Andreas Punz

Detection and Analysis of Communities on Twitter

Bachelor's Thesis

Graz University of Technology

Knowledge Technologies Institute
Head: Univ.-Prof. Dr. Stefanie Lindstaedt

First Supervisor: Ass.-Prof. Dr. Elisabeth Lex
Second Supervisor: Dipl.Ing. Dominik Kowald, BSc.

Graz, September 2016

This document is set in Palatino, compiled with pdfLaTeX2e and Biber.

The LaTeX template from Karl Voit is based on KOMA script and can be found online: https://github.com/novoid/LaTeX-KOMA-template

Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

Graz, _____ _____

Date Signature

Eidesstattliche Erklärung[1]

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Graz, am _____ _____

Datum Unterschrift

[1] Beschluss der Curricula-Kommission für Bachelor-, Master- und Diplomstudien vom 10.11.2008; Genehmigung des Senates am 1.12.2008

Abstract

Social networks are an integral part of the daily lives of millions of people. They use these social networks, like Twitter, to find out what is happening in the world and in their social circles and to converse with like-minded peers. This leads to the formation of communities. In this thesis, a method to detect communities on Twitter is proposed. Beginning with a selected seed account, the description of the seed's followers are analyzed if they contain a specific key phrase. An implementation of this method is also provided in Java, using the Twitter4J library. Further, the communities that were detected this way are examined and compared utilizing various network metrics with the help of Gephi. However, only the followers and friends, as well as some of the profile data of the users are be analyzed. Nodes that stood out in some key metrics proofed to be valuable for further data collection.

Contents

List of Figures

1 Introduction

Twitter is a social network and microblogging service that was founded in 2006. It enables users to post short messages of up to 140 characters also known as „tweets“. The character limit is inherited from SMS messages, as Twitter was initially conceived as a web platform to share those messages. Today, Twitter is one of the biggest social networks and also among the most influential.
In general, the usage of these social network services has continuously risen during the last few years and the interaction with them is for many people a part of their daily routine. Twitter stands out from other companies in this field for a few reasons. For example there is not as much of a focus on presenting personal data on Twitter than compared to Facebook or Google+. Instead, its main use is giving its users the opportunity to send short status messages to quickly convey what is happening, how they are feeling or to converse with other users. These messages can be categorized with the use of a ”hashtag”. In fact, hashtags have become so popular that Facebook added this functionality. It also employs a different kind of relationships compared to other services. In place of the mutual “Friend” type relationship, that needs to be acknowledged by both sides, Twitter uses uni-directional ”followers” and “followings” relationships. For these reasons, one can observe conversations that take place on a global scale and expand past the boundaries of more traditional social networks, leading to the formation of communities that discuss particular topics. Java et al., 2007 were trying to find out what are the reasons and motivations of the average Twitter user to use the service. They came to the conclusion that the four main points were daily chatter, conversations, sharing information and reporting news. However, as they point out, it is really difficult to figure out an individual user’s intention. Communities, on the other side, are easier to analyze and to determine their intention. Of course, communities do not only appear on Twitter. They appear in the real world, other social networks

and also in fictional works. The blog "A game of Nodes" (Misch, 2016) attempts to analyze the popular TV and book series "Games of Thrones" utilizing social network metrics. In his work, Ben Misch tries to detect the inherent communities in "Games of Thrones" and to find out how the communities and the people within those communities affect each other. To come back to Twitter, there already have been a number of studies to find methods to detect these communities there.Highfield, Harrington, and Bruns, 2013 showed an other way of displaying communities in connection with a live event. During the Eurovison Contest they used the hashtags that were displayed by TV broadcasting organizations. Utilizing these they could construct communities, that showed not only a national but in some instances also an ethical divide. They also showed how quickly those new communities could emerge during a live event.

Greene, O'Callaghan, and Cunningham, 2012 based their method on lists. Lists are a way for Twitter users to organize the accounts they follow under a particular topic name. However, there's also the possibility for several users to share a list. This is especially useful, for example, for journalists, who can more easily find tweets about breaking news. Greene et al.'s method used this "wisdom of the crowd" to identify topical communities and find clusters of users and lists that overlap. Lim and Datta, 2012 are of the opinion that traditional methods are too computationally expensive and propose a different way. Their approach is to start with a list of celebrities with a high amount of followers and order them to a category that they can represent, for example "News" or "Music". Next, they look at the followers of these celebrities and if they follow enough celebrities that belong to a certain category to pass a predetermined threshold, they are counted as part of that community. However, the initial selection of celebrities and their categorizing has to be done very carefully, or else the results may not be particularly accurate.

This paper will try to answer the following questions:

- How can communities on Twitter be identified
- What differences can one observe between these communities

This first chapter gives an overview of the topic and the goals of this paper. The next chapter ”Method” describes how the data set was acquired and which libraries were used.
The chapter ”Empirical Analysis” explains the network metrics that were used. Following that, the chapter ”Results” analyses the data set and leads into the chapter ”Conclusion and Outlook”.

2 Method

2.1 Data Source and Used Libraries

The origin for the data in this particular data set is Twitter. Most of Twitter's data is publicly accessible by use of the Twitter API (Twitter, 2016). This API can also be triggered by external programs. The program that was used to collect the data was written in Java utilizing the Twitter4J library (Yamamoto, 2016). The reason for this was the familiarity with both the Java programming language and the Twitter4J library as well as the excellent documentation that Twitter4J provides, making the access to Twitter's data easy and efficient.

2.2 Acquisition of the Data Set

The method to acquire the data was inspired by Hadgu and Jäschke, 2014. Here, the authors used the a list of a science conference and tried to connect them to Twitter accounts to use as their initial seed accounts. At first, the program was simply able to accept a list of names as input and produce a list of Twitter accounts that correspond to these names. Following that, a seed account was selected. The first account that was used as a seed for the "Star Wars" community was "theforcenet", which had around 44 thousand followers at the time that the data was collected. Every user that followed the initial seed account was considered as a potential member of the community. The deciding factor was the user's description. In the case that it contained a certain keyword, the user's account ID was

added into the data set. This was done to ensure that the user is really interested in the topic, and does not just casually follow the seed account. The same process was repeated for two more communities, the seed accounts being ”fansofmetallica” and ”AppleMacGeek” with the keywords being ”Metallica” and ”Apple”, respectively. The crawling process for the ”Star Wars” community took place in April 2016, while the data for the other two communities were collected in July 2016. Afterwards, additional information of the acquired users' accounts was gathered, like their join data, post count and especially the list of users they follow and a list of users who follow them. This data was used to compute the metrics explained in the latter chapters. The software used for this analysis was Gephi (Gephi, 2016).

2.3 Limitations of the Data Set

However, not the full data set that was retrieved from Twitter was used for the computations. Only users with less than 1000 followers and less than 1000 followings were analyzed. The reason for this is that there are several users in each community that have 10.000 or more followers. This would lead to a humongous graph, that far exceeds what the computer that was used to analyze this data set can handle. Further, the reason to introduce a limit to both the follower and the followings count is that there are in all communities a few accounts with a very unbalanced ratio of followers to friends. For example, in the ”Metallica” community, the user ”cbmethodist” has only 32 followers, but nearly 2000 followings. Keeping users like this in the data set by introducing only a limitation on the follower count would lead the analyses to be not about the community, but about these few outliers.

2.4 Implementation

The source code of the project is available at `https://svn.tugraz.at/svn/Bakk-Twitter`. The for the crawling process necessary methods are:

- **FindAccountsFromFollowers(twitter, account, term, follower_out)**
  This method retrieves all followers of the seed account that have a certain key term in their user description. The parameters are
  - "twitter", which is provided for the connection to Twitter
  - "account", which specifies the seed account
  - "term", which is the key term that is looked for in the user description
  - "follower_out", the output file for the result
- **GetFriendsAndFollowers(twitter, input_faf, output_friends, output_followers)**
  This function gathers all followers and followings of a provided list of Twitter accounts.
  - "twitter", again for the connection to Twitter
  - "input_faf", the list that contains the initial Twitter accounts
  - "output_friends", the file that the followings of the provided accounts should be written to
  - "output_followers", the file that the followers of the provided accounts should be written to
- **GetUserDetails(twitter, input_details, output_details)**
  Here, details of the users, like their post count or their time zone are collected.
  - "twitter", again for the connection to Twitter
  - "input_details", a list of Twitter accounts
  - "output_details", the file that lists the Twitter accounts including their user details

Further, these two methods were used for other tasks during the development:

- **FindAccountsFromFile(twitter, find_accounts_input, find_accounts_output)**
  This method is passed a list of names and searches Twitter to try to find a corresponding Twitter account.
  - "twitter", once again for the connection to Twitter
  - "find_accounts_input", a list of names
  - "find_accounts_output", the output file that has the Twitter accounts

- **DeleteFollowers(dinput, comp, doutput)**
  This is the only method that doesn't interact with Twitter. Instead, it was used to produce the reduced data set.
  - "dinput", the input file with a number of Twitter accounts
  - "comp", file that contains accounts that should be removed
  - "doutput", the remaining Twitter accounts

Since the input and output format is consistent throughout all methods, the user can choose to only execute a single method for a specific task, or execute several in a row. For example it is possible to use "FindAccountsFromFile" to get an account ID and use is as seed account for "FindAccountsFromFollowers" and then execute "GetUserDetails" to gather the user details of the new accounts.

3 Empirical Analysis

The following chapter will describe some general network attributes and the metrics used in the analysis of social networks that are also utilized by Gephi.

3.1 Graphs, Nodes, and Edges

In general a network is displayed in the form of a graph (G) where

$$G = (V, E).$$

V refers to the vertices or nodes while E are the edges. In a social network, in most cases the nodes are associated with users and the edges show how these users are connected. Depending on the type of connection there are two kind of networks: Directed networks and undirected networks. For example, a graph showing friendships between users on Facebook is undirected, since there is no distinction in the relationship of the node pair. On the other hand, a graph showing followers and followings on Twitter is a directed network. Here, the edges are typically depicted with arrows.

The number of edges that are connected to a nodes is the degree of the node. Nodes with a degree of 0, meaning they are not connected to any other nodes at all, are denoted as isolated, as node D is in figure 3.1. The degrees of the nodes A, B and C are 3, 1 and 2 respectively. The average degree of the graph is

$$d(G) := \frac{1}{|V|} \sum_{v \in V} d(V)$$

The average degree of the graph shown in figure 3.1 is 1.5. Further, in directed networks, one can differentiate between out-degree and in-degree.

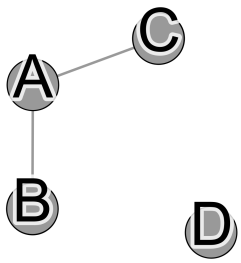

Figure 3.1: An example of an undirected network.

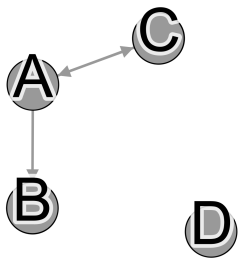

Figure 3.2: An example of a directed network.

The out-degree shows how many edges originate from the node, whereas the in-degree indicates how many edges lead to the node. In figure 3.2, the in-degree of node A is 1, while its out-degree is 2 (Diestel, 2000)

3.2 Network Diameter

In a graph the shortest path from one node to another is known as their distance. If there is no path between the nodes, the distance is set as infinity. The longest distance of all node pairs is the diameter of the network. For example, the network shown in figure 3.3 has a diameter of 2 (Diestel, 2000).

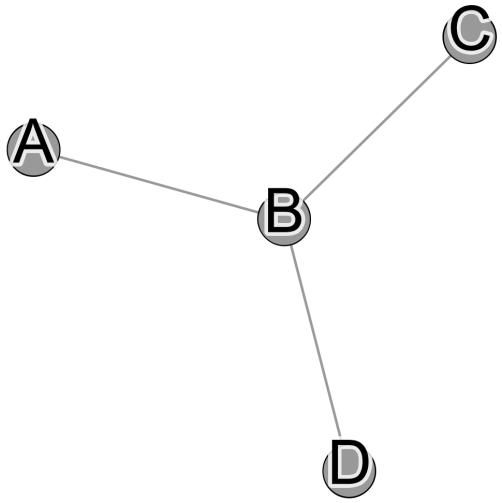

Figure 3.3: A network with a small diameter. Note that any new node that would be connected directly to node B would not change the network diameter.

3.3 Cliques

A clique is a subset of the network that has closer ties to certain parts of the network than it has with others. To be precise, a clique is the maximum sub-graph where all possible connections between the nodes are present. The smallest number a clique can have is two members, a so called dyad. Since the definition above is very strict, ”N-cliques” are commonly used, where ”N” denotes the maximum distance all nodes have to each other (Hanneman and Riddle, 2005).

3.4 Components

In an undirected graph, a components is a subgraph (a subset of nodes from the graph with all their respective edges) in which all nodes are connected within the subgraph, but disconnected from other components. An isolated node can also be a component In a directed graph there are two different kinds of components. A weakly connected component is simply a set of

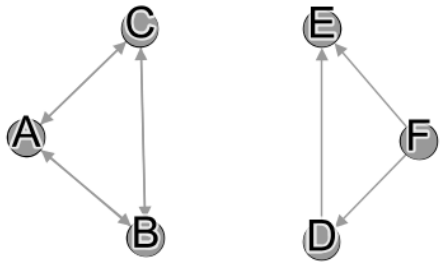

Figure 3.4: The graph on the right is a strongly connected component. The left one is only weakly connected, since there is no directed connection from E to D.

nodes that is connected with each other, the direction of the edges does not matter. On the other hand, in a strongly connected component there has to be a direct path from each node to another to be in the same component (Hanneman and Riddle, 2005).

3.5 Centrality

In a social network, the term ”centrality” refers to the influence a user has in the graph. There are several different measures for centrality.

3.5.1 Degree Centrality

The idea behind degree centrality is a simple one: the more direct connections a node to other nodes in the network has, the greater its potential influence. Therefore, the degree centrality is equal to the degree of a node. Additionally, in directed graphs a differentiation can be made if the node has a high out-degree or a high in-degree. Nodes with a high in-degree are said to be more ”prominent”, while nodes with a high out-degree are more ”influential” (Hanneman and Riddle, 2005).

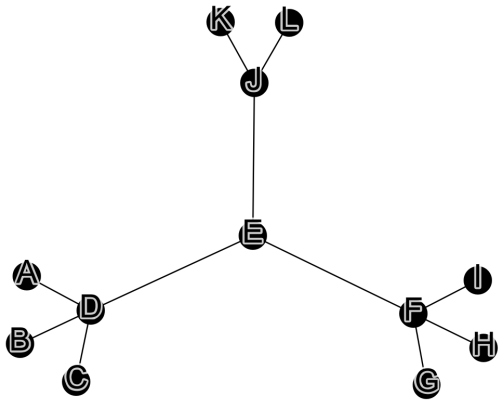

Figure 3.5: In this graph, the node E has the same degree centrality that the nodes D, F and J have, but due to its position in the graph it has both a higher closeness centrality and a higher betweeness centrality than those nodes.

3.5.2 Closeness Centrality

Since degree centrality only takes into account the direct links a node has, there are other ways to determine the centrality like the the closeness centrality. The most important aspect of the closeness centrality is the distance to the other nodes of the graph. The closeness centrality of each node is computed by taking the average of the shortest paths to all other nodes in the network (Hanneman and Riddle, 2005).

3.5.3 Betweeness Centrality

First introduced by Liemann, betweeness centrality stands for how well a node connects other nodes. To put it simple, the betweeness centrality measures how often a node appears on the shortest connection between other nodes in the network. Nodes with a high betweeness centrality present sort of a ”bridge” that connect single users or even entire communities (Hanneman and Riddle, 2005).

3.5.4 Eigenvector Centrality

A more involved approach than the above, eigenvector centrality tries to determine the influence of a node by how many connections it has to nodes that themselves have a high eigenvector centrality. Connections with nodes that have an equal or lower score are less valuable. Google's PageRank is a variant of this approach (Hanneman and Riddle, 2005, Page et al., 1999).

3.6 Eccentricity

The eccentricity of a node is based on the distance of the node it is farthest away from. The maximum eccentricity a node in a graph can have is equal to the graph's diameter (Diestel, 2000).

4 Results

4.1 The Full Data Set

First, a look at the sources of the data set and and its users.

Community	”Star Wars”	”Metallica”	”Apple”
Seed account	”theforcenet”	”fansofmetallica”	”AppleMacGeek”
# of followers	44.4k	64.9k	36k
# of members	3173	791	1465
Avg. # followers	1622.2 (42572.0)	207.2 (597.2267)	1250.3 (8887.4)
Avg. # followings	634.6 (1510.8)	415.8 (623.2)	1112.2 (5309.2)
Avg. post count	3836.9 (11030.0)	3112.8 (25227.1)	5123.2 (21755.6)
Avg. # favorites	1994.0 (10359.8)	536.8 (2199.1)	482.3 (3561.2)

Some observations can already be made here. Even though the seed accounts had a comparable number of followers, the amount of community members that were retrieved from each obviously vastly differs. The community ”Star Wars” is about twice as large as the community ”Apple”, which itself roughly doubles the ”Metallica” community”. Looking at the average number of followers and followings reinforces that ”Star Wars” and ”Apple” are bigger communities than ”Metallica”. However, even though the number of users in the ”Metallica” are less in number, the average post count and number of favorites show that they are not less commited than the users of the other two communities are.

4.2 The Reduced Data Set

As previously mentioned, not the full data set was used for the analysis, but rather a reduced one that contains only the data of users with less than 1000 followers and less than 1000 followings.

Community	"Star Wars"	"Metallica"	"Apple"
# of community members	3173	791	1465
Members in red. data set	748	110	485
% of original data set	23.6%	13.9%	33.1%
Avg. # followers	152.4 (174.8)	87.6 (140.3)	148.5 (177.7)
Avg. # followings	307.9 (253.1)	219.2 (245.8)	290.9 (263.5)
Avg. post count	2043.8 (6624.2)	1233.8 (4439.9)	2026.2 (12631.7)
Avg. # favorites	1003.5 (4971.9)	281.6 (1301.3)	140.5 (917.5)

While the number of users harshly drop, particularly for "Metallica", the attributes of the communities when compared to each other stay relatively the same. "Star Wars" and "Apple" members still have higher follower and followings counts than users of the "Metallica" community. In the full data set, the members of the "Star Wars" and "Apple" communities had on average more followers than followings, while in the reduced data set, the opposite is true. However, looking at the post count and the number of favorites, the users in this reduced data set are nonetheless highly engaged.

4.3 Network Metrics of the Communities

The following analyses the network of the acquired communities and their followers and followings. Also, for each community, nodes that stand out in particular metrics will be discussed in more detail.

4.4 Star Wars Community

Community	”Star Wars”
Number of members in reduced data set	748
Number of nodes	442879
Number edges	1046650
Average degree	2.363
Network diameter	10
Number of weakly connected components	1
Number of strongly connected components	339158

The ”Star Wars” community is the largest data set that was gathered and represents a relatively tight community.

4.4.1 Notable Nodes

Highest Degree Centrality

Id	15403842
User name	”1upGlitch”
Number of followers	997
Number of followings	1862
User description contains ”Star Wars”	Yes

Highest Eigenvector Centrality

Id	700965078
User name	”theforcenet”
Number of followers	45k
Number of followings	379
User description contains ”Star Wars”	Yes

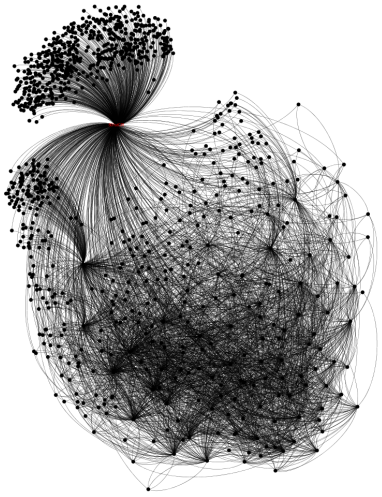

Figure 4.1: This graph is the ego-network of the node "501stCR", which has the highest PageRank in the "Star Wars" Community, with a depth of one. Even though "501stCR" has a fairly low amount of followers and followings, the graph here shows how this node connects two parts of the network.

Highest PageRank

Id	65118234
User name	"501stCR"
Number of followers	1000
Number of followings	30
User description contains "Star Wars"	Yes

The node with the highest PageRank, "501stCR" (4.1), actually a part of the initial "Star Wars" community, but just barely went over the threshold for the limited data set. However, since it is a quite well connected node, it showed up in the followers and followings of the reduced data set. Especially notable is the node with the highest eigenvector centrality, "theforcenet". This, of course, is the initial seed account. All of the three above mentioned users have "Star Wars" in their description and thus would qualify as members of the community.

4.5 Metallica Community

Community	"Metallica"
Number of members in reduced data set	110
Number of nodes	113414
Number edges	192173
Average degree	1.694
Network diameter	16
Number of weakly connected components	1
Number of strongly connected components	85011

The smallest of the three acquired communities, the Metallica community has at the same time the largest diameter and a low average degree. Hence, it is the most sparse community that is described here.

4.5.1 Notable Nodes

Highest Betweeness Centrality

Id	6207392
User name	"Korn"
Number of followers	956k
Number of followings	224k
User description contains "Metallica"	No

Highest Closeness Centrality

Id	232308850333
User name	"alcoholicalml"
Number of followers	210
Number of followings	674
User description contains "Metallica"	Yes

Highest Eigenvector Centrality

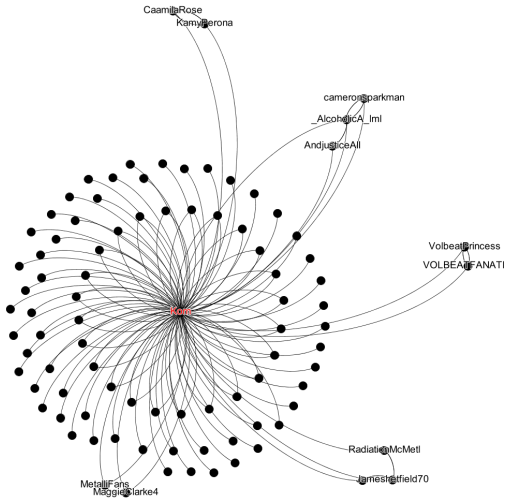

Figure 4.2: This graph is the ego-network of the node ”Korn”, which has the highest betweeness centrality in the ”Metallica” Community, with a depth of one. All labeled nodes at the outer border also have ”Metallica” in their user description.

Id	232308850333
User name	”fansofmetallica”
Number of followers	64.4k
Number of followings	1639
User description contains ”Metallica”	Yes

The node with the highest closeness centrality ”alcoholicalml” would seem unremarkable looking only at the users follower and followings count. However, this account wasn't part of the initial set of users that was gathered from the seed account, and so would qualify as a new member of the ”Metallica” community. Speaking of the seed account, it again shows up here as the node with the highest eigenvector centrality. Finally, the node with the highest betweeness centrality is the official account of the Band ”Korn”. While not part of the core ”Metallica” community, there is definitely an overlap here, as shown in 4.2.

4.6 Apple Community

Community	”Apple”
Number of members in reduced data set	485
Number of nodes	229179
Number edges	380582
Average degree	1.661
Network diameter	12
Number of weakly connected components	1
Number of strongly connected components	184834

The ”Apple” community lies in many areas between the other two, like the amount of nodes and network diameter. However, the average degree is comparable to that of the ”Metallica” community.

4.6.1 Notable Nodes

Highest Betweeness Centrality

Id	15403842
User name	”applestreem”
Number of followers	263k
Number of followings	17.2k
User description contains ”Apple”	Yes

Highest Degree Centrality

Id	700965078
User name	”xmjc99”
Number of followers	893
Number of followings	1172
User description contains ”Apple”	Yes

Highest Eigenvector Degree

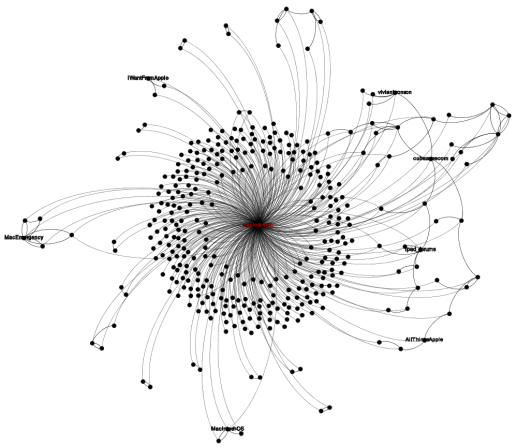

Figure 4.3: This graph is the ego-network of the node "Applestreem", which has the highest betweeness centrality in the "Apple" Community, with a depth of one.

Id	700965078
User name	"AppleMacGeek"
Number of followers	36k
Number of followings	42
User description contains "Apple"	Yes

Once again, the node with the highest eigenvector centrality is the seed account. Regarding the other two nodes, neither of them was part of the initial community that was acquired from the seed account and had less than 1000 followers and followings, but they fulfill the condition of having "Apple" in their user description and so would be part of this community.

4.7 Comparison of the Communities

The three communities vary greatly by size, going from 442879 nodes for ”Star Wars” to 229179 for ”Apple” and 113414 nodes for Metallica. However, looking at the network diameter, the largest community has the smallest diameter (10), while the far smaller ”Metallica” community has a diameter of 16. The ”Apple” community falls again between those two. This suggest that the ”Star Wars” community is more tightly knit than the ”Apple” and especially the ”Metallica” community. The average degree of the ”Star Wars” community is also higher than that of the other two. Looking at the number of strongly connected components, ”Star Wars” has 339158, Apple has 184834 and Metallica merely 85011. Admittedly, the amount of nodes is also a major factor here. On the other hand, the three communities all are equal when considering the weakly connected components: they all consist of a single one. The reason for this is the seed account, which of course all nodes of the limited data set have in common. The seed account also has the highest eigenvector centrality in all three graphs. Taking other metrics into account, the nodes with the highest degree centrality in this reduced data set actually all have a fairly low number of followers and followings and don’t seem to be valuable data points. On the contrary, nodes that came out on top for other centrality measures all have a fairly high amount of connections and influence and may be suitable to be new seed accounts.

5 Conclusion and Outlook

In this thesis, a method to detect communities on Twitter was proposed. Beginning with a selected seed account, the description of the seed's followers were analyzed if they contain a specific key phrase. An implementation of this method was also done in Java, using the Twitter4J library. The gathered communities were for the topics "Star Wars", "Apple" and "Metallica", which varied greatly in size. Nonetheless, the users of these communities where all highly engaged with Twitter, regardless of the community size. Further, the communities that were detected this way were examined and compared utilizing various network metrics with the help of Gephi. However, only the followers and friends, as well as some of the profile data of the users were analyzed. The results of this analysis were further discussed, showing that the proposed method of detecting communities gathers helpful data that can be used to get a more in-depth look on the communities and also at their members. Also, in each community, nodes with outstanding properties, especially considering their centrality were looked at in more detail and how they influence the structure of their respective community. Nonetheless, it should be mentioned that the here proposed method has the disadvantage of having to manually set the community properties, so these terms have to carefully chosen in advance. In the future, the content of the tweets that the members of a community post could be utilized to extract further information from the data set. The used algorithm could also be expanded to recursively analyze followers and followings of the accounts that were gathered from the seed account and find out if they could be added to the community.

Bibliography

Diestel, Reinhard (2000). *Graph Theory*. Second. Springer. ISBN: 0387989765 (cit. on pp. 10, 14).

Gephi (2016). *Gephi Graph Visualization and Manipulation Software*. URL: https://gephi.org/ (cit. on p. 6).

Greene, Derek, Derek O'Callaghan, and Pádraig Cunningham (2012). "Identifying topical twitter communities via user list aggregation." In: *arXiv preprint arXiv:1207.0017* (cit. on p. 2).

Hadgu, Asmelash Teka and Robert Jäschke (2014). "Identifying and analyzing researchers on twitter." In: *Proceedings of the 2014 ACM conference on Web science*. ACM, pp. 23–32 (cit. on p. 5).

Hanneman, Robert A and Mark Riddle (2005). *Introduction to social network methods*. http://faculty.ucr.edu/~hanneman/nettext/index.html (cit. on pp. 11–14).

Highfield, Tim, Stephen Harrington, and Axel Bruns (2013). "Twitter as a technology for audiencing and fandom: The# Eurovision phenomenon." In: *Information, Communication & Society* 16.3, pp. 315–339 (cit. on p. 2).

Java, Akshay et al. (2007). "Why we twitter: understanding microblogging usage and communities." In: *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, pp. 56–65 (cit. on p. 1).

Lim, Kwan Hui and Amitava Datta (2012). "Finding twitter communities with common interests using following links of celebrities." In: *Proceedings of the 3rd international workshop on Modeling social media*. ACM, pp. 25–32 (cit. on p. 2).

Misch, Ben (2016). *gameofnodes*. https://gameofnodes.wordpress.com/ (cit. on p. 2).

Page, Lawrence et al. (1999). "The PageRank citation ranking: bringing order to the web." In: (cit. on p. 14).

Twitter (2016). *REST API*. URL: https://dev.twitter.com/rest/public (cit. on p. 5).

Yamamoto, Yusuke (2016). *Twitter4J*. URL: http://twitter4j.org/en/index.html (cit. on p. 5).