Generating Summaries and Visualization for Large Collections of Geo-Referenced Photographs

Alexandar Jaffe,
Mor Naaman
Yahoo! Research Berkeley
Berkeley, CA, USA
ajaffe@cs.washington.edu,
mor@yahoo-inc.com

Tamir Tassa
Department of Mathematics
and Computer Science
The Open University of Israel
Ra'anana, Israel
tamirta@openu.ac.il

Marc Davis
Yahoo! Inc.
Sunnyvale, CA, USA
marcd@yahoo-inc.com

ABSTRACT

We describe a framework for automatically selecting a summary set of photos from a large collection of geo-referenced photographs. Such large collections are inherently difficult to browse, and become excessively so as they grow in size, making summaries an important tool in rendering these collections accessible. Our summary algorithm is based on spatial patterns in photo sets, as well as textual-topical patterns and user (photographer) identity cues. The algorithm can be expanded to support social, temporal, and other factors. The summary can thus be biased by the content of the query, the user making the query, and the context in which the query is made.

A modified version of our summarization algorithm serves as a basis for a new map-based visualization of large collections of geo-referenced photos, called Tag Maps. Tag Maps visualize the data by placing highly representative textual tags on relevant map locations in the viewed region, effectively providing a sense of the important concepts embodied in the collection.

An initial evaluation of our implementation on a set of geo-referenced photos shows that our algorithm and visualization perform well, producing summaries and views that are highly rated by users.

Categories and Subject Descriptors

H.3.3 [**Information Systems**]: Information Storage and Retrieval—*Information Search and Retrieval*

General Terms

Algorithms, Human Factors

Keywords

Photo Collections, Geo-Referenced Photos, Summarization, Clustering, Image Search, Collection Visualization

1. INTRODUCTION

With the popularization of digital photography, people are now capturing and storing far more photographs than ever before. Indeed, we are moving towards Susan Sontag's 1977 vision of a world where "everything exists to end up in a photograph" [18]. As a result, billions of images, many of which are on the Web, constitute a growing record of our culture and shared experience. Viewing and interacting with such collections has a broad social and practical importance. However, these collections are inherently difficult to navigate, due to their size and the inability of computers to understand the content of the photographs. The prospects of 'making sense' of these photo collections has become largely infeasible.

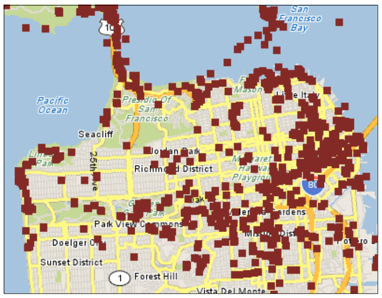
Some steps forward have been made through geo-referencing of digital photographs, whereby photos are connected to metadata describing the geographic location in which they were taken [12, 19]. Capture devices such as camera-phones and GPS-enabled cameras can automatically associate geographic data with images[1] and will significantly increase the number of geo-referenced photos available online. Already, an increasing number of photographs on the Web are associated with GPS coordinates. Such geo-referenced photos can be categorized geographically or displayed on a digital map, providing a rich spatial context in which to view subsets of a collection. Yet even here, we run into the problem of being able to filter, sort and summarize the data. The viewable space inevitably becomes cluttered after the data set has surpassed a certain size, with overlapping photographs making viewing and finding specific photographs ever more difficult as the collection grows. Figure 1(a) exemplifies the problem by showing an unfiltered view of San Francisco photos.

Our goal is thus to facilitate a system which can automatically select representative and relevant photographs from a particular spatial region. A result of our algorithm is illustrated in Figure 1(b), where a limited set of eleven photos that were selected by our system are marked on the San Francisco map. Such collection summaries will enable users to find items more easily and browse more efficiently through large scale geo-referenced photo collections, in a manner that improves rather than degrades with the addition of more photos.

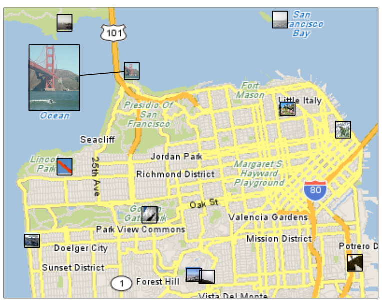
Selecting the most representative photos from a given region is a difficult task for several reasons. For instance:

[1]See, for example, the ZoneTag application at http://zonetag.research.yahoo.com.

- Image analysis alone is poor at understanding the semantic content of an image, making visual relevance insufficient for summarization.
- In multi-user sets, the biases of one user's data may skew the selection towards generally insignificant subjects.
- It is difficult for an automated system to learn and assess the relevance of photos without appropriate models of human interest, as the notion of relevance is not well defined, and often subjective.



(a) All San Francisco photos



(b) An automatic summary of San Francisco photos

Figure 1: All San Francisco photos from our dataset of 2200 geo-referenced photos, versus an automatic summary of photos, as generated by our system. One summary photo is enlarged for illustration.

We have designed and implemented a simple algorithm that attempts to address the challenges stated above. Our algorithm utilizes metadata-based heuristics that capitalize on patterns in users' photographic behavior. Foremost among these heuristics is the premise that photographs taken at a location 'vote' for the presence of something interesting at that location.

Our algorithm considers a multitude of spatial, social and temporal metadata (such as where the photo was taken, by whom, at what time), as well as textual-topical patterns in the data, such as textual tags associated with the photo. Furthermore, the algorithm can be tuned to bias the set of results using various factors such as the social network distance of the photographers to the user making the query.

The summarization algorithm can be used in a number of applications. For example, the algorithm could be used for geographic image *search*, returning a summary of photographs from a region in response to a search query (that can be specified as a text term or a map region). In addition, the summarization can be used to assist in map-based browsing of images, for example, by selecting a subset of representative photos to show according to the map's coverage and zoom level. With or without a map, summarization can help in browsing one's photos or a group of individuals' photos to get an overview of a location or discover personally interesting areas for further exploration; automatic travel guide is a scenario that comes to mind.

Key insights from our algorithm helped us generate a new way of visualizing large collections of geo-referenced photographs. We use the techniques we developed to generate map-based tag clouds, which are described in Section 6. "Tag Maps", as we call them, can be used to visualize the contents of the collection, giving a quick overview of the textual-topical concepts that appear in the data as well as their location, importance and recency. The photos themselves are not necessarily part of the visualization. Tag Maps concepts can be applied to many other multimedia (or other) applications that exhibit patterns in text and locations.

To summarize, the contributions of this paper are:

- A new approach for generating summaries of photo collections based on geographic as well as other contextual data associated with the photographic media (Section 3).
- An outline of the requirements and the useful features for these context-based summaries (Section 3).
- An implementation of an algorithm that generates such summaries using a public set of "geo-tagged" photographs (Section 4).
- A new map-based visualization technique for photo collections that helps indicate both the important regions on the map and the textual concepts represented in those regions (Section 6).
- A proposed evaluation for geo-referenced collection summaries; we use this evaluation to compare our algorithm to several baseline methods (Section 7).

In addition, Section 5 briefly touches on potential applications. We begin by discussing the related work.

2. RELATED WORK

Since 2003, a number of different research efforts have considered geographic location information associated with photographs. In [19], the authors describe WWMX, a map-based system for browsing a global collection of geo-referenced photos. Several similar map-based photo browsing systems appeared on the Web in the last few years[2], most of them using "geo-tagged" images from Flickr [5] for content. All of those systems face the problem of clutter in the map interface: as the number of photos available in each location grow, the full set of images cannot possibly be shown on the map at once. While some systems default to showing the most recent photos, the WWMX system tries to handle clutter by consolidating multiple photograph markers into a single marker according to the zoom level. In our system, we avoid clutter by utilizing the additional metadata to select the best set of photographs from a region, providing potentially a better selection than the "most recent" strategy, and a more meaningful one than the "consolidation" approach.

Several projects [12, 15] use geographic data to organize photo collections in novel ways, for example, by detecting

[2]like http://geobloggers.com and http://mappr.com

significant events and locations in a photo collection. Such structures could indeed be the basis for collection summarization. However, these projects considered personal photo collections only, and did not consider public shared pools of photos.

Looking at shared collections, some research [3, 4, 11, 14, 16] tries to use context (mostly location) information and sometimes visual features to identify landmarks in photographs. Visual analysis could be integrated in our system—once our algorithm recognizes significant locations, it can attempt to select a photo of a prominent landmark there.

Work in both [3, 11] considers, in a similar fashion to this work, patterns and distributions of textual terms that are associated with geo-referenced digital photos, and uses them to generate tag suggestions for new photographs. However, those projects are not designed to support collection summarization.

In the absence of location metadata, temporal metadata was also considered in the past for the purpose of photo collection summarization. In [8], Graham et al. describe an algorithm to heuristically select representative photos for a given time period in a personal collection, utilizing patterns in human photo-taking habits (later studied in [6]). Additional time-based work aims to detect events in personal collections (e.g., [2]), which could be the basis for collection summarization. However, again, all these projects considered single-photographer collections only. In public collections of timestamped photos, only when additional metadata is available (for example, the fact that all shared photos were taken in the same event), there exists the potential for time-based summaries [13].

Another possible approach for summarizing photo collections is using textual tags that are associated with the image. In Flickr [5], popular tags have pre-computed clusters of related tags. For example, the "San Francisco" tag on Flickr has several associated tag clusters[3]: "california, bridge, goldengate"; "baseball, giants, sbcpark", "deyoung, museum", "sfo, airport" and "halloween, castro". These clusters can potentially be used to generate a summary of San Francisco photos. This approach is not location-based, and the clusters often do not represent concepts that are distinct (e.g., one of Boston's clusters is "massachusetts, city, cambridge, building, architecture"). The tag clusters could possibly be used in conjunction with our method. In fact, we are using some tag-based computation to select summary photos. More directly related is a tag subsumption model [17] that can use the tag corpus to derive tags that are subsumed, for example, by the tag "San Francisco". Again, this approach can be integrated with our location-based summaries.

These projects, and others, consider various ways to alleviate the difficulties of browsing large collections of photographs, but do not provide effective ways to summarize multi-user photo collections or visualize them using maps. We believe that the potential of a geographic-based summarization method is significant, especially in conjunction with the current state of the art.

3. THE SUMMARIZATION APPROACH

In this section, we define the problem of summarizing a photo collection, then describe the guidelines and insights that have informed the implementation of our summarization algorithm. In Section 4 we provide the details of the algorithm.

We formalize the summarization problem as that of producing a ranking on the collection in question. In other words, we summarize a set of photos by ordering the set and selecting the top ranked photos. More formally, we are looking at the following problem: Given an album of n photos, $\mathcal{A} = \{P_1, \ldots, P_n\}$, we wish to find an ordering ω of $\mathcal{A}$ such that any k-length prefix of $\omega(\mathcal{A})$ is the best possible k-element summary of $\mathcal{A}$. A summary is loosely defined as a subset that captures representativeness, relevance, and breadth in the original collection. These notions are captured through some of the following metadata attributes that are associated with the photos:

- **Location.** Photo P_i was taken at location (x_i, y_i).[4]
- **Time.** Photo P_i was taken at time t_i.
- **Photographer.** Photo P_i was taken by user u_i.
- **Tags.** Photo P_i was manually assigned the list of tags (i.e., textual labels) w_i.
- **Quality.** Photo P_i is associated with an externally derived parameter q_i that represents its quality.
- **Relevance.** Photo P_i is associated with a relevance factor r_i. Relevance can include arbitrary bias based on parameters such as recency, the time of day, the day of the week, the social network of the user, user attributes, and so forth.

Note that The relevance attribute can introduce subjectivity, allowing us, for example, to tune the results to the user who is making the query and the context of the query.

While there is no accurate formal model for what constitutes a "good" summary of a collection of geo-referenced photographs, we follow a few simple heuristics that try to model and capture human attention, as reflected in the set of photos taken in a region. Among these heuristics are the notions that:

- Photographs are taken at locations that provide views of some important object or landmark.
- A location is more relevant if the photos around it were taken by a large number of distinct photographers.
- If available, location-based patterns of textual tags can reflect the presence of an important landmarks in a location.

In addition to the heuristics listed above, a desired summary would also (a) represent a broad range of subjects, instead of thoroughly displaying a few, and (b) allow personal or query bias to modify the algorithm's results.

In the next section we describe the summarization algorithm that we developed based on these guidelines.

4. ALGORITHM FOR SUMMARIZATION

As described in Section 3, our summarization algorithm produces a ranking of the photos in the collection; each prefix of this ranking can serve as a collection summary of the corresponding size. Producing this ranking is a two-step process, a clustering step followed by a ranking step on the resulting clustering hierarchy. In particular:

[3]http://flickr.com/photos/tags/sanfrancisco/clusters/

[4]Notice that this 'photo origin location' is different than the 'target location', the location of the photographed object.

1. We apply a modified version of the Hungarian clustering algorithm [7] to our collection of photographs. This algorithm receives the photograph locations as an input, and organizes them into a hierarchical clustered structure.

2. We compute a score for each cluster in the hierarchy.

3. Finally, we generate a flat ordering of all photos in the dataset by recursively ranking the sub-clusters at each level, starting from the leaf clusters, and ending at the root.

Note that while the clustering is a fixed one-time computation, the ranking step can be re-evaluated, allowing users to specify a personal bias or preference towards any of the metadata features. Alternatively, the ranking can also be modified to utilize implicit bias in the query context (e.g., the identity of the user making the query).

To illustrate the process and the scoring mechanism we use a hypothetical example, presented in Figure 2. In this figure, a leaf node represents a single photograph, annotated with the identity of the photographer and a single textual tag (in practice, of course, more tags can be associated with each photo). The tree represents the hierarchy created by the clustering algorithm.

Next, we describe the algorithm in detail. First, we discuss the clustering algorithm that produces the clustering hierarchy. Then, we describe how to produce a ranking of all photos in a single node of the above mentioned clustering hierarchy, assuming that all nodes in the hierarchy are associated with scores. Finally, we show how we can generate such scores for the nodes in the hierarchy.

4.1 Clustering

Our method requires a hierarchical clustering algorithm; as noted above, we use the Hungarian clustering algorithm [7]. This algorithm identifies a hierarchy of clusters within a given dataset of n points, based only on the distances between those points.

In our system, the input to the clustering algorithm is a set of points in the plane, representing the locations of the photographs,[5]

$$\mathcal{A} = \{(x_i, y_i) \in \mathbb{R}^2,\ 1 \le i \le n\}\ . \qquad (1)$$

The output is a clustering of these photo locations, $C(\mathcal{A})$, where $C(\mathcal{A})$ is a tree. Each node in the tree represents a subset of $\mathcal{A}$, the root of the tree represents the entire set, the children of each node are a partition (or clustering) of the subset that is associated with that parent node, and the leaves of the tree are the points in $\mathcal{A}$.

The classical Hungarian method is an efficient algorithm for solving the problem of minimal-weight cycle cover. In that problem, one is given a weighted graph and needs to find a cover of that graph by disjoint cycles with minimal total weight. This algorithm serves as the basic building block for a clustering method that is dubbed *The Hungarian clustering method*. Viewing $\mathcal{A}$ as a complete weighted graph, where the weight of each edge is the Euclidean (geographic, in this case) distance between the two nodes that it connects, the Hungarian clustering method subjects that graph to the classical Hungarian method. The disjoint cycles, produced by the Hungarian method, are viewed as a partition of the data-set. The clustering algorithm then proceeds by hierarchical merging of the disjoint cycles, until the produced clusters are perceived as complete clusters (through some "completeness" criteria) and then the hierarchical merging stops. We use the Hungarian Clustering algorithm because of two features that it boasts: It is an hierarchical clustering algorithm, and it does not depend on the number of clusters as an input.

The clustering hierarchy $C(\mathcal{A})$ is used to create a ranking of all photos. In order to describe the ranking algorithm, let us first assume that the nodes in the hierarchy have been assigned a score that embodies the importance of the cluster of photos that corresponds to that node.

4.2 Ranking Framework

Given a hierarchical clustering $C(\mathcal{A})$ on the locations of all photographs, and a score for every node (cluster) in that hierarchy, our goal is then to produce a ranking of all items in the collection. We describe a recursive interleaving algorithm that uses the clustered structure and the corresponding scores in order to produce a natural flat ordering. In the next section we outline a way to generate the scores.

Going bottom up, the ranking algorithm considers each node $\mathcal{B}$ in the hierarchy $C(\mathcal{A})$ and outputs an ordering $\omega(\mathcal{B})$ that represents a ranking of photos in $\mathcal{B}$. Finally, when executing on the root node that corresponds to the entire set $\mathcal{A}$, we get the ordered sequence, $S := \omega(\mathcal{A})$, that describes a ranking of all photos in $\mathcal{A}$. Applying this algorithm to the example in Figure 2, a possible output could be the ranking $S = (6, 8, 4, 5, 7)$, where the numbers in the sequence correspond to the numerals of the leaves in the tree in Figure 2.

For simplicity of notations, we describe the action of the algorithm on the root node, $\mathcal{A}$. Actions on other nodes are performed in the same manner. We assume that we identified m sub-clusters in $\mathcal{A}$, $\mathcal{A} = \bigcup_{i=1}^{m} \mathcal{A}_i$; namely, node $\mathcal{A}$ has m direct descendents. In addition, assume that the photos in each sub-cluster of $\mathcal{A}$ have been ranked recursively according to this algorithm, and that each of the nodes A_i is associated with some score $s(\mathcal{A}_i)$ such that (without loss of generality)

$$s(\mathcal{A}_1) \ge s(\mathcal{A}_2) \ge \cdots \ge s(\mathcal{A}_m)\ . \qquad (2)$$

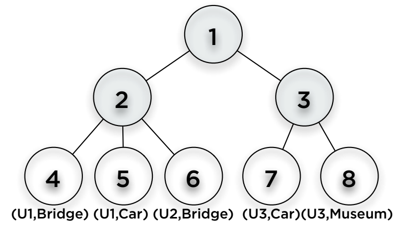


Figure 2: A sample hierarchy; the leaves are photos, each associated with a user and a single tag.

Our goal is to produce a ranking that would balance the contradicting properties of depth and breadth of coverage. In the field of Information Retrieval, some measures are used to balance results in terms of relevance (depth) and broadness (breadth) [1, 10, 20]; for various reasons, these measures are not applicable here. For our problem, depth requires

[5]For convenience, we use the same notation, $\mathcal{A}$, to denote the photo set as well as the set of photo locations.

that the photos in a cluster are selected from sub-clusters roughly according to the ratio of their scores. For example, consider the second level of the hierarchy in Figure 2, which consists of two clusters, denoted by $\mathcal{C}_2$ and $\mathcal{C}_3$, and assume that $s(\mathcal{C}_2) : s(\mathcal{C}_3) = 5 : 3$. We would like to interleave the photos from these two clusters so that in any section of the sequence S, the frequencies of photos from the two clusters relate to each other as closely as possible to their score ratio in the whole dataset, i.e., $5 : 3$. On the other hand, breadth requires that each sub-cluster should be represented to some extent early in the ranking of its parent cluster.

In order to attain some amount of depth, breadth, and consistency, we interleave photos from sub-clusters in the following manner. The ordered sequence of photos for $\mathcal{A}$ will have two parts: a short *header* $\mathcal{H}$ followed by a *trailer* $\mathcal{T}$, where $S(\mathcal{A}) = \mathcal{H}||\mathcal{T}$.

The header $\mathcal{H}$ will include a photo from all prominent sub-clusters. To that end, we define a threshold $0 < w < 1$, and then a cluster $\mathcal{A}_i$ is deemed prominent if

$$\frac{s(\mathcal{A}_i)}{\sum_{j=1}^{m} s(\mathcal{A}_j)} \geq w \ .$$

Assume that there are m' prominent sub-clusters among the m sub-clusters, with $0 \leq m' \leq m$. Then in view of assumption (2), the header is

$$\mathcal{H} = (\mathcal{A}_{1,1}, \mathcal{A}_{2,1} \cdots , \mathcal{A}_{m',1}) \ ,$$

where $\mathcal{A}_{i,1}$ is the most relevant photo from cluster $\mathcal{A}_i$. This header is then followed by a trailer, $\mathcal{T}$. In order to generate the trailer, we first remove from each sub-cluster the photo that was selected for the header, recalculate the sub-cluster scores, and then assign each sub-cluster a probability that equals its score divided by the sum of scores of all sub-clusters. Those probabilities are then used to randomly select a sub-cluster. If sub-cluster $\mathcal{A}_i$ was selected, we remove its top-ranked photo, append it to $\mathcal{T}$ and repeat, until all photos have been selected.

By now we have described how to generate the cluster hierarchy and produce a ranking on the photos in that hierarchy, under the assumption that all nodes are associated with scores. We therefore proceed to describe a key aspect of the algorithm: the computation of the scores for a given cluster (node).

4.3 Scoring Clusters

The score of a cluster $\mathcal{A}_i$ depends on several factors, including the following:

1. The sum of relevance factors (see Section 3) of all photos in the cluster,

$$\rho_i = \sum_{P_j \in \mathcal{A}_i} r_j \ .$$

2. The tag-distinguishability of the cluster, τ_i (explained below).

3. The photographer-distinguishability of the cluster, ϕ_i (explained below).

4. The density of the cluster. More specifically, let $\sigma_{x,i}$ and $\sigma_{y,i}$ denote the standard deviation of the x and y coordinates, respectively, of all points in $\mathcal{A}_i$, and let

$$\sigma_i = \left((\sigma_{x,i})^2 + (\sigma_{y,i})^2\right)^{1/2} \ .$$

We define the cluster density as

$$\delta_i = 1/(1 + \sigma_i) \ . \tag{3}$$

5. The sum of image qualities (see Section 3) of all photos in the cluster,

$$\kappa_i = \sum_{P_j \in \mathcal{A}_i} q_j \ .$$

While most of the above factors are derived only from data that is contained in the photo collection, the relevance factor can introduce bias by subjective requirements. The relevance factor r_i of a photo P_i can incorporate parameters such as recency, the time of day, the time of the week, the identity of the photographer, etc. These parameters can be specified by a user making the query, or set by the system according to the application or the query context. Each photo is assigned a score $\theta(P_i)$ in the range $[0, 1]$ for each such parameter. The final relevance score, r_i, may be a weighted average of all those parameter scores.

The two interesting factors in the score computation are the tag- and photographer-distinguishability scores of clusters. These values are intended to represent how strongly a particular cluster is associated with specific tags or photographers.

4.3.1 Tag-distinguishability of clusters

Tag-distinguishability aims at detecting distinct or unique concepts that are represented in a given cluster, as those may indicate the presence of some interesting landmarks or objects in that cluster. For example, in Figure 2, the tag "bridge" appears in two photos from Cluster $\mathcal{C}_2$, and does not appear elsewhere. As a consequence, $\mathcal{C}_2$'s score improves. On the other hand, the tag "car" appears in photos from both $\mathcal{C}_2$ and $\mathcal{C}_3$ and therefore does not help to distinguish either of them.

Formally, each photo P_j, $1 \leq j \leq n$, is tagged with tags that are drawn from a finite dictionary, T. Hence, tagging may be viewed as a mapping $P_j \mapsto T(P_j) \subset T$. For all $t \in T$ and $1 \leq i \leq m$, let

$$\mathtt{tf}_{t,i} = \frac{|\{P_j \in \mathcal{A}_i : t \in T(P_j)\}|}{|\mathcal{A}_i|} \tag{4}$$

denote the relative frequency of the tag t in $\mathcal{A}_i$, (or *term frequency* as it is referred to in Information Retrieval). We often found that this measure biases towards tags that have been used frequently by one user in the same cluster. An alternative frequency calculation can be based on the fraction of photographers in this cluster that have used the tag t:

$$\mathtt{tf}_{t,i} = \frac{|\{u \in U_i : t \in T(P_j), P_j \in \mathcal{A}_i, P_j \in B_u\}|}{|U_i|} \tag{5}$$

where U_i is the set of users that have taken photos in cluster $\mathcal{A}_i$, and B_u is a set of photos taken by user u.

We also use the *inverse document frequency*, which is a measure of the overall frequency of the tag t in the entire photo collection,

$$\mathtt{idf}_t = \frac{n}{|\{P_j \in \mathcal{A} : t \in T(P_j)\}|} \ . \tag{6}$$

There are several ways to combine these two scores to measure how the term t distinguishes the cluster $\mathcal{A}_i$ from other

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**

fastcase®
Smarter legal research.