Canonical Image Selection for Large-scale Flickr Photos using Hadoop

Guan-Long Wu
National Taiwan University, Taipei

Nov. 10, 2009, @NCHC

Communication and Multimedia Lab (通訊與多媒體實驗室),
Department of Computer Science and Information Engineering, NTU (台大資訊系)
http://www.csie.ntu.edu.tw/~b95109

Note that parts of the slides are thanks to Prof. Winston Hsu and Liang-Chi Hsieh, *CMLab*, NTU

Team Members (MiRA group, *CMLab, NTU*)

- Prof. Winston H. Hsu

- Liang-Chi Hsieh
- Kuan-Ting Chen
- Chien-Hsing Chiang
- Yi-Hsuan Yang
- Guan-Long Wu
- Chun-Sung Ferng
- Hsiu-Wen Hsueh
- Angela Charng-Rurng Tsai

Who am I?

- A senior undergraduate student of NTU CSIE

- Research Interests
  - Multimedia (*CMLab, NTU.* Advisor: Winston H. Hsu)
  - Artificial Intelligence (*iAgent Lab, NTU*. Advisor: Jane Yung-jen Hsu)
  - Bioinformatics (*NYMU*. Advisor: Yeou-Guang Tsay)

- Contact ➡ *b95109@csie.ntu.edu.tw*

Outline

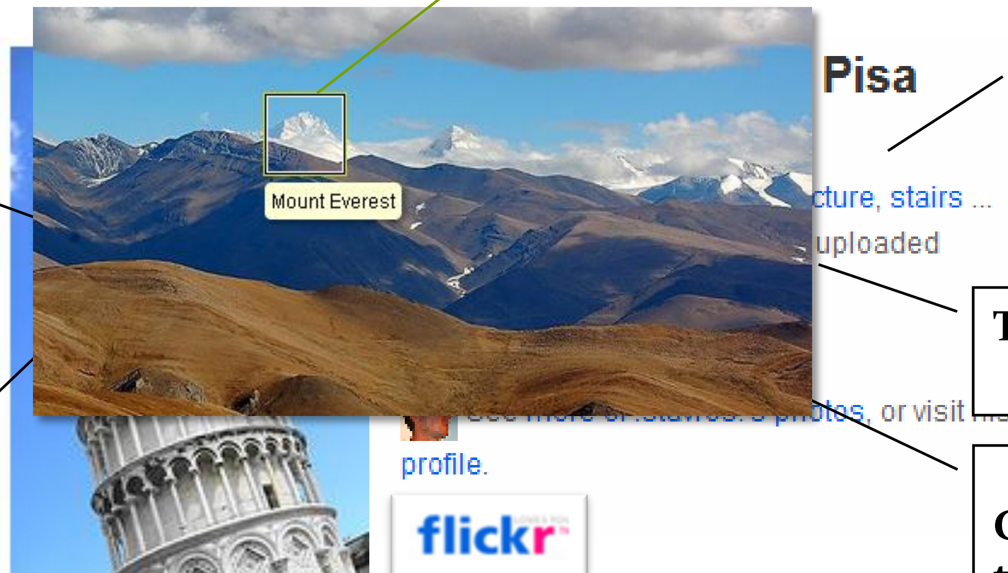
- Introduction – context cues in social media
- Efficient image search result clustering
- Demo
- Concept of Hadoop Implementation
  - Image Pairwise Image Similarity
  - Affinity propagation
- Comparing with previous approaches
- Conclusions

Challenges and Opportunities from Large-Scale Social Media

- Growing practice of online media sharing
- Billion-scale magnitude
- Bringing profound impacts to new applications and user scenarios
- The technologies do not keep pace with the growth
  - e.g., search, mining, visualization, and other promising applications

Rich Context Cues in Social Media – Flickr Example

notes – object-level comments



User-provided tags

Visual features
- color, texture
- visual word

Time stamp
yy/mm/dd

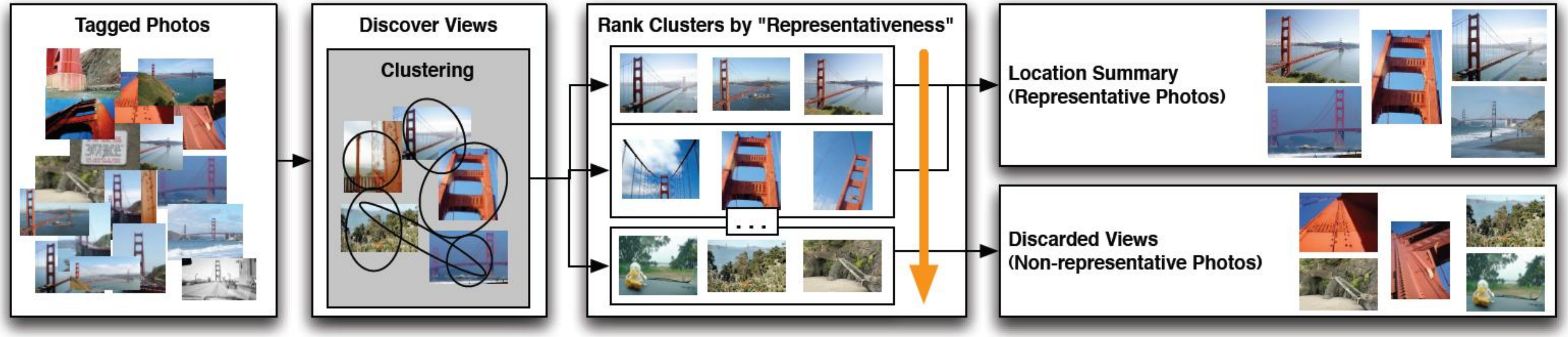
Concept detectors
(tower, sky, building)

Geo-tag

- Rich textual and visual cues, device metadata, and user interactions for social and organizing purposes
  - Geo-locations, time, camera settings (e.g., shutter speed, focal length, flash, etc.)
  - User-provided tags, descriptions, notes, etc.
  - Comments, bookmarks ,favorites (subjective)

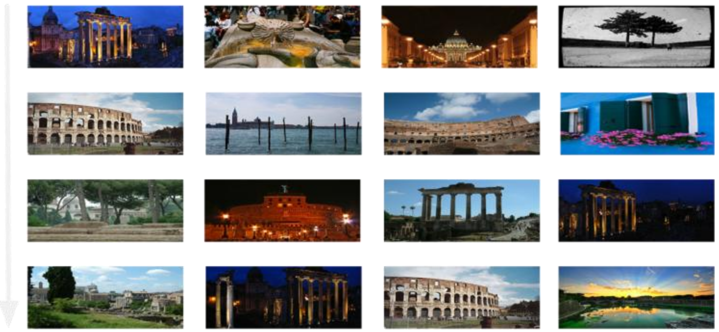
Social Media Visualization

- Select canonical views to represent a landmark [Kennedy et al., WWW'08]
  - Apply clustering algorithm (e.g. K-means) from tagged photos
  - Select one image from each cluster (assumed to be visually dissimilar)
- Extremely time-consuming and **NOT** for online image search result clustering
  - Pair-wise similarity
  - Clustering algorithms

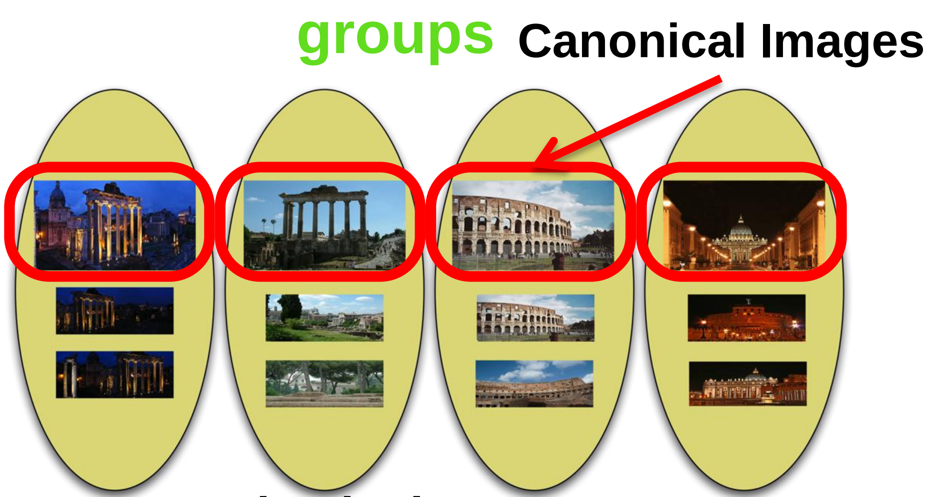


Efficient image search result clustering

	Current	Proposed
Feature	Keyword-based	Textual and visual-based
Organi-zation	N/A	Graph-based clustering
Display	Image list	Semantic image groups



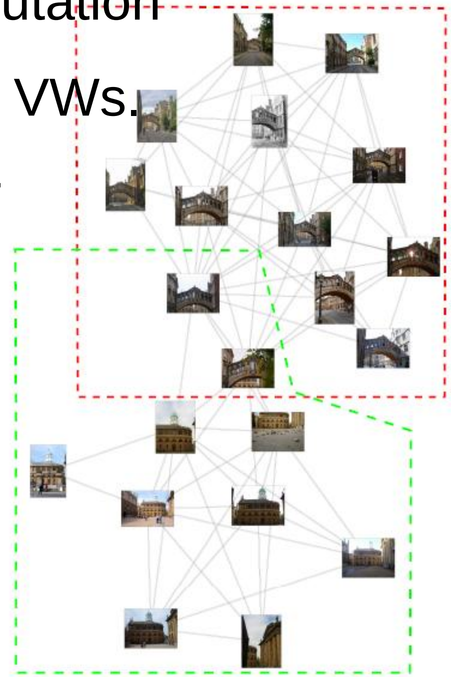
Text-based similarity



Browsing by image groups

Image Pairwise Image Similarity with MapReduce

- Goal – Speeding up image pairwise *cosine* similarity calculation by MapReduce (Hadoop) over large-scale images, represented by large VWs
- Constructing similarity "**hyperlinks**" in image collections for visualization and improving search quality; offline computation
- *tf-idf* cut is more powerful than *df-cut* when dealing with VWs.
- 69+ times speed-up over 18 Hadoop nodes with similar performance (MAP) (11K images with 10K VWs)



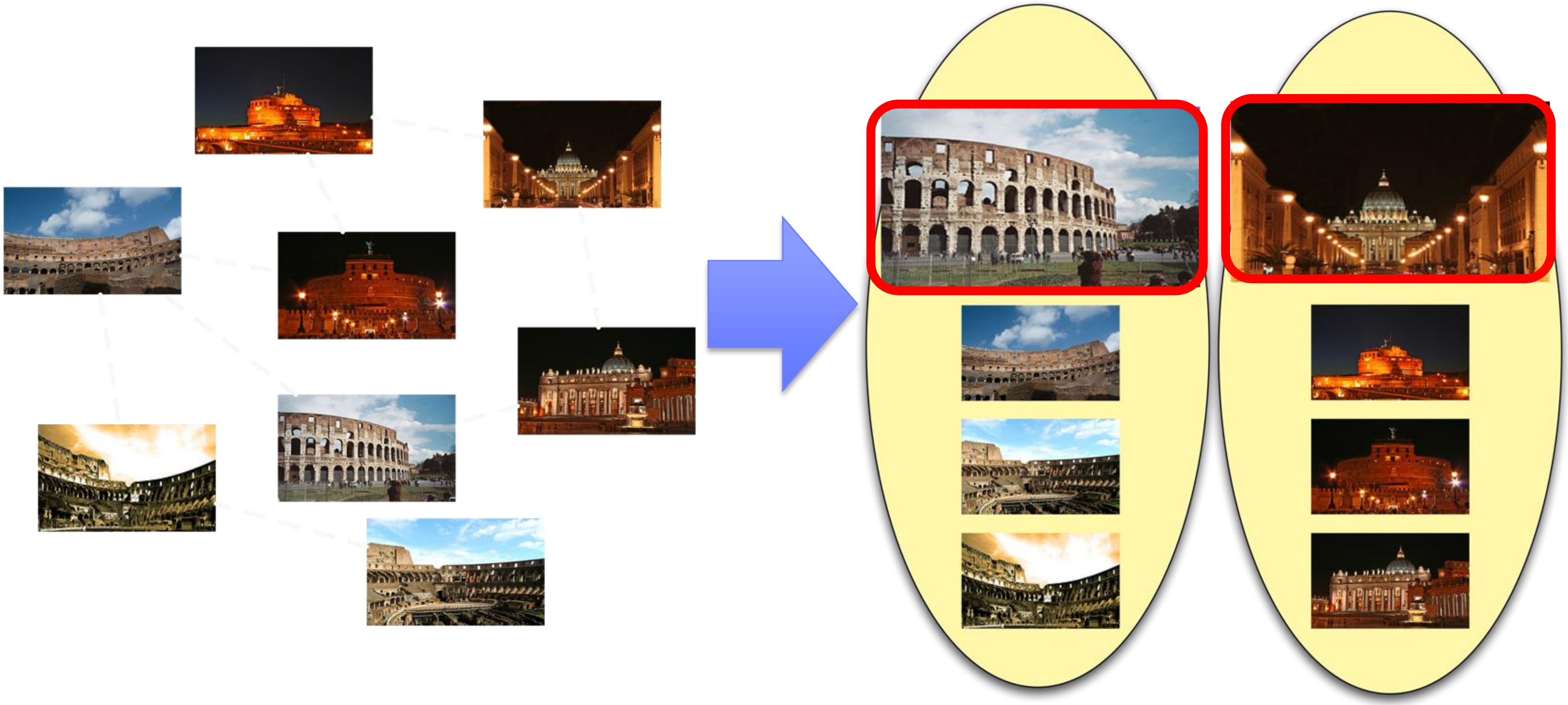
Cloud computing

- Leveraging MapReduce framework to scale up graph construction
- Computing huge image graph on a 18-node Hadoop cluster

dataset	Single machine	Hadoop Platform
Flickr11k	1.6hrs	83 secs
Flickr550k	unknown	42 mins

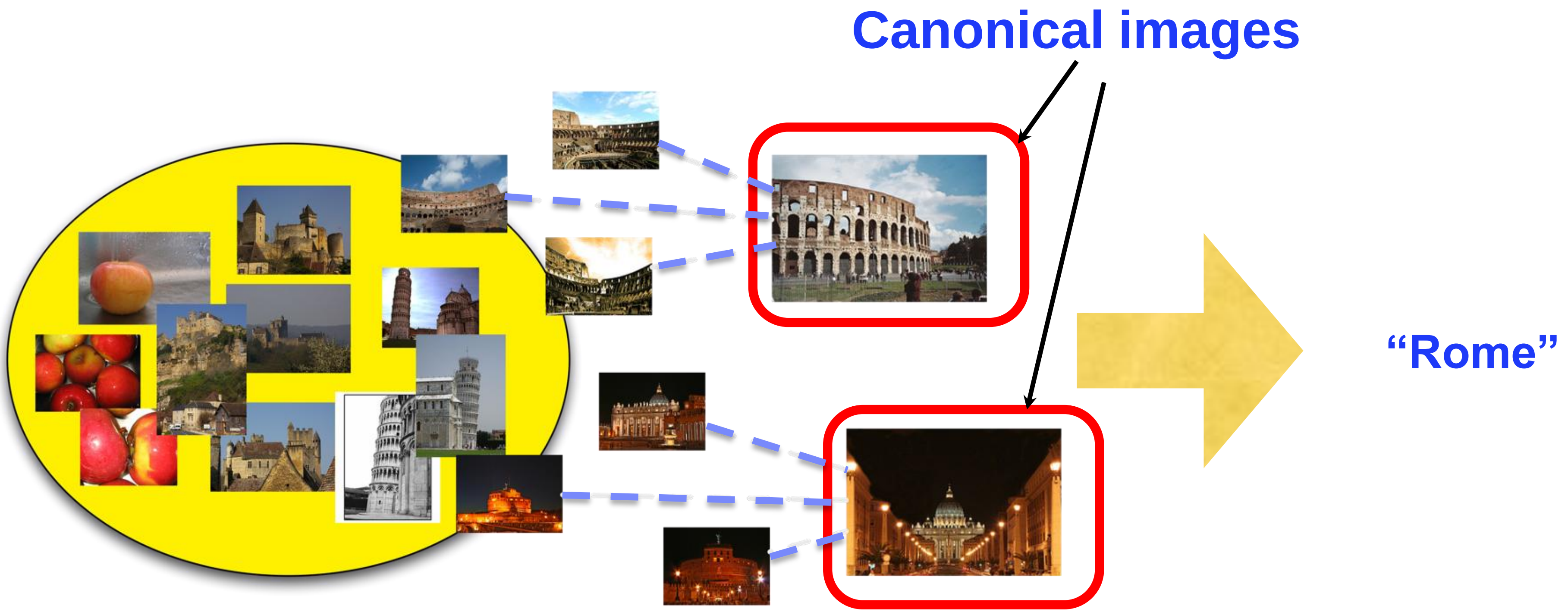
Offline clustering

- Clustering and canonical (representative) image selection by Hadoop-based Affinity Propagation



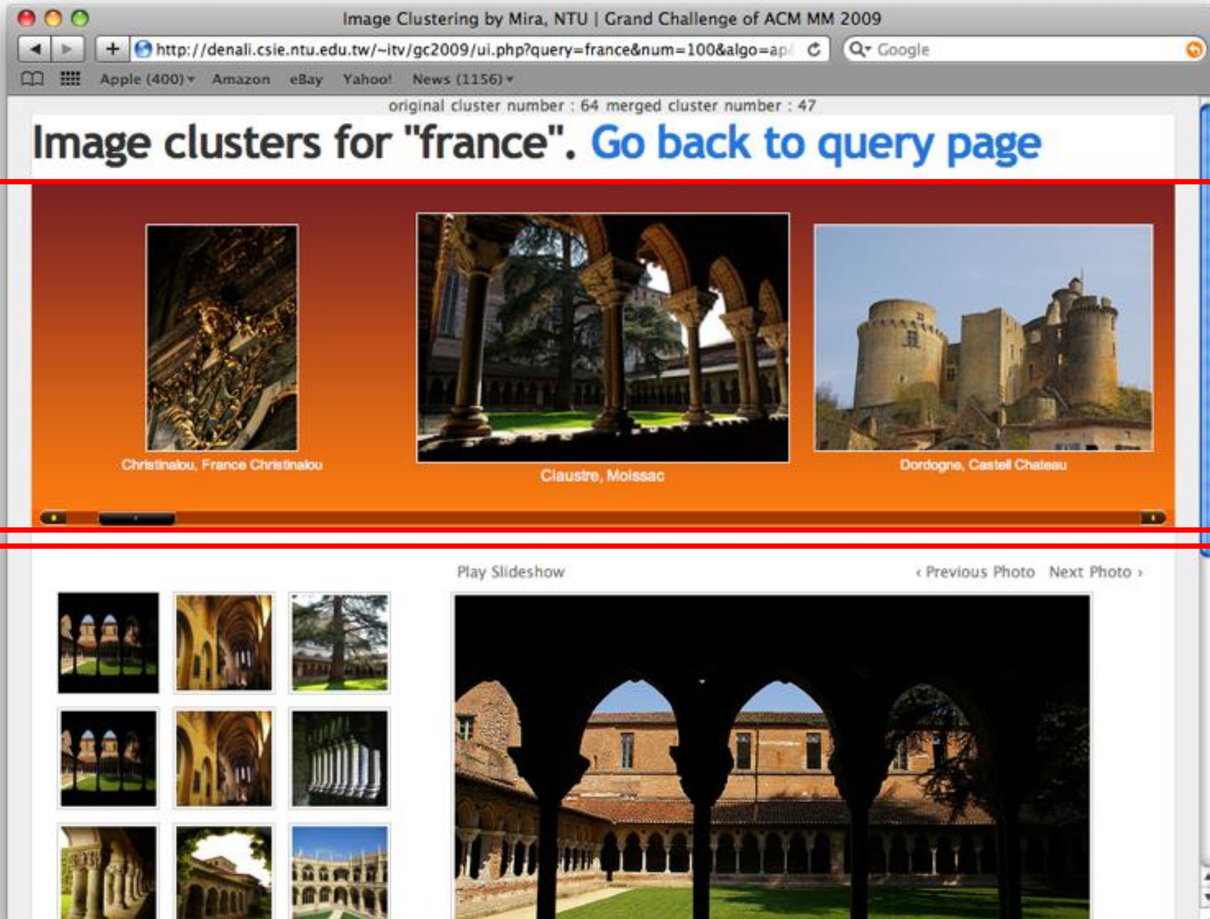
On-the-fly image search result clustering

- Real-time image search result clustering by pulling from pre-computed clusters



Demo!

demo



Canonical Images

Thumbnails and image viewer

Image Pairwise Image Similarity with MapReduce

- Indexing phase: vector → inverted index (utilize sparse vectors)

$$\text{similarity} = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|}.$$

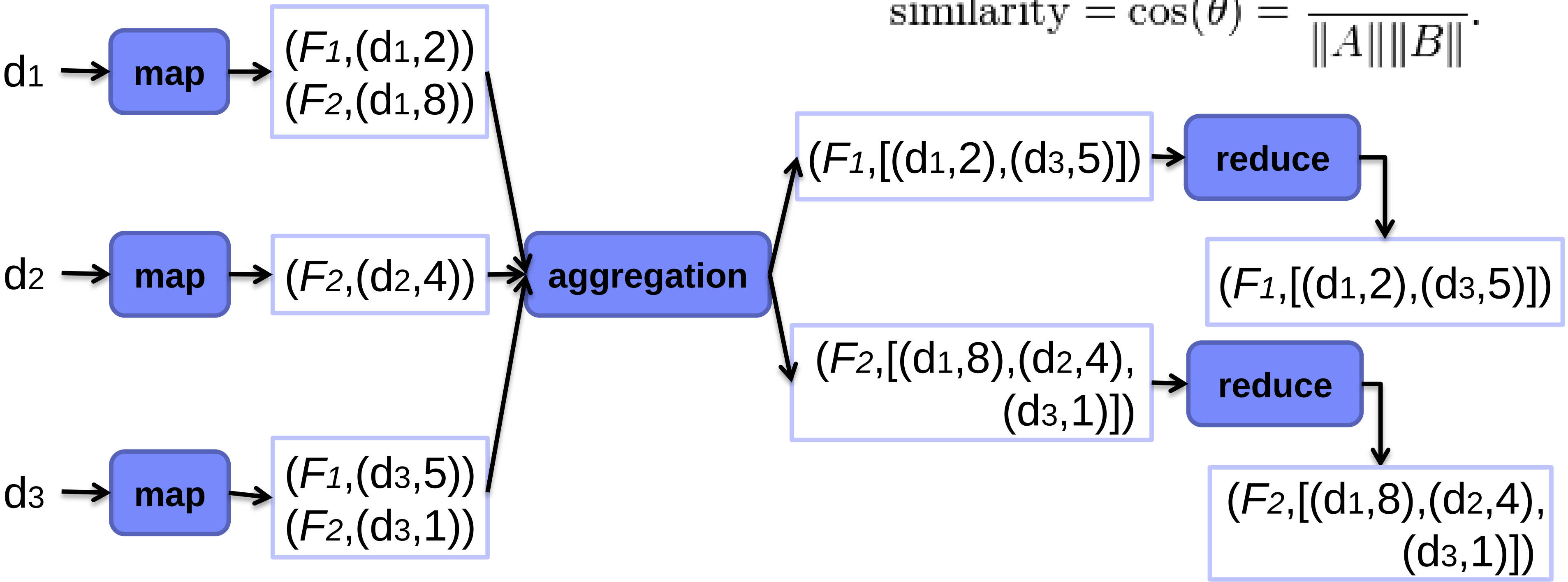
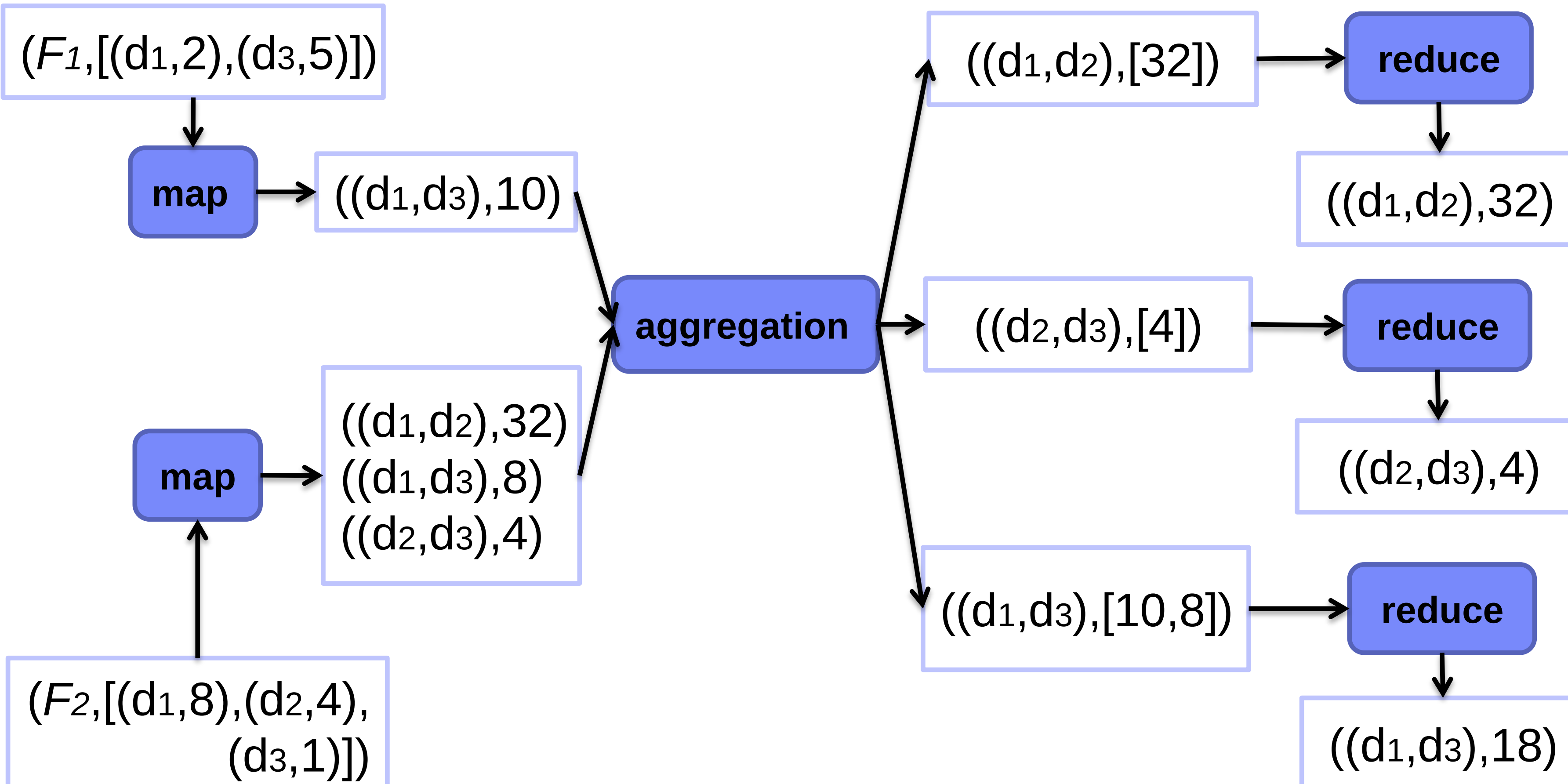


Image Pairwise Image Similarity with MapReduce

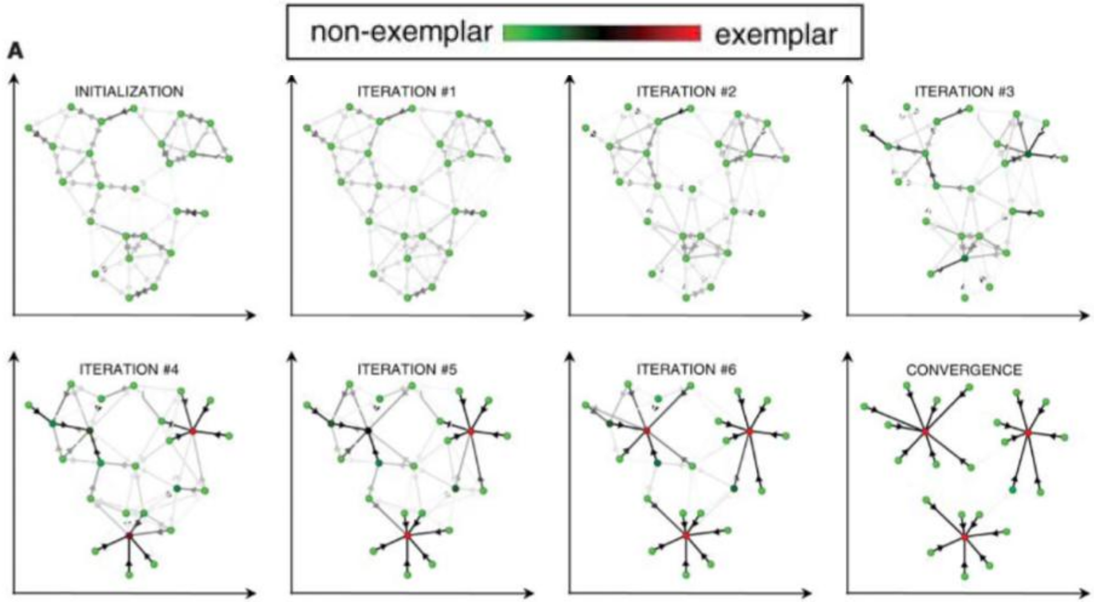
- Calculation phase: inverted index → pairwise similarity



Affinity propagation

[Frey et al., Science, 07]

- Data points can be exemplar (cluster center) or non-examplar (other data points).
- Message is passed between exemplar (centroid) and non-exemplar data points.
- The total number of clusters will be automatically found by the algorithm.



Hadoop Implementation of Affinity Propagation

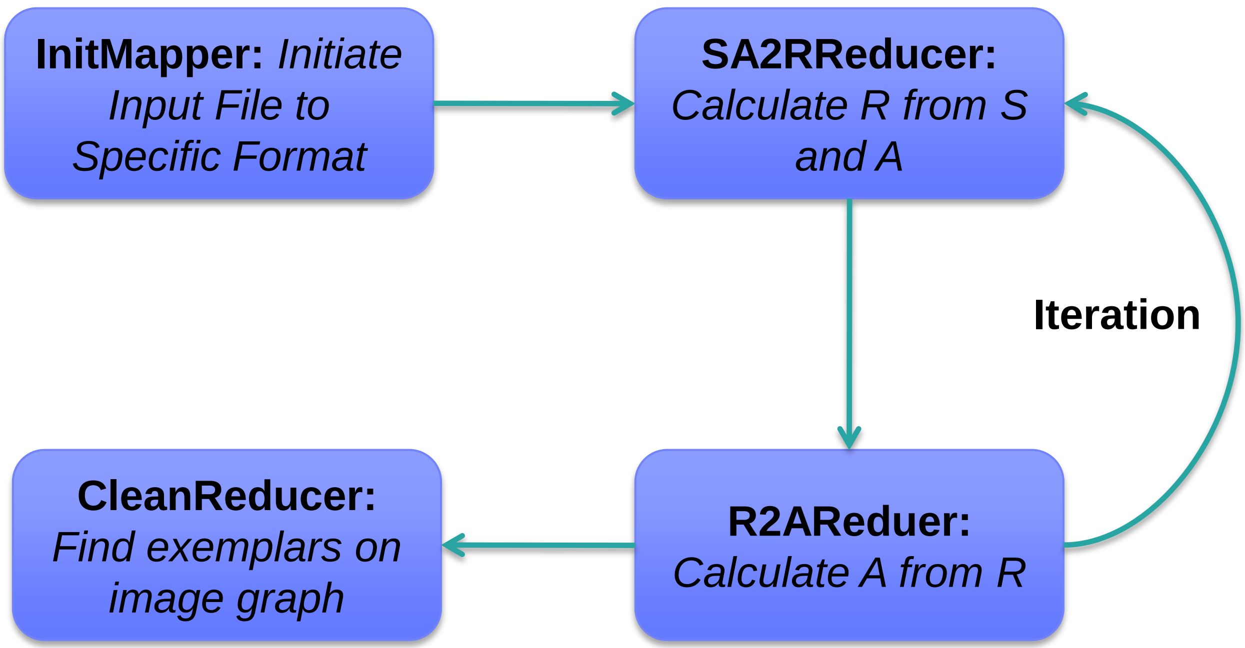
[Wang et al. ICHL 2008]

- S: similarity *s(i, k)*
- R: responsibility *r(i, k)*
- A: Availability *a(i, k)*

$$r(i,k) \leftarrow s(i,k) - \max_{k' \neq k}\{a(i,k') + s(i,k')\}$$

$$a(i,k) \leftarrow \min\left\{0, r(k,k) + \sum_{i' \neq i,k} \max\{0, r(i',k)\}\right\}$$

$$a(k,k) \leftarrow \sum_{i' \neq k} \max\{0, r(i',k)\}$$



Comparing with previous approaches

	Response Time	Feature	Scalability
SRC-based[1]	Fast	Textural only	No
Online-clustering[2]	Slow	Visual only	No
Our approach[3]	Faster	Textural and Visual	Yes

[1] Feng Jing et al., IGroup: web image search results clustering, *ACM MM 2006*

[2] Reinier H. van Leuken et al., Visual diversification of image search results, *WWW 2009*

[3] Hsieh et al., Canonical Image Selection and Efficient Image Graph Construction for Large-Scale Flickr Photos, *ACM MM 2009*

Conclusions

- The proposed system can organizing image search results in semantic clusters at query time.

- The efficiency is achieved with the help of offline-computed image context graphs by distributed computing methods.

Acknowledgements

- National Center for High-Performance Computing (NCHC), Taiwan, for the Hadoop platform and technical supports in cloud computing