Email Communities of Interest

Lisa Johansen, Michael Rowell, Kevin Butler, and Patrick McDaniel
Systems and Internet Infrastructure Security Laboratory (SIIS)
The Pennsylvania State University, University Park, PA 16802
{johansen, butler, mcdaniel}@cse.psu.edu,rowell@math.psu.edu

ABSTRACT

Email has become an integral and sometimes overwhelming part of users' personal and professional lives. In this paper, we measure the flow and frequency of user email toward the identification of communities of interest (COI)–groups of users that have a common bond. If detectable, such associations will be useful in automating email management, e.g., topical classification, flagging important missives, and SPAM mitigation. An analysis of a large corpus of university email is used to drive the generation and validation of algorithms for automatically determining COIs. We examine the effect of the structure and transience of COIs with the algorithms and validate algorithms using user-labeled data. Our analysis shows that the proposed algorithms correctly identify email as being sent from the human-identified COI with high accuracy. The structure and characteristics of COIs are explored analytically and broader conclusions about email use are posited.

1. INTRODUCTION

Electronic mail has profoundly changed the nature of personal communication. It allows users to communicate with anyone, anywhere, at any time. It is easy to use, reliable, and fast. It is paradoxically asynchronous and immediate. Email is arguably the most influential and widely used application in existence. However, the technical community is only beginning to understand dynamics of its use.

In this paper, we measure the flow and frequency of user email toward the identification of communities of interest. A *community of interest* (COI) is a set of entities that share a common bond [5]. These sets can be of interest for various group studies. COIs have been studied in systems such as the telephone system and computer networks [1, 5, 13]. These studies can provide highly utilitarian results. For example, COIs can be used to identify normal communication in end-user hosts and servers. Such techniques were shown to effectively suppress worm behavior within a LAN when used to automatically generate host-level firewall rules [13].

This paper is preliminary work and is the first to apply COIs to the characterization of email. We construct COIs by measuring features of past email traffic. The volume, directionality, and frequency of the email traffic are used to determine association between members. We build algorithms based on these email traffic features to determine the members in a COI. The algorithms are analyzed and validated over a large corpus of university email (more than 3 million messages spanning 4 months) by assessing their ability to predict the priority of email as indicated by the recipients. In addition, we examine how the relationships between email users may indicate further information about their COIs through transitive connections.

Unlike other characterizations based on content or external information [4, 12, 15, 14, 20, 3], e.g., email subject, body, address books, the only inputs to our algorithms are the email volume and frequency. That is, the algorithms develop each COI based solely on the senders, recipients, and features of email traffic. This departure from traditional email analysis is significant. It hypothesizes that *email traffic flow alone is highly reflective of social activities, and that those activities can be accurately modeled using the features of the traffic*.

We further found that our COI detection algorithms could correctly identify email priority with greater than 90% accuracy. This supports our hypothesis: COIs based on to whom, when and how frequently a user sends and receives email are highly reflective of their social connections.

The evaluation of COIs in email is significant because COIs have an obvious application to the problem of automated email organization, where online services prioritize and categorize incoming email as it arrives–thus aiding in the increasingly intractable problem of dealing with huge bodies of incoming email. Applications to spam filters, virus detection, workflow management, HCI, sociology, and many others exist.

We begin by first examining possible applications of COIs in email. In Section 3, we study the underlying characteristics of a large set of email traffic. We then present measurement and analysis of these characteristics, namely volume and frequency. Section 4 defines algorithms to determine COIs based on the results of the analysis in Section 3. These algorithms are evaluated and validated in Section 5 and Section 6 concludes the work.

2. APPLICATIONS OF EMAIL COI

Communities of interest have shown to be highly applicable in both phone and data networks. Communities of interest in phone networks lead to the identification of account delinquents and criminal accomplices [5]. In data networks, COIs were used as part of a security mechanism by detecting anomalous behavior and automatically setting firewall rules [13]. Due to the benefits seen in these areas, we believe that COIs will be equally beneficial in an email environment. Here we examine possible applications of identified communities of interest within email.

2.1 Email Filtering

Due in part to the overwhelming bombardment of spam [2, 6], spam filtering has been at the forefront of email research. Current email filters [17, 18] have been developed based on widely distributed blacklists [19], whitelisting, and content analysis. Social network dynamics have also been beneficial in improving spam filters [4]. While social dynamic based-approaches have weaknesses as stand-alone spam filters, our technique of identifying COIs could provide social dynamic characteristics to assist an existing spam filter. We leave the implementation and testing of such a system to future work.

Generic email filtering includes the automatic organizing of incoming email based on some user-identified feature [14]. This filtering can use content information, address books, communication patterns, etc. to determine how to classify an email. Similar to the proposed use in spam filtering, information about COIs could also be used as input to these generic email filters. COIs provide an alternative mode of classifying activity to inference methodologies relying on naive Bayes classifiers, e.g., [8].

Automatic email prioritizing is one popular type of email filter. The ability to automatically sort email based on its priority level can significantly reduce the amount of time a user spends manually sorting through emails. We examine the application of COIs to priority-based email filtering in this paper. We believe that this application will best test the usefulness of COI within filters.

2.2 Guilt by Association

Identifying communities of interest within the telephone network proved highly applicable and beneficial in identifying fraudulent accounts [5]. Examination of the COI of a known fraudulent account could, with high probability, determine other fraudulent accounts as fraudulent users typically associate with each other. COIs in email could be used to determine similar behavior. For example, organizations could use COIs to identify accomplices in unauthorized behavior.

2.3 Malicious Email Identification

The application of COIs could also prove beneficial because of their ability to link users with relating email patterns. This can aid preventative and reactive measures such as those proposed by Stolfo et al. [20], and improve the forensic analysis of these events to identify the source of a virus and infected groups. We consider clustering methods for COI identification in a similar manner to the methodologies employed by epidemiological researchers investigating the transmission of infection vectors, e.g., [16].

2.4 Automatic Group Generation

Identifying email COIs can also assist in automatic email mailing list generation. Currently, email groups or lists are generated statically. Because COIs automatically identify users associations, they could be applied to automatically generate lists of users that may need to be on a given email distribution list. This automatic generation could greatly reduce the work of a system administrator or other list manager.

3. DATA EVALUATION

In order to define a community of interest for an email user, we need to know with whom they associate. In this analysis, we want to find what information email traffic reveals about the associativity of email users. Specifically, we want to understand what the volume, direction, and frequency of email reveal about the association of email users. We analyze one month of email traffic data in order to gain information about these email features. Latter sections use the results from this analysis to determine email COIs.

3.1 Source Data

The data set used in our analysis is from the Computer Science and Engineering Department at Penn State. This network consists of nearly 3000 email accounts with diverse usage habits. The majority of user accounts actively send and receive emails on a daily basis. Over the course of four months, the log files captured more than 3 million emails. The information in the server email files consists of a unique message ID, *to* and *from* email addresses, a timestamp, and host names and addresses. The data contains no information about the subject or contents of the email. In order to preserve privacy, every email and host address was anonymized through a one way keyed hash. The anonymized email log file data was then pre-processed such that a sender, receiver, and timestamp identified each unique email.

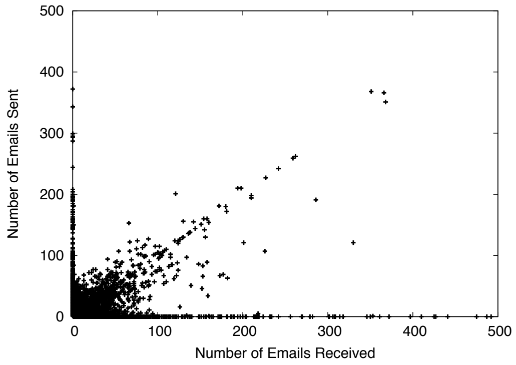


Figure 1: Email communications between users

3.1.1 Outliers

After an initial evaluation of our data set, we found that there was some email activity that was atypical for a normal email user. Figure 1 shows a graph of the number of emails sent versus the number of emails received for all communicating pairs of email users. This graph indicates that there are some email users that communicate excessively with one other email user. Our research is being conducted based on typical email users such that the application of COIs can be beneficial to them. Much of the email activity seen in Figure 1 is fundamentally different from a typical user's email activity. This kind of activity is due to system admin emails, automated tools using tripwire, etc. The utility of these emails is very different from that of typical user emails and we are not concerned with characterizing this type of email activity. Thus, we remove this outlying data in order to gain a more complete understanding of COIs based on typical email user activity. The removed outliers comprise less than 0.5% of our overall data set.

3.2 Email Volume

An analysis of our data reveals characteristics of email traffic that indicate attributes of an association among email users. The number of communications between two email users can be used to determine the existence of an association. For example, a large number of communications indicates an association whereas a fewer number of communications does not.

Our goal is to determine what volume of received emails indicates an association between a receiver and the sender. We determine this value through a partitioning of communication volumes: one partition includes values that indicate an association and the other partition includes volumes that do not. In order to create this partition, we perform k-means clustering on the number of emails

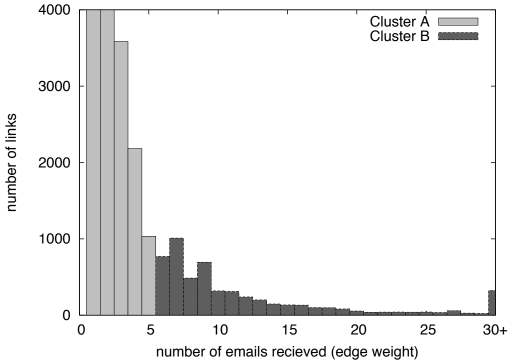


Figure 2: k-means Clusterings of Inbound weights

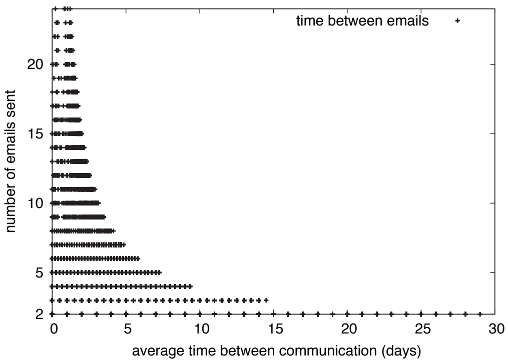


Figure 3: Email Volume and Frequency

sent between email users in our set of email traffic. This clustering groups communication volumes into k partitions such that the values in each partition have minimal variance from the other values within the partition. Because we are determining inclusion in a set, we use $k = 2$. The clustering is performed on inbound email volumes and separately on outbound email volumes. Our hypothesis is that the result from this clustering will indicate the partition in the data between associative and non associative communication volumes. This hypothesis is confirmed in Section 5.

We performed clustering on the inbound communication volumes. The clustering was performed on three separate months of data. The sum of distances was minimized within each partition, and each clustering was repeated 100 times. We found that, in each of our tests, the k-means clustering of the inbound links arrived at the same partition of the data. Thus, because our analysis resulted in the same partition in three sets of non-intersecting data, we conclude that this is a specific characteristic of these data sets.

The resulting partition of the inbound communication volumes is depicted in Figure 2. The x-axis labels are the weights, or number of emails from one user to another. The number of pairs of users that share a given communication volume within our email traffic data are shown on the y-axis. Here we see that cluster A includes communication volumes with inbound emails from 1 to 5, while cluster B includes those with inbound emails greater than or equal to 6. Our intuition indicates that cluster B is a partition of the inbound email volumes which indicate association between nodes. Thus, our clustering indicates that 6 inbound emails is enough to indicate an association between two email users over the course of a month.

Outbound email traffic differs from inbound traffic. The number of users to which another user sends email is most often far less than the number of users from which he receives email. This fewer number of data points causes difficulty in clustering. When performing clustering the outbound email volumes, the results never stabilized around one specific partition. Thus, our tools were unable to converge on a specific partition of the data. This led us to examine the characteristics of outbound traffic. If an email user sends an email, he obviously has some common bond or interest with the recipient in that communication. By definition, user association is indicated by communication between a sender and a receiver which indicates a shared bond. Thus, one outbound email is sufficient in indicating a relationship with a contact. This conclusion explains our inability to find clusters in outbound email traffic.

3.3 Email Frequency

We now investigate frequency as a determinant of an association. We measured the average interarrival time for received email for each email recipient from a given sender (connection). Due to the fact that 1 sent email indicates association, there is no need to evaluate the frequency of sent emails. This measurement was performed on one month of data[1].

Figure 3 shows the average interarrival time against the volume of email received by that user. In Section 3.2, we saw that 6 received emails within one month was the smallest number that would indicate association. In Figure 3, we see that the maximum frequency used for sending 6 emails is less than 6. This seems intuitive, as there are 30 days in a month.

To further study these interarrival times, we examined them separately; interarrival times of email volumes which indicate association ($\geq$ 6), and those which do not ($<$ 6). Figure 4 shows the graphs of the number of connections with a given interarrival time against the interarrival times. Comparing Figures 4(a) and 4(b), we see that Figure 4(b) clearly depicts the short interarrival times, or bursty nature, of email volumes which indicate association. Although this could be attributed to the large volume of the emails sent during a month, there is activity that cannot be explained by that alone. Thus, an association between two email users may be identified by email frequency. Because of this, we can look to frequency as an additional factor in determining association and thus, COI.

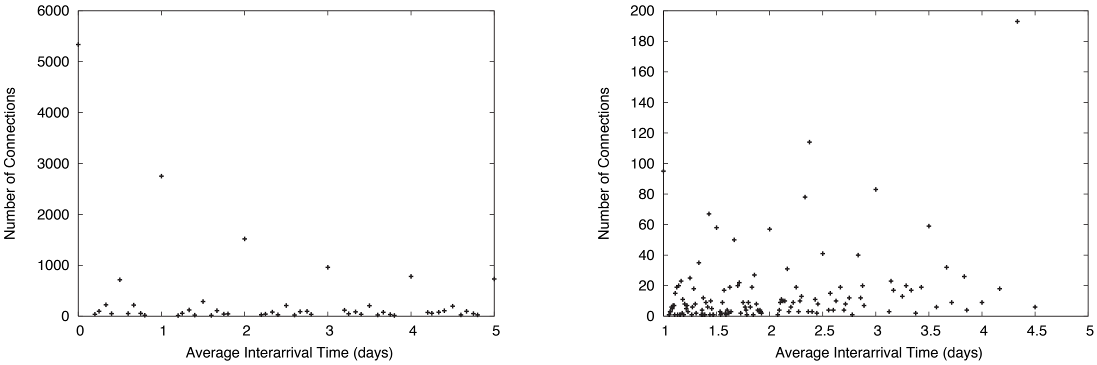
In summary, the measurement and analysis presented in this section indicates some fundamental characteristics of the associations as indicated by email traffic:

- Outbound email traffic is more indicative of an association than inbound traffic.
- Large email volume is indicative of an association.
- Frequent email is indicative of an association.

The measurements also indicate some initial results:

- 6 inbound emails indicate association
- 1 outbound email indicates association

[1] The experiment was repeated over the same three months of data on which volume clustering was performed. Each experiment resulted in nearly identical results.



(a) Interarrival Times, Volume < 6

(b) Interarrival Times, Volume ≥ 6

Figure 4: Interarrival Times for Association and Non-Association Volume

- An email received within 5 days of a previously received email from the same sender indicates association

The algorithms defined in the following section are based on the resulting characteristics.

4. COI DETECTION ALGORITHMS

In this section, we develop three algorithms of increasing complexity for determining COI: a *basic* algorithm based solely on communication volume and direction, a *frequency-based* algorithm that is sensitive to the frequency of email traffic, and a *decaying frequency-based* algorithm where COIs place priority on recent communications. The methods used to create the algorithms presented in this section are modeled after those in previous COI work [1, 5, 13].

The algorithms are each based on a value $C_{(a,b)}$ which captures characteristics of communication from one email user to another. This connection value would indicate to user a the "value of connection" with another user b. This value is adjusted based on email traffic flows. These adjustments are defined by the algorithms. If a connection value is above a certain threshold, τ, also defined within the algorithms, then user b is considered part of user a's COI.

4.1 Basic Algorithm

Our basic algorithm determines a COI solely on email volume. Each connection value, $C_{(a,b)}$, is adjusted every time there is email activity between user a and user b.

$$C_{(a,b)} = \begin{cases} C_{(a,b)} + 1 & \text{if email is received,} \\ C_{(a,b)} + \lambda & \text{if email is sent.} \end{cases} \tag{1}$$

We introduce the parameter λ as a way of weighting outbound and inbound emails. The value of λ is a ratio of the weight of outbound emails to inbound emails. If $\lambda > 1$, outbound emails are more indicative of COI membership whereas if $\lambda < 1$ inbound emails are more indicative of COI membership. Based on our analysis in Section 3.2 sent emails should increase the value more than received emails, thus $\lambda > 1$.

In our basic algorithm, the threshold for determining COI membership, τ, should be the same as the number of received email required to indicate membership. This value is based on Equation 1 where reception of email results in a constant value increase.

The basic algorithm assigns weights to the edges independently of time. Thus, if user a receives 6 emails from user b over the course of a year, the weight on that edge assigned by the basic algorithm would be the same if those messages were received over the course of one day.

4.2 Frequency-Based Algorithm

To address effects of the frequency of email communication, we introduce a new algorithm that is dependent upon interarrival times (times between received emails). This algorithm determines COI membership based on outbound and inbound emails in a manner similar to the basic algorithm.

In order to define our connection values as dependent on time, we must consider how time affects the weight of email messages that are sent and received. If we refer back to Section 3.3 we note a contact should be included in a COI an email is received within 5 days of the previous received email. This will serve as the basis for the initial construction of our new connection values.

First we discuss the effect of time on outbound emails. In Section 3.2 we found that one sent email was enough to determine COI membership. Thus, send frequency is not a factor when considering sent emails.

When considering received emails, frequency is a factor. Based on the analysis in Section 3.3, the shorter the interarrival times, the greater the increase of the connection value should be. If user a receives an email from user b more than once every 5 days, the contribution to its weight should be more than 1. Accordingly, if user a receives emails from user b less often than every 5 days, its contribution should be less than 1. This value is denoted by μ.

$$C_{(a,b)} = \begin{cases} C_{(a,b)} + \mu & \text{if email is received,} \\ C_{(a,b)} + \lambda & \text{if email is sent.} \end{cases} \tag{2}$$

Both the frequency-based algorithm and the basic algorithm weight the graph edges with a monotonically increasing function. This implies that once a contact enters a user's COI, he will never be removed from it.

4.3 Decaying Frequency-Based Algorithm

A monotonically increasing COI may not be realistic when evaluating a user's email communications. In order to address this issue of permanent COI membership, we introduce a modification to the frequency-based algorithm. We extend the frequency-based algorithm such that the connection values decay over periods of inactivity. Our decay function will decrease the current connection value at the turn of each day with the following restrictions: 1) if the connection value is less than $\tau - 1$, then it should not decrease, and 2) for a connection value larger than $\tau - 1$, it should decrease faster for larger values. The reason for our first restriction is so a user that was once a part of a COI will not be completely forgotten and can be quickly reintroduced into that COI. Our second restriction ensures that someone in a COI with a small connection value will not be ejected too quickly and those with a large connection value will not be forever in the COI. If $C_{(a,b)}$ is the current connection value, then at the increment of each day, we apply the following decay function

$$C_{(a,b)} = \begin{cases} C_{(a,b)} & \text{if } 0 \leq C_{(a,b)} \leq \tau - 1, \\ C_{(a,b)} - \frac{C_{(a,b)} - (\tau - 1)}{\delta} & \text{otherwise.} \end{cases} \quad (3)$$

We introduce δ as our decay coefficient, which can be varied depending on the purpose of the algorithm. It should be noted that the two parameters λ and δ will affect the speed at which a member of a COI will be removed. A larger value of δ would allow a user in a COI to stay longer without any communication, while a smaller value would eject the user after a shorter period of time. While our analysis from Section 3.2 implies that $\lambda \geq \tau$, it should be noted that an even larger value of λ will allow a user in a COI to stay longer without any communication. In Section 5, we examine the behavior of COIs under different values of δ.

4.4 Effect of Transitive COIs

The algorithms in the previous sections developed individual COIs in isolation from the knowledge of external COIs. We propose an extension to our algorithms such that COIs exploit transitivity. COI transitivity involves sharing COIs among associated nodes. For example, two users may not share a direct association; however, they may be indirectly linked through members of their individual COIs. We model transitivity by including COI ”neighborhoods”, e.g., transitivity including the COI members up to n hops away. We believe that this extra inclusion will result in more accurate identification of COI members due to the natural commonalities shared between users with associations. An evaluation of this extension is presented in Section 5.

5. ALGORITHM VALIDATION

An email user’s community of interest consists of members with whom the user shares a common bond. Based on our definition, the emails sent between an email user and his COI members should be of interest to the user. The amount of interest that a receiver has in an email is commonly indicated by priority. Thus, the emails received from the members of a user’s COI should be high priority emails. In order to validate the usefulness of our algorithms, we test their ability to correctly identify the senders of high priority email. We then present overall results from our email COI research.

5.1 Validation Data

Information about the priority of an email is determined solely by the receiver. Thus, in order to gain information about email priority, we performed a user study. Fifteen volunteers from our email network collected all of their received email for one month, which encompassed approximately 9,000 emails. At the end of the month, the volunteers labeled their data based on high and low priority. The volunteers were not told how the information was used. This training data was then anonymized with the same one way keyed hash as the original server log data. These prioritized messages were then integrated with the server log data by labeling the messages with their assigned priority level.

We performed the experiments by considering each email received as a sequential trace, allowing the COI algorithms to incrementally update communication values. Volunteers prioritized a subset of the three million emails; both training and test data were used to create communication values and test the correctness of classifications made. This trace-based method allowed testing data to be updated concurrently with other training data, presenting us with the opportunity to observe the evolution of the resulting models. Emails belonging to the training subset (i.e., those labeled with a priority) were tested for inclusion in the COI, then used to update communication values.

During the testing, we measured four values: true positives, true negatives, false positives, and false negatives. True positives are high priority emails where the algorithm recognized the sender of the email as being included in the receiver’s COI. True negatives are low priority emails that the algorithm did not recognize the sender of the email as being included in the receiver’s COI. False positives occurred when the sender of low priority email is incorrectly recognized as a member of the receiver’s COI, and false negatives occur when the sender of high priority email is incorrectly recognized as not being in receiver’s COI.

We are most concerned about the ability of our algorithms to correctly identify the senders of high priority email thus, false negatives are the worst kind of false classification. We evaluate two statistics to highlight these results: correct identification of high priority email senders (HIGH) and correct overall identification of both high and low priority senders (OVERALL).

5.2 Results

Our testing revealed that the COI algorithms were capable of successfully determining the priority of an email by its sender for over 90% of both high priority email (HIGH) and the full corpus of prioritized email (OVERALL). Table 1 shows the percentage of email correctly classified by our basic and frequency-based algorithms. The numbers in bold represent the percentage when the parameters used are those obtained from our analysis in Section 3: $\lambda = 6$ and $\tau = 6$.

Table 1 shows that our algorithms yield one of the highest percentages of correct overall and high priority classifications when $\lambda = 6$. This implies that if a receives 6 emails from b in a month, then b belongs in a’s COI. It should also be noted that for larger values of λ our algorithm does not classify as well, implying that both sent and received emails contribute to association.

We also see that $\lambda = \tau$ yields one of the highest percentages of correct classifications. This would imply that a single email from user a to user b represents an inclusion in a COI. This observation suggests that our intuition of the association strength of sent mail was correct. Sent emails are a stronger indication of an association than received emails.

Figure 5 compares the basic and frequency-based algorithms for a given user. The y-axis represents the connection value of a given user a to another user. We see that our frequency-based algorithm introduces the user into a’s COI faster than our basic algorithm. The results from the validation of frequency-based algorithm are shown in Tables 1(c) and 1(d). This suggests that introducing an algorithm dependent on time improves the classification of email, validating our claims that time is an important criterion for evaluating email communication. Because the frequency-based results improve over the results from the basic algorithm, we can conclude

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**