Semi-supervised Image Segmentation

Gergana Angelova Lazarova
PhD Student at Sofia University, Bulgaria
part-time teaching assistant

Semi-Supervised Learning (SSL)

- Labeled examples

$$D_1 = \{(x_i, y_i)\}_{i=1}^{nl}$$

- Unlabeled examples

$$D_2 = \{x_j\}_{j=1}^{nu}$$

- Data

$$D = D_1 \cup D_2$$

- Usually few labeled examples are present, but we have access to large amounts of unlabeled examples
- Unlabeled examples are cheap to collect

Semi-Supervised Image Segmentation

- Problem formulation: Image Segmentation
  - partitioning a digital image into multiple segments
  - Classification - assign a label to every pixel in an image
- Approach: Semi-Supervised Learning
  - Few labeled pixels – a teacher labels them
  - Use unlabeled pixels for learning
- An innovative approach for semi-supervised image segmentation is proposed
  - modification of the standard co-training algorithm

Multi-view learning

- Multiple sources of data
- X = (X1, X2), X1 and X2 represent feature sets
- Combining the results of the two sources
- Examples
  - People recognition – combining face recognition, voice recognition, etc.
  - Web-page classification – words on the web pages, hyperlinks pointing to the web pages
  - Image Segmentation – RGB values, coordinates of the pixels

Co-Training – original algorithm

- Two views: $X = (X1, X2)$
- Each view (set of features) is sufficient for learning
- The two views (feature sets of each instance) are conditionally independent given the class.
  - $P(X1|Y, X2) = P(X1|Y)$
  - $P(X2|Y, X1) = P(X2|Y)$

Co-Training – original algorithm

- Learn L1 using U1, Learn L2 using U2
- Label all unlabeled examples
  - Probabilistically label all unlabeled examples using L1. Add L1's most confident examples to U2
  - Probabilistically label all unlabeled examples using L2. Add L2's most confident examples to U1
- Go to 1 until there are no more unlabeled examples or some other stop criterion is met

Multi-View Teaching Algorithm (MTA)

- Two views: X = (X1, X2)
- Modification of the standard Co-Training Algorithm
- One of the views is weaker than the other and may worsen the final result
- Improve only the weaker view and combine the results
- The two views (feature sets of each instance) are conditionally independent given the class.
  - $P(X1|Y, X2) = P(X1|Y)$
  - $P(X2|Y, X1) = P(X2|Y)$

Multi-View Teaching Algorithm (MTA)

- Learn L1 based on view1.
- Add more labeled examples to L2
  - For each example *xi* calculate its most probable classification
  - For each class *yj* find the most confident examples and if they exceed some threshold add them to U2
  - Learn L2 based on view 2, using also the new labeled by L1 examples
- Combine the results of the two views

Combining the Views

- Multiply the results of the separate learners

$$P(y_j \mid x_i) = P(y_j \mid x_i, \theta_1) P(y_j \mid x_i, \theta_2)$$

$$\arg\max_{y_j} P(y_j \mid x_i, \theta_1) P(y_j \mid x_i, \theta_2) =$$
$$\arg\max_{y_j} \log P(y_j \mid x_i, \theta_1) + \log P(y_j \mid x_i, \theta_2)$$

Learners/Classifiers used

- Naive Bayes Classifer
- Supervised Classifier based on Multivariate Normal Distribution
- Aim:
  - Compare Semi-supervised MTA, based on Naive Bayes Classifer to its supervised equivalent
  - Compare Semi-supervised MTA, based on Multivariate Normal Distribution to its supervised equivalent

Naive Bayes Classifier

- This classifier is a simple probabilistic classifier and relies on the preposition that the attributes are independent.

$$P(y_j)P(x_i \mid y_j) = P(y_j)\prod_{k=1}^{m} P(a_k \mid y_j)$$

- In order to classify new examples it chooses the hypothesis that is most probable. The corresponding classifier is the function *f** defined as:

$$f^* = \arg\max_{y_j} P(y_j)P(x_i \mid y_j)$$

Supervised learning, based on Multivariate Normal Distribution

- The multivariate normal distribution is often used to describe any set of correlated real-valued random variables each of which clusters around a mean value.

$$P(D\mid\theta)=\prod_{i=1}^{l}P(x_i,y_i\mid\theta)=\prod_{i=1}^{l}P(y_i\mid\theta)P(x_i\mid y_i,\theta)$$

$$P(y_i\mid x_i,\theta)=\frac{P(y_i)P(x_i\mid\theta,y_i)}{P(x_i)}=\frac{P(y_i)N(x_i,\mu_y,\Sigma_y)}{P(x_i)}$$

$$N(x,\mu_y,\Sigma_y)=\frac{1}{(2\pi)^{\frac{D}{2}}\,|\Sigma y|^{\frac{1}{2}}}e^{\frac{-(x-\mu_y)^T\Sigma_y^{-1}(x-\mu_y)}{2}}$$

Experimental Framework

- Monte Carlo cross-validation
- Construction of the training and test sets:
  - The training set consists of a fraction of labeled examples from the original dataset. Randomly a small amount of pixels are chosen, they are added to D1. The rest of the instances have their classifications removed. These unlabeled examples are added to D2. A final training set is constructed: D = D1 U D2.
  - The test set contains all the examples in D2.

Experimental Results

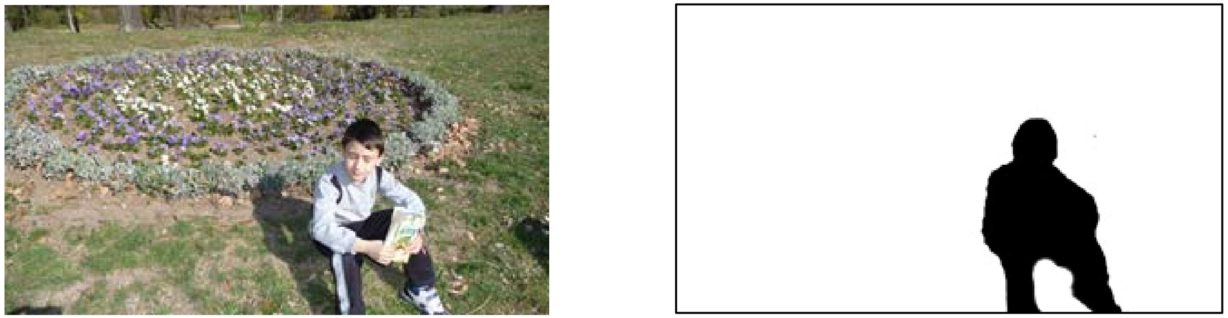


Fig. 1. (a) - original image, (b) – desired segmentation

Fig. 2. (a) - original image, (b) – desired segmentation

Experimental Results

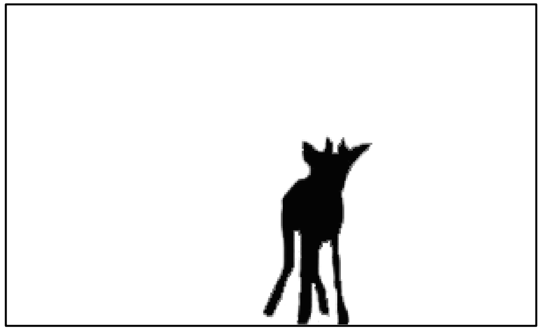


Fig. 3. (a) - original image, (b) – desired segmentation

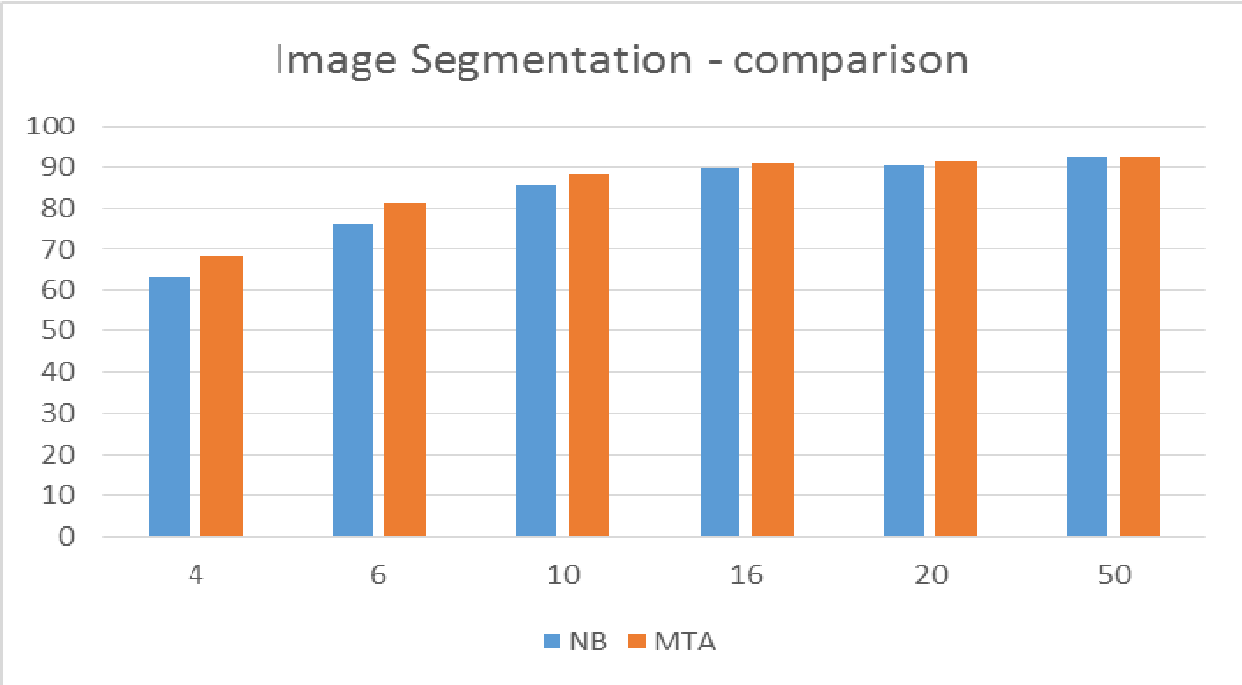
- Multi-view teaching algorithm based on Naïve Bayes Classifiers(MTA) vs Supervised Naïve Bayes Classifier (NB)
- Multi-view teaching algorithm based on MND-SL(MTA-MD) vs Supervised MND - SL (MD)

MTA vs. Supervised Naïve Bayes Classifier

Algorithm	4	6	10	16	20	50
NB	63.30%	76.23%	85.44%	89.57%	90.33%	92.37%
MTA	68.62%	81.30%	88.14%	90.74%	91.24%	92.51%

Table 1. Comparison of the two algorithms, based on the number of labelled pixels (Image 1)

MTA vs. Supervised Naïve Bayes Classifier



MTA vs NB – classification accuracy comparison, based on the amount of labeled examples

MTA vs. Supervised Naïve Bayes Classifier

	MTA	NB
Image 1	90.74%	89.57%
Image 2	80.76%	78.82%
Image 3	90.10%	89.12%

Table 2. Comparison of the two algorithms, based on 16 initial labeled examples

MND-SL(MTA-MD) vs Supervised MND - SL (MD)

	MTA-MD	MD
Image 1	84.36%	79.22%
Image 2	79.14%	73.74%
Image 3	86.02%	80.18%

Table 3. Comparison of the two algorithms, 16 initial labeled examples

Thank you!

Благодаря за вниманието!

どうもありがとうございます！