Data Mining, Project 5 report
Mining Flickr

Armin Eftekhari, Alejandro Weinstein

December 4, 2009

1 Introduction

Flickr is one of the most popular image hosting website, with more than 4 billion pictures [3]. For each picture, the community can add comments, tags, or call the picture his or her favorite. Also, the details of the camera used to take the picture are available. So, an interesting question arise: is there any correlation between the popularity of a photo and the camera used to take it? The purpose of this project is to try to answer this, and to find other interesting patterns.

2 Generating the data set

There is no dataset directly available with the data needed to analyze Flickr. However, there is a Flickr API that allows to build such dataset. We wrote a Python script, using the Python "Flickr API kit" [2] to do that.

We decided to build a dataset with the following attributes:

Id: the unique id used by Flickr to identify a photo.

Views: the number of views of the photo.

Location: the location of the photo.

Comments: the number of comments of the photo.

Tags: the number of tags of the photo.

Favorites: the number of Flickr users that call the photo a favorite.

Make: the maker of the camera used to take the photo.

Model: the model of the camera used to take the photo.

Since it is not feasible to get information about all the pictures in Flickr, a key point at the moment of building the dataset is to determine how to sample the website. We decided to get the data of 100 picture for each of the most popular tags. Unfortunately, we didn't find an API function to get the most popular tags, so we copy the list of the most popular tags from the "All time most popular tags" page [1]. We also looked at photos at least one year old, to guarantee that the pictures has been exposed to the community for a time long enough.

We built the dataset in four steps:

1. Get the 100 ids for each of the popular tags: For each of the popular tags, we get the ids of 100 photos. In this step we ask for pictures at least one year old. We got 14400 records.
2. Eliminate the duplicate ids. Since a given photo can be associated to more than one tag, it is possible to have duplicated ids. We eliminated the duplicate ids, reducing the 14400 records to 7342.
3. Get the data for each photo id.
4. Write the data as a CSV file.

It is interesting to note that the time Flickr take to respond to each API call is roughly one second, and consequently, building the dataset took several hours.

The examination of the dataset revealed that the *Make* attribute was inconsistent, since we got slightly different names for a given maker. For instance, "Olympus" cameras were labeled as "Olympus Imaging Corp.", "Olympus optical Co. Ltd" or "Olympus corporation". We wrote another Python script to solve this inconsistency, reducing the number of unique manufacturers from 62 to 36.

3 Summary statistics

After finishing the preprocessing of the dataset, the next step on the data mining process is to generate some summary statistics, to get a general idea of the structure of the data. Table 1 shows the statistics for the numerical attributes. For the categorical attributes "make" and "model", the mode is "Canon" and "Nikon D200", respectively.

Figure 1 shows the histograms for the numerical attributes. As can be seen, all the numerical attributes share a *long tail* behavior, where most of the photos are unpopular (in terms of a low number of views, comments, etc), and a relatively small amount of the photos are very popular. Figure 2 shows an histogram for the "Make" attribute.

Table 1: Summary statistics for the numerical attributes.

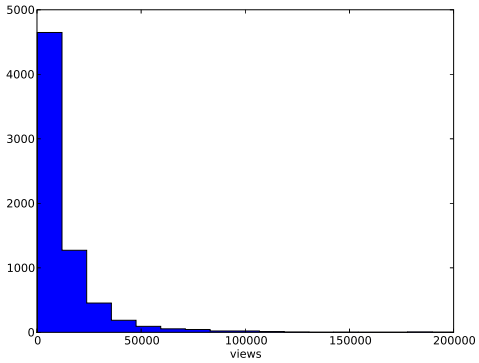
views	
Min	164
Max	1184897
Mean	14778.5
Std dev	35718.6

comments	
Min	6
Max	2891
Mean	150.2
Std dev	161.8

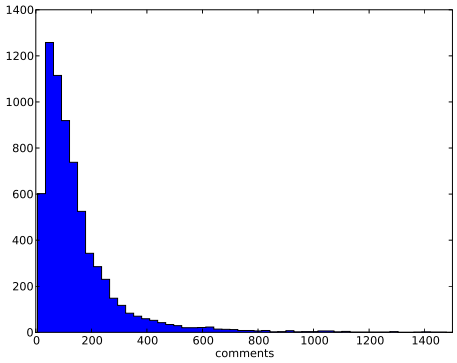
tags	
Min	2
Max	279
Mean	30.2
Std dev	17.5

favorites	
Min	0
Max	8539
Mean	243.0
Std dev	348.2

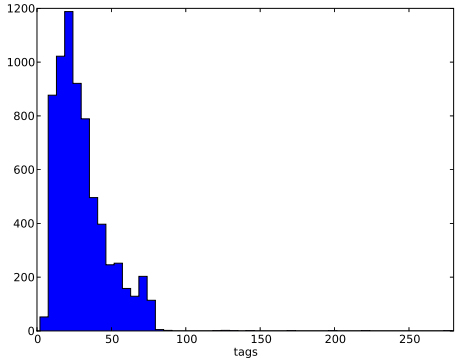
First of all, it can be noted that a significant amount of the photos (44%) don't have information about the maker of the camera used to take the picture. As we will discuss later in section 5, this will play a detrimental role in our data mining analysis. Secondly, we can see that the two mayor players are *Canon* and *Nikon*, followed by a handful of other brands with a significant share of the photos. Most of the remaining manufactures are associated with only one photo.



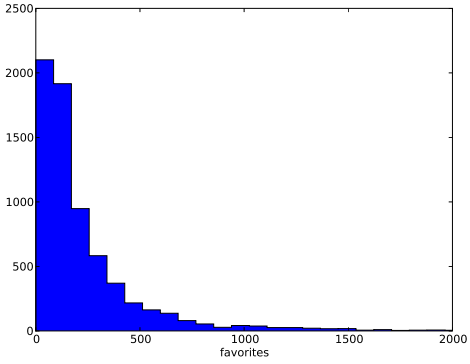
(a) "Views" histogram



(b) "Comments" histogram



(c) "Tags" histogram



(d) "Favorites" histogram

Figure 1: Histogram for "views", "comments", "tags" and "favorites".

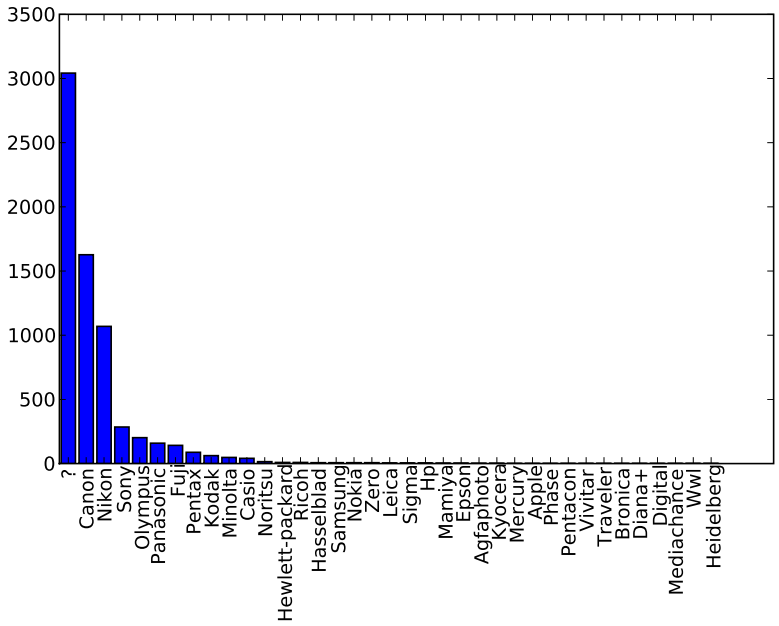


Figure 2: Histogram for the "maker" attribute.

Figure 3 shows a scatter plot, where the x axis correspond to the number of comments and the y axis correspond to the number of views. The color of each point is mapped to the logarithm of the number of favorites[1]. This plot illustrate, once again, the *long tail* behavior of the data, where we can see a big cluster of photos close to the origin, and a set of very popular outliers.

Notice that all the plots for the summary statistics were produced using Python and the Matplotlib library, since we think that the ones produced by Weka are very low quality.

[1]A logarithmic map is used due to the huge range of the variable. Otherwise, most of the points end with the same color

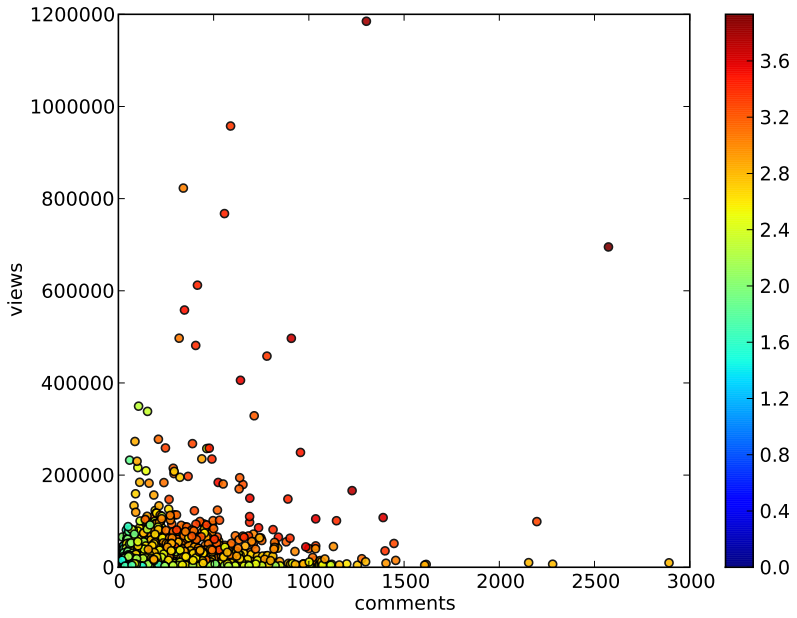


Figure 3: Scatter plot. The color represente the logarithmic of the "favorite" attribute.

4 Clustering

5 Classifiers

6 Discussion

The Flickr API provides a wonderful opportunity to explore very interesting datasets. However, an important point to consider in order to get meaningful results is the sampling strategy. While the sampling strategy we used is reasonable, there are other ways to sample the data that should be explored.

It was not surprising that our data showed a long tail behavior, since this is typical for data originated on social networks. This fact leads us to think that it would be interesting to investigate some techniques specially tailored to this kind of distribution.

This work triggers some new interesting question that could be answering by mining Flickr, like for instance:

- Using geolocation information of the photos, can we identify a correlation between location and type of camera used? Producing, for examples, a statement like. "people in the USA use mostly Canon cameras, while people in Europe use mostly Nikon".

- So far we ignored the users, but some patterns may arise by looking at this dimension. For instance we could try to answer questions like, "are some users with a lot of popular photos?", "can we identify "different classes" of users?", etc.
- Can we found some semantic of the photos by looking at the data?

References

[1] Flickr. Popular tags on flickr, 2009. [Online; accessed 29-November-2009 http://www.flickr.com/photos/tags/].

[2] Stvel.eu. Projects—python flickr api kit, 2009. [Online; accessed 29-November-2009 http://stuvel.eu/projects/flickrapi].

[3] Wikipedia. Flickr — wikipedia, the free encyclopedia, 2009. [Online; accessed 29-November-2009 http://en.wikipedia.org/w/index.php?title=Flickr&oldid=326710624].