Gathering and Ranking Photos of Named Entities with High Precision, High Recall, and Diversity

Bilyana Taneva
Max-Planck Institute for Informatics
Saarbrücken, Germany
btaneva@mpi-inf.mpg.de

Mouna Kacimi*
Free University of Bozen-Bolzano
Italy
Mouna.Kacimi@unibz.it

Gerhard Weikum
Max-Planck Institute for Informatics
Saarbrücken, Germany
weikum@mpi-sb.mpg.de

ABSTRACT

Knowledge-sharing communities like Wikipedia and automated extraction methods like those of DBpedia enable the construction of large machine-processible knowledge bases with relational facts about entities. These endeavors lack multimodal data like photos and videos of people and places. While photos of famous entities are abundant on the Internet, they are much harder to retrieve for less popular entities such as notable computer scientists or regionally interesting churches. Querying the entity names in image search engines yields large candidate lists, but they often have low precision and unsatisfactory recall.

Our goal is to populate a knowledge base with photos of named entities, with high precision, high recall, and diversity of photos for a given entity. We harness relational facts about entities for generating expanded queries to retrieve different candidate lists from image search engines. We use a weighted voting method to determine better rankings of an entity's photos. Appropriate weights are dependent on the type of entity (e.g., scientist vs. politician) and automatically computed from a small set of training entities. We also exploit visual similarity measures based on SIFT features, for higher diversity in the final rankings. Our experiments with photos of persons and landmarks show significant improvements of ranking measures like MAP and NDCG, and also for diversity-aware ranking.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms

Algorithms, Experimentation

*This work has been done at the Max-Planck Institute for Informatics

WSDM'10, February 4–6, 2010, New York City, New York, USA.

1. INTRODUCTION

1.1 Motivation

Advances in automatic information extraction and the proliferation of large knowledge-sharing communities like Wikipedia have enabled the construction of large general-purpose knowledge bases with an entity-relationship or RDF-like data model. Projects along these lines include DBpedia [3], Freebase [1], TrueKnowledge [2], TextRunner [4], or YAGO [24]. These are rich sources of facts about people, locations, organizations, sports events, etc. For example, they would know the Alma Mater of scientists and awards that they have won, spouses and "romantic affairs" of entertainment stars, or the location and architect of culturally important buildings (churches, temples, museums, castles, etc.). However, the knowledge bases are still fairly sparse in terms of multimodal information about entities, like photos, videos, audio recordings, etc. Even Wikipedia does not know how Moshe Vardi (professor at Rice University) or the Five-Finger Tower in Darmstadt (a German city) look like.

On the other hand, photos and videos of people and landmarks have become abundant on the Internet. Web 2.0 portals such as Flickr and Youtube even offer extensive tags and metadata (e.g., GPS coordinates), but these are often noisy or incomplete, and sometimes wrong. Recently, various projects such as [21, 26, 30, 22, 31, 10, 8, 32] have started analytic mining of photo-tag or photo-GPS co-occurrences in order to improve the semantic organization of such data collections. However, with the exception of ImageNet [10] discussed below, none of them addresses the integration of photos into knowledge bases with formalized notions of entities, types, and facts.

Our goal in the work described in this paper is to populate an existing knowledge base with photos of people and landmarks. We use YAGO [25], which contains about 2 million typed entities, including all people, buildings, mountains, lakes, etc. from Wikipedia, and about 20 million relational facts like birthdates, awards, etc. In principle, it is not difficult to find photos of people or monuments using search engines like `images.google.com` or `images.bing.com` or searching `flickr.com` by tags. This works well for entertainment stars, important politicians, and tourist attractions. However, it remains difficult to find photos for entities in the "long tail": lesser known but still notable people and places. Typically, a direct query with the entity name returns many photos with good results in the top ranks but quickly degrading precision with decreasing ranks. For a human user who knows the entity of interest, it may be

good enough if the top 10 or top 20 contain a handful of correct photos, but this is insufficient for automatically enhancing a high-quality knowledge base. Moreover, even for more prominent targets, it is desirable to have a diverse collection of photos (e.g., from different time periods), some of which might be rare and difficult to locate using search engines. In some cases, the ambiguity of the entity name dilutes the search engine results. An example is the Berkeley professor and former ACM president David Patterson. None of the top-20 Google results (as of July 2009) show him; most show the governor of New York (whose name is actually David Paterson). The top-20 Bing results include several photos of our target entity, but it is difficult even for a human and extremely difficult for the computer to discriminate these from the photos of other people (with the same or very similar name).

None of the methods in the photo-tag-mining projects mentioned above can solve this problem. The closest project to our work is ImageNet [10], which enhances the WordNet thesaurus [11] with photos. In contrast to our goal, however, the task there is to find representative photos of semantic classes such as towers, churches, mosques, cats, tigers, etc. There is no consideration on photos of individual entities such as the Five-Finger tower in Darmstadt, the Blue Mosque in Istanbul, etc. Moreover, a small number of good photos is considered sufficient, whereas we aim at finding a large number of diverse photos for the same named entity.

1.2 Problem Statement

We consider named entities e of different types t, for example, scientists, politicians, buildings, mountains, etc. We assume that, for each type t, we have specific relations $R_i(subject\ e, object\ o)$ $(i = 1..m(t))$ populated in the knowledge base. These could be, for example, the affiliation, Alma Mater, and scientific field for scientists; the geographic areas (country, state, city) of activities and political positions held by politicians; the country and height of mountains and the person who climbed it first; and so on. We can use these facts to generate specific queries that we can send to image search engines or other services on the Internet. Finally, we assume that we have training data for each entity type t: examples of photos and their URLs that correctly show a given entity, for a small set of entities.

Our goal is to automatically gather photos for other, previously unseen, entities of known types. We focus on people and places who are notable but not extremely prominent, or have ambiguous names. We aim at both high precision and high recall, so that quality measures like MAP (mean average precision) or NDCG (normalized discounted cumulative gain) are maximized across a large set of results for the same entity. Note that the target entities are disjoint from the training entities. We only need to know the type-specific relations (and any information learned about them). In fact, we would typically train with popular entities, but apply our method to less known ones which may not even have a photo in Wikipedia.

1.3 Contribution

Our approach constructs a set of expanded queries for each entity of interest, where the expansions are automatically derived from already known facts in the knowledge base. For example, to find photos of the Berkeley professor David Patterson, we would use the hasAffiliation or worksInField relations of YAGO and search for "David Patterson Berkeley" or "David Patterson computer science". These expanded queries are then posed to image search engines. The collected results are ranked based on aggregating the results from different query expansions, with specific weights for different expansions. The weights are automatically learned from training samples. This approach can be seen as a form of (probabilistic) consistency checking of search engine results, as reflected in the overlap of the results for different expansions. In addition, we consider image-content similarities among different result candidates, using SIFT and MPEG-7 features.

The novel contribution by this paper has the following salient properties: 1) We show how to harness relational facts about named entities for gathering diverse photos of the entities with high precision and high recall. 2) We develop robust methods for estimating model parameters, so that our approach is applicable to a wide variety of different entity types. 3) We integrate image-similarity computations for improving the final ranking of result photos. 4) Our experimental studies demonstrate the practical viability of our approach.

The rest of the paper is organized as follows. Section 2 briefly reviews related work. Section 3 presents our scoring model and its training and ranking algorithms. Section 4 discusses how image similarity measures can be harnessed for an alternative ranking with improved diversity. Section 5 presents our prototype implementation. Section 6 demonstrates our experimental results.

2. RELATED WORK

Photo retrieval by visual similarity has been investigated very extensively; see [9] for a recent survey. Internet image search engines, on the other hand, index and retrieve photos primarily by keywords and other non-visual context that surround a photo on a Web page [16]. But they do provide options for smart processing like re-ranking obtained results by visual similarity or filtering them by photo types (e.g., person faces). Also, it seems that they employ models or heuristics for diversifying the top-ranked results, but details are not public.

Recently, a number of projects have focused on enhancing the *semantic organization of image data collections*: TinyImage [26], LabelMe[22], the work by Yao et. al. [31], and ImageNet [10]. TinyImage [26] is a dataset of low resolution images collected from the Internet by sending all nouns in WordNet[11] as queries to several image search engines. It uses the hypernymy relation of WordNet in conjunction with nearest-neighbor methods to automatically classify the retrieved images. LabelMe [22] is a large collection of images with ground truth labels to be used for object detection and recognition research. It aims at object class recognition (e.g., bridge) as opposed to instance recognition (e.g., Golden Gate Bridge), and learning about objects embedded in a scene (incl. bounding boxes and polygons). Similarly to LabelMe, Yao et. al. [31] have developed a labeling framework with rich representations for scene-level geometry, object segmentations and decompositions, and local geometric features.

ImageNet [10] is the closest project to our work. Unlike the projects sketched above, ImageNet addresses the problem of integrating photos into a knowledge base with formalized entities and types, namely, WordNet. It builds a large-

scale labeled image collection based on the taxonomic hierarchy of WordNet. To this end, ImageNet exploits the hypernymy relation between entity classes and nearest-neighbor-based classification with visual features. While ImageNet focuses on finding photos of semantic classes such as towers, churches, etc., our work in this paper addresses photos of *individual entities* such as the Five-Finger tower in Darmstadt, the Blue Mosque in Istanbul, etc. Moreover, ImageNet considers a small number of good photos as sufficient, whereas we aim at finding a large number of diverse photos for the same named entity.

Other projects [8, 21, 30, 32] pursue the dual aim of *mining text and visual information* to learn tagging-like properties of images. Crandall et. al. [8] present techniques to automatically identify places shown on photos, using correlations between photos with GPS metadata and tagged but GPS-less photos on flickr. Quack et. al. [21] propose an unsupervised-learning approach to structure, interpret, and annotate large photo collections. Yagnik et. al. [30] present a learning paradigm to address the problem of learning face models for people names. Similarly, Zhao et. al. [32] propose a system that can learn and recognize faces by combining signals from weakly labeled text in image and video corpora.

Some systems that focus on *photo management and retrieval* make use of ontologies [13, 20]. Gupta et. al. [13] propose Medialife, a system that captures semantic relations between the concepts of images. The goal is to support life chronicle queries such as *create an album of 20 photos of my son's graduation party ordered by time*. To achieve this task, Medialife builds a model upon a personal ontology to create and incrementally update an information association graph over a collection of photos and annotations. Popescu et. al. [20] propose RetrievOnto, an ontology-based IR system that supports both keyword-based and query-by-example search. The ontology is used to reformulate queries and to structure the resulting images. It is also used to perform content-based search in different subsets of the conceptual hierarchy.

All of the above projects exploit some form of semantic information about images to provide automatic annotation tools, and improve data retrieval and the organization of photo collections. However, none of them pursues the integration of photos of individual entities into knowledge bases with formal notions of typed entities and relational facts. Our unique goal in this paper is to automatically populate such a knowledge base with diverse sets of photos of different types of people and landmarks.

3. SCORING MODEL

3.1 Ensemble Voting Model

The easiest way of obtaining photos of a given entity (person or landmark) is by using the entity's name to issue a query to an image search engine. However, the results with this simple approach are often unsatisfactory. Even if good results appear on some of the top ranks, the entire ranking, say the top-100 results, is noisy and contains a significant number of incorrect photos or near-duplicates (although more and better photos exist at much lower ranks). We exploit the knowledge base to issue a variety of meaningful query expansions, each separately, and then analyze the results and rankings of different queries for agreement.

This can be seen as an *ensemble voting method* to arrive at a consistent ranking of the entire pool of retrieved photos. The ensemble consists of different queries $q_1(e)$, $q_2(e)$, ..., $q_m(e)$ about the entities of interest. Query $q_1(e)$ always is just the common name of the entity; all other queries are generated from specific relations that the knowledge base has for the given entity type $t(e)$. For example, we discriminate people into types like scientists, actors, pop musicians, politicians, etc. Interesting relations for generating queries are birthdate, affiliation, Alma Mater, scientific field or genre, awards, contributions (publications, movies, songs and albums, etc.), and so on. Different entity types should favor different relations even if they were applicable uniformly, for reasons explained below.

In principle, the queries q_2 through q_m would only yield subsets of the results that we obtain from the simple name query q_1. However, the results exhibit significant differences in their rankings. As search engines often return hundred thousands of results, we can practically access only top-ranked subsets of the query results, so that virtually no two queries show any subset-superset relationship. Therefore, photos returned by (the top-100 or top-1000 of) many queries for the same entity are more likely to be correct matches.

Each query expansion assigns high ranks to photos from Web pages where the query keywords appear prominently and close to the photos. Although this is an oversimplified view of how modern image search engines work, it reflects the essence of their ranking criteria. Thus, accepting a photo if and only if multiple queries *agree* on the photo being relevant can improve the precision of the overall result set. Each query *"votes"* for a photo, and receiving many votes indicates a better result.

Binary Voting. More formally, with each photo p in the union of the result sets (actually the top-k prefixes of the result lists that we retrieve) of queries $q_i(e)$ ($i = 1..m$) for entity e, we associate indicator variables $X_i(p)$ ($p = 1..|results|$) set to 1 if p occurs in the result of query $q_i(e)$ and 0 otherwise. Then the *voting score of a photo p with regard to entity e* is computed by the aggregation:

$$s(p,e) = \sum_{i=1..m} X_i(p)$$

On first glance, it seems that this method merely helps improving the precision of the overall results by simple ensemble voting. However, it can also improve recall and diversity of the results for a given entity. The reason is that we are not able to retrieve the complete result for a given $q_i(e)$ from any of the big search engines. Thus, running different queries whose results have very different ranks in different queries allows us to fetch a wider variety of photos at affordable cost.

Weighted Voting. Not all of the possible query expansions $q_i(e)$ have good yield. Some are overly specific and thus return too few results. An example would be adding the exact birthday of a person to the person's name; there are not that many biographies on the Web that have this information and at the same time contain a good photo. In contrast, query expansions with the birth year are often helpful in disambiguating an entity (e.g., Jim Gray or David Patterson whose names are fairly common). Other query expansions are too unspecific, lose focus, and are susceptible to topic drifting. For example, expanding a musician's name with names of songs or albums may return photos of the al-

bum cover. On the positive side, however, many expansions help in focusing the photo search. For example, searching for computer scientists who wrote popular text books simply by person names often returns book covers or figures from a book; in some cases search engines return photos of collaborators or former students. These problems may be overcome by query expansions that add the affiliation, an important award, or similarly salient facts about the person of interest.

The variability in the precision and recall of different query expansions is taken care off by giving different weights w_i to the various queries $q_i(e)$ in our voting scheme. It is straightforward to extend our approach into a *weighted voting score*:

$$s(p,e) = \sum_{i=1..m} w_i X_i(p)$$

The weights in this scheme could be the same across all entity types or specifically chosen for each type. The latter is more powerful and indeed advantageous for our scenarios. For example, while the birth year is a beneficial expansion for scientists, it is not nearly that helpful for musicians.

Parameter Estimation. The proper weights for a given entity type can be learned from explicitly labeled training data. We assume that we have at least a few correct photos of a few entities, for each type. These may be celebrities or famous landmarks where photos are ample (incl. photos in Wikipedia); the test cases for our calibrated model would then be less prominent entities. We estimate the query-specific weights w_i for a set T of training entities of type t, each with a ground-truth set of correct photos $P(e)$ and query results $Q_i(e)$ for query expansion $q_i(e)$, by:

$$w_i = \frac{1}{|T|} \sum_{e \in T} \frac{|Q_i(e) \cap P(e)|}{|P(e)|}$$

The weights w_i do not reflect the true fraction of correct photos retrieved by query q_i because we do not have ground-truth labels for all photos in the result set of q_i. The w_i values reflect the relative recall of the various query expansions.

Rank-based Voting. A final piece of information that we can exploit in the scoring function is the fact that Internet search engines return ranked lists rather than result sets. Photos at higher ranks are usually better matches, with a higher likelihood of really showing the entity of interest. This is a reasonable postulate regardless of our treating search engines as black boxes. It suggests moving from binary voting to *rank-based voting*, with the same query-specific weighting. Let $r_i(p)$ denote the rank of photo p in the result of query q_i. These are numbers 1, 2, etc., with low numbers denoting high ranks. The score of p should thus decrease with the value of $r_i(p)$, which leads to rankings based on the following scoring formula for result pools gathered by retrieving the top k results of each q_i:

$$s(p,e) = \sum_{i=1..m} w_i \frac{k+1-r_i(p)}{k}$$

3.2 Logistic Regression Model

Instead of the above direct estimation of model parameters, we could alternatively model our problem as a classification task for recognizing correct photos or as a regression problem for scoring the retrieved results, and then use Bayesian arguments for parameter learning. Consider the following binary random variables:

Y = a given photo is a true photo of target entity e

X_i = a given photo is retrieved by the query $q_i(e)$

and the integer-valued random variable:

R_i = a given photo is returned at rank R_i by query q_i. We can devise a Bayesian standard model that reasons about the probability $P[Y|X_1 \ldots X_m]$ or, analogously, a rank-based model using R_i instead of X_i variables. If we assume the maximum-entropy principle for unobserved data, this leads to a logistic-regression model of the following form [19]:

$$P[Y|X_1 \ldots X_m] = \frac{exp(\sum_{i=1..m} w_i X_i)}{1 + exp(\sum_{i=1..m} w_i X_i)}$$

where w_i are feature weights that are learned by maximizing the (regularized) log-likelihood of the training data using Quasi-Newton optimization methods. A new test photo is accepted by a logistic-regression classifier if its in-class probability exceeds the out-of-class probability. An analogous model can be learned with rank-based features R_i.

4. RANKING WITH VISUAL SIMILARITY

Query result lists for entities may contain many duplicate or near-duplicate photos. Since one of the goals in this work is to find rankings of diverse images, we need a way to capture similarity or identity of photos. Merely comparing result images by their URI's does sometimes not give satisfactory results. There are many identical photos for a given entity with different URI's. Moreoever, there are many near-duplicates that have, for example, different sizes, slightly different illuminations, or are simply cropped. As a remedy, we exploit visual similarities in order to remove near-duplicates and produce a better *diversity-aware ranking* of the images.

We can use visual similarities in two different steps of the scoring model described in Section 3: in the parameter estimation step, and in the final result ranking step.

Parameter estimation. We make use of photo similarities as follows. As in Section 3, assume that we estimate query-specific weights w_i for a set T of training entities of type t, each with a ground-truth set of correct photos $P(e)$ and query results $Q_i(e)$ for query expansion $q_i(e)$. The weights w_i can be alternatively estimated by checking how many of the images in $Q_i(e)$ are similar to the images of the ground-truth set $P(e)$. More formally:

$$w_i = \frac{1}{|T|} \sum_{e \in T} \frac{\sum_{p \in P(e)} \sum_{x \in Q_i(e)} sim(x,p)}{|P(e)|}$$

where $sim(x,p)$ is a binary variable set to 1 if x and p are similar images in the sense described below and 0 otherwise. This way we boost the weights for "good" relations, which find photos that are similar to those in the ground-truth set.

Final ranking. With these similarity-enhanced weights, we can compute the ranked results for a new entity as outlined in Section 3. But we can further enhance this ranking into a potentially better one by the following procedure. For each photo p in the union of result lists of queries $q_i(e)(i = 1..m)$ for entity e we compute its voting score by the aggregation:

$$s(p,e) = \sum_{i=1..m} w_i \left(\sum_{x \in Q_i(e)} sim(x,p) \frac{k+1-r_i(x)}{k} \right)$$

where k is the number of results of $q_i(e)$ and $r_i(x)$ is the rank of photo x in $q_i(e)$'s result list. This way we give high ranks to those images that have many near-duplicates in the result lists across all queries. In fact, this happens often for the entities that are notable but not famous. They have very few photos and the result lists of their queries have many photos with different URI's but very similar content.

Visual Features. We estimate visual similarities of photos using SIFT-based feature descriptors [18]. The SIFT feature descriptors are specific for each image and are based on particular points of interest in the image. The descriptors are known to be invariant under affine transformations and also robust to changes in the illumination. After estimating the feature descriptors for each image p in the union of result lists of queries $q_i(e)(i = 1..m)$ for each entity e, we need to check if two images are similar. We do this by performing the following steps. For every two images we find the nearest neighbor matchings between the two sets of feature descriptors. We use kd-Trees [6] and a Best-Bin-First algorithm [5] for finding approximate nearest neighbors. Having the feature correspondences, we apply RANdom SAmple Consensus (RANSAC) [12] to find the best affine transformation of the two images. With RANSAC we geometrically verify if the images are indeed near-duplicates and hence obtain higher precision in the results.

The computations described above are expensive. So to reduce their costs, we perform a filtering step beforehand, by using MPEG-7 global feature descriptors [23]. We use Edge-Histogram and Scalable-Color descriptors to identify those images that have high differences in these two descriptions, and thus save SIFT-based comparisons between clearly dissimilar photos.

5. IMPLEMENTATION

We have implemented the presented scoring models in a Java-based prototype system. The overall system architeture is illustrated in Figure 1. The system consists of five major components:

- The *query generator* obtains relational facts about entities from the knowledge base and generates keyword queries from them. Queries always contain the original entity name as well.
- The *photo search* component invokes queries on different photo search engines and retrieves the top-100 results for each query.
- The *parameter estimation* uses the results for the training entities to compute best suitable weights for the voting model or for the logistic-regression classifier. For the latter we use the ridge logistic regression provided by the WEKA toolkit [29].
- The *result ranking* applies the scoring model in order to rank the results for new entities. For the logistic-regression model, the regression function values are used for ranking.
- As an optional component, the visual similarity testing can be applied to two photos for near-duplicate detection, or to an entire set of photos. In the latter case, the photos are grouped into equivalence classes of near-duplicates (see Section 4). For visual similarity, we have used MPEG-7 and SIFT features. The extraction procedures for these features are implemented by the Lire [17] and IVT[14] software libraries.

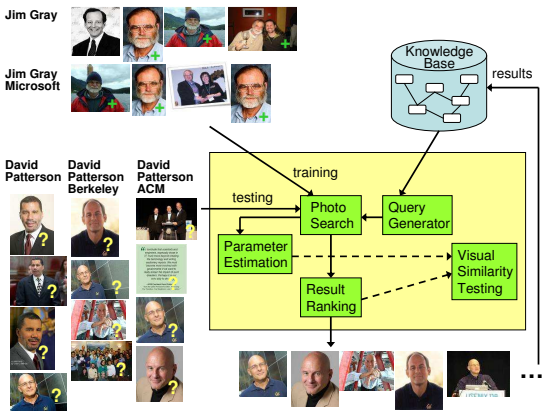

Figure 1: System architecture

At training time, the query generator invokes photo search to retrieve photos of the training entities, and then uses the results for parameter estimation. Optionally, the retrieved photos are grouped by the photo similarity testing, which results in different parameter values for the scoring model. At harvesting time, the query generator and photo search work the same way, but for new, previously unseen, entities. The retrieved photos are fed into the result ranking, which uses the parameters previously learned from the training entities. The ranking can optionally use photo similarity testing for grouping near-duplicates and showing only per-group representatives to the user. This aims to enhance the diversity of the final results. Finally, the best photos gathered this way are added to the knowledge base, along with information about their provenance and confidence (based on our scoring model).

6. EXPERIMENTS

6.1 Setup

We evaluated the scoring models for *normal ranking* (Section 3) and *diversity-aware ranking with visual similarity* (Section 4), using binary relevance assessment on a set of result rankings. We used four classes of entities: scientist, politician, religious building, and mountain. Each class contains 15 training entities and 10 test entities (disjoint from the training set). Each of the training entities has between 10 and 100 hand-selected photos, depending on whether the entity is highly notable or not so notable. To generate the queries for each entity we use relational facts specific for each class. Table 1 lists a few test entities and a subset of their relational facts.

For each test entity we posed the generated queries to Google and Bing for the people classes and to Google and Flickr for the landmark classes. We collected the top 100 from each result list, and applied our scoring models. We showed the entire pool of results to human judges for relevance assessment. The judges considered a photo as relevant if they could clearly recognize the target entity, possibly after reading the Web page where the image was found. For

		Google			Bing/Flickr		
		Original	Voting	Regression	Original	Voting	Regression
scientist	MAP	0.6200	0.7181	0.6004	0.6144	0.6837	0.5450
	NDCG	0.8181	0.9141	0.8211	0.8285	0.8909	0.8182
	BPREF	0.6002	0.7553	0.7039	0.6642	0.7683	0.6572
politician	MAP	0.7885	0.8011	0.7190	0.7302	0.7586	0.6739
	NDCG	0.9376	0.9453	0.9064	0.9138	0.9291	0.8786
	BPREF	0.7373	0.7804	0.7062	0.7469	0.8135	0.7489
building	MAP	0.7284	0.7721	0.6958	0.7800	0.8389	0.8000
	NDCG	0.8669	0.9069	0.8519	0.8750	0.9073	0.8909
	BPREF	0.6274	0.7455	0.6498	0.6597	0.7405	0.6624
mountain	MAP	0.8053	0.8303	0.8204	0.8571	0.8466	0.8053
	NDCG	0.9308	0.9562	0.9480	0.9637	0.9608	0.9485
	BPREF	0.6696	0.7101	0.6885	0.7266	0.7120	0.6859

Table 2: Evaluation measures for normal result rankings.

Entity	Relational Facts
class scientist	
Alfred Louis	field: Mathematics institution: Saarland University
David Patterson	known for: RISC, RAID institution: University of CA, Berkeley awards: ACM IEEE Eckert-Mauchly Award
Niklaus Wirth	Alma Mater: ETH Zürich awards: Turing Award known for: Pascal, Algol W, etc.
class politician	
Jon Huntsman	political party: Republican position: Governor of Utah, etc.
Ignatz Bubis	birthplace: Breslau; death year: 1999 profession: Jewish leader
Niels Annen	political party: SPD position: Jusos, etc.
class religious building	
Wat Arun	location: Bangkok known for: Buddhist temple names: Temple of the Dawn, etc.
Einsiedeln Abbey	known for: Benedictine monastery location: Switzerland, etc.
Boyana Church	location: Sofia known for: Boyana Master, etc.
class mountain	
Siula Grande	location: Peru; height: 6344 range: Cordillera Huayhuash
Mount Ararat	names: Mountain of Pain location: Dogubayazit location: Agri Province, Turkey
Dreieckhorn	range: Bernese Alps; height: 3811 location: Switzerland

Table 1: Examples for entities and relational facts.

the people classes, not only personal photos were accepted, but also when the person could be recognized in a group with others. For the landmark classes the judges accepted images that show the place including unusual perspectives, but disgarded those images that did not show anything specific for the place and could have been taken in many other places (e.g., a close-up of a snow patch on a mountain).

For each entity type and search engine we compare three methods: 1) the *original* search engine rankings, 2) our *voting* results using rank-based weighted voting, and 3) the rankings from the standard *regression* model with binary features. We present results for two different kinds of rankings: a) *normal rankings* and b) *diversity-aware rankings with visual similarity*. For the former we considered photos to be duplicates only by URI comparison. For the latter we used near-duplicate grouping by visual similarity; in this case only one representative of each cluster was shown to the human judges.

To compare the results of the different methods, we use three quality measures: Mean Average Precision (MAP), Normalized Discounted Cumulative Gain (NDCG), and a preference-based measure (*bpref*). The MAP measure is the mean of the precision scores obtained at the ranks of each relevant image, which is an interpolated approximation of the area under the precision-recall curve. It is computed as follows:

$$MAP(E) = \frac{1}{|E|}\sum_{j=1}^{|E|}\frac{1}{R_j}\sum_{k=1}^{R_j} precision@(rank_{jk})$$

where E is the set of test entities e_j, R_j is the number of relevant photos $\{p_1, \ldots, p_{R_j}\}$ for e_j, and $rank_{jk}$ is the rank of the kth relevant photo. Additionally, we compute NDCG to measure the usefulness (gain) of images based on their (geometrically weighted) positions in the result list. It is computed as follows:

$$NDCG(E,k) = \frac{1}{|E|}\sum_{j=1}^{|E|} Z_{kj}\sum_{i=1}^{k}\frac{2^{rel(j,i)}-1}{log_2(1+i)}$$

where Z_{kj} is a normalization factor calculated to make NDCG at k equal to 1 in case of perfect ranking, and $rel(j,i)$ is the relevance score of an image at rank i for entity e_j. In our setting, relevance scores $rel(j,i)$ were binary.

Recall that our use of search engine queries can practically retrieve only a small subset of the full result sets, the top 100 in our setup. By inspecting only the top k results for each query, it is impossible to know whether a relevant image has not been found at all or simply because the rank of the image is higher than k. And some sophisticated queries may return less than k results. This situation is rectified as follows (using TREC-style practice). Consider m methods (runs) under comparison. Each method returns a ranked list, truncated at rank k. Suppose we have a total of N distinct results from all the result lists ($N \leq k \times m$)). From the N results, the human assessors give us a set of R relevant images. The next step is to pad each result list with the

		Google			Bing/Flickr		
		Original	Voting	Regression	Original	Voting	Regression
scientist	MAP	0.5605	0.6318	0.5428	0.5187	0.6015	0.5438
	NDCG	0.7943	0.8778	0.7989	0.7751	0.8619	0.8287
	BPREF	0.6327	0.8086	0.8061	0.7118	0.7971	0.7630
politician	MAP	0.7271	0.7679	0.6615	0.6564	0.7210	0.6258
	NDCG	0.9151	0.9358	0.8703	0.8858	0.9174	0.8603
	BPREF	0.7477	0.8460	0.7933	0.7083	0.8259	0.7754
building	MAP	0.6650	0.7259	0.6715	0.7292	0.8222	0.7782
	NDCG	0.8453	0.8782	0.8394	0.8603	0.9041	0.8850
	BPREF	0.5726	0.8085	0.7449	0.6313	0.7892	0.7322
mountain	MAP	0.7637	0.8219	0.8287	0.8235	0.8284	0.8051
	NDCG	0.9208	0.9540	0.9567	0.9540	0.9567	0.9494
	BPREF	0.6054	0.7573	0.7691	0.6691	0.7400	0.7102

Table 3: Evaluation measures for diversity-aware result rankings.

	entity name	birth year	field	institutions	political party	positions
scientist	0.594/1.3	0.328/0.829	0.411/1.066	0.314/ 0.814	n/a	n/a
politician	0.579/1.254	0.314/0.731	n/a	n/a	0.461/0.878	0.367/0.66

Table 4: Normal weights / similarity weights for the scientist and politician classes using Google.

	entity name	location	height	range	known for
building	0.598/1.448	0.514/1.222	n/a	n/a	0.351/0.863
mountain	0.573/1.079	0.354/0.703	0.256/0.619	0.294/0.616	0.257/0.622

Table 5: Normal weights / similarity weights for the religious building and mountain classes using Google.

missing relevant images. For each method m_j that has R_j ($R_j < R$) relevant results and k results overall, we add the remaining $R - R_j$ relevant results on (virtual) ranks $k+1$, $k+2$, etc. If method m_j has only $k' < k$ results overall, then we consider ranks $k'+1$, $k'+2$,$\cdots$,k as non-relevant and add the remaining $R - R_j$ relevant results at ranks $k+1$, $k+2$, etc. This way all methods are evaluated as if they had 100% recall, based on the pooled results of all methods, and we can compute the standard MAP measure.

Note that because 1) the true recall can be much larger than our pooled result set and 2) each method in our setup typically returns a very small subset of the full recall (top-100 out of potentially many thousands of photos), the padded result lists tend to have similar MAP values when $R >> k$. For this reason, we also computed the *bpref* measure which is highly correlated to MAP when complete information is provided and more robust otherwise. For a bounded ranked list with top k results and a total of R relevant results, *bpref(k)* is defined as follows:

$$bpref(k) = \frac{1}{R} \sum_r 1 - \frac{|\#n \text{ ranked higher than } r|}{k+R}$$

where the summation ranges over the ranks r of relevant result and $\#n$ counts non-relevant results. *bpref* does not depend on potential results (from the pool of all methods' results) on ranks $> k$. Thus, it does not degrade as much as MAP when $R >> k$.

6.2 Results

Normal Ranking. The results for normal ranking are shown in Table 2. For all baselines Google, Bing, and Flickr, our voting method almost always improves all three measures MAP, NDCG, and *bpref*. (The one exception is the Flickr ranking for mountains; see discussion below.) We observe that the gains vary depending on the entity type. For example, for the scientist class, when using Google, the MAP value increases from 0.62 to 0.7181. In contrast, for the politician class the absolute improvement is less than 2%. We note that bpref shows higher gains for reasons discussed above. Similar observations hold for Bing and Flickr. The results also show that the logistic regression model does not perform well in the grand total. Our unusual notion of "features" derived from noisy query results seems to be difficult to handle by standard machine learning. However, for a few individual entities, the regression model actually performed best.

Tables 4 and 5 show the weights for (a subset of) different types of relational facts that our voting method uses, based on its parameter estimation from the training entities. Note that the weights are not normalized (and do not need to be). Not surprisingly, the original name tends to have the highest weight, and there are big differences in the usefulness of the other relations. The most useful relations were: the field for scientists, the party for politicians, and the location for the two landmark classes.

Diversity-Aware Ranking with Visual Similarity. We have also applied the extended scoring model with visual similarity to the three methods Original, Voting, and Regression. In this case, near-duplicates are clustered and only one representative of each cluster is included in the final result list. Table 3 shows the different measures for these diversity-aware rankings. Similarly to the results with normal ranking, our voting method consistently improves all three measures (now without any exceptions). On average,

		Google			Bing/Flickr		
		Original	Voting	Regression	Original	Voting	Regresssion
scientist	Alfred Louis	0.0199	0.8095	0.0783	0.0849	0.7672	0.4546
	David Patterson	0.1300	0.5690	0.4326	0.1566	0.6846	0.7712
	Emmy Noether	0.8382	0.8917	0.8472	0.9079	0.9561	0.9573
politician	Ignatz Bubis	0.5390	0.7363	0.6378	0.6948	0.7489	0.6862
	Jon Huntsman	0.8510	0.9516	0.9534	0.8187	0.8451	0.8517
	Renate Blank	0.5069	0.5902	0.4744	0.5240	0.6646	0.6200
building	Church of Christ Pantocrator	0.3012	0.6776	0.4128	0.3618	0.7789	0.5660
	San Lorenzo	0.0289	0.1422	0.0689	0.0195	0.0303	0.0195
mountain	Pilatus	0.4061	0.5327	0.5292	0.8794	0.8860	0.9099
	Mönch	0.3092	0.6685	0.7557	0.9861	0.9958	0.8719

Table 6: Examples for MAP values of normal rankings for individual entities.

		Google			Bing/Flickr		
		Original	Voting	Regression	Original	Voting	Regresssion
scientist	David Patterson	0.1054	0.4164	0.2626	0.1439	0.5512	0.5718
	Emmy Noether	0.5928	0.6981	0.6471	0.7482	0.8416	0.8891
	William Vickrey	0.4986	0.6911	0.5848	0.5579	0.6726	0.5782
politician	Ignatz Bubis	0.5496	0.6956	0.5626	0.5903	0.6974	0.6008
	Stephen Crabb	0.5615	0.6178	0.4697	0.5488	0.6792	0.4421
	Luisa Diogo	0.7657	0.8228	0.7863	0.7203	0.7669	0.6812
building	Church of Christ Pantocrator	0.2335	0.4418	0.2647	0.3346	0.7685	0.5916
	Boyana Church	0.7546	0.7817	0.7075	0.7379	0.8444	0.7997
mountain	Tre Cime di Lavaredo	0.9542	0.9738	0.9884	0.8330	0.8685	0.8165
	Aiguille d'Argentière	0.7871	0.7881	0.6735	0.8758	0.8953	0.8770

Table 7: Examples for MAP values of diversity-aware rankings for individual entities.

the gains over the baseline competitors were even higher here than in the normal ranking comparison. The *bpref* measure shows the largest improvements. For example, for scientists using Google, our method improved *bpref* from 63% to 80% and achieved a similar gain for politicians. For buildings, we gained even more: from 57% to 80%; and even for the difficult mountain class, the bpref improvement is substantial (from 60% to 75%).

But there are again major differences in the magnitude of the improvement, depending on the entity type. Note that the absolute values of MAP, NDCG, and *bpref* are slightly lower than for normal rankings, because duplicates and near-duplicates of good results are now discounted. Also, the relative weights of different types of relational facts are adjusted (see Tables 4 and 5) because visual similarity is considered for the photos of the training entities as well. For example, the *knownFor* relation is additionally boosted with visual similarity, relative to other relations such as *location*. In fact, our experiments show that this leads to better results.

6.3 Discussion

Our experimental results show our voting method almost always outperforms the native rankings of image search engines, by a significant margin. Sometimes, however, the gains are small and generally depend on the entity type or even on the individual instance. In the following, we discuss some of the specific strengths of our method by means of anecdotic examples. We also point out limitations of our approach.

Specific strengths. We are performing particularly well for entities with ambiguous names or when an entity is very rare in the Internet photo space. Examples are shown in Tables 6 for normal ranking and 7 for diversity-aware ranking. Figure 2 shows top-ranked result photos, with visual-similarity clustering, for our method vs. those ranked high by image search engines. Each block shows the top-5 clusters (from top to bottom). Only up to 3 photos per cluster are shown; some clusters contained many photos, others were small.

In the scientist class, the search engines confused David Patterson with the New York governor Paterson. This is shown in the upper right part of Figure 2. Our voting method's result is not perfect either, but at least has 4 correct (groups of) photos in the top 5. William Vickrey, in the upper left part of Figure 2, turned out to be a difficult case because many of his photos are on content-rich Web pages with lists of Nobel Prize winners in Economy and many photos. Here, our top-5 results are perfect, whereas the search engine got only 3 out of 5 results right. Other difficult cases were Emmy Noether, as search engines also returned winners of an Emmy Noether Fellowship (by the German Science Foundation, named after her), Alfred Louis, as his last name is also a common first name. In the politicians class, we performed particularly well on lesser known people such as Ignatz Bubis or Renate Blank. Their names do occur often in news about parliamentary debates and other events of this kind, but these news contain photos of other people related to the same event.

We observed similar effects for the two landmark classes. For example, the mountain Pilatus, shown in the lower left part of Figure 2, turned out to be ambiguous because there is also an aircraft model called Pilatus. For landmarks, some

individual entities were challenging due to the fact that they are often mentioned on tourist sites that have many photos but not for every attraction that they talk about. For example, in the Google results for the Church of Christ Pantocrator (in Nessebar, Bulgaria), shown in the lower right part of Figure 2, 3 out of the top 5 results are wrong. They show an icon and a relief from other churches and a similar but different church, all of which are mentioned together on popular tourist sites about Balkan culture. In contrast, our voting model avoided two of these bad results and thus achieved 80% precision in the top 5 results. In general, for entities of this difficult nature, we achieved major gains over the baseline competitors.

Limitations. Although we aimed at entities in the "long tail" of notable but not famous people and places, the need for manually assessing the correctness/relevance of results entailed that our test entities were actually a mix of still fairly popular entities and some lesser known ones. For the popular entities, it was virtually impossible to beat the top-100 results of the two image search engines (unless the entity name was highly ambiguous). When search engines can choose from result sets with hundred thousands of photos, their ranking criteria obviously work extremely well. Thus, for famous people such as Frank Wilczek or Nelson Mandela we could not gain anything over Google and Bing, and occasionally even lost slightly in precision.

Likewise, for popular places, Flickr seems like a gold standard, given its rich tagging assets, and Google also performed extremely well. For example, the results for Wat Arun or Mount St. Helens could simply not be beaten. We realized, however, that Flickr tags are sometimes noisy; for example, an entire photo series on a Himalaya trip was uniformly tagged with "Himalaya", "India", "Tibet", "Everest", "Kailash", etc., although it is geographically impossible to have both Mount Everest and Mount Kailash displayed in the same photo. Unfortunately, these wrong tags also misled our method. In this regard, it would be interesting to use voting across results of different search engines. The combination of results from Flickr *and* Google, for different query expansions, may have the potential for overcoming this issue with noisy tags.

7. CONCLUSION

Retrieval and ranking of photos has received great attention in the prior literature. In this paper, we viewed this problem from the new angle of populating a knowledge base about people and places with a large set of diverse photos. In contrast to previous work on photos of celebrities, we aimed at a general approach for different entity types and paid particular attention to entities in the long tail of popularity.

For pragmatic reasons, our experiments were limited to retrieving only the top 100 results for each query expansion. Exploring many thousands of per-query results may be worthwhile in order to find rare photos, and could also add to the diversity-aware rankings with visual similarity. On the other hand, it does pose efficiency and scalability challenges. For exotic entities - local politicians, mountains off the beaten path, or cultural landmarks of regional interest -, the relational facts that we build on also tend to include some equally rare details. To overcome this issue, it may be interesting to generalize our model to allow instance-specific weights instead of weights on a per-entity-type basis. We are currently investigating these potential enhancements of our approach.

8. ACKNOWLEDGMENTS

This work was supported by the EU Project Living Knowledge.

9. REFERENCES

[1] http://www.freebase.com/
[2] http://www.trueknowledge.com/
[3] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, Z. G. Ives. Dbpedia: A nucleus for a web of open data. *ISWC/ASWC* 2007
[4] M. Banko, M. J. Cafarella, S. Soderland, M. Broadhead, O. Etzioni. Open information extraction from the web. *IJCAI* 2007
[5] J. S. Beis, D. G. Lowe. Shape indexing using approximate nearest-neighbour search in high-dimensional spaces. *CVPR* 1997
[6] J. Bentley. Multidimensional binary search trees used for associative searching. *CACM*, 18(9), 1975
[7] H. Chen, Z. J. Xu, Z. Q. Liu, S. C. Zhu. Composite templates for cloth modeling and sketching. *CVPR* 2006
[8] D. Crandall, L. Backstrom, D. Huttenlocher, J. Kleinberg. Mapping the world's photos. *WWW* 2009
[9] R. Datta, D. Joshi, J. Li, J. Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Comput. Surv.*, 2008.
[10] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. *CVPR* 2009
[11] C. Fellbaum. *WordNet: An Electronical Lexical Database*. MIT Press, 1998
[12] M. A. Fischler, R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *CACM*, 24(6), 1981
[13] A. Gupta, S. Rafatirad, M. Gao, R. Jain. Medialife: from images to a life chronicle. *SIGMOD* 2009
[14] IVT: Integrating Vision Toolkit. http://ivt.sourceforge.net/.
[15] L. S. Kennedy, M. Naaman. Generating diverse and representative image search results for landmarks. *WWW* 2008
[16] Google's Peter Linsley Interviewed by Eric Enge. http://www.stonetemple.com/articles/interview-peter-linsley.shtml, 2009.
[17] Lire (Lucene Image REtrieval library). http://www.semanticmetadata.net/lire/.
[18] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2), 2004
[19] T. M. Mitchell. *Machine Learning*. McGraw Hill, 1997 http://www.cs.cmu.edu/∼tom/mlbook.html, 2006.
[20] A. Popescu, C. Millet, P.-A. Moëllic. Ontology driven content based image retrieval. *CIVR* 2007
[21] T. Quack, B. Leibe, L. J. V. Gool. World-scale mining of objects and events from community photo collections. *CIVR* 2008
[22] B. C. Russell, A. Torralba, K. P. Murphy, W. T.

Figure 2: Example results with visual similarity grouping

Freeman. Labelme: A database and web-based tool for image annotation. *IJCV*, 77(1-3), 2008

[23] P. Salembier, J. R. Smith. Mpeg-7 multimedia description schemes. *IEEE Trans. on Circuits and Systems for Video Technology*, 2001

[24] F. M. Suchanek, G. Kasneci, G. Weikum. Yago: a core of semantic knowledge. *WWW* 2007

[25] F. M. Suchanek, G. Kasneci, G. Weikum. Yago: A large ontology from wikipedia and wordnet. *J. Web Sem.*, 6(3), 2008

[26] A. Torralba, R. Fergus, W. T. Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. *IEEE TPAMI* 30(11), 2008

[27] R. H. van Leuken, L. Garcia, X. Olivares, R. van Zwol. Visual diversification of image search results. *WWW* 2009

[28] C. Wang, L. Zhang, H.-J. Zhang. Learning to reduce the semantic gap in web image retrieval and annotation. *SIGIR* 2008

[29] I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann, 1999.

[30] J. Yagnik, A. Islam. Learning people annotation from the web via consistency learning. *MIR* 2007

[31] B. Yao, X. Yang, S.-C. Zhu. Introduction to a large-scale general purpose ground truth database: Methodology, annotation tool and benchmarks. *Energy Minimization Methods in Computer Vision and Pattern Recognition*, 2007

[32] M. Zhao, J. Yagnik, H. Adam, D. Bau. Large scale learning and recognition of faces in web videos. *FG* 2008