# Image Annotation using Metric learning in Semantic Neighbourhoods



① Given Training set $T = \{(I_1, Y_1), \ldots, (I_t, Y_t)\}$.
which has ~~I~~ $t$ no of images, & each image ~~has~~ annotated with some labels ~~f~~ from $Y$ set.

Maths ② Prepare semantic groups$^{sets}$ for each label $y_i$ in $Y$ set ('i' no of subsets)

$$T_i \subseteq T \Rightarrow \quad T_1 = \{(I_1, y_1) \ldots (I_r, y_1)\}$$
$$T_2 = \{(I_1, y_2), (I_2, y_2) \ldots\}$$
$$\vdots$$
$$T_l = \{(I_1, y_l), (I_3, y_l) \ldots\}$$

③ Given "J" as unannotated image.

Statistics image-to-label, ① weak labelling ② class imbalance ④ Pick $K_1$ images from each semantic group, that are most similar to J.
& form corresponding sets $T_{J,i} \subset T_i \Rightarrow T_{J,1} = \{(I_1, y_1) \ldots (I_k, y_1)\}$
$T_{J,2} = \{(I_4, y_2), \ldots (I_k, y_2)\}$
$T_{J,l} = \{(I_1, y_l) \ldots (I_k, y_l)\}$

⑤ Merge all the sets $T_{J,i}$ to form $T_J \subset T$ (specific to J)

image-image ⑥ Calculate weighted sum over samples in $T_J$ to assign importance to labels based on image similarity.

Maths ① The posterior probability for J given label $y_k \in Y$ as.

$$P(J/y_k) = \sum_{(I_i, Y_i) \in T_J} \theta_{J, I_i} \cdot P(y_k / I_i)$$

$$= \sum_{(I_i, Y_i) \in T_J} \exp(-D(J, I_i)) \cdot \delta(y_k \in Y_i)$$

here $\theta_{J,I_i} = \exp(-D(J, I_i))$
= contribution of image $I_i$ in predicting the label $y_k$ for J depending on their visual similarity.

$P(y_k / I_i) = \delta(y_k \in Y_i)$
= denotes presence / absence of label $y_k$ in the label set $Y_i$ of $I_i$
$\Rightarrow \delta(\cdot) = 1$ (for true) $= 0$ (for false).

② Assume 1st pass of 2PKNN will give subset of training data where each label has equal freq. so in uniform distribution, prior probability i.e.

$$P(y_i) = \frac{1}{|T|} \quad \forall i = \{1, 2, \ldots, l\}$$

③ Assume conditional probabilities $P(A/y_i)$ that model the feature distribution of an image A given a semantic ~~cont~~ concept $y_i \in Y$.

$$P(y_i / A) = \frac{P(A/y_i) \cdot P(y_i)}{P(A)}$$

④ Given an unannotated image J, the best label for it will be $\Rightarrow \boxed{y^* = \arg\max_i P(y_i / J)}$

# Metric learning



① Most of classification ML algos, try to increase inter-class & reduce intra-class distances. (binary manner)

② image annotation is a multi-label classification, here two samples are relate in the continuous space! So can't be applied directly.

③ ML: learn weights over multiple features as well as base distance that max the annotation performance.

④ Extend the LMNN algo, for multi-label prediction.

---

① Let 2 images A & B represented by n features.

$A = \{f_A^1, f_A^2, \ldots f_A^n\}$ , $B = \{f_B^1, f_B^2, \ldots f_B^n\}$

② Each feature is a ~~multi-dimensional~~ ~~veto~~ vector $(N_i)$: $f^i \in R^{N_i}$

$f^i \in R^{N_i}$ for $i = \{1, 2 \ldots n\}$

③ The distance b/n 2 images is computed by finding distance b/n their corresponding features (using $L_1 \Rightarrow$ color histograms, $\chi^2 \Rightarrow$ SIFT features) and then combining them all.

Eg:- Given any 2 corresponding feature vectors $f_A^i$ & $f_B^i$.

vector $\Rightarrow$ $d_{AB}^i \in R_+^{N_i}$

$$d_{AB}^i(j) = |f_A^i(j) - f_B^i(j)| \;, \forall j \in \{1, \ldots, N_i\}$$

then $L_1$ distance b/n the 2 feature vectors can be written as:

$$L_1(f_A^i, f_B^i) = v^i \cdot d_{AB}^i$$ , where $|\cdot|$ = absolute value.

*⇒ $v^i$ = normalized unit vector (generally)

= can be replaced by any non-negative real-valued normalized vector that assigns appropriate weights to individual dimensions of a feature vector in feature space.

* also learn weights $w \in R^n$, in the distance space to optimally combine multiple feature distances.

Distance b/n A & B

$$D(A,B) = \sum_{i=1}^{n} w(i) \cdot \sum_{j=1}^{N_i} v^i(j) \cdot d_{AB}^i(j)$$