

Machine learning at Xerox

From statistical machine translation to large-scale image search

Cedric Archambeau

Machine learning for optimisation and services group

cedric.archambeau@xerox.com

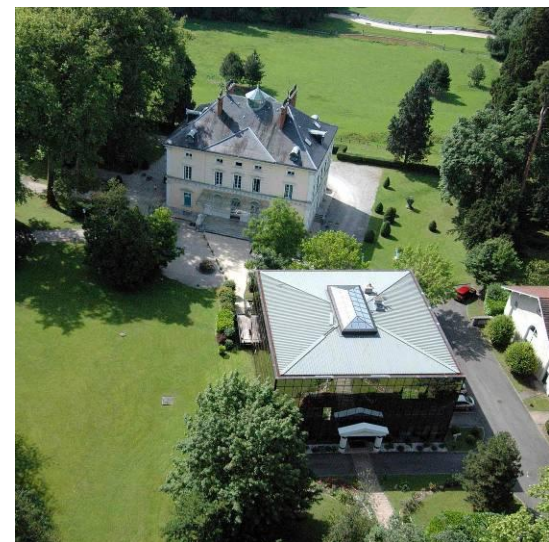


Outline

- Xerox research centre Europe
- Mail2Wiki
- Statistical machine translation
- Large-scale image search and categorisation
- Collaborative filtering

Xerox, Innovation and XRCE

Quick overview



Who we are



“We operate our businesses in ways through which economies grow, societies benefit and the environment is protected. Some call it the triple bottom line. We call it the best thing for our business success.”

Ursula M. Burns, Chairman and CEO

With sales of \$22 billion in 2010, we are the world’s largest enterprise for business process and document management.

- **Employees:** 136,000 worldwide
- **Presence:** 160 countries
- **Active patents:** +9,400
- **History:** Founded in 1906 as The Haloid Company; named Xerox in 1958 and Xerox Corporation in 1961; acquired Affiliated Computer Services in 2010.
- **Headquarters:** Norwalk, CT.

In **Fortune’s** 2009 ranking of the world’s Most Admired Companies, Xerox is ranked No. 1 in the computer industry.

What we do



Document Technology
and Services

E.g. Printing, XLS



Business Process
Outsourcing

E.g. Customer care, F&A



IT Outsourcing

Xerox R&D capabilities

Research as an engine of growth

1131 patents in 2010 (Xerox/Fuji Xerox)
Among the 25 world leaders for patents

R&D investments

4 % Revenue dedicated to R&D for Xerox
and Fuji Xerox

Xerox Research Centre of Canada
Mississauga, Ontario, Canada



Xerox Research Centre Europe
Grenoble, France



Palo Alto Research Center, Inc
Palo Alto, California, USA



Xerox Research Center Webster
Webster, NY, USA



Xerox India Innovation Hub
Chennai, India

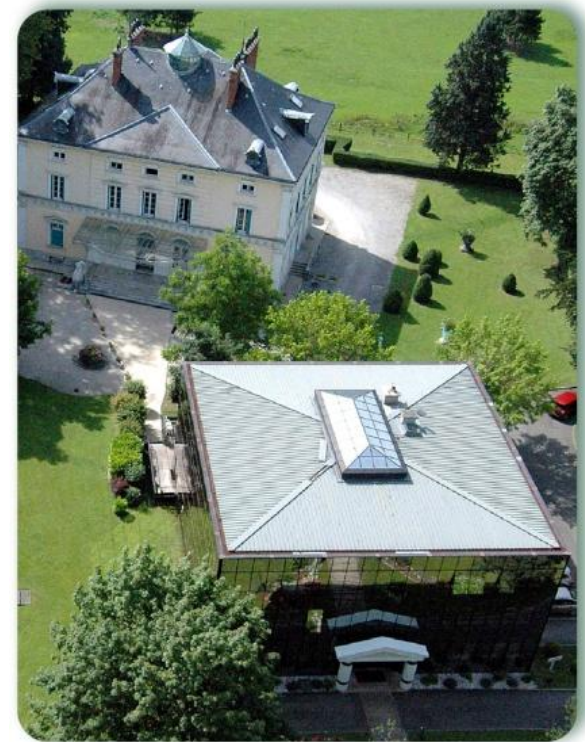


Fuji Xerox
Japan

Xerox Innovation Group: 800 researchers and engineers world-wide.

Xerox research centre Europe (XRCE)

- Established 1993
- Supports Xerox Europe (head office near London):
 - XRCE drives Xerox services and solutions business
 - Smarter Document ManagementSM technologies
- +40 permanent researchers, approx same number of interns, PhD students and Post-docs:
 - Activities linked to natural language and image processing, social network analysis, information extraction and retrieval, etc.
 - Strong machine learning component
- **European technology showroom**
 - Showcase for some of the best technology ideas (smart document touch table)
 - Monitor reactions of customers to new ideas
- **Advanced development laboratory**
 - Validation and realisation of results from research labs
 - Technology incubation and transfer to the business



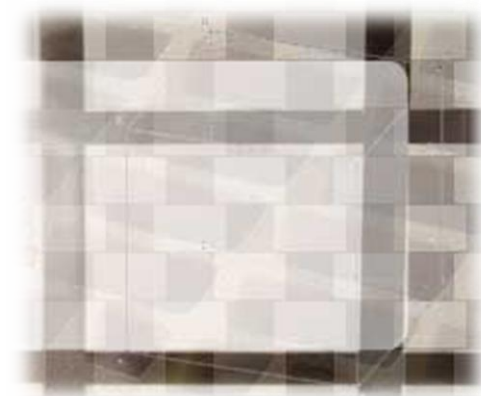
Document content lab (research)



Organise and make sense of abundance of unstructured information by creating **Smarter Document ManagementSM technologies**, and make them available as services to Xerox's customers to support business process automation, office optimisation and decision making.

- Parsing and semantics:
 - Part-of-speech tagging, entity recognition, co-reference resolution
 - Semantic disambiguation, sentiment analysis
 - Ontology acquisition (concept & relations)
- Textual and visual pattern analysis:
 - Image categorization and retrieval
 - Aesthetic quality assessment
 - Handwritten word spotting and recognition
- Machine learning for document access and machine translation:
 - Automated translation
 - Removing language barriers through industry- and function-specific solutions
 - Multilingual and multimodal document categorization and clustering

Services innovation lab (research)



Address key challenges raised by services and **service delivery systems**: online platforms for services delivery, trends towards self-provision and (mass) customization, fading distinction between products and services, enriched product value through service agreements.

- Machine learning for optimisation and services:
 - Statistical machine learning (e.g. Bayesian networks, structured prediction)
 - Mechanism design (e.g. Vickrey auction) and game theory
 - Process control and optimisation of service provision
- Work practice technologies:
 - Ground the design of systems in a thorough understanding of the workplace
 - Ethnographic studies focusing on various industrial and commercial enterprises
- Document structure:
 - Technologies to qualify XML resources in distributed/collective ecosystems
 - Document understanding and processing in domain-specific contexts of use
 - Document network understanding and social network analysis

Mail2Wiki

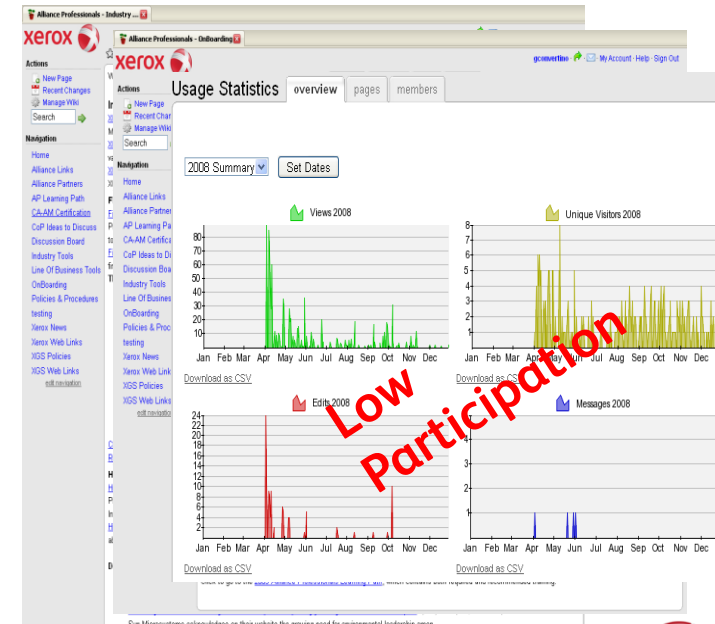
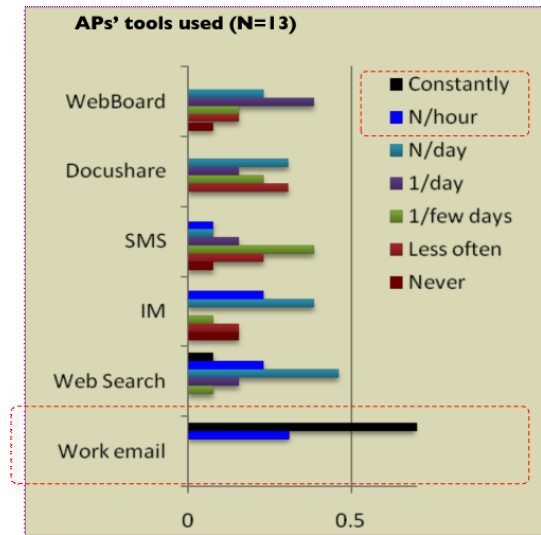
Bridging the gap between private and shared information repositories

Joint work with Guillaume Bouchard (XRCE), Antonietta Grasso (XRCE), Gregorio Convertino (PARC) and Ed Chi (ex-PARC, now at Google)



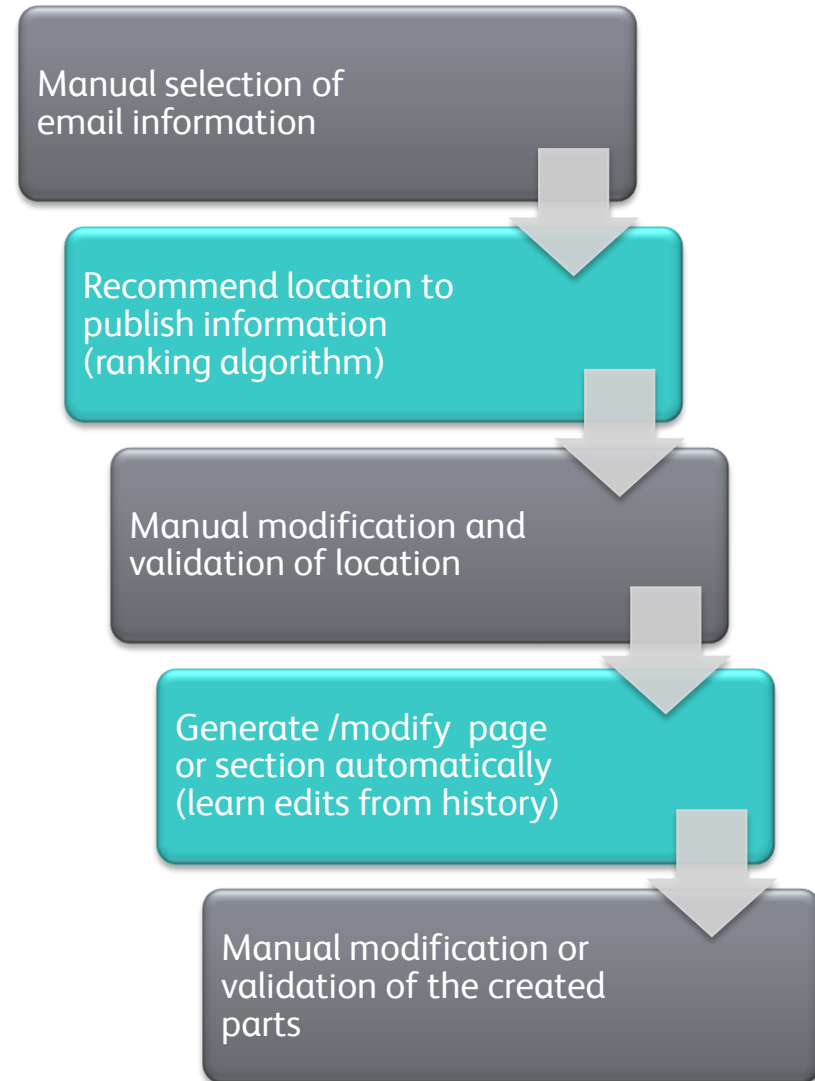
Facts about email and corporate wikis

- Email content is locked into personal mailbox:
 - 100 % of workers check email ≥ 1 per hour (study by Alliance Professionals)
 - Email is key content management tool but content is not shared
 - Not designed for long term archiving, re-use or curation
 - Information transfer inefficiencies; knowledge retention issue
- Contribution to corporate wikis is low:
 - Wikis are well adopted by enterprises
 - But very low contribution rate
 - Cost to share is too high:
 - No immediate benefit to share
 - Too many steps to contribute?
 - Ownership issues & lack of recognition



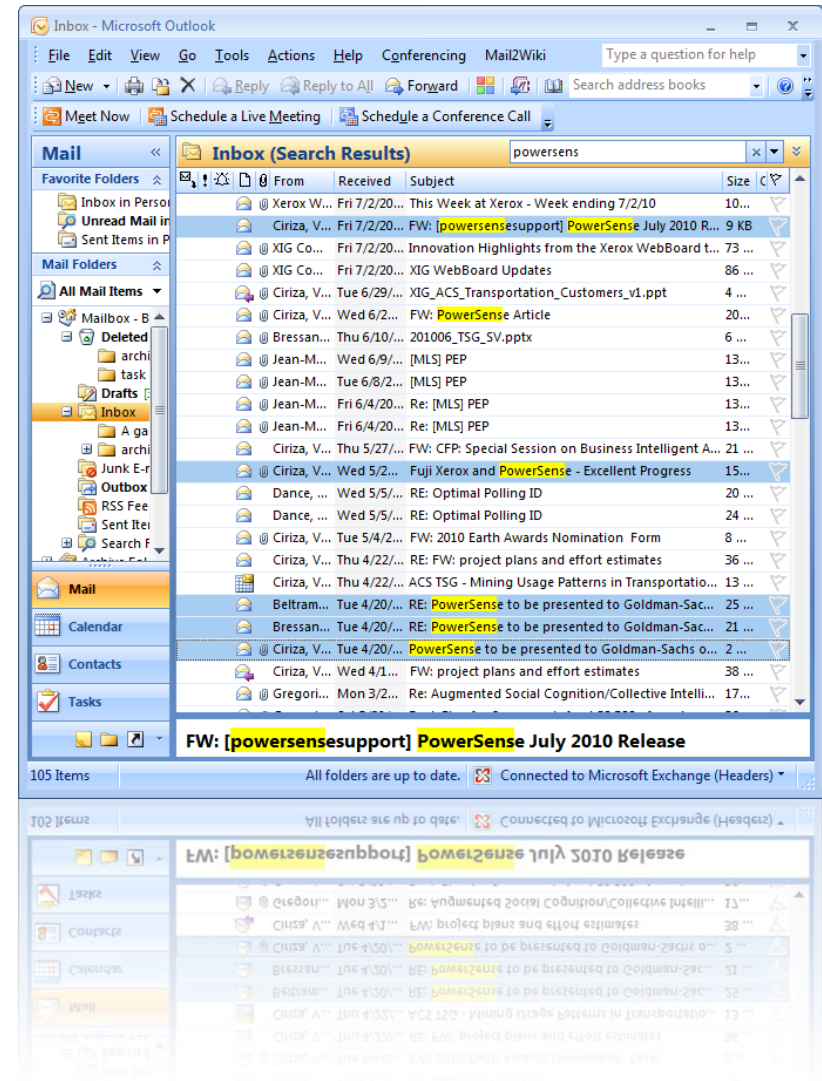
Mail2Wiki

- Semi-automatic assistance tool to transfer email content to wiki space:
 - Lightweight version (no plug-in)
 - Advanced version (outlook plug-in):
 1. Push mode: email client with a wiki view
 2. Pull mode: wiki with an email view
- System benefits:
 - Better and faster knowledge sharing
 - Faster content curation
 - Less duplication and easier re-use
 - Access to contextual information
- Advanced functionalities such as page organisation and create links
- Especially relevant in extended projects!



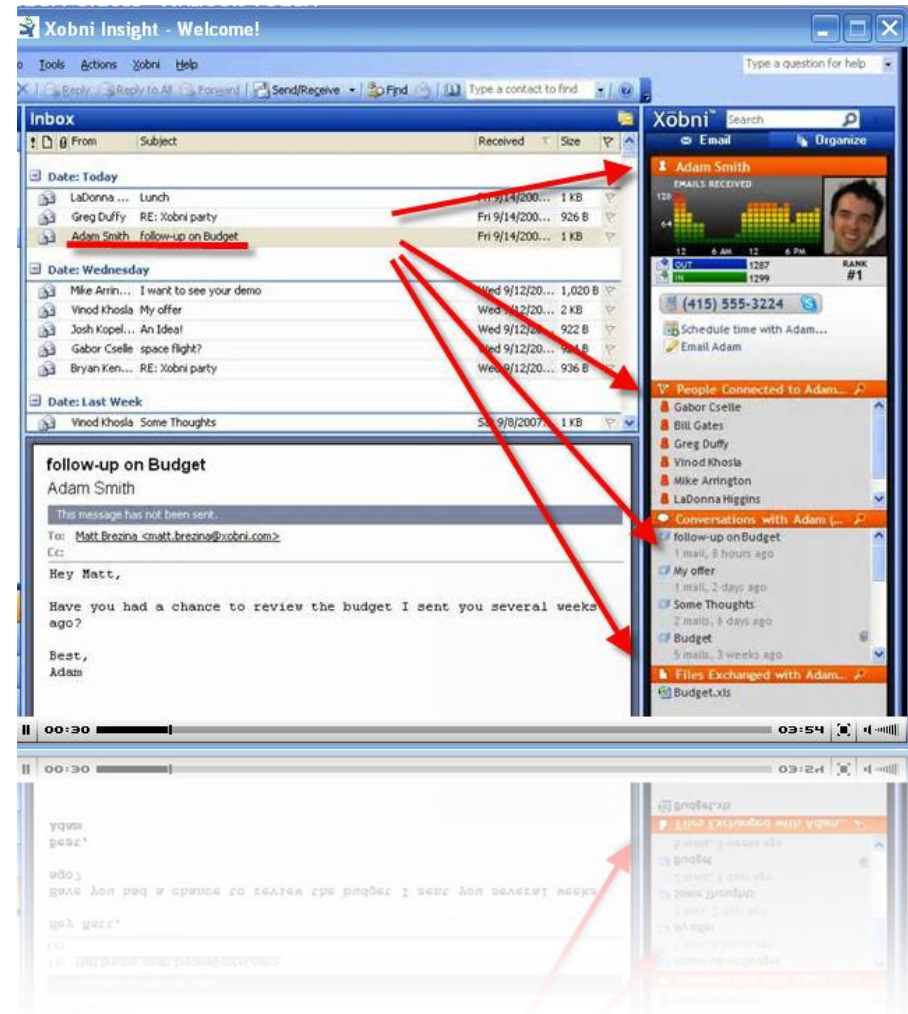
Usage example

- I'm knowledge worker X
- Our team worked last year on a project called Y
- This year we are less involved, but other teams are still working on Y
- We are informed by emails but no one has the time to read (remember) them
- An up to date project page would be really useful



Related tools

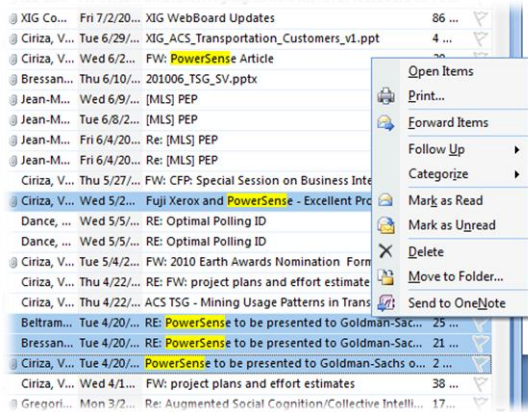
- XOBNI, MeshIn, etc.
 - Alternate personal view of mailbox
 - Pulls in external information (e.g. from social network)
 - Augment email client (email centripetal)
- Mail2Wiki:
 - Creates low-cost sharing space (social function)
 - Offloads re-usable content (email centrifugal)
 - Provide tools to organise and maintain shared content



Creating a wiki in 4 clicks (no plug-in)

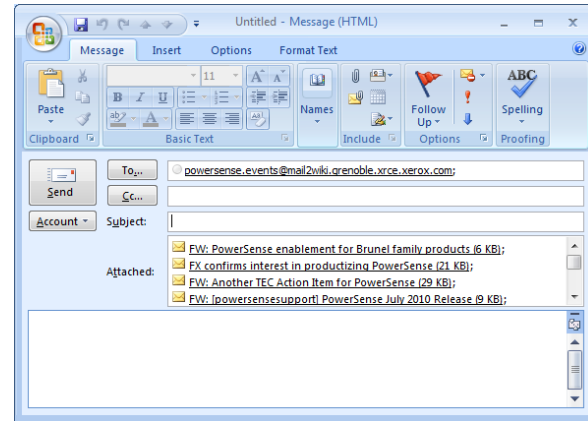
1

Select and forward emails



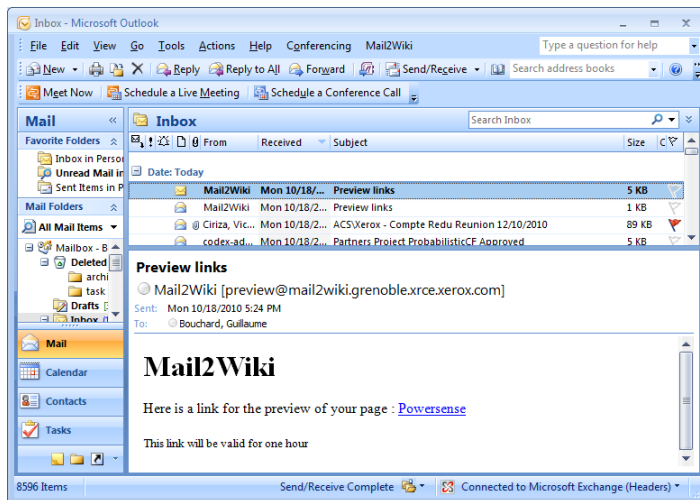
2

Choose a wiki name



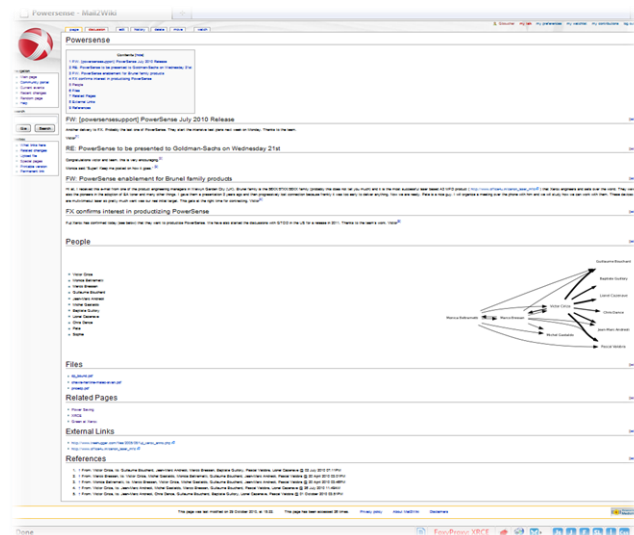
3

Receive a link to a preview page



4

Validate the wiki in the browser



Format of the automatically generated page

The screenshot shows a web browser window displaying a Mail2Wiki page titled "Powersense". The page layout includes a navigation bar at the top with tabs for "page", "discussion", "edit", "history", "delete", "move", and "watch". Below the navigation bar is a search box and a sidebar with navigation links. The main content area is divided into several sections: "Contents", "Text", "People", "Files", "Related Pages", "External Links", and "References". The "Text" section contains several email excerpts, including one from "FW: [powersensesupport] Powersense July 2010 Release" and another from "RE: Powersense to be presented to Goldman-Sachs on Wednesday 21st". The "People" section features a network diagram with nodes for various individuals and arrows indicating relationships. The "Files" section lists several PDF attachments. The "References" section contains a list of links to related documents.

- Main sections correspond to aggregated email content (emails with similar subjects appear in same section)
- People section: extracted names + social network (width of the link proportional to number of emails)
- Files section: lists all attachments of the email batch (attachments are uploaded and sent to the author)
- Related pages section: pages with similar content (automatically computed using a topic model)
- External links section: all URLs mentioned in emails
- References: meta-information on emails as footnotes



Advanced features (outlook plug-in)

- Sidebar with recommended wiki pages
- Wiki page outline and recommended sections
- Drag & drop of email content
- Fast and easy

The screenshot displays the Outlook interface with the Mail2Wiki plug-in. The left pane shows a search for 'A - Evaluation' with results for 'Hanrahan, Ben <Ben.Hanrahan@parc.com>' dated Wed 7/28. The center pane shows an email from Hanrahan, Ben, with the subject 'For NLP - Lit Review - Using NLP to Augment Wikis'. The right pane shows a 'Mail2Wiki' sidebar with a list of recommended pages, including 'NLP-Literature Review' which is highlighted. The bottom pane shows a detailed view of an email from Gregorio Convertino, with the subject 'Re: shared schedule & todos'. The email content includes a shared schedule and todos, and a list of recommended pages on the right side of the email view.

Recommendation module (logistic regression)

- Goal: recommend the pages for the email (or selection)
- Formalised as a binary classification problem per wiki page
- Generalised linear model:

$$P(C_1|\mathbf{x}_n) \equiv y(\mathbf{x}_n) = \sigma(\mathbf{w}^\top \phi(\mathbf{x}_n))$$

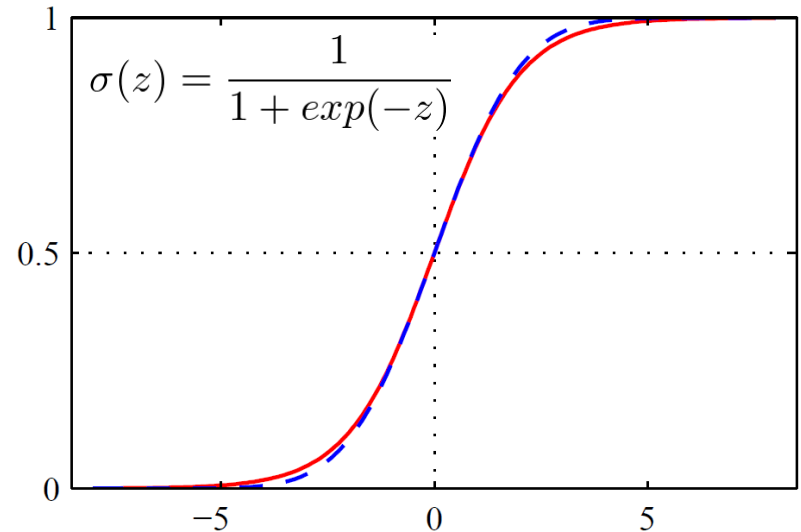
- Users implicitly label the data:

$$p(\mathbf{t}|\mathbf{y}, \mathbf{w}) = \prod_n y_n^{t_n} (1 - y_n)^{1-t_n}$$

- Pages are ranked according to their posterior probabilities
- Features include email meta-data, topics, BM25, etc.
- Iterative reweighted least squares (Newton method):

$$\mathbf{w}^* = \underset{\mathbf{w}}{\operatorname{argmax}} \ln p(\mathbf{t}|\mathbf{y}, \mathbf{w}) \quad \mathbf{w} \leftarrow \mathbf{w} - \mathbf{H}^{-1} \nabla \ln p(\mathbf{t}|\mathbf{y}, \mathbf{w})$$

- Maximum likelihood can lead to overfitting if data is linearly separable!



TF-IDF and BM25

TF-IDF:

- Bag-of-words assumption (word sequence ignored)
- Document ranking function popular in information retrieval
- Term frequency (how important is word v in a document d):

$$\text{tf}_{vd} = \frac{n_{vd}}{n_{.d}}$$

- Inverse document frequency (relative importance of word v):

$$\text{idf}_v = \log \frac{D}{n_d(v)}, \quad n_d(v) = |\{d : n_{vd} > 0\}|$$

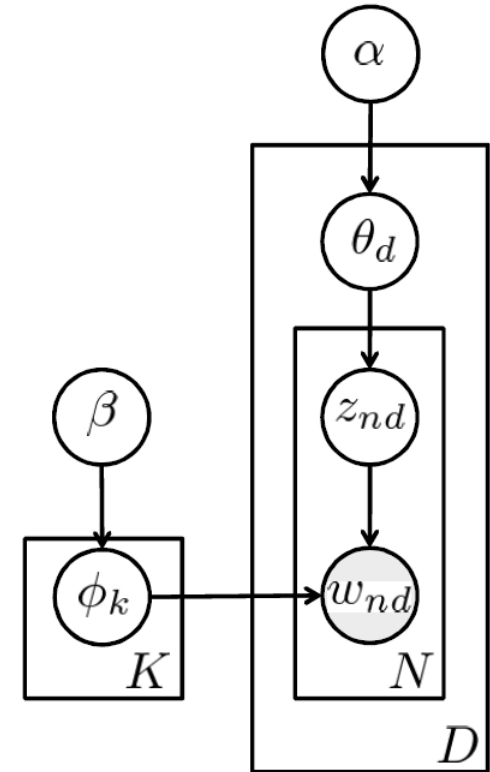
BM25 :

$$s_{vd} = \text{idf}_v \frac{\text{tf}_{vd}(a+1)}{\text{tf}_{vd} + a(1 - b + b \frac{n_{.d}}{n_{..}/D})}, \quad \text{idf}_v = \log \frac{D - n_d(v) + 0.5}{n_d(v) + 0.5}$$

$$(0 \leq a, 0 \leq b \leq 1)$$

Topic model (latent Dirichlet allocation)

- Goal: organise wiki pages and compute email features
- Bag-of-words assumption
- Generative model for documents
 1. Define every topic by drawing a vector of vocabulary proportions
 2. For every document:
 - Draw a Poisson number of words
 - Draw a vector of topic proportions
 3. To generate a word in the document:
 - First draw its topic indicator
 - Then draw a word from that topic distribution
- Unsupervised method to organise large text corpora
- Captures semantic information (themes/topics)
- Known in statistics as an ad mixture model:
 - Every data point (\sim document) is a mixture
 - Components are shared, but weight vector per data point



Latent Dirichlet allocation (Blei et al., JMLR 2003)

- Generative model:

$$\phi_k \sim \text{Dirichlet}(\beta \mathbf{1}_V),$$

$$N_d \sim \text{Poisson}(\lambda),$$

$$\theta_d \sim \text{Dirichlet}(\alpha \mathbf{1}_K),$$

$$z_{nd} | \theta_d \sim \text{Discrete}(\theta_d),$$

$$w_{nd} | z_{nd}, \{\phi_k\} \sim \text{Discrete}(\phi_{z_{nd}}),$$

- Gibbs sampler (Bayesian statistics):

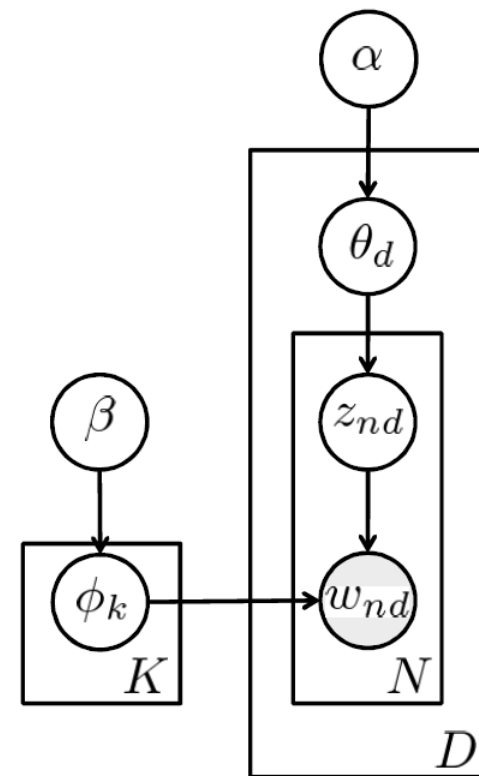
- Special case of Metropolis-Hastings algorithm
- Integrate out topic and vocabulary proportions
- Use exchangeability of words (de Finetti's theorem)

$$P(z_{id} = k | \mathbf{z}^{\setminus id}, \mathbf{w}) = \frac{P(\mathbf{w}, \mathbf{z})}{P(\mathbf{w}, \mathbf{z}^{\setminus id})} \propto \frac{(\alpha + n_{\cdot kd}^{\setminus id})(\beta + n_{vk}^{\setminus id})}{V\beta + n_{\cdot k}^{\setminus id}}$$

- Non-negative matrix factorisation interpretation:

$$\mathbb{E}\{\mathbf{V}\} \approx \Phi \Theta, \quad \Phi \in \mathbb{R}^{V \times K}, \quad \Theta \in \mathbb{R}^{K \times D}$$

- How to select the number of topics?



Conclusion

- Mail2wiki involves a lot of clever software engineering
 - Scalability issues are key component (users and data)
 - Natural interfaces and seamless interaction
 - Many simple rule-based tricks can do a good job
- More advanced functionalities require computational machine learning
 - Recommendation
 - Topic model
 - Structured prediction
 - Personalisation
 - Meta-page creation
 - ...
- Convenient test bed to experiment new algorithms!

Recent references:

1. Hanrahan et al., Mail2Wiki email plugin, design rationale and system architecture. Communities and Technologies 2011.
2. Kong et al., VisualWikiCurator: A Corporate Wiki Plugin. CHI 2010.

Generic visual toolbox

Large-scale image search and categorisation

Work by Florent Perronnin (XRCE) and Chris Dance (XRCE)



Image classification benchmarks

- Popular benchmarks:
 - CalTech 101 (2003): 101 classes, <9K images
 - CalTech 256 (2006): 256 classes, 30K images
 - PASCAL VOC 2007: 20 classes, 10K images
- Recent benchmarks:
 - ILSVRC 2010: 1K classes , 1.46M images
 - Full ImageNet dataset: 11.6K classes, 11.2M images
- How to efficiently learn classifiers on such large quantities of data?
 - Non-linear (dual): learning cost between $O(N^2)$ and $O(N^3)$
 - Linear (primal): learning cost scales in $O(N)$

Image classification: the state-of-the-art

- Bag-of-visual-words (BOV) representation:
 - Images described by histograms of quantised local features
 - Popular features are SIFT (scale-invariant feature transform)
 - Colour features
 - ...
- Kernel classification use SVMs with non-linear kernels:

Additive Kernels:

$$K_{bha}(a, b) = \sum_{i=1}^N \sqrt{a_i b_i},$$
$$K_{chi2}(a, b) = 2 \sum_{i=1}^N \frac{a_i b_i}{a_i + b_i},$$
$$K_{int}(a, b) = \sum_{i=1}^N \min(a_i, b_i).$$

Exponential Kernels:

$$K_{bha}^{exp}(a, b) = \exp \left(-\frac{\gamma}{2} \sum_{i=1}^N (\sqrt{a_i} - \sqrt{b_i})^2 \right),$$
$$K_{chi2}^{exp}(a, b) = \exp \left(-\frac{\gamma}{2} \sum_{i=1}^N \frac{(a_i - b_i)^2}{a_i + b_i} \right),$$
$$K_{int}^{exp}(a, b) = \exp \left(-\gamma \sum_{i=1}^N |a_i - b_i| \right).$$

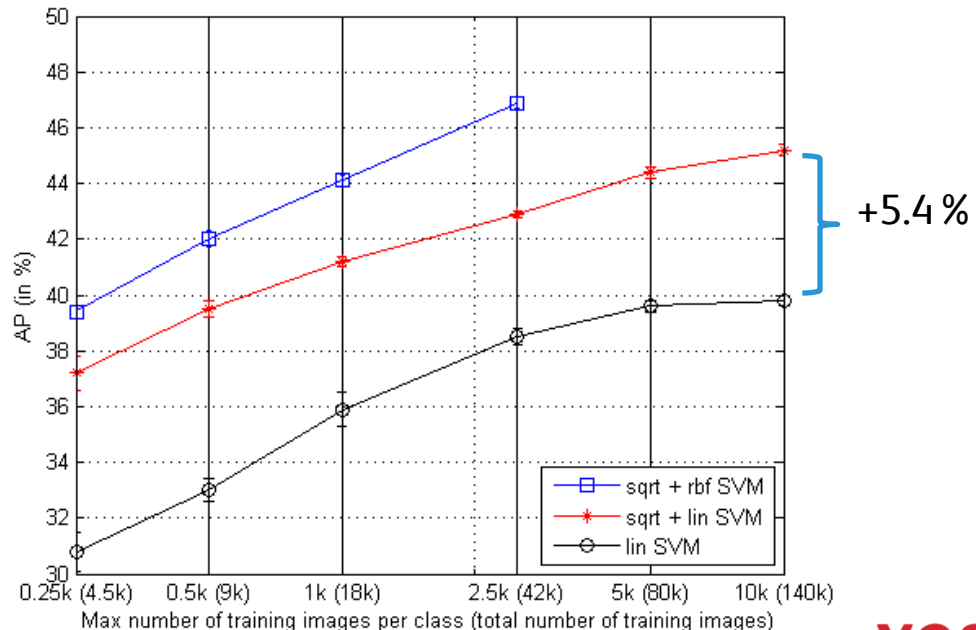
Kernel trick

- Mercer kernel corresponds to implicit embedding:

$$K(x, z) = \varphi(x)' \varphi(z)$$

- Non-linear classification in the original space corresponds to linear classification in the new space
- Perform an approximate explicit embedding of the data
- Learn a linear classifier in the new space (optimise in primal)
- E.g. Bhattacharyya kernel:

- $\varphi(z) = \sqrt{z}$
- More data helps!
- Training from ImageNet
- Test VOC 2007



Fisher kernel framework (Jaakkola and Haussler, NIPS 11)

- The Fisher information measures the amount of information carried by a sample X about a parameter λ under the model p :

$$F_\lambda = E_{x \sim p} [\nabla_\lambda \log p(x|\lambda) \nabla_\lambda \log p(x|\lambda)']$$

- Model a sample X by its deviation from a probabilistic model p :

$$G_\lambda^X = \nabla_\lambda \log p(X|\lambda)$$

- Measure the similarity using the “Fisher kernel”:

$$K(X, Y) = G_\lambda^{X'} F_\lambda^{-1} G_\lambda^Y$$

- This is a dot product on normalized “Fisher vectors” (FV):

$$\mathcal{G}_\lambda^X = L_\lambda G_\lambda^X \quad \text{with} \quad F_\lambda = L_\lambda' L_\lambda$$

- Learning the kernel classifier is equivalent to learning the linear classifier on FV!

Representing images with Fisher vectors (Perronnin & Dance, CVPR 2007)

- Consider D -dimensional local descriptors (e.g. SIFT) extracted from an image and a (diagonal) GMM:

$$X = \{x_t, t = 1 \dots T\}$$

$$p(x) = \sum_{i=1}^N w_i p_i(x)$$

$$\lambda = \{w_i, \mu_i, \Sigma_i, i = 1 \dots N\}$$

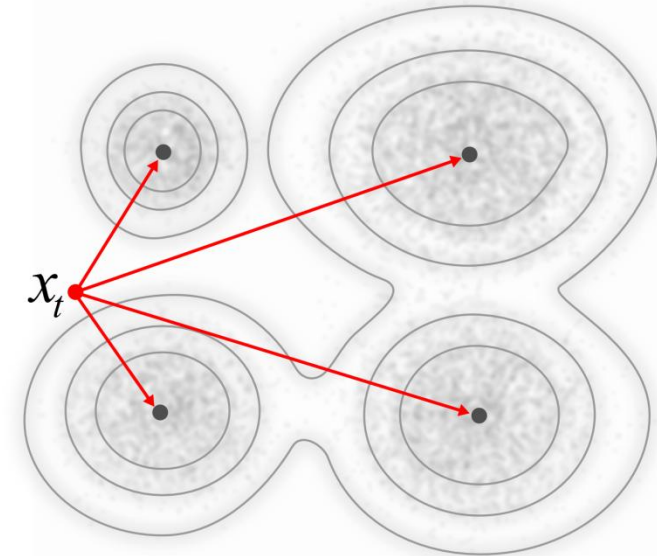
- The mixture parameters are trained on a large set of local descriptors (~visual vocabulary)
- Gradient vectors are given by

$$\mathcal{G}_{\mu,i}^X = \frac{1}{T\sqrt{w_i}} \sum_{t=1}^T \gamma_t(i) \left(\frac{x_t - \mu_i}{\sigma_i} \right)$$

$$\gamma_t(i) = \frac{w_i p_i(x_t)}{\sum_{j=1}^N w_j p_j(x_t)}$$

$$\mathcal{G}_{\sigma,i}^X = \frac{1}{T\sqrt{2w_i}} \sum_{t=1}^T \gamma_t(i) \left[\frac{(x_t - \mu_i)^2}{\sigma_i^2} - 1 \right]$$

- Fisher vectors obtained by concatenation: $2ND$ -dimensional



Why do Fisher vectors make sense?

- A large mixture weight corresponds to a background (i.e. frequent) visual word
- Assume responsibilities are close to one (hard assignment):

$$\gamma_t(i) \approx 1$$

- Image descriptor (mean and variance) is only significant if far from a background visual word:

$$\left\| \frac{x_t - \mu_i}{\sigma_i} \right\|^2 \text{ is large}$$

- Images are described by what makes them different from the others!

Improving the Fisher vector representation

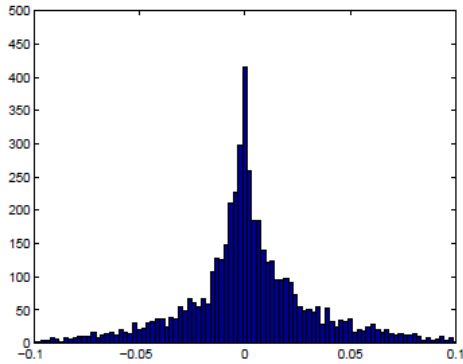
- L2 normalisation:
 - Rescaling of the gradient vector components
 - The image-independent part is automatically discarded from the FV
 - Removes dependence on the *amount* of image-independent information
- Power normalisation:

$$f(z) = \text{sign}(z)|z|^\alpha \text{ with } 0 \leq \alpha \leq 1$$

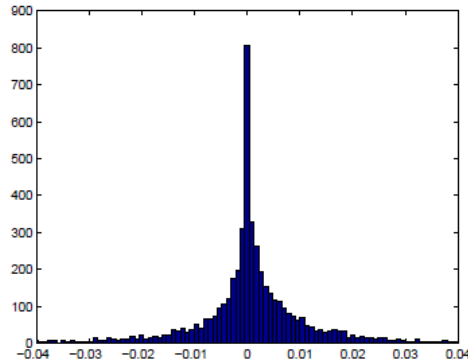
- Reduce influence of bursty elements (account for heavy tail)
- “Unsparsify” FV to prevent the dot-product to become a poor similarity measure
- Near optimal α value for $16 \leq N \leq 256$ is 0.5
- Spatial pyramid: one FV per image region and concatenate ($R = 1+4+3$):



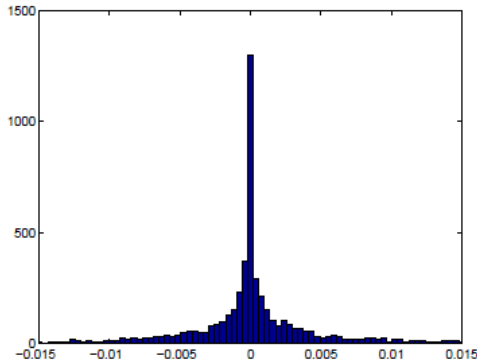
Illustration of power normalisation



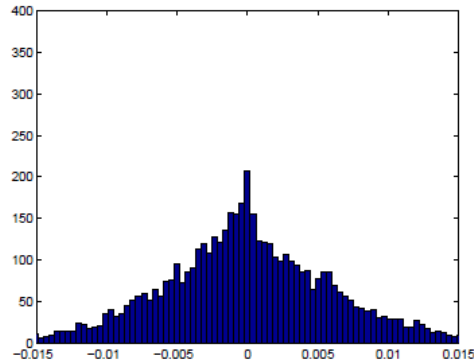
(a)



(b)



(c)



(d)

Distribution of Fisher vector values (first dimension) as estimated on VOC 2007.

(a) 1 Gaussian

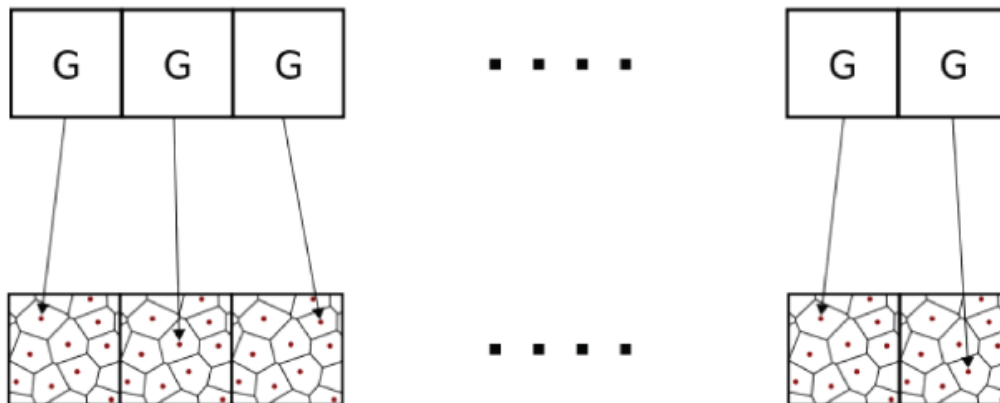
(b) 16 Gaussians

(c) 256 Gaussians

(d) 256 Gaussians with normalisation

Fisher vector compression

- Problem: FVs are dense and of dimension $E = 262,144$ ($D = 64, N = 256, R = 8$)
 - 4-byte floating point encoding leads to 1MB per image
 - Storing ILSVRC 2010 would require 1.4TB (per low-level feature type)
- Dimensionality reduction:
 - Dense projections (e.g. PCA) are too costly
 - Sparse projections (e.g. Hash Kernel) lead to significant decrease in accuracy
- Vector quantisation (K -means) is infeasible: codebook cardinality is $O(2^{bE})$
- Product quantisation: cardinality of codebook is $O(2^{bG})$
 - Split FV into small sub-vectors (e.g. G)
 - Perform vector quantisation on each sub-vectors
 - Represent FV as a vector of codebook indices



Classification results

- Experimental setup:
 - Grid sampling, 256 Gaussians
 - SIFT and colour descriptors + PCA ($D = 64$)
 - Linear SVM learned with Bottou's stochastic gradient decent
- Small scale experiment:

PASCAL VOC 2007: \approx 10K images (5K training + 5K test) of 20 classes

→ no compression applied

Accuracy measured with mean Average Precision (mAP).

Table 1: Impact of Power Normalization (PN), L2-normalization (L2) and Spatial Pyramids (SP) on VOC 2007.

PN	L2	SP	SIFT	Col	S+C
-	-	-	47.9	34.2	45.9
✓	-	-	54.2	45.9	57.6
-	✓	-	51.8	40.6	53.9
-	-	✓	50.3	37.5	49.0
✓	✓	✓	58.3	50.9	60.3

[Zhou *et al.* , ECCV'10] report 64.0% with closely related super-vector and more costly dense SIFT extraction.

Large-scale classification results

PASCAL VOC 2010: \approx 20K images (10K training + 10K test) of 20 classes.
Also downloaded \approx **1M images** from 18 Flickr groups (no manual labeling).
Accuracy measured using mAP.

Table 2: Learning from different training resources: V = VOC 2010, F = Flickr groups, V+F = late fusion of V and F. NUSPSL designates the VOC 2010 winner.

Train	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow
V	87.1	59.6	59.9	69.7	31.3	76.4	62.9	64.3	52.5	42.4
F	92.7	68.0	69.0	79.9	29.3	81.4	60.0	78.0	45.0	62.9
V+F	92.7	68.4	68.5	80.4	38.2	81.8	66.9	77.8	55.0	62.1
NUSPSL	93.0	79.0	71.6	77.8	54.3	85.2	78.6	78.8	64.5	64.0

Train	table	dog	horse	moto	person	plant	sheep	sofa	train	tv	mean
V	55.1	59.7	64.3	70.4	83.9	32.6	53.3	50.4	80.0	67.6	61.2
F	31.6	69.2	71.2	78.6	78.0	34.0	67.3	-	82.7	-	-
V+F	56.5	70.1	71.4	79.4	85.0	40.0	67.2	51.8	84.6	67.6	68.3
NUSPSL	62.7	69.6	82.0	84.4	91.6	48.6	64.9	59.6	89.4	76.4	73.8

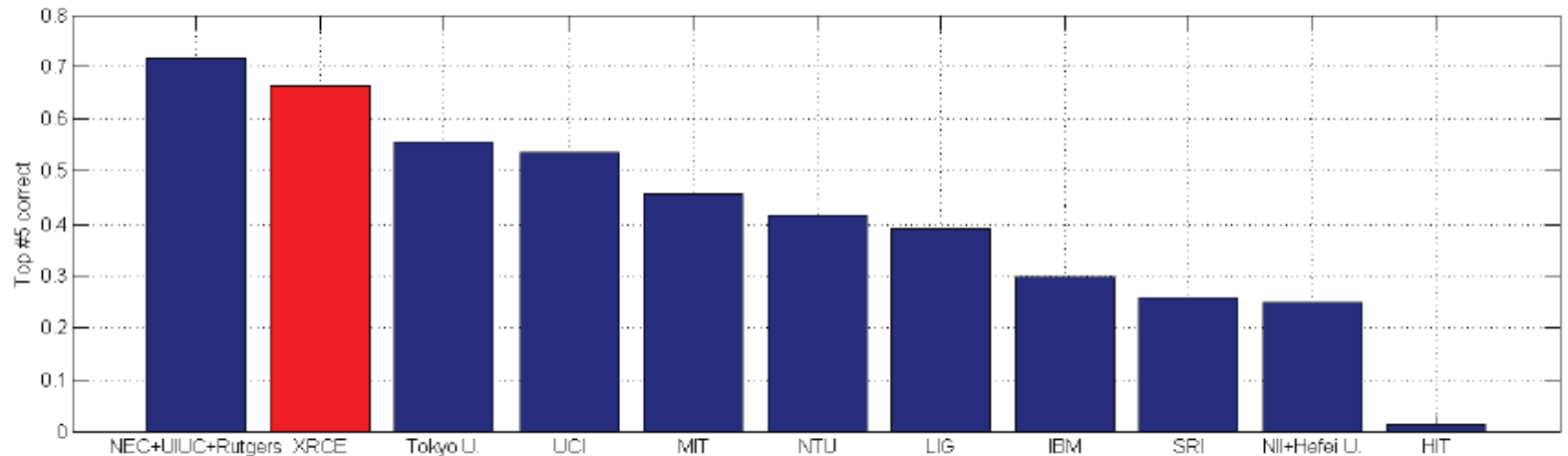
ImageNet large-scale visual recognition challenge

Imagenet Large-Scale Visual Recognition Challenge (ILSVRC) 2010:

≈ 1.4M images of 1,000 classes.

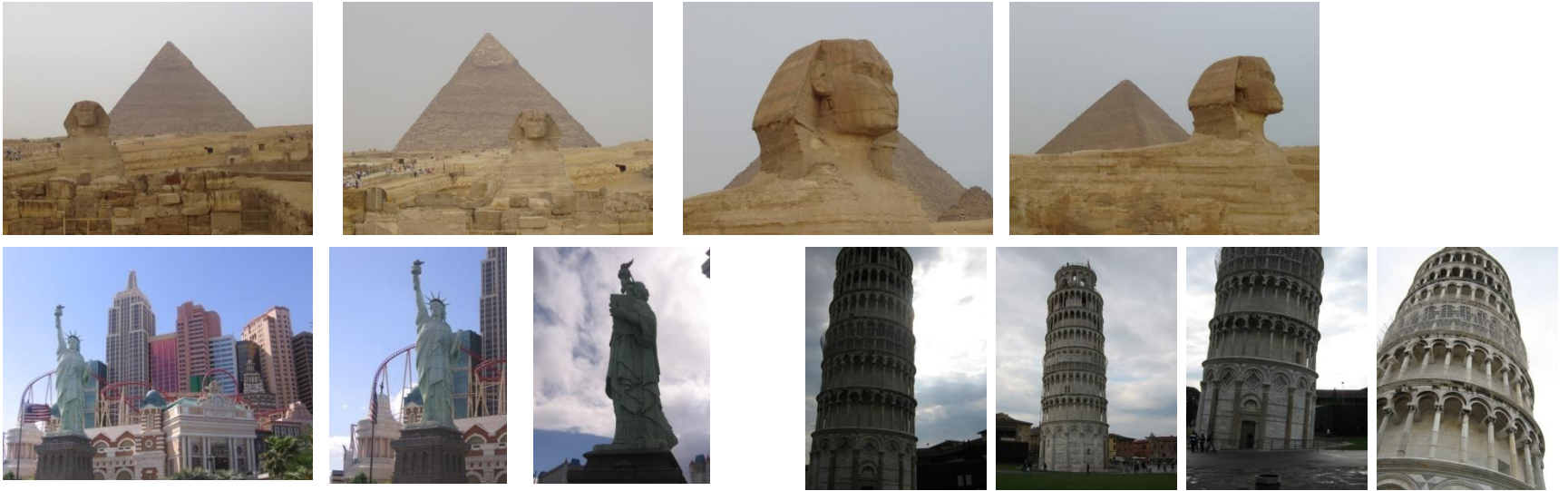
1.26M training + 50K validation + 150K test.

Accuracy measured as top #5 correct.



Retrieval results

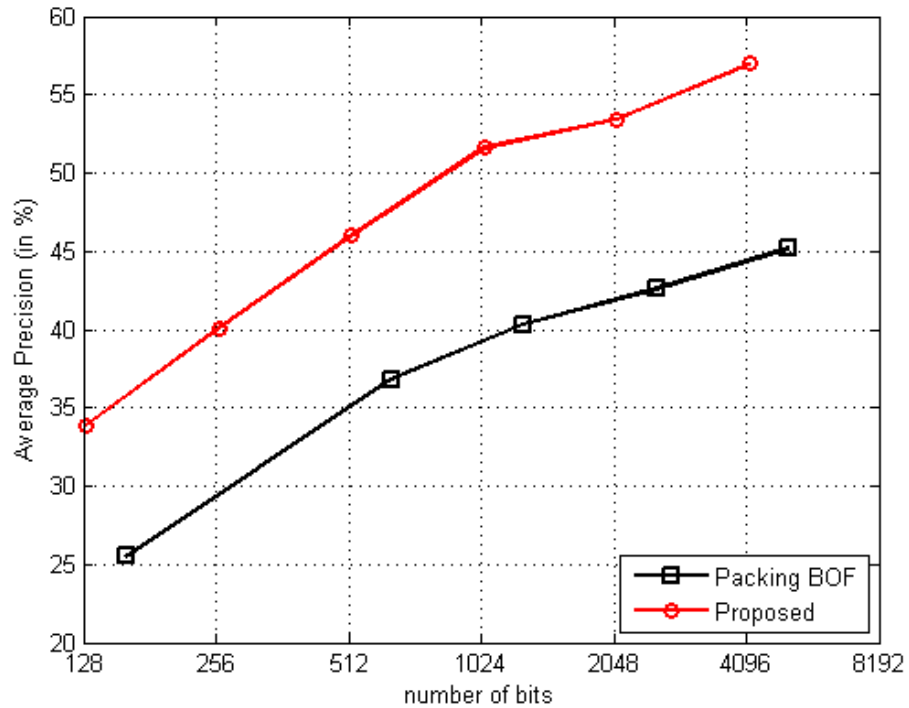
The INRIA Holiday dataset



- 1,491 images of 500 scenes/objects
- Variations in scaling, view point, lighting, occlusion, etc.
- Evaluation:
 - Query with one image, retrieve images of the same scene
 - Accuracy measured with Average Precision (AP)

Retrieval results

The INRIA Holiday dataset



Generic visual toolbox demo!

Conclusion

- Fisher vector framework:
 - State-of-the-art performance with linear classifier (high accuracy at low computational cost)
 - Large-scale image classification and large-scale image retrieval
- Extremely simple to implement: a few lines of codes on top of any GMM package
- Using compression techniques inspired by source coding, we can scale to datasets with millions of images
- Learning 20 classifiers on 350K Flickr groups images takes < 1 day on a single core Xeon machine:
 - 15h for SIFT + PCA extraction
 - 30 min to learn the GMM, 4h for FV extraction
 - 2h to learn 20 linear classifiers

Recent references:

1. F. Perronnin, Y. Liu, J. Sánchez and H.Poirier, “Large-scale image retrieval with compressed Fisher vectors”, CVPR 2010.
2. F. Perronnin, J. Sánchez and T. Mensink, “Improving the Fisher kernel for large-scale image classification”, ECCV 2010.

Interested in an internship or a PhD in the French Alps?

- XRCE (Grenoble, France): www.xrce.xerox.com
- Opportunities all year round, so don't hesitate to get in touch!



- Statistical machine learning: [myself](#) or Guillaume.Bouchard@xerox.com
- Statistical machine translation: Nicola.Cancedda@xerox.com
- Image retrieval and categorisation: Florent.Perronnin@xerox.com
- Mechanism design: Onno.Zoeter@xerox.com

