Faceted Exploration of Image Search Results

Roelof van Zwol
Yahoo! Research
Barcelona, Spain
roelof@yahoo-inc.com

Börkur Sigurbjörnsson
Yahoo! Research
Barcelona, Spain
borkur@yahoo-inc.com

ABSTRACT

This paper describes MediaFaces, a system that enables faceted exploration of media collections. The system processes semi-structured information sources to extract objects and facets, e.g. the relationships between two objects. Next, we rank the facets based on a statistical analysis of image search query logs, and the tagging behaviour of users annotating photos in Flickr. For a given object of interest, we can then retrieve the top-k most relevant facets and present them to the user. The system is currently deployed in production by Yahoo!'s image search engine[1]. We present the system architecture, its main components, and the application of the system as part of the image search experience.

Categories and Subject Descriptors

H.3 [**Information Systems**]: Information Search and Retrieval; H.4 [**Information Systems**]: Information Systems Applications; H.5 [**Information Systems**]: Information Interfaces and Presentation

General Terms

Design, Human Factors, Management

Keywords

Faceted browsing, image search, object extraction, ranking

1. INTRODUCTION

Search forms an integral part of the Internet, allowing people to navigate on the Web, find information on a particular topic, or as a starting point for entertainment. The keyword-based query formulation mechanism offered by Web search engines allows users to quickly spin-off in a particular direction of interest. It is repeatedly reported in literature that the query length on average is very short, with 90% of the query volume being accounted for by queries with a length $l(q) \leq 4$ [3, 13]. As a consequence, most queries entered by users leave room for some sort of ambiguity [19] and will lead to a diverse set of search results [1, 18].

In this paper we present MediaFaces, the back-end system that enables the faceted exploration of images in Yahoo!'s image search engine. Moving towards a "Web-of-objects" experience [6], MediaFaces provides a service that enables faceted exploration of large (media) collections. The objective of the system is to aggregate the knowledge and high-quality content that is available on and off the Yahoo! network, and to support the user in their quest for information by identifying the most relevant aspects of a query. The user, when presented with the facets, is likely to discover new facets of the query that they were not aware of before. When clicking on a facet, they will narrow down their search by expanding the original query with the suggested facet.

Faceted exploration of search results is widely used in search interfaces for structured databases such as shopping catalogues[2], job listing[3], house search[4], etc. Faceted browsing for semi structured sources has been proposed in research literature for some years [20]. But only recently they are also appearing in on-line search engines such as Yahoo![5], Google[6], and Bing[7] in the form of search assistants. These search assistants mostly derive their query suggestions from query log analysis [8]. MediaFaces also uses query log analysis, but only to rank facets from a large pool of candidates, where the facet candidates have been extracted from the trusted sources. Filtering for known facets allows us to provide conceptual query refinements of high quality.

The application of the system is best demonstrated with an example. When a user types in a search for a known entity, such as *London UK*, the system shows the user diverse images of the location but also gives the user the option to explore different facets of the entity. Figure 1 shows the search engine result page when the user queries for *London UK*. On the right-hand side the traditional image search results are shown. It can be clearly seen that the image search results shown are diverse, due to the broad nature of the query. On the left-hand side there is the list of facets for London UK. In this case, a list of prominent landmarks in London is displayed. If the user clicks on one of the facets a new set of photos is shown, displaying only photos that correspond to the clicked facet. Figure 2 shows the search engine result page when the user has clicked on the *London Eye* facet. The left-hand side facet list remains the same but the image results are now focused on the London Eye.

This paper is further organized as follows. We discuss

---

[1] http://images.search.yahoo.com/

[2] http://shopping.yahoo.com/
[3] http://hotjobs.yahoo.com/
[4] http://www.realtor.com/
[5] http://search.yahoo.com/
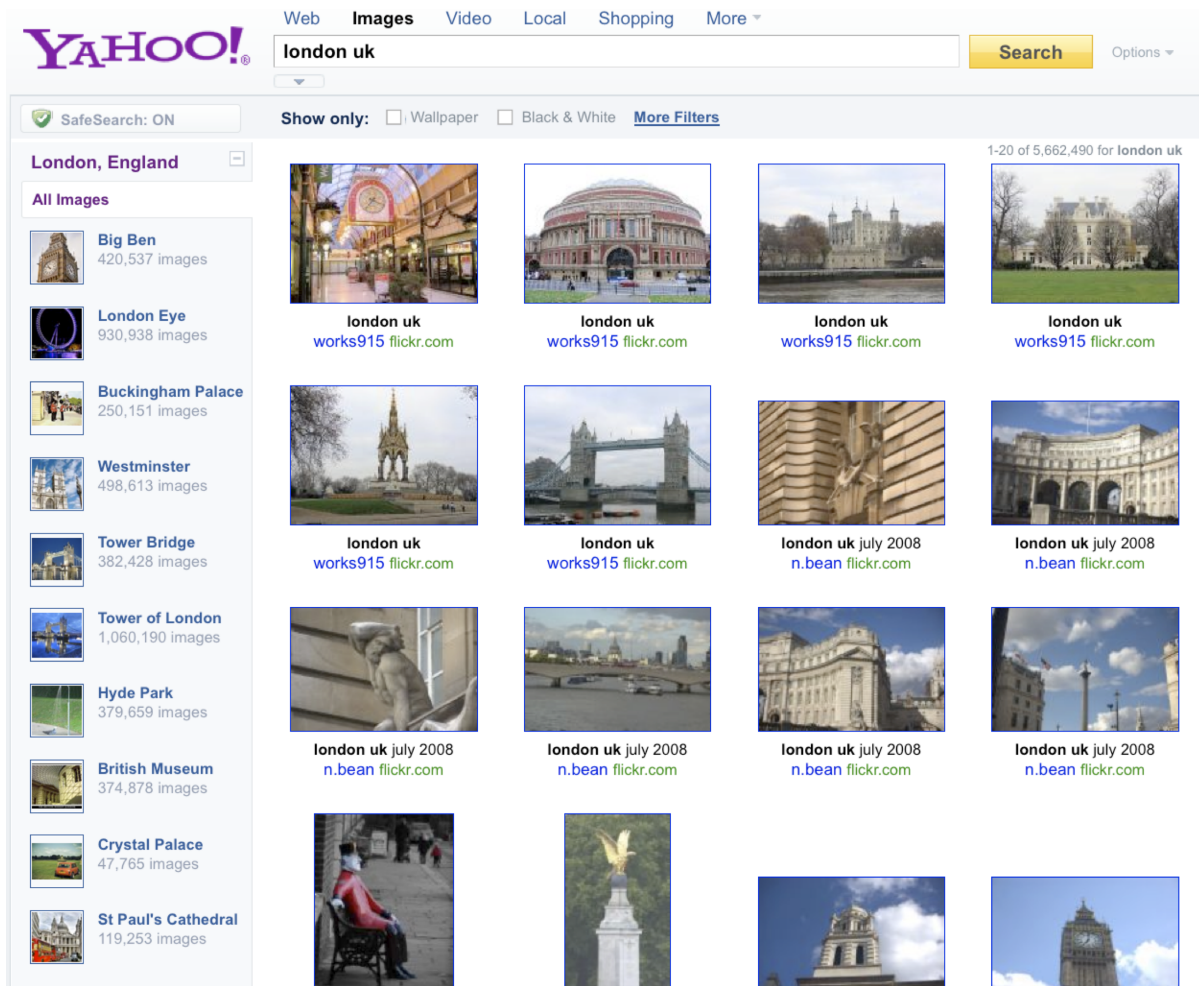[6] http://google.com/
[7] http://bing.com/

Figure 1: Screen capture of the search result page when the user queries for London UK.

related work in Section 2. The system architecture is described in Section 3 and Sections 4 to 7 describe the main components of the system: the facet repository, facet extraction, facet ranking, and the facet server. Finally, we present our conclusions in Section 8.

2. RELATED WORK

In this section we will discuss the related work on search assistants, faceted browsing, and media exploration and clustering.

2.1 Search Assistants

Interactive query refinement – or search assistance – has been studied extensively in the literature. The refinement terms can be derived either based on the underlying corpus [2, 15], from the set of retrieved documents [2, 14], or from query logs [8, 15].

Anick and Tipirneni suggest a method for generating query refinement terms using lexical dispersion of a word – the number of different lexical compounds in which the word occurs [2]. The algorithm takes a list of documents as input and returns the most disperse words as facets and the most frequent lexical compounds as value of the facets. Joho et al. investigate interactive query refinement [14]. They compare

a hierarchical query term presentation to a linear presentation. They show that there is little difference in retrieval performance but the hierarchical presentation results in significantly less effort and is superior in terms of user satisfaction. The presentation of the MediaFaces facets in Yahoo! image search uses two layers, as the facets are grouped by category. This is explained in more detail in Section 7.

Kraft and Zien present a query refinement method using anchor text and median rank aggregation [15]. They evaluate it in a user study and compare performance to refinement methods using document content and queries. Their evaluation shows that using anchor texts or queries gives similar performance values. Using document text is however inferior. Fonseca et al. present a query refinement strategy based on mining association rules from query logs [8]. The refinement terms are clustered into concepts where similar terms appear together. In an interactive experiment the user is asked to choose the concept that best describes the information need. When the concept is used for query expansion significant improvement in retrieval performance is shown. The MediaFaces system uses query logs as one of the main sources for ranking facets. We mine the image search query logs to rank the candidate facets which have been extracted from the trusted sources.
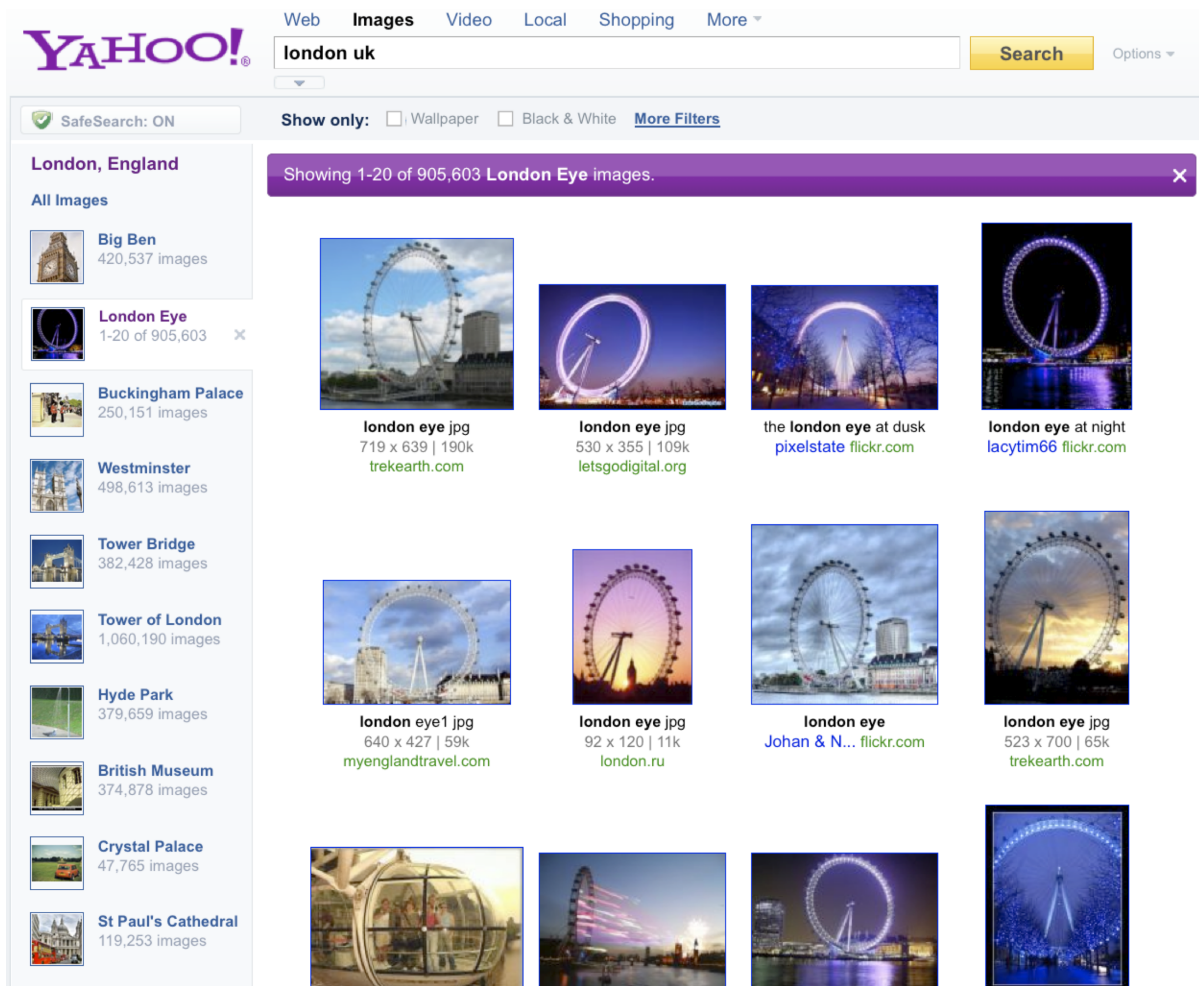
Figure 2: Screen capture of the search result page when the user queries for London UK and has clicked on the London Eye facet.

2.2 Faceted Browsing

Faceted browsing has been applied for numerous settings, using different types of facets. Burke et al. present faceted browsing for apartment rentals [4]. The user can refine queries based on facets such as price, size, and neighbourhood characteristics. The facets are extracted by applying a parser specialized for parsing classified ads. Yee et al. present an interface for searching and browsing images using faceted meta-data [20]. They apply the system for searching an art catalogue where the meta-data describes different facets such as artist names, types of media, dates and textual description of the art item content. The faceted meta-data is partially provided by the collection itself and partially extracted using WordNet. Based on our experience with MediaFaces, we can literally have thousands of such facets, while only a few can be presented to the user. The challenge therefore is to retrieve the most relevant facets from a large set of candidate facets. Zhang and Marchionini present a faceted browsing interface and apply it to a film database [21]. The faceted meta-data consists of film names, genres, actors, etc. Stewart et al. present a faceted browsing interface to search for subject-verb-object patterns in news corpora [16]. Their facets are based on parsing subject-

verb-object triples from natural language texts. The wide availability of social media, and the structured information provided by users participating in the social media offers an alternative to natural language processing, which we gratefully adopted for MediaFaces.

User evaluation has shown that faceted browsing of search results is particularly useful for exploratory search tasks [20, 21, 16]. The usefulness and implementation does however depend on the underlying data and is not a silver bullet that is appropriate for all types of data or all tasks [10, 5]. The best application of facets may depend on a balance between carefully hand crafted facet meta-data and fully automatic facet extraction. In the MediaFaces system we strike this balance by combining data extracted from high quality data sources with statistical analysis from query logs and photo annotations.

2.3 Media Exploration and Clustering

The research on (visual) exploration and clustering of image collections is extensive. A recent survey by [7] lists a number of the most significant contributions to the field of image retrieval, and stresses that image retrieval is more than a keyword based search often interactive of nature.

More focused on visually exploring image collections is the survey by Heesch [11]. Heesch also states that content based image retrieval (CBIR) has traditionally been investigated within a framework that emphasises the explicit formulation of a query: users initiate an automated search for relevant images by submitting an image or draw a sketch that exemplifies their information need. Its limitation is clear: There is often little support for exploratory search and scaling to very large collections is problematic. Moreover, the assumption that users are always able to formulate an appropriate query is questionable. An effective, albeit much less studied, method of accessing image collections based on visual content is that of browsing.

One such example where visual exploration is at the core of the approach is described in Heesch et al. [12]. The approach is based on an automatically constructed network of images that can be navigated quickly by following its edges. The browsing experience is enhanced in a number of ways including multi-dimensional scaling of the graph neighbourhood for display purposes, Markov clustering of the image network to provide summaries of its content, and automated annotation of the images to allow users to access the network through text queries.

Alternatively, Hare [9] introduces a faceted model of image semantics which attempts to express the richness of semantic content interpretable within an image. The interesting aspect of this paper is that it proposes to merge the topical exploration with a visual counterpart to analyse the image content. In context of the MediaFaces system this is of particular interest when it comes to ensuring that the images retrieved are relevant to the faceted query. Currently this is outside the scope of the MediaFaces system. However the notion of visual facets once the topical ambiguity is resolved is certainly high on the agenda [18, 17].

3. SYSTEM ARCHITECTURE

In this section we discuss the system architecture of the MediaFaces system, which is presented in Figure 3. The system has 5 main components which are described below. The system interacts with a number of structured sources on and off the Yahoo! network, such as GeoPlanet[TM], Yahoo! Movies, Wikipedia, etc. It uses these sources to extract structured information, e.g., known objects and candidate facets. On the other hand it uses query log analysis and the annotations provided by Flickr users to rank the facets.

Facet Repository. The facet repository is responsible for managing the extracted objects and facets - the relationship between a pair of objects. Section 4 describes the facet repository in detail.

Facet Extractor. The facet extractor module extracts objects and facets from the trusted sources. Facet extraction is performed whenever a new data source becomes available or an existing data source is updated. For example whenever a fresh dump becomes available, or when new items become available through an RSS feed. The facet extractor processes the input data, extracts objects and passes them onto the facet builder that is responsible for storing the extracted data in the facet repository. Facet extraction is described in Section 5.

Facet Ranking. The facet ranking module ranks the candidate facets for a given object. Facet ranking is performed periodically. All existing facets are ranked using the latest query logs from image search and the Flickr data. Section 6 provides a detailed overview of the facet ranking process.

Facet Server. The facet server is responsible for the interaction with the application. Given a user query a ranked list of facets and some additional meta-data is returned through the MediaFaces API. Serving of facets is done on demand when the user enters a query. The query is mapped to zero or more MediaFaces objects and the top ranked facets are served to the user. The facet server and the application of MediaFaces as part of the Yahoo! image search interface is described in Section 7.

Facet Builder. The facet builder is a communication module between the facet repository and all other components. When the facet extractor processes a new input source it passes the extracted facets to the facet builder, which is responsible for storing them in the facet repository. When the facet ranking module is activated it asks the facet builder for a list of facets to rank and then returns the ranked facets back to the facet builder which updates the scores in the facet repository. When the facet server receives an input query it asks the facet builder for a list of candidate facets to be served.

4. FACET REPOSITORY

The facet repository takes care of the back-end data storage of the MediaFaces system. The facet repository is organized around two main concepts: *objects* and *facets*. Following is a definition of the two concepts and examples of both can be found in Figure 4.

id	14	**id**	17
name	Bangalore, India	**name**	George Clooney
aliases	Bangalore	**aliases**	George T. Clooney
	Bengaluru	**type**	person
type	location	**subtypes**	actor
subtypes	city		director
details	lon=77.5...	**details**	dob=1961-05-06
	lat=12.9...	**sources**	Y! Movies
sources	GeoPlanet		Y! OMG
	Wikipedia		

(a) MediaFaces objects

source	Bangalore, India	**source**	George Clooney
target	Cubbon Park	**target**	Ocean's Eleven
type	subsumes	**type**	played in

(b) MediaFaces facets

Figure 4: Examples of objects and facets in the MediaFaces repository.

Object. An object is defined as a real-world object or entity. Examples of objects can be places such as *London, UK*, and *New York City*; celebrities such as *Jennifer Aniston*, and *Brad Pitt*; movies and tv-shows such as *Fight Club* and *Friends*; etc. Each object has a number of attributes:

- **id**, a unique identifier for the object;
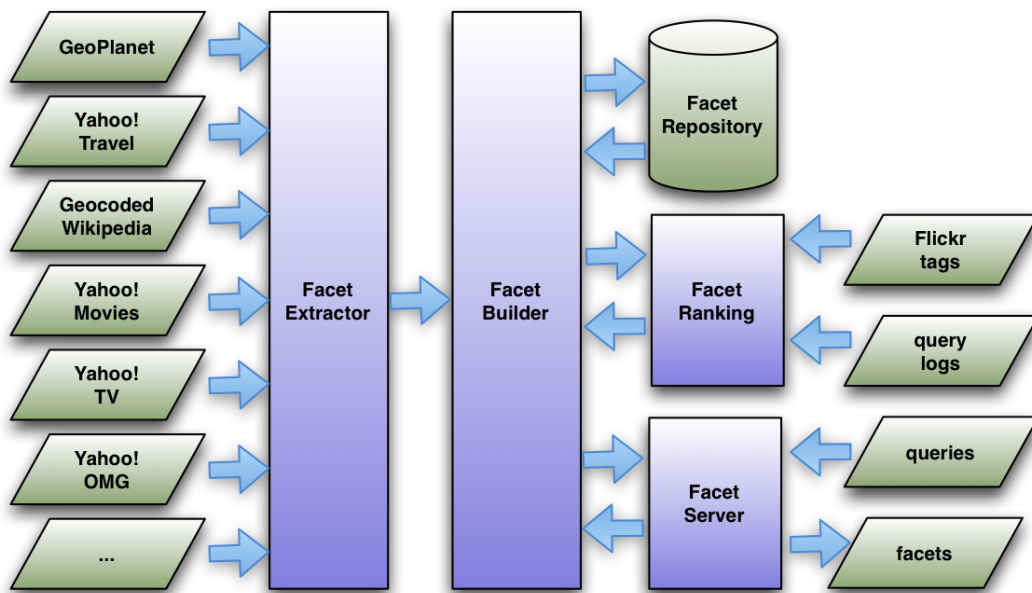- **name**, the common name under which the object is known;

Figure 3: System architecture of the MediaFaces system

- **aliases**, a list of alternative names for the object;

- **type**, a high level type for the object;

- **subtypes**, a list of fine grained types for the object;

- **details**, a attribute-value mapping that can be used to store additional attributes of the object;

- **sources**, a list of sources where the object has been detected.

Facet. A facet is defined as a directed mapping from one object to another. Each facet has a number of attributes:

- **source object**, the object to which the facet belongs;

- **target object**, the object that represents the facet;

- **type**, the type of the facet relation.

Figure 4 shows an example of two MediaFaces objects and two facets. For the sake of clarity, the examples in this paper identify the source and target objects by their name but in the actual system they are identified by their unique identifier.

5. FACET EXTRACTION

The facet extraction module of the MediaFaces system is responsible for processing incoming content sources and extracting objects and facets. The system is general enough to handle any type of data but in the first instance of the system we included only sources with geographic and celebrity information. We carefully selected a set of semi-structured sources where the objects and facets are marked up explicitly. Most of the sources are maintained internally at Yahoo! The sources that have been incorporated are:

- **GeoPlanet^{TM}**, a resource for managing all geo-permanent named places on Earth[8].

---

[8]http://developer.yahoo.com/geo/geoplanet/

Domain	Source	Object type
Geo	GeoPlanet	Countries, cities, states, lakes mountains, landmarks, etc.
Geo	Y! Travel	Attractions
Geo	Wikipedia	Geo-coded Wikipedia pages
Celeb	Y! Movies	Actors, directors, and movies
Celeb	Y! TV	Actors, directors, and tv-shows
Celeb	OMG!	Celebrities

Table 1: Objects extracted from the semi-structured sources.

- **Yahoo! Travel**, a comprehensive travel guide[9].

- **Wikipedia**, a collaboratively edited encyclopedia[10].

- **Yahoo! Movies**, a movie information portal[11].

- **Yahoo! TV**, a TV information portal[12].

- **Yahoo! OMG**, a celebrity gossip and news site[13].

Table 1 shows an overview of the type of objects extracted from the various sources and their domain. Since objects are marked up explicitly in the sources there was no need of performing entity recognition. In the case of Wikipedia, each geo-coded article is considered to be an object.

Table 2 shows an overview of the kind of facets extracted from the various sources. In the case of GeoPlanet we use the built-in object hierarchy to map between places (countries, states, and cities) and points of interest (mountains, lakes, etc.) For Yahoo! Travel attractions and geo-coded Wikipedia pages we use the associated longitude-latitude

---

[9]http://travel.yahoo.com/
[10]http://wikipedia.org/
[11]http://movies.yahoo.com
[12]http://tv.yahoo.com
[13]http://omg.yahoo.com/

Source	Facet	Type
GeoPlanet	Place* → Point of interest**	subsumes
Y! Travel	Place* → Attraction	subsumes
Wikipedia	Place* → Geo-coded page	subsumes
Y! Movies	Person → Movie	played in
Y! Movies	Movie → Person	has cast
Y! Movies	Person → Person	co-acted with
Y! TV	Person → TV show	played in
Y! TV	TV show → Person	has cast
Y! TV	Person → Person	co-acted with
OMG!	Person → Person	appeared with

* Where place can be a country, state, or city.
** Where point of interest can be a lake, mountain, landmark, etc.

Table 2: Facets extracted from the semi-structured sources.

coordinate to map the attraction/page to countries, states and cities from the GeoPlanet resource. For Yahoo! Movies and Yahoo! TV the facets are explicitly defined in the data structure. In case of OMG!, we add a facet for every pair of celebrities that appear in the same news article. In the OMG! feed the celebrities are also explicitly marked-up, thus there is no need for entity extraction.

Having run the extraction process for all the input data sources the facet repository contains millions of objects and tens of millions of facets. Many of the objects have hundreds of outgoing facets. It is thus important to be able to rank them in order to be able to serve the most relevant ones to the user.

6. FACET RANKING

Ranking of candidate facets is based on the statistical analysis of query terms and query sessions that are derived from the image search logs. In addition, the tags associated with the public photos in Flickr are used to complement the knowledge derived from the search logs. After pre-processing the three sources, a common format is derived that is used to perform the statistical analysis. We have experimented with various metrics, both symmetric and a-symmetric, to produce a ranking of facets for a given (source) object of interest. In this section we'll describe how the sources are processed to derive the data in the form of the common format, we give an overview of the metrics evaluated, and explain how an aggregated ranking is derived based on a linear combination of the three sources.

6.1 Common Format

For the statistical analysis, we need to transform the data from the three different sources into the common format:

```
CFDATA      ::=
    EventId <tab> UserId <tab> TimeStamp
    <tab> EVENTDATA <newline>
EVENTDATA  ::=
    OBJECTENTRY ( <comma> OBJECTENTRY) *
OBJECTENTRY ::=
    (
      Object |
      <open_bracket>
        Object <pipe>
        Object ( <comma> Object)*
      <close_bracket>
    )+
```

For instance the image search query "*Cubbon park in Bangalore India*" entered by a user generates the following result based on the query term analysis as will be explained in more detail below:

```
e1001 u01 t1 cubbon+park,{bangalore+india|bangalore,india}
```

EventId
> The EventId (`e1001`) is a unique identifier within the defined event space. For Flickr, the event space is the collection of public photos, and the photo-id would uniquely identify a photo in this space. In case of query term analysis this is a page view, and for query session analysis a set of consecutive page-views that take place within a certain time-window.

UserId
> A UserId (`u01`) uniquely identifies a particular user. Typically this can be a browser cookie or a user's (anonymized) account id.

TimeStamp
> The time stamp (`1256395594`) registers the start time of the event, and is stored in Unix time format.

EVENTDATA
> The section EVENTDATA describes the objects that have been detected during the event.
> (`cubbon+park,{bangalore+india|bangalore,india}`)

OBJECTENTRY
> An OBJECTENTRY can be a single object reference such as "*cubbon park*", or a composed reference. This might occur whenever a phrase "*Bangalore, India*" is detected. Besides the phrase we have also objects in the repository that refer to the individual terms: "*Bangalore*" and "*India*".

Once the data is formatted correctly, the same set of statistical metrics can easily be derived from the different sources based on the co-occurrence analysis of objects within a given event.

6.2 Query Term Analysis

Our main source for ranking facets is based on the query term analysis. The queries entered by users in the Yahoo! image search engine provide us with a massive amount of information. We have collected a large set of queries spanning a period of several months. As for Web search the queries posed by users tend to be short. In their work, Bendersky and Croft report that 90% of the query volume consists of queries with a length $l(q) \leq 4$ [3]. For the ranking the candidate facets in the MediaFaces system we are particularly interested in multi-term queries as input for the co-occurrence analysis. However a straight-forward tokenization of the query based on word-boundaries is insufficient, as the vast majority of the objects in our facet repository consist of phrases, e.g. person names, movie titles, location names, etc. We therefore need to derive a more intelligent segmentation of a query.

To that purpose, we first tokenize the query based on the word boundaries, and apply an NFD normalization[14] of the tokens. We then use a sliding window over the tokens to find

---

[14]More information at: http://en.wikipedia.org/wiki/Unicode_normalization.

User query:	Cubbon park in Bangalore, India
Tokenization:	Cubbon+park+in+Bangalore+India
Normalization:	cubbon+park+in+bangalore+india
Segmentation:	cubbon+park+in+<u>bangalore+india</u>
Object detection:	cubbon+park,
	{bangalore+india|bangalore,india}

Figure 5: Transformation steps for query term analysis.

the object references in the query and segment the query. Once the query is segmented, the data can be transformed in the common format. Figure 5 illustrates how the objects are detected for our example query in four steps.

Once the query is segmented, the event information can be encoded using the common format. Note that we have detected four object references, and that the term "*in*" does not match with any object in the database and is therefore omitted.

6.3   Query Session Analysis

The query session analysis uses the same query log data as is collected for the query term analysis. The event space is a query session, which is defined as a set of consecutive queries issued by the same user within a certain amount of time, for example 15 minutes. Consider the following scenario, where a user (*u01*) first searches for "*India*", then expands his query into "*Bangalore, India*", and finally decides to search for "*Cubbon park*" within a 15 minute time frame. The following data will be collected for this query session:

```
e9001 u01 t2 india,bangalore+india,cubbon+park
```

For the query session analysis we tokenize and normalize the query as is done for the query term analysis, but there is no further segmentation of the query. Only whole queries are matched against the objects in the facet repository when doing the object detection. Due to the exploratory nature of image search, a user is likely to enter several queries during one session. We observe that the average number of queries entered in a query session exceeds the average number of query terms. Furthermore, the user is likely to change to several related topics within a session. We refer to this behaviour as a lateral exploration. The objective of MediaFaces is to support a faceted exploration, rather than a lateral exploration. We therefore appreciate the outcome of the query session analysis to be inferior to the query term analysis.

6.4   Flickr Tag Analysis

The Flickr tag analysis is based on the tags defined for a large set of 250 million photos that are publicly available on Flickr. An event is defined around the tags used to annotate a photo. Suppose that a user has annotated their photo with the tags: *Cubbon park, Bangalore, India*. We can then for each of the three tags simply apply the tokenization and normalization as used for the other two sources. However, we preserve the tag boundaries as defined by the user, which leads to the following result:

```
e8008 u01 t3 cubbon+park,bangalore,india
```

src	trg	sid	tid	P(trg | src)
bangalore	cubbon+park	21	345	0.0034
bangalore+india	cubbon+park	21	345	0.0016
...	...	...	...	...
india	bangalore	16	21	0.064

Table 3: Conditional user probabilities and mapping to objects.

6.5   Statistical Analysis

In this subsection we explain the procedure followed for computing the ranking of facets per source. Continuing with the most complex example derived during the query term analysis:

```
e1001 u01 t1 cubbon+park,{bangalore+india|bangalore,india}
```

The first step is to compute all possible co-occurring objects for this event:

```
cubbon+park - bangalore+india
cubbon+park - bangalore
cubbon+park - india
bangalore - india
```

We can now directly compute a series of metrics, for example consider the following variant of the conditional probability:

$$P(target|source) = \frac{\mid source \bigcap target \mid}{\mid source \mid},$$

with $|source|$ defined as the number of users that have used a source object in an event, and $\mid source \bigcap target \mid$ as the number of users that have used both the source and target object in an event.

Rather than counting the absolute number of times an object, or pair of objects appears, we count the number of distinct users, using that object, or pair of objects. This makes the metric less prone to the impact a single user can have on the probability score.

Besides the conditional probability, we have experimented with an number of other metrics, and combinations thereof:

- **atomic metrics**: probability, entropy.

- **symmetric**: joint probability, point-wise mutual information (PMI), cosine similarity.

- **a-symmetric**: (reverse) conditional probability, (reverse) KL divergence.

Based on an empirical evaluation, of which we cannot disclose the details, we have found that the conditional user probability, as described above, performs best across all three sources. This performance is followed closely by the joint user probability and PMI metrics.

6.6   Aggregated Ranking

To compute the final ranking of facets for a given object of interest, we first map all object references to their corresponding object ids. Continuing our Bangalore example, Table 3 illustrates the consequence of this.

The object references *bangalore* and *bangalore+india* refer to the same object (*21*) in the facet repository, but have different probabilities. As we are working with real

world data, the same object can be referred to by different names, and different objects are referred to by the same name. These two problems are unavoidable. The first problem is solved by choosing the maximum probability as the facet score: $P(345|21)_{max} = 0.0034$. The second problem most dominantly occurs for locations and is handled by the FacetBuilder.

Finally, when for each facet in the facet repository its conditional user probability is computed per ranking source, we compute an aggregated score per facet using a linear combination. Most weight is given to the conditional probability score of the query term analysis, followed by the Flickr tag analysis and query session analysis respectively. The motivation is twofold: First, we want MediaFaces to support a faceted search experience. Both the query term analysis and the Flickr tag analysis are good at finding facets of a given object, while the query session analysis supports a more lateral search experience such as celebrities that share certain characteristics, but do not have a direct (faceted) relationship. Similar one can think of major cities in Europe like: Paris, London, Rome, Barcelona, Amsterdam, which clearly appear to be related according to the query sessions analysis. Second, we prefer query term analysis over Flickr tag analysis due to the nature of image search, which tends to be broader than Flickr. For instance, it has a better coverage of the celebrity and entertainment business.

7. FACET SERVER

After extracting and ranking the facets we are ready to serve them to the user when appropriate. The process of on-line serving of facets is shown in Figure 6. The user types a query in the search box. The query is mapped to zero or more query objects in our facet repository. If no object matches the query, normal image search results are shown. If one object matches the query, the facets are shown. If multiple objects match the user is given the option to refine their query to indicate which object they meant.

7.1 Mapping Queries to Objects

The first task in the on-line processing is to map the query to zero or more objects. We do that using an exact string match on the object name or one of its alias names. The string match is insensitive to case and punctuation. As an example, the Bangalore object shown in Figure 4 would be mapped to the queries *bangalore*; *bangalore, india*; etc.

Each query string can be mapped to zero or more query objects. If no object is mapped the normal image search results page is shown. If one object is mapped the facets are displayed, as shown in Figure 1. For each facet, we display the name, an automatically selected and generated thumbnail, and the number of corresponding images. A query string can also be mapped to multiple objects. As an example the query *cambridge* is mapped to both *Cambridge, England* and *Cambridge, MA*. In this case the user is given the opportunity to disambiguate the query and choose either one of the objects (see Figure 7). If the user chooses one of the disambiguation objects the facets for that source object are shown.

7.2 Mapping an Object to Facets

When a query object has been uniquely determined the system retrieves a ranked list of facets originating in the query object. The facet object list is processed in a decreas-

ing relevance order and facets are chosen for displaying if they match the following criteria:

- *The facet has sufficient number of photos.* We can estimate the number of photos returned by composing a query of the concatenation of the names of the source and the target objects of the facet. If the number of photos fails to fill a full result screen we do not select the facet.

- *The target object string is not a near duplicate of previous target object string.* In some cases when an object is ingested from multiple sources two instances of the object may exist in the facet repository with identical or near identical name. E.g., one source may refer to the famous New York City as *Empire State Building*, while another may refers to it simply as *Empire State*. In this case we check if the currently processed object name overlaps with a previously processed object name and choose the object with the longest name.

When 10 facets have been chosen the processing terminates and the list is returned to the front-end.

7.3 Displaying Facet Types

In the geography query example shown in Figure 1 all the target objects of the facets are of the same type – e.g. *location*. In the case of celebrities the system offers a variety of types. E.g., for a given celebrity the facet list retrieved may contain other people related to the celebrity or movies the celebrity played in. This information can be used by the interface to further organize the related facets. Figure 8 shows the facet lists generated for two geography queries and two celebrity queries. For the celebrity queries the facet lists are broken up into related people, related movies, and related TV-shows. The objective of this categorization is to help the user to get a better overview of the facets displayed.

8. CONCLUSIONS

In this paper we presented MediaFaces, a system for serving facets given a query, which is currently used in production by Yahoo! image search. The system aids the user in their exploratory search tasks, which are particularly frequent in image search. MediaFaces enables a faceted search experience by aggregating high-quality content that is available on and off the Yahoo! network. Combined with the analysis of image search query logs and Flickr tagging behaviour we are able to present the user with conceptual query refinements. Finally, We have shown how we serve top-k facets as a response to a user query in the Yahoo! image search setting.

9. ACKNOWLEDGEMENTS

10. ADDITIONAL AUTHORS

Ramu Adapala (Yahoo!), Lluís García Pueyo (Yahoo! Research), Abhinav Katiyar (Yahoo!), Kaushal Kurapati (Yahoo!), Mridul Muralidharan (Yahoo! Research), Sudar Muthu (Yahoo! Research), Vanessa Murdock (Yahoo! Research),
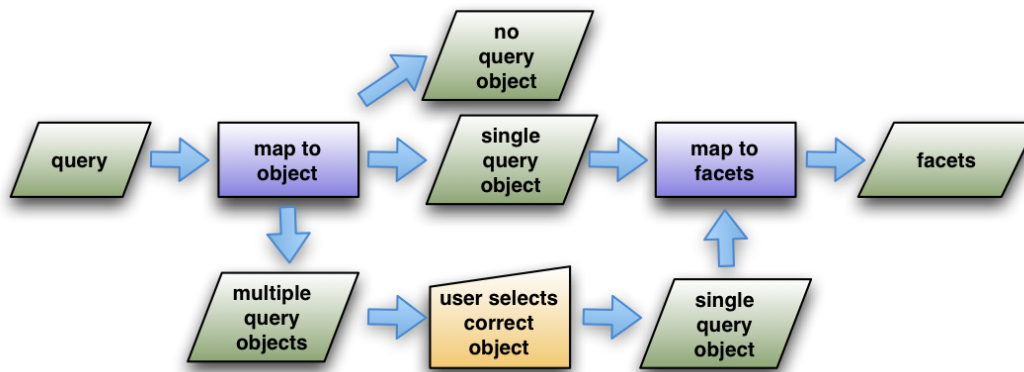
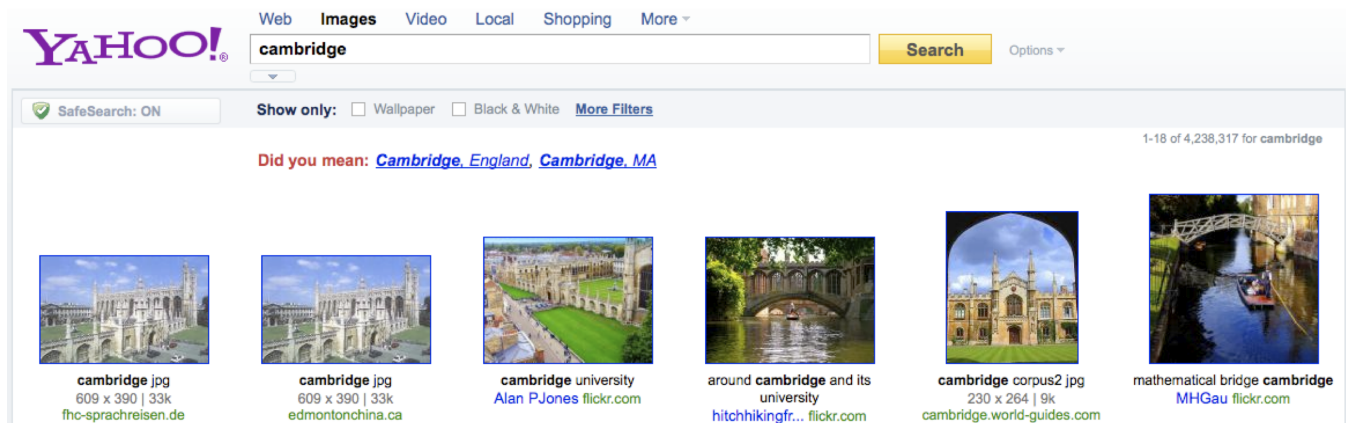Figure 6: System process flow when a query is entered into the system.

Figure 7: Object disambiguation.

Polly Ng (Yahoo!), Anand Ramani (Yahoo!), Anuj Sahai (Yahoo!), Sriram 'Thiru' Sathish (Yahoo!), Hari Vasudev (Yahoo!), and Upendra Vuyyuru (Yahoo!)

11. REFERENCES

[1] R. Agrawal, S. Gollapudi, A. Halverson, and S. Ieong. Diversifying search results. In *WSDM '09: Proceedings of the Second ACM International Conference on Web Search and Data Mining*, pages 5–14, New York, NY, USA, 2009. ACM.

[2] P. G. Anick and S. Tipirneni. The paraphrase search assistant: terminological feedback for iterative information seeking. In *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 153–159, New York, NY, USA, 1999. ACM.

[3] M. Bendersky and W. B. Croft. Analysis of long queries in a large scale search log. In *WSCD '09: Proceedings of the 2009 workshop on Web Search Click Data*, pages 8–14, New York, NY, USA, 2009. ACM.

[4] R. D. Burke, K. J. Hammond, and B. C. Young. Knowledge-based navigation of complex information spaces. In *Proceedings of the 13th National Conference on Artificial Intelligence*, pages 462–468, 1996.

[5] R. Capra, G. Marchionini, J. S. Oh, F. Stutzman, and Y. Zhang. Effects of structure and interaction style on distinct search tasks. In *JCDL '07: Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries*, pages 442–451, New York, NY, USA, 2007.

[6] L. Cornett. Welcome to the new Yahoo! search. http://www.ysearchblog.com/2009/09/22/welcome-to-the-new-yahoo-search/, September 2009.

[7] R. Datta, D. Joshi, J. Li, and J. Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Comput. Surv.*, 40(2):1–60, 2008.

[8] B. M. Fonseca, P. Golgher, B. Pôssas, B. Ribeiro-Neto, and N. Ziviani. Concept-based interactive query expansion. In *CIKM '05: Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 696–703, New York, NY, USA, 2005. ACM.

[9] J. S. Hare, P. H. Lewis, P. G. B. Enser, and C. J. Sandom. Semantic facets: an in-depth analysis of a semantic image retrieval system. In *CIVR '07: Proceedings of the 6th ACM international conference on Image and video retrieval*, pages 250–257, New York, NY, USA, 2007. ACM.

[10] M. A. Hearst. Clustering versus faceted categories for information exploration. *Commun. ACM*, 49(4):59–61, 2006.
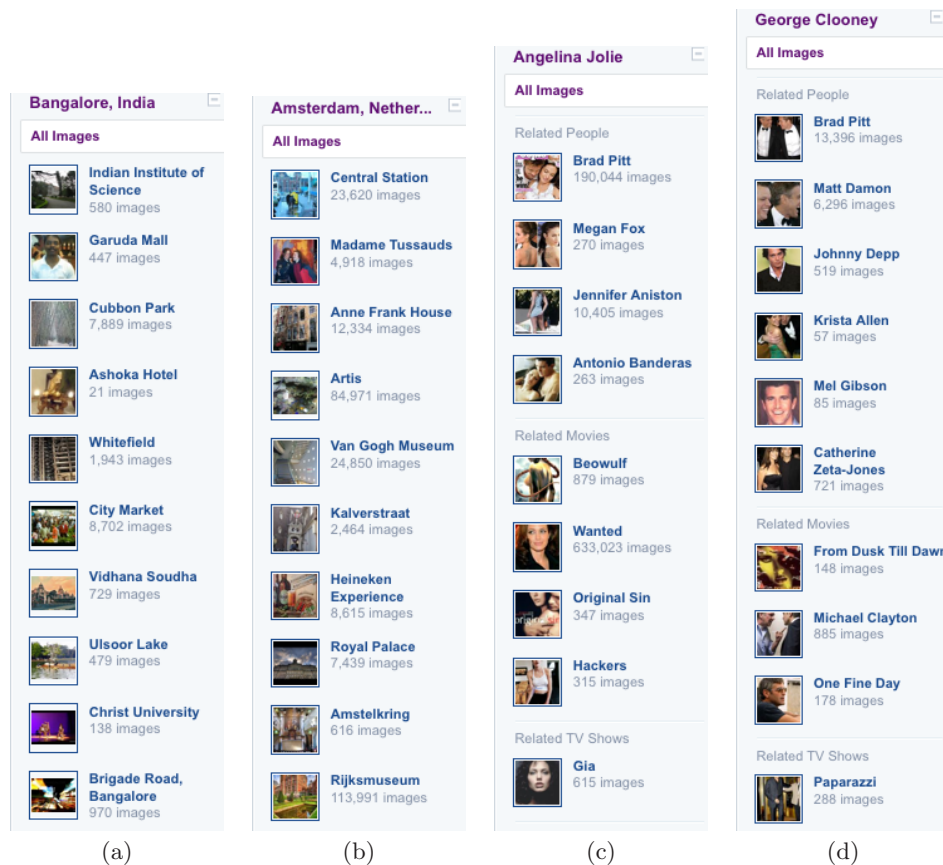
[11] D. Heesch. A survey of browsing models for content

Figure 8: Example facets for (a) Bangalore, India, (b) Amsterdam, the Netherlands, (c) Angelina Jolie, (d) George Clooney.

based image retrieval. *Multimedia Tools Appl.*, 40(2):261–284, 2008.

[12] D. Heesch, A. Yavlinsky, and S. Rüger. Nnk networks and automated annotation for browsing large image collections from the world wide web. In *MULTIMEDIA '06: Proceedings of the 14th annual ACM international conference on Multimedia*, pages 493–494, New York, NY, USA, 2006. ACM.

[13] B. J. Jansen, A. Spink, and T. Saracevic. Real life, real users, and real needs: a study and analysis of user queries on the web. *Information Processing & Management*, 36(2):207–227, March 2000.

[14] H. Joho, C. Coverson, M. Sanderson, and M. Beaulieu. Hierarchical presentation of expansion terms. In *SAC '02: Proceedings of the 2002 ACM symposium on Applied computing*, pages 645–649, New York, NY, USA, 2002. ACM.

[15] R. Kraft and J. Zien. Mining anchor text for query refinement. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 666–674, New York, NY, USA, 2004. ACM.

[16] R. Stewart, G. Scott, and V. Zelevinsky. Idea navigation: structured browsing for unstructured text. In *CHI '08: Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, pages 1789–1792, New York, NY, USA, 2008.

[17] R. H. van Leuken, L. Garcia, X. Olivares, and R. van

Zwol. Visual diversification of image search results. In *WWW '09: Proceedings of the 18th international conference on World wide web*, pages 341–350, New York, NY, USA, 2009. ACM.

[18] R. van Zwol, V. Murdock, L. Garcia, and G. Ramirez. Diversifying image search with user generated content. In *Proceedings of the International ACM Conference on Multimedia Information Retrieval (MIR 2008)*, Vancouver, Canada, October 2008.

[19] K. Weinberger, M. Slaney, and R. van Zwol. Resolving tag ambiguity. In *Proceedings of the 16th International ACM Conference on Multimedia (MM 2008)*, Vancouver, Canada, November 2008.

[20] P. Yee, K. Swearingen, K. Li, and M. Hearst. Faceted metadata for image search and browsing. In *Proceedings of ACM CHI*, 2003.

[21] J. Zhang and G. Marchionini. Evaluation and evolution of a browse and search interface: Relation browser++. In *dg.o 2005: Proceedings of the 2005 national conference on Digital government research*, pages 179–188. Digital Government Society of North America, 2005.