The Application Of Active PU Learning In Content Based Image Retrieval

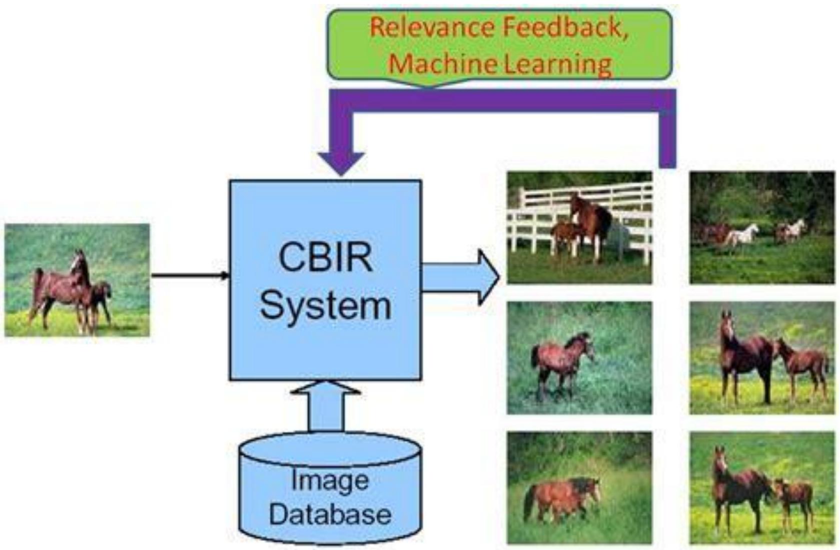
Reported by: Cai Jia-Jia

Outline

- Introduction
  - Relevance Feedback in Content Based Image Retrieval
- The Methods
  - SSAIRA: (Semi-Supervised Active Image Retrieval with Asymmetry)
  - Adaptive threshold
- Try

RF in CBIR

- Relevance Feedback in content based image retrieval



Characteristic:
- Small sample
- Asymmetrical training sample
- Real time requirement

Connection with PU Learning

- Only positive example(s) and unlabeled data
- Imbalance between positive and negative
- RF: active learning

SSAIRA

Base Learner:

$$L_i(\boldsymbol{x}, \mathcal{P}_i, \mathcal{N}_i) = \left(\sum_{\boldsymbol{y} \in \mathcal{P}_i} \frac{Sim_i(\boldsymbol{x}, \boldsymbol{y})}{|\mathcal{P}_i| + \varepsilon} - \sum_{\boldsymbol{z} \in \mathcal{N}_i} \frac{Sim_i(\boldsymbol{x}, \boldsymbol{z})}{|\mathcal{N}_i| + \varepsilon} \right) / \mathcal{Z}_{norm} \qquad (1)$$

Note that similar matrix can be computed offline.

Real time requirement.

Co-training:

$Getfeedback(\mathcal{P}^*, \mathcal{N}^*)$
$\mathcal{P} \leftarrow \mathcal{P} \cup \mathcal{P}^*; \mathcal{N} \leftarrow \mathcal{N} \cup \mathcal{N}^*; \mathcal{U} \leftarrow \mathcal{U} - (\mathcal{P}^* \cup \mathcal{N}^*)$

for $i \in \{1..2\}$ **do**

$\mathcal{N}$

$\mathcal{P}_i \leftarrow \mathcal{P}$
$\mathcal{N}_i \leftarrow \mathcal{N} \cup \{\arg\min_{\boldsymbol{x} \in \mathcal{U}} L_{(3-i)}(\boldsymbol{x}, \mathcal{P}, \boxed{Generalize(\mathcal{N}, \mathcal{U}, k)})\}$

Negative is too diverse.

Active Learning

Label images on which the learners disagree the most, or both learners are with low confidence.

$\mathcal{N}$

$$\textbf{for } \boldsymbol{x} \in \mathcal{U} \textbf{ do } Rank(\boldsymbol{x}) \leftarrow \frac{1}{Z_{norm}} \sum_{i \in \{1..2\}} L_i(\boldsymbol{x}, \mathcal{P}_i, \boxed{Generalize(\mathcal{N}_i, \mathcal{U}, k})$$

Query set

Retrieval result

$$Pool \leftarrow \emptyset;\ Result \leftarrow \emptyset$$

$$\textbf{for } i \in \{1..poolsize\} \textbf{ do } Pool \leftarrow Pool \cup \{\arg\min_{\boldsymbol{x} \in \mathcal{U}} Abs(Rank(\boldsymbol{x}))\}$$

$$\textbf{for } i \in \{1..resultsize\} \textbf{ do } Result \leftarrow Result \cup \{\arg\max_{\boldsymbol{x} \in \mathcal{U}} Rank(\boldsymbol{x})\}$$

Query selection

Asymmetrical training sample

- Positive examples can be regarded as belonging to the same relevant class, but the negative examples may belong to different irrelevant classes.
- Assume each negative example is a representative of a potential semantic class.
- The k-nearest neighboring unlabeled examples are identified for each negative example and then the feature vectors of these k+1 examples are averaged to derive a virtual example, which is used by the learners instead of the original negative example.

Generate **representative** negative example

Table II. Pseudo-code describing the GENERALIZE function

GENERALIZE($\mathcal{D}_n$, $\mathcal{D}_u$, k)

Input: $\mathcal{D}_n$: A data set whose elements are to be generalized
$\mathcal{D}_u$: A data set in which neighbors are to be identified
k: Number of neighbors used in generalizing

$\mathcal{D}^* \leftarrow \emptyset$
for $\boldsymbol{x} \in \mathcal{D}_n$ **do**
$\mathcal{D}' \leftarrow Neighbors(\boldsymbol{x}, \mathcal{D}_u, k)$ %% $\mathcal{D}'$ stores the k-nearest neighbors of $\boldsymbol{x}$ in $\mathcal{D}_u$
$\mathcal{D}' \leftarrow \mathcal{D}' \cup \{\boldsymbol{x}\}$
$\boldsymbol{x}' \leftarrow Ave(\mathcal{D}')$ %% $\boldsymbol{x}'$ is the average feature vector of the feature vectors in $\mathcal{D}'$
$\mathcal{D}^* \leftarrow \mathcal{D}^* \cup \{\boldsymbol{x}'\}$

Output: $\mathcal{D}^*$

Pseudo-code

Table I. Pseudo-code describing the SSAIRA method

SSAIRA(*query*, *DB*, L_1, L_2, *k*, *poolsize*, *resultsize*)

Input: *query*: User query
DB: Image database
L_i ($i \in \{1..2\}$): Learners
k: Number of neighbors used in generalizing negative examples
poolsize: Number of images in the pool
resultsize: Number of images to be returned

$\mathcal{P} \leftarrow \{query\}$; $\mathcal{N} \leftarrow \emptyset$; $\mathcal{U} \leftarrow DB$
In each round of relevance feedback:
$Getfeedback(\mathcal{P}^*, \mathcal{N}^*)$
$\mathcal{P} \leftarrow \mathcal{P} \cup \mathcal{P}^*$; $\mathcal{N} \leftarrow \mathcal{N} \cup \mathcal{N}^*$; $\mathcal{U} \leftarrow \mathcal{U} - (\mathcal{P}^* \cup \mathcal{N}^*)$
for $i \in \{1..2\}$ **do**
  $\mathcal{P}_i \leftarrow \mathcal{P}$
  $\mathcal{N}_i \leftarrow \mathcal{N} \cup \{\arg\min_{\boldsymbol{x} \in \mathcal{U}} L_{(3-i)}(\boldsymbol{x}, \mathcal{P}, Generalize(\mathcal{N}, \mathcal{U}, k))\}$
for $\boldsymbol{x} \in \mathcal{U}$ **do** $Rank(\boldsymbol{x}) \leftarrow \frac{1}{Z_{norm}} \sum_{i \in \{1..2\}} L_i(\boldsymbol{x}, \mathcal{P}_i, Generalize(\mathcal{N}_i, \mathcal{U}, k))$
$Pool \leftarrow \emptyset$; $Result \leftarrow \emptyset$
for $i \in \{1..poolsize\}$ **do** $Pool \leftarrow Pool \cup \{\arg\min_{\boldsymbol{x} \in \mathcal{U}} Abs(Rank(\boldsymbol{x}))\}$
for $i \in \{1..resultsize\}$ **do** $Result \leftarrow Result \cup \{\arg\max_{\boldsymbol{x} \in \mathcal{U}} Rank(\boldsymbol{x})\}$

Output: *Result*; *Pool*

Experiment

- Dataset: **Corel Image Features Data Set**
    - **Contain 100 classes,** each class has 100 images
- First dataset: 20 classes
- Second dataset: 100 classes

Table III. Geometrical effectivenesses of SSAIRA (SA), BALAS (B), SVM$_{Active}$ (V), and NAIVE (N) when $C = 20$

	$F = 5$				$F = 7$				$F = 9$			
	SA	B	V	N	SA	B	V	N	SA	B	V	N
$\bar{\eta}^0_{200}(\%)$	**43.8**	28.5	38.7	40.2	**45.8**	25.4	40.8	42.7	**47.7**	24.4	42.5	44.4
$\bar{\eta}^1_{200}(\%)$	**45.0**	28.5	39.6	41.6	**49.6**	28.0	41.8	44.0	**50.3**	25.6	43.1	44.5
$\bar{\eta}^2_{200}(\%)$	**46.8**	29.9	40.3	43.1	**51.1**	26.9	42.9	46.4	**53.0**	26.6	44.6	46.7
$\bar{\eta}^3_{200}(\%)$	**49.7**	29.1	40.9	44.2	**52.9**	26.6	45.4	47.3	**54.6**	27.8	46.7	47.7
$\bar{\eta}^4_{200}(\%)$	**52.0**	28.8	43.9	45.1	**54.6**	26.0	46.8	47.6	**55.9**	27.8	48.5	49.2
$\bar{\eta}^5_{200}(\%)$	**53.2**	29.7	46.7	45.7	**56.1**	28.4	47.5	49.3	**57.6**	29.5	50.5	49.9

Table IV. Geometrical effectivenesses of SSAIRA (SA), BALAS (B), SVM$_{Active}$ (V), and NAIVE (N) when $C = 100$

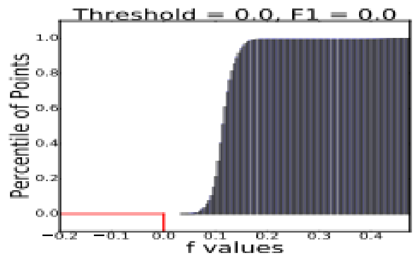
	$F = 5$				$F = 7$				$F = 9$			
	SA	B	V	N	SA	B	V	N	SA	B	V	N
$\bar{\eta}^0_{200}(\%)$	**19.0**	8.6	13.9	15.7	**22.1**	9.9	16.7	18.6	**23.2**	10.9	17.6	19.5
$\bar{\eta}^1_{200}(\%)$	**20.0**	12.3	14.7	15.3	**22.7**	13.4	17.3	16.8	**23.5**	14.6	17.6	17.8
$\bar{\eta}^2_{200}(\%)$	**20.9**	14.2	14.0	17.7	**24.4**	15.1	16.5	20.3	**25.2**	16.4	17.7	20.9
$\bar{\eta}^3_{200}(\%)$	**22.7**	15.2	14.9	19.7	**26.0**	16.4	17.1	22.3	**26.7**	18.0	18.2	23.3
$\bar{\eta}^4_{200}(\%)$	**23.5**	16.2	16.8	21.2	**27.0**	17.8	18.0	23.6	**27.3**	19.2	20.6	24.8
$\bar{\eta}^5_{200}(\%)$	**24.6**	17.3	18.1	21.4	**27.7**	18.5	20.7	24.5	**28.6**	19.9	23.1	26.3

Adaptive threshold

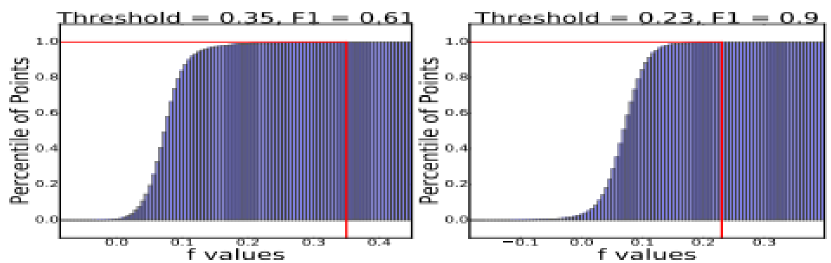
Algorithm 3 Algorithm for adaptive threshold

(Online steps)

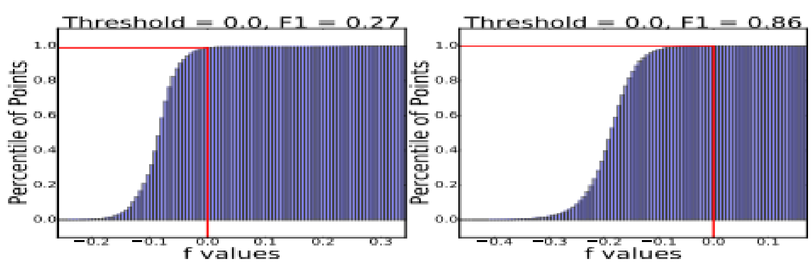
$i \Leftarrow 1$
$f^* \Leftarrow$ final f^* for all features
$\Theta \Leftarrow 0$
while $i \neq$ Maximum number of active learning rounds **do**
  $x \Leftarrow$ Point whose f^* value is closest to Θ
  Query the label for x
  if Predicted label of $x \neq$ actual label of x **then**
    if Predicted label of x = -1 **then**
      $\Theta \Leftarrow \Theta - \frac{1}{\alpha i}$.
    else
      $\Theta \Leftarrow \Theta + \frac{1}{\alpha i}$
    end if
  end if
  Add queried point to the set of labeled points
  Obtain f^* using the new set of labeled points
  $i \Leftarrow i + 1$
end while



(d) Zeroth iteration



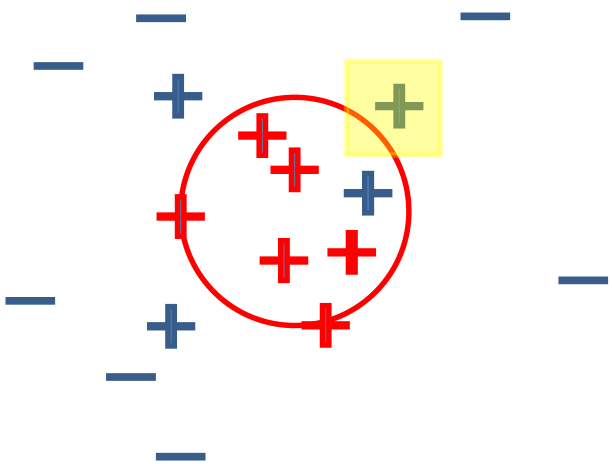
(e) With adaptive threshold after 20 and 200 labeling rounds



(f) With constant threshold after 20 and 200 labeling rounds

Some Trying

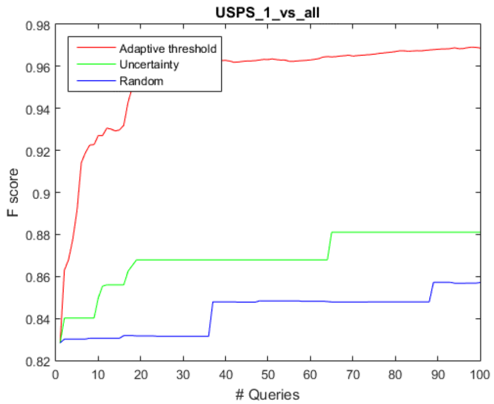
- One-class active learning:
  - At first, the handful positive examples contain little distribution information about target class.
  - How to enlarge the positive set?

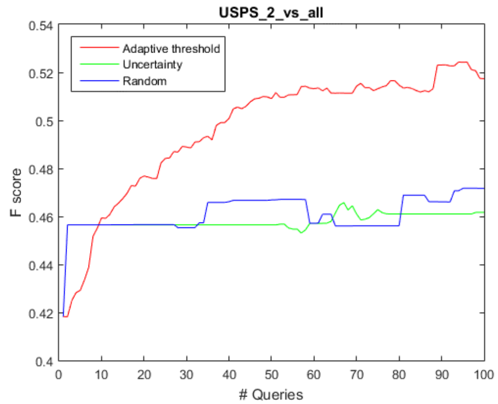


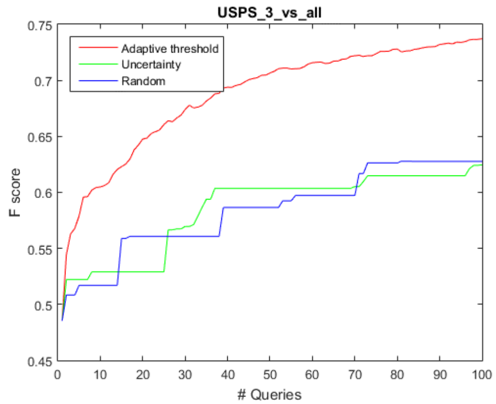
Base learner: one-class SVM
Query selection:

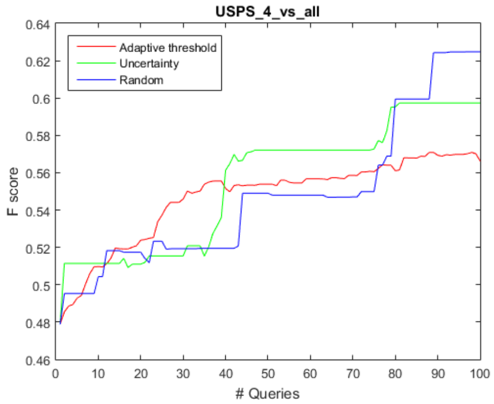
- Combine two similarity metric methods to evaluate similarity between an instance and the target class
- Adaptive threshold:
  - If positive, threshold += step_size
  - If negative, threshold -= step_size

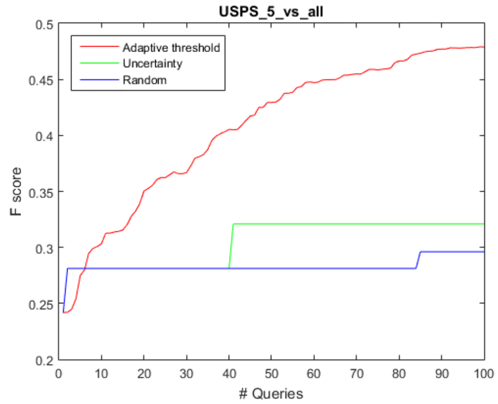
USPS


USPS_1_vs_all
Adaptive threshold
Uncertainty
Random
F score
Queries


USPS_2_vs_all
Adaptive threshold
Uncertainty
Random
F score
Queries


USPS_3_vs_all
Adaptive threshold
Uncertainty
Random
F score
Queries


USPS_4_vs_all
Adaptive threshold
Uncertainty
Random
F score
Queries


USPS_5_vs_all
Adaptive threshold
Uncertainty
Random
F score
Queries