GBGI9U07: multimedia document: description and automatic retrieval

4. Active Learning for Multimedia

Georges Quénot and Philippe Mulhem

Multimedia Information Indexing and Retrieval Group

Laboratory of Informatics of Grenoble

January 2018

Outline

- Introduction / example
- Active learning principles
- Application categories
- Implementation aspects
- Some works in active learning
- Conclusion

Introduction

Active learning

Two meanings:

- Human active learning: when the teacher requires an active participation of the pupils not just that they passively listen.

- Machine active learning: supervised machine learning in which the learning system interacts with a teacher / annotator / oracle to get new samples to learn from.

We consider here only machine active learning.

Learning a concept from labeled examples

Raw data: need for a teacher / annotator / oracle / user
$\rightarrow$ human intervention $\rightarrow$ high cost

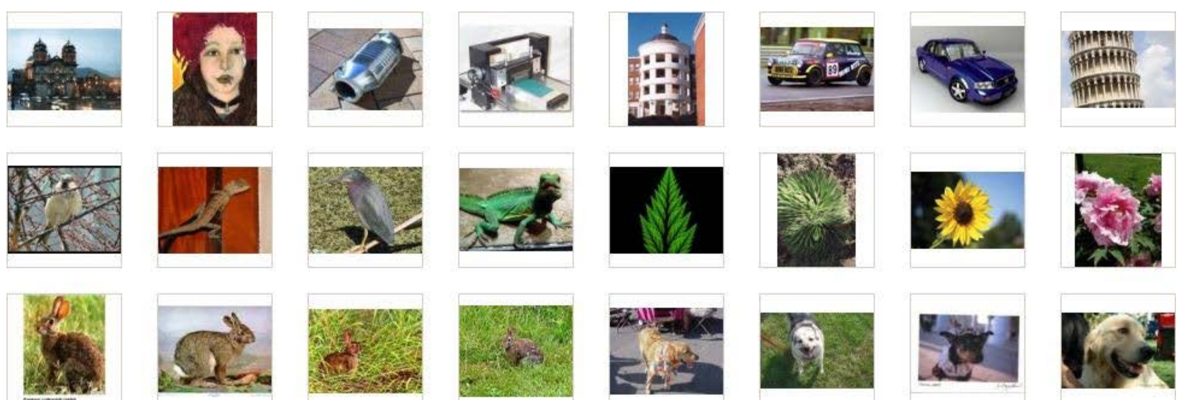
Learning a concept from labeled examples

Full annotation: possibly optimal in quality but highest cost

Cats:

Cats?

Non cats:



Learning a concept from labeled examples

Partial annotation: less costly, possibly of similar quality but need to select “good” examples for annotation

Cats:






Cats?



Non cats:







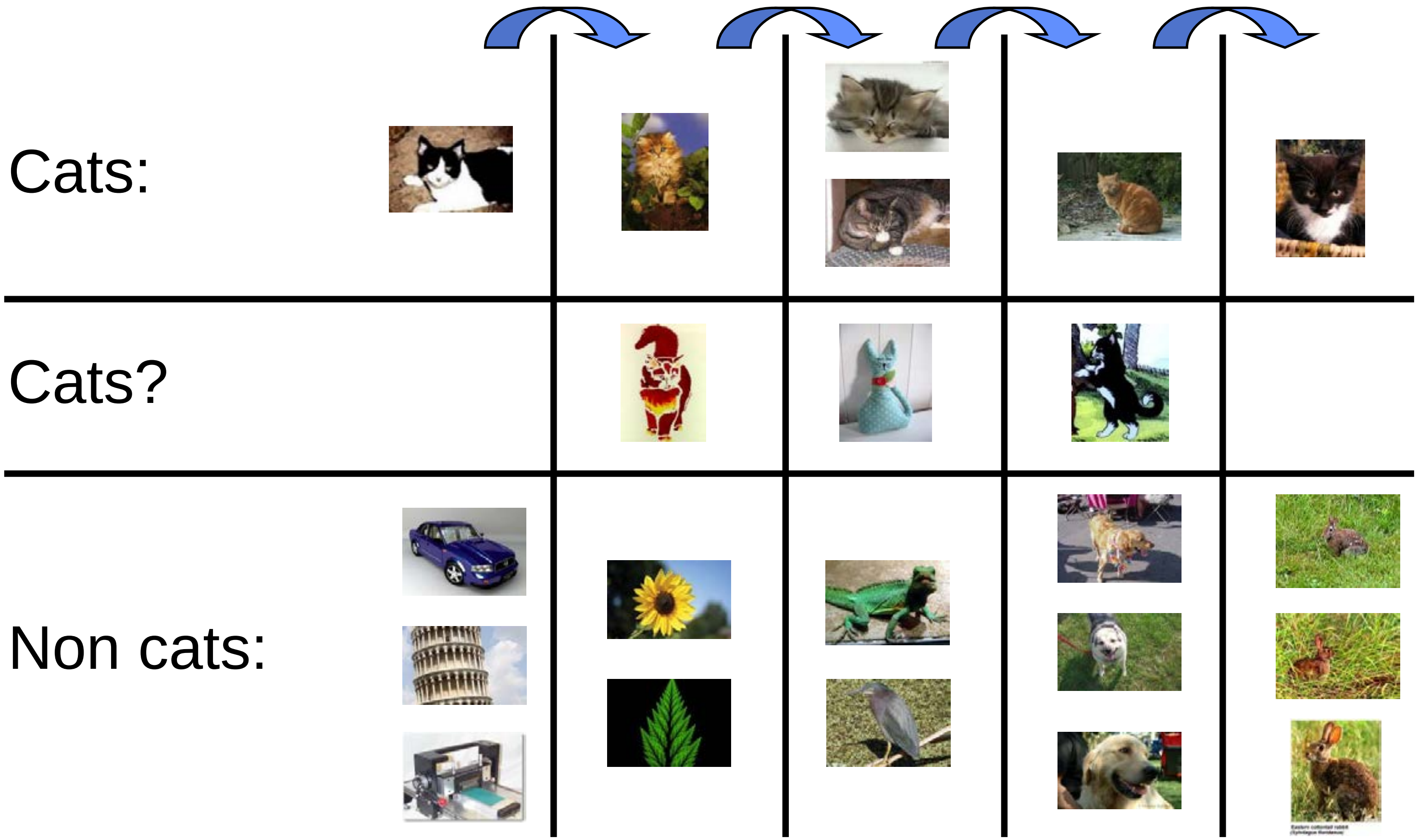




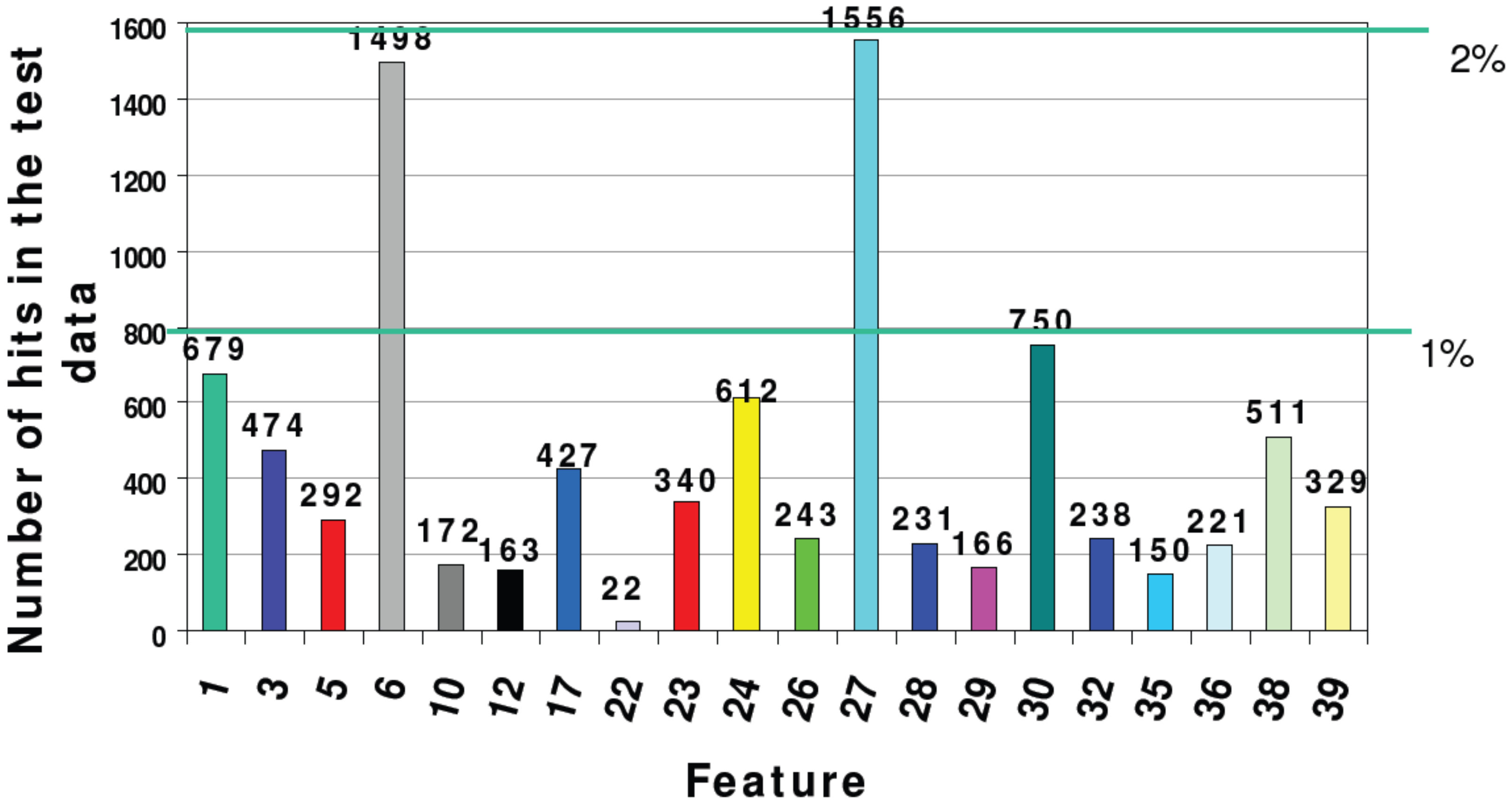


Learning a concept from labeled examples

Incremental partial annotation: samples for annotation are selected on the basis of a class prediction using a learning system $\rightarrow$ relevance feedback or query learning



Frequency of hits by features
[from Paul Over and Wessel Kraaij, 2006]



Active learning principles

Active learning

- Machine learning:
  - Learning from data.
- Supervised learning:
  - Learning from labeled data: human intervention.
- Incremental learning:
  - Learning from training sets of increasing sizes,
  - Algorithms to avoid full retrain of the system at each step.
- Active learning:
  - Selective sampling: select the “most informative” samples for annotation: optimized human intervention.
- Offline active learning: indexing (classification).
- Online active learning: search (relevance feedback).

Supervised learning

- A machine learning technique for creating a function from training data.
- The training data consist of pairs of input objects (typically vectors) and desired outputs.
- The output of the function can be a continuous value (regression) or a class label (classification) of the input object.
- The task of the supervised learner is to predict the value of the function for any valid input object after having seen a number of training examples (i.e. pairs of input and target output).
- To achieve this, the learner has to generalize from the presented data to unseen situations in a "reasonable" way.
- The parallel task in human and animal psychology is often referred to as concept learning (in the case of classification).
- Most commonly, supervised learning generates a global model that helps mapping input objects to desired outputs.

(http://en.wikipedia.org/wiki/Supervised_learning)

Supervised learning

- Target function: $f : X \to Y$
  $$x \to y = f(x)$$
  - x : input object (typically vector)
  - y : desired output (continuous value or class label)
  - X : set of valid input objects
  - Y : set of possible output values

- Training data: $S = (x_i, y_i)_{(1 \le i \le I)}$
  - I : number of training samples

- Learning algorithm: $L : (X \times Y)^* \to Y^X$
  $$S \to f = L(S)$$

- Regression or classification system: $y = [L(S)](x) = g(S,x)$

$$\left((X \times Y)^* = \bigcup_{n \in N} (X \times Y)^n \right)$$

Model based supervised learning

- Two functions, “train” and “predict”, cooperating via a Model

- General regression or classification system:

  $y = [L(S)](x) = g(S,x)$

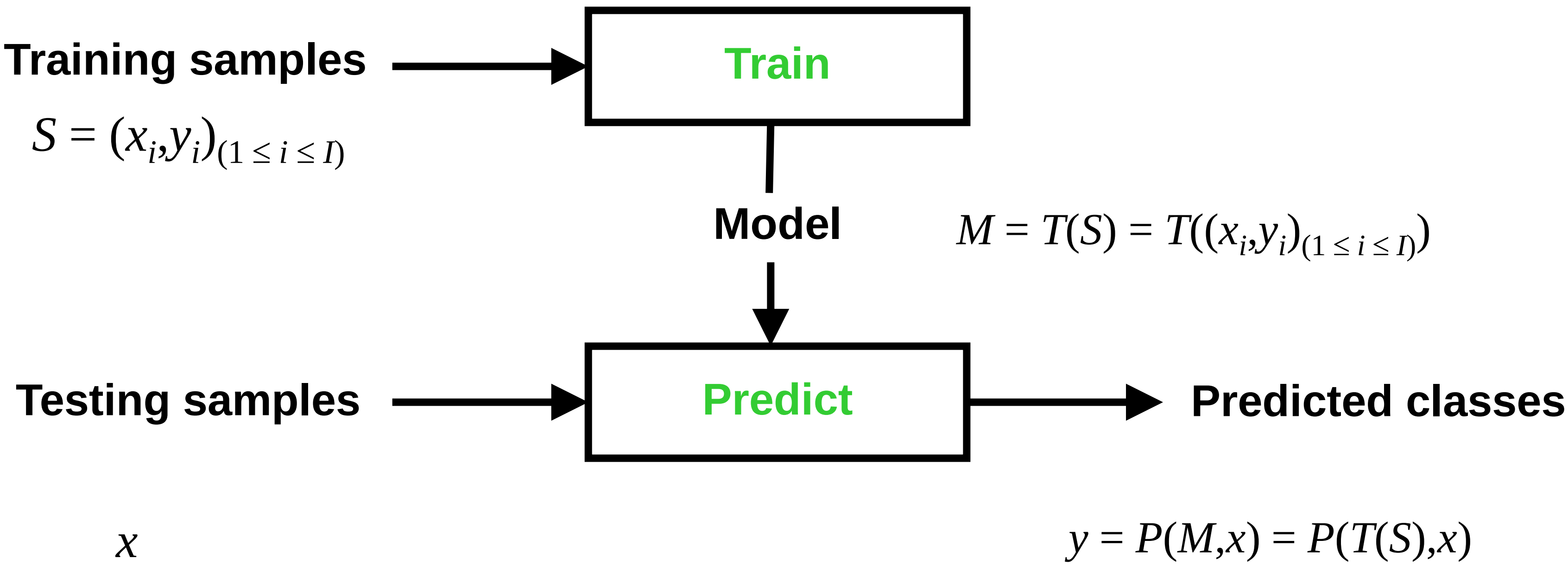
- Building of a model (train):

  $M = T(S)$

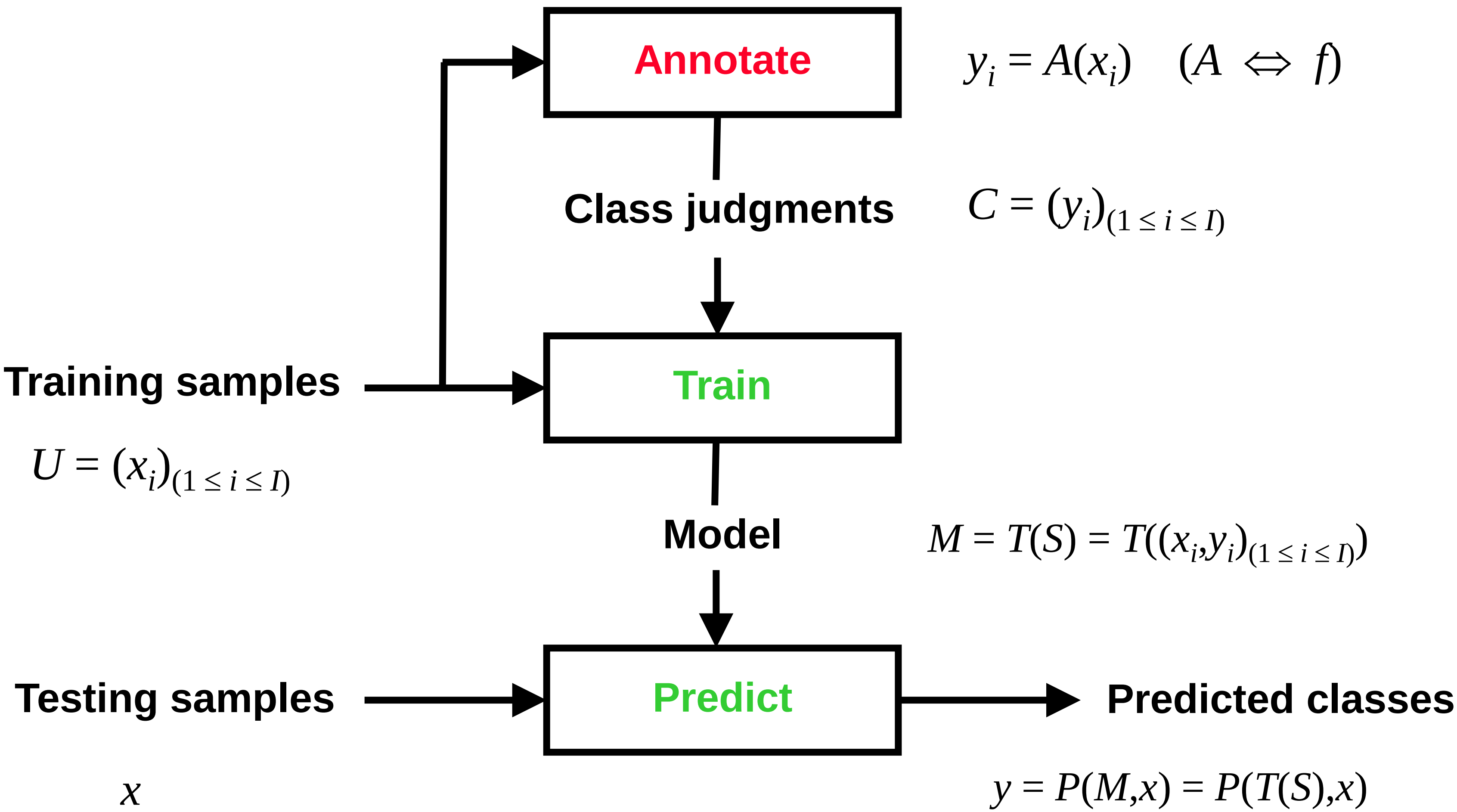
- Prediction using a model (predict):

  $y = [L(S)](x) = g(S,x) = P(M,x) = P(T(S),x)$

Supervised learning
Classification problem



Supervised learning
Classification problem



Incremental supervised learning
Classification problem

- Training set of increasing sizes $(I_k)_{(1 \le k \le K)}$:

  $S_k = (x_i, y_i)_{(1 \le i \le I_k)}$ $\quad$ ($U_k = (x_i)_{(1 \le i \le I_k)}$ $\quad$ $C_k = (y_i)_{(1 \le i \le I_k)}$)

- Model refinement:

  $M_k = T(S_k)$

- Prediction refinement:

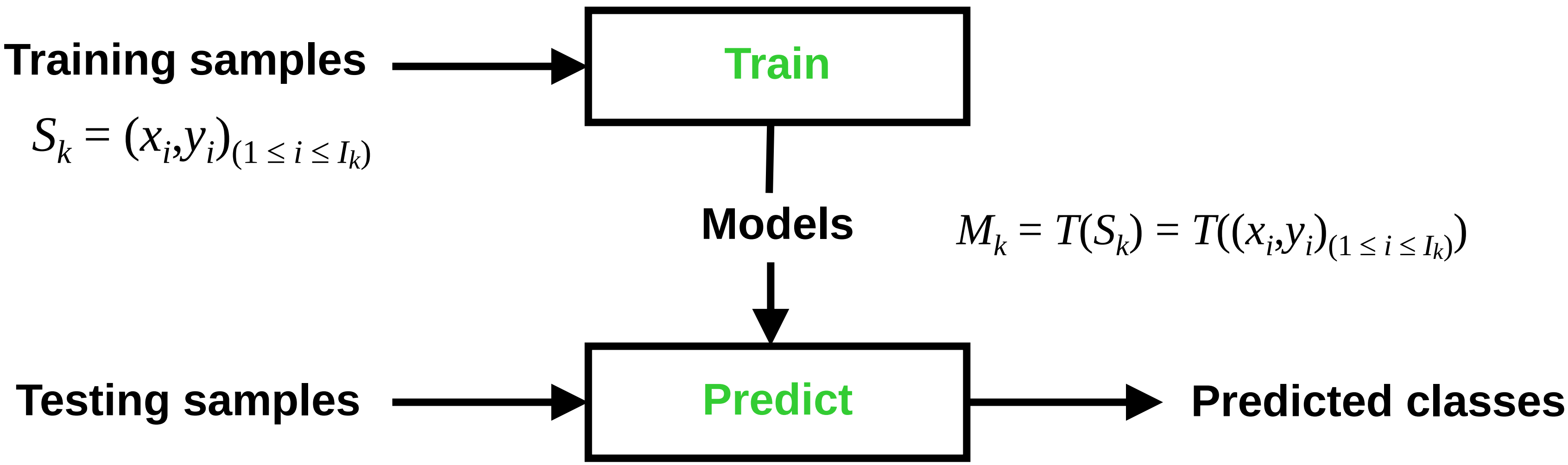
  $y_k = P(M_k, x)$ $\quad$ $y = P(M_K, x)$

- Possible incremental estimation ($k > 1$):

  $M_k = T'(M_{k-1}, S_k - S_{k-1})$

- Useful for large data sets, model adaptation (concept drift), …

Incremental supervised learning
Classification problem



$S_k = (x_i, y_i)_{(1 \le i \le I_k)}$

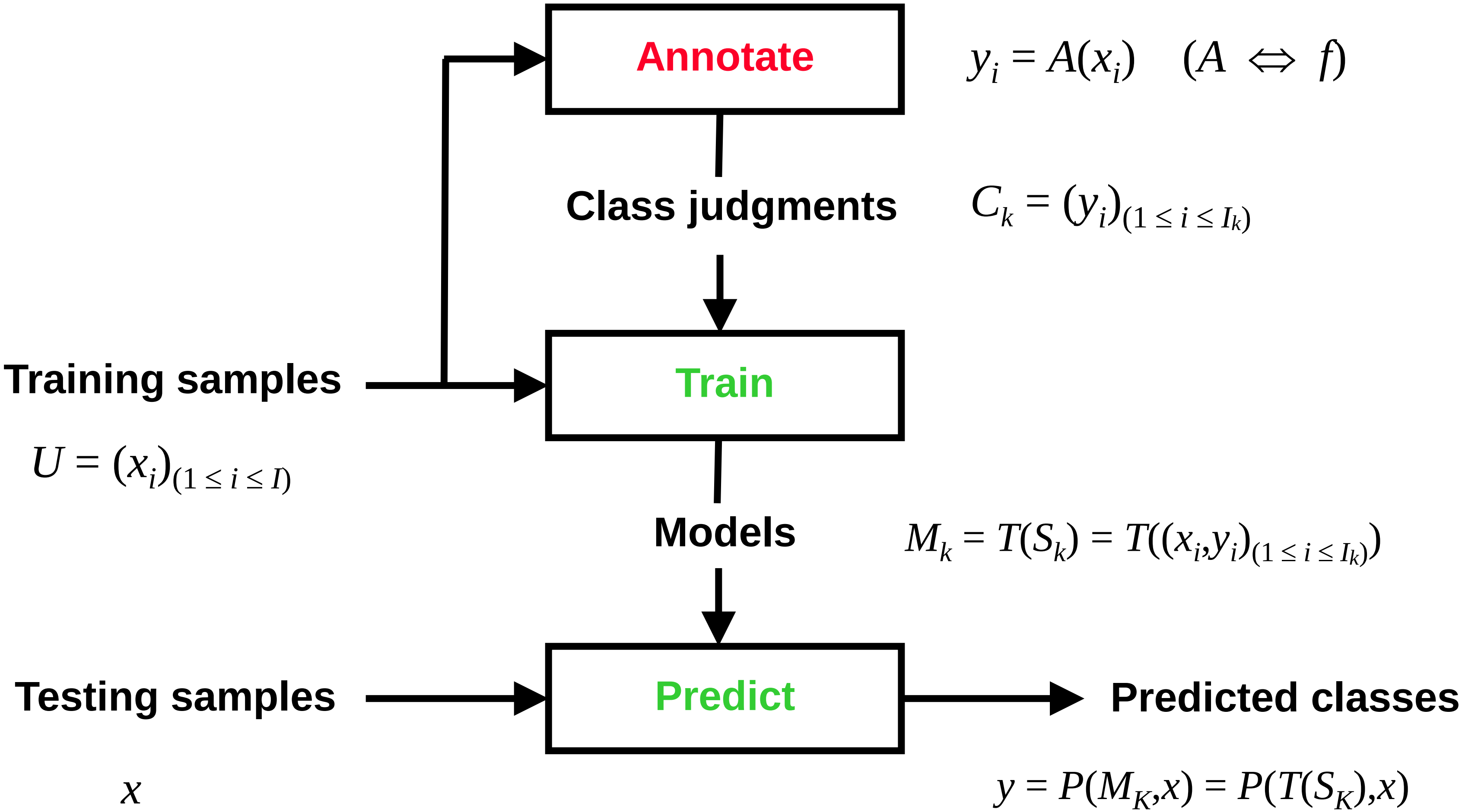
$M_k = T(S_k) = T((x_i, y_i)_{(1 \le i \le I_k)})$

x

$y_k = P(M_k, x) = P(T(S_k), x)$

$y = P(M_K, x) = P(T(S_K), x)$

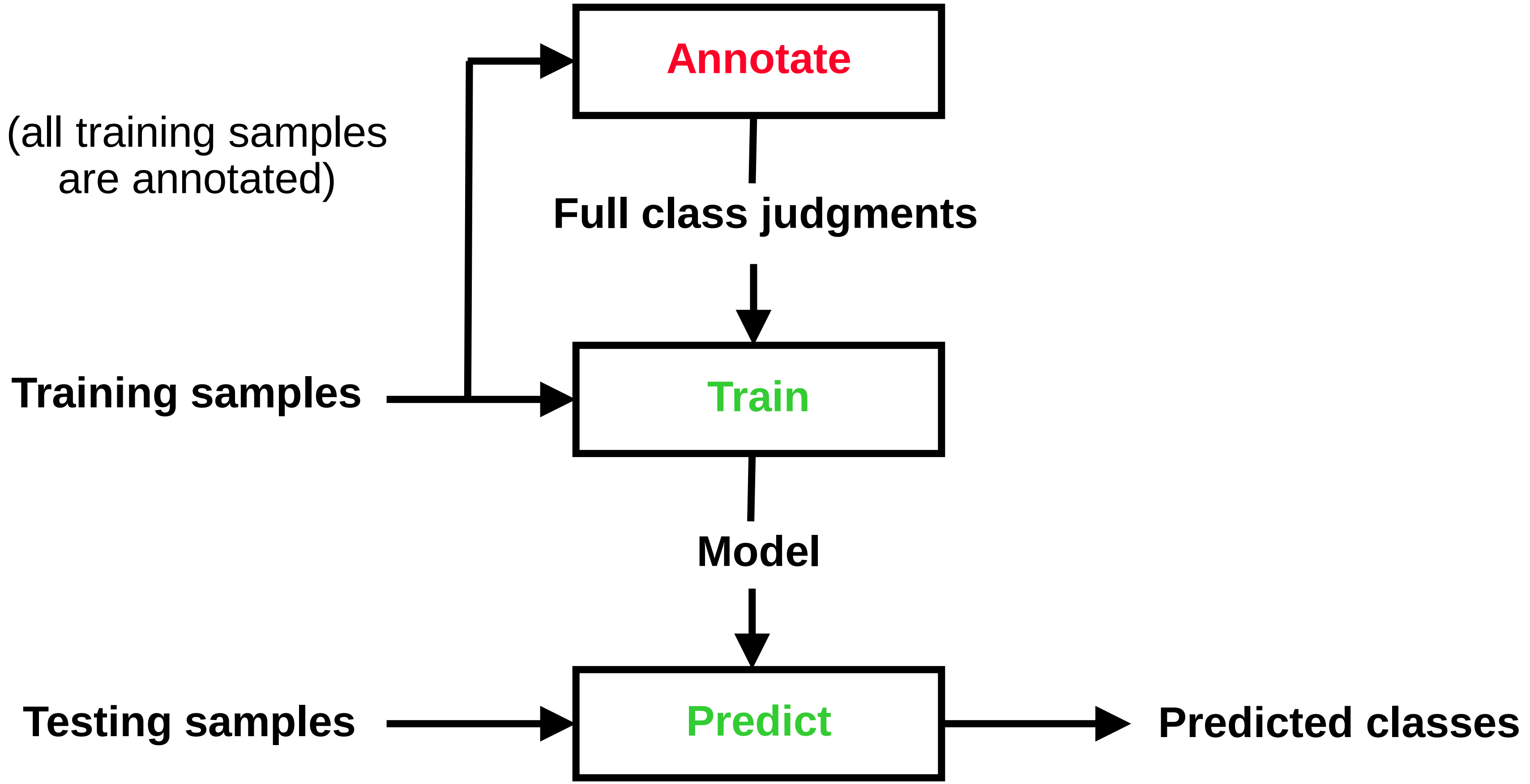
Incremental supervised learning
Classification problem



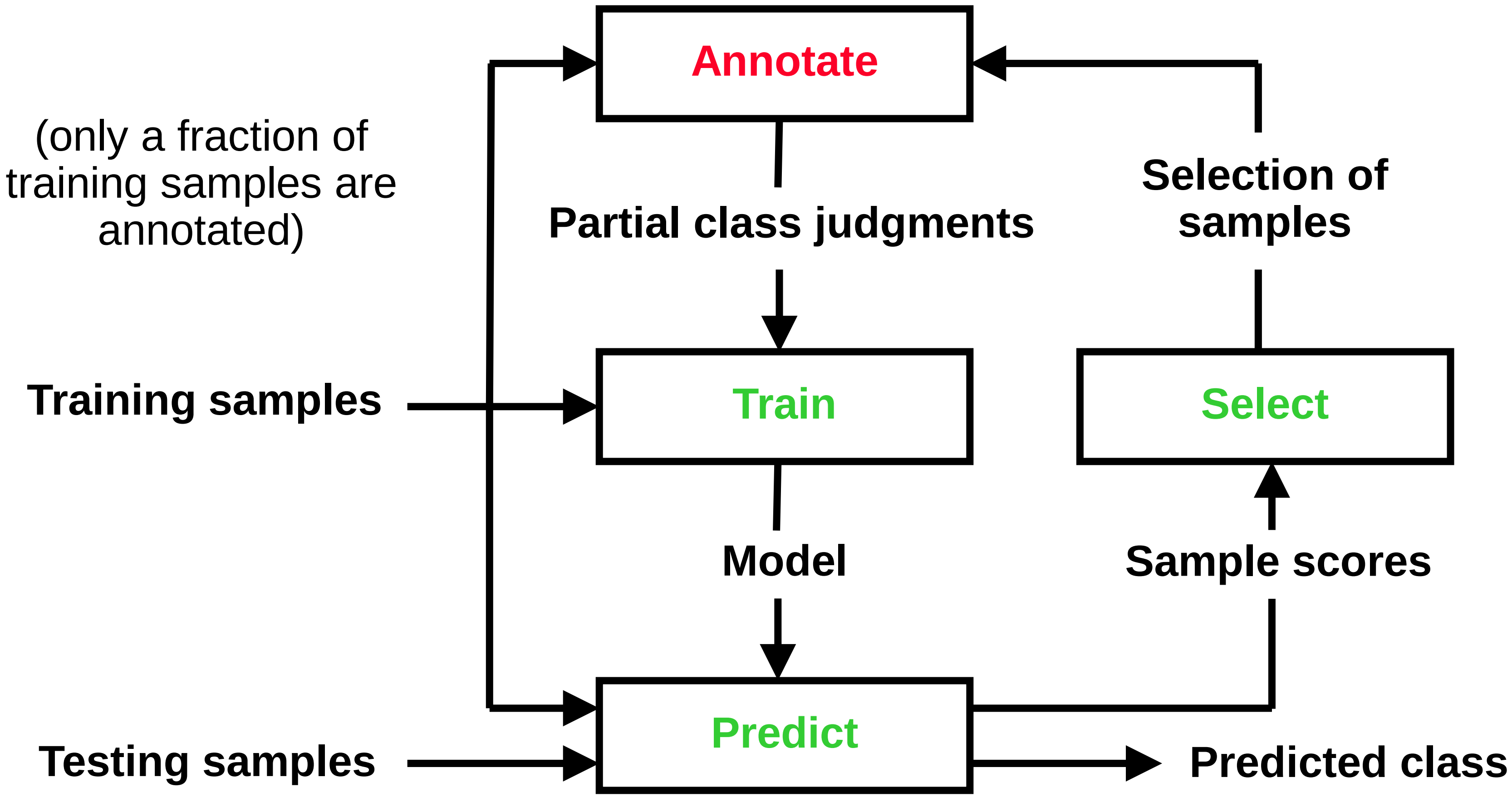
Active learning basics

- Concept classification $\rightarrow$ "Semantic gap" problem.
- Improve classification performance ?
  - Optimize the model and the train/predict algorithm.
  - Get a large training set: quantity, quality, …
- Cost of corpus annotation:
  - Getting large corpora is (quite) easy and cheap (already there).
  - Getting annotations on it is costly (human intervention).
- Active learning:
  - Use an existing system and heuristics for selecting the samples to annotate $\rightarrow$ need of a classification score.
  - Annotate first or only the samples that are expected to be the most informative for system training $\rightarrow$ various strategies.
  - Get same performance with less annotations and/or get better performance with the same annotation count.

Supervised learning
Classical approach



Active learning classification



Active learning basics

- Incremental process:
  - Needs at least one classification system (several for some strategies).
  - Small increments are better $\rightarrow$ compromise with system re-training cost.
  - “Cold start” problem: needs at least a few sample for each class to bootstrap or start with a “random” or cluster-based strategy.
  - True incremental learning (actual model adaptation) is possible but not necessary.

- Use for classification system training (offline)
- Use for corpus annotation (offline)
- Use during search (relevance feedback, online)

Active learning strategies

- Query by committee (Seung, 1992): choose the samples which maximize the disagreement amongst systems.
- Uncertainty sampling (Lewis, 1994): choose the most uncertain samples, tries to increase the sample density in the neighborhood of the frontier between positives and negatives $\rightarrow$ improve the system's precision.
- Relevance sampling: choose the most probable positive samples, tries to maximize the size of the set of positive samples (positive samples are most often sparse within the whole set and finding negative samples is easy).
- Choose the farthest samples from already evaluated ones, tries to maximize the variety of the evaluated samples $\rightarrow$ improve the system's recall.
- Combinations of these, e.g. choose the samples amongst the most probable ones *and* amongst the farthest from the already evaluated ones.
- Choose samples by groups which maximize the expected global knowledge gain (Souvanavong, 2004).

Simulated active learning

- How efficient is the active learning approach?
- Experiment on strategies and problem parameters.
- Simulated (artificial) active learning:
  - Use of a fully annotated training set.
  - Simulate incremental annotations of the training set using various strategies.
  - Use a distinct testing set (if possible with posterior contents) for concept learning evaluation (not for corpus annotation evaluation)
  - Analyze the effect of various parameters.
- Some reasonable assumptions, e.g. the order in which the annotations are done by the evaluators does not significantly influence their judgments.

Application categories

Application: concept learning

- Most popular application.
- Offline use: mainly used for classifier training, not for interaction with a user.
- Goals:
  - Increase the learning performance for a given annotation cost or
  - Reduce the annotation cost for a given learning performance or
  - Seek for a best annotation cost versus learning performance compromise.
- Evaluation:
  - Simulated active learning.
  - Distinct development and test collection.
  - Mean Average Precision performance metrics.
  - MAP as a function of the annotated fraction of the development set.
  - Comparison with different AL strategies and/or parameter values.
- It does work: huge effects reported in a variety of areas

Application: corpus annotation

- Growing application.
- Offline use: used for corpus annotation, not for interaction with a user.
- Principles:
  - A fraction of the corpus is manually annotated.
  - The remainder of the corpus is automatically annotated using a classifier trained using the manually annotated part.
  - The classifier is only temporarily used for the corpus annotation, not a goal.
- Goals:
  - Increase the full corpus annotation quality for a given manual annotation cost or
  - Reduce the manual annotation cost for a given corpus annotation quality or
  - Seek for a best manual annotation cost versus full annotation quality compromise.
- Evaluation:
  - Simulated active learning.
  - Same collection for development and test.
  - Error rate performance metrics.
  - Error rate as a function of the manually annotated fraction.
  - Comparison with different AL strategies and/or parameter values.
- It does work: significant effects reported in several areas.

Application: search (relevance feedback)

- Popular application.
- Online use: used for interaction with a user.
- Principles:
  - The user information need is considered as a concept to be learnt.
  - An incremental supervised learning system is trained using user feedback.
  - The classifier is only temporarily used for the search task, not a goal.
- Goals:
  - Increase the search result quality for a given number of feedback cycles or
  - Reduce the number of feedback cycles for a given search result quality or
  - Combination of both.
- Evaluation:
  - Simulated active learning and user interaction.
  - Same collection for development and test.
  - MAP on the last result list performance metrics.
  - MAP as a function of the number of feedback cycles.
  - Comparison with different AL strategies and/or parameter values.
- No random baseline (or baseline is no feedback): only comparison between strategies.

Implementation aspects

Online versus offline active learning

- Relevance feedback: everything is online.

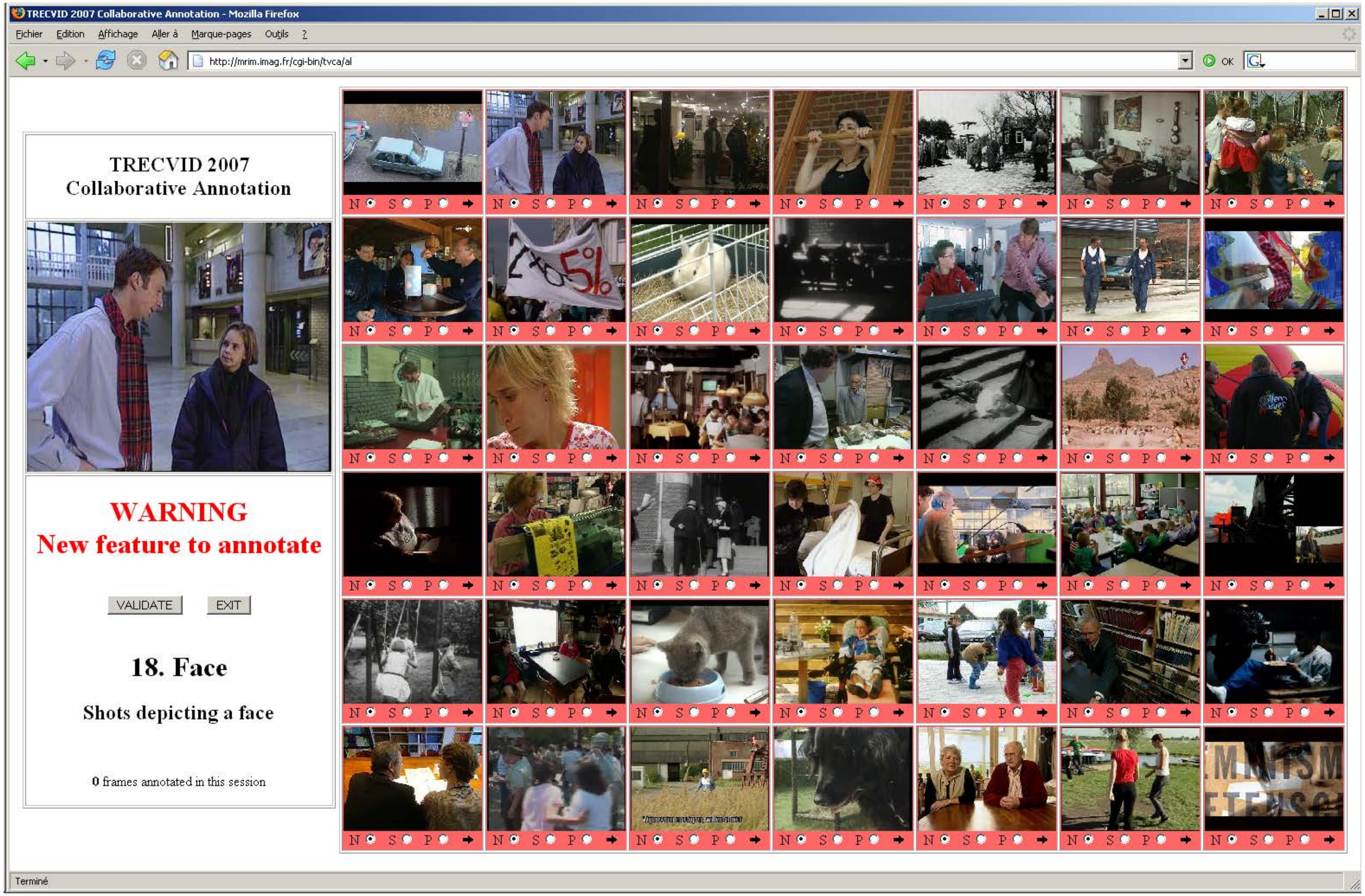
- Concept learning and corpus annotation:
  - Offline relatively to the final use but online relatively to the teacher.
  - The teacher may have to wait for the system to select new samples for annotation.
  - The system may have to wait for the teacher to do new annotations.

- The AL iteration cycle has to be optimized
  - Fast system training $\rightarrow$ possible use of a simplified learning system.
  - Efficient user work $\rightarrow$ annotate several samples in a series.

Active learning iteration cycle

- Compromise about the step (or chunk) size
  - The smaller the better for AL efficiency.
  - The larger the better for training time cost and teacher work efficiency.
- Interlacing of training and annotating phases for different concepts
  - Better for user work: possibly continuous activity.
  - Still need for a compromise about the step or chunk size.
  - Consider the relation between annotation time and training time $\rightarrow$ annotation driven classifier retraining.
  - Largely error prone because of frequent changes in the concepts to be annotated.

Parallel annotation of one concepts on several shot / key frame: TRECVID 2005 and 2007



User effects

Annotation errors or ambiguities:

- Inconsistencies of up to 3% between two different annotators have been reported for concept annotations in video shots even in good conditions (TRECVID collaborative annotation 2005). Even worse with more than two annotators.
- Actual ambiguities (how many stairs to make a stairway?).
- True human errors: many possible causes (e.g. helicopter as an airplane or fail to notice a change in concept to annotate).
- Significant impact: false positives and false negatives really hurts system performance.
- Active learning makes things worse because of frequent context changes.

User effects

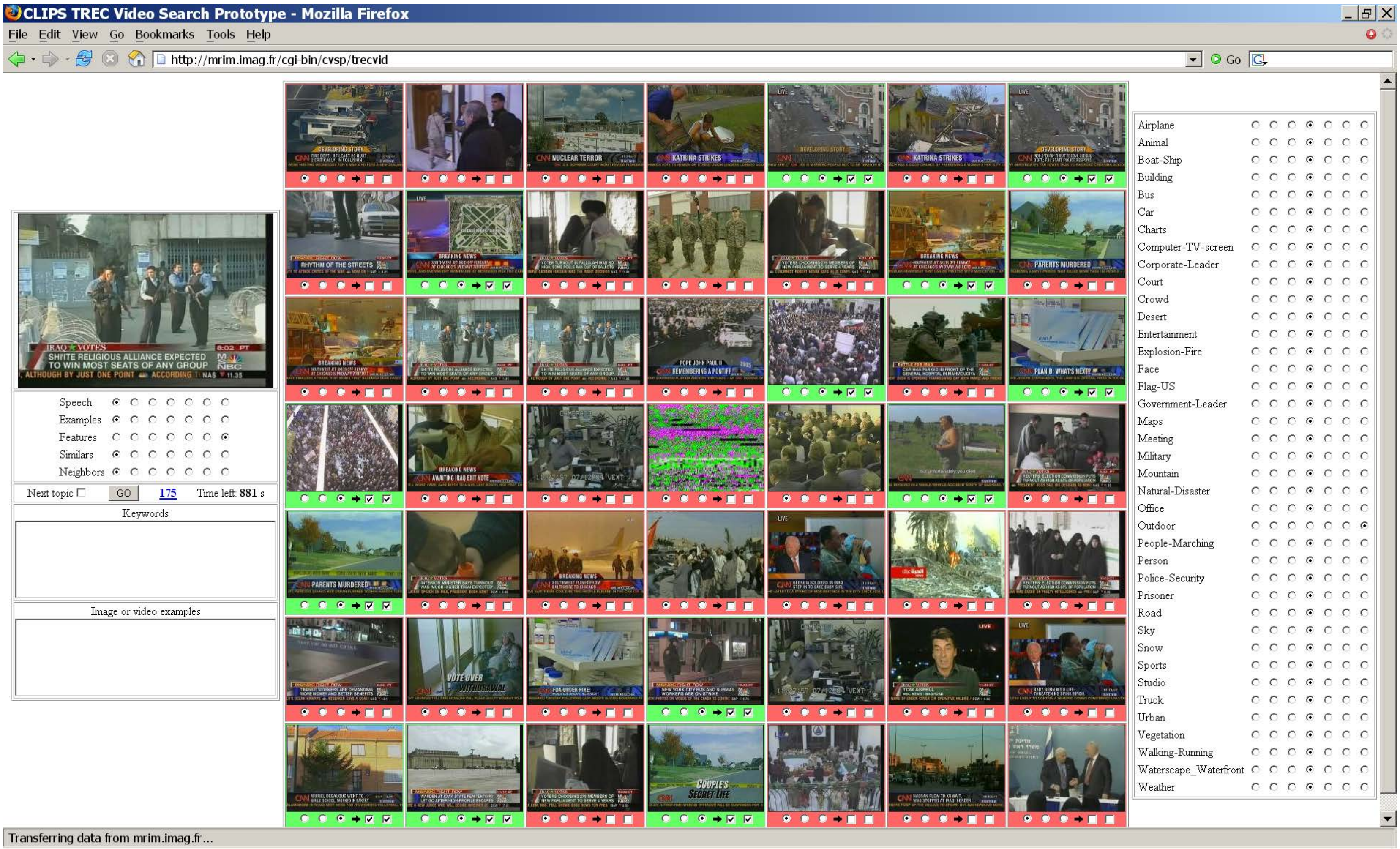
Active learning can help:

- Avoid full double or triple annotations to remove errors or solve easy ambiguities.
- Ask for a second opinion only for those samples that were misclassified with a strong confidence $\rightarrow$ Active cleaning.
- Ask for a third opinion only if the second opinion is inconsistent with the first one.
- Use only consistent samples for retraining.
- Could be evaluated using simulated active learning with a multiply fully annotated corpus (e.g. LSCOM – TRECVID 2005).
- Need to arbitrate between new annotations and reannotations.

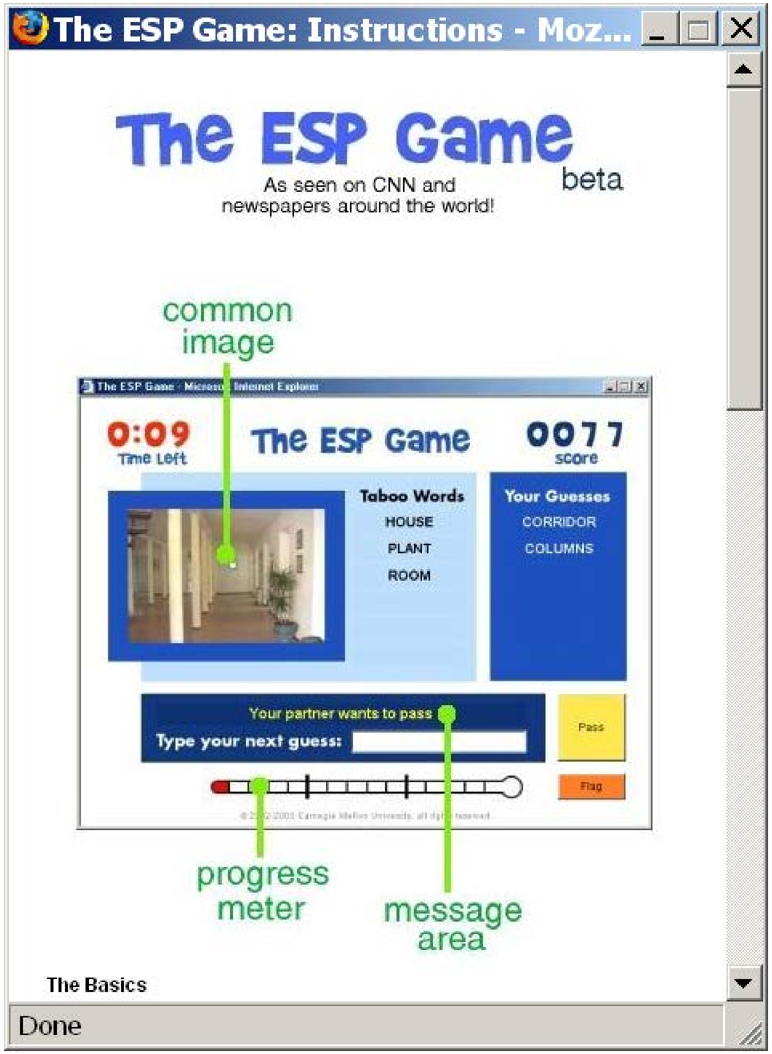
Use of a fully featured search system

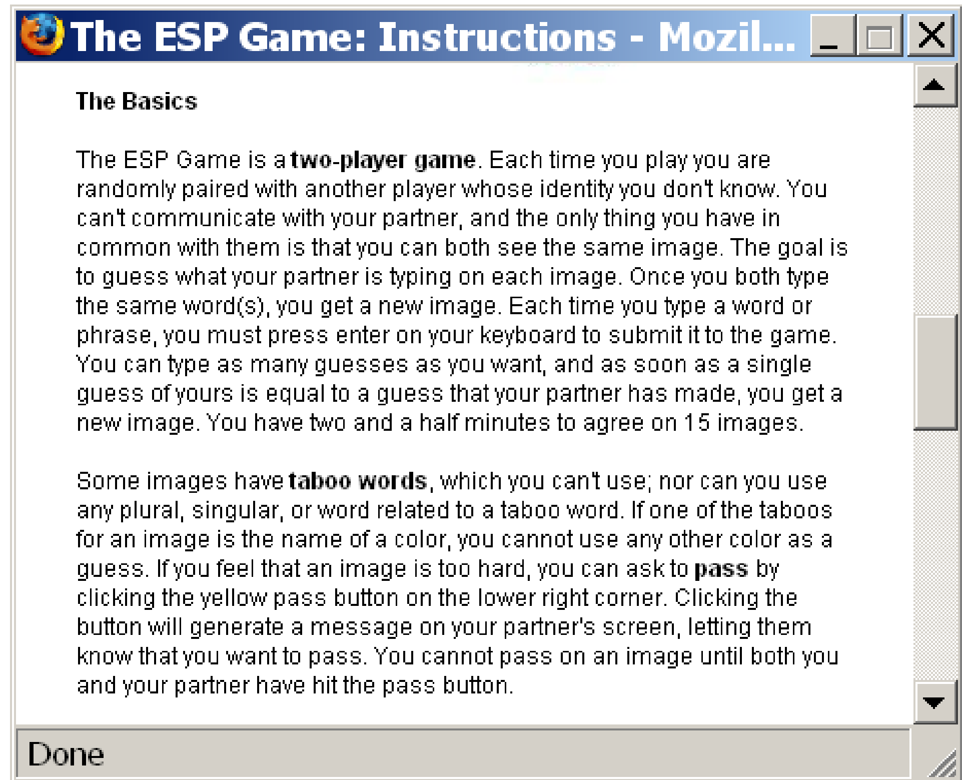
- Beyond the use of a simple classifier.
- Use of a general purpose content-based search system.
- The concept to be annotated is the query: search by active learning but not limited to relevance feedback.
- Multi-criteria search (video example, not exhaustive):
  - Keywords from Automatic Speech Recognition.
  - Image examples.
  - Already trained and indexed (other) concepts.
  - Visual similarity to already found positive samples.
  - Temporal closeness to already found positive samples.
- Mostly useful for sparse concepts (general case).
- Possible solution to the cold start problem (but may require some system training from other sources).

Use of a fully featured search system



Game-based collaborative annotation




The ESP Game: Instructions - Mozil...

The Basics

The ESP Game is a **two-player game**. Each time you play you are randomly paired with another player whose identity you don't know. You can't communicate with your partner, and the only thing you have in common with them is that you can both see the same image. The goal is to guess what your partner is typing on each image. Once you both type the same word(s), you get a new image. Each time you type a word or phrase, you must press enter on your keyboard to submit it to the game. You can type as many guesses as you want, and as soon as a single guess of yours is equal to a guess that your partner has made, you get a new image. You have two and a half minutes to agree on 15 images.

Some images have **taboo words**, which you can't use; nor can you use any plural, singular, or word related to a taboo word. If one of the taboos for an image is the name of a color, you cannot use any other color as a guess. If you feel that an image is too hard, you can ask to **pass** by clicking the yellow pass button on the lower right corner. Clicking the button will generate a message on your partner's screen, letting them know that you want to pass. You cannot pass on an image until both you and your partner have hit the pass button.

Done

Game-based collaborative annotation

The ESP Project - Mozilla Firefox

The ESP Game beta

As seen on CNN and newspapers around the world!

31,280,732

Labels collected since October 5, 2003
(This number is updated every 12 hours)

Labeling an image means associating word descriptions to it, as shown below. Computer programs can't yet determine the contents of arbitrary images, but the ESP game provides a novel method of labeling them: players get to have fun as they help us determine their contents. If the ESP game is played as much as other popular online games, we estimate that all the images on the Web can be labeled in a matter of weeks!

guys
sitting
music
picture

Done

Game-based collaborative annotation

Features (not exhaustive)

- Low-level visual features:
  - Color: color histograms or color moments in various color spaces.
  - Texture: Gabor or wavelets transforms.
  - Motion: motion vectors or statistics on motion vectors.
  - Local and global features: Principal Component Analysis for data dimension reduction and noise cleaning.
- Low and intermediate audio features:
  - Word vector representations from Automatic Speech Recognition (ASR).
  - Music / noise / gender detection.
  - Mel Frequency Cepstral Coefficients.
- Intermediate features:
  - Output from further preprocessing: text categories, …

Some works in Active Learning

Queries and Concept Learning
[Dana Angluin, 1988]

- Mostly cited in literature about active learning but refers to Shapiro's [1981,1982,1983] Algorithmic Debugging System that uses queries to the user to pinpoint errors in Prolog programs and to Sammut and Banerji's [1986] system also for concept learning.
- Queries to instructors for concept learning tasks.
- Queries from the system to a human being (the opposite of a queries in an information retrieval system).
- Problem: identify an unknown set L_* from a finite or countable hypothesis space $L_1, L_2, \ldots$ of subsets of a universal set U.
- The system has access to oracles that can answer specific kinds of queries about the unknown concept L_* : *membership*, *equivalence*, *subset*, *superset*, *disjointness*, *exhaustiveness*.
- Majority vote strategy: Identification of the target set in $L_1, \ldots, L_N$ in $\lfloor \log_2 N \rfloor$ steps.
- Not easily transposable for multimedia indexing because indexing at the sample level usually supports only the *membership* query type.

Query by Committee
[H.S. Seung et al, 1992]

- Mostly cited in literature.
- Committee of students (learning programs).
- Queries from the system to a human being ($\neq$ queries in IR).
- The next query is chosen according to the principle of maximal disagreement.
- Parametric models with continuously varying weights
- Teacher: $\sigma_0(\mathrm{X})$ X: input vector (output space is $\{-1,+1\}$)
- Student: $\sigma(\mathrm{W};\mathrm{X})$ W: weight vector of the student function
- The training set is built up one sample at a time: $S_P = (\mathrm{X}^t, \sigma^t)_{(1 \leq t \leq P)}$
- Version space: set of all W which are consistent with the training set: $\mathcal{W}_{\mathrm{P}} = \{ \mathrm{W} : \sigma(\mathrm{W};\mathrm{X}^t) = \sigma^t \text{ , } 1 \leq t \leq P \}$

Query by Committee
[H.S. Seung et al, 1992]

- Flat prior distribution $\mathcal{P}_0(\mathrm{W})$:
  $\mathcal{P}(\mathrm{W} \mid S_P) = 1/V_P$ if $\mathrm{W} \in \mathcal{W}_\mathrm{P}$, 0 otherwise with V_P = volume($\mathcal{W}_\mathrm{P}$)
- Information gain: $I_{P+1} = -\log(V_{P+1}/V_P)$
- Choose the X^{P+1} that maximizes the information gain (not trivial)
- Two test applications: high-low game and perceptron learning of another perceptron.
- Query by committee learning:
  - Asymptotically finite information gain: the volume consistent with the observation in the parameter space is divided by a fixed finite factor.
  - Generalization error decreases exponentially.
- Random sampling:
  - Asymptotically null information gain.
  - Generalization error decreases with an inverse power law.

Query by Committee
[H.S. Seung et al, 1992]

- Suggestion of a criteria for a good query algorithm: asymptotically finite information gain.
- Closer to the multimedia indexing problem (membership only queries) but assumptions that
  - The actual teacher function can be reached by a given W_0.
  - The next sample can be chosen arbitrarily in the input space.
  - The parameter space does not vary with the number of samples $\rightarrow$ correct for a perceptron with a fixed architecture but not for classifiers in which the number of parameters is adjusted to or depends upon the size of the training set (e.g. Support Vector Machines).

Uncertainty sampling
[David Lewis and William Gale, 1994]

- A sequential Algorithm for Training Text Classifiers.
- Membership queries (from system to human, again).
- Use of a probabilistic classifier.
- Algorithm:

  1. Create an initial classifier
  2. While teacher is willing to label examples
     (a) Apply the current classifier to each unlabeled example
     (b) Find the b examples for which the classifier is least certain of class membership
     (c) Have the teacher label the subsample of b examples
     (d) Train a new classifier on all labeled examples

- Really close to the multimedia indexing/retrieval problem

Uncertainty sampling
[David Lewis and William Gale, 1994]

- Newswire classification task, use of simulated active learning.
- 319,463 training documents, 51,991 test documents, 10 categories.
- Cold start with 3 randomly chosen positive examples.
- Comparison between:
  - Random sampling (3+7),
  - Relevance sampling (3+996), increment by 4,
  - Uncertainty sampling (3+996), increment by 4,
  - Full annotation (3+319,463).
- The uncertainty sampling reduced by as much as 500-fold the amount of training data that would have to be manually classified to achieve a given level of effectiveness.
- Uncertainty sampling performs better than relevance sampling.

	uncertainty	random	relevance	full
F1	0.453	0.107	0.248	0.409

SVM active learning

[Simon Tong and Edward Chang, 2001]

- Support Vector Machine Active Learning for Image Retrieval.
- Relevance feedback for learning a “query concept”.
- Select the most informative images to query a user.
- Quickly learn a boundary that separates the images that satisfies the user query concept from the rest of the dataset.
- Algorithm:
  - Cold start with 20 randomly selected images.
  - Iterations with uncertainty sampling: display the 20 images that are the closest to the SVM boundary.
  - Final output with relevance sampling: display the 20 images that are the farthest to the SVM boundary (on the positive side).
- Significantly higher search accuracy that traditional query refinement schemes after just three of four rounds of relevance feedback.

Active learning for CBIR
[Cha Zhang and Tsuhan Chen, 2002]

- An active learning framework for Content Based Information Retrieval
- Indexing phase and Retrieval phase.
- Annotation of multiple attributes on each object.
- Indexing via uncertainty sampling based active learning.
- Uncertainty is estimated via the expected knowledge gain.
- Each object (either in the database or from the query) receives a probability associated to each feature: 0 or 1 if annotated, computed probability from the trained classifier otherwise.
- Retrieval via semantic distance between query objects and objects in the database: attribute probabilities are used as a feature vector.
- Weighted sum with low-level features.
- Experiments on a database of 3D objects: discriminate aircrafts form non aircrafts.
- Performance increases with the number of annotated objects.
- Active learning outperforms random sampling based learning.

Partition sampling
[Fabrice Souvannavong et al, 2004]

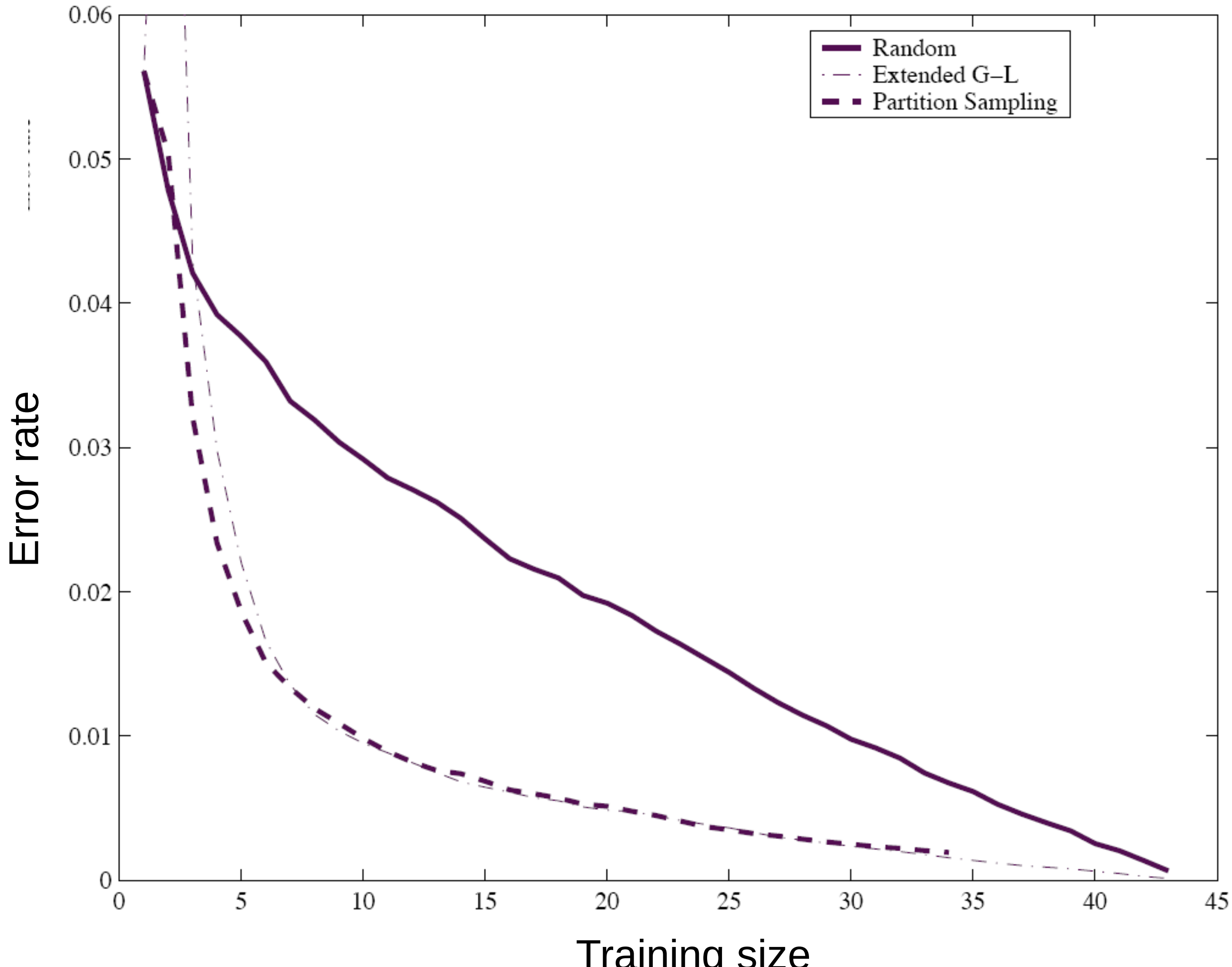
- Partition sampling for active video database annotation.
- Focus on the simultaneous selection of multiple samples.
- Select samples such that their contribution to the knowledge gain is complementary and optimal.
- Partition the pool of uncertain sample using the k-means clustering technique and select one sample in each cluster $\rightarrow$ the samples are both mostly uncertain and far from each other.
- Practical implementation:
  - HS color histograms and Gabor energies on keyframes
  - Latent Semantic Analysis (LSA) to capture local information
  - *k*-Nearest Neighbors (*k*NN) classification

Partition sampling
[Fabrice Souvannavong et al, 2004]

- Use of TRECVID 2003 development data and annotation
- The task is corpus annotation, not concept learning
  → the development and the test set are identical
  → the performance measure is the error rate on the whole set
- Comparison between
  – Random sampling
  – Greedy maximization of the error reduction
  – Partition sampling
- The partition sampling is significantly better (up to 30%) than greedy AL strategy only when a small fraction of the corpus is annotated.
- No significant difference after the annotation of about 1/6th of the corpus (no more “far” uncertain samples?).
- 0.5 % error rate after the annotation of half of the corpus against 2 % for random sampling: ~4-fold error reduction.

Partition sampling

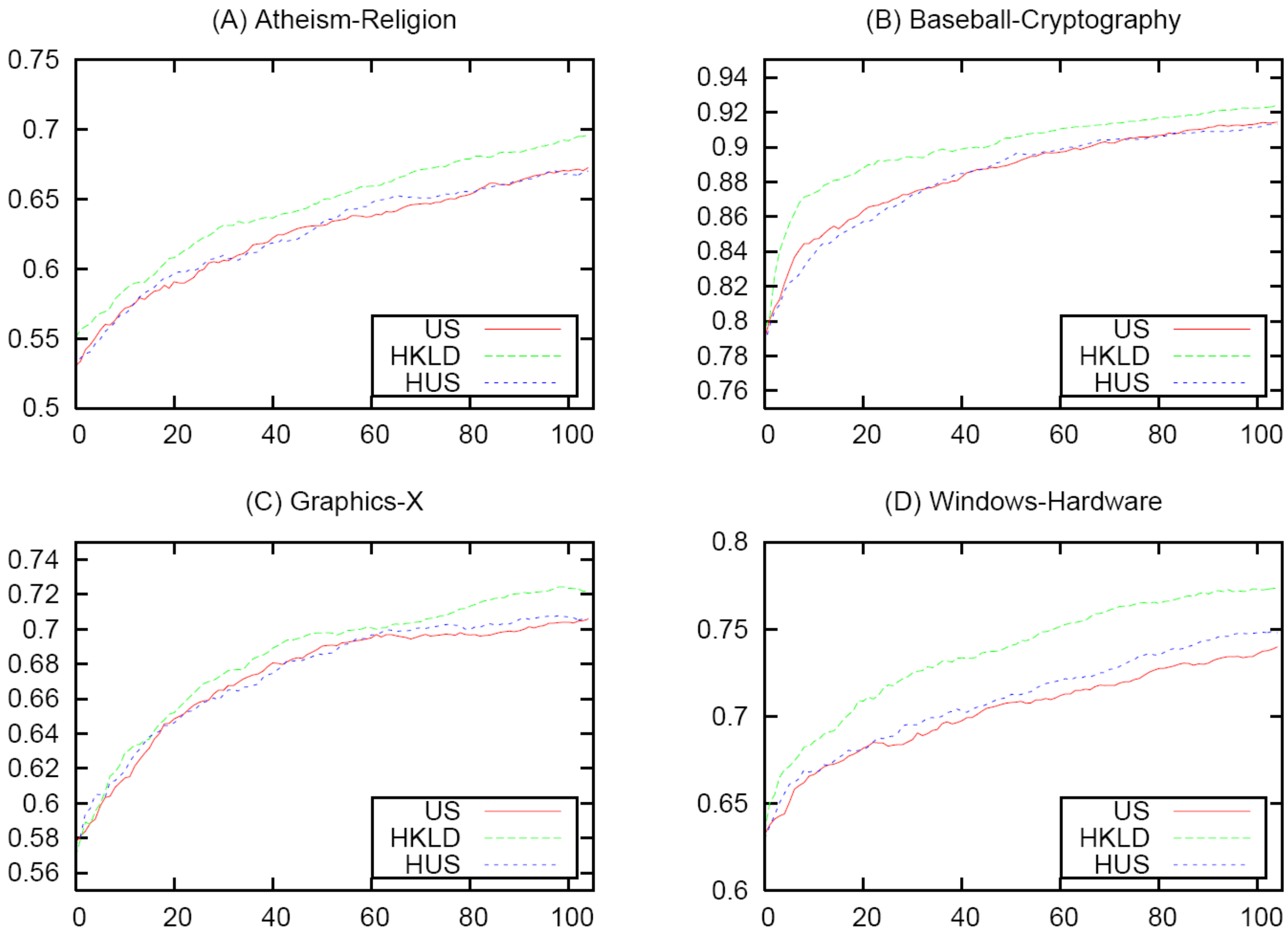
[Fabrice Souvannavong et al, 2004]



History or instability sampling
[McCallum and Nigam, 1998, Davy and Luu, 2007]

- Active Learning with History-Based Query Selection for Text Categorization [Davy and Luu, 2007].
- Select the sample which have the most erratic label assignments.
- Similar to query by committee where the committee members are the classifiers of the k previous iterations.
- History uncertainty sampling: average the uncertainty on the k previous iteration.
- Use of class distributions: works with multiple classes, all possible classes are annotated at once when a sample is selected for annotation.
- History Kullback-Leibler Divergence (KLD): average on The KLD between average distribution and committee member distributions.
- Improvement over both uncertainty sampling and history uncertainty sampling.

History or instability sampling
[Davy and Luu, 2007]



Global conclusion

Global conclusion

- Active learning greatly improve the annotation cost versus system performance quality.
- Moderate additional cost in complexity.
- Main applications: classifier training, corpus annotation and relevance feedback during search.
- Main strategies: relevance sampling, uncertainty sampling and sample clustering (partition sampling) plus combinations of them including evolving strategies.
- Integration with classification techniques: SVM active learning.
- Other parameters: cold start, step size, user effects, concept difficulty, concept frequency, ...
- Practical implementation: organization of the human-system iteration cycle.
- Annotation driven active learning.