Multi Instance Multi Label

Lorenzo Niccolai

lorenzo.niccolai3@stud.unifi.it

Fabio Vittorini

fabio.vittorini@stud.unifi.it

Machine Learning

University of Florence, Department of Information Engineering

July 16, 2017

Overview

Classification problem

Goal: To produce a classifier able to decide whether an object belongs to one or more classes.

Idea: Supervised Learning: Given a dataset of already classified examples, the classifier *learns* a function that solves classification problem.

Notation

Dataset is set of classified examples:

$$D = \{(X_i, Y_i) |\ i \in [1, n]\}$$

$$X_i \in \mathbb{R}^f$$

$$Y_i \in \{-1, +1\}$$

- A vector $x \in \mathbb{R}^f$ represents an object using f *relevant* features
- A number $y \in \{-1, +1\}$ indicates wether the example belong to target class

While learning the target function, the dataset is divided in *training set* and *test set*.

- The idea is to compute the *maximum-margin hyperplane* $w^T x + b$ which best separates positive examples from negative examples
- Samples on the margin are called the *support vectors*

Optimization problem is:

$$argmin_w \frac{1}{2}||w||^2$$

$$Y_i(w^T X_i + b) \geq 1 \; \forall i \in [1, n]$$

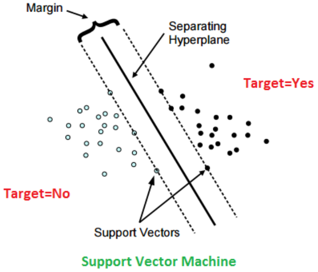

Figure 1: *Solution of maximum-margin hyperplane*

SVM with slacks

- The examples may not be linearly serparable and so the problem would not have any solutions because constraints are not satisfied. Then we introduce slack variables ξ

Optimization problem becomes:

$$argmin_{w,\xi}\frac{1}{2}||w||^2 + C\sum_{i=1}^{n}\xi_i$$

$$y_i(w^T x_i + b) \geq 1 - \xi_i \ \forall i \in [1, n]$$

$$\xi_i \geq 0 \ \forall i \in [1, n]$$

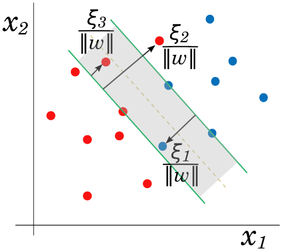

Figure 2: *Solution with slacks*

Multi instance classification

MODIFICARE Motivation:

- Sometimes a complex item can be well represented by a set of *instances*
- A single instance may belong or not to a class or *label* (positive or negative)
- An example, or *bag*, is positive if at least one of its instances is positive, where as a negative bag consists of only negative instances
- A label is provided for the entire bag, not to instances
- We have a *semi-supervised learning* problem

[1]

Notation

Dataset is now a set of bags, where each bag is a set of instances:

$$D = \{(X_i, Y_i)|\ i \in [1, n]\}$$

$$X_i = \{x_{i,k}|\ k \in [1, k_i], x_{i,k} \in \mathbb{R}^f\}$$

Notice that each bag can be made of any number of instances, but every instance has a fixed number of features f.

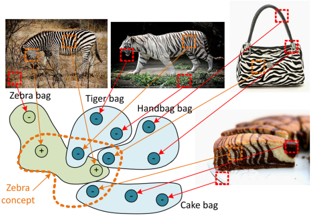

Figure 3: *Example of images instances refer to "zebra concept"*

SIL

The first naive approach makes the following label assignment:

- If an instance belongs to a negative bag, sets its label to -1
- If an instance belongs to a positive bag, sets its label to $+1$

The resulting problem can be solved using a regular SVM, treating each instance as a whole document.

Using this approach makes almost useless multi-instance formulation.

mi-SVM

Instances label assignment:

- If an instance belongs to a negative bag we can say that its label is -1
- If an instance belongs to a positive bag we don't know for sure its label

This leads to 2 new constraints in SVM problem:

$$y_{i,k} = -1 \ \textit{if} \ \ Y_i = -1$$

$$\sum_{k=1}^{k_i} \frac{y_{i,k} + 1}{2} \geq 1 \ \textit{if} \ \ Y_i = +1$$

mi-SVM

Our SVM problem becames the following:

$$min_Y \, min_{w,\xi} \frac{1}{2}||w||^2 + C \sum_{i=1}^{n} \xi_i$$

$$y_{i,k}(w^T x_{i,k} + b) \geq 1 - \xi_i \; \forall i \in [1, n], k \in [1, k_i]$$

$$\xi_i \geq 0 \; \forall i \in [1, n]$$

$$y_{i,k} = -1 \; if \; Y_i = -1$$

$$\sum_{k=1}^{k_i} \frac{y_{i,k} + 1}{2} \geq 1 \; if \; Y_i = +1$$

That is an intractable mixed optimization problem

mi-SVM algorithm

A feasible algorithm that finds a non optimal solution is the following:

MI-SVM(X, Y)

$y_{i,k} = -1$ *if* $Y_i = -1$
$y_{i,k} = +1$ *if* $Y_i = +1$
do
    Solve regular SVM finding w, b
    $y_{i,k} = sign(w^T x_{i,k} + b)$ *if* $Y_i = +1$
    Adjust each positive bag to satisfy constraints
while ($y_{i,k}$ change)

MI-SVM

This approach uses directly the dataset in its bag form:

$$argmin_{w,\xi} \frac{1}{2}||w||^2 + C\sum_{i=1}^{n}\xi_i$$

$$y_i(max_k w^T x_{i,k} + b) \geq 1 - \xi_i \ \forall i \in [1, n]$$

$$\xi_i \geq 0 \ \forall i \in [1, n]$$

This is possible by selecting a *witness* from each bag instance.

MI-SVM algorithm

A feasible algorithm that finds a solution is the following:

$\text{MI-SVM}(X, Y)$

```
x̄_i = avg(x_{i,k}) ∀x_{i,k} ∈ X_i positive bag
do
    Assign ᾱ_i ∈ [0, C] to each x̄_i
    Assign α_{i,j} with ∑_{j=1}^{k_i} α_{i,j} ∈ [0, C] ∀x_{i,k} ∈ X_i negative bag
    Solve regular SVM finding w, b
    Find new x̄_i by selecting the best one for each positive bag
while (witnesses change)
```

Other common frameworks for MI

In addition to these methods we can cite:

- **Diverse density (DD)**: It computes a probabilistic measure searching a *concept point* which lies close to at least one instance of every positive bag and far away from instances of negative bags [7]
- **EM-DD**: It combines *EM* [4] with the extended *DD* algorithm [11]
- **Citation kNN**: It uses *minimum Hausdorff distance* to measure the distance between bags and allows *kNN* algorithms to be adapted to the MI problem [10]
- **MIL Random forest (MIL RF)**: It uses *decision trees* to form *Random Forests* to form a classifier [6] [3]

[2]

Citation kNN

The notation *citation* means that the method takes not only into account the neighbors of a bag b (**references**) but also the bags that count b as neighbor (**citers**).

- *References* are computed as *R-nearest neighbors* according to the Hausdorff distance.
  The *minimum Hausdorff distance* is defined as:

$$Dist(A, B) = \min_{\substack{1 \leq i \leq n \\ 1 \leq j \leq m}} (Dist(a_i, b_j)) = \min_{a \in A} \min_{b \in B} \|a - b\|$$

  where A and B are two bags, a_i and b_j are instances from each bag.
- *Citers* are computed as *C-nearest citers* of a bag b in BS:

$$Citers(b, C) = b_i \mid Rank(b_i, b) \leq C, b_i \in BS$$

  where $Rank(a, b)$ is a rank function according to the similarity of examples coming from the same bag.

Citation kNN

Let $p_b = R_{b,p} + C_{b,p}$ the number of positive references and positive citers of the bag b and $n_b = R_{b,n} + C_{b,n}$ the same for the negatives.

The *Citation-KNN* is the *KNN* algorithm in which p_b and n_b are computed by using the Hausdorff distance and classification is defined as:

$$y_b = \begin{cases} positive, & \text{if } p_b > n_b \\ negative, & \text{otherwise} \end{cases}$$

where y_b is the class of the bag b

[10]

Multi label classification

Motivation:

- Sometimes a complex item can be well represented by a set of *labels*
- Helps single label classification when the concept is more complicated or general

Solutions [9]:

- Problem transformation
- Algorithm adaptation

Notation

A set of labels $L = \{y_1, y_2, ...y_l\}$ is given.
Each object contained in the dataset is associated with a set of labels:

$$D = \{(X_i, Y_i | i \in [1, n]\}$$

$$X_i \in \mathbb{R}^f$$

$$Y_i = \{y_{i,h} | h \in [1, h_i], y_{i,h} \in L, h_i \leq l\}$$

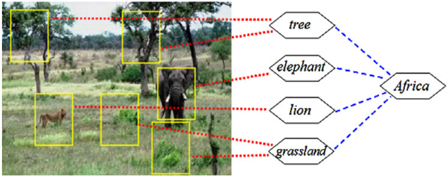

Figure 4: *Multi label example*

Problem transformation

Attempt to convert the multilabel problem in a regular binary task.
Two lossy methods:

- Randomly discard each label information except one from each instance
- Remove instances that have actually more than one label

Other solutions:

- Train a binary classifier for each existing combination of labels
- Train a binary classifier for each label (used in this work)

An algorithm adaptation approach

The idea is to focus on ranking rather than binary classification [5]

- Train a classifier $f_l : X \to \mathbb{R}$ for each label
- For each test example sort label list according to predicted rank
- Set size prediction feeding dataset and thresholds $t(X_i)$ to a classifier

$$t(X_i) = argmin_t\{k \in Y_i \ s.t. \ f_k(X_i) \leq t\} + \{k \in \bar{Y}_i \ s.t. \ f_k(X_i) \geq t\}$$

- Take the best labels according to set size prediction:

MIML problems combine motivations of multi instance and multi label ones.
Given a set of labels $L = \{y_1, y_2, ...y_l\}$

$$X_i = \{x_{i,k} | k \in [1, k_i], x_{i,k} \in \mathbb{R}^f\} \, Y_i = \{y_{i,h} | h \in [1, h_i], y_{i,h} \in L, h_i \leq l\}$$

$$D = \{(X_i, Y_i) | i \in [1, n]\}$$

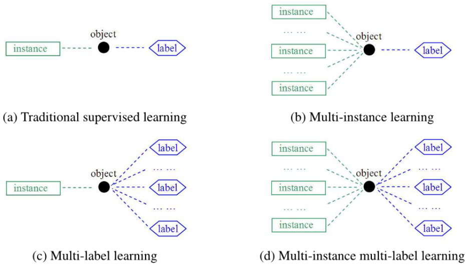

Figure 5: *Different learning frameworks*

[10]

SVM Solution

To allow regular SVMs to solve this problem, we use *problem transformation*.

There are 2 possibilities:

- MIML → MISL → SISL (used in this work)
- MIML → SIML → SISL

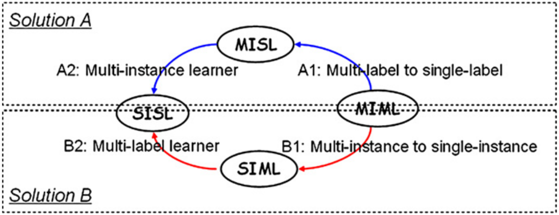

Figure 6: *Two possible solutions to implement MIML*

Multi label to single label

Excluding lossy approaches, the idea is to train a multi-instance (single label) classifier for each label.
Given a MIML dataset $D = \{(X_i, Y_i) | i \in [1, n]\}$ we produce l datasets as follows:

$$D_{y_j} = \{(X_i, Y_{y_j}) | i \in [1, n]\} \ \forall j \in [1, L]$$

Where

$$Y_{y_j} = \begin{cases} +1 \ if \ \ y_j \in Y_i \\ -1 \ otherwise \end{cases}$$

Then we train L regular multi-instance SVMs and collect their results.

Multi instance to single instance

Given one of MISL datasets produced at previous step, we compared the 3 methods previously exposed:

- SIL
- MI-SVM
- mi-SVM

They all use a standard SISL SVM as subroutine.

Our work

The aim of our work is to replicate a part of the results of [12] using the **MIML framework** and compare the different metrics.

- We focused on the *text categorization* using text documents (*bags*) belonging to categories (*labels*)
- We choose to use the MimlBoost solution using multi-instance learning as the bridge between MIML and SISL
- Bag of words approach to Reuters-21578 dataset
- Multi instance tasks solved with SIL, MI-SVM and mi-SVM approaches

Test

Documents selection:

1. Removed every document with 0 labels
2. Removed short documents (less than 30 words)
3. Removed randomly documents with 1 label to obtain 2000 examples

Dictionary creation:

1. Performed stemming
2. Removed stopwords
3. Removed rare words keeping 2% of them (about 210)

Multi instance data

1. Splitted documents in passages of 50 words max
2. Removed empty instances (according to dictionary)

Evaluation criteria

Four criteria are used for performance evaluation:

- **hamming loss**:

$$hloss_S(h) = \frac{1}{p}\sum_{i=1}^{p}\frac{1}{|\mathcal{Y}|}|h(X_i)\Delta Y_i|$$

- **one-error**:

$$one - error_S(h) = \frac{1}{p}\sum_{i=1}^{p}[[\arg\max_{y\in\mathcal{Y}} h(X_i, y)] \notin Y_i]$$

- **coverage**:

$$coverage_S(h) = \frac{1}{p}\sum_{i=1}^{p}\max_{y\in Y_i} rank^h(X_i, y) - 1$$

- **ranking loss**:

$$rloss_S(h) = \frac{1}{p}\sum_{i=1}^{p}\frac{1}{|Y_i|\cdot|\bar{Y}_i|}|\{(y_1, y_2) \in Y_i \times \bar{Y}_i \ s.t. \ h(X_i, y_1) \leq \ h(X_i, y_2)\}|$$

More metrics

Other metrics used by reference article:

- **average precision**:

$$avgprec_S(h) = \frac{1}{p}\sum_{i=1}^{p}\frac{1}{|Y_i|}\sum_{y\in Y_i}\frac{|\{y' \mid rank^h(X_i,y') \leq rank^h(X_i,y),\ y' \in Y_i)\}|}{rank^h(X_i,y)}$$

- **average recall**:

$$avgrecl_S(h) = \frac{1}{p}\sum_{i=1}^{p}\frac{|\{y \mid rank^h(X_i,y) \leq |h(X_i)|,\ y \in Y_i)\}|}{|Y_i|}$$

- **average F1**:

$$avgF1_S(h) = \frac{2 \times avgprec_S(h) \times avgrecl_S(h)}{avgprec_S(h) + avgrecl_S(h)}$$

[8]

Results

Algorithms	Metrics hloss	one-error	coverage	rloss	aveprec	averecl	aveF1
MimlBoost	.053±.004	.094±.014	.387±.037	.035±.005	.937±.008	.792±.010	.858±.008
MimlSvm	.033±.003	.066±.011	.313±.035	.023±.004	.956±.006	.925±.010	.940±.008
$MimlSvm_{mi}$	.041±.004	.055±.009	.284±.030	.020±.003	.965±.005	.921±.012	.942±.007
MimlNn	.038±.002	.080±.010	.320±.030	.025±.003	.950±.006	.834±.011	.888±.008
AdtBoost.MH	.055±.005	.120±.017	.409±.047	N/A	.926±.011	N/A	N/A
RankSvm	.120±.013	.196±.126	.695±.466	.085±.077	.868±.092	.411±.059	.556±.068
MlSvm	.050±.003	.081±.011	.329±.029	.026±.003	.949±.006	.777±.016	.854±.011
$Ml - knn$	.049±.003	.126±.012	.440±.035	.045±.004	.920±.007	.821±.021	.867±.013
SIL	.072±.002	.129±.017	.104±.036	.025±.004	.865±.012	.797±.020	.829±.016
MISVM	.134±.004	.015±.008	.666±.054	.214±.011	.636±.019	.425±.008	.509±.011
$mi - SVM$							

Considerations

- Scores are quite low, but *one-error* is low even if it's not a good metric for evaluating multilabel performance
- Selected labels' frequencies are 520, 434, 283, 222, 223, 220, 187 over 2000 documents.
- Test repeated for best 2 labels with following results

Algorithms	Metrics						
	hloss	one-error	coverage	rloss	aveprec	averecl	aveF1
SIL	.072±.002	.129±.017	.104±.036	.025±.004	.865±.012	.797±.020	.829±.016
MISVM	.134±.004	.015±.008	.666±.054	.214±.011	.636±.019	.425±.008	.509±.011
$mi - SVM$							

References I

[1] Stuart Andrews, Ioannis Tsochantaridis, and Thomas Hofmann. Support vector machines for multiple-instance learning. In *Advances in neural information processing systems*, pages 577–584, 2003.

[2] Sanghamitra Bandyopadhyay, Dip Ghosh, Ramkrishna Mitra, and Zhongming Zhao. Mbstar: multiple instance learning for predicting specific functional binding sites in microrna targets. *Scientific reports*, 5, 2015.

[3] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[4] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38, 1977.

[5] André Elisseeff and Jason Weston. A kernel method for multi-labelled classification. In *Advances in neural information processing systems*, pages 681–687, 2002.

References II

[6] Tin Kam Ho. Random decision forests. In *Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on*, volume 1, pages 278–282. IEEE, 1995.

[7] Oded Maron and Tomás Lozano-Pérez. A framework for multiple-instance learning. In *Advances in neural information processing systems*, pages 570–576, 1998.

[8] Mohammad S Sorower. A literature survey on algorithms for multi-label learning. *Oregon State University, Corvallis*, 18, 2010.

[9] Grigorios Tsoumakas and Ioannis Katakis. Multi-label classification: An overview. *International Journal of Data Warehousing and Mining*, 3(3), 2006.

[10] Jun Wang and Jean-Daniel Zucker. Solving multiple-instance problem: A lazy learning approach. 2000.

References III

[11] Qi Zhang and Sally A Goldman. Em-dd: An improved multiple-instance learning technique. In *Advances in neural information processing systems*, pages 1073–1080, 2002.

[12] Zhi-Hua Zhou, Min-Ling Zhang, Sheng-Jun Huang, and Yu-Feng Li. Multi-instance multi-label learning. *Artificial Intelligence*, 176(1):2291–2320, 2012.