics.misc and proceed as follows. After building tfidf vectors for each of the four categories, we generate positive examples by concatenating vectors x_1 from rec.auto with randomly drawn vectors x_2 from sci.space to construct multi-view examples (x_1, x_2). Negative examples are constructed analogously with x_1 from comp.graphics and x_2 from talk.politics.misc. This procedure guarantees views which are perfectly compatible (either group can be discriminated from the other) and independent (peers are selected randomly).

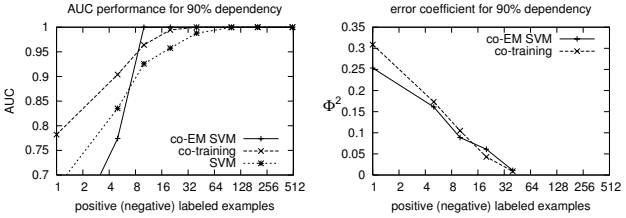

Figure 2: AUC and error correlation coefficient for increasing numbers of labeled data.

In order to add a controlled amount of dependence into the data set, we adapt an experimental setting of [18] and [15]. Each vector is a concatenation of attributes $x_1, \ldots, x_k$ (view V_1), and $x_{k+1}, \ldots, x_{2k}$ (View V_2). For each example, each attribute $k + i$ assumes the value of attribute i (as opposed to its original value) with probability p_{dep}. For $p_{dep} = 0$, the views V_1 and V_2 are perfectly independent. For $p_{dep} = 1$, the projections of each instance into either view are equal; the views are totally dependent. This procedure allows adding much stronger dependencies than the related procedure proposed by [15]. Figure 1, top row shows the AUC curves for co-training and the co-EM SVM with five positive and five negative labeled examples; the corresponding Φ^2 values are shown in the center row. All curves are averages of 20 runs of the focused algorithm, with distinct samples due to randomly drawn labeled examples. As expected, the performance of both algorithms decreases when the dependency of the views increases. Simultaneously, the actual Φ^2 values increase proportionally to the amount of added dependencies. We observe a clear relationship between the dependency of the views and Φ^2. Note that some Φ^2 curves end before the 30th iteration; here, at least one classifier of the 20 averaged co-EM SVM is always correct. In this case, Φ^2 is undefined.

What is the relation between the error correlation coefficient Φ^2 of the classifiers and the benefit of multi-view learning? We subtract the AUC values of the "vanilla" single-view SVM from the final AUC values of the multi-view hypothesis and plot the resulting AUC improvements against the initial and final (Figure 1, bottom row) Φ^2 values. A positive value indicates that multi-view learning is an improvement over single-view learning. We see a clear correlation between Φ^2 of the final – and even the Φ^2 value of the initial classifier – and the benefit of multi-view learning. This finding indicates that the initial Φ^2 value (that is measured on the basis of two runs of the SVM with only labeled data) carries substantial information on the potential impact of multi-view learning.

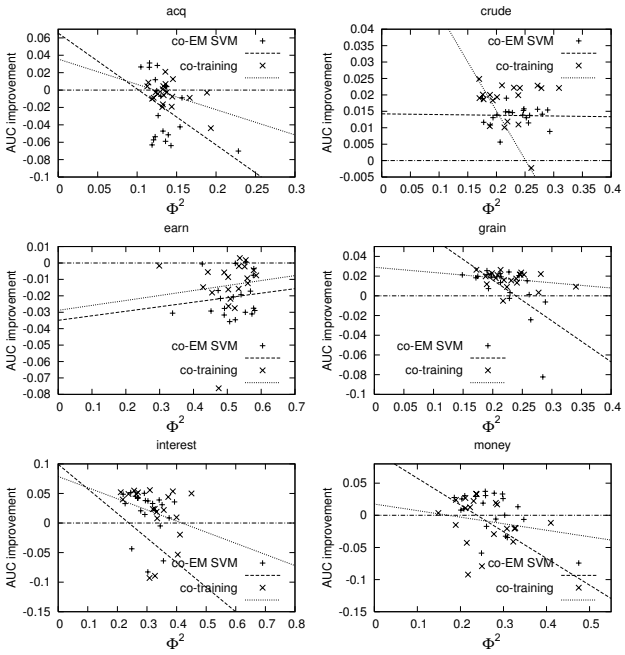

Figure 3: Initial Φ^2 and corresponding AUC improvements for the six most frequent categories of the Reuters data set. Feature splits were chosen randomly for each repetition.

If the dependency p_{dep} exceeds 90% for co-training and approximately 75% for the co-EM SVM and consequently Φ^2 exceeds 0.2 and 0.1, respectively, then multi-view learning is detrimental, compared to single-view learning. Except for dependencies exeeding 90%, co-training behaves more robustly than the co-EM SVM.

What is the relation between the number of labeled data and the error correlation coefficient? Figure 2 (left hand side) shows the AUC obtained by the "vanilla" SVM, the co-trained SVM, and the co-EM SVM over the labeled sample size. Figure 2 (right hand side) shows the development of Φ^2 while the labeled sample size is increased. All points are averages over 20 randomly drawn labeled samples with $p_{dep} = 0.9$. We see that the error correlation coefficient decreases as the sample size is increased. Co-EM SVM is most effective for small values of Φ^2, co-training more robust against larger Φ^2. As the labeled sample grows, the "vanilla" SVM approaches an AUC of 1.

Can we observe a similar correlation between Φ^2 and the benefit of multi-view learning for real-world text classification problems? We study the six most frequent categories of the Reuters-21587 problem. We split the attributes *at random* into two views, we use 1% of the 19,043 examples as labeled sample, the remaining instances serve as unlabeled data and hold-out set. For each category, we conduct 20 repetitions with distinct random attribute splits and distinct randomly drawn labeled samples. Figure 3 shows the measured AUC improvements over the measured error correlation coefficient for co-training and co-EM; the interpolating lines are generated by linear regression.

Again, the correlation between Φ^2 and the AUC improvement is clearly visible. For one of the classes ("earn"), we observe a negative correlation. Here, all measured Φ^2 values are above 0.5; we believe that the lack of examples with small Φ^2 values causes an inaccurate result of the linear regression. These results show that even when we split the attributes at random, multi-view learning improves the performance provided that the error correlation coefficient of the initial machines is small. In addition, we see that attribute splits which minimize Φ^2, on average also maximize the impact of multi-view learning. This result paves the way to algorithms that automatically split the available attributes such that the expected benefit of multi-view learning is obtained.

Table 3: Results for the course problem.

Method	Error rate
naive Bayes	13.0%
co-trained NB	5.0%
co-EM NB (65 labeled ex.)	$5.08 \pm 0.7\%$
SVM	$10.39\% \pm 0.7\%$
co-trained SVM	$4.45\% \pm 0.9\%$
co-EM SVM	$\mathbf{0.99\% \pm 1.3\%}$

How do the co-trained and the co-EM Support Vector Machine perform on real multi-view data? We conduct experiments using the WebKB course data set. Here, the task is to predict whether a web page is a course home page, based on the content (view V_1) and the anchor texts occurring in the inbound hyperlinks (view V_2). Figure 4 shows the AUC performance of co-training and co-EM, depending on the number of labeled and unlabeled data. Co-EM consistently outperforms co-training and the "vanilla" SVM.

For this data set, several results using 3 positive and 9 negative labeled examples have been published, based on co-trained and co-EM naive Bayes [1, 17, 15]. We compare our observations to these published findings. Table 3 summarizes the results. After 100 rounds, the co-trained SVM achieves an error of 4.45% while the co-EM SVM outperforms all other support vector algorithms significantly with an error rate of 0.99%. Since 3 positive and 9 negative examples do not reflect the true prior distribution we used the natural ratio of 2 positive and 8 negative examples for shifting the decision hyperplane.

7. CONCLUSION

Our paper focused on the question: *for which learning problems is multi-view learning beneficial, how can these instances of learning problems and appropriate partitionings of the attributes into views be identified.* We introduced the *error correlation coefficient* Φ^2 and discussed its properties. We conducted experiments focusing on the benefit of the co-trained SVM and co-EM SVM and the correlation between this benefit and the error correlation coefficient. We studied a range of problems, including problems with controlled violations of the independence assumption as well as the Reuters and WebKB course text classification data sets. Based on the results of these experiments, we can draw a number of conclusions.

1. Multi-view learning can significantly improve the performance of a classifier, even when the attributes of a single view problem are artificially split into two views.

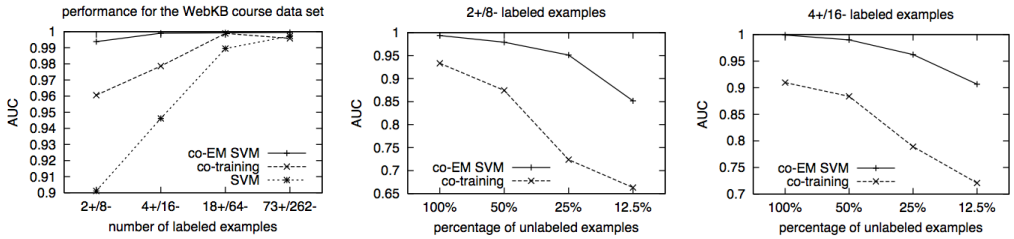

Figure 4: AUC performance of co-training and co-EM SVM for the WebKB course data set with increasing numbers of labeled (left) and unlabeled examples (middle and right).

On average, multi-view learning is beneficial when the error correlation coefficient of the initial classifiers is small; ideally, below 0.2.

2. The error correlation coefficient of the initial classifiers can be measured easily; two runs of the SVM are required, using only the labeled data. This paves the way to algorithms which first split the available attributes to minimize the error correlation coefficient, and then apply multi-view learning.

3. Our controlled experiments with the semi-artificial 20 newsgroups data set have revealed that the co-trained SVM is more robust to violations of the independence assumption than the co-EM SVM.

4. Our experiments on the course data set have shown that the co-EM SVM outperforms all other single- and multi-view approaches on this natural multi-view problem. Furthermore, the multi-view SVM substantially outperforms multi-view naive Bayes.

8. REFERENCES

[1] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the Conference on Computational Learning Theory*, pages 92–100, 1998.

[2] A. Bradley. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159, 1997.

[3] U. Brefeld, P. Geibel, and F. Wysotzki. Support vector machines with example dependent costs. In *Proceedings of the European Conference on Machine Learning*, 2003.

[4] U. Brefeld and T. Scheffer. Co-em support vector learning. In *Proceedings of the International Conference on Machine Learning*, in print.

[5] M. Collins and Y. Singer. Unsupervised models for named entity classification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 1999.

[6] D. Cooper and J. Freeman. On the asymptotic improvement in the outcome of supervised learning provided by additional nonsupervised learning. *IEEE Transactions on Computers*, C-19:1055–1063, 1970.

[7] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society B*, 39, 1977.

[8] F. Denis, A. Laurent, R. Gilleron, and M. Tommasi. Text classification and co-training from positive and unlabeled examples. In *ICML Workshop on the Continuum from Labeled to Unlabeled Data*, 2003.

[9] R. Ghani. Combining labeled and unlabeled data for multiclass text categorization. In *Proceedings of the International Conference on Machine Learning*, 2002.

[10] T. Joachims. Transductive inference for text classification using support vector machines. In *Proceedings of the International Conference on Machine Learning*, pages 200–209, 1999.

[11] S. Kiritchenko and S. Matwin. Email classification with co-training. Technical report, University of Ottawa, 2002.

[12] M. Kockelkorn, A. Lüneburg, and Tobias Scheffer. Using transduction and multi-view learning to answer emails. In *Proceedings of the European Conference on Principle and Practice of Knowledge Discovery in Databases*, 2003.

[13] A. McCallum and K. Nigam. Employing em in pool-based active learning for text classification. In *Proceedings of the International Conference on Machine Learning*, 1998.

[14] D. Mladenic. Learning word normalization using word suffix and context from unlabeled data. In *Proceedings of the International Conference on Machine Learning*, pages 427–434, 2002.

[15] I. Muslea, C. Kloblock, and S. Minton. Active + semi-supervised learning = robust multi-view learning. In *Proceedings of the International Conference on Machine Learning*, pages 435–442, 2002.

[16] I. Muslea, C. Kloblock, and S. Minton. Adaptive view validation: A first step towards automatic view detection. In *Proceedings of the International Conference on Machine Learning*, pages 443–450, 2002.

[17] K. Nigam and R. Ghani. Analyzing the effectiveness and applicability of co-training. In *Proceedings of Information and Knowledge Management*, 2000.

[18] Kamal Nigam, Andrew K. McCallum, Sebastian Thrun, and Tom M. Mitchell. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39(2/3), 2000.

[19] F. Provost, T. Fawcett, and R. Kohavi. The case against accuracy estimation for comparing inductive algorithms. In *Proceedings of the International Conference on Machine Learning*, pages 445–453, 1998.

[20] M. Seeger. Learning with labeled and unlabeled data. (technical report, University of Edinburgh, 2001.