FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Analyzing and visualizing long-term microblogging data

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Medieninformatik

eingereicht von

Christian Drescher, BSc
Matrikelnummer 0430240

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Ao. Univ. Prof. Margit Pohl
Mitwirkung: Dipl.-Ing. Dr. Simone Kriglstein
Dipl.-Ing. Dr. Guenter Wallner

Wien, 4. Februar 2019

Christian Drescher | Margit Pohl

FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Analyzing and visualizing long-term microblogging data

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Media Informatics

by

Christian Drescher, BSc
Registration Number 0430240

to the Faculty of Informatics

at the TU Wien

Advisor: Ao. Univ. Prof. Margit Pohl
Assistance: Dipl.-Ing. Dr. Simone Kriglstein
Dipl.-Ing. Dr. Guenter Wallner

Vienna, 4th February, 2019

Christian Drescher | Margit Pohl

Erklärung zur Verfassung der Arbeit

Christian Drescher, BSc
Alex-Wedding-Str. 5/1005, 10178 Berlin

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 4. Februar 2019

Christian Drescher

Acknowledgements

I would first like to thank my thesis advisor Ao. Univ. Prof. Margit Pohl of the Institute of Visual Computing and Human-Centered Technology at the University of Technology Vienna, as well as Dipl.-Ing. Dr. Simone Kriglstein and Dipl.-Ing. Dr. Guenter Wallner. They always had an open door for me whenever I had a question about my research or writing and provided valuable feedback on the application of this work and on the thesis itself.

I would also like to thank Prof. Anders Drachen of the Department of Computer Science of the University of York and Rafet Sifa of the Fraunhofer Institute Intelligent Analysis and Information Systems IAIS for providing the in-game data set on which the use case of this thesis is based, and also for their valuable feedback and suggestions regarding the the application implemented as part of this work.

Furthermore, I am grateful to Johannes Binder for his comments and advice on different software architectural and programming issues that came up while working on the application of this work.

Finally, I must express my very profound gratitude to my parents Veronika and Günther, to my brothers Klaus and Roland, and to my partner Jen-Yie for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

Kurzfassung

Jeden Tag nutzen Millionen von Menschen Soziale Netzwerke und Mikroblogging Plattformen um Persönliches aus ihrem Leben zu veröffentlichen, sich über ihre Interessen zu unterhalten, an Diskussionen zu aktuellen Neuigkeiten teilzunehmen oder ihre Meinung zu verwendeten Produkten oder Diensten kundzugeben. Hand in Hand mit immensen Wachstum über die letzten Jahre und der einfachen Verfügbarkeit dieser mit zahlreichen Meta-Informationen angereicherter Daten, stieg auch das wissenschaftliche Interesse und resultierte in unterschiedlichen Forschungsfeldern rund um Verarbeitung und Nutzbarkeit dieser Daten.

Während sich viele Forschungsprojekte auf einzelne, markante Ereignisse konzentrieren, ergründet diese Arbeit die Verwendung von Langzeit-Mikrobloggingdaten mit Bezug auf ein bestimmtes Produkt. Eine Analyse- und Visualisierungsapplikation wird vorgestellt, die es ermöglicht, *Twitter*-Daten mit Benutzeraktivitäten eines Produktes zu verknüpfen und zu visualisieren.

Mit Hilfe eines Use Cases wird die explorative Herangehensweise in der Verwendung der Applikation veranschaulicht. Es wird unter anderem gezeigt, dass mit Hilfe der verwendeten Visualisierungen einflussreiche *Twitter* Benutzer identifiziert werden können und dass das Bilden eines täglichen, generellen Stimmungswertes von *Twitter*-Aktivitäten durch die starke Abhängigkeit von einzelnen Events trügerisch sein kann. Außerdem wurde für den gewählten Use Case keine generelle Korrelation zwischen der Anzahl an *Tweets* und den Benutzeraktivitäten eines Produktes gefunden und weitere Untersuchungen sind notwendig, um herauszufinden, ob dies auch auf andere Use Cases zutrifft.

Abstract

Millions of people use social media networks and microblogging services on a daily basis to talk about their personal lives, interests, current news and events, and experiences with products and services. The combination of enormous growth of such platforms over the recent years and the availability of all this data, enriched with a variety of meta data, led to an increase in scientific interest resulting in many different fields related to processing this data.

While many projects focus on distinctive events, this work explores possibilities to use long-term microblogging data related to a specific product. An analysis and visualization application is introduced which enables analysts to link *Twitter* data to the usage data of a product and utilize various visualizations to gain insights.

Furthermore, a use case is presented which illustrates the explorative process of using the application and outlines the main findings of this work: influential user accounts can be identified based on both *Twitter* and product usage data, the overall sentiment of *Twitter* daily activities might not be a correct representation of what is really happening, and no general correlation between the amount of *tweets* posted and product usage data could be found within the scope of the use case. Therefore, additional research is required in order to see if this applies to other use cases as well.

Contents

CHAPTER 1

Introduction

Each day millions of people turn to various social media and microblogging services in order to share information about all kinds of topics: amongst other things they publish posts, which can contain not only text, but also multimedia data like images, videos or audio files. They communicate about their personal lives, talk about their interests, participate in discussions about current news and events, or share their views about products and services they use. Below the surface a vast amount of meta-information is available: information like categorization data, which is automatically generated by extensive algorithms based on the posts' content, demographic data of users, relations between interacting users, or geo-tagging data is attached to each and every user-generated piece of information posted on social media or microblogging services.

Therefore, the interest from businesses and researchers alike in understanding and harnessing this *crowd-sourcing power* has rapidly increased over the last couple of years: social media and microblogging services became an extremely useful information source because they enable researchers to access these huge amounts of data, which can be analyzed in order to gain valuable insights about the interests and opinions of users from a potentially highly accurate target group. As a result, various research fields have emerged, related to the analysis of behaviors, characteristics and demographics of social media and microblogging users.

While many studies focus on single, short-term incidents that send strong, easily detectable repercussions throughout the social web, analyzing user-generated data related to a product or service over a longer period of time can yield a multitude of valuable insights and be of utmost importance to any business: due to the immediate nature of social media and microblogging communication, businesses can receive feedback virtually instantly, which allows them to quickly react to issues, they can deepen their understanding about their target audience in order to develop strategies to enhance customer loyalty or simply to leverage marketing for their products and services. As several studies have shown, electronic word-of-mouth and the sentiment of people's conversations do considerably

impact purchase decisions [BSDB16], sales performance [Del13], or early product adoption [HTWF15].

Hence, using social media and microblogging services in order to understand how people react to news, updates and changes of a product or service is key for any business, which is information that until recently could only be gained by asking for customer feedback directly or by conducting consumer surveys. Learning how information is distributed on social media and microblogging platforms and at which pace information *travels* throughout the social web if people are using a platform's sharing functionality is as important as identifying people of great influence within a thematic domain, determining locations and areas with a higher amount of users from a target group, or extracting the sentiment of what people share on the social web. But besides collecting, filtering and processing data, visualizations pose a big challenge due to the immense amount of data that can be obtained from social media and microblogging services. In addition, a broad range of different visualization types is required in order to represent this multivariate data in a clearly arranged and understandable way.

To sum up, this work seeks to provide answers to the following research questions:

- How can long-term microblogging data be used in order to analyze the behavior of a product's users?
  - Can influencing microblogging users be identified?
  - How is the sentiment overall?
  - Does the sentiment change after the release of product changes or updates?
  - Does the developer react to user feedback and how?
  - Are there regional differences in user feedback?
- Can microblogging activities be connected to a product's usage data?
  - Can account names of a product's users be identified through microblogging activities?
  - Does higher microblogging activity correlate with increased usage of a product?
- How can multivariate long-term data be visualized?

To answer these questions, this work introduces an interactive web application that facilitates a visual data analysis approach in order to explore large-scale microblogging data related to a specific product or service. Besides discussing the development process and structure of the application, various visualizations based on analysis methods such as sentiment analysis are presented, that illustrate ways to find patterns and relations within multivariate microblogging data, but also set this data in relation with actual usage data of a product or service.

The application was designed in a way that allows building custom dashboards by enabling analysts to use different widths for visualizations and to freely arrange them. As a result, this work has many applications, but can be especially useful to researchers that want to visually explore large amounts of microblogging data related to a specific topic or to businesses that want to learn about their target audience, track and enhance the performance of their products and services as well as their social media activities. By relating social media and microblogging activities directly to the usage of products and services, the impact of news associated to the chosen subject, as well as updates and extensions of the product or service themselves can be measured nearly instantly, and analyzed in order to leverage the insights towards improving products and services.

In addition to introducing the application itself, this work presents a use case, which shows and discusses various usage scenarios of the application based on collected *Twitter* data related to the online-only massively multiplayer first-person shooter video game *Destiny* [Bun]. While the domain of video games in general proved to be a very good choice as a use case due to the growth and activity of gamers on *Twitter*, as pointed out by Bateman [Bat16], there were more reasons to choose *Destiny*: i) it provides an Application Programming Interface (API) [Bun17] which makes it possible to access various types of in-game data like account information and all sorts of in-game activity related metrics, ii) the game is constantly updated and extended with new in-game content by its developer *Bungie* resulting in times with increased feedback from users, and iii) an extensive and very active community has formed because a lot of events are taking place in-game featuring challenges and rewards leading to a high volume in activity on social media and microblogging platforms.

Twitter was used as microblogging data source because it is currently one of the most popular social media services and also provides public APIs [Twic] which not only give access to massive amounts of multivariate data, but at the same time support complex search queries in order to mine only data that is relevant to the selected product. The obtained *Twitter* data was then analyzed and set into relation to actual in-game data, so that a more holistic view of microblogging activities and in-game behavior could be achieved. To give an example, social media activities of players could shed light upon the question why in-game activities decreased or increased at certain times.

As will be discussed more detailed in the literature review, this work's main contribution to knowledge is an application that allows analyzing and visualizing long-term microblogging data which is set into relation to usage data of a product and service. As the use case of this work demonstrates, the additionally provided context is a valuable benefit: the reasons for fluctuations in in-game usage data, for example, can be better understood by analyzing what people were talking about on *Twitter* at the time those fluctuations took place. The other way around, influential *Twitter* accounts can not only be identified solely based on *Twitter* statistics such as follower count, or the amount of *re-tweets* or likes their *tweets* receive, but also by incorporating in-game metrics such as their activity or performance in certain game modes.

To summarize, the methodological approach of this works consists of two steps: i)

Microblogging data related to *Destiny* was acquired by using the public *Twitter* API. In total 1,062,390 *tweets* from 246,881 users over a period of roughly 14 months were collected and stored in a database. The sentiment of each *tweet* was analyzed by using the *Sentiment140* [GBH09b] API[1], while *Twitter* user profiles were searched for *Xbox Live* (XBL) gamertags or *PlayStation Network* (PSN) IDs used for the *Destiny* API to collect gameplay data of 3,548 players during a period of about 6 months which was provided by colleagues of the *University of York* and *Fraunhofer IAIS*. ii) Data is displayed via various interactive chart and timeline visualizations, as well as data tables in order to reveal patterns and insights in an easy and comprehensible way.

Furthermore, this work is structured as follows: i) First, it takes a look at the state of the art in order to deliver insights into what researchers in recent years have accomplished in this rapidly growing field of analyzing social media and microblogging data. ii) Secondly, an application system overview is presented in order to illustrate the inner workings of the application and which components it consists of. iii) After that, the *Twitter* API and its multivariate data structure, as well as the in-game data set is explained at a deeper level in order to show how the data sample was collected and analyzed, iv) followed by a detailed explanation of how the visualization component of the application works and which visualization types are supported. v) Hereupon, the use case *Destiny* is presented giving a brief introduction to the game and presenting visualizations and findings. This work is completed by vi) discussing challenges, limitations and future work, and vii) a conclusion.

The rough idea of connecting microblogging data to a product's usage data, on which this work is based, as well as using *Destiny* as a use case was initially defined by Dipl.-Ing. Dr. Kriglstein S. and Dipl.-Ing. Dr. Wallner G., while a partnership with Prof. Drachen A. of the Department of Computer Science of the *University of York* and Sifa R. of the *Fraunhofer Institute Intelligent Analysis and Information Systems IAIS* regarding the contribution of a *Destiny* in-game data set was already in place. Multiple feedback sessions took place throughout the project, which resulted in the refinement of the initial idea and evolution of the application into its final state. Apart from that, the work was solely done by the author including the review of existing literature, the research of frameworks and tools, the development of the application, the crawling of the *Twitter* data set, the extraction of XBL gamertags and PSN IDs, and the writing of the thesis in its entirety.

[1] Sentiment140 provides APIs for classifying tweets: http://help.sentiment140.com/api/, last accessed: 2017-09-26

CHAPTER 2

Related work

Over the course of the last two decades various social network sites have appeared and impacted the lives of millions of people. As defined by Boyd and Ellison [BE08], social network sites are web-based services that at their core consist of the following three features: i) users can create profiles containing self-reported data about themselves like age, location, interests and a profile picture, which can be publicly or semi-publicly visible, ii) they can view a list of other users with whom they have a connection in common, and iii) they can examine their own list of connected users as well as connections of other people, although the definition and the term of those connections can vary from site to site. Although the term *social network site* is superficially used as a synonym or general term for *social media*, the latter has its differences and peculiarities, and was defined by Kaplan and Haenlein as "Social Media is a group of Internet-based applications that build on the ideological and technological foundations of Web 2.0, and that allow the creation and exchange of User Generated Content" [KH10]. Furthermore, the terms *microblog* or *microblogging service* can be seen as a special form of social media which is "a combination of blogging and instant messaging that allows users to create short messages to be posted and shared with an audience online" [NM17] with *Twitter* being one of the most prominent examples for microblogging services which grew from 30 million monthly active users in the first quarter of 2010 to 328 million monthly active users in the first quarter of 2017 [Sta17], resulting in around 500 million *tweets* posted by users each day [Asl17].

This immense growth fueled an enormous amount of interest within the scientific community throughout a vast amount of research areas.

2.1 Collecting and analyzing social media and microblogging data

As Goonetilleke et al. [GSZS14] pointed out, there are issues to solve around the big data nature of *Twitter*. After reviewing various approaches of research projects working with microblogging data they identified the following challenges and research issues.

First of all, the data collection process itself faces multiple challenges due to the request restrictions [Twib] set by *Twitter*. A so-called *focused crawler* is required, which retrieves *tweets* based on a predefined set of parameters and is optimized on accessing the API by minimizing the number of requests in regards to the rate limits. Valkanas et al. [VSG14], for example, introduced the basic structure for such a data crawler, which is based on the Streaming [Twia] and Search [Twic] API provided by *Twitter* and honors its different rate limits. They argue that while conventional surface web crawlers are specialized in handling only URLs of the following pages, their suggested crawler utilizes all aspects and features of the provided APIs by adding three new components to the architecture: i) the so-called *STREAMER* component is specialized on handling data from the Streaming API, which has been nicely described by *Twitter* as "making a very long lived HTTP request, and parsing the response incrementally" [Twi17a]. ii) The *SEEDER* component is responsible for storing the results of the previous request as well as updating and defining new queries. iii) Finally, the *RANKER* component handles the initialization of required resources like the database relations, and the provisioning for the crawler in order to initiate the next query. In addition to the aforementioned new components, the classic scheduling and queuing component are enhanced in order to cope with all the different query types of the *Twitter* APIs, as well as enforcing the rate limits.

In addition to that, Goonetilleke et al. [GSZS14] brought forward the argumentation that the data collection process should be further optimized because relevant information might be overlooked despite having a very good knowledge of the target keywords. Quite similar to that, Wang et al. [WTCP13] argue that current approaches, which are based on using only a predefined set of keywords for crawling data, can be at risk to miss a significant amount of relevant data, and propose a more adaptive approach with the goal to enhance the initial keyword set. In the presented keyword adaptation algorithm emerging keywords which are related to the Olympic Games, that took place in 2012 in London, are identified and classified based on their relation to the event before updating the query parameters sent to the *Twitter* API at certain intervals resulting in additional event-relevant information when compared to the data sample based on the initial keyword set only.

A second, not-so-straight-forward issue Goonetilleke et al. [GSZS14] identified, is related to pre-processing *Twitter* data: they argue that although a lot of research has been done in the field of topic detection, part-of-speech tagging, summarization, as well as other linguistic processes, these studies were focused on well-formed documents such as news articles, and that applying such methods on microblogging data is way more trickier due to its uncommon characteristics. Until recently [New17], *tweets* were limited to 140

characters due to technical restrictions dating back to *Twitter's* launch in 2006 [McC16]. This limitation led to an increased usage of abbreviations and uncommon grammar constructions, and therefore raises additional challenges for all text processing related activities, which in most cases can not be applied directly due to the immense amount of available data. What's more, microblogging data can contain extensive contextual information and entities like events, persons and organizations, or places and GPS locations, and, as will be shown later on, a great deal of literature is focusing on the detection of those entities.

Finally, Goonetilleke et al. [GSZS14] present their views on data management challenges: while most projects rely on relational or Resource Description Framework (RDF) models, graph-based models are not that common but promise diverse and interesting ways to access microblogging data since its focus would lie more on the network and relationships of users and the structural properties of *tweets*. To make accessing graph data faster and simpler, they continue by proposing a SQL-like query language, which can handle large volumes of information-rich data, but also point out that indexing and encoding mechanisms have to be efficient.

When talking about limitations imposed by *Twitter* when accessing data via the Stream API, it also has to be said that besides the rate limits [Twib], which only allow a certain amount of requests within 15 minutes for each endpoint, the received data itself is not complete: *Twitter* offers a free, basic version of its APIs which "is focused on relevance and not completeness" [Twi17c] and therefore does not return all data. But it provides various premium APIs for business partners that contain the full amount of data, even going back until *Twitter's* launch in 2006. Morstatter et al. [MPL14, MPLC13] conducted studies to find out if the data sample *Twitter* provides for free is biased. According to their definition, "a hashtag is “biased” if the relative trend is statistically significantly over- or underrepresented in contrast to its true trend on Twitter" [MPL14]. They compared a collected data sample of the free, basic Streaming API, which returns at most 1% of the whole volume and is sampled in a way unknown to the public, to a sample collected with identical parameters from the premium *Firehose* stream, which returns the full volume of data posted on *Twitter*, by investigating different statistical metrics such as top hashtags, discovered topics or geo-location of *tweets*, and measuring various network metrics like the *re-tweet* networks among users. They could show that the topical analysis is more accurate if the volume of data from the Streaming API is high, whereas content-related measurements such as top hashtags can be misleading if the volume is low. Geo-tagged *tweets* on the other hand were almost completely available in the free sample. They summarize that "the results of using the Streaming API depend strongly on the coverage and the type of analysis that the researcher wishes to perform" [MPLC13]. Furthermore, they [MPL14] compared the Sample API, which returns a small and random sample of all public *tweets* in real-time [Twi17b], to a sample of the Streaming API. They could show that the Sample API data is representative in regards to the Streaming API data and, by collecting sample data at different times and from different locations, that the time and the location from which the sample is requested, does not result in significantly

different sample data.

To sum up, this section presented an overview of challenges revolving around the big data nature of social media and microblogging data and is highly relevant to the application developed as part of this work. A faceted crawler similar to the one suggested by Valkanas et al. [VSG14] has been implemented, which uses *Twitter's* Search API [Twic] to collect data while honoring the rate limitations [Twib], before applying pre-processing and filtering methods, and saving data to a relational database. A detailed description of the system and the applied methods is given in Chapter 3.

2.2 Detecting influencers

As briefly touched upon, a vast amount of research fields has emerged alongside the rapid growth of social media and microblogging services. Due to the connections among users such as friends lists and *followers* and the algorithms that define what is displayed in the timeline of users, it is of high interest to detect individuals that accumulated a lot of influence within certain domains. Mazumder et al. [MMP15], for example, presented their work on identifying news-caster of *Twitter*. These are influential users who use *Twitter* as means to share information from online news outlets with their followers in a periodically and consistent manner. Their framework, which has been labeled *NCFinder*, consists of three components: first, i) *tweets* related to a news headline are crawled periodically by using so-called *news concepts*, which are nouns or phrases extracted via Part-of-Speech tagging from a news headline, as keywords and combining it with *http* in order to receive only *tweets* containing links from the *Twitter* Search API. Next, ii) the *tweets* are filtered by checking if the URL used in the *tweet* does indeed point to an authentic news source in order to remove unrelated or fraudulent *tweets* masquerading as news update. Finally, iii) user account details such as the user name of the profile description is tested against the word "news" in an attempt to remove accounts of online news websites themselves, before calculating a daily score based on the user's activity and usage of relevant and authentic news sources and ranking the average score values resulting in a news-caster top list for a specific period of time.

Subbian et al. [SAS16] approached this subject differently and proposed a framework that focuses on processing social media data in real-time, tracking relevant influential metrics and creating a flow path based data structure that is updated incrementally. Since such a tree data structure is hard to maintain due to the potentially extreme large amount of paths, they presented a pruned version that keeps track of the most significant paths resulting in reduced complexity. This enables more efficient queries which return influencers with a higher quality than baseline methods based on individual, context specific or temporal influence values of users.

Moreover, Kalaitzis et al. [KGL+16] presented a system which aims to identify influencing gamers by predicting gaming-related properties such as gaming performance, preferred gaming platform or knowledge on computer games and technology in general. They built

classification and regression models for each trait by using supervised machine learning and could show that a prediction of a user's traits based on the user's *tweets* is possible.

For this work, detecting influencers is also of high interest: from the perspective of a product developer or service provider there could be multiple benefits resulting from identifying influential individuals among social media and microblogging communities, most notably the amplifying impact on the distribution of information which can be utilized for marketing purposes. Although the approach in the application of this work is not as sophisticated as, for example, the *NCFinder* [MMP15], and does not facilitate a tree data structure as proposed by Subbian et al. [SAS16], it does provide a basic overview and enables the user of this application to identify accounts based on various metrics related to both, *Twitter* and *Destiny* in-game data.

2.3 Using location-based data

The next extensive field aims at leveraging location-based microblogging data. *Twitter* supports usage of GPS data, but due to privacy reasons it is disabled by default and the user has to manually activate adding the GPS location to a *tweet* in the account settings. Therefore only a small percentage of all *tweets* does include geo location data. Leetaru et al.[LWC+13], for example, revealed that in their data sample only 2.02% of all *tweets* contained location meta data, of which 1.8% included a so-called place indicator, 1.6% contained exact GPS coordinates and 1.4% had both.

As a result many studies focus on deriving locations based on a *tweets* content: Li and Sun [LS14] presented a solution called *PETAR* (**P**oint-of-Interest **E**xtractor with **T**emporal **A**wa**r**eness). At first, they built a POI inventory consisting of words and phrases related to a POI by leveraging crowd-sourced data from Foursquare[1] check-ins. In addition to that they developed a *time-aware* POI tagger which is able to extract POI names in *tweets* based on lexical, grammatical, geographical and BILOU scheme features. The latter is a scheme which "identifies **B**eginning, **I**nside and **L**ast word of a multi-word POI name, and **U**nit-length POI name", while "words that do not appear in any POI names are identified by the Outside label" [LS14]. Their approach showed promising results as well as performance and could be applied in real-time applications.

Moreover, Ferracani et al. [FPD14] presented a web application called *LiveCities* for unveiling city zones and its dynamics. They extracted location-based information from user's profiles on *Facebook*, consisting of status updates, posts, events or tagged photos, and classified them by using the Foursquare API. Venues were then clustered resulting in labeled regions which provide insights to city zones by either displaying the exact location of each venue including basic information and its category, or viewing colored clusters that highlight the differences between zones.

[1] Foursquare uses location intelligence to build meaningful consumer experiences and business solutions: https://foursquare.com/, last accessed: 2018-08-23

Abbasi et al. [ARMW15] conducted similar work, but focused less on venues and more on the traveling aspects of microblogging data: they argue that since many people tend to talk about their current activities or immanent plans on *Twitter*, it is possible to extract travel attributes such as purpose and location of a trip from microblogging data. Their presented framework initially identified residents and tourists by splitting their data set of *tweets* containing geo-location data into four samples on the basis of four time periods. Users are then labeled as residents if they appear in at least three samples and as tourists if they appear in only one or two samples. Next, they analyzed the content of all *tweets* by checking them against relevant word clusters and marking the *tweets* with activity tags if similarities were found. Although they were able to tag only about 20% of all *tweets*, this project showed the potential of using social media and microblogging data to get valuable insights that could be leveraged by transport planning, management and operations entities.

Krueger et al. [KSB+16] focused even more on movement reconstructed from *Twitter* data: they crawled *tweets* containing geo-location meta data, removed unrelated data like bots or weather reports and applied a movement classification to split data into pedestrian, ground transportation and flights by setting specific boundaries in regards to the geographic and temporal distance between two *tweets*. Edge splatting, normalization and color mapping is used to visualize the resulting trajectories. They also presented results of their case studies which compared movement data derived from *Twitter* to data sets such as movement data from the pilgrimage to Mecca, taxi traffic data of New York or global flight schedules, showing the great potential of comparing movement patterns restructured from *Twitter* data to different global and local movement data sets.

To summarize this section, leveraging social media and microblogging data that contains location-based data can provide fascinating insights. However, the usage of *Twitter's* geo-tagging features is sparse. This can also be confirmed by the data sample of this work: out of over one million *tweets*, only 15,107 *tweets* where geo-tagged. 2,947 different places have been extracted, with Los Angeles, the host city of the annual *Electronic Entertainment Expo* (E3)[2], being used most (329 times). Due to the sparse amount of geo-location data, this approach has not been pursued further in the course of the chosen use case *Destiny*.

2.4 Detecting events

Event detection is another widely studied field: As an example, Gao et al. [GCHL13] presented their approach in detecting geographical social events by combining geographical temporal pattern mining with content analysis of related *tweets*. At first, they identified unusual geographic locations by splitting the data of each day into 6-hour samples which are then compared to the same time periods of the previous day. If the *tweet* activity does show significant differences within a geographical region, this region is tagged as showing unusual activity or a region-of-interest (ROI). After that k-means clustering is

[2] https://www.e3expo.com/, last accessed: 2018-08-23

applied to the content of *tweets* from this region and time period and if the number of *tweets* within such a cluster surpasses a certain threshold, it is flagged as a social event taking place in this region. The researchers conducted an experiment based on a 2-month data sample collected from *Sina Weibo*[3], a very popular microblogging service in China. They applied their method which resulted in detecting 13 of the 20 events that took place in the Beijing area.

Furthermore, the work of Lanagan and Smeaton [LS11] focused less on detecting an event itself but more on finding highlights during an event because data of microblogging activities is richer in information and available faster than approaches that rely solely on audio and video analysis: they collected *tweets* related to multiple soccer and rugby matches by filtering hashtags, analyzed the activity numbers within multiple time segments and generated keywords based on the content of the *tweets*. They combined the results with a video shot-boundary detection algorithm, that detects highlights by processing audio and video of a stream, and could show that their approach appears to be very effective: all of the goals in soccer matches were correctly detected, 16 of 18 tries in rugby matches were found whereas the two missing occurred almost immediately after other significant events.

Another interesting approach in detecting highlights during events has been shown by Hsieh et al. [HLCH12]: they introduced a moving-threshold burst detection algorithm that analyzes *tweets* related to an event. After defining time periods and calculating the mean and standard deviation of the *tweet* activities within those time periods, they adjust the threshold based on those calculations while activities surpassing this threshold are marked as highlights. In the next step, the content of *tweets* from time periods flagged as highlights is analyzed by applying word stemming, stop-word elimination and calculating, and ranking frequency scores of remaining words in order to get the semantics of those *tweets*. They evaluated their method for multiple sports events: again, all goals in soccer games were detected correctly, while the performance for the nearly equivalent events like aces in tennis and dunks in basketball was poorer. The researchers think that the reason for that is that those events happen more often during a game compared to goals in soccer.

Kim et al. [KLK13] presented a more generic approach in analyzing microblogging activities not limited to events, but focused on hot topics in general. They collected *tweets* with attached geo-location data and calculated the word frequencies before categorizing the top keywords into 9 social topics such as weather, weekend or TV show. Fluctuations in activities related to those topics were observed and many topics like a same-sex marriage issue by a US court or a soccer World Cup qualification game between the US and Mexico have been detected. Additionally, the researchers applied geographic clustering methods to the sample related to the weather topic which enabled them to identify four communities matching the geographic weather conditions respectively.

A quite similar work was done by Klomklao et al. [KRP16]: they built a tool that collected

[3] https://weibo.com, last accessed: 2018-08-23

geo-tagged *tweets* from the *Twitter* API each day and visualized the top hashtags on an interactive world map. This enabled the user to view which hashtags are prominent each day in different countries, but also provided insights into the worldwide geographic distribution of hashtags due to the usage of different colors based on the ranking of hashtags on the country-level.

To sum up, event and highlight detection focuses on finding sharp increases in activity within social media and microblogging service activities. This is also part of this work and mostly the starting point for an analysis by inspecting the daily *tweet* activities in a timeline in order to find events that impact microblogging activities. Since interesting events and highlights can occur on a much smaller and local scale, they could currently be missed. Therefore this is definitely an aspect that could be addressed and enhanced in the future by implementing approaches as suggested by Gao et al. [GCHL13], or even the approach by Hsieh et al. [HLCH12] in case fine-grain highlight detection has to be achieved.

2.5 Crisis informatics

In regards to event detection, the field of crisis informatics is especially interested in using social media and microblogging services: Sakaki et al. [SOM13], for example, proposed an algorithm which basically transforms *Twitter* users into *social sensors*. They process *Twitter* data related to an event and analyze matching *tweets* by using probabilistic models for temporal and spatial detection which results in an approximated time and location for the event. To evaluate their algorithm as an application, the researchers present an earthquake reporting system called *Toretter* and chose Japan as target region, since numerous earthquakes happen in Japan each year and the density of *Twitter* users throughout Japan is very high. Users of *Toretter* can sign up to receive email notifications if an earthquake is detected. The researchers confirm that the system has been in operation since 2010 and reveal that email notifications were sent within one and a half minutes on average, which is a lot faster than the official announcements of the Japan Meteorological Agency (JMA). They also note that setting the threshold for the amount of positive classified *tweets* is challenging and a trade off between the detection rate and precision of the system: a low amount of positive *tweets* led to an impressive 93% detection rate of earthquakes stronger than seismic intensity scale 3 of JMA but also resulted in a lot of false-positives, whereas the detection rate decreases when the threshold of positive *tweets* is increased.

In a broader sense, Palen et al. [PAM+10] presented their vision on how the future of emergency management could look like in regards to information and communication technology (ICT) in general, as well as social media and microblogging services specifically. This vision has been and continues to be realized in the course of a project called EPIC (Empowering the Public with Information in Crisis)[4]. At its core, EPIC views "citizenry

[4]EPIC is funded by the US National Science Foundation. Since its launch in 2009 numerous studies and research projects have been published: http://epic.cs.colorado.edu, last accessed: 2018-08-23

as a powerful, self-organizing, and collectively intelligent force"[PAM+10] and aims to leverage this force in order to enhance information flow for both, citizens affected by disasters, as well as crisis management and emergency response entities. They have split the research program in five main topics: it is of high importance to understand i) what kind of data is posted by social media and microblogging users during emergencies, how users react to that content and how accuracy and trustworthiness is assessed. Furthermore, ii) the vast amounts of unstructured data has to be processed in order to gain insights, iii) new information extraction and natural language processing strategies need to be investigated and developed which are fit for handling the diverse and swift nature of real-time communication, as well as iv) considering privacy and security related issues that emerge when handling data from unknown sources or geo-location information from users. Finally, v) various legal and policy-related issues might emerge from results of this vision.

Similar to that vision, MaxEachren et al. [MJR+11] presented *SensePlace2*, an web-based crisis management tool and ongoing research project focused on extracting and visualizing information from social media and microblogging services in order to enhance the understanding of crisis-related information. It combines filtering of *tweet* content based on geographical, temporal and thematic information with visualizations such as heat maps, word clouds and *tweet* lists in order to provide overview as well as detailed views about various situations. They evaluated mockups of their work in the course of a structured survey with crisis management professionals which revealed not only how the status quo in crisis management looks like but also that there is an openness among professionals to include social media into their work processes.

Vieweg et al. [VHSP10] also worked towards enhancing situational awareness and analyzed *Twitter* data collected during the Red River floods and Oklahoma grass fires which ocurred in spring 2009. After filtering and cleaning the *tweet* data, categorization and coding methods were applied in regards to geo-location and information that references a location, situational update categories such as warning, flood level, road conditions, evacuation information or damage/injury reports, and additional *tweet* characteristics like so-called *high-yield twitter users*. These are users that appear to carefully craft *tweets* in order to maximize the contained information within *Twitter's* character limit. The analysis of those two data samples showed that there is potential for the development of frameworks which leverage the identified features of information generated during disasters in order to enhance situational awareness.

The field of crisis informatics involves a lot of research projects that aim at understanding which role social media and microblogging services can play during natural or man-made disasters, and leveraging these insights to implement applications that support a broad range of people and institutions such as first responders, emergency relief units and crisis management professionals, governmental and non-governmental organizations and citizens themselves. In regards to this work, crisis informatics shows another possibility for an application since the functionality of extracting and visualizing information in multiple ways exist and could be extended to match the requirements of crisis management

organizations in the future.

2.6 Tweets during events

Not only crisis informatics is interested in studies revolving around tweeting during events. Shamma et al. [SKC09], for example, analyzed *Twitter* usage during the 2008 presidential debate in the US featuring Barack Obama and John McCain. Over a time period of 150 minutes, of which the first 97 minutes were the actual debate, the researchers crawled all *tweets* using related hashtags. They analyzed the *Twitter* activities, which increased towards the end of the debate, and mapped them to multiple segments representing the discussed topics during the debate which resulted in an overview of local maxima and minima. In addition to that, they constructed network graphs of users and their activities with a special focus on *Twitter's* mentions feature, which can be used as a response as well as a *call-out* to another user, giving insights into how users are connected and which user accounts are of higher importance. Unsurprisingly, the three top accounts were the official *Twitter* accounts of Barack Obama, McCain and the official *NewsHour* account, which hosted the debate. Finally, the top keywords were calculated for each topical segment which showed some contextual similarities but revealed that the vocabulary in the debate was different to the one used on *Twitter*, and that *tweets* often were not topic-related discussions, but more reactionary like posting scores about which topics have been won by which candidate.

Han et al. [HHK17] recently conducted a study revolving around *tweet* activities during sports events: they collected *tweets* related to the 2013 *Super Bowl* which were posted during the time period the game took place. *Tweets* were then grouped into five-minute intervals in order to get an activity overview, which then was mapped to actual, distinctive situations that happened during the game. A power outage during the third quarter of the game, for example, was such a special event that led to a significant increase in *Twitter* activity. By analyzing the content of *tweets* during this time period the researchers found out that 63.8% of these *tweets* were indeed related to the blackout.

A broader approach to this topic was done by Buschow et al. [BSU14]: they collected *tweets* related to different German TV-shows that were posted during the broadcast times in order to find out how *tweet* activities during TV-shows look like and how different types of TV programs influence those activities. *Tweets* were collected based on hashtags related to broadcasts such as *#dsds* for *Deutschland sucht den Superstar*, the German equivalent to *American* or *Pop Idol*, before cluster algorithms were applied resulting in categories such as emotions or evaluation of shows and actors. As a result, the researchers identified engaging with the program and interacting with the community as the two main motivations of *Twitter* usage during TV programs. They also showed that different TV shows lead to different *tweet* activities: while shows like *The Voice of Germany* or *Deutschland sucht den Superstar* initiated more *tweets* with an evaluating character, political talk shows led to a higher interaction between *Twitter* users, as well as starting discourses based on the topics of the show.

Similar work has been done by Wohn and Na [WN11], who collected *tweets* posted during Barack Obama's acceptance speech of the *Nobel Peace Prize* and an episode of ABC's *So You Think You Can Dance* in October 2009. The content of those *tweets* was analyzed and each *tweet* was assigned to one of the four categories attention, emotion, information and opinion. The results showed that the majority of microblogging activities are related to actual situations, but also that people are tweeting more during commercial breaks. The researchers believed that this is caused by the *storyline* becoming more dramatic before a commercial break or that viewers were bored during the break and used *Twitter* to pass time until the program resumed.

To summarize, the focus of this section was on analyzing *tweets* that have been posted during an event in order to understand if and what people on social media and microblogging services write about an ongoing event and how they react to things that happen during the event. For this work and especially for its use case, *tweets* collected during events such as a livestream of an influential user or the game developer itself, or presentations during the aforementioned E3, can provide interesting insights into the reaction of viewers containing instant feedback from the community.

2.7 Sentiment analysis

In regards to analyzing the content of social media and microblogging activities the analysis of the sentiment has been of high interest to the research community: Araújo et al. [AGCB14], for example, presented a web tool called *iFeel*, which takes a look at 8 different sentiment analysis tools such as *SentiWordNet*[5], *SenticNet*[6], *SentiStrength*[7] or *Sentiment140*[8]. The tool enables the user to enter a sentence or upload a text file which is then analyzed by all supported sentiment analysis tools, resulting in a great overview of tools available as well as how they perform.

As a practical example of applying sentiment analysis, Yu and Wang [YW15] analyzed the sentiment of *tweets* that were posted by U.S. sports fans during five games of the *2014 FIFA World Cup*. At first, the collected data was cleaned by removing hashtags, usernames or URLs, then the *tweets* were tokenized, converted to lowercase, and cleaned of stop words, before stemming and lemmatization was applied. Finally, they determined the sentiment of the resulting word lists based on an enhanced word-emotion association lexicon in combination with the frequency the words appeared within *tweets*. The researchers could validate their hypothesis by showing that the involvement of the U.S. team in games led to negative sentiments such as anger and fear when the U.S. team conceded a goal, while negative sentiments decreased and positive sentiments such as joy and anticipation increased when the U.S. team scored a goal or showed positive signs. Games without involvement of the U.S. team mainly resulted in joyful sentiments. The

[5]http://sentiwordnet.isti.cnr.it, last accessed: 2018-08-23
[6]http://sentic.net, last accessed: 2018-08-23
[7]http://sentistrength.wlv.ac.uk, last accessed: 2018-08-23
[8]http://www.sentiment140.com, last accessed: 2018-08-23

researchers therefore argued, that being fan of a team enhances negative and positive sentiments alike based on the performance of the supported team, whereas being fan, not necessarily of one of the teams, but of the sport itself, mainly resulted in enjoyment.

Hoeber et al. [HHW+13] proposed a tool called *Visual Twitter Analytics* (*VISTA*) which focuses on extracting data related to a topic, analyzes the sentiment of the collected *tweets* by using the *Sentiment140* [GBH09a] service, before visualizing an interactive timeline of the microblogging activity color-coded based on its sentiment. Additionally, they presented a case study conducted during *Le Tour de France* talking place from June 29 to July 21, 2013. The tool enabled users to easily find distinctive moments on the overall timeline such as an increase in activity during a race day's finish, but also to conduct deeper analysis by zooming into smaller temporal ranges and inspecting *tweets* themselves.

Sentiment analysis faces multiple challenges due to the fast-paced and multivariate nature of microblogging data. As Araújo et al. [AGCB14] showed, there are a lot of sentiment analysis projects out there. Sentiment analysis is also a core component of the application of this work and various visualizations are based on the sentiment of *tweets*. *Sentiment140* was chosen as analysis tool due to its focus on *Twitter* and because it is easy to integrate into an application as it supports batch requests that can process a multitude of *tweets* at once.

2.8 Visualizations & Analysis tools

There have been a lot of research projects with similar approaches: Kaye et al. [KLJ+12], for example, presented *Nokia Internet Pulse*, a corporate system aiming at customer feedback that analyzes *tweets* related to *Nokia* or a *Nokia* product before displaying a color-coded word list based on the *tweets* sentiment. Castellanos et al. [CGL+11] also focused on customer support and proposed a tool called *LivePulse*, which also uses a combination of sentiment analysis and color-coding to display wordclouds and real-time activity charts. As a demo, they used microblogging data published about products of *HP*. Chen et al. [CCCJ15] proposed a cross-media sentiment analysis system which allows the user to view the resulting data organized in regions, topics or by content, providing insights into what kind of sentiment is prevailing in which region. MacDonald and Moffat [MM16] analyzed and compared the sentiments of *tweets* related to the theme of each *Global Game Jam*[9] since 2010 in order to gain insights into how people on social media and microblogging services feel about the topic. Chatzakou et al. [CKB+17] conducted a study about the *Gamergate* controversy[10], revealing that *Gamergate* participants tend to

[9] The Global Game Jam is the largest collaborative meetup of game enthusiasts which takes place yearly at multiple physical locations around the world. It specifies a theme and participants create games related to this theme over the following 48 hours. https://globalgamejam.org, last accessed: 2018-08-22

[10] The Gamergate controversy can be seen as an cultural war concerning issues of sexism and progressivism and originated in August 2014 when anonymous social media users started harassment campaigns targeted at female game developers and female media critics that even included rape and death threats: https://en.wikipedia.org/wiki/Gamergate_controversy, last accessed: 2018-08-22

post more *tweets* that are classified as negative and that they seem to be less joyful than regular users.

Furthermore, Morstatter et al. [MKLM13] presented *TweetXplorer*, a visual analytics tool that allows analysts to quickly get an overview as well as deeper understanding of an event without before-hand knowledge by combining a broad range of visualization techniques such as tag clouds, heat maps, calendars, and trees. As a use case they focused on *tweets* related to the Hurricane Sandy in 2012, revealing interesting insights about the communication and information before, during and after the hurricane hit the U.S., ranging from information about pet shelter locations in evacuation areas and rumors to reports about damages, power outages and floodings. While *TweetXplorer* aimed at gaining a general understanding of an event, *TopicFlow*, an application developed by Malik et al. [MSH+13], focused on topic detection and visualizing trends resulting in interactive graphs that provide great insights into how topics are related to each other and evolve over time. Kraker et al. [KWJL11] also presented a system for trend detection which aims at extracting data that is most important within the *Twitter* stream. Based on that data which can be crawled based on a specific set of keywords, a list of user accounts or a combination of both, they visualized two different graphs in order to reveal the temporal evolution as well as the relation between different topics. Dewan et al. [DGGK13] combined many of the aforementioned features and proposed a framework called *MultiOSN*: data related to a specified event is crawled from five social media and microblogging services (*Facebook*, *Twitter*, *YouTube*, *Google+*, and *Flickr*), various metrics are analyzed such as activity numbers including the amount of URLs within the data set, tag clouds and sentiment analysis, as well as geographical analysis, before results are presented in a dashboard and detailed analysis pages. Due to these analytical features and the possibility to switch between multiple events the researchers argue that *MultiOSN* is especially useful for people and organizations related to law and order. *Westeros Sentinel*, developed by Scharl et al. [SHHJ+16], is another example for combining multiple analysis and visualization types: as a case study they collected data from news websites and four social media networks (*Twitter*, *Facebook*, *Google+*, and *YouTube*) which was related to the fourth season of the *Game of Thrones* TV show and applied a broad range of analytical processes such as named entity recognition, sentiment extraction, or trend detection. The results were presented in a dashboard consisting of trend charts, maps, word clouds, trees, which enabled the user to gain interesting insights such as the perception of new story elements, characters and actors by social media and microblogging service users.

This section gave a small glimpse into the broad range of research projects based on social media and microblogging data with many of them containing similarities to aspects of the application presented in this work such as color-coded word clouds, activity charts or the usage of sentiment analysis. They also are relevant regarding future work discussed in Section 7.3: like *MultiOSN* [KWJL11] or *Westeros Sentinel* [SHHJ+16] the application could benefit greatly from extending the data collection process in order to support more social media networks and microblogging services such as *Facebook*, *Google+*, or *YouTube*,

or adding more complex visualization types as proposed by *TweetXplorer* [MKLM13] or *TopicFlow* [MSH+13].

2.9 (Electronic) Word-of-Mouth

After reviewing a multitude of literature directly related to mining, analyzing and visualizing data from social media and microblogging services, subsequently the remaining paragraphs of this chapter take a look at other, more abstract aspects related to this work. The effects of electronic word-of-mouth (eWOM), for example, was consistently studied by Rosario et al. [BSDB16]: first, they collected studies that revolve around the impact word-of-mouth has on sales performance, analyzed the effect sizes defined and used by these studies, and developed a coding protocol based on those effect sizes. After that, the researchers identified more than 40 online platforms consisting of e-commerce, social media, news and review websites, as well as 26 product categories such as books, movies or digital cameras. Next, they collected historical data of the aforementioned online platforms by using the *Wayback Machine*[11], before computing the effect sizes based on correlations between eWOM metrics and sales. The researchers were able to present multiple findings such as eWOM generally having a stronger impact on sales when displayed on e-commerce websites than being part of social media activities while the sales of services are more impacted by social media activities, homophily details being more important than the trustworthiness of the poster, or eWOM having a stronger link to sales of new products.

While the approach of Rosario et. al. [BSDB16] aimed at finding a general, all-purpose framework, Hennig-Thurau et al. [HTWF15] focused on the effects that microblogging word-of-mouth (*MWOD*) has on early adopters: they collected *tweets*, which were posted within the first 24 hours after the opening of 105 movies, filtered spam and non-evaluative *tweets* before splitting the data set in *tweets* with positive and negative sentiment. After that, the researchers analyzed if *tweets* with evaluative character had an impact on the box office revenue of the following weekend (Saturday and Sunday). They could show, that although positive *tweets* did not result in an increased revenue, negative *tweets* were indeed followed by a detectable decrease in revenue. The researchers then conducted a survey with customers who are active on *Twitter* and decided to not watch a movie after reading negative *tweets* about it, which shed light into the movie decision process. For example, 67% of all survey participants agreed that negative *tweets* have a higher influence on the decision because they stand out from the positive marketing information available. Negative *tweets* are also seen as more honest than trailers and reviews, which both are described as generally having a positive bias. Furthermore, 44% acknowledged that reading a very negative *tweet* indicating that the movie is not worth the time and money, led to the decision to not see the movie, 26% joined a discussion about the movie

[11] The Internet Archive: Wayback Machine is a non-profit service that stores snapshots of websites as they appeared at a certain date. Over 306 billion web pages have been saved over time and can be viewed on https://archive.org/web, last accessed: 2018-08-22

on *Twitter*, 31% started searching for an alternative movie, while the rest decided to watch another movie at home or at the cinema.

Deloitte [Del13], on the other hand, could prove an impact of positive classified word-of-mouth *tweets* on sales as well when conducting a study based on three data sets: sales data of the 100 bestselling video games for Xbox 360 and PlayStation 3 in the UK in 2012, advertising data related to those games, and *Twitter* data from a time period of 10 weeks prior and 20 weeks after release of each game. Then, Deloitte developed a state-space hierarchical Bayesian model, which could handle the significant differences of sales and advertising data for each game, as well as effectively estimate the predictive impact of *Twitter* data. They revealed that *tweets* had a significant impact on sales in the UK, whereas the influence of positive *tweets* is generally stronger than that of negative *tweets*, and also that higher numbers of positive *tweets* would have more impact on sales numbers than a similar increase in advertising.

To summarize, studies revolving around (electronic) word-of-mouth could link social media and microblogging activities to an increase or decrease in sales of an entertainment product. Therefore, it is of great importance to developers or service providers to understand the audience of their products and services. The application presented in this work can be helpful by, for example, identifying and reacting to problems people voice on *Twitter*, or using the amplifying effect of influential users from the community to spread information.

2.10 Social media and gaming

According to Bateman [Bat16], gamers are becoming one of the most active and engaged group of users on *Twitter*, which puts gaming communities more into the literature spotlight: Seay et al. [SJLK04] conducted a survey among players of Massively Multiplayer Online Games (MMOGs) in order to gain insights into social experiences inside, as well as outside of games with a focus on communication: 78% of the survey participants were members of guilds, which are collaborative groups of players who unite in order to progress in the gaming world together. 69% declared to use communication tools such as online discussion boards, guild websites, instant messenger programs, or email outside of games to connect with fellow gamers, with coordination and scheduling of events, providing help and giving advice being the primary use cases. The researchers furthermore argue that communication has an impact on the commitment of members towards their guild and that game designers and developers should provide or simplify access to communication tools.

Ducheneaut et al. [DYNM06, DYNM07] approached the subject of player communities from a different angle when they extensively studied the MMOG *World of Warcraft* (WoW). All authors spent a large amount of time as players in the game leveling up their characters and joining guilds. Besides gaining immense amounts of qualitative knowledge in WoW, the authors also logged data by periodically using an in-game command which outputs basic data of each character that is currently active on a server. They extracted

three variables from this data: i) the zone information, which basically is the current region within the gaming world a player is active in, ii) the grouped flag, which indicates if a player is currently alone or grouped together with other players, and iii) the name of the guild the current player is a member of. Based on this, the researchers could not only monitor the progress of over 300,000 unique characters, but also model the structures and group dynamics of guilds, revealing interesting insights such as that guilds are more successful when splitting its members into small sub-groups of similar character levels in order to progress faster. Since out-of-box tools to simplify management and communication of large-scale communities in-game are sparse in most MMOGs and guilds usually rely on external tools such as message boards, the authors developed the so-called *Social Dashboard*. This tool monitors the aforementioned variables and provides an overview of the current state of a guild including certain thresholds. For example, guilds are marked as dangerous or critical if the size of a guild declines over time to a point it might not survive, or if the number of sub-groups of characters within a guild drops, which implies that guild members have problems forming groups due to gaps in character classes or levels, or general incompatibilities in scheduling.

By comparison, Chung et al. [CHC+14] focused more on the social interactions when analyzing an extensive data set of logged actions by players of the MMOG *Aion*, which was provided by its developer *NCsoft*. The researchers studied the usage of all social interaction possibilities available in-game such as direct or group chats, trading or friend requests, and their impact on group dynamics. They presented interesting differences regarding communication and economics between groups with growing, stable, and declining member numbers: communication of growing groups, for example, tends to be more balanced and cohesive than groups with declining numbers, while groups that fail to communicate actively and evenly across all members, are bound to lose members.

Getting back to social media and microblogging services, Hicks et al. [HGK+15] explored possibilities to use social media networks as a game platform. They conducted two studies based on the game *Hashtag Dungeon*, a game designed to use *Twitter* for creating, storing and promoting the game. The game is designed in a way that players collaboratively create levels and share those levels via *Twitter*, which then can be played or further extended by other players. The first study consisted of a questionnaire focusing on the playability of the game itself and the experience in relation to the usage of social media: while participants of this questionnaire overall agree that the basic gameplay mechanics are good, the social media mechanics of the game to interact directly with other players have not been used much. The second, follow-up study therefore aimed at understanding the player perspective related to the games' *Twitter* integration. From the interview data the researchers identified four recurring themes: i) the negative aspects of the *Twitter* integration was the biggest theme with concerns from spamming the users own *Twitter* timeline with posts containing similar or the same content multiple times within a short period of time, followed by ii) social interaction, which highlighted the reasons for using *Twitter* such as scrolling through their timeline, posting, and re-tweeting *tweets*, promoting themselves, but also an interest and curiosity related to the cryptic nature

of *tweets* sent by *Hashtag Dungeon* leading to discussions about the game. Next, iii) positive aspects of the integration were mentioned: the possibility to choose how to send *tweets* was praised, and an option to authorize the game to automatically and seamlessly send *tweets* was seen as helpful. Finally, the participants talked about features related to iv) encouraging participation such as a possibility to manually add "human input" to the *tweets* generated by the game in order to mitigate the spam-like nature of those *tweets*, while highlighting the collaborative aspects of *Hashtag Dungeon*. Based on those results, the researchers proposed strategies for game developers encouraging them to limit the usage of *tweets* to meaningful posts within a reasonable time period, allowing users to enrich posts with personal content, and implementing or enhancing tools that allow players to easily share aspects of the game with a personal meaning to them.

To sum up, this section highlights various research projects revolving around the social nature of gaming communities focusing on collaborative aspects and communication within groups of gamers. As Bateman [Bat16] stated, gamers are becoming the most active and engaged user group on *Twitter*, and they are therefore the beneficial choice for the use case of this work.

2.11 User feedback

Finally, there have been some research projects related to social media as source of user feedback: Bajic and Lyons [BL11] analyzed *UserVoice*[12] feedback data of 46 software companies and conducted an interview study with 20 companies of different size ranging from start-ups to multinational corporations, in order to identify how much of a role social media plays for gathering user feedback based on the four factors company size, transparency, software deployment and amount of social media tools in use. They could validate that small companies and start-ups relied more on social media and microblogging services for user feedback than bigger companies, while transparency and the amount of social media tools in use did not have a significant impact on the use of social media for user feedback. Regarding the type of software deployment, they found out that Software-as-a-Service (SaaS) vendors use social media more for receiving feedback and gaining an edge against competitors while vendors, who deploy stand-alone solutions, tend to use social media to prioritize development of features and facilitate traditional feedback gathering systems over social media.

That users can have an impact and even force a developer to react to problems and changes has been shown by Jordan et al. [JBSR16], who conducted a narrative case study based on the MMOG *KingsRoad*. They played the game themselves in order to gain an understanding about the gameplay mechanics, and they joined the official game forum as a passive observer monitoring the reaction of players to significant gameplay changes. The presented five cases that reveal commercially-driven changes that negatively impact the experience of the players who voice their frustration and anger in the forums, form

[12] UserVoice is a developer of product feedback management software: https://www.uservoice.com, last accessed: 2018-08-22

petitions and protests, and even threaten to stop playing the game: to give one example, the developer introduced new equipment slots for the virtual character and started a three-week competition with the possibility for top-ranked players to win an exclusive belt available only in this competition. Since the game focused on gaining and leveling up characters by killing monsters, gathering better items, gold and gems, and a premium currency, which can also be bought with real money through micro-transactions, many players spent a lot of time and money in order to be ranked at the top of the weekly leaderboards. But, on the last day of the tournament the developer made the rarest belt available to buy with premium currency in their in-game shop causing a furious reaction by players like a top ranked one, who claimed to have barely slept and eaten, and to have invested around 190.000 gems, while the price of the as unique and competition-only marketed belt was first at 5.399 gems, then even dropped to 3.849 gems in the in-game shop. The developer reacted by apologizing multiple times, removing the item from its in-game shop and giving free upgrades to the top 3 ranked players and after some more voiced anger to all players ranked in the top 100. Overall, the authors revealed a lack of communication from the developer towards the players, trivializing the resources, such as time and money, spent by players, and leaning towards enhancing the experience of players, who spent real money, justifying game changes by market logic.

Regarding this work, user feedback is one major objective of this application: as will be shown in following chapters in more detail, people tend to voice their feelings on social media and microblogging services, expressing enjoyment about news and things they have experienced, but also frustration about non-working products and problems. For developers and service providers it is therefore essential to have an open ear about what their target audience is talking about on social media and microblogging services not only to gather input in order to improve their products and services, but also to react to user feedback faster and establish a customer-friendly presence which can benefit each business greatly.

CHAPTER 3

Application system overview

This chapter takes a deeper look into the application of this work, which is illustrated in Figure 3.1 and basically consists of four components: i) the crawler component collects data from the *Twitter* Search API and saves the results to the storage, ii) the visualizer component requests data from the storage and presents it by using various interactive visualizations, iii) the importer component imports data from a file and saves it to the storage, while iv) the export component requests data from the storage and saves it to a CSV file. As shown in the application system overview all components are operated by a client and have access to the storage which is used to either request or store data. Before diving into the details of each component, the development process and the used frameworks and libraries will be discussed in the following section.

3.1 Process, frameworks and libraries

The application of this work is based on Node.js[1], an efficient and lightweight open source server framework which enables the execution of JavaScript code directly on a web server. In addition to that, the JavaScript package manager NPM[2] was utilized to access and use various open source packages in order to accelerate the development process. A git[3] repository hosted on Bitbucket[4] was used as a version control system, while the application was set up and executed in a Debian-powered virtual machine on a local computer. The development of the application started by implementing the basics of the *Twitter* crawler, before setting up the storage component in form of a MySQL database, and adding rudimentary CRUD[5] functions in order to save the results of the

[1] https://nodejs.org/en, last accessed: 2018-08-23
[2] https://www.npmjs.com, last accessed: 2018-08-23
[3] https://git-scm.com, last accessed: 2018-08-23
[4] https://bitbucket.org, last accessed: 2018-08-23
[5] CRUD is an acronym for the basic SQL commands Create, Read, Update, Delete

crawler to the database. After the crawler was up and running, the focus of development switched to the visualizer component, and a variety of visualization types was added and continuously extended. In between these optimizations and extensions, the import and export components were added in order to provide functionality relevant to the use case such as exporting a user data sample with gamertag information as a CSV file, or importing an extensive set of in-game data. Finally, the visualizer component was extended in order to support the newly added data in relation to the collected *Twitter* data. The following sections will give an overview of all used frameworks and libraries in the order of its integration respectively.

3.1.1 NodeBootstrap Framework

NodeBootstrap[6] was used as core of the application because it provides a basic structure for any Node.js project, takes care of basics like startup scripts, configuration and testing, and also includes a broad range of modules and libraries out-of-the-box such

[6]http://nodebootstrap.io, last accessed: 2018-08-23

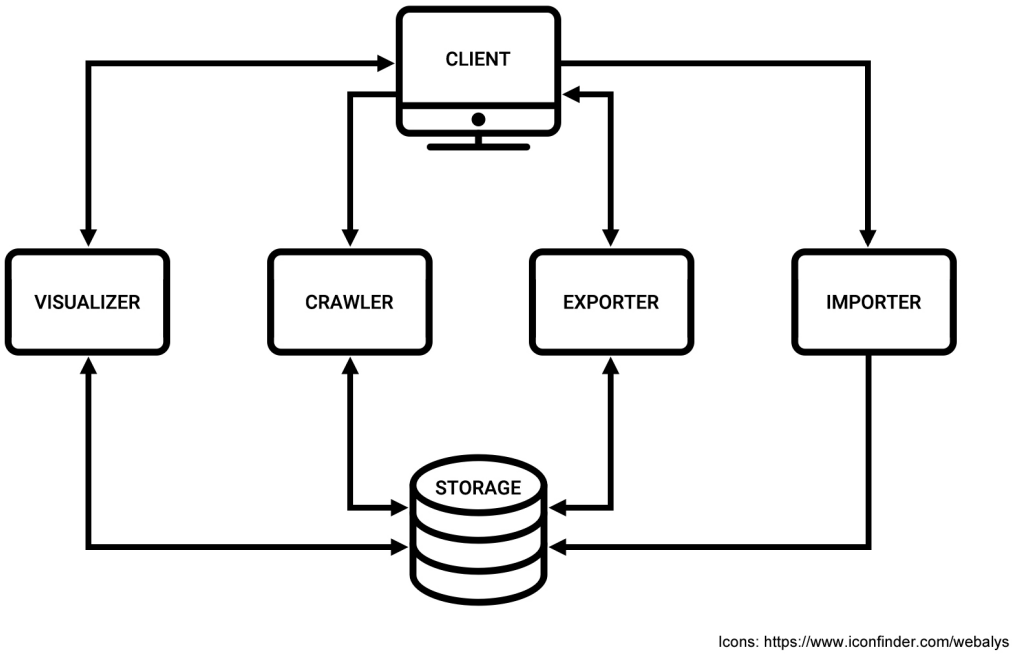


Figure 3.1: This system overview illustrates structure, request and data flow of the application which consists of the four components: crawler, visualizer, importer and exporter

as Handlebars[7] as a templating engine, which is a very fast way to render views in a browser, the feature-rich JavaScript library jQuery[8], or the front-end component library Bootstrap[9].

3.1.2 Twit Package

Next, twit[10], a Node.js wrapper for *Twitter's* Search and Streaming API, was added to the project. twit simplifies the whole API request process by handling the authentication process with *Twitter*, constructing and sending queries with the correct formatting based on configurable arguments, and receiving results from *Twitter* API requests.

3.1.3 MySQL Package and Database

After testing and setting up a basic *Twitter* API request for getting *tweets* based on search terms, the MySQL[11] package, which is a Node.js client that, similar to what twit is doing in regards to the *Twitter* API, exposes the MySQL protocol in order to simplify communication to a MySQL database. At the same time the database scheme was specified fitting the multivariate nature of *Twitter* data. *Tweet* data was logically split up and saved to the following four database tables: i) the *user* table holds basic user account data such as id, name, biography, or followers and friends count and was extended later on to also contain game-related user data such as XBL gamertags or PSN IDs. ii) The *profile* table contains additional customization account data that can been set by the *Twitter* user such as profile and background image URLs or colors. iii) The *tweet* table contains all *tweet* related data such as id and text of the *tweet*, re-tweet and favorite counts, but also the user id as a relation to the user account, or the analyzed sentiment value. iv) Finally, the *place* table holds data of all *Twitter* place objects that have been attached to *tweets*, such as country, name or geo-location data in the form of a bounding box.

3.1.4 Async Module

Next, the process of splitting up *tweet* data into the aforementioned structures was implemented with the help of the utility module async[12], which was used in order to structure and better control the flow of API calls, processing results and running SQL queries to request, save or update data. The result of this implementation phase was the crawler component which will be explained later in this chapter, with the exception of the sentiment analysis, which was included in the crawling process at a later stage.

[7] http://handlebarsjs.com, last accessed: 2018-08-23
[8] https://jquery.com, last accessed: 2018-08-23
[9] http://getbootstrap.com, last accessed: 2018-08-23
[10] https://github.com/ttezel/twit, last accessed: 2018-08-23
[11] https://github.com/mysqljs/mysql, last accessed: 2018-08-23
[12] http://caolan.github.io/async, last accessed: 2018-08-23

3.1.5 Contextmenu Plugin

After the crawler was up and running, focus switched to the *Twitter* data analysis process, especially identifying gamertags of *Twitter* users as part of the use case of this work, since a list of XBL gamertags or PSN IDs was required in order to receive in-game data related to users in the *Twitter* data sample. Detecting gamertags in the biography of a user proofed to be tricky since the *Twitter* biography contains text freely written by users and gamertags do not follow a certain pattern that can be detected fully automatically. Inspecting the biography of multiple user accounts containing gamertags revealed that most users tend to add acronyms and words such as "gamertag", "GT", "XBL" or "PSN" next to the gamertag in order to help others identify the gamertag when reading the biography. Therefore, user accounts that did not contain those terms and accounts that had published only one topic-related *tweet* were dismissed.

Example for the biography of a *Twitter* user account:

N7 Pathfinder. Year 1 Guardian. Tamriel adventurer. Metalhead and music nerd. A member of @theQueensCors PSN: strongwiccan Flawless: 2-6-17.

The remaining biographies were manually checked and gamertags extracted with the help of the jQuery plugin contextmenu[13].

3.1.6 Sentiment140 and the Request Module

After extracting gamertags, further work on analyzing *Twitter* data has been done, starting with sentiment analysis which was implemented by using the *Sentiment140* tool and the request client[14] which is a simple way to make HTTP requests. *Sentiment140* provides a bulk classification service[15] which processes thousands of *tweets* per request and returns the calculated sentiment value. Since there is a timeout window of 60 seconds, the authors of *Sentiment140* suggest a limit of 5000 *tweets* per request, which is honored by this application. Furthermore, in order to not flood the *Sentiment140* service during the crawling process, the sentiment analysis with Sentiment140 was kept as a separate process that after starting automatically analyzed all *tweets* that had not been analyzed.

3.1.7 Highcharts

With sentiment analysis added to the application the focus switched to implementing the visualizations. Research into JavaScript visualization libraries revealed a broad range of interesting solutions such as *D3.js*[16], *vis.js*[17], *Chart.js*[18] or *Highcharts*[19]. Finally,

[13] https://github.com/joewalnes/jquery-simple-context-menu, last accessed: 2018-08-23
[14] https://github.com/request/request, last accessed: 2018-08-23
[15] http://help.sentiment140.com/api, last accessed: 2018-08-23
[16] https://d3js.org, last accessed: 2018-08-23
[17] http://visjs.org, last accessed: 2018-08-23
[18] http://www.chartjs.org, last accessed: 2018-08-23
[19] https://www.highcharts.com, last accessed: 2018-08-23

Highcharts was chosen because it provides a vast amount of features out-of-the-box with a simple and minimal configuration in comparison to the other three libraries. It proofed to be a great choice since it was set up swiftly and running, while the documentation on the libraries website contained extensive examples for all kinds of visualization types.

3.1.8 Word Cloud and the Wordfreq Module

After implementing basic visualizations that gave insights into the *tweet* activity over time, a word cloud visualization was added in order to gain an understanding of the cause of spikes or sharp declines in the *tweet* activities. The word cloud had to be calculated on-the-fly due to the possibility to define a time period and filter *tweets* by sentiment. *wordcloud2.js*[20] was used in combination with the *wordfreq*[21] module, a simple library for text corpus calculation, which was slightly modified to better handle URLs.

3.1.9 Salient

In addition to the sentiment analysis provided by *Sentiment140* the *salient*[22] toolkit was added to the application in order to not only have a second sentiment analysis value for each *tweet*, but also have a more autonomous solution that does not require requests to an external service. Another benefit of using *salient* was that the sentiment values were not strictly negative, neutral or positive but float values, in the collected *tweet* data sample ranging from -19 to 18.6667, which made it possible to customize the thresholds of the sentiment values that belong to negative, neutral or positive *tweets*. After setting up *salient* and analyzing all collected *tweets*, the crawler component was extended to automatically analyze the sentiment of newly crawled *tweets*.

3.1.10 DataTables

Next, the *jQuery* plugin *DataTables*[23] was added to the visualizer component in order to present interactive tables such as influencers data or the results of the visualizer's search functionality. The data in these tables can be filtered by search terms and sorted by the column's content.

3.1.11 JSONStream and moment.js

Finally, the importer and exporter components were implemented. In order to handle big data files, the *JSONStream*[24] module was used to import massive amounts of data by streaming the files contents, while the exporter wrote the results of queries into CSV files. Furthermore, the *moment.js*[25] was used to enhance the handling dates and times.

[20] https://timdream.org/wordcloud2.js, last accessed: 2018-08-23
[21] https://github.com/timdream/wordfreq, last accessed: 2018-08-23
[22] https://github.com/nyxtom/salient, last accessed: 2018-08-23
[23] https://datatables.net, last accessed: 2018-08-23
[24] https://github.com/dominictarr/JSONStream, last accessed: 2018-08-23
[25] https://momentjs.com, last accessed: 2018-08-23

After importing the use case relevant in-game data sample, the visualizer component was extended and more visualization types were added in order to support the additional data in relation to the existing *Twitter* data.

To summarize, this section described the development process which started off by implementing the crawler component including various analysis methods such as sentiment analysis with two different tools, or splitting up the multivariate *Twitter* data into predefined, easily processable structures, followed by the visualizer component, the import and the export component. It also showed what kind of frameworks, modules and libraries were utilized when developing the application of this work. The remaining sections of this chapters will take a more detailed look into each of the aforementioned components.

3.2 Crawler

As illustrated in Figure 3.2, the crawler component basically requests data from the *Twitter* Search API endpoint, analyzes the results, saves data to the database and restarts the process by building the next query. It essentially consists of six parts of which five parts are related to the crawling process itself and the sixth part is an extension for the *Sentiment140* service. The following sections take a closer look at these parts.

3.2.1 Initializer

As its name suggests, the initializer, which is triggered by a client, is the entry point for the crawling process and handles the essential task of setting the initial parameters required for the subsequent steps: these are, on the one side, the *Twitter* rate limits which are pulled from the *Twitter* `application/rate_limit_status`[26] endpoint, and on the other side, the default query parameters consisting of the search query term and the most recent *tweet* ID, which is one element in handling ID ranges in *Twitter*.

To elaborate a little bit further, many Search API endpoints, including the *Twitter* search[27] endpoint used by the crawler component, support the optional parameters `since_id` and `max_id`: while `since_id` sets the limit for how old returned *tweets* are allowed to be, the latter causes queries to return results only older than the set ID. Since the *Twitter* Search API returns a maximum of 100 *tweets* per query, utilizing `since_id` and `max_id` is essential when collecting historically gapless *Twitter* data, because without them, the *Twitter* API simply returns the 100 most recent *tweets* on every request.

Therefore, the initializer pulls the highest *tweet* ID from the storage in order to set the bottom threshold, with the first, initial query of a data set being an exception because

[26] https://developer.twitter.com/en/docs/developer-utilities/rate-limit-status/api-reference/get-application-rate_limit_status, last accessed: 2018-08-23

[27] https://developer.twitter.com/en/docs/tweets/search/api-reference/get-search-tweets, last accessed: 2018-08-23

when starting a new data set the crawling process will stop by itself as soon as the default data limit of *Twitter* itself is reached, which returns only *tweets* published within the past seven days. In addition to this, it has to be noted, that the Search API of *Twitter* is based on a *Twitter* internal sampling of recent and popular *tweets* which "is focused on relevance and not completeness"[28]. This will also be further addressed in Section 7.2.

[28]https://developer.twitter.com/en/docs/tweets/search/overview/standard, last accessed: 2018-08-23

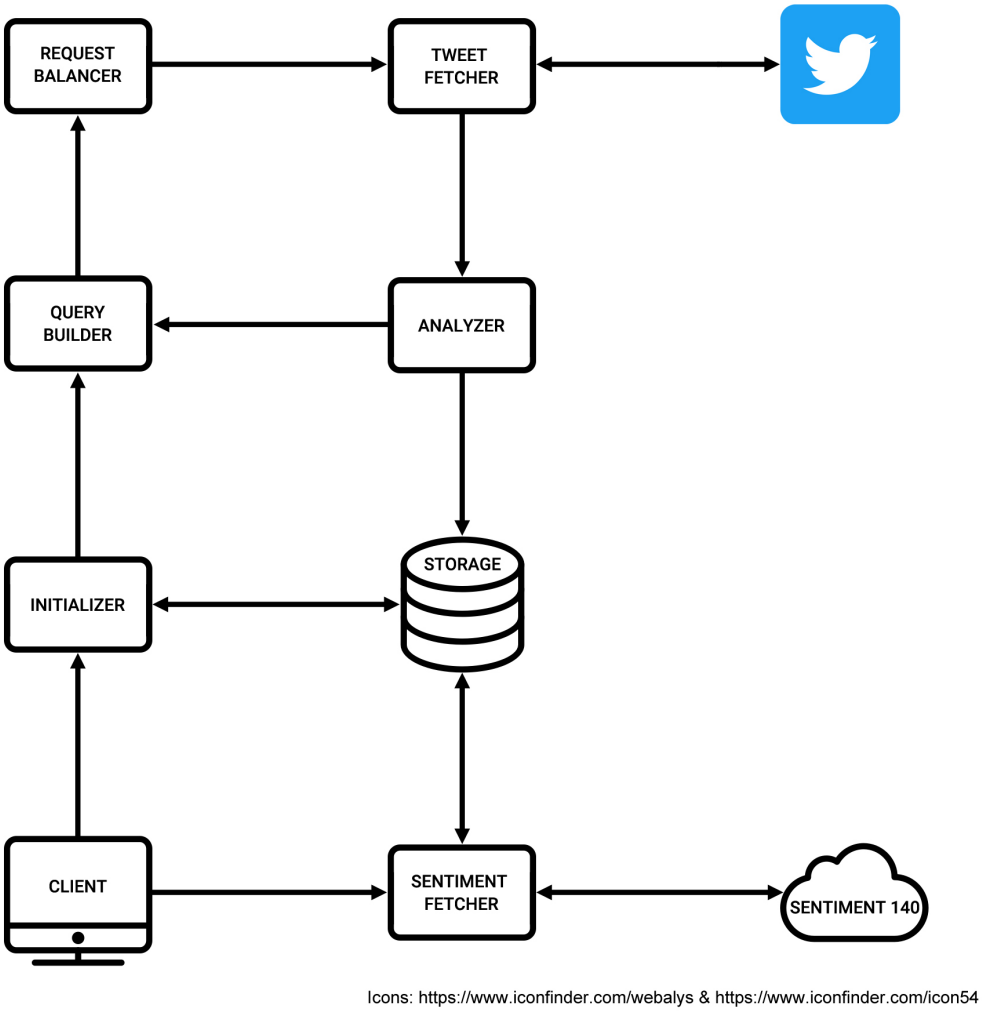


Figure 3.2: This overview illustrates structure, request and data flow of the crawler component and the Sentiment140 extension

3.2.2 Query Builder

The next part of the crawler component is the query builder: it simply takes incoming arguments, which originate either from the initializer or the analyzer, and creates a query seeded with these parameters.

3.2.3 Request Balancer

The request balancer checks and enforces *Twitter's* rate limits which have been set by the initializer. The search endpoint for example has a rate limit of 180 requests per 15-minutes time window for user authentication and 450 requests per 15-minute time window for app authentication, with the latter being used by this application. In the case the amount of requests by the crawler surpasses these 450 requests, the request balancer stops the crawling process for the remaining time and continues after the reset timestamp received by the *Twitter* Search API has passed.

3.2.4 Tweet Fetcher

Next, the *tweet* fetcher is the part of the crawler that sends the query to the *Twitter* Search API, receives the result, handles errors and passes the resulting data on to the analyzer.

3.2.5 Analyzer

Finally, the analyzer completes the cycle of the crawling functionality and basically has two tasks to perform: i) first, the analyzer iterates the received *tweet* objects, extracts, sets, and formats the information according to the schema of the four database tables users, *tweets*, places and profiles containing additional user information, before saving the resulting data to the database. After adding the aforementioned salient library to the application, the analyzer was extended by incorporating the salient sentiment analysis process before saving the data. ii) Second, if the received result set is not empty, the analyzer prepares and passes the `max_id` argument to the query builder in order to continue the crawling process.

3.2.6 Sentiment Fetcher

As mentioned before, the sentiment analysis based on the *Sentiment140* service was kept separated from the crawling process to not flood the service with a huge amount of requests during the crawling process. It can, however, be seen as an extension to the crawler. The client is used to start the process which consists of three steps: i) the sentiment fetcher requests *tweets* from the database that do not yet have a sentiment value limited to 5,000[29] *tweets*, ii) then sends the *tweets* to the *Sentiment140* service, iii) and finally updates all analyzed *tweets* in the database by setting the received sentiment

[29]The threshold of 5,000 tweets is a suggested value in regards to the Sentiment140 bulk classification service: http://help.sentiment140.com/api, last accessed: 2018-09-11

value. After that the process is simply restarted for as long as there are *tweets* without sentiment value.

To sum up this section, a detailed view on the crawler component has been provided. Started by the client, the initializer fetches basic data, which is then used by the query builder to construct the query, and by the request balancer in order to honor *Twitter's* rate limits. Next, the *tweet* fetcher sends the query to the *Twitter* Search API endpoint and passes the received data to the analyzer. There, the data is processed in order to save the results to the database and the `max_id` is passed to the query builder, which re-starts the crawling process. Since the application should not flood the *Sentiment140* service with many requests in a short period of time, this sentiment analysis was detached from the crawling process in order to process a larger amount of *tweets* at once.

3.3 Visualizer

The visualizer is an interactive front-end, which will be discussed in more detail in Chapter 5. It is used to request data for different visualization types as well as customizing various settings such as a period of time or the sentiment. Based on the visualization type and these settings, the visualizer handles multiple tasks: i) it builds and runs SQL queries based on the received settings, ii) combines different data samples such as *tweet* and in-game activities, iii) provides on-the-fly tokenization and word count functionality, and iv) formats the resulting data to a specific scheme that can be handled by the respective visualization. Finally, the data is returned to the front-end and the selected visualization is rendered.

3.4 Importer

The importer component was implemented in order to process a second data source in addition to the data collected via *Twitter* Search API.

The sample of in-game data was provided by colleagues of the *University of York* and *Fraunhofer IAIS* in the form of a large JSON file. Importing the file's content was a two-step process: since the data was structured as a list with each entry being the respective gamer account at its top level and containing lists of in-game activities segmented into metrics such as online characters related to the account, gameplay type and date of the activity, the importer at first iterated the list on its highest level and simply storing each entry into a temporary database table. After that, each of those entries were processed separately and saved in a scheme that allowed easy and fast access. In order to further reduce on-the-fly calculations, which went hand in hand in the course of complex and slow SQL queries, two database tables were added that contained the daily summaries of in-game activities for each player and in total.

As will be elaborated further in Section 7.3, more work has to be done in order to alter the importer component into a more general component that can be configured to work

with different sources and convert this custom data to a more practical form.

3.5 Exporter

Finally, the exporter component was implemented in order to export the resulting data sample, a combination of *Twitter* activities and the sum of all in-game metrics per user as a CSV file. CSV was chosen since it is a very common format, which can not only be processed programmatically very easy, but can also be imported to spreadsheet programs such as Microsoft Excel[30] for a more human-readable form. Similar to the importer, the exporter could be extended in the future in order to provide an export for each data set that is used by the visualizer as raw data.

To sum up, this chapter presented various insights into the inner workings of the application of this work including a description of each component it consists of, as well as an overview of all used frameworks, libraries and modules, and the development process. While the crawler component certainly is the most complex component of this application and therefore was described in great detail, the visualizer is much more simple structured and designed to basically request and format received data based on settings chosen in the front-end view. Finally, the importer and exporter were designed for specific processes related to the use case of this work and therefore require attention in the future in order to provide a generalized functionality.

[30] https://products.office.com/en/excel, last accessed: 2018-08-23

CHAPTER 4

Processing microblogging and in-game data

After taking a closer look at the technical structure of the application and its components in Chapter 3, this chapter discusses in more depth how *Twitter* works, how *tweets* are structured and how the analyzer component processes and converts this structure to a form that is more useful for the application of this work. Additionally, the last section of this chapter discusses the data structure of the in-game data sample and how it is processed and stored.

4.1 Understanding Twitter

Twitter is a microblogging service that was founded in 2006 and quickly grew to be one of the biggest social network services in the world with currently around 100 million daily active users publishing around 500 million *tweets* per day [Asl17]. It was initially designed as an SMS-based communication platform [Mac17] with a limit of 140 characters, which was increased to 280 characters in 2017 [New17].

The basic principle of *Twitter* is to share short status updates called *tweets*. If the profile of a user is not set to private, these *tweets* are generally publicly visible and can be seen by visiting the profile page of the user, but they are also automatically displayed to other subscribed users called followers, the result of *Twitter's* unidirectional networking feature: as opposed to the friends list system of other social network sites such as *Facebook*, where a user has to accept the friend request from another user, *Twitter* enables a user (A) to subscribe to or "follow" another user (B), which causes each *tweet* posted by user B to be displayed in the timeline of user A.

As MacArthur [Mac17] points out, user innovation played a big role in the development of *Twitter*: the re-tweet feature, for example, which allows a user to simply forward the

tweet written by someone else to his or her own followers, resulting in an interesting multiplier effect, started with users copying interesting *tweets* and manually adding "RT" and the original author to the content in order to give credit, before it was added as a core functionality to *Twitter*. Another example for user driven innovation given by MacArthur [Mac17] would be the feature of tagging other users by adding their *Twitter* user name preceded by the @-sign, a functionality that did not exist when *Twitter* was launched, but was added soon after users started to use this syntax in order to reply to or simply address other users. The same is true for hashtags, which were added in order to enable users to tag their *tweets* with certain topics preceded by the #-sign.

4.2 Tweets: Explaining Twitter's data structure

In general, *tweets* are multivariate data structures that contain much more data than what is visible on the surface. Each *tweet* object[1] consists of multiple *tweet* attributes as well as various child objects, which contain further information related to a certain aspect of a *tweet*. The following sections give an overview of those aspects.

4.2.1 Tweet attributes

The basic *tweet* attributes are a combination of fundamental attributes such as the date and time a *tweet* was posted, the ID and text of a *tweet*, multiple numerical values such as the amount of how often a *tweet* has been re-tweeted, quoted, replied to or marked as favorite, as well as attributes that provide additional information such as a user or *tweet* ID, if it is a reply, a language identifier or flags in order to indicate if the *tweet* has been quoted, re-tweeted, replied to, marked as favorite or possibly contains sensitive content. In addition to that, *tweet* objects might be present containing the original *tweet* if it has been quoted or re-tweeted.

4.2.2 The User Object

Furthermore, each *tweet* has a user object[2] attached which contains all public meta data of the user who has posted the *tweet*. These attributes are a mixture of static attributes, that do not change, such as the ID of the user or the time and date the account was created, attributes that can be changed by the user in the profile settings, such as the account's name, screen name, biography, language or location, and attributes that are calculated based on the user's activity such as the amount of *tweets* posted, or the follower, friends, favorites and list counts. In addition to that, there are various attributes related to the appearance of the user's profile that can be customized by the user such as background colors and images.

[1] https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/tweet-object, last accessed: 2018-08-23

[2] https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/user-object, last accessed: 2018-08-23

4.2.3 Geo-Location Data

Next, a *tweet* can contain geo-location meta data[3] in the form of GPS coordinates or a place object. While the coordinates object consists of longitude and latitude values and is only present if an exact location has been added to a *tweet*, the place object is present whenever a user has geo-tagged the *tweet*: when writing a *tweet*, the user has the possibility to add a location to the *tweet* by selecting a place from a list of places suggested based on user input.

The place object itself is basically a representation of a location containing an ID and name, a place type, a country code and name as well as a bounding box object which basically is a list of four longitude and latitude coordinates forming an area in which the place is located.

4.2.4 Entities

Finally, a *tweet* can contain so-called entities[4]. Entities are meta data objects extracted from a *tweet's* content by *Twitter*. Hashtags, for example, are extracted and presented in a list containing the name of the hashtag and indices marking the start and end position of the hashtag in the *tweets* content. Other entities are media objects such as images or videos, URLs, mentioned users, symbols and polls, with each entity having specific meta data attributes related to the respective entity.

To sum up this section, a single *tweet* is a multivariate data object which contains in addition to the information that is rendered on the *Twitter* website or its applications such as the *tweet's* content, date and time of its creation or the user who published the *tweet*, a vast amount of meta data is available related to the *tweet* and its author. While the attached user object enables a detailed view of the user who posted the *tweet*, coordinates and places, if present, add a geo-location reference to the *tweet*, entities extracted by *Twitter* provide an overview of *Twitter* features such as user mentions, hashtags or the usage of media files and URLs.

4.3 Processing and storing multivariate data

After taking a look at *Twitter* and how a *tweet* object is structured, this section discusses how data is processed by the analyzer of the crawler component. As mentioned in Section 3.2, each *tweet* object returned by the *Twitter* Search API was split up into four data objects: user, profile, place and *tweet*. These were then stored in respective tables in the database, as shown in Figure 4.1. The following sections give an overview about each data object, before explaining the procedure of processing and storing the *Twitter* Search API request results.

[3] https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/geo-objects, last accessed: 2018-08-23

[4] https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/entities-object, last accessed: 2018-08-23

4.3.1 User and Profile

The user and profile objects essentially hold all the data from *Twitter's* user object with attributes that specify the visual settings of a *Twitter* user's account put into a separate profile object[5].

The list of attributes of the profile object can be seen in Table 4.1

The user object on the other hand holds all data attributes that currently are used for analysis and visualization purposes in the application. The list of attributes of the user object can be seen in Table 4.2.

In addition to the attributes listed in Table 4.2, which were directly derived from the user object of *Twitter*, more attributes have been added in order to handle information extracted in the course of the analyzing process such as the gamertags which were required

[5] At the current state of the application profile data is not used, but can be useful in the future, if, for example, the profile of a user has to be displayed while considering the users chosen visual settings.

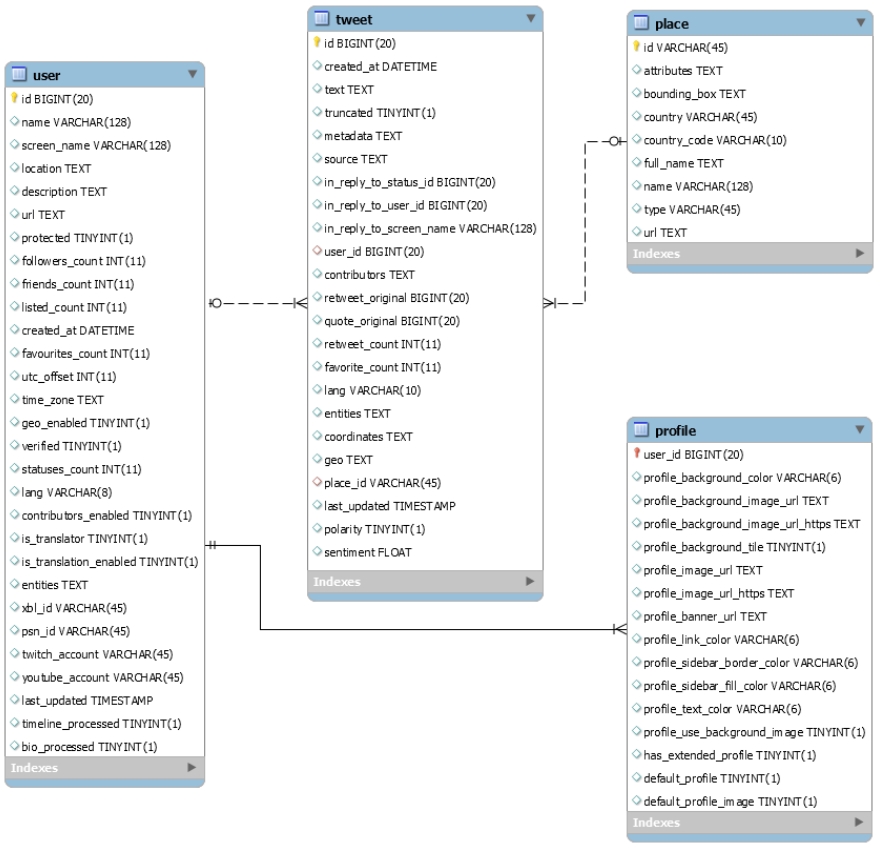


Figure 4.1: ER Diagram of the tweet, place, user and profile database tables.

in order to collect *Destiny* in-game data. The list of additional attributes can be seen in Table 4.3. Self-evidently, these attributes belong to the user object shown in Table 4.2.

4.3.2 Place

The place object holds the data of *Twitter's* place object. As mentioned before, it is a representation of a real-world location and marked by an internal *Twitter* ID. The complete attribute list of the place object can be seen in Table 4.4.

4.3.3 Tweet

The *tweet* object holds data from *Twitter's tweet* object, with the exception of the user and place objects which are being replaced by the respective ID. Additionally, a `last_updated` attribute was added, a date and time value which was used in the process of fetching the latest amounts of re-tweets and favorites. Furthermore, the *tweet* object was extended by two attributes holding sentiment analysis values, one for the result of the *Sentiment140* analysis, and the second one for the result of the sentiment

Attribute Name	Description
`user_id`	ID of the user
`profile_background_color`	Hexadecimal (HEX) value of the background color
`profile_background_image_url`	Link to a background image
`profile_background_image_url_https`	HTTPS-Link to a background image
`profile_background_tile`	A flag indicating if background images shall be displayed as tiles
`profile_banner_url`	Link to a banner image
`profile_image_url`	Link to a profile image
`profile_image_url_https`	HTTPS-Link to a profile image
`profile_link_color`	HEX value of the link color
`profile_sidebar_border_color`	HEX value of the sidebar border color
`profile_sidebar_fill_color`	HEX value of the sidebar background color
`profile_text_color`	HEX value of the text color
`profile_use_background_image`	A flag indicating if the user defined background image shall be used
`has_extended_profile`	This flag is part of the *Twitter* Search API's response, but it is not specified in the official documentation
`default_profile`	A flag indicating if the user has changed the profile's theme or background
`default_profile_image`	A flag indicating if the user has uploaded a profile image

Table 4.1: Overview of the profile object attributes

analysis based on the *salient* package, as described in Chapter 3. The complete attribute list of the *tweet* object is shown in Table 4.5.

4.3.4 Processing Twitters Search API request results

After taking a look at the four different data objects, this section describes how the data received from the *Twitter* Search API is processed. The result set for each request to the *search/tweets* endpoint of the *Twitter* Search API contains a list of up to 100 *tweet* objects. After receiving a result set, this list of *tweets* is iterated and the following process is applied to each entry in this list:

1. A new user object is created based on the user information attached to the *tweet* with all attributes listed in Table 4.2 being derived from *Twitter's* user object with

Attribute Name	Description
`id`	ID of the user
`name`	Name of the user
`screen_name`	Screen name of the user which is used to reply to or address other users (`@username`) in *tweets*
`location`	Self-reported location of the user, which might not be a location at all
`description`	Profile biography text written by the user
`url`	A user-defined URL
`protected`	A flag indicating if the user profile is public or not
`followers_count`	Amount of users who follow this user
`friends_count`	Amount of users this user follows
`listed_count`	Amount of lists this user has been added to
`created_at`	Date and time this user has joined *Twitter*
`favourites_count`	Amount of tweets this user has liked
`utc_offset`	Offset from the GMT/UTC time in seconds
`time_zone`	Self-reported timezone the user is in
`geo_enabled`	A flag indicating if the user has enabled geo-tagging
`verified`	A flag indicating if the user has a verified account
`statuses_count`	Amount of *tweets* this user has published (including re-tweets)
`lang`	Code of language the user has chosen for the user interface
`contributors_enabled`	A legacy flag indicating if the user has the contributor mode active (*tweets* can be co-authored)
`is_translator`	A deprecated flag indicating if the user is part of *Twitters* translator community
`is_translation_enabled`	This attribute is not mentioned by the official documentation
`entities`	JSON string containing entities extracted from the user's profile (renamed to "derived" at the time of writing)

Table 4.2: Overview of the user object attributes

three exceptions: the `created_at` attribute is converted from the UTC datetime format used by *Twitter* to the MySQL datetime format, the `entities` attribute is converted to a string containing the JSON object, while the `last_updated` attribute is not set because this is automatically handled by the database.

2. The profile object is created with all attributes listed in Table 4.1 being derived from *Twitter's* user object without any changes to the data.

3. A place object is created, if the *tweet* has place data attached. The attributes listed in Table 4.4 are derived from the place data with the `attributes` and `bounding_box` attributes being converted to a string containing the respective JSON object.

4. The *tweet* object is created based on the data of the current *tweet* object. All at-

Attribute Name	Description
`xbl_id`	XBL gamertag extracted from the user's biography (if available)
`psn_id`	PSN gamertag extracted from the user's biography (if available)
`twitch_account`	*Twitch* Account extracted from the user's biography (if available)
`youtube_account`	*YouTube* Account extracted from the user's biography (if available)
`last_updated`	Date and time when the user attributes have been updated last
`timeline_processed`	A flag indicating if the timeline of the user has been processed (further information will be provided in Chapter 6)
`bio_processed`	A flag indicating if the biography of the user has been processed

Table 4.3: Overview of the additional user object attributes

Attribute Name	Description
`id`	ID of the place as assigned by Twitter
`attributes`	Although part of Twitter's place object, the official documentation does not specify its meaning. Furthermore, this attribute is empty for the whole data sample collected by this application
`bounding_box`	JSON object containing sets of latitude and longitude coordinates forming a box in which this place is located
`country`	Country name of this place (e.g., United States)
`country_code`	Code of the country of this place (e.g., US)
`full_name`	Full name of this place (e.g., Manhattan, NY)
`name`	Short name of this place (e.g., Manhattan)
`place_type`	Type of the place (e.g., city)
`url`	URL to *Twitter's* API endpoint which contains additional meta data of this place

Table 4.4: The complete attribute list of the place object.

tributes listed in Table 4.5 except the `last_updated`, `polarity` and `sentiment` attribute are derived from the current *tweet* data.

While the `metadata`, `contributors`, `entities`, `coordinates` and `geo` attributes are converted to a string containing the respective JSON object, the `created_at` and `last_updated` attributes are handled similar to the user object, the former being converted from UTC to MySQL datetime format, and the latter not being set because it is handled automatically by the database. Furthermore, the `place_id` is derived from the place object, if present, whereas the sentiment of the *tweet's* content is analyzed and the resulting value assigned to the `sentiment` attribute.

Attribute Name	Description
`id`	ID of the *tweet*
`created_at`	Date and time the *tweet* was posted
`text`	Content of the *tweet*
`truncated`	A flag indicating if the *tweet's* content has been truncated by *Twitter* to fit the character limit
`metadata`	JSON object containing meta data; this attribute is not mentioned by the official documentation and has been removed from the API response at the time of writing; the data sample of this work does still contain meta data
`source`	HTML-formatted string linking the website or app from which the user posted the *tweet* (e.g. *Twitter* for Mac)
`in_reply_to_status_id`	ID of source *tweet*, if this tweet is a reply
`in_reply_to_user_id`	ID of user, if this *tweet* is a reply
`in_reply_to_screen_name`	Screen name of the user, if this *tweet* is a reply
`user_id`	ID of the user who posted this *tweet*
`contributors`	List of user IDs; this attribute is not mentioned by the official documentation and is empty in the data sample of this work
`retweet_original_id`	ID of the original *tweet* if this *tweet* is a re-tweet
`quote_original_id`	ID of the original *tweet* if this *tweet* is a quote
`retweet_count`	Amount how often this *tweet* has been re-tweeted
`favorite_count`	Amount how often this *tweet* has been liked
`lang`	Code of the *tweet's* language *Twitter* has detected
`entities`	JSON object containing entities *Twitter* has extracted from the *tweet's* content
`coordinates`	Deprecated JSON object containing GPS coordinates
`geo`	Deprecated JSON object containing geo-location data
`place_id`	ID of a place referencing a place object
`last_updated`	Date and time of this *tweet's* last update
`polarity`	Sentiment value as analyzed by *Sentiment140*
`sentiment`	Sentiment value as analyzed by *salient*

Table 4.5: The complete attribute list of the tweet object.

The user, profile, place and *tweet* objects created in each iteration are added to a respective list. After the iteration each list is stored to the database: first, the users list is saved, followed by the profiles and places lists, with the *tweets* list being stored at the end. The succession of saving the lists is determined by the foreign key constraints, as can be seen in the ER Diagram in Figure 4.1: the profile contains `user_id`, so the user has to exist, before a profile can be saved, while the *tweet* contains `user_id` and possibly `place_id`, which is why the user and place have to exist, before a *tweet* can be saved. Already existing *tweets*, users, profiles and places are ignored.

Name	Description*
`activitiesCleared`	Number of completed activities
`activitiesEntered`	Number of activities the player has joined
`activitiesWon`	Number of activities the player has won
`allParticipantsCount`	Number of players in activity
`allParticipantsScore`	Aggregated score of players in activity
`allParticipantsTimePlayed`	Aggregated time of players in activity
`assists`	Number of times player helped killing an enemy
`dailyMedalsEarned`	Number of medals the player has earned
`deaths`	Number of times player has died
`fastestCompletion`	Number of shortest time to complete activity
`highestCharacterLevel`	Number of highest character level
`highestLightLevel`	Number of highest light level
`highestSandboxLevel`	Number of highest sandbox level
`kills`	Number of enemies killed by player
`maximumPowerLevel`	Number of highest power level
`maximumWeaponLevel`	Number of highest weapon level
`objectivesCompleted`	Number of completed objectives
`precisionKills`	Number of precision kills (e.g. head shots)
`publicEventsCompleted`	Number of completed public events
`publicEventsJoined`	Number of public events player has joined
`remainingTimeAfterQuitSeconds`	Number of remaining seconds after player quit
`resurrectionsPerformed`	Number of players resurrected by this player
`resurrectionsReceived`	Number of times this player has been resurrected
`score`	Player's score in activity
`secondsPlayed`	Number of seconds played
`suicides`	Number of times player has committed suicide
`teamScore`	Team's score in activity
`totalActivityDurationSeconds`	Total number of seconds this activity lasted
`totalDeathDistance`	Total distance from which player has been killed
`totalKillDistance`	Total distance from which player has killed

*The official documentation of `Bungie's Destiny` API is very light and does not include a list or description of in-game metrics. The description is therefore derived from the respective metric name and defined to the best of the author's knowledge.

Table 4.6: List of daily in-game metrics.

4.4 The in-game data set

As mentioned in Section 3.4, the in-game data set used in this work was provided by colleagues of the *University of York* and *Fraunhofer IAIS* and was based on the list of gamertags which were collected as described in Section 3.1.5. Since the importing process of this data set has already been discussed in Section 3.4, this section takes a look at the resulting three data base tables. Although the game *Destiny* itself will be discussed in depth in Chapter 6, a few details will be presented in the following as they are relevant to the data set.

Destiny is an online first-person action game in which players are able to participate in different game modes. The in-game data set featured following six game modes:

1. *Arena*[6] is a three-player matchmaking player-versus-environment (PvE) mode that consists of up to six rounds with different challenges such as fighting off multiple waves of attacking non-player characters (NPCs) or a boss fight, and was introduced with the release of *House of Wolves*[7], *Destiny's* second expansion pack which was released on May 19, 2015[8].

2. *Patrol*[9] is *Destiny's* free roaming mode that allow players to freely explore various maps, complete different short missions such as kill specific enemies or reach a certain location to scan an object, or join public events that occur at various location at different times.

3. The Player-versus-player (PvP) mode[10] features various death match and objective-based modes, in which players compete against each other in order to earn experience and other rewards.

4. *Raids*[11] are challenging PvE activities featuring multi-stage missions for teams of six high-level players, that reward players with unique items, which is why rewards are limited to one chest per raid every week.

5. The story mode[12] features a PvE campaign consisting of short missions for up to three players, serving as an introduction to the world of *Destiny*.

6. Quite similar to *raids*, vi) *strikes*[13] feature missions with multiple objectives and a boss fight at the end, but are less difficult and therefore offer common rewards, but can be completed by a single player or a team of up to three players.

[6] https://www.destinypedia.com/Arena, last accessed: 2018-08-23
[7] https://www.bungie.net/en/pub/houseofwolves, last accessed: 2018-08-23
[8] https://www.bungie.net/en/News/Article/12858, last accessed: 2018-08-23
[9] https://www.destinygamewiki.com/wiki/Patrols, last accessed: 2018-08-23
[10] https://www.destinygamewiki.com/wiki/The_Crucible, last accessed: 2018-08-23
[11] https://www.destinygamewiki.com/wiki/Raids, last accessed: 2018-08-23
[12] https://www.destinygamewiki.com/wiki/Story_Missions, last accessed: 2018-08-23
[13] https://www.destinygamewiki.com/wiki/Strikes, last accessed: 2018-08-23

In order to dive deeper into the data set, Table 4.6 lists all in-game data metrics that are utilized in the database tables `daily_game_history`, which contains a sum of all in-game activities separated by game mode for each player and each day, and `aggregated_daily_game_history_sum`, which contains the aggregated amount of all in-game activities for each day combined for all players.

In addition to that, Table 4.7 lists all columns of the `aggregated_game_history` database table, which contains selected in-game activities for each user over the data sample's complete period of time.

Name	Description*
`mem_id`	Internal membership ID within *Destiny's* system
`user_id`	User ID within this application (*Twitter* User ID)
`xbl_id`	XBL gamertag
`psn_id`	PSN ID
`pvp_average_lifespan`	Average lifespan of player in PvP
`pve_average_lifespan`	Average lifespan of player in PvE
`pvp_best_single_game_kills`	Highest number of kills in a PvP match
`pve_best_single_game_kills`	Highest number of kills in a PvE session
`pvp_best_single_game_score`	Highest score in a PvP match
`pvp_combat_rating`	Destiny-internal rating of player performance
`pvp_kills_deaths_ratio`	Kill/Death ratio in PvP
`pve_kills_deaths_ratio`	Kill/Death ratio in PvE
`pvp_longest_kill_spree`	Highest amount of kills without dying in PvP
`pve_longest_kill_spree`	Highest amount of kills without dying in PvE
`pvp_longest_single_life`	Highest amount of seconds without dying in PvP
`pve_longest_single_life`	Highest amount of seconds without dying in PvE
`pvp_win_loss_ratio`	Win/Lose ratio in PvP
`pve_weapons`	JSON string with PvE weapon usage information
`pvp_weapons`	JSON string with PvP weapon usage information

List of metrics already discussed in Table 4.6: `pvp_activities_entered`, `pve_activities_entered`, `pvp_assists`, `pvp_assists_pga`†, `pve_assists`, `pve_assists_gpa`†, `pvp_deaths`, `pvp_deaths_pga`†, `pve_deaths`, `pve_deaths_gpa`†, `pvp_highest_character_level`, `pvp_highest_light_level`, `pve_highest_character_level`, `pve_highest_light_level`, `pvp_kills`, `pvp_kills_pga`†, `pve_kills`, `pve_kills_gpa`, `pvp_objectives_completed`, `pvp_objectives_completed_pga`†, `pve_objectives_completed`, `pve_objectives_completed_gpa`, `pvp_precision_kills`, `pvp_precision_kills_pga`†, `pve_precision_kills`, `pve_precision_kills_gpa`, `pve_public_events_completed`, `pve_public_events_joined`, `pvp_score`, `pvp_score_pga`†, `pvp_seconds_played`, `pvp_seconds_played_pga`†, `pve_seconds_played`, `pve_seconds_played_gpa`, `pvp_total_activity_duration_seconds`, `pvp_total_activity_duration_seconds_pga`†, `pve_total_activity_duration_seconds`, `pve_total_activity_duration_seconds_gpa`†

*The official documentation of `Bungie's Destiny` API is very light and does not include a list or description of in-game metrics. The description is therefore derived from the respective metric name and defined to the best of the author's knowledge.

†The `"_pga"` ending of the metric indicates a per-game-average value

Table 4.7: List of aggregated in-game metrics and player details.

To sum up, this chapter presented a deeper look into *Twitter* and the multivariate data structure of *tweets*, before describing the four data objects: users, profiles, places and *tweets*. In addition to that, it discussed how the results received from *Twitter's* Search API are processed, formatted into the aforementioned objects and saved to the database, before concluding by taking a closer look at the in-game data structure.

CHAPTER 5

Visualizing data

After discussing the structure of this application in Chapter 3 and taking a detailed look at how multivariate *Twitter* data was processed by the analyzer module in Chapter 4, this chapter will focus on the visualizer. It starts off by presenting the interactive front-end and all currently supported settings, before providing an example for each visualization type.

5.1 Web client

To control the visualizer and render the different visualization types and its settings a web client is required, which can be any modern browser with *JavaScript* enabled. At the time of writing, the visualizer front-end has been successfully tested on a *Microsoft Windows 10* machine with *Google Chrome*[1], *Mozilla Firefox Quantum*[2] and *Microsoft Edge*[3].

Subsequently, this section is separated into two parts: first, the canvas is explained in which the visualizations are rendered after loading data, and second, the configuration and all settings, which are used to determine the visualization types and data sets, are presented.

5.1.1 Canvas

The canvas area is located at the top of the page and, as can be seen in Figure 5.1, initially contains one empty canvas and the possibility to add additional canvases. Each canvas consists of an area for a visualization, a label stating the name of the canvas, and a select box which is used to pick the width of the canvas. Two options, full size and half

[1] Version 69.0.3497.92, latest run: 2018-09-12
[2] Version 62.0 (64-bit), latest run: 2018-09-12
[3] Version 42.17134.1.0, latest run: 2018-09-12

size, are available. The former is the default size, resulting in a canvas using up all the available width of the page, while the latter one causes the canvas to shrink to 50% of the width enabling the user of the application to have two visualizations side-by-side.

Below the canvas area, there is an area called "Active Canvas", which is used to add additional canvases by clicking the respective button and to choose the canvas from the select box which shall contain the visualization for the next request.

The functionality of adding multiple canvases and defining the canvas width makes it possible to build reports in the form of dashboard-like overviews that can highlight different approaches and underline specific stories a data analyst might want to tell. Furthermore, these reports can directly be printed by using the print functionality of a browser.

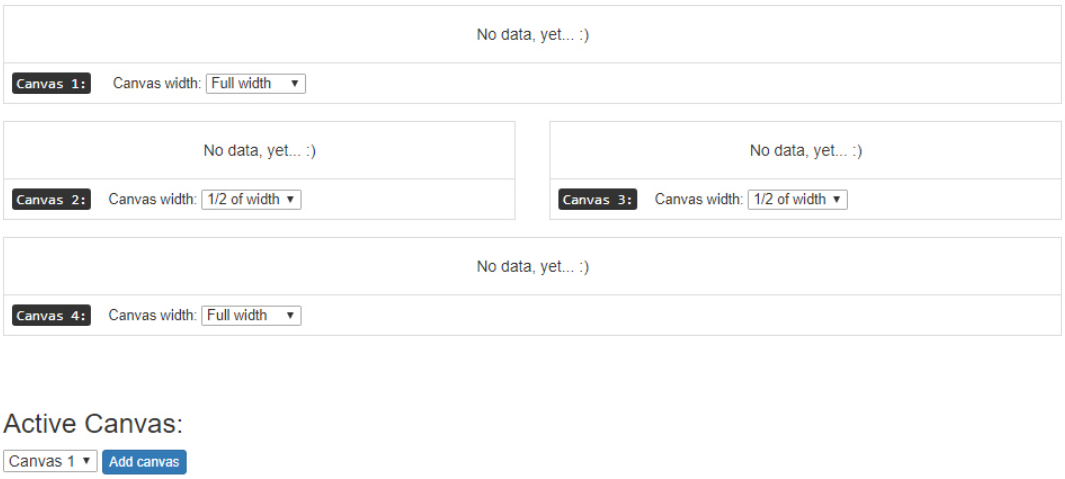


Figure 5.1: Screenshot of the visualizer's canvas area containing four empty canvases.

5.1.2 Configuration

The configuration is located below the canvas area. As Figure 5.2 shows, it consists of seven areas, which will be discussed subsequently, and a load button at the bottom to asynchronously request a visualization based on the configured settings.

Visualization Type Selection

This area is used to select the visualization the user wants to load. Table 5.1 shows which options are available including a short description.

Date Selection

The date selection area consists of two date inputs that enable the user to define a date range. The input fields are using the HTML5 date type and are rendered with an advanced date select dropdown depending on the respective browser, which makes it easy to choose the desired dates.

Sentiment

The sentiment area is split into two parts. i) A list of checkboxes enables the analyst to choose which sentiments shall be used as a base for the next visualization that supports this setting. By default, the "All" checkbox is selected which, for example, causes the word cloud to be displayed without taking a sentiment into account. To give another example, by setting only the "Positive" checkbox, the word cloud will be displayed based only on *tweets* with a positive sentiment value. In addition to this list of checkboxes, ii) a pair of radio buttons is available which are used to select what kind of sentiment analysis method the analyst wants to use: as discussed in Section 3.1.9, the two sentiment analysis tools used are the external *Sentiment140* service, and the internally running *salient* module.

Visualization	Description
Daily *Tweets*	Line chart of amount of daily *tweets* over time. If sentiment is selected, a stacked bar chart is displayed instead of line chart color-coded based on sentiment
Tweets vs Game Data	Line chart of amount of daily *tweets* compared to amount of daily in-game activities
Tweets vs Game Data (Scatterplot)	Scatterplot of users, comparing total amount of *tweets* to total amount of in-game activities
Word Cloud	Word Cloud of the 200 most used words
Retweet / Fav Count	Scatterplot of *tweets*, comparing amount of *re-tweets* to the amount of favorite markings, color-coded based on sentiment value
Daily Top *Tweets*	Bubble chart of the daily 20 most *re-tweeted tweets* color-coded based on their sentiment
Top Influencer	Filterable and sortable table of most influencing Twitter accounts
Top Players	Filterable and sortable table of players
Search *Tweets* by Term	Filterable and sortable table of *tweets* based on a keyword search against *tweet* texts
Search *Tweets* by Username	Filterable and sortable table of *tweets* posted by a defined user name
All *Tweets* per Day	Filterable and sortable table of *tweets* posted on a defined day

Table 5.1: List of visualizations supported by the application

Word Cloud Specific Settings

This area contains a "WordCloud Multiplier" input field: since the number of words can vary considerably due to the possibility to set a date range, and the word cloud can be displayed in canvases of different sizes, this setting is used to basically re-render the word cloud with larger or smaller words.

Scatterplot Specific Settings

The scatterplot specific settings enable the analyst to highlight the data points of up to two user accounts in scatterplots. To do so, there is an input field for the user name and a HTML5 color selector for each user.

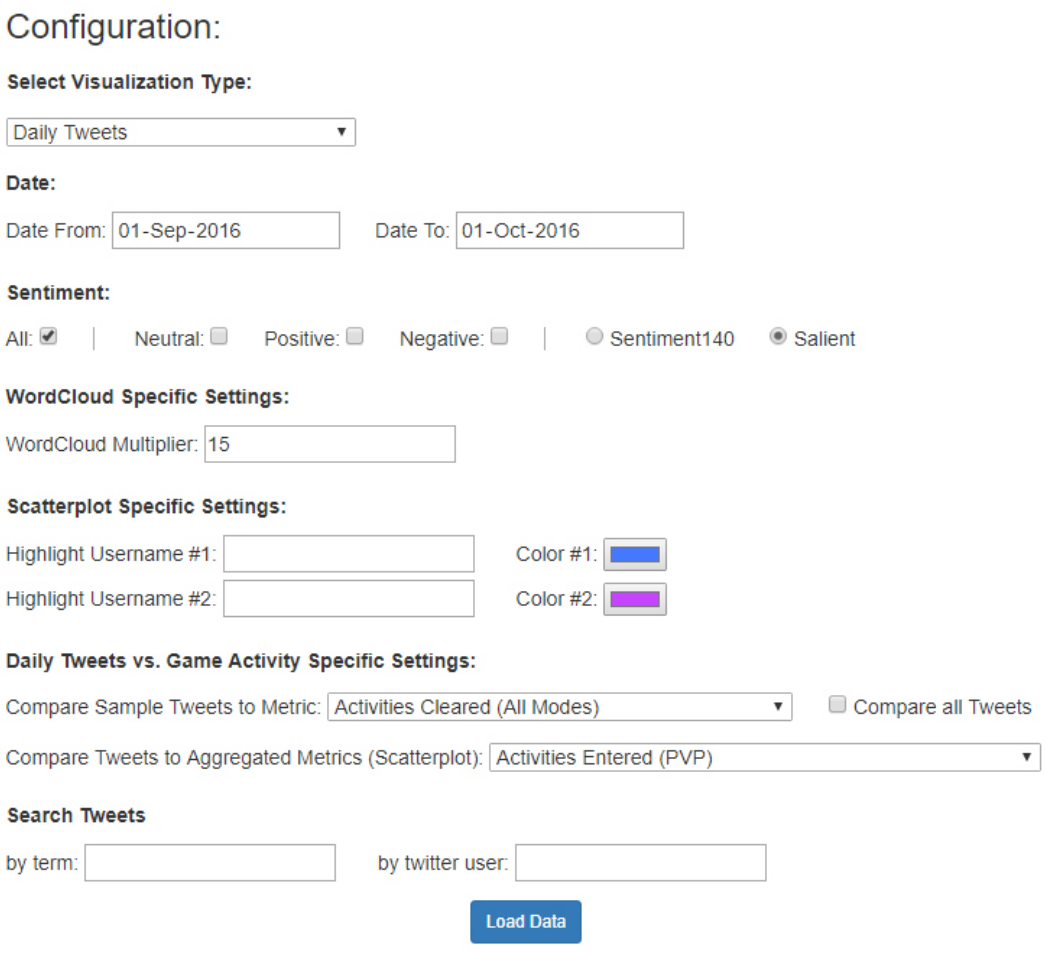


Figure 5.2: Screenshot of the visualizer's configuration area

Daily *Tweets* vs. Game Activity Specific Settings

This area is also consists of two parts: i) the upper select box allows the analyst to select an in-game gameplay metric and compare its amount to the daily *tweet* activity. By default, the daily *tweet* activity of users which are also part of the in-game data sample is used, but it is also possible to compare in-game metrics to the *tweet* activities of all users by setting a checkbox. ii) The part below enables the analyst to choose aggregated in-game gameplay metrics, which is then presented as a scatterplot comparing user's *tweet* activities to the selected in-game metric.

Search *Tweets*

The last area of the configuration holds two input fields, with the first one defining a search term or keyword, and the second one, setting a *Twitter* user name. Both input fields are used to search for *tweets*, either by words the *tweet* text contains or by the author who has posted the *tweet*.

To sum up this section, a web client is used to display the front-end of the visualizer, consisting of at least one canvas, a container, in which a visualization gets rendered, and a configuration area. The configuration is used by an analyst to set basic information such as selecting the visualization type that shall be shown, the temporal range in which the *Twitter* or in-game activities took place, or the sentiment values the data set shall be based on, but also to customize advanced settings such as highlighting users in scatterplots or selecting different in-game gameplay metrics used for comparisons.

5.2 Visualizations

This section will take a closer look on the different visualization types supported by the visualizer. All visualizations in some form provide interactive features. Moreover, visualizations based on Highcharts 3.1.7, which currently are line and bar charts 5.2.1, scatterplots 5.2.2, and the bubble chart 5.2.3, support an out-of-the-box functionality to print or export the visualization as PDF document, PNG or JPEG image, or SVG vector image. A small menu with the list of those options is located at the top right corner of each visualization.

5.2.1 Time Series: Activities Over Time

In total, three visualization types present activity data over a timeline.

Amount of daily *tweets* over time

The amount of daily *tweets* over time is visualized as a line chart with each data point representing the amount of *tweets* for each day. As can be seen in Figure 5.3, the x-axis represents time, while the y-axis shows the *tweet* count on a linear scale. The visualization is interactive and can be zoomed in by clicking and dragging within the plot area. In

addition to that, a tooltip is shown when moving the cursor near a data point containing the date and exact *tweet* count of the respective data point.

Sentiment of daily *tweets* over time

The sentiment of daily *tweets* over time is visualized as a stacked bar chart. As shown in Figure 5.4 each bar consists of the percentage of the selected sentiments, as represented on the y-axis, with 100% being the accumulated amount of *tweets* allotted to the respective

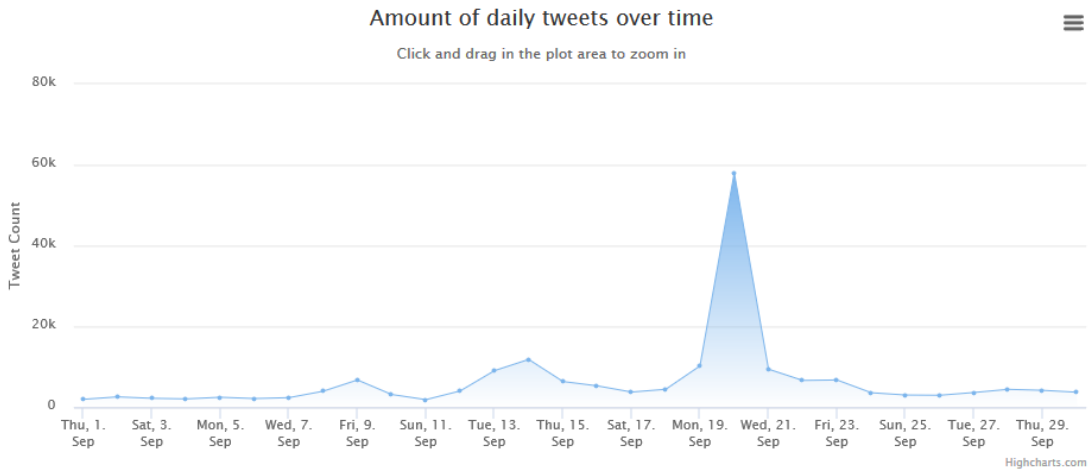


Figure 5.3: Amount of daily *tweets* over time (from 2016-09-01 to 2016-09-30)

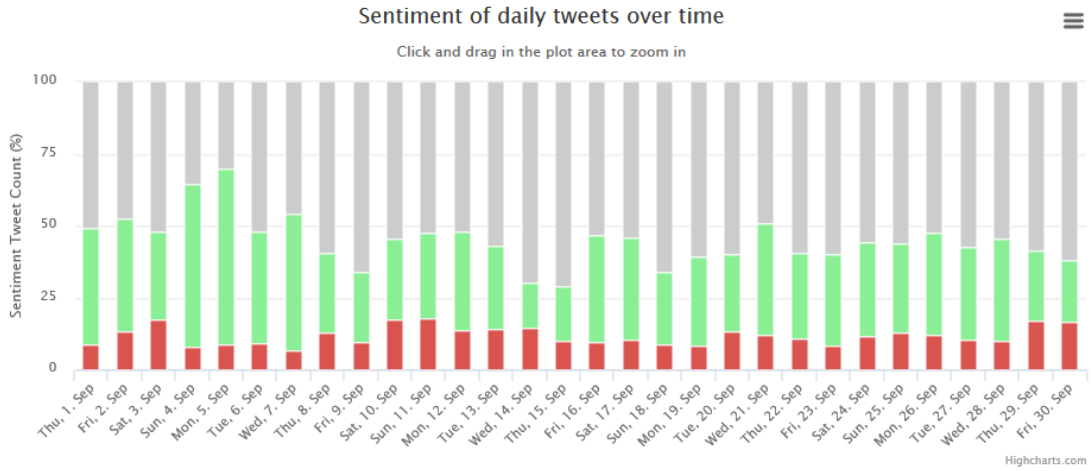


Figure 5.4: Stacked barchart with percentagees of positive, negative and neutral *tweets* over time (from 2016-09-01 to 2016-09-30)

sentiments per day, which is shown on the x-axis. Again, the visualization is zoomable by selecting a desired area within the visualization container. By hovering with the cursor over a bar, a tooltip is displayed revealing the raw values of the sentiments present in the data set of the selected day, including the percentage values rounded to two decimals and the total amount of *tweets*.

Tweets vs Game Data

The *tweets* vs game data visualization contains two line charts: for each day, represented by the x-axis, i) the amount of daily *tweets* is plotted on the y-axis with the scale labeled on the left side of the visualization, while ii) a second data set containing daily in-game activities is also plotted on the y-axis with the scale being labeled on the right side of the visualization. In the example shown in Figure 5.5, the activities entered (all modes) metric has been chosen, which, as discussed in Section 4.4, contains the amount of times all players in the in-game data set have entered an in-game activity regardless of the game mode. Like the previous two discussed visualizations, the *tweets* vs game data visualization is interactive and can be zoomed by selecting an area within the visualization, but this time not only the x-axis is zoomable, but also the y-axis. In addition to that lines can be disabled and enabled by clicking the respective label at the bottom of the visualization.

5.2.2 Scatterplots

Scatterplots are used in order to visualize the comparison of two metrics which additionally is or can be enhanced by color-coding. Currently, there are two different visualizations

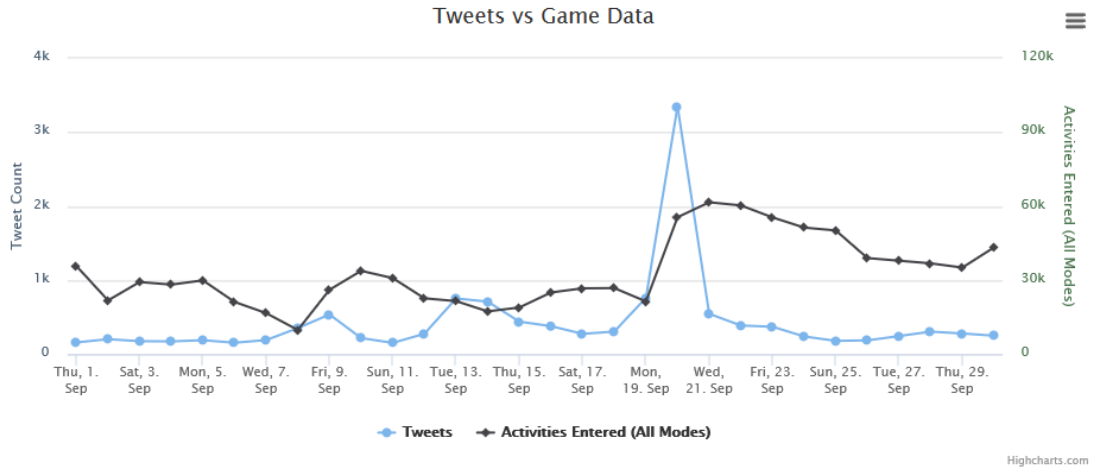


Figure 5.5: Line charts comparing the amount of *tweets* to the in-game data metric "activities entered (all modes)" over time (from 2016-09-01 to 2016-09-30)

supported: i) the favorite count vs *re-tweet* count scatterplot, as shown in Figure 5.6, presents each *tweet* within the set time frame as a dot, with the number of favorites being represented on the y-axis and the amount of *re-tweets* being plotted on the x-axis. Each axis is scaled logarithmically in order to better deal with outliers. In addition to that, each dot is color-coded based on its sentiment, with the positive *tweets* being colored green, the negative red and the neutral in light-gray. Furthermore, a tooltip is shown when hovering over a dot containing the sentiment, the exact amount of *re-tweets* and favorites, the author of the *tweet* and the *tweet's* content.

ii) The single user *tweets* vs game data scatterplot, as shown in Figure 5.7, presents each *Twitter* user with in-game data as a dot, with the amount of *tweets* plotted on the y-axis and the chosen in-game metric such as "Kills (PVP)", on the x-axis. Optionally, up to two users can be color-coded in order to highlight their dots and to more easily compare different metrics to each other. Again, a tooltip is displayed when hovering over a data point containing the user's name and the raw values of the *tweet* count and the chosen in-game metric.

Additionally, both scatterplots are interactive: data sets can be disabled and enabled by clicking on the respective caption, while the scatterplot is zoomable by selecting an area within the visualization.

5.2.3 Bubble Charts

At the current state of the application, one bubble chart is supported. As shown in Figure 5.8, the top 20 *tweets* of each day in regards to the amount of *re-tweets*, are

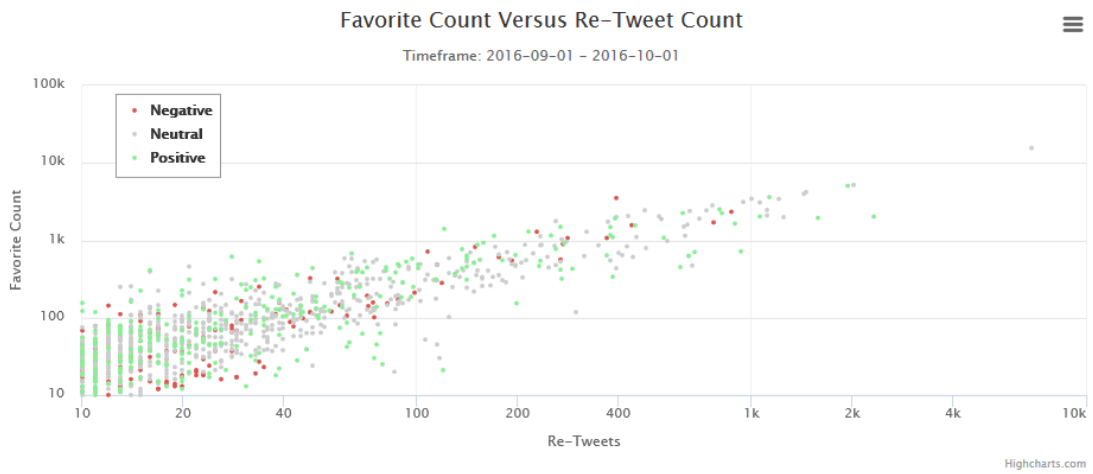


Figure 5.6: Sentiment color-coded scatterplot comparing the favorite count to the amount of *re-tweets* of *tweets* with more than 10 favorites and *re-tweets* and being posted from 2016-09-01 to 2016-10-01

displayed as a bubble: while the size and position of the bubble indicates the amount of *re-tweets*, the color of the bubble represents the sentiment of the *tweet* with green standing for positive, red for negative and light-gray for neutral *tweets*. Each day of the chosen time frame is shown on the x-axis, while the amount of *re-tweets* is plotted on the y-axis with a logarithmic scale. When hovering over a bubble, a tooltip is displayed containing the amount of *re-tweets* and favorites, the user and screen name of the *tweets* author, as well as the *tweet's* content.

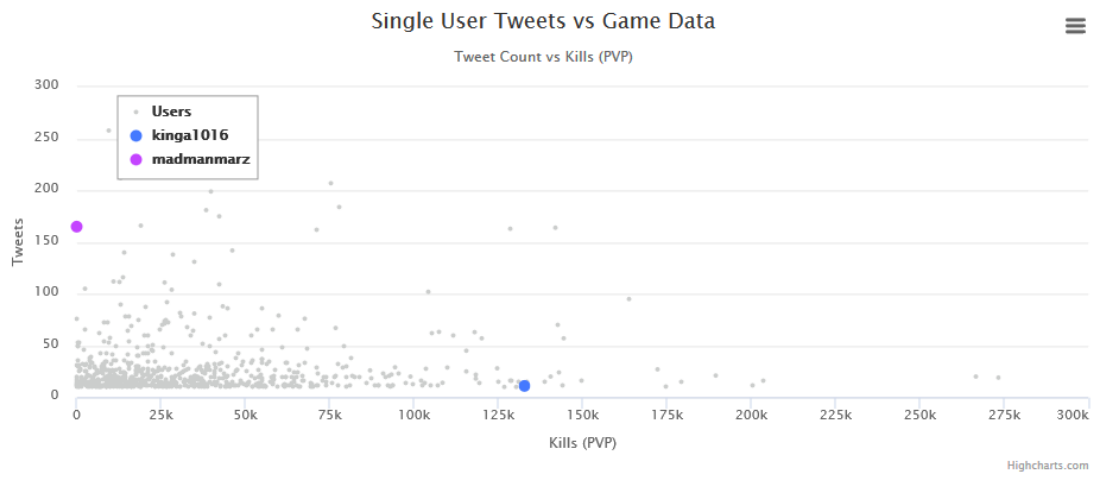


Figure 5.7: Scatterplot comparing the amount of *tweets* to the in-game data metric "Kills (PVP)" for each player over the complete data sample with two players being highlighted.

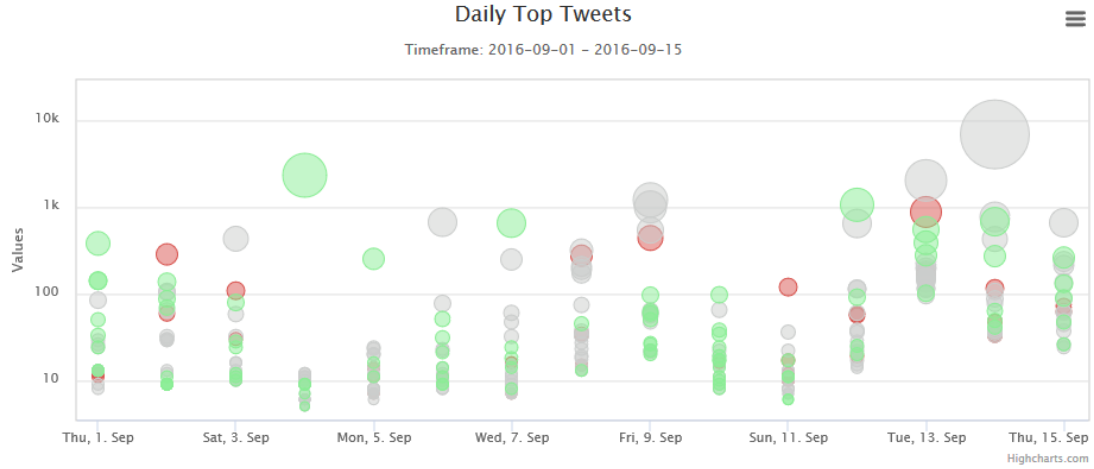


Figure 5.8: Bubble chart presenting the top 20 *tweets* for each day (from 2016-09-01 to 2016-09-15).

5.2.4 Word Cloud

As shown in Figure 5.9, the word cloud visualization displays the 200 most used words in *tweets* posted in the chosen time frame. The size of the words indicates how often they appeared in *tweets*, with bigger words at the center of the word cloud and less used words positioned outwards. The word cloud can be displayed separately for each sentiment with black being all sentiments or *tweets* that have been classified as neutral, red representing negative *tweets* and green illustrating positive *tweets*. Although the word cloud does not provide the same level of interactivity as the other visualization types, words can be clicked on with two different effects: i) if the word is an URL, it is opened in a new browser window, which enhances the workflow when conducting qualitative analysis, and ii) otherwise, the word is copied to the search term input field, which – if desired – can then be used to get a list of all *tweets* containing this word.

5.2.5 Data Tables

Finally, a data table visualization is used in order to display a large amount of different data metrics related to user accounts, in-game data or *tweets*. As shown in Figure 5.10, a data table consists of multiple columns labeling each data metric, and a row for each data entry. The data table is interactive and can be filtered by using the search input field positioned in the top right corner of the visualization, while each column can be sorted. In addition to that, the data table supports pagination and therefore can display

Figure 5.9: Word Cloud containing the 200 most used words in *tweets* (from 2016-09-01 to 2016-09-30).

large data sets across multiple pages. The navigation for the pagination which enables the user to switch to other pages, is positioned on the bottom right, while the setting to define how many entries are shown per page is positioned top left. All settings can be changed on the fly making it possible to analyze data sets very fast. Furthermore, if a data table contains the user account name, clicking on it causes the name to be copied into the *Twitter* user name search field which can be used to quickly trigger a search query resulting in a list of *tweets* posted by this user in the specified time frame.

To summarize, this chapter presented a detailed description of how the front-end of the visualization component is structured before giving an example for each visualization type supported by this application in its current state. In short, the front-end consists of a canvas area which contains one or more canvases in which the visualizations are rendered, and a configuration area which enables the user to configure various settings such as the visualization type, a time frame, the sentiment or additional visualization specific settings.

Examples and deeper descriptions of each supported visualization were also given: line

Show 10 entries Search:

Text	Retweet Count	Favorite Count	Twitter User	Date
Bungie servers right now. #RiseofIron https://t.co /wyGKH227u5	1112	2088	@tripleWRECK	2016-09-20
Servers are slammed so we had some time to burn. #RiseofIron https://t.co/1EQlkveLNc	629	1503	@Arbys	2016-09-20
Destiny servers are not available..... The scene at Bungie for #RiseOfIron https://t.co/WzE0FystOV	444	608	@FisherWrestling	2016-09-20
Destiny servers in a nut shell ... #RiseofIron https://t.co /TUl0V7DGdY	401	687	@SmileB4DEATH_	2016-09-20
"The servers are ready, boss." "Ok. Let's fire it up." #RiseofIron #Tapir https://t.co/VtzYZ2yrtp	270	564	@GameGuyPGH	2016-09-20
I wish my friends tried to contact me as much as I'm trying to contact Destiny servers right now #RiseofIron	197	643	@thegeek_chic	2016-09-20
Destiny servers right now #RiseofIron https://t.co /hd5pE4WeqD	166	261	@senormarkymark	2016-09-20
Just when you think you got into the @DestinyTheGame servers tonight. https://t.co/WhP9StqGFL	149	513	@franmirabella	2016-09-20
"Destiny servers are offline" *kicks me back 100,000 spots in the que* #RiseofIron https://t.co/D96NrLxuCW	88	173	@senormarkymark	2016-09-20
Destiny Servers Got Me Like #RiseofIron https://t.co /wXJlxBSJw1	62	86	@RichJus_	2016-09-20

Showing 1 to 10 of 2,965 entries Previous 1 2 3 4 5 … 297 Next

Figure 5.10: A data table containing search term results based on the term "servers" on 2016-09-20.

charts and bar charts are used to present data over a chosen time line, scatter plots are used to compare two data metrics against each other, whereas a bubble chart visualizes the most *re-tweeted tweets* per day. Furthermore, the most used words in *tweets* are represented by a word cloud, while multiple data tables are used to display large amounts of different data related to *tweets* or *Twitter* users.

CHAPTER 6

Use case "Destiny"

After giving an overview of the application of this work in Chapter 3, discussing how the application processes data in Chapter 4, and presenting all supported visualizations in Chapter 5, this chapter introduces the video game *Destiny* [Bun], discusses why it was chosen as the use case of this work before going through some scenarios in order to present examples how the application can be used.

6.1 Introducing Destiny

Destiny is a first-person action game developed by *Bungie Inc.*[1] and published by *Activision Publishing Inc.*[2] worldwide on September 9, 2014 for the *Xbox 360*, *Xbox One*, *PlayStation 3* and *PlayStation 4* video game consoles[3]. Set in a science-fiction world, the player takes on the role of a so-called "guardian", and sets out to explore diverse landscapes all over the solar system fighting various alien life-forms that threaten to wipe out humanity[4]. While *Bungie* refrained from marketing *Destiny* as a massively multiplayer online first-person shooter game (MMOFPS), they labeled it a "shared world shooter", because it can be fully played alone, but incorporates massively multiplayer online features to seamlessly connect to other players[5].

6.1.1 Why Destiny?

As part of this introduction, it is also beneficial to answer the question why *Destiny* has been chosen as use case of this work. In total, four reasons can be stated:

[1] Website of Bungie Inc.: https://www.bungie.net/, last accessed: 2018-01-25
[2] Website of Activision Publishing Inc.: https://www.activision.com/, last accessed: 2018-01-25
[3] https://www.activision.com/games/destiny/destiny, last accessed: 2018-01-25
[4] Official website of the game: https://www.destinythegame.com/d1, last accessed: 2018-01-25
[5] http://www.ign.com/articles/2013/02/17/bungies-destiny-a-land-of-hope-and-dreams, last accessed: 2018-01-25

- The history of *Bungie Inc.* itself was a strong indicator for increased initial in-game and *Twitter* activity. As the developer of the highly successful *Halo* franchise, *Bungie* has been known for providing astounding, cinematic sci-fi experiences combined with action-packed multiplayer modes. In addition, *Bungie* has gained a huge and active community in recent years due to a lot of community-friendly activities like running an active blog filled with posts answering community questions, sharing fan art or introducing *Bungie* employees to the community[6], or the *Bungie Favorites* listing screenshots, videos and other in-game content created by the community[7], and lots of special events such as the *Bungie vs. The World Steaktacular*, a special *Halo: Reach* multiplayer playlist, in which gamers around the world could play against teams consisting of *Bungie* employees, and with players beating the *Bungie* team by a 20 or more kills margin receiving a steak[8]. As a result, *Destiny* has been highly anticipated with a bar set very high.

- Although becoming the "most successful new video game franchise launch ever"[9] *Destiny* received only mixed reviews at launch[10], with many players voicing their disappointment about repetitive mission design, soul-less characters and an outright boring story[11]. Being far from perfect at release made *Destiny* a great choice in order to investigate how the developer reacts to feedback and if the sentiment changes over time.

- It is even more interesting when considering that long before *Destiny's* release, the publishing contract between *Bungie* and Activision has been made public in the course of a court case[12], revealing interesting details: in total, four *Destiny* games as well as multiple downloadable content packs (DLC) were planned during a 10-year period with the games being released every two years and DLCs in between. At the time the data acquisition for this work began, the first installment of *Destiny* as well as the DLCs *The Dark Below*[13], *House of Wolves*[14], and *The Taken King*[15] had been released, addressing many of the initial problems, gradually improving the overall experience. Furthermore, the launch of the fourth and last major DLC called *Rise of Iron* was already scheduled and took place within the data acquisition

[6] https://halo.bungie.net/news/blog.aspx, last accessed: 2018-02-15

[7] https://halo.bungie.net/online/bungiefavorites.aspx, last accessed: 2018-02-15

[8] https://halo.bungie.net/News/content.aspx?type=topnews&link=steaktakular, last accessed: 2018-02-15

[9] http://www.businessinsider.com/destiny-is-now-the-most-successful-launch-for-a-new-video-game-franchise-ever-2014-9, last accessed: 2018-01-30

[10] https://www.cinemablend.com/games/Destiny-Reviews-Did-Bungie-Shooter-Sink-Or-Swim-67259.html, last accessed: 2018-01-30

[11] http://www.metacritic.com/game/playstation-4/destiny/user-reviews, last accessed: 2018-01-30

[12] http://www.gamesindustry.biz/articles/2012-05-22-bungies-lucrative-contract-with-activision-is-revealed, last accessed: 2018-01-30

[13] http://www.ign.com/wikis/destiny/The_Dark_Below, last accessed: 2018-01-30

[14] http://www.ign.com/articles/2015/05/28/destiny-house-of-wolves-review-2, last accessed: 2018-01-30

[15] http://www.ign.com/articles/2015/09/16/destiny-the-taken-king-review, last accessed: 2018-01-30

period. Surprisingly, *Rise of Iron* was received as just a mediocre expansion[16] for the main game.

- *Bungie* provides an API for accessing *Destiny* in-game data.

6.2 Usage scenarios

This section focuses on the visualization part of the application and presents usage scenarios based on all supported visualization types discussed in Section 5.2. After an initial inspection of the 14 months of *Twitter* data and the 6 months of in-game data, September 2016 looked like the most interesting month in the data samples. Therefore, it was chosen as starting point for the following usage scenarios.

6.2.1 Exploring *Twitter* data

When exploring *Twitter* data with the visualization part of the application, it is most helpful to start out with the amount of daily *tweets* over time visualization (Section 5.2.1). As can be seen in Figure 6.1, a huge spike in *Twitter* activities occurred on the 20th of September 2016: while the amount of daily *tweets* surpassed the 10,000 *tweets* mark on only two other days in September, the number of activities surged to over 58,000 *tweets* on this day making it obvious that something major has happened.

[16] http://www.metacritic.com/game/playstation-4/destiny-rise-of-iron, last accessed: 2018-01-31

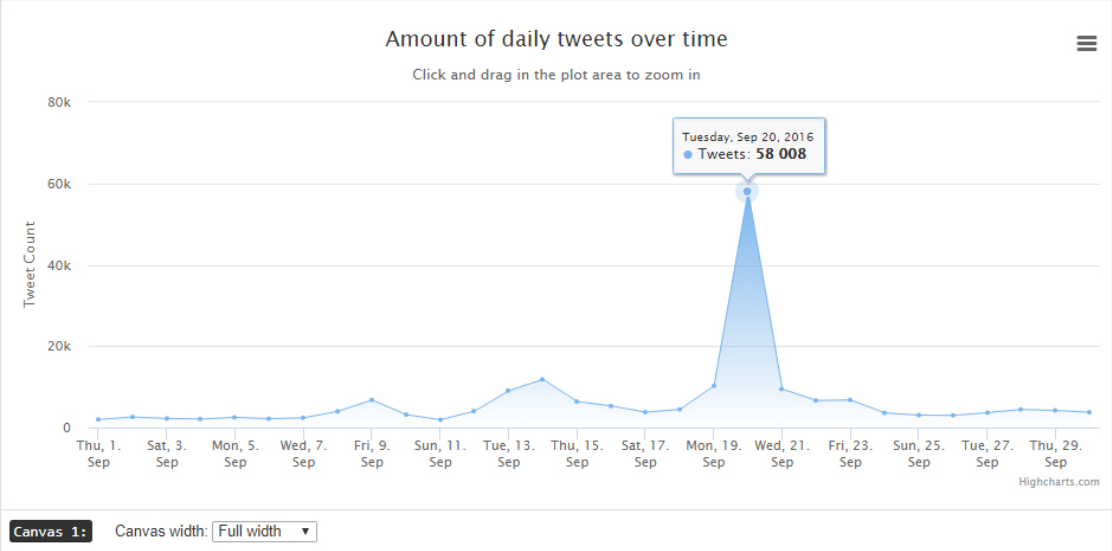


Figure 6.1: The daily *tweets* over time visualization for the period of 2016-09-01 to 2016-09-30 clearly shows a massive spike in *Twitter* activity on the 20th of September 2016.

In order to explore the reasons for such an increase in activity numbers, it is of interest to inspect the overall sentiment of this day, which is done by using the sentiment of daily *tweets* over time visualization (Section 5.2.1). As Figure 6.2 shows, the composition of *tweets* with neutral, positive and negative sentiment is not out of the ordinary, but an increase in the percentage of negatively classified *tweets* to 13.27% can be seen when compared to 8.09% and 8.88% of the previous two days.

The word cloud (Section 5.2.4) can then be used to explore the context of the increased activity. While in Figure 6.3 the 200 most frequently occurring words of all *tweets* are listed as word cloud, Figure 6.4 is based on all *tweets* classified as negative. Both visualizations feature the words *riseofiron* and *bungie*, but in the word cloud of negatively classified *tweets*, words such as *servers*, *queue* and a variety of swearwords are visualized more prominently.

An overall context can be drawn from the word clouds: it seems to be the *launch* day of the *Rise of Iron* expansion, but there seems to be a problem with the *servers*.

To confirm this, *tweets* of this day containing a certain search term can be listed in a data table (Section 5.2.5). As can be seen in Figure 5.10, searching for the *servers* term results in an extensive list of *tweets* ranting or making fun of *Bungie*'s servers being offline and people not being able to play.

In addition to that, URLs in the word cloud and in the data table provide further details and reveal a vast quantity of amusing images and GIFs targeting those technical problems, as can be seen in the example in Figure 6.5.

To present another usage scenario, an approach to quickly gain an overview of the context

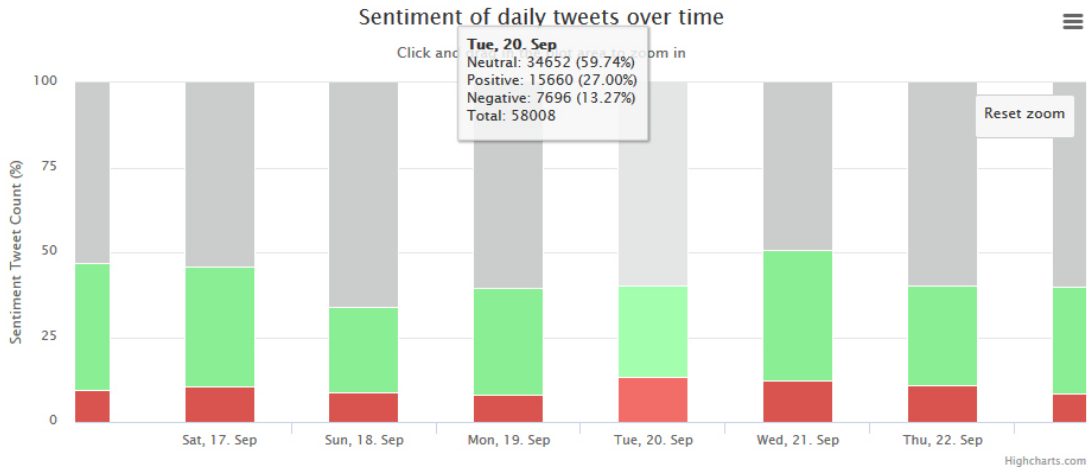


Figure 6.2: The sentiment of daily *tweets* over time visualization zoomed to the period of 2016-09-17 to 2016-09-22 visualizes the composition of *tweets* classified as neutral (grey), positive (green) and negative (red) for each day.

Figure 6.3: Word cloud of the 200 most frequently occurring words in *tweets* on 2016-09-20.

Figure 6.4: Word cloud of the 200 most frequently occurring words in *tweets* classified as negative on 2016-09-20.

Figure 6.5: Example of a *tweet* making fun of the server outage on the launch day of the *Rise of Iron* expansion on 2016-09-20 (https://t.co/wyGKH227u5, last accessed: 2018-09-17).

of *Twitter* activities on a daily basis is to study the daily top view visualization. As discussed in Section 5.2.3, this bubble chart presents the top 20 *tweets* for each day and contains the content, author and sentiment of a *tweet* as well as the *re-tweet* and favorite statistics. Figure 6.6, for example, visualizes all top *tweets* throughout September and contains the tooltip of the most *re-tweeted tweet* from this month posted on the 14th of September by the official *@DestinyTheGame* account:

Rise up and become an Iron Lord. Play Destiny: Rise of Iron on 9.20.16. https://t.co/oDXnwaHVq1

The *tweet* is about the upcoming *Rise of Iron* DLC and includes the official Rise of Iron launch trailer.

Unsurprisingly, most top *tweets* were posted by *Twitter* accounts controlled by *Bungie* (*@Bungie*, *@DestinyTheGame*, *@BungieHelp*): in September alone, they account for the most *re-tweeted tweet* on 20 out of 30 days. The other top ranked *tweets* in this month were claimed by major video game magazines or eSports related websites such as *@IGN*, *@Kotaku* or *@MLG*, and a handful of *Twitter* accounts belonging to YouTuber or Live-Streamer who were holding raffles like *@loriipops* on the 4th of September:

Want a Union of Light emblem code for #RiseofIron ? ???? RT & Follow for a chance to win! Drawing on my bday ???? 8 Sept! https://t.co/OjVj6Xmp71

Interestingly, this *tweet* by *@loriipops* was by far the most *re-tweeted tweet* on this day, which, apart from this *tweet*, saw a low amount of activity in regards to favorite count or *re-tweets* and did not feature a *tweet* by other major *Twitter* accounts.

An example for the opposite is the release day of the *Rise of Iron* expansion pack on the 20th of September, which not only saw a major increase in the amount of posted *tweets*, but also in the number of interactions. As a result, all top ranked *tweets* on this day are close to each other and to the other top ranking *tweets* in September. The most *re-tweeted tweet* on the 20th of September was posted by *@Bungie* confirming the launch of the *Rise of Iron* expansion:

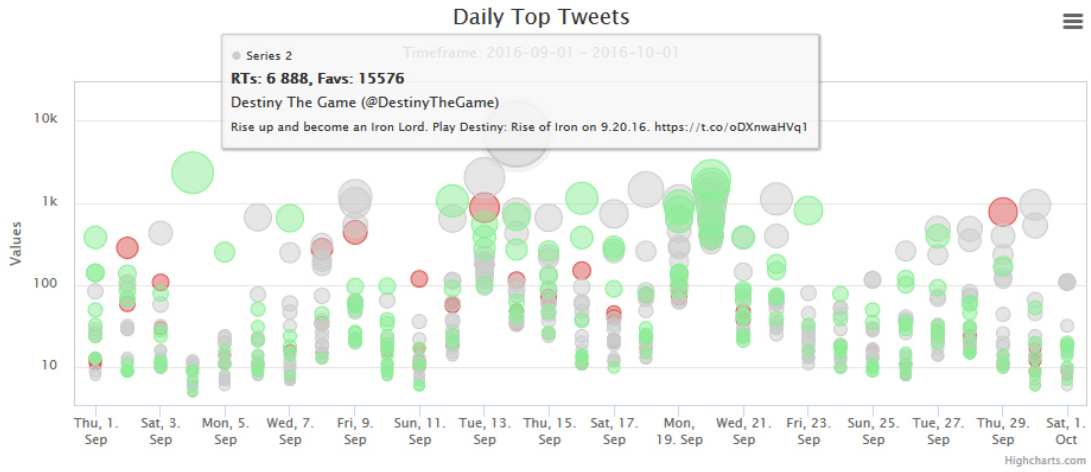
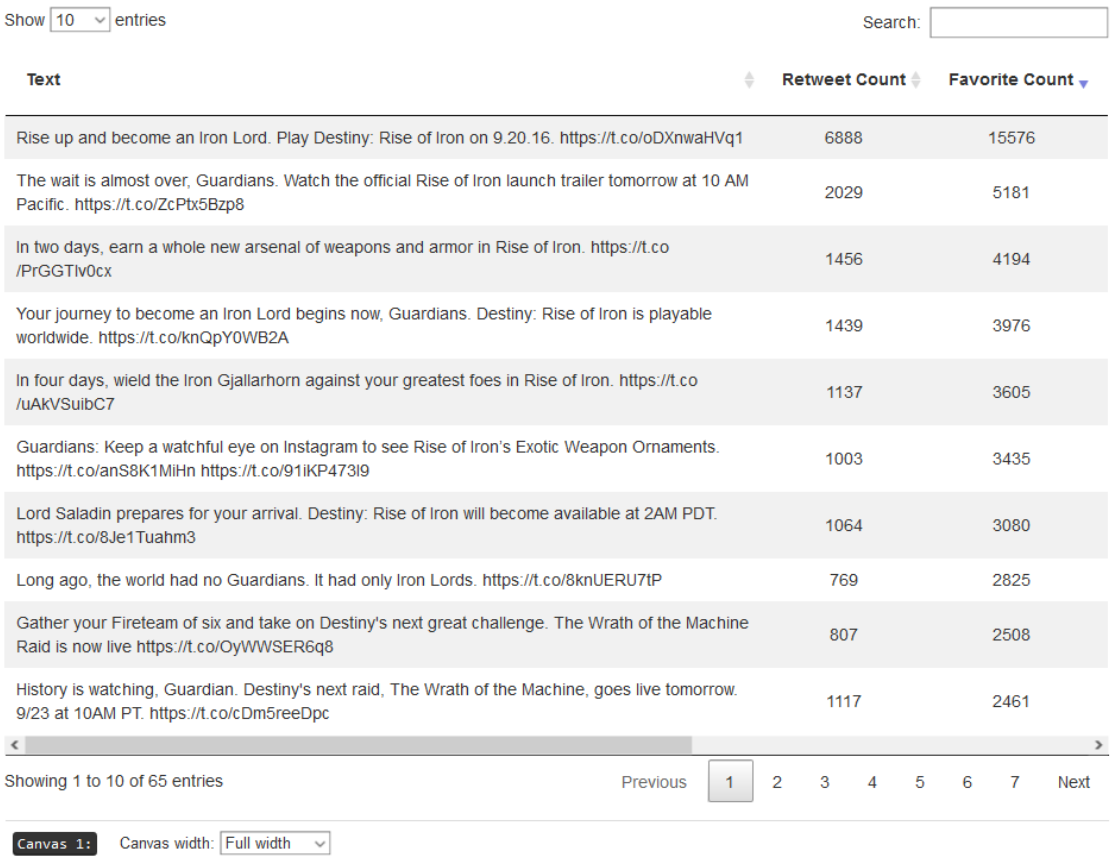


Figure 6.6: Top *tweets* from 2016-09-01 to 2016-10-01 including the overlay containing information about the *tweet* with the most *re-tweets*.

Destiny: Rise of Iron is go for launch. Return to orbit and begin your next adventure. Become an Iron Lord. https://t.co/Q2fooufhKl

In addition to that, data tables (Section 5.2.5) and the possibility to list *tweets* based on search terms, user names and date can be used in order to conduct more extensive qualitative research. Figure 6.7, for example, lists all *tweets* from the *Twitter* data sample posted by the official *@DestinyTheGame Twitter* account in September 2016 ordered by favorite count: 67 *tweets* have been posted mostly containing marketing material like videos and images about new or upcoming in-game content, multiple *re-tweets*, as well as information and reminders about challenges and events. Ordering the list by date also gives a nice overview of topics throughout September and might also provide insights about the marketing strategy leading up to the release of the *Rise of Iron* expansion.


Show 10 entries Search:

Text	Retweet Count	Favorite Count
Rise up and become an Iron Lord. Play Destiny: Rise of Iron on 9.20.16. https://t.co/oDXnwaHVq1	6888	15576
The wait is almost over, Guardians. Watch the official Rise of Iron launch trailer tomorrow at 10 AM Pacific. https://t.co/ZcPtx5Bzp8	2029	5181
In two days, earn a whole new arsenal of weapons and armor in Rise of Iron. https://t.co /PrGGTIv0cx	1456	4194
Your journey to become an Iron Lord begins now, Guardians. Destiny: Rise of Iron is playable worldwide. https://t.co/knQpY0WB2A	1439	3976
In four days, wield the Iron Gjallarhorn against your greatest foes in Rise of Iron. https://t.co /uAkVSuibC7	1137	3605
Guardians: Keep a watchful eye on Instagram to see Rise of Iron's Exotic Weapon Ornaments. https://t.co/anS8K1MiHn https://t.co/91iKP473l9	1003	3435
Lord Saladin prepares for your arrival. Destiny: Rise of Iron will become available at 2AM PDT. https://t.co/8Je1Tuahm3	1064	3080
Long ago, the world had no Guardians. It had only Iron Lords. https://t.co/8knUERU7tP	769	2825
Gather your Fireteam of six and take on Destiny's next great challenge. The Wrath of the Machine Raid is now live https://t.co/OyWWSER6q8	807	2508
History is watching, Guardian. Destiny's next raid, The Wrath of the Machine, goes live tomorrow. 9/23 at 10AM PT. https://t.co/cDm5reeDpc	1117	2461

Showing 1 to 10 of 65 entries Previous 1 2 3 4 5 6 7 Next

Canvas 1: Canvas width: Full width

Figure 6.7: Data table with all *tweets* posted by *@DestinyTheGame* from 2016-09-01 to 2016-09-30 ordered by favorite count.

6.2.2 *Twitter* vs in-game data

After focusing on *Twitter* data in the previous section, this section leverages the in-game data sample in order to find interesting relations to the *Twitter* data sample. Again, the amount of daily *tweets* over time visualization in Figure 6.1 is a good place to start and find days suggesting interesting occurrences. The *tweets* vs in-game data visualization is the next step, which can reveal a variety of information.

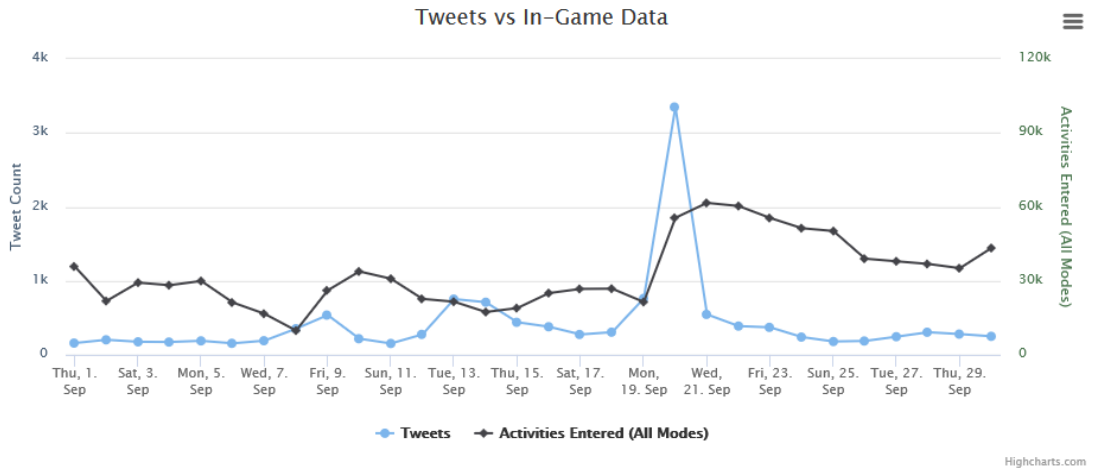


Figure 6.8: *Twitter* vs in-game data visualization of activities entered across all game modes from 2016-09-01 to 2016-09-30

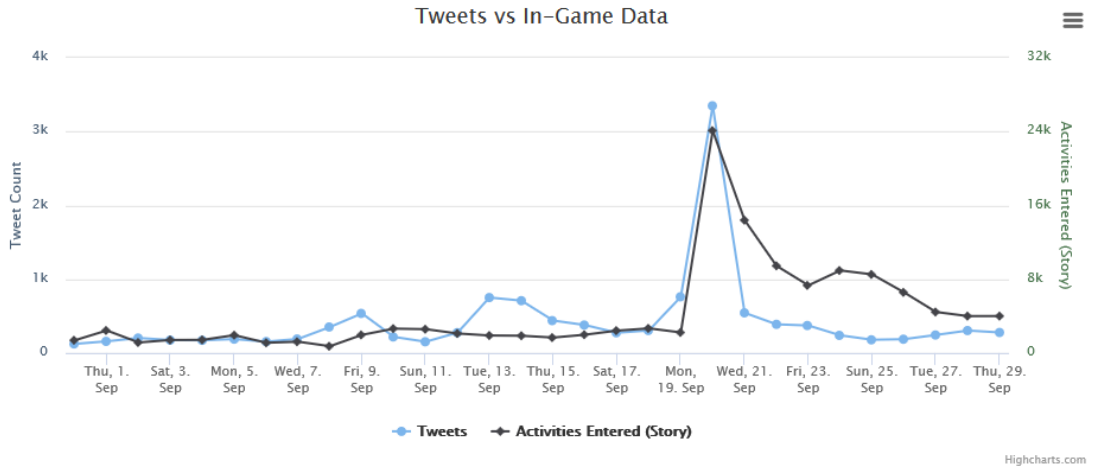


Figure 6.9: *Twitter* vs in-game data visualization of story activities entered from 2016-09-01 to 2016-09-30

Although the *Twitter* data in Figure 6.8 is limited to the *Twitter* accounts for which in-game data was accumulated, it nonetheless contains the massive spike in the amount of *tweets* posted on the 20th of September 2016. At the same time, the amount of activities players have entered across all available game modes (Section 4.4), also increased and stayed at a higher level compared to the days leading up to the 20th of September.

As listed in Table 4.6, there is a vast amount of metrics that are supported by this application and the *tweets* vs in-game data visualization: Figure 6.9, for example, shows

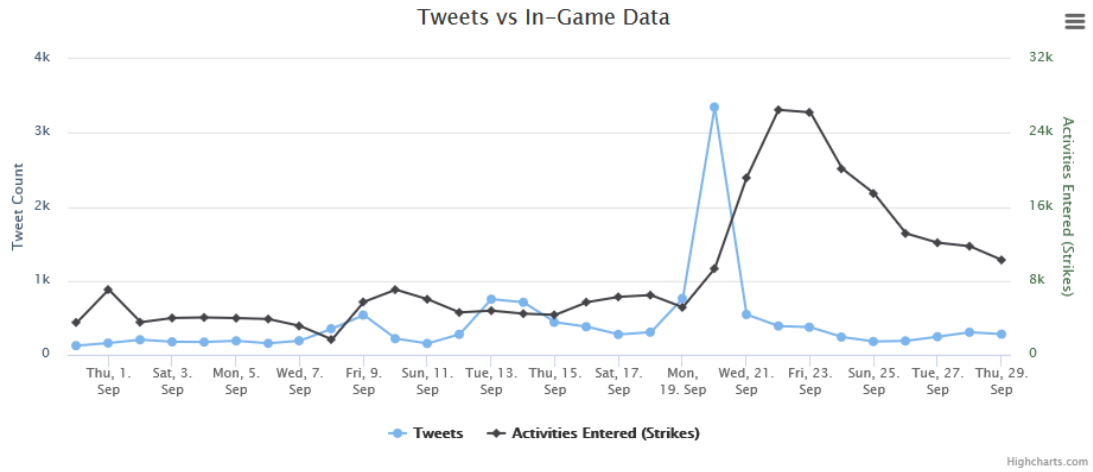


Figure 6.10: *Twitter* vs in-game data visualization of strike activities entered from 2016-09-01 to 2016-09-30

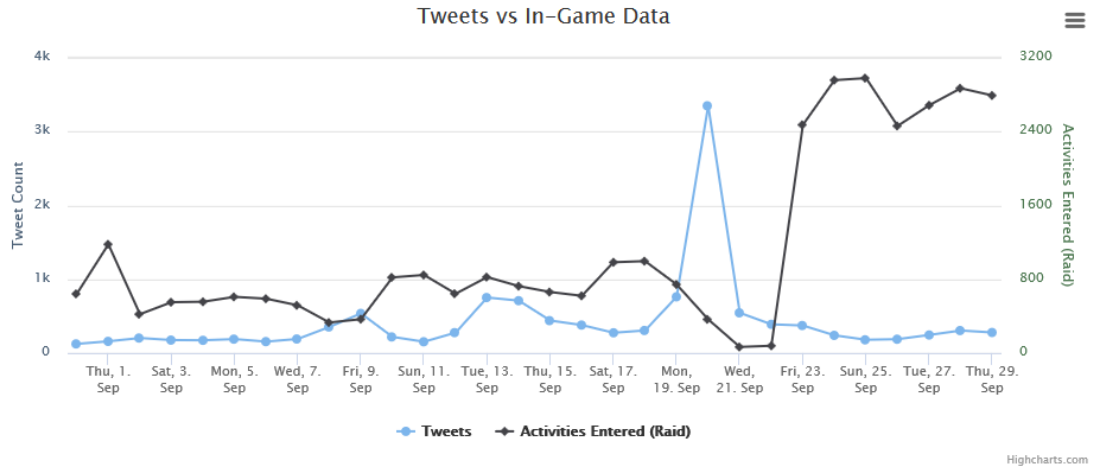


Figure 6.11: *Twitter* vs in-game data visualization of raid activities entered from 2016-09-01 to 2016-09-30

a more than 10-fold increase in activities entered related to the story game mode on the 20th of September, but also a steep decline in the days to follow. Figure 6.10 also shows a spike in activities entered related to the strikes game mode, but slightly delayed when compared to the abrupt spike of story related activities in Figure 6.9. Even more interesting is the amount of raid activities entered, as shown in Figure 6.11: while the amount of activities fell to very low numbers on the 20th, 21st and 22nd, it saw an immense rise on the 23rd of September 2016 and stayed at these high levels after that.

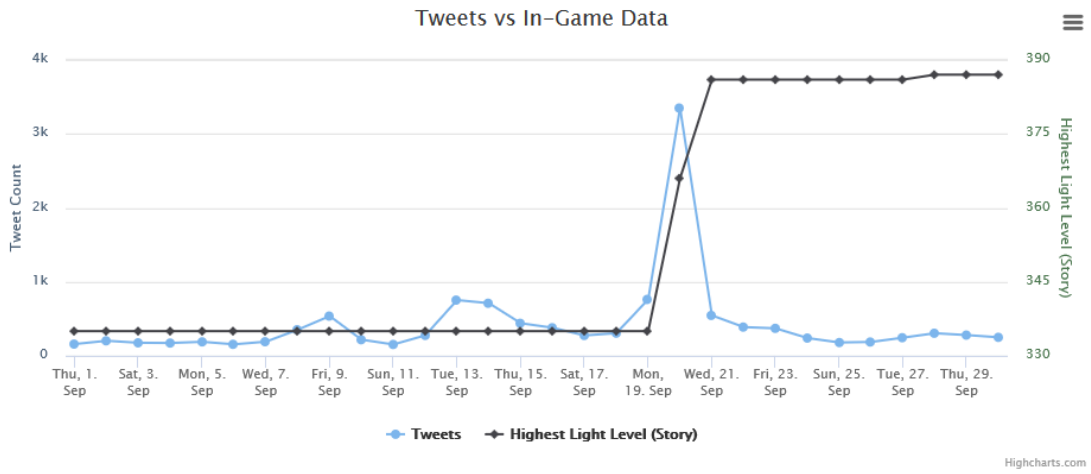


Figure 6.12: *Twitter* vs in-game data visualization of the highest light level in story mode from 2016-09-01 to 2016-09-30

Show 10 entries Search:

User	Followers	Tweets (overall)	XBL Account	PSN Account	Tweets (in sample)	RTs (avg)	Favorites (avg)
PlayStation (@PlayStation)	13977765	23587			5	745.80	384.00
Xbox (@Xbox)	11798890	171330			2	644.50	412.50
Mashable (@mashable)	8821852	237012			3	17.67	15.67
IGN (@IGN)	5861913	91550			5	108.20	253.20
Twitch (@Twitch)	4243800	15467			2	145.50	480.00
Sony (@Sony)	4086076	18050			2	174.00	96.00
COMMON (@common)	3658424	10247			2	34.50	336.00
GameSpot (@gamespot)	3314871	79853			5	49.20	113.40
Forbes Tech News (@ForbesTech)	2599859	96227			1	3	6
Game Informer (@gameinformer)	2299163	58469			7	37.43	96.86

Showing 1 to 10 of 39,132 entries Previous 1 2 3 4 5 ... 3914 Next

Figure 6.13: Data table containing *Twitter* metrics of users active between 2016-09-01 and 2016-09-30 ordered by the amount of followers.

When taking the differences of *Destiny's* game modes into account, which were discussed in Section 4.4, this paints the following picture of what happened on the 20th of September 2016: while the general increase in in-game activities suggests the release of new in-game content, the spike of story related activities is a strong indication for additional quests for the story mode. Since this is followed by a delayed rise in strike and raid related activities, also those two modes received additional content. These three visualizations also clearly show, that players started out with the story mode, before switching to strikes, which is content for mid-level players, and finally raids, which is the game mode designed for high-level players.

Figure 6.12 shows another indication for added in-game content: the highest light level in story mode, which is derived from a player's gear, increased during the 20th of September and the days after that from 335 to the new maximum of 387. In other words, new high-level items such as weapons and armor pieces have been added to the game, which were looted and equipped by players during this time period in order to increase their avatar's light level.

6.2.3 Identifying influencers

Two data tables can be used in order to identify influencing accounts: the first one is aimed at identifying influencers based on *Twitter* activities. As can be seen in Figure 6.13, the list of active accounts from the 1st to the 30th of September is ranked by the amount of followers, which puts well-known *Twitter* accounts such as the official *@PlayStation* and *@Xbox* accounts to the top, followed by media powerhouses and entertainment accounts like *@mashable*, *@IGN* or *@Twitch*. Ordering the table by other metrics, such as the amount of *tweets* or average amount of *re-tweets* and favorites within the current data sample in the chosen time period, reveals other *Twitter* accounts that do not have the highest follower numbers, but have posted *tweets* that resulted in high user engagement.

The second data table is focused on *Twitter* accounts with existing in-game activities. In addition to *Twitter* data, it offers a multitude of in-game metrics and makes it possible to find users that excel in-game, but also have a strong presence on *Twitter*. Figure 6.14, for example, shows a part of the data table which lists all users that have been active on *Twitter* and in-game in September 2016 ordered by the amount of kills in the PvP mode. The list reveals three users among the top ten that have more than 1,000 followers on *Twitter*. Changing the order of those metrics, such as the kill/death ratio in PvP mode, which is, in combination with the activities entered in PvP or the total playtime in PvP, a strong metric regarding the competitive skill of the player, brings other accounts to the top and ultimately enables an analyst to find influencing users within a specific sub-domain of the in-game data sample.

It also makes it possible to compare users and their metrics to each other by using the *tweets* vs in-game data scatterplot, which was discussed in Section 5.2.2. Figure 6.15 contains all users that posted two or more *tweets* in September 2016 and positions the data points according to the amount of posted *tweets* and of activities entered in PvP

mode. Two accounts are highlighted in this visualization: *@Swizze94*, who is ranked highest when sorting the top players data table of Figure 6.14 by the amount of kills in PvE mode, and *@itspervy*, who takes the top spot when sorting by the amount of kills in PvP mode. Unsurprisingly, *@itspervy* had spent way more time in PvP mode, joining a total of 17,359 PvP activities in September, while *@Swizze94* only entered 4,637 PvP activities. These numbers support the case that *@Swizze94* might be a less experienced PvP player then *@itspervy*. Inspecting more metrics, such as the average kills per PvP

Show 10 entries Search:

Activities Entered (PVP)	Activities Entered (PVE)	Combat Rating (PVP)	Kills (PVP)	Kills PGA (PVP)	Kills (PVE)	Kills PGA (PVE)	K/D (PVP)
17359	3655	114.464	219225	12.6289	235062	64.3124	1.9076
16625	3128	110.069	203996	12.2704	181581	58.0502	1.84565
15561	4367	117.611	203794	13.0965	337093	77.191	1.50636
12437	3444	128.626	183087	14.7212	242394	70.3815	1.95572
11306	3424	128.847	178444	15.7831	202139	59.0359	1.84062
17732	7945	86.9625	177123	9.98889	592025	74.5154	1.46218
11000	4674	129.99	172514	15.6831	323077	69.1222	2.06638
10102	3272	150.111	170311	16.8591	147017	44.9318	2.52387
12963	7697	116.263	168326	12.9851	498021	64.7033	2.15248
9653	3117	146.961	164036	16.9933	270574	86.8059	2.04948

Showing 1 to 10 of 2,973 entries Previous 1 2 3 4 5 … 298 Next

Figure 6.14: Data table containing *Twitter* and in-game metrics from users active between 2016-09-01 and 2016-09-30 ordered by the amount of kills in PvE mode.

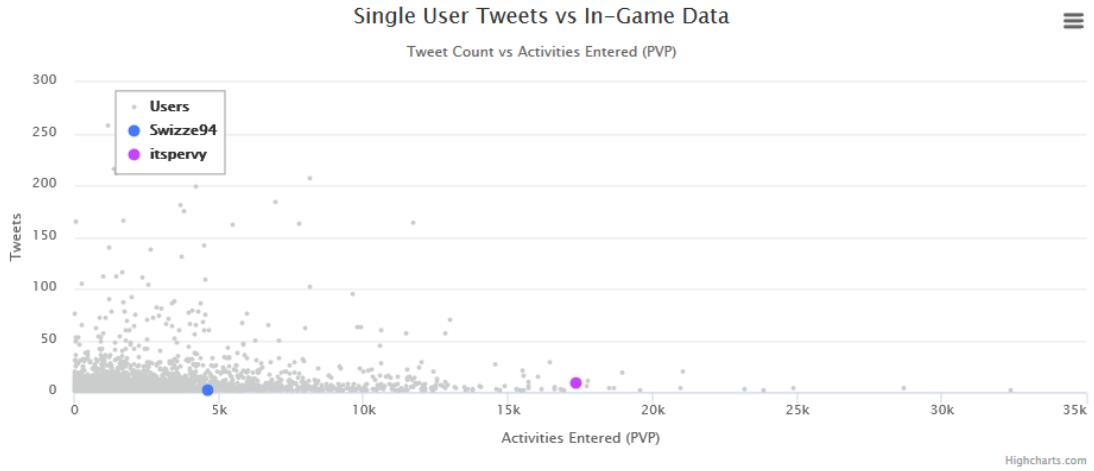


Figure 6.15: Scatterplot of *tweets* vs PvP activities entered from 2016-09-01 to 2016-09-30 highlighting *@Swizze94* and *@itspervy*.

game in Figure 6.16, reveals that in September, they performed nearly at an even level with *@itspervy* accumulating 12.6289 kills on average per game, while *@Swizze94* ranks slightly higher with 12.7026 kills on average per game. Since kills are only one aspect regarding a player's performance, the average amount of deaths per PvP game is a great addition: As Figure 6.17 shows, *@itspervy* is indeed a more experienced PvP player with only 6.62 deaths per game on average compared to *@Swizze94*, who died 10.67 times per PvP game on average.

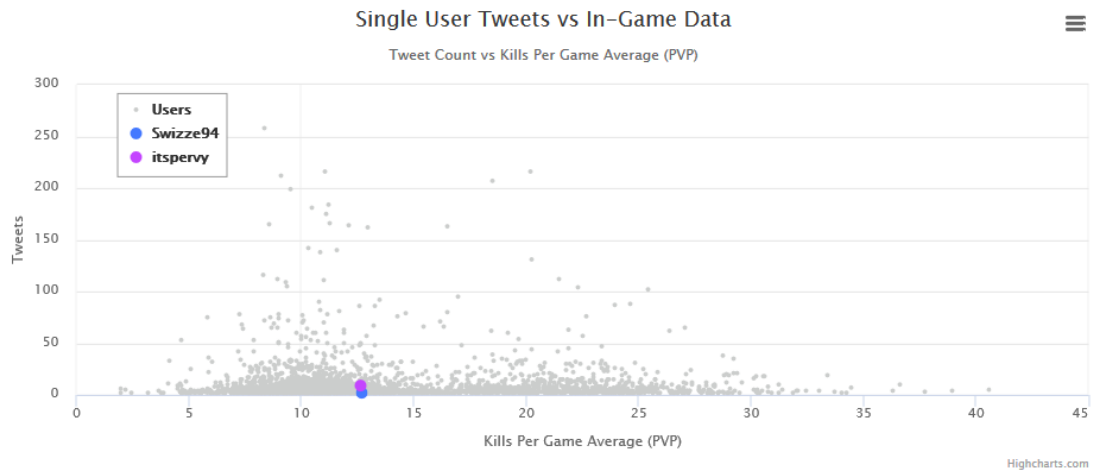


Figure 6.16: Scatterplot of *tweets* vs kills per game average in PvP mode from 2016-09-01 to 2016-09-30 highlighting *@Swizze94* and *@itspervy*.

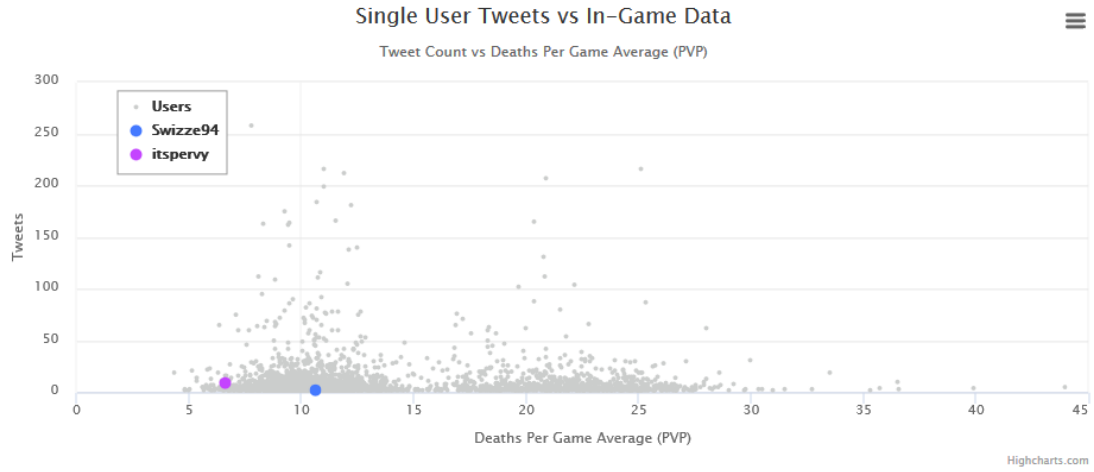


Figure 6.17: Scatterplot of *tweets* vs deaths per game average in PvP mode from 2016-09-01 to 2016-09-30 highlighting *@Swizze94* and *@itspervy*.

But not only accounts with in-game data can be compared to each other. Starting with the data table containing a list of all active *Twitter* users in September 2016 in Figure 6.13, the favorite count vs *re-tweet* count scatterplot can be used to compare the overall *Twitter* activities and the performances of each *tweet* within the defined period of time. Figure 6.18 shows a scatterplot of *tweets* posted in September 2016 that received more than ten *re-tweets* and favorites, and highlights all *tweets* by *@PlayStation*, which is the top ranked account according to the amount of followers, and the official *@DestinyTheGame* account. The visualization reveals that *tweets* of both accounts performed very well compared to the *tweets* of all other users. But *@DestinyTheGame* was clearly more active in this data sample and posted more *tweets* with higher *re-tweet* and favorite counts than *@PlayStation*.

To sum up, this chapter introduced the game *Destiny*, and the reasons why it was chosen as a use case for this work: it is a high-profile video game developed by *Bungie*, with a broad fan base and a lot of expectations even before the launch of the game. It was known that it is planned as a series of four games with multiple expansion packs. At the time this work started, the first installment as well as three expansion packs had been released, initially receiving only mixed reviews, but improving a lot with each expansion pack. Therefore, a lot of activity on *Twitter* was to be expected and eventually proved to be true.

After that, this chapter presented usage scenarios in order to show how the application and its visualizations can be used to explore different aspects of the *Twitter* and in-game data, as well as a comparison of *tweet* activities and in-game metrics to each other. By inspecting multiple visualizations, the context of what was going on in certain periods of time can be understood.

Although the given examples are mostly based on events that resulted in a significant

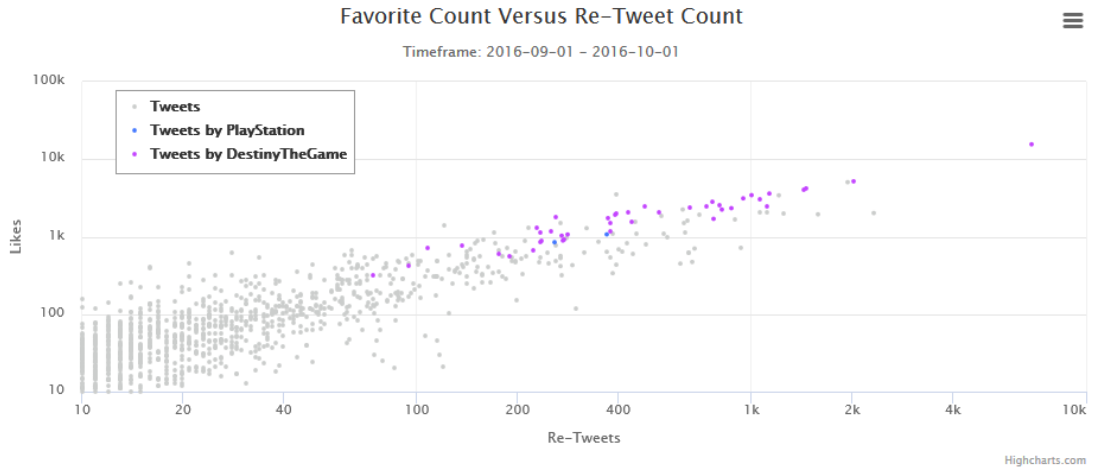


Figure 6.18: Scatterplot of favorite vs *re-tweet* count of *tweets* posted between 2016-09-01 and 2016-09-30 highlighting the *tweets* of *@PlayStation* and *@DestinyTheGame*.

impact on the overall amount of *Twitter* and in-game activity such as the release of the *Rise of Iron* expansion pack on the 20th of September 2016, this application also works on a much smaller scale and can help uncovering events and their impact on a day-to-day basis.

CHAPTER 7

Discussion, Future Work and Limitations

After presenting the application of this work in Chapters 3, 4 and 5, and showing examples of exploration of multivariate data sets in Chapter 6, this chapter discusses the findings of this work by answering the research questions introduced in Chapter 1. This is followed by a future work section, which addresses ideas and features that would extend and enhance the usefulness of the application, before concluding with a discussion about the limitations the application faces at its current state.

7.1 Discussing the research questions

In total, this work is based on three comprehensive research questions:

1. "How can long-term microblogging data be used in order to analyze the behavior of a product's users?" aims at analyzing activities of users on the microblogging service *Twitter* over a longer period of time in order to explore behavioral aspects such as the influence of user accounts due to the amount of followers and other criteria, or the sentiment of user's *tweets* in relation to the usage of a product.

2. "Can microblogging activities be connected to a product's usage data?" delves deeper into the relation of *Twitter* data and a product's usage data.

3. "How can multivariate long-term data be visualized?" focuses on presenting the diverse data set in a meaningful way.

All three questions are discussed in deeper detail in the subsequent sections.

7.1.1 How can long-term microblogging data be used in order to analyze the behavior of a product's users?

With the growth of social media platforms and microblogging services in recent years, the amount of data related to the usage of a product or service has also increased. In order to utilize this data for purposes such as collecting feedback or analyzing a market or a community, suitable metrics related to the selected purpose have to be identified. With the goal to provide an easy-to-use, yet powerful tool for analysts, this work chose a set of basic metrics that are listed in Section 4.2 in full detail.

The following list gives a brief overview of the most important metrics used in the application of this work:

- Timestamps: the date and time of when a *tweet* was posted represents the main ingredient for all temporal analysis and visualizations. The application of this work aggregates the daily number of *tweets* based on the timestamp of their creation, which then represents the overall *Twitter* activity of this day.
- *Tweet* text: the text of a *tweet* contains user-generated content and is used to analyze the sentiment of the *tweet*, to generate word clouds. It is also visualized as a whole in order to provide qualitative context.
- Biography: the biography is a profile description written by the user and is analyzed in order to find user names related to the usage of a product or service such as XBL or PSN.
- *Re-tweet*/Favorite/Follower counts: these *Twitter* specific metrics are indicators for the significance of a *tweet* or a user account.

Using these metrics opens up many possibilities to gain insights into the behavior of a product's user base: the combination of activity charts and contextual analysis methods such as sentiment analysis, data tables and word clouds, allows analysts to not only get overviews quickly and identify events that have an impact on *Twitter* activities, but also gain insights about the context of an event down to the *tweets* of single users. To give more examples of utilizing microblogging data for the analysis of the behavior of a product's users, each of the following sections addresses a different aspect related to this first research question.

Can influencing microblogging users be identified?

As discussed in Section 2.2 of the related work section, detecting influencers is currently a high-interest area within the scientific community. Each social network or microblogging platform has some form of inter-user connection such as friends lists or subscription-based systems in place. Naturally, there are user accounts that have evolved in to so-called influencers, which are individuals that have accumulated large numbers of active followers.

They are not only reaching a lot of users with their own content, but also have an amplifying impact on the reach of information when sharing content of other users by, for example, using *Twitter's re-tweet* feature. From the viewpoint of a product developer or service provider, identifying such user accounts can be crucial due to their influence within the target audience. Electronic word-of-mouth, discussed in Section 2.9, is an interesting, related topic, which suggests that influencers can make or break a product or service.

Therefore, a lot of research has been done related to the process of identifying influential users on social media and microblogging platforms resulting in sophisticated frameworks such as the flow path based data structure by Subbian et al.[SAS16] or the *NCFinder* by Mazumder et al.[MMP15]. Identifying influencers in the application of this work is not based on such advanced frameworks and is kept rather simple by using interactive data tables as discussed in Section 5.2.5 and illustrated by an example in Section 6.2.3. The reason for choosing simplicity over a more sophisticated approach is linked to the amount of different metrics, especially when combining *Twitter* data and in-game data: finding users that, for example, perform very well in PvP requires other metrics than users that spent a lot of time in PvE. Therefore, a complex and highly configurable approach is needed, which would have gone beyond the scope of this work, but surely is an interesting challenge in the future, as will be discussed in Section 7.3.

To summarize and answer the question: yes, influencing microblogging users can be identified by using the sortable and filterable data tables of this application, which enable an analyst to create lists of user accounts ranked by a variety of *Twitter* and in-game data metrics. While the given examples and the gaming-related data set of this work contains very domain specific metrics, the approach itself is valid for other domains and the used metrics are certainly exchangeable.

How is the sentiment overall?

Sentiment analysis is one of the core components of the application of this work and as Araújo et al. [AGCB14] showed, it is another heavily researched area with a lot of different approaches and projects. Many of them are focusing on sentiment analysis of social media and microblogging data by solving problems related to its fast-paced and multivariate nature. This application utilizes the *Sentiment140* tool (Section 3.1.6), as well as the *salient* library (Section 3.1.9).

As can be seen in Figure 5.4, Figure 5.6, Figure 5.8 or Figure 6.4, many visualizations of this application are directly based on the sentiment values and utilize color-coding to display differences in sentiment, with neutral values being visualized as gray, negative as red and positive as green. Using this approach makes it possible to examine overviews and, as the example in Figure 6.2 shows, find events that seem to have a big impact on the overall sentiment by looking out irregularities and changes in the composition of neutral, positive and negative *tweets*. In addition, data can be explored in detail by, for example, comparing word clouds of different sentiment settings to each other, such as

Figure 6.3 and Figure 6.4 in Section 6.2.1, in order to learn more about the context of *tweets* classified as positive or negative.

Therefore, the question of how the overall sentiment is, can only be answered on a day to day basis and strongly depends on local events: Figure 7.1 presents the sentiment of daily *tweets* over time from the 1st of July 2016 to the 30th of November 2016. The following three days are marked in order to give examples: the a) 7th of July 2016 represents the day with the highest percentage of positive *tweets* in this time period with 53.67% of the *tweets* being classified as positive. Exploring the context by checking the word cloud of this day, which is shown in Figure 7.2, reveals that the 7th of July is the so-called *Bungie Day*, a day which is celebrated by *Bungie* and its community every year[1]. On the b) 20th of September 2016 the *Rise of Iron* expansion pack has been released, which has been discussed in Section 6.2.1 and saw an increase in negative classified *tweets* due to server

[1] https://www.bungie.net/en/News/Article/44934/7_HAPPY-BUNGIE-DAY, last accessed: 2018-02-19

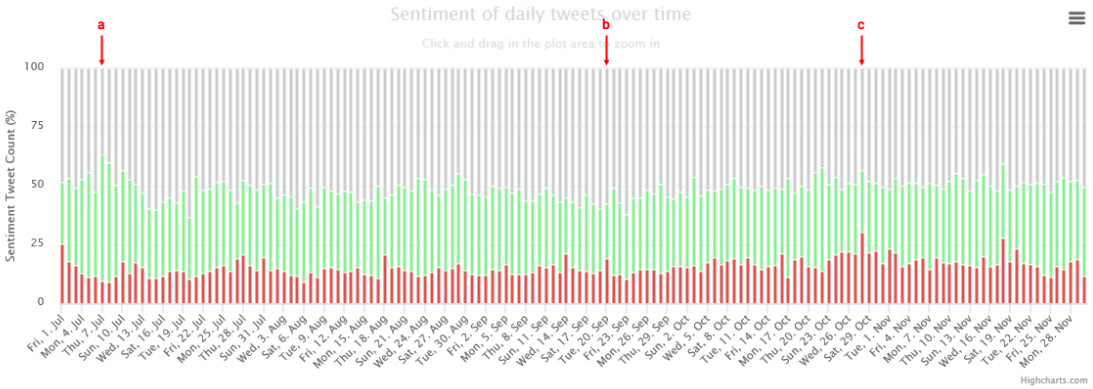


Figure 7.1: Sentiment of daily *tweets* over time from 2016-07-01 to 2016-11-30 color-coded with neutral *tweets* being gray, positive green and negative red.

Figure 7.2: Word cloud of positive classified *tweets* on 2016-07-01.

Figure 7.3: Word cloud of negative classified *tweets* on 2016-10-28.

outages. Finally, the c) 28th of October 2016 represents the day with 30.46 percent in negative classified *tweets*, the highest value in this time period. The context analysis via word cloud, as depicted in Figure 7.3, and data tables shows, that the negative *tweets* were focused on *cheaters* and *Bungie's* strategy to deal with them. Gaming website *@Polygon*, for example *tweeted*:

Cheaters won't prosper for long in Destiny's Trials of Osiris mode, says Bungie. https://t.co/FLexI5R85p

Many others shared their frustration about the topic, such as *@MSandersonD*:

This is why I dislike PvP Destiny: Bungie has already banned some of the Trials of Osiris DDoS cheaters you reported https://t.co/4DriVf2hYB

Or *@originaldrdoom*, who wrote:

Fucking laggers/cheaters are out in full force in Trials. @Bungie @BungieHelp @DestinyTheGame

Does the sentiment change after the release of product changes or updates?

As discussed previously, the sentiment is strongly influenced by local events. While the release of major product changes or updates, such as the launch of the *Rise of Iron* expansion, certainly influences the sentiment on the release day as well as the days leading up to and following the launch, a significant shift in the sentiment on a larger scale can not be determined. A reason for this could be the limitation of the data sample, because only one major release took place during the time period the data sample for this work was accumulated, and this release additionally was received as only a mediocre update, as discussed in Section 6.1.1.

On a smaller scale, multiple updates and patches have been released addressing bugs and other issues, adjusting the balancing of characters and weapons, and generally improving the quality of the game[2]. However, as the analysis of such updates reveals, they were accompanied with many voicing their frustration about server downtime, large download files or the changes of updates themselves.

The update on the 14th of June 2016, for example, resulted in *tweets* like:

I have been a longtime advocator of @DestinyTheGame, but @Bungie's nerf to Hunters in update 2.3.0 is aggravating. https://t.co/PxuOShHlKC

[2]http://destiny.wikia.com/wiki/List_of_Destiny_Updates, last accessed: 2018-02-19

by *@Subcult619*

So many angry #huntards after this update lol (despite the fact that they got massive PvE BUFFS!!!) #Destiny https://t.co/INwUpi1anh

by *@Machiavoriel*

Destiny is acting crazy since today's update @BungieHelp . Every player in crucible is invisible. Many friends can't launch out of orbit.

by *@joecrowe08*

The update on 23rd of June 2016 resulted in *tweets* such as:

Can we all just collectively agree that Destiny updates do not function properly and ask @Bungie to address it? Enough is enough.

by *@IamProvocateur_*

was having a killer crucible match and right towards the end i get kicked out for a destiny update thanks a ton @Bungie

by *@WhispInTheW1nd*

@Bungie why is my PS4 telling me that this update is 10GB and continues to fail to download? #Help #Destiny #Update

by *@JmzMc*

To give one last example, the updates on the 26th and 28th of July 2016 resulted in *tweets* such as:

I keep getting beavered while trying to get to the tower. Wasn't an issue before today's update. #Destiny @Bungie @DestinyNews_net

by *@Frizzwise*

Tried to revisit Destiny tonight but it looks like Bungie has broken its own game with the latest update ???????? https://t.co/UaZepPOVWB

by *@tomphillipsEG*

Fuck sake #Destiny #Bungie Your recent update has bugs, not me!!! https://t.co/abUMtYQW6A

by *@TheGAMERer*

It can be summarized, that besides sharing update news and patch notes, people tend to post negative *tweets* aimed at certain issues associated with the update process itself like server downtimes or downloading large files, but also about broken things they encountered in-game. Constructive criticism as well as acknowledgments are sparse, especially on the day of a release. Furthermore, when exploring the content of top *tweets* on such a release day, they are mostly unrelated to the update itself, but are more aimed at promoting the game: on the 14th of June 2016, for example, the top *tweet* by the official *@DestinyTheGame* account introduced a new enemy type, followed by multiple *tweets* about *Bungie's* presentation at *Sony's E3* press conference. The update itself was only briefly addressed by the official *@BungieHelp* account announcing the scheduled maintenance including links to *Bungie's* website giving more insights into the server maintenance and update status.

Therefore, changes in sentiment can be detected, but deeper analysis is required to fully understand its context, since an overview, as shown in Figure 7.1, by itself is not sufficient: major server outages or other in-game problems that affect a lot of players indeed increases the amount of negative classified *tweets*, but if on the same day other announcements or community activities take place, which are classified as positive and spread throughout *Twitter*, the overall sentiment of this day can still be positive.

Does the developer react to user feedback and how?

Analyzing the *tweet* activities by the three official *Bungie* accounts *@Bungie*, *@BungieHelp* and *@DestinyTheGame* reveals that *Bungie's* interactions on *Twitter* are limited to occasional *re-tweets*, but they do not react to complaints or questions by users. The 30th of June and the 1st of July 2016, for example, saw a lot of *tweets* related to server issues and people unable to connect to the *Destiny* servers which directly tagged the official accounts in their *tweets*, such as *@iLynXxEU's tweet*:

@BungieHelp @Bungie are you aware of the issues that players are having when trying to sign in to destiny?

While all of those *tweets* remain unanswered, the *@BungieHelp* account publishes *tweets* related to issues affecting a larger amount of people, such as following *tweet* on the 1st of July 2016 including notes for affected people to turn to the help section of their forum linked in the *tweet*:

Services in Destiny are returning to normal. If you're still unable to play, tell us about it here: https://t.co/cOAIctERNC

Acknowledgements of known issues are also posted on *Twitter*, like the *tweet* by *@BungieHelp* on the 19th of July 2016:

> *Destiny Character Data is currently unavailable on https://t.co/4ucwiEYycJ and the Destiny Companion App. We are investigating.*

However, whether those reactions are a result of user feedback posted on *Twitter* or not, can not be determined.

Although this question is very specific to this work's use case, *Bungie's* unknown strategy regarding the usage of social media networks and microblogging platforms to gather user feedback or provide support is an example nonetheless. It is a legitimate strategy to use *Twitter* mainly as a one-way communication channel with a focus on promotions and status updates, especially since *Bungie* hosts a community forum with dedicated feedback and help sections[3].

In general, *Twitter* can very well be used to get user feedback or provide support: the official *@XboxSupport Twitter* account, for example, answers questions by users and helps resolving user's problems[4]. Users, as well as product developers or service providers can benefit greatly from including social media networks and microblogging platforms into the feedback and support strategy. On the one side, users are already used to share their opinions and problems on these platforms. On the other side, developers and service providers can react quickly and transparent to mentioned problems. Even if these issues are more complex or take longer to resolve, just a public reaction to a user's post shows a higher degree of customer care.

Are there regional differences in user feedback?

As briefly discussed in Section 2.3, the amount of *tweets* that contained geo-location or *Twitter* places in the collected data sample of this work was very low: only 1.42% of the 1,062,390 collected *tweets* had been tagged with places, which conforms with the results of the work of Leetaru et al.[LWC+13]. They revealed, that only 2.02% of the *tweets* of their data sample contained location meta data.

As research projects such as Li and Sun's *PETAR*[LS14], Ferracani et al.'s *LiveCities* [FPD14] or Abbasi et al.'s framework[ARMW15] demonstrate, it is certainly possible to extract geo-location data from the content of a *tweet*. However, this approach does only work if people talk about real events and locations, such as specific restaurants, cinema visits, and so on. Therefore, an approach of extracting geo-location data from the content of *tweets* for the use case of this work would only work for a minority of *tweets* referencing events and event locations. To give an example, a *tweet* by *@ThatTomHam*

[3] Bungie.net's feedback section: https://www.bungie.net/en/Forums/Topics?tg=Feedback, last accessed: 2018-04-27

[4] The official @XboxSupport Twitter account: https://twitter.com/XboxSupport/with_replies, last accessed: 2018-04-28

posted on the 2nd of September 2016 mentions the *Washington State Convention Center* and links to a photo on the author's Instagram[5] account:

I want that. #DestinytheGame #TouchofMalice #PaxWest2016 @ Washington State Convention Center https://t.co/awCsrzFRSV

Since the majority of *tweets* in the data sample of this work is about news related to *Destiny*, community activities or in-game topics, which invalidates any content-based analysis for extracting geo-location data, and due to the sparse amount of geo-location data already present in the *Twitter* data sample, any approach to utilize geo-location data has not been pursued further. Therefore, this question remains unanswered at the current state of this work.

7.1.2 Can microblogging activities be connected to a product's usage data?

The use case of this work shows that it is possible to connect microblogging activities to the usage data of a certain product by extracting account information from a *Twitter* user's biography. However, the chosen approach is not fully automatic, which impacts the time required for executing the process and ultimately results in a lower amount of user accounts.

In addition to that, the approach might also work best in the online gaming domain since this user group is especially engaged on *Twitter* according to Bateman [Bat16], with gamers tending to share their account names on social media and microblogging services in order to connect to other players. For non-gaming domains, it may be less likely to find account names in microblogging data and other approaches are required to establish a link between a product's or service's usage data and social media networks or microblogging platforms, which will be discussed in more detail in the following section.

Can account names of a product's users be identified through microblogging activities?

As elaborated in Section 3.1.5, XBL gamertags and PSN IDs can be extracted from the biography of *Twitter* users, which is a freely customizable text that can be added to a *Twitter* account by the user. Although the extraction process could also be applied on *tweets* directly, the approach of using just the biography was chosen, because a gamertag mentioned in a *tweet* must not necessarily mean that this gamertag is related to the author of the *tweet*. However, it is almost certain that a gamertag mentioned in a user's biography belongs to the user. The resulting list of XBL gamertags and PSN IDs represents a direct link from *Twitter* data to the in-game data sample described in Section 4.4.

[5] https://www.instagram.com, last accessed: 2018-02-23

As mentioned above, identifying account names in microblogging data might work inferior or even not work at all outside of the domain of online games. Therefore, an approach is required that yields sound results domain-independently. An example would be a reversal of the approach applied by this work: if user accounts are available in a product's or service's usage data, they could be used to identify users on social media networks and microblogging platforms. Additionally, many products and services facilitate account registration and logins via social media networks or microblogging services, or simply allow users to add their social media and microblogging account names to the user profile of this product or service.

Does higher microblogging activity correlate with increased usage of a product?

A higher microblogging activity does not necessarily correlate with an increased usage of a product. As discussed in Section 6.2.2, there are events that lead to an increased amount of in-game as well as *Twitter* activity, such as the release of the *Rise of Iron* expansion pack on the 20th of September 2016, which is shown in Figure 6.8 or Figure 6.9. However, there are also events that cause a higher activity on *Twitter*, but a lower amount of in-game activity: the 14th of June 2016, for example, which saw an in-game update as mentioned in Section 7.1.1, had a reduced amount of *in-game activities entered* across all game modes, while at the same time, the amount of *tweets* surged. Analysis of word cloud and the *tweets* data table reveals that the higher amount of *tweets* is mostly related to *Bungie's* appearance on stage presenting *Rise of Iron* at the E3 2016, while a small amount of *tweets* was related to the maintenance server downtime due to the update.

Therefore, microblogging activity does generally not correlate with the amount of in-game activities, but instances can be found that substantiate an increase or decrease in in-game activity by analyzing the context of microblogging data. Of course, changes in the amount of in-game activities could also be caused by events that do not appear on *Twitter*.

To sum up, the use case of this work shows that it is possible to connect microblogging activities to in-game data by extracting gamertags from a *Twitter* user's biography and therefore establish a link between *Twitter* and the in-game data set. While there are instances in which a higher microblogging activity correlates with increased in-game activities, such as the release of the *Rise of Iron* expansion, a more general correlation can not be determined. A reason for that might be the fast-paced and diverse nature of microblogging services such as *Twitter*, which can see a lot of different topics on a single day making it harder to link a certain event to in-game activities. To give an example, a player might not be able to play the game on a certain day because of a scheduled maintenance and might express disappointment about the server downtime on *Twitter*, but can at the same time spread excitement and anticipation about an upcoming DLC release or participate in discussions and community events.

7.1.3 How can multivariate long-term data be visualized?

A lot of research has already been done in the area of visualizing social media and microblogging data. Many of them set the focus on a specific scope, such as Kaye et al.'s *Nokia Internet Pulse* [KLJ+12] or Castellanos et al.'s *LivePulse* [CGL+11], which utilize sentiment analysis and color-coded word clouds to represent activities in real-time. Others, like Morstatter et al. [MKLM13], Malik et al. [MSH+13] or Dewan et al. [DGGK13] took a broader approach and combined multiple visualization techniques in order to provide an overview but also allow an analyst to dig deeper into a topic to understand the context or how a topic progresses over time. While many aspects of these frameworks can in some form be found in the application of this work, it also takes another step and visualizes microblogging data in combination with a secondary data set, the *Destiny* in-game data as described in Section 4.4. This makes it possible to explore reciprocal effects and to back up events found in *Twitter* data by the in-game data set. As the usage scenarios focusing on the in-game data set in Section 6.2.2 showed, many events that are talked about on *Twitter* are in some form reflected by in-game activities or provide even more details: the release of the *Rise of Iron* DLC on the 20th of September 2016, for example, saw a spike in both, *Twitter* and in-game activity. By using *Twitter* data alone, the contextual analysis of this event only confirms the release itself. Analyzing the in-game data, however, reveals not only an expected increase in overall in-game activity, but also a spike in story mode activity while the activity of other game modes increases with a delay on the following days. As discussed and visualized by the examples presented in Section 6.2.2, this suggests that the DLC added new content to each game mode and that the players start out with the story mode before switching to more challenging game modes.

Section 5.2 gave an overview about all visualizations that are supported by the application of this work in its current state: activities over time visualizations such as Figure 5.3 or Figure 5.5 form the backbone of the application and are usually the origin for each analysis. Sentiment analysis in combination with color-coding also plays an important part for a variety of visualizations such as Figure 5.4 or Figure 5.8. To explore the context of in-game and microblogging activities, word clouds such as Figure 5.9 and data tables such as Figure 5.10 are utilized. The comparison between different metrics is another aspect of the application, as can be seen in Figure 5.7 or Figure 5.6.

Since multivariate long-term data can be visualized in many different ways, this application was designed to provide a basic set of visualizations that allow analysts to quickly get an overview, but also dig deeper into the context of microblogging activities in combination with a second data set in the form of *Destiny* in-game data.

7.2 Limitations

At its current state, the application of this work faces various limitations. First of all, the *Twitter* data sample has been collected with the basic, free version of *Twitter's* Search API, which returns only a small data sample of the overall activities sampled by *Twitter* itself,

which might be biased. Morstatter et al. [MPL14, MPLC13] extensively investigated this bias and could show that the data sample collected with the free version of the Streaming API indeed can be inaccurate in comparison to the full data set provided by *Twitter's* premium service if the overall volume is low. In order to resolve this limitation, *Twitter's* premium or enterprise service could be used, which allows access to the complete data set[6].

Another limitation is the process of extracting gamertags, which includes a manual step, since gamertags are part of the biography of a *Twitter* user's profile. While the fact that biographies are freely written by users themselves already poses a challenge, which is bypassed by searching only for user accounts whose biography contains acronyms or words such as "gamertag", "GT", "XBL" or "PSN", gamertags do not follow a specific pattern that can be automatically detected. Although it is unclear if this approach itself can be completely automated at all, there are other approaches that could resolve this limitation: by starting out with a data set of in-game accounts, for example, the *Twitter* Search API can be used to find *tweets* or user accounts containing the gamertag in order to identify *Twitter* accounts of in-game users.

A third limitation is the sentiment analysis itself and the tools used by the application of this work: as discussed in Section 2.1, processing the content of *tweets* faces challenges due to *Twitter's* character limit and the increased use of abbreviations and uncommon grammar constructions. Additionally, environment specific wording, such as slang words used by gamers or terms from *Destiny* itself aggravate the sentiment analysis and result in inaccurate values. As an example, the following *tweet* posted by *@tlovetech* on the 1st of July 2016 was classified as negative by the *salient* toolkit (Section 3.1.9), but positive by the *Sentiment140* tool (Section 3.1.6):

Switching up to do a @DestinyTheGame Raid with my Team! Come and Join us as we murder some Taken! https://t.co/0DptU0pzbi

There are two approaches to reduce inaccurate analysis results: i) Like Yu and Wang [YW15], a more advanced sentiment analysis framework could be developed and trained based on the already existing data set and an extended word-emotion association lexicon, which would greatly enhance its ability to handle environment specific language. As an alternative and similar to Araújo et al.'s approach with *iFeel*[AGCB14], ii) more existing tools could be added to the application of this work in order to use the tool yielding the best results for a specific domain, or to calculate the mean of the resulting values, which would reduce the amount of inaccuracies.

[6] https://developer.twitter.com/en/docs/tweets/search/api-reference/premium-search, last accessed: 2018-03-01

7.3 Future Work

The application of this work in its current state provides the basic functionality to collect a data sample by utilizing the free Search API provided by *Twitter* (Section 3.2), to analyze and process (Section 4.3.4), export (Section 3.5) and import (Section 3.4) data sets, and to visualize data in various forms (Section 5). It clearly facilitates an explorative approach to analyze microblogging data standalone or in combination with an additional data set consisting of *Destiny* in-game activities.

In Section 7.2, certain limitations were discussed including approaches how to resolve them: while the limitation resulting from the usage of the free version of the *Twitter* Search API can be eliminated by paying for the premium API in order to get access to the full, historical data set, other limitations require additional work.

Extracting gamertags fully automatically poses a great challenge that should be addressed in order to speed up the process of identifying players based on *Twitter* profiles and therefore would greatly increase the amount of in-game data. A gamertag is an identifier specific to gaming platforms such as XBL or PSN, and since gamers are becoming the most engaged group of users on *Twitter* according to Bateman [Bat16], it is not that surprising that *Twitter* users share their gamertags on *Twitter*. This might not be the case for other domains. Therefore, detecting user names of products and services in microblogging activities and profile data such as the biography, could be a lost cause and another approach might yield better results: starting out with a product's or service's data set containing user account names, as argued in Section 7.2, can prove to be effective for identifying *Twitter* users.

In order to tackle sentiment analysis inaccuracy, further sentiment analysis tools can be added to the application to provide analysts with a selection of tools to chose from. Alternatively, a custom-built sentiment analysis framework could be developed, featuring possibilities to train the analysis algorithm with domain-specific data.

Apart from issues mentioned in the limitations section, there are various other aspects that can be improved. In its current state, a lot of the application's functionality is focused on handling the *Destiny* in-game data in all its forms. In order to increase the usefulness of the application it is mandatory to loosen this dependency and move to a more general approach which is able to handle all kinds of data by, for example, providing a powerful configuration that enables an analyst to customize the data structure of a data set before it is imported and processed.

Next, a sophisticated reporting functionality has to be added enabling an analyst to export customizable reports. While the visualizer, which was discussed in Chapter 5, is designed to support multiple visualizations on a single page that can be exported by using the browser built-in print functionality, a lot can be improved in order to enhance its usability, especially in regards to reporting. As explained in Section 5.1.1, visualizations are rendered in customizable canvases. In order to allow faster changes in the structure of a report, these canvases could be extended with a drag and drop functionality enabling

an analyst to quickly change the positions of visualizations without re-rendering. In order to improve the story-telling aspect of such a report, the visualizer can be extended to provide the possibility to add and edit notes, highlight data points and other parts of a visualization, as well as define special canvases that can hold fully customizable content. Finally, the visualizer should not only enable an analyst to export the finished report as PDF, but also to save and load a report.

Moreover, the visualizer should be further extended with additional, more advanced visualization types. Graphs and trees can be used to visualize relations between data points in order to gain insights about how topics and trends evolve over time similar to Malik et al.'s *TopicFlow* [MSH+13], how users are connected with each other or how information spreads on *Twitter* via the *re-tweet* functionality.

Finally, the application can benefit greatly from extending the crawler (Section 3.2) to support more social media platforms and microblogging services besides *Twitter*. As research projects such as Dewan et al.'s *MultiOSN* [DGGK13] or Scharl et al.'s *Westeros Sentinel* [SHHJ+16] showed, using multiple platforms as data sources results in a lot more meta data, especially if profiles of a user can be identified on different social media platforms and microblogging services: data unavailable on *Twitter*, such as geo-location data, might be available on other platforms like *Facebook*, *Google+*, or *YouTube*. Many platforms provide additional meta data that can be utilized: *Facebook*, for example, allows its users to tag posts with their current mood, which could be used as an important factor when analyzing the sentiment of a post. Also, reactions of other users to a post are more versatile than on *Twitter* since it is possible to specify a sentiment such as love, sad or angry when using the like feature, which could be relevant for sentiment analysis.

It should also be mentioned that although the application conforms to *Twitter's* developer terms[7], the crawling and data import process should be adjusted to make sensitive personal data fully anonymous.

To summarize, this chapter first elaborated on the research questions, which this work is based on: all questions have been discussed and answered, with the exception of the regional differences in user feedback, which was not answered because the approach of using geo-location data was not pursued further due to the sparse amount of data. After that, the limitations of this work have been addressed, which mainly consist of having an incomplete data set due to the usage of the basic, free version of *Twitter's* Search API, the gamertag extraction process not being fully automatic, and the used sentiment analysis tools occasionally resulting in inaccurate values. Finally, possible further steps and features of the application of this work have been discussed ranging from general usability enhancements and a more sophisticated reporting functionality to extending the crawler of this application to support additional social media and microblogging services, and adding more advanced visualization types to facilitate deeper insights.

[7]Developer Agreement and Policy: https://developer.twitter.com/en/developer-terms/agreement-and-policy.html, last accessed: last accessed: 2018-12-02

CHAPTER 8

Conclusion

Social media networks and microblogging platforms took the world by storm in recent years, and the importance of this new form of communication grew within the scientific community. A lot of research has been done revolving around social media and microblogging data ranging from strategies and proposals related to the data collection process, frameworks aimed at the analysis of different aspects such as the detection of influencers, the usage of location-based data or the detection of events, to applications and tools using various visualization techniques in order to represent social media and microblogging data in a certain way. In addition to that, research projects have been discussed that are related to the setting of this work such as (electronic) word-of-mouth, gaming communities and user feedback.

This work set out to explore the interdependency of microblogging data and product or service usage by implementing an application that analyzes microblogging activities related to a product, puts usage data of a product in relation to this data and presents various visualizations. *Twitter* was used as source for microblogging data, whereas the popular video game *Destiny* was utilized as a use case. A basic overview of the application of this work has been given, beginning with an introduction of frameworks and libraries that have been used developing this application, before discussing its four components in greater detail: while the importer and exporter component handle the functionality related to processing of the second data set, the crawler component deals with all things concerned with collecting, processing and storing information fetched from the *Twitter* API, and the visualizer component loads and presents data using different visualization types.

This is followed by a discussion of *Twitter* and the data structure of *tweets*, the in-game data sample and how both data samples have been processed and stored by the application of this work. After *tweets* are returned from *Twitter's* Search API data is extracted from the results in order to form four data objects (tweet, user, user profile and place). In addition to that, the sentiment of each *tweet* is analyzed using the *Sentiment140* service

and the *salient* toolkit, before storing all data objects in the database. Using this process over a period of about 14 months resulted in a data set containing 1,062,390 *tweets* from 246,881 users. The profile descriptions of these users have been analyzed in order to extract gamertags linking *Twitter* accounts to XBL or PSN accounts. A secondary data set was then collected by colleagues of the *University of York* and *Fraunhofer IAIS* containing the in-game data of 3,548 players from a time period of about 6 months. Processing this in-game data set resulted in three different data formats containing selected in-game activities for each user over the complete period of time, in-game activities aggregated for each player per day and game mode, as well as a per-day-total of all in-game activities for all players combined.

A detailed look at the visualizer component of the application of this work gave insights into the structure of the web client, explained how the configuration works and what kind of visualization types are supported in the application's current state. Besides visualizing different sets of data as a line chart on a timeline, which can be used to quickly gain an overview of activities, scatter plots, bubble charts, word clouds and data tables can be utilized in order to explore the context of *Twitter* and in-game activities. Based on that, various usage scenarios of such explorative approaches are presented, displaying multiple ways how analysts can leverage the application of this work to gain insights into certain events causing irregularities in *Twitter* or in-game activities.

Finally, the research questions are revisited and the findings of this work are discussed. Although the application of this work is in many ways not as advanced as research projects focusing on a single aspect related to social media and microblogging data, it combines a broad selection of analysis and visualization techniques and allows an analyst to explore *Twitter* data, but also *Destiny* in-game data and relations between the two data sets. To summarize the key findings of this work:

- By using sortable and filterable data tables, influential *Twitter* users as well as players can be identified. Combining various *Twitter* and in-game metrics enables an analyst to find influencers in specific domains, such as players that perform very well in PvP and have accumulated a lot of followers.
- As contextual analysis related to the sentiment of *tweets* on a daily basis revealed, the overall sentiment per day can be deceiving since it is strongly dependent on single events. Releases of game updates, for example, are mostly accompanied with server downtimes, causing a lot of users to voice their frustration on *Twitter* which results in a higher number of negative-classified *tweets*.
- While certain events such as the release of a big DLC led to a simultaneous increase or decrease in *Twitter* and in-game activity, a general correlation between these two metrics could not be determined. Due to *Twitter* emphasizing a fast-paced information transfer, people tend to post about a lot of different topics across a single day, which makes it more challenging to link specific events to in-game activities.

- It is possible to identify XBL and PSN account names via *Twitter* data and thereby establish a link from a user's *Twitter* account to their in-game data.

In addition to that, this work could confirm the outcome of Leetaru et al.'s research [LWC+13], which revealed that only 2.02% of the *tweets* in their data sample contained geo-location meta data. Due to this sparse amount of geo-location data, the approach to explore regional differences in user feedback was discontinued. Furthermore, this work also investigated if and how the developer *Bungie* reacts to feedback posted on *Twitter*, but could not find any data supporting the initial assumption that user feedback via *Twitter* influences the developer's work. However, there are instances when one of the official *Twitter* accounts posts links to the support forums or acknowledgments about known issues.

The limitations this work are addressed as well as possible solutions to resolve those limitations. The *Twitter* data set might contain bias since it was collected with the free, basic version of the *Twitter* Search API, an issue that can be prevented by switching to the pricey premium API. The sentiment analysis tools currently used can result in inaccurate sentiment values due to *Twitter's* increased use of abbreviations and uncommon grammar constructions, as well as environment specific wording related to *Destiny*. Resolving this limitation might require a higher effort such as developing a custom sentiment analysis framework as proposed by Yu and Wang [YW15]. Alternatively, more sentiment analysis tools could be added to the application of this work and allow analysts to choose a tool producing the best results in their domain similar to Araújo et al.'s *iFeel* [AGCB14]. Lastly, the process of extracting XBL gamertags and PSN IDs from *Twitter* biographies is not fully automatic and might require a completely different approach.

This work concludes by discussing the future work. Apart from resolving the aforementioned limitations, the current dependency of the application on handling *Destiny* in-game data should be reduced in order to be able to process data sets more generically. The usability of the visualizer can also be enhanced by adding a sophisticated reporting functionality allowing analysts to manipulate the dashboard by easily changing the structure using drag and drop, or appending customizable content such as notes, highlighted areas and content. Besides adding features to improve the usability, the application could benefit greatly by extending core components such as the visualizer by adding additional visualization types such as maps, trees or graphs, and the crawler itself by supporting other social media networks or microblogging platforms besides *Twitter*.

List of Figures

List of Tables

Bibliography

[AGCB14] Matheus Araújo, Pollyanna Gonçalves, Meeyoung Cha, and Fabrício Benevenuto. iFeel: A Web System that Compares and Combines Sentiment Analysis Methods. *e Int. World Wide Web Conf. Comm.*, pages 75–78, 2014.

[ARMW15] Alireza Abbasi, Taha Hossein Rashidi, Mojtaba Maghrebi, and S. Travis Waller. Utilising Location Based Social Media in Travel Survey Methods. In *Proc. 8th ACM SIGSPATIAL Int. Work. Locat. Soc. Networks - LBSN'15*, pages 1–9, New York, New York, USA, nov 2015. ACM Press.

[Asl17] Salman Aslam. Twitter by the numbers: Stats , demographics & fun facts. https://www.omnicoreagency.com/twitter-statistics/, aug 2017.

[Bat16] Samuel Bateman. How to launch a video game on Twitter. https://blog.twitter.com/marketing/en_gb/a/en-gb/2016/how-to-launch-a-video-game-on-twitter.html, oct 2016.

[BE08] danah m. Boyd and Nicole B. Ellison. Social Network Sites: Definition, History, and Scholarship. *J. Comput. Commun.*, 13(1):210–230, oct 2008.

[BL11] Dejana Bajic and Kelly Lyons. Leveraging social media to gather user feedback for software development. In *Proceeding 2nd Int. Work. Web 2.0 Softw. Eng. - Web2SE '11*, pages 1–6, New York, New York, USA, may 2011. ACM Press.

[BSDB16] Ana Babić Rosario, Francesca Sotgiu, Kristine De Valck, and Tammo H.A. Bijmolt. The Effect of Electronic Word of Mouth on Sales: A Meta-Analytic Review of Platform, Product, and Metric Factors. *J. Mark. Res.*, 53(3):297–318, jun 2016.

[BSU14] Christopher Buschow, Beate Schneider, and Simon Ueberheide. Tweeting television: Exploring communication activities on Twitter while watching TV. *Communications*, 39(2):129–149, jan 2014.

[Bun] Bungie Inc. Destiny the Game | Destiny 1 Home. https://www.destinythegame.com/d1.

[Bun17] Bungie Inc. Documentation for BungieNet.Platform.DestinyServices. https://www.bungie.net/d1/platform/Destiny/help/, 2017.

[CCCJ15] Chao Chen, Fuhai Chen, Donglin Cao, and Rongrong Ji. A Cross-media Sentiment Analytics Platform For Microblog. In *Proc. 23rd ACM Int. Conf. Multimed. - MM '15*, pages 767–769, New York, New York, USA, oct 2015. ACM Press.

[CGL+11] Malu Castellanos, Riddhiman Ghosh, Yue Lu, Lei Zhang, Perla Ruiz, Mohamed Dekhil, Umeshwar Dayal, and Meichun Hsu. LivePulse: Tapping Social Media for Sentiments in Real-Time. In *Proc. 20th Int. Conf. companion World wide web - WWW '11*, page 193, New York, New York, USA, mar 2011. ACM Press.

[CHC+14] Taejoong Chung, Jinyoung Han, Daejin Choi, Taekyoung Ted Kwon, Huy Kang Kim, and Yanghee Choi. Unveiling group characteristics in online social games. *Proc. 23rd Int. Conf. World wide web - WWW '14*, pages 889–900, 2014.

[CKB+17] Despoina Chatzakou, Nicolas Kourtellis, Jeremy Blackburn, Emiliano De Cristofaro, Gianluca Stringhini, and Athena Vakali. Measuring #GamerGate: A Tale of Hate, Sexism, and Bullying. In *Proc. 26th Int. Conf. World Wide Web Companion - WWW '17 Companion*, pages 1285–1290, New York, New York, USA, 2017. ACM Press.

[Del13] Deloitte. Tweets for Sales Gaming. Technical Report April, Twitter UK Ltd, 2013.

[DGGK13] Prateek Dewan, Mayank Gupta, Kanika Goyal, and Ponnurangam Kumaraguru. MultiOSN: Realtime Monitoring of RealWorld Events on Multiple Online Social Media. In *Proc. 5th IBM Collab. Acad. Res. Exch. Work. - I-CARE '13*, pages 1–4, New York, New York, USA, oct 2013. ACM Press.

[DYNM06] Nicolas Ducheneaut, Nicholas Yee, Eric Nickell, and Robert J. Moore. "Alone together?": exploring the social dynamics of massively multiplayer online games. In *Proc. SIGCHI Conf. Hum. Factors Comput. Syst. - CHI '06*, pages 407–416, New York, New York, USA, 2006. ACM Press.

[DYNM07] Nicolas Ducheneaut, Nicholas Yee, Eric Nickell, and Robert J. Moore. The life and death of online gaming communities: a look at guilds in world of warcraft. In *Proc. SIGCHI Conf. Hum. Factors Comput. Syst. - CHI '07*, page 839, New York, New York, USA, 2007. ACM Press.

[FPD14] Andrea Ferracani, Daniele Pezzatini, and Alberto Del Bimbo. User Profiling for Urban Computing. In *Proc. 3rd ACM Multimed. Work. Geotagging Its Appl. Multimed. - GeoMM '14*, pages 17–20, New York, New York, USA, nov 2014. ACM Press.

[GBH09a] Alec Go, Richa Bhayani, and Lei Huang. General Information - sentiment140.com. http://help.sentiment140.com/, 2009.

[GBH09b] Alec Go, Richa Bhayani, and Lei Huang. Twitter Sentiment Classification using Distant Supervision. Technical report, Stanford, dec 2009.

[GCHL13] Xingyu Gao, Juan Cao, Qin He, and Jintao Li. A novel method for geographical social event detection in social media. In *Proc. Fifth Int. Conf. Internet Multimed. Comput. Serv. - ICIMCS '13*, page 305, New York, New York, USA, aug 2013. ACM Press.

[GSZS14] Oshini Goonetilleke, Timos Sellis, Xiuzhen Zhang, and Saket Sathe. Twitter Analytics: A Big Data Management Perspective. *ACM SIGKDD Explor. Newsl.*, 16(1):11–20, sep 2014.

[HGK+15] Kieran Hicks, Kathrin Gerling, Ben Kirman, Conor Linehan, and Patrick Dickinson. Exploring Twitter as a Game Platform; Strategies and Opportunities for Microblogging-based Games. *Proc. 2015 Annu. Symp. Comput. Interact. Play - CHI Play '15*, pages 151–161, 2015.

[HHK17] Youngsub Han, Beomseok Hong, and Kwangmi Ko Kim. Super Bowl Live Tweets: The Usage of Social Media during a Sporting Event. In *Proc. 8th Int. Conf. Soc. Media Soc. - #SMSociety17*, pages 1–5, New York, New York, USA, 2017. ACM Press.

[HHW+13] Orland Hoeber, Larena Hoeber, Laura Wood, Ryan Snelgrove, Isabella Hugel, and Dayne Wagner. Visual Twitter Analytics: Exploring Fan and Organizer Sentiment During Le Tour de France. *Proc. VIS 2013 Work. Sport. Data Vis.*, (September):1–7, 2013.

[HLCH12] Liang-Chi Hsieh, Ching-Wei Lee, Tzu-Hsuan Chiu, and Winston Hsu. Live Semantic Sport Highlight Detection Based on Analyzing Tweets of Twitter. *2012 IEEE Int. Conf. Multimed. Expo*, pages 949–954, 2012.

[HTWF15] Thorsten Hennig-Thurau, Caroline Wiertz, and Fabian Feldhaus. Does Twitter matter? The impact of microblogging word of mouth on consumers' adoption of new movies. *J. Acad. Mark. Sci.*, 43(3):375–394, may 2015.

[JBSR16] Philipp Jordan, Wayne Buente, Paula Alexandra Silva, and Howard Rosenbaum. Selling out the magic circle: free-to- play games and developer ethics. In *Proc. 1st Int. Jt. Conf. DiGRA FDG*. Digital Games Research Association and Society for the Advancement of the Science of Digital Games, 2016.

[KGL+16] Alfredo Kalaitzis, Maria Ivanova Gorinova, Yoad Lewenberg, Yoram Bachrach, Michael Fagan, Dean Carignan, and Nitin Gautam. Predicting Gaming Related Properties from Twitter Profiles. In *Proc. - 2016 IEEE 2nd Int. Conf. Big Data Comput. Serv. Appl. BigDataService 2016*, pages 28–35, 2016.

[KH10] Andreas M Kaplan and Michael Haenlein. Users of the world, unite! The challenges and opportunities of Social Media. *Bus. Horiz.*, 53(1):59–68, 2010.

[KLJ+12] Joseph 'Jofish' Kaye, Anita Lillie, Deepak Jagdish, James Walkup, Rita Parada, and Koichi Mori. Nokia Internet Pulse: A Long Term Deployment and Iteration of a Twitter Visualization. In *Proc. 2012 ACM Annu. Conf. Ext. Abstr. Hum. Factors Comput. Syst. Ext. Abstr. - CHI EA '12*, pages 829–844, New York, New York, USA, 2012. ACM Press.

[KLK13] Hwi-Gang Kim, Seongjoo Lee, and Sunghyon Kyeong. Discovering hot topics using Twitter streaming data. In *Proc. 2013 IEEE/ACM Int. Conf. Adv. Soc. Networks Anal. Min. - ASONAM '13*, pages 1215–1220, New York, New York, USA, aug 2013. ACM Press.

[KRP16] Thatchaphon Klomklao, Panat Ratanarungrong, and Santi Phithakkitnukoon. Tweets of the Nation: Tool for visualizing and analyzing global tweets. In *Proc. 2016 ACM Int. Jt. Conf. Pervasive Ubiquitous Comput. Adjun. - UbiComp '16*, pages 1349–1357, New York, New York, USA, 2016. ACM Press.

[KSB+16] Robert Krueger, Guodao Sun, Fabian Beck, Ronghua Liang, and Thomas Ertl. TravelDiff: Visual comparison analytics for massive movement patterns derived from Twitter. In *2016 IEEE Pacific Vis. Symp.*, pages 176–183. IEEE, apr 2016.

[KWJL11] Peter Kraker, Claudia Wagner, Fleur Jeanquartier, and Stefanie Lindstaedt. On the way to a science intelligence: Visualizing TEL tweets for trend detection. In *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, volume 6964 LNCS, pages 220–232, 2011.

[LS11] James Lanagan and Alan F. Smeaton. Using Twitter to Detect and Tag Important Events in Sports Media. *Proc. Fifth Int. AAAI Conf. Weblogs Soc. Media*, jul 2011.

[LS14] Chenliang Li and Aixin Sun. Fine-grained location extraction from tweets with temporal awareness. In *Proc. 37th Int. ACM SIGIR Conf. Res. Dev. Inf. Retr. - SIGIR '14*, pages 43–52, New York, New York, USA, jul 2014. ACM Press.

[LWC+13] Kalev Leetaru, Shaowen Wang, Guofeng Cao, Anand Padmanabhan, and Eric Shook. Mapping the global Twitter heartbeat: The geography of Twitter, apr 2013.

[Mac17] Amanda MacArthur. The Real History of Twitter, In Brief. https://www.lifewire.com/history-of-twitter-3288854, nov 2017.

[McC16] Harry McCracken. A Brief History Of Twitter's 140-Character Limit. https://www.fastcompany.com/3060165/a-brief-history-of-twitters-140-character-limit, 2016.

[MJR+11] Alan M. MacEachren, Anuj Jaiswal, Anthony C Robinson, Scott Pezanowski, Alexander Savelyev, Prasenjit Mitra, Xiao Zhang, and Justine Blanford. SensePlace2: GeoTwitter analytics support for situational awareness. In *VAST 2011 - IEEE Conf. Vis. Anal. Sci. Technol. 2011, Proc.*, pages 181–190. IEEE, oct 2011.

[MKLM13] Fred Morstatter, Shamanth Kumar, Huan Liu, and Ross Maciejewski. Understanding Twitter data with TweetXplorer. *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discov. data Min. - KDD '13*, page 1482, 2013.

[MM16] Brian McDonald and David Moffat. Using Sentiment Analysis to track reaction to the Global Game Jam Theme. In *Proc. Int. Conf. Game Jams, Hackathons, Game Creat. Events - GJH&GC '16*, pages 50–53, 2016.

[MMP15] Sahisnu Mazumder, Sameep Mehta, and Dhaval Patel. Identifying Top-k Consistent News-Casters on Twitter. *CIKM '15 Proc. 24th ACM Int. Conf. Inf. Knowl. Manag.*, pages 1875–1878, oct 2015.

[MPL14] Fred Morstatter, Jürgen Pfeffer, and Huan Liu. When is it Biased? Assessing the Representativeness of Twitter's Streaming API. In *Proc. 23rd Int. Conf. World Wide Web - WWW '14 Companion*, pages 555–556, New York, New York, USA, apr 2014. ACM Press.

[MPLC13] Fred Morstatter, Jürgen Pfeffer, Huan Liu, and Kathleen M. Carley. Is the Sample Good Enough? Comparing Data from Twitter's Streaming API with Twitter's Firehose. In *Proc. Seventh Int. AAAI Conf. Weblogs Soc. Media*, jun 2013.

[MSH+13] Sana Malik, Alison Smith, Timothy Hawes, Panagis Papadatos, Jianyu Li, Cody Dunne, and Ben Shneiderman. TopicFlow: Visualizing Topic Alignment of Twitter Data over Time. In *Proc. 2013 IEEE/ACM Int. Conf. Adv. Soc. Networks Anal. Min. - ASONAM '13*, pages 720–726, New York, New York, USA, 2013. ACM Press.

[New17] Casey Newton. Twitter just doubled the character limit for tweets to 280. https://www.theverge.com/2017/9/26/16363912/twitter-character-limit-increase-280-test, 2017.

[NM17] Daniel Nations and Elise Moreau. What Is Microblogging? A Definition of Microblogging with Examples. https://www.lifewire.com/what-is-microblogging-3486200, 2017.

[PAM+10] Leysia Palen, Kenneth M Anderson, Gloria Mark, James Martin, Douglas Sicker, Martha Palmer, and Dirk Grunwald. A vision for technology-mediated support for public participation & assistance in mass emergencies & disasters. *Proc. 2010 ACMBCS Visions Comput. Sci. Conf.*, pages 1–12, 2010.

[SAS16] Karthik Subbian, Charu C. Aggarwal, and Jaideep Srivastava. Querying and Tracking Influencers in Social Streams. In *Proc. Ninth ACM Int. Conf. Web Search Data Min. - WSDM '16*, pages 493–502, New York, New York, USA, feb 2016. ACM Press.

[SHHJ+16] Arno Scharl, Alexander Hubmann-Haidvogel, Alistair Jones, Daniel Fischl, Ruslan Kamolov, Albert Weichselbraun, and Walter Rafelsberger. Analyzing the public discourse on works of fiction - Detection and visualization of emotion in online coverage about HBO's Game of Thrones. *Inf. Process. Manag.*, 52(1):129–138, jan 2016.

[SJLK04] A. Fleming Seay, William J. Jerome, Kevin Sang Lee, and Robert E. Kraut. Project Massive: A Study of Online Gaming Communities. *Ext. Abstr. 2004 Conf. Hum. factors Comput. Syst. - CHI '04*, page 1421, 2004.

[SKC09] David a Shamma, Lyndon Kennedy, and Elizabeth F Churchill. Tweet the debates: Understanding Community Annotation of Uncollected Sources. In *Proc. first SIGMM Work. Soc. media - WSM '09*, page 3, New York, New York, USA, 2009. ACM Press.

[SOM13] Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. Tweet Analysis for Real-Time Event Detection and Earthquake Reporting System Development. *IEEE Trans. Knowl. Data Eng.*, 25(4):919–931, apr 2013.

[Sta17] Statista Inc. Number of monthly active Twitter users worldwide from 1st quarter 2010 to 2nd quarter 2017 (in millions). https://www.statista.com/statistics/282087/number-of-monthly-active-twitter-users/, 2017.

[Twia] Twitter Inc. Filter realtime Tweets — Twitter Developers. https://developer.twitter.com/en/docs/tweets/filter-realtime/overview.

[Twib] Twitter Inc. Rate limits — Twitter Developers. https://developer.twitter.com/en/docs/basics/rate-limits.

[Twic] Twitter Inc. Search Tweets — Twitter Developers. https://developer.twitter.com/en/docs/tweets/search/overview.

[Twi17a] Twitter Inc. Connecting to a streaming endpoint — Twitter Developers. https://developer.twitter.com/en/docs/tweets/filter-realtime/guides/connecting, 2017.

[Twi17b] Twitter Inc. Snapshot of Tweets in real-time. https://developer.twitter.com/en/products/tweets/sample, 2017.

[Twi17c] Twitter Inc. Standard search API — Twitter Developers. https://developer.twitter.com/en/docs/tweets/search/api-reference/get-search-tweets.html, 2017.

[VHSP10] Sarah Vieweg, Amanda L Hughes, Kate Starbird, and Leysia Palen. Microblogging during two natural hazards events. In *Proc. 28th Int. Conf. Hum. factors Comput. Syst. - CHI '10*, page 1079, New York, New York, USA, 2010. ACM Press.

[VSG14] George Valkanas, Antonia Saravanou, and Dimitrios Gunopulos. *A Faceted Crawler for the Twitter Service*, volume 8787 of *Lecture Notes in Computer Science*. Springer International Publishing, Cham, 2014.

[WN11] D. Yvette Wohn and Eun-Kyung Na. Tweeting about TV: Sharing television viewing experiences via social media message streams. *First Monday*, 16(3):1–14, feb 2011.

[WTCP13] Xinyue Wang, Laurissa Tokarchuk, Félix Cuadrado, and Stefan Poslad. Exploiting hashtags for adaptive microblog crawling. In *Proc. 2013 IEEE/ACM Int. Conf. Adv. Soc. Networks Anal. Min. - ASONAM '13*, pages 311–315, New York, New York, USA, aug 2013. ACM Press.

[YW15] Yang Yu and Xiao Wang. World Cup 2014 in the Twitter World: A big data analysis of sentiments in U.S. sports fans' tweets. *Comput. Human Behav.*, 48:392–400, jul 2015.