Modeling User Concerns in the App Store: A Case Study on the Rise and Fall of Yik Yak

Grant Williams* and Anas Mahmoud†
The Division of Computer Science and Engineering, Louisiana State University
Baton Rouge, LA, 70803
*gwill83@lsu.edu,†amahmo4@lsu.edu

***Abstract*—Mobile application (app) stores have lowered the barriers to app market entry, leading to an accelerated and unprecedented pace of mobile software production. To survive in such a highly competitive and vibrant market, release engineering decisions should be driven by a systematic analysis of the complex interplay between the user, system, and market components of the mobile app ecosystem. To demonstrate the feasibility and value of such analysis, in this paper, we present a case study on the rise and fall of Yik Yak, one of the most popular social networking apps at its peak. In particular, we identify and analyze the design decisions that led to the downfall of Yik Yak and track rival apps' attempts to take advantage of this failure. We further perform a systematic in-depth analysis to identify the main user concerns in the domain of anonymous social networking apps and model their relations to the core features of the domain. Such a model can be utilized by app developers to devise sustainable release engineering strategies that can address urgent user concerns and maintain market viability.**

I. Introduction

The rapid evolution of mobile phone technology in the past decade has led to a paradigm shift in the way software is being produced and consumed [1], [2]. As this technology is becoming more accessible, more consumers are migrating to their phones and tablets as their main computing devices [2], [3]. In response to this growth, mobile application stores (e.g., Google Play and the Apple App Store) have emerged as a new model of online distribution platforms. These stores have expanded in size in the past five years to host millions of apps, offering end-users of mobile software virtually unlimited options to choose from. For instance, as of March 2017, the Apple App Store alone has reported around 2.25 million active apps, growing by over 1000 apps per day[1].

After they are published, apps enter a long phase of feature optimization in order to maintain market viability [4], [5], [6], [7], [8], [9]. From an evolutionary point of view, this process is equivalent to acquiring traits that can lower the chances of elimination by natural selection [10]. This *survival-of-the-fittest* effect can be observed in the app store. Specifically, similar to business firms competing in the market, apps are competing actors in an ecosystem of finite resources—only a handful of apps dominate downloads and revenue, leaving only fractions of market value for other apps to compete over [11], [12], [2], [9].

[1]https://www.statista.com/topics/1729/app-stores/

With such an unprecedented level of competition, it is becoming increasingly harder for apps to stand out. In fact, recent app growth statistics have shown that the majority of apps lose around 80% of their users in the first 90 days of their release [13]. These observations have forced software providers to seek and explore more effective strategies for user acquisition and retention. An underlying tenet is that user engagement in the software production process can play a major role in gaining a competitive advantage, and ultimately, surviving in the market [14]. Recent work in this domain has been focused on mining user reviews available on mobile app stores for technical feedback [15], [16], [17], [18]. Analysis of large datasets of app store reviews has revealed that almost one third of these reviews contain useful information (e.g., bug reports and feature requests) [19], [20], [16]. Social media platforms, such as Twitter, have also been exploited as a more open, widespread, and instant source of technical user feedback. Recent research has shown that almost 50% of software-relevant tweets contain useful information for developers [21], [22], [23], [24]. However, despite this effort, there is still a research gap on how such information can be systematically utilized to enhance app survivability and competitiveness.

In an attempt to bridge this gap, in this paper, we present a case study on the rise and fall of the social networking app Yik Yak. Between 2013 and 2016, Yik Yak managed to grab a massive share of the social networking market by offering a unique combination of features that attracted users' attention. However, following a series of feature updates, in mid 2017, the development team of Yik Yak announced that the app would be shutting down due to a massive loss in their user base. The story of Yik Yak presents a unique opportunity for analyzing one of the most recent major failures in the app market. *Learning from failure* has long been used in market research as a means to build organizational knowledge and prevent future failures [25], [26].

To conduct our analysis, we collect and synthesize user feedback available on app stores, social media, and several news outlets to track the story of Yik Yak. We examine its most successful features, the design decisions that led to its decay, and eventually, death. We further track user migration patterns to other competing apps and analyze the main concerns of users of these apps. The main objective of our analysis is to demonstrate the feasibility of systematically

generating unified domain models that can integrate multiple heterogeneous sources of user, market, and system information to reflect an accurate picture of the current state of the domain and its most desirable features.

The remainder of this paper is organized as follows. In Section II, we track the history of Yik Yak and identify its main rival apps. In Section III, we perform an in-depth feature-oriented analysis of the domain of anonymous social networking apps. Section IV discusses the main findings of our analysis and their potential practical significance. Section V reviews seminal related work. Finally, in Section VI, we conclude the paper and outline prospects of future work.

II. Yik Yak's History and Market Analysis

In this section, we track the history of Yik Yak, including its main features, most notable updates, decline, and eventually death. We further perform market analysis to identify the set of rival apps that former Yik Yak users have migrated to.

A. The Rise and Fall of Yik Yak

Yik Yak was a location-based social networking app that allowed users to post and vote on short messages (known as yaks). Yik Yak distinguished itself from its competitors (e.g., Twitter, Facebook, and Snapchat) by two main features: anonymous communication and geographical locality. In particular, users could only post and engage in discussions within a 1.5 mile radius around their location. This feature ensured that most of the posts were related to activities in a local area (e.g., a college campus or a small town). Yaks were sorted primarily by distance, so that posts about particular buildings or activities on a campus would find the most relevant local audience. In addition, users had the option to set a home community, called a *herd*. Herds allowed a user to communicate with friends even if the user was temporarily outside of his/her original geographical area.

One of Yik Yak's most notable features was anonymity: users did not need to register a name in order to participate in any of the app's features, including creating threads, posting replies, and voting on messages. This feature was widely praised, with news outlets such as Wired stating,

> *From my interactions on Yik Yak, I can tell how well the buses are running on a given day, the least crowded times at the gym, and student reactions to administrative decisions. Instead of being worried about the negative aspects of anonymity on Yik Yak, we should embrace the service's ability to help our young people grow and develop a strong sense of community. [27]*

The combination of anonymity with locality proved successful; at the end of 2014, Yik Yak had become one of the most popular social networking platforms on college campuses. In fact, due to this massive success, in 2015, the company was valued at close to $400 million. Business insider writes,

> *Goetz ended up leading a $62 million Series B round in Yik Yak at a valuation that approaches $400 million. Goetz admits the app's current scale may not seem to justify $400 million, but he says its engagement was enthralling. [28].*

The anonymity provided by Yik Yak had a downside, however. Due to the lack of personal identifiability, Yik Yak posts were an easy vector for cyber-bullies to anonymously harass other users at their schools or universities. This problem became significant when the app was used to make threats credible enough for institutions to request police assistance. The New York Times reported,

> *Local police traced the source of a yak — 'I'm gonna [gun emoji] the school at 12:15 p.m. today' — to a dorm room at Michigan State University. The author, Matthew Mullen, a freshman, was arrested within two hours and pleaded guilty to making a false report or terrorist threat. [29]*

In addition to threats, Yik Yak became a platform for spreading toxic behavior in high schools, causing a *change.org* petition for shutting down the app to receive more than 78,000 signatures. In an attempt to minimize the frequency of such incidents, the app's founders implemented *geo-fences*, which detected when the app was being used from within a high school and prevented access. About this measure, Business Insider writes,

> *they proactively blocked about 100,000 middle schools and high schools from accessing the app to keep under-agers off Yik Yak. They also implemented community policing tools and keyword targeting for hate speech [30].*

While Geo-fences significantly restricted negative behavior in middle and high schools, other measures were still necessary to prevent toxicity elsewhere. Following this effort, Yik Yak began to roll out mandatory *handles*, where users would be forced to choose a screen name. At first, the option to hide these handles allowed users to continue posting completely anonymously, but with the release of version 4.0.0, even this ability was removed. The response from the community was swift and overwhelmingly negative, with tweets such as *"Rip yikyak- was only good when it was anonymous"* becoming more common on the app's Twitter feed. App store reviews were similarly hostile, with reviews such as *"Anonymity is the sole reason for which I downloaded this app in the first place. Handles just ruins it!"* and *"I see no point of using this app if its going to be no longer anonymous!"* began to dominate the reviews.

This series of feature updates, along with other updates (e.g., removing the herd feature), had catastrophic consequences on Yik Yak's market performance. The number of downloads (as estimated by prioridata[2]) sharply dropped as shown in Fig. 1. Yik Yak's attempt to reverse course by reintroducing anonymous posting, was never successful in recapturing its previous popularity. This massive loss of users, besides other miss-calculated business decisions, led the company to officially suspend operations in May of 2017. Eventually, the company was acquired for a fraction of its original valuation. On that matter, USA Today writes,

> *The Atlanta-based company had been for sale, but apparently had few takers. It was able to move some of its engineers to Square, the online-payments app, for what Bloomberg says was $3 million. [31]*

[2]https://prioridata.com/apps/yik-yak-com.yik.yak/performance

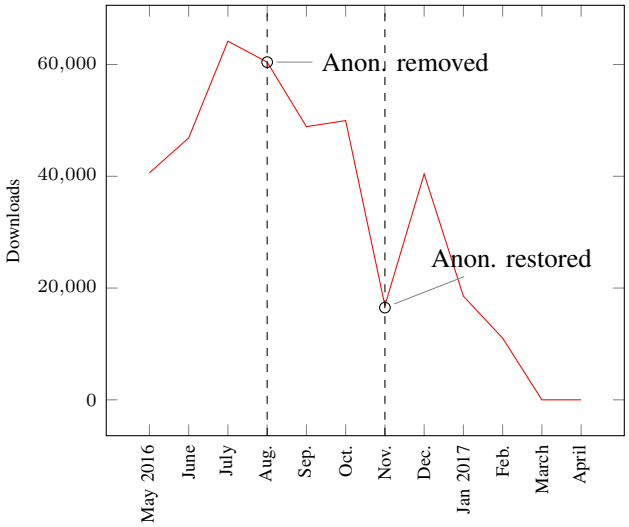

Fig. 1: The number of monthly downloads after anonymity (Anon.) was removed and then restored in Yik Yak.

B. User Migration

Yik Yak offered a unique form of anonymous, locally-focused, and democratic communication. After the app was suspended, several competitor apps emerged to take advantage of the demand for a replacement. Numerous threads on social media appeared with former users discussing potential alternatives.

In order to understand Yik Yak users' migration behavior, we identified the set of popular apps that attempted to imitate Yik Yak's feature set. We looked especially for apps which users on Reddit, Twitter, and app stores explicitly stated that they were switching to as a result of Yik Yak's suspension. Gathering and synthesizing knowledge from mainly user-driven sources can help to narrow down the domain to its most fit elements from a user perspective, thus saving the unnecessary effort of analyzing hundreds of apps that, even though share the same core functionality, failed to survive or grab any user attention (i.e., eliminated by market selection).

Recent analysis has revealed that users typically discuss competing apps in their app store reviews [32]. Such information can be leveraged to identify other apps that can be potentially classified under the same domain. For example, after Yik Yak's failure to control for cyber-bullying, names of rival apps that implement a better content moderation started appearing in reviews of the type *"I'm moving to [Spout], its less infested with trolls and racists and it has a more mature and interesting user base."* and *"I will be switching to [Swiflie] after Yik Yak shutdown."*. Furthermore, people often resort to social media (e.g., Twitter) to express concerns about their apps, such as discussing alternative apps and providing recommendations to other users about potential competing apps to migrate to. For instance, after Yik Yak's suspension, tweets of the nature *"Any alternatives to #YikYak?"* and *"@YikYak we cant lose our community where should we go now"* started appearing on Twitter. Other social media outlets, such as the social media aggregator Reddit, also commonly include discussion threads about app alternatives. For example, one user posted a Reddit thread titled *"Alternative to Yik Yak?"*[3] trying to convince other users to try the now defunct competitor app *Candid*. Part of the post reads,

> *As for Candid it's much better IMHO. You can share links, not have to be on location at a college to post in the college's feed, you can share post links and edit posts and comments. Most importantly it's 100% anonymous. So try it out if you're interested!.*

Comments on that post also expressed support for other alternative apps, such as Swiflie, Jodel, and Spout. Similar recommendations were also made in several other online news outlets, such as the news articles *"Five Best Yik Yak Alternatives You Should Check Out"* and *"Stay Anonymous Online With the 5 Hottest Yik Yak Alternatives"*. Names of competitor apps have also appeared on *Alternative To*[4], a crowd-sourced website for submitting and discovering alternatives to software applications.

Our effort to identify Yik Yak's competing apps included manually analyzing Yik Yak related tweets (any tweets including *#YikYak* or *@YikYak*) and reviews that appeared in May and June of 2017 along with any Reddit or news posts that discussed the dying app. Specifically, we kept track of any names of potential competing apps that users expressed interest in. Furthermore, we only considered apps with an average of 1,000 downloads over the months of May and June of 2017. This number is essentially a lower bound. More specifically, it is difficult to imagine a social network with fewer than 1,000 monthly downloads achieving coverage over major universities and metropolitan zones. Table I summarizes the number of global downloads of popular alternative apps around the time of Yik Yak's death. These numbers were obtained from Prioridata[5], a web service which provides app download estimates based on the freely-available global download rankings from both app stores. While other factors, such as user retention and usage are also necessary to accurately quantify popularity [8], the number of downloads alone can be a valid proxy, or at least an indication, of app popularity [2]. In what follows, we describe the main alternative apps identified in our analysis along with their most popular features:

- **Kik**[6]**:** Kik is a social media app which actually predates Yik Yak by four years, with the first tracked version being released in 2010. Unlike Yik Yak, Kik has never had a locality requirement: messages can be viewed by any member of a group no matter how far away they are. Additionally, it has never allowed truly anonymous communication, meaning that there was never a user outcry against such a feature being removed. Based on download numbers alone, Kik is clearly the most popular alternative with over one million downloads per month.

[3]https://www.reddit.com/r/yikyak/comments/4y5lpl/alternative_to_yik_yak/
[4]http://alternativeto.net/software/yik-yak/
[5]https://prioridata.com/
[6]https://www.kik.com/

App	May	June	Change
Kik	1,672,000	1,695,200	1.4%
Jodel	262,300	200,400	-23.6%
Firechat	41,100	42,700	3.8%
Whisper	142,800	148,400	3.9%
Swiflie	26,468	10,100	-61.8%
Spout	3390	515	-84.8%

TABLE I: Estimated combined number of global downloads (Google Play and the Apple App Store) during May and June of 2017 for each of Yik Yak's rival apps.

However, this popularity has a downside. Whereas Yik Yak struggled to cope with cyber-bullying, Kik has been tainted by numerous, serious incidences of being used for kidnapping, child exploitation, and murder. The New York Times writes about one such incident,

> *The death of Nicole Madison Lovell, a liver transplant and cancer survivor from Blacksburg, Va., has put Kik in the spotlight. Neighbors say that the day before she died, Nicole showed them Kik messages she had exchanged with an 18-year-old man she was to meet that night. [33]*

- **Jodel**[7]**:** Launched at almost the same time as Yik Yak, Jodel primarily targeted a European userbase. After Yik Yak's poorly-received 4.0.0 release, users from English speaking countries began migrating to Jodel in large numbers. Jodel's developers noticed this trend and reached out to Reddit's Yik Yak community for suggestions to improve the experience for former *Yakkers*. Among the most popular suggestions were access to herds (now called *hometowns* in Jodel), better moderation for anti-bullying and threat purposes, and improved notifications. All three suggestions were implemented to varying degrees, enabling Jodel to emerge as a popular alternative to Yik Yak.
- **Firechat**[8]**:** Firechat is an unconventional Yik Yak alternative that utilizes mesh networking to allow local, anonymous discussions. Unlike other social networking apps, Firechat connects users in-range directly to one another's phones rather than sending information over the Internet. Users' posts propagate by being encrypted and re-sent across nearby devices running Firechat until no more users can be reached. Posts can be made to anyone in the local area, or pre-filtered into focused chat groups. Firechat's direct messaging system makes it usable when cell or data connections are unreliable or unavailable. Business Insider writes,

  > *FireChat became hugely popular in Iraq after the country faced restrictions on internet use, and it was an integral part of the 2014 Hong Kong protests and the 2015 Bersih anti-corruption movement in Malaysia. [34].*

- **Whisper**[9]**:** Whisper is a unique app which allows users to post anonymous thoughts and confessions they may be reluctant to share with their identity attached. Unlike its competitors, Whisper is not designed around comment threads, where users reply to each other in sequence. Instead, posts can have many replies in parallel, and each reply can have its own set of replies, often forming a vast tree containing hundreds of posts that spiral into new discussions. This lack of direct discussion threads makes intimate conversations unwieldy, nonetheless, it facilitates the sharing of a wide variety of opinions on a single topic. According to Wired magazine, this organization of discussion may reduce toxic behavior,

  > *Whisper aspires to be a social network that doesn't rely on a social graph. The app has features for finding people nearby or forming groups, but the feed defaults to showing users popular posts from all over the world. The result is far less deeply offensive material [35]*

  Besides a unique method of conversation, Whisper also offers features designed to encourage local communication. Users' locations are tracked, and the posts in a user's area can trigger notifications indicating that a local discussion is taking place.
- **Swiflie**[10]**:** Swiflie is a recent social networking app that was created specifically to provide an alternative to Yik Yak after the latter lost the ability to post anonymously. Swiflie adopted the tagline *"Be yourself, or not"* to market this fact. Optional profiles are also allowed, along with the ability to post to user groups, which replaces Yik Yak's herd functionality. In a major departure from Yik Yak, users are allowed to view and post on chat groups outside of their local area. Such groups exist for many large universities, metropolitan areas, and even social groups. Swiflie is specifically designed to encourage timely posts, and every post is eventually removed. Posts normally last for about one day, but this lifespan can be extended if a post receives enough *likes* from fellow users.
- **Spout**[11]**:** Spout was released in October 2016, shortly after the 4.0.0 release of Yik Yak. Spout offers the closest experience of Yik Yak among the various alternatives, with an almost exact duplication of the feature set. Despite this, there are a few differences, including the ability to see posts from various user-selectable distances and significantly more active moderation. The moderation is praised by users looking for friendlier and less toxic communities to get involved with. For instance, one review states,

  > *Great app, and a worthy alternative to Yik Yak. Like Yik yak, posts can be up-voted and downvoted, and you can also post images or GIFs in replies . . . Spout is less infested with trolls and racists than rival apps such as Candid and Swiflie, and it has a more mature and interesting user base as a result.*

[7]https://jodel-app.com/
[8]https://www.opengarden.com/firechat.html
[9]http://whisper.sh/
[10]https://www.swiflie.com/
[11]https://www.spout-app.com/

However, many more users appear to regard the moderation as heavy-handed, with a large number of one-star reviews of users complaining about being banned. For example, one review states,

> *This app is very strict about banning people, if anyone doesn't like your post they have the option to report you which will accordingly ban you from using this app forever. I used this app for about a week and got banned quickly . . . The app has cool features that you can not find in (RIP) yikyak, but the banning is really bad.*

III. Domain Analysis

In this section, we systematically analyze and model the main user concerns in the domain of Yik Yak and its competing apps along with their relations to the core features of the domain. In software engineering, feature-oriented analysis is often applied to describe the main reusable assets (e.g., requirements, design, code) of Software Product Lines (SPLs). An SPL can be described as a set of software-intensive systems sharing a common, managed set of features that satisfy the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way [36]. While reuse is not the main focus of this paper, SPL analysis can be a powerful tool to obtain an in-depth look into the main features and user goals of our domain. Specifically, our analysis in this section can be described as a three-step procedure:

1) Extracting the core features of the domain
2) Extracting the main user concerns of the domain
3) Modeling the relations between user concerns and the domain features

In what follows, we describe these steps in greater detail.

A. Extracting Domain Features

We start our analysis by creating a feature matrix based on the observed behavior of each of Yik Yak's rivals identified earlier. A feature matrix describes the main features of a group of systems in a certain domain. A feature can be defined as any observable functional behavior of the system. Each row of the matrix represents a feature vector, showing the features that each system in the domain supports [37], [12]. To create such a matrix, we downloaded and exercised the features of all apps in our domain. In particular, each author downloaded all of the apps on their personal smart devices, created an account for each app (if necessary), and then used each app multiple times over the period of two weeks to get a sense of the most dominant features they provide. Furthermore, the textual description available on the app store page of each app was used to capture the main features the developers of each app emphasize. A discussion session was then held to map our observations into a domain-specific feature matrix. Our analysis resulted in the feature matrix in Table II. The following is a detailed description of these features.

- **Local posting (Loc.)**: apps supporting this feature use the location of posters to sort and filter posts for display, or alternatively, detect appropriate groups for users to join based on their location.
- **Encryption (Enc.)**: apps supporting this feature use some sort of data encryption to prevent outside actors from determining the content of an intercepted message. While encryption can give users a feeling of security in their anonymity, it can also prevent police from following up on crimes or threats of self-harm.
- **Profiles (Prof.)**: apps supporting this feature implement special pages for users to describe themselves. Profiles reveal when posts have a common author, which might hurt anonymity as active users can be identified based on their posts over time.
- **Away posting (Away.)**: refers to the ability of users to post outside their local area.
- **Groups (Grps.)**: are chat rooms that allow users to post messages only to people interested in a particular topic, thus allowing more focused discussions.
- **Moderation (Mod.)**: is considered present if there are easily accessible mechanisms to report toxic behavior. Given the consistent cyber bullying and harassment that Yik Yak had suffered from, it is unsurprising that most of the apps in our domain have strong moderation.
- **Sign-ups (Sign.)**: refers to whether an app has a mandatory sign up before users can access its features. Given the difficulty of balancing anonymity and convenience with the need to provide effective moderation, it is unsurprising that there were several apps erring in each direction, either requiring users to sign up or not.
- **Advertising (Ads.)**: is a common monetization strategy for free apps. The specific implementation varies among apps. For example, Whisper shows ads as regular posts, while Kik allows users to chat with a virtual representative of the advertising entity.
- **Anonymity (Anon.)**: is a feature common to all apps in our domain. Apps with a mandatory sign-in feature associate posts with a persistent username, but none requires that these assumed names to be associated with real-world identities.
- **Blocking (Blk.)**: enables users to manually hide unwanted posts and disallow direct messages made by other users. Blocking essentially allows users to act as their own moderators.
- **Notifications (Not.)**: allows social networking applications to inform users of new posts in their area or any new comments on their posts. All apps in our domain send notifications, but to different extents. For example, Kik and Spout use notifications to inform users of replies to their threads, while Firechat uses notifications to push ads on users.

B. User Concerns

Under this phase of our analysis, we are interested in users' concerns. A user concern can be defined as any direct or indirect, desirable or undesirable, behavior of the app that might result as a side-effect of deploying the app, or any of its

App	Loc.	Enc.	Prof.	Away	Grps.	Mod.	Sign	Ads	Anon.	Blk.	Not.
Kik			X	X	X	X	X		X	X	X
Jodel	X	X	X	X		X			X		X
Firechat	X	X	X	X	X		X	X	X	X	X
Whisper	X	X		X	X	X		X	X	X	X
Swiflie	X	X	X	X	X	X	X	X	X	X	X
Spout	X	X		X		X			X		X

TABLE II: A feature matrix extracted from the domain of Yik Yak and its rival apps.

features, in its environment. For instance, users might perceive a video game as being *entertaining* or *boring*, or a social networking app as being *toxic* or *engaging*. The objective of our analysis is to extract and synthesize the main concerns of users in the domain of Yik Yak and its competitors.

We start our data collection process by extracting the most recent (i.e., from the date of data collection and going backward) reviews and tweets for each app. We only included posts that actually contained a legitimate user concern. Posts with no useful information were discarded (spam, praise, insults, etc.). Posts were manually classified in order of most to least recent until obtaining a total of 50 posts expressing actual user concerns from each type of post (50 most recent concerns from the App Store reviews, 50 from Google Play reviews, and 50 concerns from Twitter). This process was repeated for all apps in our domain. For Spout and Swiflie, we ended up with only 91 and 122 concerns in total before we ran out of reviews and tweets for these two apps. Each relevant post was examined by the authors to determine which concern was most appropriate for classification. To control for complexity, if multiple goals applied, the one mentioned more, or which seemed to be the primary concern of the user, was picked.

Our qualitative analysis shows that a total of seven concerns were present in the data[12]. Examples of posts describing each of these concerns are shown in Table III. Fig. 2 shows the total post count for each concern. In general, the set of concerns raised by the users in the domain of anonymous social networking apps can be described as follows:

- **Expressiveness:** refers to the ability of users to freely express themselves on the app. Many users react with hostility in their reviews and tweets to perceived breaches of their free speech, especially when it results in a ban.
- **Entertainment:** refers to how enjoyable the experience of using the app actually is. Entertainment has been identified as an important user requirement for social networking apps, often associated with the ability to communicate with a broad range of people [38], [39].
- **Avoiding toxicity:** is the desire for a friendly environment for discussion. Toxicity is unavoidable on the internet, but effective moderation and reporting mechanisms can mitigate the extent to which users are exposed to such undesirable behavior.
- **Anonymity:** refers to users' ability to hide their identity when posting. Anonymity is highly desirable in our

[12]data is available at http://seel.cse.lsu.edu/data/re18.zip

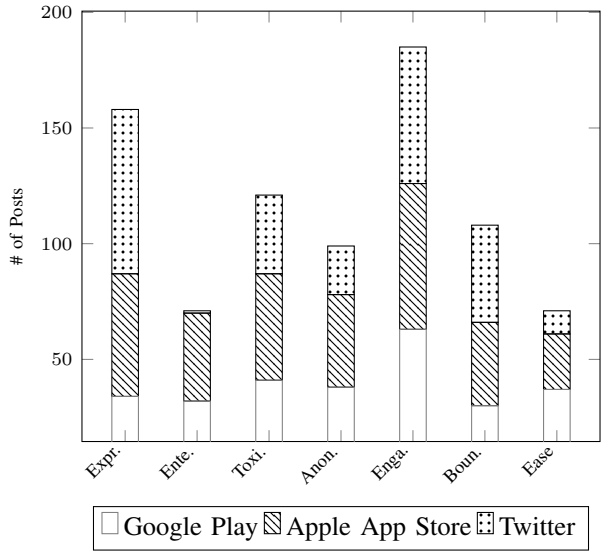

Fig. 2: The number of posts (reviews and tweets) per category of user concerns.

Soft Goal	Example Post
Expressiveness	*"I love the concept of expression through this app [Whisper]. Overall try it! Beware you might get hooked!"*
Entertainment	*"If you feel bored or just want to laugh, give it [Jodel] a try. Highly entertaining."*
Toxicity	*"The user base [on Spout] is extremely positive and supportive of each other, negative/bullying posts are downvoted quickly"*
Anonymity	*"I like the idea [of Jodel] and I think that the anonymous character is a lot [of] fun because people are discussing things they wouldn't do if the world knows who's behind the numbers"*
Engagement	*"After Tapt and Yak went down there was no where to go. Then Swiflie came along and brought us all back together; and in a whole new and better way."*
Maintaining boundaries	*"I like the concept of this app [Firechat]. The thing I don't like is that ANYONE can text you ANYONE any strangers can text you without your permission they should make it wear[sic] you have to request to message someone."*
Ease of Use	*"I hate that the video chat button is so easy to accidentally click [in Kik]. I've clicked it by mistake so many times during conversations."*

TABLE III: Example posts of the different user concerns.

domain as it grants users the freedom to express their unpopular opinions. On the dark side, being anonymous makes it easier for users to behave in a toxic manner without jeopardizing friendships or social status.

- **Engagement:** refers to the ability of users to use the app in a way that affects their day-to-day life, especially regarding local posts, groups, and friends. In fact, making new contacts and engaging with existing friends have been shown to be the most important reason for people to use social networking apps [38].
- **Maintaining boundaries:** refers to the ability of users to control who talks to them and what posts are visible in their feeds. The presence of spam, excessive notifications, and annoying ads indicates the inability of users to set boundaries.
- **Ease of use:** is a common goal among almost all application domains. However, it meets our criteria of inclusion due to its particular importance in social networking apps [40]. Ease of use refers to the overall ability of users to access the app's features without frustration.

C. Modeling

In this section, we generate a unified user-driven feature model for the domain of Yik Yak and its competing apps (anonymous social networking apps). Several domain modeling techniques have been presented in the literature. One of the earliest frameworks proposed in the software engineering literature for modeling domain knowledge is Feature Oriented Domain Analysis (FODA) [41]. A feature model (FM) in FODA is represented as a hierarchical tree graph, showing the mandatory, alternative, and optional features of the domain along with their commonalities and variabilities [42]. While FMs can be very effective for representing the functional features of the domain, they often ignore the non-functional requirements (NFRs). NFRs describe a set of quality attributes that a software system should exhibit, such as its usability, security, and reliability [43]. To represent such attributes, another type of models, known as the Softgoal Interdependency Graph (SIG), is used [44]. In SIG diagrams, softgoals are the basic units for representing NFRs. The interdependency relations in the graph are accompanied with plus and minus signs to indicate the type (trade-offs *vs.* synergy) and degree of impact between softgoals.

In our analysis, we adapt a hybrid domain modeling technique, known as Feature-Goal analysis (F-SIG), to describe the main goals and features of our domain of interest. F-SIGs enable a comprehensive qualitative reasoning about the complex interplay between the functional and non-functional features of a domain, allowing developers to record design rationale and evaluate different design choices early in the project [45]. In F-SIGs, a functional feature is represented using a rectangle shape, a softgoal is represented using a cloud shape. The edges of the graph represent the interrelationships among the softgoals and features of the domain. These relations are typically accompanied with plus and minus signs to indicate the type and degree of impact among softgoals and features (`--Breaks`, `-Hurts`, `?Unknown`, `+Helps`, `++Makes`). The F-SIG we derive to represent our domain is shown in Fig. 3.

The make/break associations expressed in the diagram are determined using the following coding scheme: posts (tweets and reviews) containing a specific user concern are grouped together. Each post in the group is then examined to determine which functional feature the user is concerned about (linking to the concern). These features are extracted and added to the graph. To determine the interrelationship (positive or negative) between a specific feature and the concern, we manually examine posts relating that feature to the concern. If more than 67% (two thirds) of these posts are leaning toward a certain sentiment polarity (+ *vs.* -), the relation is assigned to that polarity (`+Helps` *vs.* `-Hurts`). Otherwise, the relation is marked as `?unknown`. We then use the proportion of posts related to that specific feature to determine the relation intensity. In particular, assuming users have linked their concern to a number of *N* domain features, if the feature appears in more than $1/N$ of the posts, we mark that feature as having out-sized importance using `++Makes` and `--Breaks` instead of just `+Helps` and `-Hurts`.

For example, assuming 120 posts were found to be related to the concern *anonymity*. In their posts, users linked this concern to the features *sign-in*, *profiles*, and *encryption*. Assuming the majority of users perceive the *profiles* feature to hurt their anonymity (more than 67% of these posts related to having mandatory profiles are negative), we mark this relation as `-Hurts`. Assuming out of the 120 posts, more than $1/3 = 33\%$ are about profiles, we then upgrade the relation to `--Breaks`, thus indicating that this feature might have more severe negative impact on users' experience.

D. Model Interpenetration

Our coding procedure shows that *engagement* and *expressiveness* account for noticeably more user opinions. The most common form of *engagement* related posts expressed a user's desire to connect with lots of people. Thus, suggesting that a major factor for users to use a certain app is simply the ability to find friends on that app. Posts related to *Expressiveness* were mostly common on Twitter, primarily because many users were seeking to overturn a ban, often protesting that the terms of service were overly harsh on free speech. Twitter appears to be more popular than app store reviews for discussing bans, possibly because it is a low-latency method of communication with app developers. Many users also described problems with moderation, causing them to feel uncomfortable about *expressing* their opinions. For example,

> *with the creators or mods controlling everything, I found that they ban people if they disagree with their opinion despite not being hateful and not for actual hate or racism.*

Our analysis has also revealed that *toxicity* complaints were very common. For instance, Spout seems to come the closest to Yik Yak in terms of bullying behavior, with reviews such as,

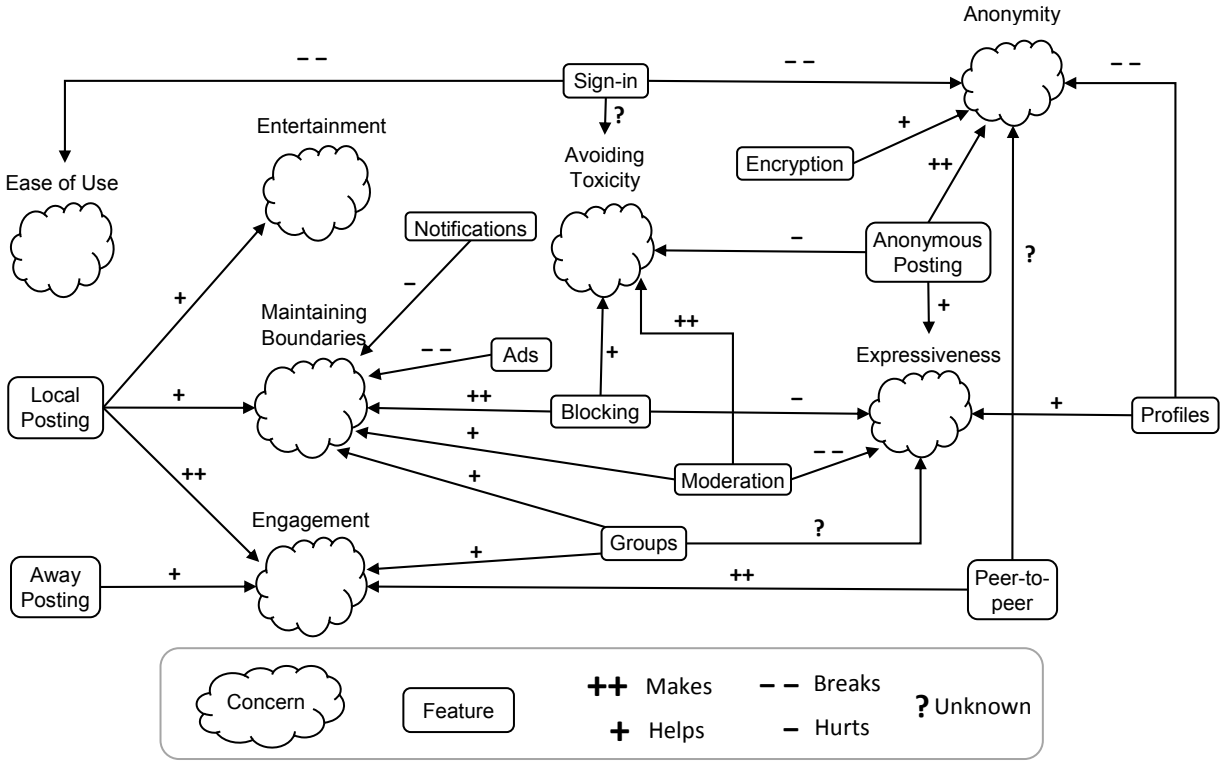

Fig. 3: A model of the common user concerns and their relationships to the core features of the domain of anonymous social networking apps.

> *Truly a place where you can see the ugliness of the internet and cyber bullying. The mob mentality of many users in this app makes it unappealing. It will never be better than yik yak*

Anonymity was also a concern for all users on all apps, indicating that supporting this feature is a must for any app to be able to compete in this domain. Jodel had the most posts about *anonymity*, but many are actually positive, with users expressing satisfaction at the level of privacy offered by the app. Jodel is one of Yik Yak's alternatives that does not require any sort of sign up to use,

> *Love how you don't have to sign up because of anonymity. Sometimes we need to ask questions for people around us and get just the answers. No introduction or getting to know one another.*

Firechat also received numerous *anonymity* related reviews, but for the opposite reason. Despite not requiring internet access, Firechat still requires users to create a global, unique profile online before it can be used. This behavior was remarked on angrily by users, with posts such as,

> *It's supposed to be decentralized and operate locally and therefore it [should] just store accounts locally instead of signing up through your centralized servers*

Entertainment concerns were common in reviews, but extremely rare on Twitter. This may be explained based on the fact that the tweets considered in our analysis were intended to be a direct communication with the app developer, thus, users are less likely to tell the developer *"your app is fun"*. However, despite the lack of tweets, expressions about *entertainment* were still common in reviews, with many posts such as *"It's fun!"* and *"I enjoy using it!"*. In general, our model shows that a local posting feature would help apps to be more *entertaining* and increase users' *engagement* while using the app.

Posts about *maintaining boundaries* included users discussing the ability to limit their communication with only certain people. In some other cases, users were concerned about the behavior of the app itself. Users particularly wanted apps which were not intrusive, and which did not force them to be aware of activities they would rather ignore. Notifications and advertisements were a common cause of complaint; a Whisper user tweeted *"@Whisper can I pay to remove ads on the app? They're so intrusive and they kill the experience"*. Firechat feared the worst in this regard, with users outraged by push notifications being used as advertisements.

In general, our model suggests that the majority of users are concerned about *toxicity*, at the same time, they do not want to lose their *anonymity*. Therefore, some sort of a moderation policy is a must for any app trying to compete in this market. However, app developers must carefully consider the impact of that policy on users' freedom of *expression* as strict policies seem to turn users away from the app. Furthermore, our analysis suggests that apps in this domain should be careful about the amount and type of notifications they send to their users. Allowing users to block other users or control what

type of notifications to see, or finding less intrusive ways for rolling out ads, would help users to *maintain their boundaries*. Features such as local and away posting are also important to enhance users *entertainment* and *engagement*, while user profiles and mandatory sign-up should probably be avoided as they are perceived by users to be hurting their *anonymity* and overall *usability* of the app.

IV. Discussion and Expected Impact

The domain model in Fig. 3 shows that several in-depth insights can be gleaned from analyzing user concerns along with their relations to the core features of the domain. Specifically, the model provides an effective mechanism for externalizing domain knowledge, providing access to information that can be otherwise invisible to app developers. Through the model's dependency relations, developers can get a sense of the synergy and trade-offs between features and concerns, and thereby adjust their release strategies to focus on features that enhance desirable concerns and mitigate the negative ones. For example, according to our model, a new app in the domain of anonymous social networking apps should start by offering a feature for anonymous posting with some sort of moderation. A second priority feature would be to add local and away posting functionalities to enhance the sense of community (engagement). A low priority feature might include peer-to-peer (P2P) posting. This feature can be postponed to later releases since users seem to be less concerned about it (Currently only supported in Firechat). Similarly, if an existing app wants to attract former Yik Yak users, or to avoid a similar destiny, they can consult our model to get an up-to-date picture of what features to tweak, add, or drop. For instance, existing apps could enforce a more strict moderation policy. However, this policy should be implemented in such a way that does not overly constrain users' expressiveness. Furthermore, an app which offers anonymous posting should start thinking about dropping the mandatory user registration (profiles) feature to avoid losing users. The model also shows that apps pushing ads on users might need to tweak their monetization strategy to avoid invading users' space (i.e., Maintaining boundaries).

The type of information provided by our model can be particularly useful for smaller businesses and startups trying to break into the app market [46], [47]. Startups are different from traditional companies in the sense that they have to immediately and accurately identify and implement a product, often known as the Minimum Viable Product (MVP), that delivers actual customer value [46], [48]. Our model will serve as a core asset that will help startup companies, with little operating history, to get a quick and comprehensive understanding of the history of the domain, providing information about how specific user concerns, features, and their relations have emerged and evolved as apps in the domain have evolved. After release, developers can further use the model to automatically track users, as well as rival apps, reactions to their newly-released features. Such knowledge can then be utilized to make more informed release engineering decisions for future releases of the MVP.

In summary, our case study provides initial evidence regarding the feasibility and value of creating such models for other domains. Our overarching goal is to be able to generate such models automatically for any domain, focusing on active areas of the app market where apps are facing a greater risk of failure or losing their user base to other competing apps. Furthermore, our work has shown the value of leveraging failure in the app market. The ability to analyze individual cases of failure represents a unique opportunity for gaining individual and organizational knowledge about what went wrong, the key learning points, and how to prevent such failures in the future [49].

In terms of threats to validity, the analysis presented in the paper takes the form of a case study. In empirical software engineering research, case studies are conducted to investigate a single entity or phenomenon in its real-life context within a specific time space [50]. While case studies can be more realistic and easier to plan than experiments, their results can be difficult to generalize. Furthermore, the majority of the analysis was carried out manually by the authors. Therefore, some of our outcomes might have been impacted by subjectivity. However, case-studies are observational in nature. Therefore, relying on qualitative manual analysis is not uncommon. While these threats are inevitable, we attempted to partially mitigate them by using a systematic process to examine and interpret the multiple sources of information included in our analysis and by having the two coders (authors) working independently to identify the main features of the domain (Table II). We further proposed a systematic feature-concern coding procedure to impose objectivity and enable others to replicate our results.

V. Related Work

The research on app store analysis has noticeably advanced in the past few years. Numerous studies have been conducted on app user feedback classification, summarization, and prioritization. A comprehensive survey of these studies is provided in [16]. In this section, we selectively review and discuss important work related to app success and survivability.

Lim et al. [37] introduced *AppEco*, an artificial life simulation which simulates the app ecosystem using simple procedural rules. Apps were represented as sets of features, where each feature can be present or absent. These sets were compared with the desires of virtual users, who downloaded the app if it implemented some of their desired features. The simulation revealed that the most successful strategy by far was simply to copy the features of the most popular apps. The results also showed that more experienced developers optimized their apps better, while copycats did not improve their performance over time.

In [12], Lim et al. followed up on their earlier study in [37] to determine which strategies for presenting apps to users was best for the app ecosystem's health as a whole. In their simulation, users would search through lists of top apps, new apps, and various keyword searches, and would browse apps in order to choose the one to download. The results

showed that the most effective ranking schemes were those that responded the fastest to changing desires. Specifically, ranking searches according the previous day's downloads led to the most relevant results.

Gómez et al. [14] introduced App Store 2.0, a visionary app store which contains a risk analysis tool to locate potential performance and crash problems, aggregate crash logs, and automatically generate reproducible scenarios for bug testing. To test the feasibility of their vision, the authors implemented a prototype app store back-end, released as a set of tools and APIs that developers apply to their apps to generate the necessary raw data for analysis. The risk analysis tool was applied to mine more than $10,000$ potentially crash-prone apps [51]. The crash analysis tool was found to be useful for fixing bugs.

Petsas et al. [2] studied the download distribution of apps to determine the ideal developer strategy for pricing in four popular Android app marketplaces. The analysis revealed that free apps followed a different distribution than paid apps. More specifically, downloads for free apps demonstrated a *clustering effect* with 90% of the downloads going to 10% of the apps. The results also showed that free apps with an ad-based revenue strategy may result in higher financial benefits than paid apps.

In order to help developers stay informed about their app's community, Fu et al. [52] developed Wiscom, a system for discerning major app user concerns and preferences. The proposed system uses Latent Dirichlet Allocation (LDA) [53] to determine the major topics of complaints in negative reviews. The proposed system was evaluated on a high-quality corpus of reviews gathered from over 50,000 apps. The results showed that Wiscom was able to detect the inconsistencies between user comments and ratings, identify the major reasons why users disliked an app, and learn how users' complaints changed over time.

Li et al. [54] analyzed one month of data from the Chinese app store Wandoujia to determine when and why users installed apps. By examining app co-installation rates, the authors found that music apps were often paired with games, and apps for communication often coincided with apps related to video sharing. Uninstallation patterns were also analyzed to determine the fate of abandoned apps. The results showed that 93% of abandoned apps did not return to prominence (e.g., apps tend to be abandoned by users rapidly and with finality).

Our brief review shows that related work in the literature has focused on either mining user reviews for technical feedback or studying the factors that influenced success in the app market. Our work in this paper extends existing work along multiple dimensions. First, in our analysis, we focus on extracting and modeling user feedback at a domain, rather than individual app, level. Second, we introduce failure as a main source of knowledge that can be utilized in order to enable a better understanding of the dynamic nature of user requirements in the mobile app ecosystem. Third, we demonstrate the feasibility of integrating human (user feedback on Twitter and the app store), market (download rates and user migration patterns), and system (feature analysis) information to generate a unified domain model that reflects an accurate picture of the current state of the domain.

VI. Summary and Future Work

The case study reported in this paper systematically analyzed the ripple effect of the failure of a widely successful app on its competitors and their main features. Specifically, in this paper, we investigated the rise and fall of the social networking app Yik Yak and the impact it left on the app market. Our analysis can be described as a three-step procedure. First, we tracked multiple news outlets, social media platforms, and app download statistics, to identify the main competing apps that former Yik Yak users migrated to. We then performed a qualitative feature-oriented domain analysis, using a direct analysis of the app features and a systematic analysis of user feedback, to identify the main functional features and user concerns of the domain. A domain model was then created, using F-SIG notation, to depict the interrelationships between extracted user concerns and the core features of the domain.

The case study reported in this paper represents a first-of-its-kind in-depth analysis of modeling user concerns in the app market. Our expectation is that, applying such analysis and modeling at a large scale will provide app developers with a systematic understanding of their domain as well as help them to predict the impact of feature changes on their end-user acquisition and retention rates. To achieve these long term goals, the work presented in this paper will be expanded along two dimensions:

- More case studies: Similar studies focused on other cases of success and failure in the app store (e.g., Vine and Pokèmon Go) will be conducted. Our objective is to build a knowledge-base of success and failure in the mobile app ecosystem. Such knowledge will be later used to derive a formal theory that can be used to explain the different factors that control app survival.
- Automated modeling: the qualitative analysis carried out in this paper was mainly manual. However, repeating such type of analysis over larger domains with hundreds of apps and millions of reviews and tweets can be a time-consuming and laborious task. Therefore, our future work will be focused on automating the process, utilizing existing automatic app store and social media mining methods [15], [16], [17], [18] as well as automated domain modeling techniques [55], [56], [57], [58], [59]. Our goal is to devise automated methods for data collection, domain feature analysis, and model generation. These methods will be evaluated over large datasets of app data to ensure their practicality and accuracy.

References

[1] L. Villarroel, G. Bavota, B. Russo, R. Oliveto, and M. Di Penta, "Release planning of mobile apps based on user reviews," in *International Conference on Software Engineering*, 2016, pp. 14–24.

[2] T. Petsas, A. Papadogiannakis, M. Polychronakis, E. Markatos, and T. Karagiannis, "Rise of the planet of the apps: A systematic study of the mobile app ecosystem," in *Conference on Internet Measurement*, 2013, pp. 277–290.

[3] R. Basole and J. Karla, "Value transformation in the mobile service ecosystem: A study of app store emergence and growth," *Service Science*, vol. 4, no. 1, pp. 24–41, 2012.

[4] P. Coulton and W. Bamford, "Experimenting through mobile apps and app stores," *International Journal on Mobile Human Computer Interaction*, vol. 3, no. 4, pp. 55–70, 2011.

[5] A. Finkelstein, M. Harman, Y. Jia, W. Martin, F. Sarro, and Y. Zhang, "App store analysis: Mining app stores for relationships between customer, business and technical characteristics," University of College London, Tech. Rep. rN/14/10, 2014, Tech. Rep., 2014.

[6] M. Harman, Y. Jia, and Y. Zhang, "App store mining and analysis: MSR for app stores," in *IEEE Working Conference on Mining Software Repositories*, 108–111, p. 2012.

[7] S.-Y. Ihm, W.-K. Loh, and Y.-H. Park, "App analytic: A study on correlation analysis of app ranking data," in *International Conference on Cloud and Green Computing*, 2013, pp. 561–563.

[8] G. Lee and T. Raghu, "Product portfolio and mobile apps success: Evidence from app store market," in *Americas Conference Information Systems*, 2011, pp. 3912–3921.

[9] Z. Svedic, "The effect of informational signals on mobile apps sales ranks across the globe," Ph.D. dissertation, School Business Faculty, Simon Fraser University, 2015.

[10] T. Mens, M. Claes, P. Grosjean, and A. Serebrenik, *Studying Evolving Software Ecosystems based on Ecological Models*. Springer Berlin Heidelberg, 2014, pp. 297–326.

[11] O. Carare, "The impact of bestseller rank on demand: Evidence from the app market," *International Economic Review*, vol. 53, no. 3, pp. 717–742, 2012.

[12] S. Lim and P. Bentley, "Investigating app store ranking algorithms using a simulation of mobile app ecosystems," in *Congress on Evolutionary Computation*, 2013, pp. 2672–2679.

[13] Localytics. (2016) Cheat sheet: Overall app benchmarks h2 2016. Localytics. [Online]. Available: https://www.localytics.com/lp/cheat-sheet-overall-app-benchmarks-h2-2016/

[14] M. Gómez, B. Adams, W. Maalej, M. Monperrus, and R. Rouvoy, "App store 2.0: From crowdsourced information to actionable feedback in mobile ecosystems," *IEEE Software*, vol. 34, no. 2, pp. 81–89, 2017.

[15] N. Jha and A. Mahmoud, "Mining user requirements from application store reviews using frame semantics," in *Requirements Engineering: Foundation for Software Quality*, 2017, pp. 273–287.

[16] W. Martin, F. Sarro, Y. Jia, Y. Zhang, and M. Harman, "A survey of app store analysis for software engineering," *IEEE Transactions on Software Engineering*, vol. 43, no. 9, pp. 817–847, 2017.

[17] W. Maalej, Z. Kurtanović, H. Nabil, and C. Stanik, "On the automatic classification of app reviews," *Requirements Engineering*, vol. 21, no. 3, pp. 311–331, 2016.

[18] E. Groen, S. Kopczyńska, M. Hauer, T. Krafft, and J. Doerr, "Users: The hidden software product quality experts?: A study on how app users report quality aspects in online reviews," in *International Requirements Engineering Conference*, 2017, pp. 80–89.

[19] N. Chen, J. Lin, S. Hoi, X. Xiao, and B. Zhang, "AR-Miner: Mining informative reviews for developers from mobile app marketplace," in *International Conference on Software Engineering*, 2014, pp. 767–778.

[20] W. Maalej and H. Nabil, "Bug report, feature request, or simply praise? On automatically classifying app reviews," in *Requirements Engineering Conference*, 2015, pp. 116–125.

[21] E. Guzman, R. Alkadhi, and N. Seyff, "A needle in a haystack: What do Twitter users say about software?" in *International Requirements Engineering Conference*, 2016, pp. 96–105.

[22] E. Guzman, M. Ibrahim, and M. Glinz, "A little bird told me: Mining tweets for requirements and software evolution," in *International Requirements Engineering Conference*, 2017, pp. 11–20.

[23] G. Williams and A. Mahmoud, "Mining Twitter feeds for software user requirements," in *International Requirements Engineering Conference*, 2017, pp. 1–10.

[24] M. Nayebi, H. Cho, H. Farrahi, and G. Ruhe, "App store mining is not enough," in *International Conference on Software Engineering Companion*, 2017, pp. 152–154.

[25] D. Levinthal and J. March, "A model of adaptive organizational search," *Journal of Economic Behavior and Organization*, vol. 2, no. 4, pp. 307–333, 1981.

[26] P. Madsen and V. Desai, "Failing to learn? The effects of failure and success on organizational learning in the global orbital launch vehicle industry," *Academy of Management Journal*, vol. 53, no. 3, pp. 451–476, 2010.

[27] R. Junco. (2015) Yik Yak and online anonymity are good for college students. Wired. [Online]. Available: https://www.wired.com/2015/03/yik-yak-online-anonymity-good-college-students/

[28] A. Shontell. (2015) The inside story of Yik Yak. Business Insider. [Online]. Available: http://www.businessinsider.com/the-inside-story-of-yik-yak-2015-3

[29] J. Mahler. (2015) Who spewed that abuse? Anonymous Yik Yak app isn't telling. New York Times. [Online]. Available: https://www.nytimes.com/2015/03/09/technology/popular-yik-yak-app-confers-anonymity-and-delivers-abuse.html?mcubz=1

[30] A. Shontell. (2015) Why a girl who was viciously bullied on Yik Yak now believes in the anonymous app's future. Business Insider. [Online]. Available: http://www.businessinsider.com/elizabeth-long-was-bullied-on-yik-yak-2015-3

[31] J. Graham. (2017) Yik Yak, the once popular and controversial college messaging app, shuts down. USA Today. [Online]. Available: https://www.usatoday.com/story/tech/talkingtech/2017/04/28/yik-yak-shut-down/101045670/

[32] Z. Kurtanović and W. Maalej, "Mining user rationale from software reviews," in *International Requirements Engineering Conference*, 2017, pp. 61–70.

[33] S. Stolberg and R. Peŕez-Penã. (2016) Wildly popular app kik offers teenagers, and predators, anonymity. New York Times. [Online]. Available: https://www.nytimes.com/2016/02/06/us/social-media-apps-anonymous-kik-crime.html?mcubz=1

[34] S. Hoeller. (2016) This app lets you text without WiFi or a data plan anywhere in the world — even on a plane. Business Insider. [Online]. Available: http://www.businessinsider.com/firechat-app-lets-you-text-without-wifi-or-data-2016-9

[35] M. Katz. (2017) These failed apps discovered a hidden rule of the web. Wired. [Online]. Available: https://www.wired.com/2017/03/these-failed-apps-discovered-a-hidden-rule-of-the-web/

[36] K. Pohl, G. Boeckle, and F. van der Linden, *Software Product Line Engineering : Foundations, Principles, and Techniques*. Springer, 2005.

[37] L. Lim and J. Bentley, "How to be a successful app developer: Lessons from the simulation of an app ecosystem," *ACM SIGEVOlution*, vol. 6, no. 1, pp. 2–15, 2012.

[38] P. Brandtzæg and J. Heim, "Why people use social networking sites," in *International Conference on Online Communities and Social Computing*, 2009, pp. 143–152.

[39] K. Lin and H. Lu, "Why people use social networking sites: An empirical study integrating network externalities and motivation theory," *Computers in Human Behavior*, vol. 27, no. 3, pp. 1152–1161, 2011.

[40] M. Lane and P. Coleman, "Technology ease of use through social networking media," *Journal of Technology Research*, vol. 3, pp. 1–12, 2012.

[41] K. Kang, S. Cohen, J. Hess, W. Novak, and S. Peterson, "Feature-oriented domain analysis (FODA) feasibility study," Software Engineering Institite, Carnegie Mellon University, Tech. Rep. CMU/SEI-90-TR-21, 1990.

[42] J. Mylopoulos, L. Chung, S. Liao, H. Wang, and E. Yu, "Exploring alternatives during requirements analysis," *IEEE Software*, vol. 18, no. 1, pp. 92–96, 2001.

[43] M. Glinz, "On non-functional requirements," in *International Requirements Engineering Conference*, 2007, pp. 21–26.

[44] B. Gonzales-Baixauli, J. C. S. Prado Leite, and J. Mylopoulos, "Visual variability analysis for goal models," in *International Requirements Engineering Conference*, 2004, pp. 198–207.

[45] S. Jarzabek, B. Yang, and S. Yoeun, "Addressing quality attributes in domain analysis for product lines," *IEE Software Proceedings*, vol. 153, no. 2, pp. 61–73, 2006.

[46] A. Nguyen Duc and P. Abrahamsson, "Minimum viable product or multiple facet product? The role of MVP in software startups," in *Agile Processes in Software Engineering and Extreme Programming*, 2016, pp. 118–130.

[47] C. Giardino, X. Wang, and P. Abrahamsson, "Why early-stage software startups fail: A behavioral framework," in *International Conference of Software Business*, 2014, pp. 27–41.

[48] N. Paternoster, C. Giardino, M. Unterkalmsteiner, T. Gorschek, and P. Abrahamsson, "Software development in Startup companies: A systematic mapping study," *Information and Software Technology*, vol. 56, no. 10, pp. 1200–1218, 2014.

[49] J. Storey and E. Barnett, "Knowledge management initiatives: Learning from failure," *Journal of Knowledge Management*, vol. 4, no. 2, pp. 145–156, 2000.
[50] C. Wohlin, P. Runeson, M. Höst, M. Ohlsson, B. Regnell, and A. Wesslèn, *Experimentation in Software Engineering*. Springer, 2012.
[51] M. Gómez, R. Rouvoy, M. Monperrus, and L. Seinturier, "A recommender system of buggy app checkers for app store moderators," in *Mobile Software Engineering and Systems*, 2015, pp. 1–11.
[52] B. Fu, J. Lin, L. Li, C. Faloutsos, J. Hong, and N. Sadeh, "Why people hate your app: Making sense of user feedback in a mobile app store," in *Conference on Knowledge Discovery and Data Mining*, 2013, pp. 1276–1284.
[53] D. Blei, A. Ng, and M. Jordan, "Latent Dirichlet Allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.
[54] H. Li, X. Lu, X. Liu, T. Xie, K. Bian, F. Lin, Q. Mei, and F. Feng, "Characterizing smartphone usage patterns from millions of android users," in *Conference on Internet Measurement*, 2015, pp. 459–472.
[55] M. Acher, A. Cleve, G. Perrouin, P. Heymans, C. Vanbeneden, P. Collet, and P. Lahire, "On extracting feature models from product descriptions," in *International Workshop on Variability Modeling of Software-Intensive Systems*, 2012, pp. 45–54.
[56] K. Czarnecki, S. She, and A. Wasowski, "Sample spaces and feature models: There and back again," in *International Software Product Line Conference*, 2008, pp. 22–31.
[57] A. Ferrari, G. Spagnolo, and F. Dell'Orletta, "Mining commonalities and variabilities from natural language documents," in *International Software Product Line Conference*, 2013, pp. 116–120.
[58] N. Hariri, C. Castro-Herrera, M. Mirakhorli, J. Cleland-Huang, and B. Mobasher, "Supporting domain analysis through mining and recommending features from online product listings," *IEEE Transactions on Software Engineering*, vol. 39, no. 12, pp. 1736–1752, 2013.
[59] X. Lian, J. Cleland-Huang, and L. Zhang, "Mining associations between quality concerns and functional requirements," in *International Requirements Engineering Conference*, 2017, pp. 292–301.