Topic Monitoring in the Pharmaceutical Industry

Master Team Project

presented by
Hailian Hou (123456789)
Chia-Chien Hung (123456789)
Lu Lifei (123456789)
Olga Pogorelaya (123456789)
Ngoc Nam Trung Nguyen (123456789)
Md. Raziul Hasan Al Tariq (123456789)
Alexander Weiß (1420719)

submitted to the
Data and Web Science Group
Prof. Dr. Heiko Paulheim
University of Mannheim

August 2017

Contents

List of Algorithms

List of Figures

List of Tables

Acronyms

API	Application Programming Interface.
HTTP	Hypertext Transfer Protocol.
REST	Representational State Transfer.

Chapter 1

Introduction

Today in 2017 social media is ubiquitous and is taken for granted in our day-to-day life. Nearly every single shop in the city to large companies own social media accounts to communicate with customers or try to engage new ones. The number of active users on the biggest social media platforms is tremendously high. On Facebook we have around 1.871 million active users as of January 2017. Whatsapp and Facebook messenger share the number of 1.000 million active users. For Twitter in comparison to Facebook, the numbers may seem small but with 317 million active users it belongs to the biggest social media platform of our time. [1]
More than ever people share their thoughts about current global events like conflicts between different countries, political events, terrorism, rising diseases as well as the opinions to companies and products. For companies nowadays it is easier than ever to find out what customer think and talk about their company and products. With Twitter we have one the highest reactive social media platforms of our time. Almost instantly people write messages (tweet) to react accordingly to new events in the world. Due to this fact companies try to effectively monitor Twitter and other platforms to get insights into the current customer situation regarding their product and company reputation. In the course of the Master Team Project - Topic monitoring in the pharmaceutical industry of University of Mannheim for Master Data Science and Master Business informatics students, this work especially focuses on the monitoring of social media for the pharmaceutical context. It is processed in cooperation with AbbVie Inc. located in Ludwigshafen, Germany, a pharmaceutical company focused on both biopharmaceuticals and small molecule drugs. The goal of this project is to find efficient and effective ways to retrieve data from different social media platforms to analyze sentiments, public opinions, emotions towards different events, find suitable topics and detect rising trends in social media. To accomplish this task, this work utilizes different machine learning

algorithms to detect the requested features in the messages of users. The declared aim is to find a good combination of different of those algorithms to provide reliable results and findings so that especially AbbVie can use and incorporate them into their attempt to monitor social media.
This report describes how to gather data from different social media platforms and what restrictions will apply to them. It focuses on Facebook and Twitter, because this selection covers the most used and the most reactive social media platform today. Furthermore it deals with different machine learning algorithms, explains how they could be applied to different use cases and suggests situation when and how to use them.

1.1 Problem Statement

1.2 Contribution

1.3 Related Work

Chapter 2

Theoretical Framework

At first we want to provide an explanation to some fundamental terms which will be used in this report. It should give an overview what the different terms mean and in which context they are meant.

2.1 Social Media Data

2.2 Sentiment Analysis

2.3 Topic Detection

2.4 Trend Detection

Chapter 3

Methodology

Here we tell you something

3.1 Data Collection

The first step we take is to collect data from related social media platforms, in our case Facebook and Twitter. We utilize the Application Programming Interface (API) of those to retrieve posts, comments (Facebook) and tweets (Twitter) from users. This is done simultaneously for both Facebook and Twitter. At first R packages were used to crawl the APIs, later on we switched to Javascript implementations to do that. We constructed a list of keywords related to pharmaceuticals industry, companies, diseases and products as ”anchors” for the crawlers, to collect suitable and appropriate data. These keywords were used for both APIs.

Products	Companies	Diseases
adalimumab	abbvie	ankylosing spondylitis
humira	amgen	arthritis
enbrel	johnson&johnson	hepatitis
ibrutinib		psoriasis
		rheumatoid arthritis
		trilipix

Table 3.1: Anchor keywords for collecting data

3.1.1 Facebook API

Overview

Like every major web based platform, Facebook offers an Application Programming Interface (API) to programmatically interact with Facebook. It allows developers to build application which can utilize the social connection and profile information to make them more involving. Furthermore they can have access to public data on Facebook as well as publishing posts and messages to the news feed. Officially the API is called Graph API, because it is follows the concept of a Social Graph [3]. It consists of three key objects:

1. Nodes
    - Describe all things that are shown on the Facebook web page: User Pages, Posts, Comments and Images
2. Edges
    - Describe the relation between these things. For example the comments to an Image posted on a Page
3. Fields
    - Contain additional information to nodes. For example the name of pages or the relationship status of users

The Graph API is based on the simple Hypertext Transfer Protocol (HTTP) and can therefore work with every programming language which implements a HTTP library. Due to this fact it is very easy to implement the API into an application. It supports common HTTP request with GET for retrieving information, POST for uploading data to Facebook and DELETE for deleting data. With this report we exclusively use GET requests to get data from Facebook, as we do not want modify data on Facebook.

Restrictions

Facebook offers a highly efficient and very good documented API for getting data from the social network. So it would be actual a very good data source for social media monitoring, but unfortunately Facebook there are some restrictions which apply. In the following the most serious ones are listed.

1. Data scraping terms of service

- Facebook has a dedicated terms of service document for collecting data from Facebook through automated means. Before any automatic data collection takes place, a written permission has to be obtained by Facebook. Without this permission it is not allowed to scrape any data. Also other restrictions like not selling, or transferring the collected data apply at any given time. So it is absolutely necessary to read through these terms of services before choosing Facebook as a reliable datasource. [2]

2. Streaming API

   - Facebook API offers no streaming endpoint, so no real time feed about new content on the platform can be obtained programmatically. It is up to the developer to create processes to constantly crawl for updated data on Facebook with static endpoints.

3. Rate limits

   - Like every major API, also Facebooks API has some rate limits which will apply if too many requests are sent. During this work the rate limit only appeared once while a lot of pages for one keyword were crawled. So we assume that the rate limitation will not apply many times while using the API. [4]

4. No access to public posts

   - On April 30th, 2015 made a breaking change to their search API endpoint. By then it was possible to search for public posts on Facebook through `/search?type=post&q=foobar`. After this day the endpoint was deprecated and only exclusively available for approved companies. From then on, it was not possible to get any public user post anymore. With this change the API lost the most promising endpoint for companies for monitoring the network. Without the opportunity to crawl public posts from users the only available reliable source for getting posts about different keywords is the page endpoint. With this it is possible to search the pages directory regarding specific keywords and retrieve posts from them. In fact, a company could try to apply for a license to get access to the post search endpoint, but it seems that this is only available to a limited set of media publishers. [5]

Overcome Restrictions and Workflow

The restrictions explained in 3.1.1 lead us to nearly drop Facebook from our workflow. But we tried to find a suitable way to get the most out of the remaining available endpoints.
To overcome the restriction of the Facebook API, we tried to get as much information out of the pages search endpoint. This is the base endpoint for our dataset. It offers the opportunity to specify keywords which the pages need to match returned by the API. We used this to only get pages about AbbVie, their competitors, products from them and diseases which should be healed by those products. Afterwards it is necessary to restrict the retrieved pages to only those which are in a pharmaceutical context. Then the posts and comments from the pages can be crawled and saved for later work.
One major drawback of this approach is that the number of pages and posts on these pages with pharmaceutical context are very low. Only a few conversation about adverse drug reaction, opinions about drugs and companies take place on pages. Most of those discussion arise from user posts, which are not accessible programmatically. Furthermore the conversation inside Facebook pages is very limited and seems to be very biased related to the keyword of the page. So this approach may not lead to a good overview about opinions on different keywords but for us it was the most suitable way to get data from Facebook.

The dataset - some numbers

Conclusion

At first Facebook seemed to be a good and reliable data source for this project. But quickly it became clear that this is not the case. With the restriction of the public post search it is impossible to get all user posts about a keyword. Its only possible to get posts of pages and the corresponding comments to them. But if often occurs that posts on pages are very biased regarding the keyword of the page. Also a lot of pages in the pharmaceutical context seem to not get updated very often and some seem to be very abandoned. So the breaking point here definitely is the depreciation of the API endpoint for searching posts. This leads to the fact, that Facebook can not be used as a suitable data source for topic monitoring social media. Therefore we dropped the Facebook dataset out of the calculation and algorithms.

3.1.2 Twitter API

Overview

Like Facebook, Twitter also offers APIs to programmatically access data. They are divided into two different parts. One part are Representational State Transfer (REST) conform ones which use simpleHTTP requests to create new tweets, read user profile, search for tweets and more.
The second part are dynamic APIs, better known as the Streaming API. They give applications access to the global stream of Twitter data, without any overhead or by pulling data from the static API endpoints. In total there are three different streaming endpoints.

1. Public stream: For public Twitter data
2. User stream: For data associated with only one user
3. Site stream: Multi user version of user streams

For this work the public stream is the most suitable one. It accepts keyword filter to only retrieve data of events which matches those keywords. We can use the keywords from Table 3.1 to only crawl suitable tweets, which match our needs. [6]

Restriction

Unlike Facebook, Twitter does not have that many restrictions we have to respect in our use case.

1. Date restriction in the search API (Static REST API)
    - When you want to use the static search endpoint, where you can search for tweets with specific keywords only the latest two weeks from the date of the request are available to search. So it is generally impossible to search for historical tweets through the API. There are some companies like GNIP [**GGnip2015-10-20**], which are offering service to get access to historical data. To get access to this data you have to contact those companies to get a custom historical dataset, but this option was not suitable for this project as it would had involve payments and our goal is to avoid that.
2. Requests per minute (Static REST API)
    - The REST endpoints, in contrast to the streaming endpoints are rate limited. So developers have to take care of the amount of requests that are sent to the API. A full overview about the different limits

can be found at `https://dev.twitter.com/rest/public/rate-limits` [7]

3. Concurrent streaming APIs (Streaming API)

   - Any application which is using any of the different streaming APIs can only open only one stream to Twitter. This have to be kept in the mind if a developer wants to use multiple streaming endpoint.

Overcome Restrictions and Workflow

This work heavily utilizes the public streaming API, to get the tweets for specific keyword mentioned in table 3.1. Therefore the only restriction which applies is that we do not open multiple streams at a given time. This restriction can be easily overcome by specifying multiple keywords in the API requests. The keywords can be either be separated by a comma which will work as an OR concatenation or separated by an whitespace which will work as an AND concatenation. An example request to the API could look like the following:

```
https://stream.twitter.com/1.1/statuses/filter.json?&track=abbvie
    ,humira,adalimumab...
```

With that request it is possible to save tweets to a database every time a new tweet matches the specified keywords.

The dataset - some numbers

As of August 20th 2017 we stopped crawling data from the Twitter API. At this point our dataset from Twitter contains the following amount of tweets and user:

Type	Amount
Tweets	250732
Retweets	27631
Answer to tweets	597
User	48555

Table 3.2: Common numbers of the Twitter dataset

Keyword	Amount	Proportion in %
johnson & johnson	70240	28.0140
psoriasis	46070	18.3742
hepatitis c	35089	13.9946
rheumatoid arthritis	28940	11.5422
amgen	21404	8.5366
abbvie	14149	5.6431
bristol myers	13355	5.3264
johnson&johnson	5641	2.2498
arthritis	3512	1.4007
ankylosing spondylitis c	3473	1.3851
humira	3029	1.2081
ibrutinib	1851	0.7382
hepatitis	1662	0.6629
NULL	764	0.3047
enbrel	669	0.2668
adalimumab	667	0.2660
imbruvica	172	0.0686
trilipix	45	0.0179

Table 3.3: Twitter tweets per keyword

Conclusion

After we had this massive disappointment with the Facebook dataset, Twitter is by far the better source for building up a dataset for data mining. It offers the opportunity to crawl tweets in real-time with no need to access the static endpoints. The API gives us access to every user tweet matching the keywords, which ensures that at most of the time only appropriate and suitable tweets are crawled. In addition to those advantages the streaming endpoint do not have an rate limitation, which makes it even easier to programmatically crawl tweets. This advantages make sure that Twitter is the primary data source for this project.

Bibliography

[1] Dave Chaffey. *Global social media research summary 2017*. Ed. by Dave Chaffey. Apr. 27, 2017. URL: http://www.smartinsights.com/social-media-marketing/social-media-strategy/new-global-social-media-research/ (visited on 08/18/2017).

[2] Facebook. *Facebook Site Scraping Terms of Service*. Ed. by Facebook. Apr. 10, 2010. URL: https://facebook.com/apps/site_scraping_tos_terms.php?hc_location=ufi (visited on 08/19/2017).

[3] Facebook. *Graph API Overview*. Ed. by Facebook. Feb. 18, 2015. URL: https://developers.facebook.com/docs/graph-api/overview/ (visited on 08/18/2017).

[4] Facebook. *Graph API Rate Limit*. Ed. by Facebook. June 9, 2016. URL: https://developers.facebook.com/docs/graph-api/advanced/rate-limiting?locale=de_DE (visited on 08/19/2017).

[5] Facebook. *Public Feed API*. Ed. by Facebook. Sept. 6, 2013. URL: https://developers.facebook.com/docs/public_feed?locale=en_US (visited on 08/19/2017).

[6] Twitter. *Streaming API Overview*. Ed. by Twitter. May 15, 2012. URL: https://dev.twitter.com/streaming/overview (visited on 08/19/2017).

[7] Twitter. *Twitter API Rate Limits Chart*. Ed. by Twitter. Nov. 22, 2016. URL: https://dev.twitter.com/rest/public/rate-limits (visited on 08/20/2017).