

Social Web Mining

Summer Term 2016



Mining Facebook I

May 25th 2016

Dr. Darko Obradovic

Insiders Technologies GmbH Kaiserslautern
d.obradovic@insiders-technologies.de

Retrospective & Agenda

- Last Week:
 - Frequency Analysis on Twitter
 - Counting Entities and Plotting
- Today:
 - The Facebook API and Page Analysis

Outline for Today

- 5.1 Motivation
- 5.2 The Facebook API
- 5.3 Analyzing Pages
- 5.4 Feeds and Links
- 5.5 Summary & Follow-Up



5.1

Motivation

Some Facts

- Founded in 2004 for Harvard students only
- Slowly Opened for other colleges and high schools
- Publicly accessible since 2006
- Most social networking features supported
- Evolved into a complete platform
- 1.6B users as of August 2015
- Market capitalisation exceeded \$300B

User Activities on Facebook

- 1.6 billion users
 - About half of them daily active
- Updating statuses, posting photos, exchanging messages, chatting in real-time, checking in to physical places, playing games, shopping, etc.

From social graph to interest graph

- Facebook accounts are still real persons
- Facebook pages can be anything
- Likes connect both to an interest graph



The Facebook API

Facebook API

- Graph API
- Facebook Query Language FQL
- Graph API Explorer App
- OAuth 2.0 for access token generation
 - Short-term token
 - Very fine-granular access privileges

Graph API Basics

- View on the Facebook World as a Social Graph
- Nodes:
 - All visible entities like users, photos, comments, ...
- Edges:
 - Connections between entities, e.g. photos on a page
- Fields:
 - Meta data to entities like names

Basic Access

for nodes

GET graph.facebook.com/<node-id>

for edges

GET graph.facebook.com/<node-id>/<edge-name>

same with POST and DELETE

using specific API versions

GET graph.facebook.com/v2.6/<node-id>

Graph API HTTP Queries

```
import requests
```

```
base_url = 'https://graph.facebook.com/me'
```

```
Fields = 'id,name,friends.fields(likes)'
```

```
Url = '%s?fields=%s&access_token=%s' % \  
      (base_url, fields, ACCESS_TOKEN,)
```

```
Content = requests.get(url).json()
```

Limiting Results

```
fields = 'id,name,friends.fields(likes)'
```

```
fields = 'id,name,friends.limit(10).fields(likes)'
```

```
fields = 'id,name,friends.offset(10).limit(10).fields(likes)'
```

```
fields = 'id,name,friends.limit(10).fields(likes.limit(10))'
```

Using a Python Library

```
import facebook # pip install facebook-sdk  
import json
```

```
g = facebook.GraphAPI(ACCESS_TOKEN)
```

Get Objects

```
# get_object(self, id, **args)
```

```
g.get_object('me', metadata=0)
```

```
# g.get_objects(['me', '12345'], metadata=0)
```

```
{  
  "first_name": "Axel",  
  "last_name": "Ackermann",  
  "verified": true,  
  "name": "Axel Ackermann",  
  "locale": "de_DE",  
  "gender": "male",  
  "updated_time": "2016-03-09T14:29:47+0000",  
  "link": "https://www.facebook.com/id/206365329724553/",  
  "timezone": 2,  
  "id": "206365329724553"  
}
```


Get Connections

```
# get_connections(self, id, connection_name, **args)  
  
g.get_connections('me', 'friends', metadata=0)
```

```
{
  "paging": {...},
  "data": [
    {
      "name": "Sandra Mayer",
      "id": "01234"
    },
    {
      "name": "Martin Schmitt",
      "id": "56789"
    },
    ...
  ]
}
```

Requests

```
# request(self, path, args=None, post_args=None)
g.request("search", {'q' : 'social web', 'type' : 'page'})
```

```
{
  "paging": {...},
  "data": [
    {
      "category": "Book",
      "name": "Mining the Social Web",
      "id": "146803958708175"
    },
    ...
  ]
}
```

```
{
  "website": "http://miningthesocialweb.com",
  "can_post": true,
  "founded": "January 2011",
  "likes": 2432,
  "id": "146803958708175",
  "category": "Book",
  "has_added_app": false,
  "talking_about_count": 8,
  "is_community_page": false,
  "username": "MiningTheSocialWeb",
  "description": "Facebook, Twitter, LinkedIn, Google+, ...",
  "awards": "Jolt Productivity Award (1st Ed.) - http://drdobbs.com/joltawards/...",
  "genre": "Data Mining",
  "checkins": 0,
  "name": "Mining the Social Web",
  "release_date": "January 2011",
  "cover": {
    "source": "https://scontent.xx.fbcdn.net/t31.0-0/p180x540/1073953_522831997.jpg",
    "cover_id": "522831997772034",
    "id": "522831997772034"
  },
  "were_here_count": 0,
  "link": "https://www.facebook.com/MiningTheSocialWeb/",
  "is_published": true
}
```

Result Paging / Cursoring

```
{
  "paging": {
    "cursors": {
      "after": "NTE0",
      "before": "MAZDZD"
    },
    "next": "https://graph.facebook.com/v2.0/search?access_token=ABC
            &q=social%20web&type=page&after=NTE0",
    "previous": "https://graph.facebook.com/v2.0/..."
  },
  "data": [
    ...
  ]
}
```

Search Types

- User
 - Search Facebook users by name if allowed
- Page
 - Search any page by name
- Event
- Group
- Place

Beyond

Facebook's Walled Gardens

The Open Graph Protocol

- Include external web pages into Facebook's Social Graph
- Add a few lines of RDFa on the web page
- Facebook will be able to include it and link it
- Examples:
 - IMDB movie pages
 - O'Reilly book catalogue pages

IMDb Movies as Open Graph Objects

<http://graph.facebook.com/http://www.imdb.com/title/tt0117500>

```
{  
  'id': '114324145263104',  
  'name': 'The Rock (1996)',  
  'picture': '...',  
  'link': 'http://www.imdb.com/title/tt0117500',  
  'category': 'Movie',  
  'description': 'Directed by Michael Bay. With Sean Connery, ...',  
  'likes': 16345  
}
```

```
<html
  xmlns:og="http://ogp.me/ns#"
  xmlns:fb="http://www.facebook.com/2008/fbml">
  <head>
    <meta property="og:url"
      content="http://www.imdb.com/title/tt0117500/" />
    <meta property="pageId" content="tt0117500" />
    <meta property='og:image'
      content="http://ia.media-imdb.com/images/...jpg" />
    <meta property='og:type' content="video.movie" />
    <meta property='fb:app_id' content='115109575169727' />
    <meta property='og:title' content="The Rock (1996)" />
    <meta property='og:site_name' content='IMDb' />
    <meta property="og:description" content="Directed by Michael Bay.
      With Sean Connery, Nicolas Cage, Ed Harris, John Spencer.
      A mild-mannered chemist and an ex-con must lead the
      Counterstrike when a rogue group of military men, led by a
      renegade general, threaten a nerve gas attack from Alcatraz
      against San Francisco." />
```

O'Reilly Books as Open Graph Objects

<http://graph.facebook.com/http://shop.oreilly.com/product/9780596529321.do>

```
{
  "category": "Book",
  "about": "This fascinating book demonstrates how you can build web applications to mine...",
  "talking_about_count": 0,
  "description": "This fascinating book demonstrates how you can build web applications to mine...",
  "has_added_app": false,
  "can_post": true,
  "name": "Programming Collective Intelligence",
  "link": "http://shop.oreilly.com/product/9780596529321.do",
  "likes": 17,
  "is_community_page": false,
  "were_here_count": 0,
  "checkins": 0,
  "id": "221689464577245",
  "is_published": true
}
```

Entity Recognition and Disambiguation

- Open Graph Objects have an ID of their own
- Duplicates to Facebook pages will exist
- Entity Recognition to classify OGP objects
- Entity Disambiguation to unify OGP objects and pages



Analysing Pages

Interesting Questions

- How popular is the page?
- How engaged are the page's fans?
- Are any fans particularly participative?
- What are the most common topics talked about there?

Pepsi vs Coca-Cola

```
g.request('search', {'q':'pepsi', 'type':'page', 'limit':10})
g.request('search', {'q':'coca cola', 'type':'page', 'limit':10})

pepsi_id = '56381779049' # could also use 'PepsiUS'
coke_id  = '40796308305' # could also use 'CocaCola'

# format large numbers
def int_format(n): return "{:,}".format(n)

print "Pepsi likes:", int_format(g.get_object(pepsi_id)['likes'])
print "Coke likes:", int_format(g.get_object(coke_id)['likes'])
```


Pepsi vs Coca-Cola

Pepsi likes: 35,294,951

Coke likes: 97,670,428

Pepsi vs Coca-Cola

- Market capitalisation:
 - NYSE:PEP \$145 billion
 - NYSE:KO \$190 billion
- Brand value estimation:
 - Pepsi: \$18 billion
 - Coca-Cola: \$80 billion

Twitter reveals insights from best social campaigns of 2015

Twitter UK's head of brand strategy reviews some of the best campaigns on the social media platform last year and shares insights to help brands in 2016.



Coca-Cola's #ChooseHappiness: one of the best brand campaigns on Twitter last year



Feeds and Links

Feeds and Links Terminology

- Feeds
 - Collection of any content on a page or user wall
- Posts
 - Any content posted on a page or wall
- Status Update
 - A status posted on the user's own wall

Getting a Page's Feed and Shared Links

```
g.get_connections(pepsi_id, 'feed')  
g.get_connections(pepsi_id, 'links')
```

```
g.get_connections(coke_id, 'feed')  
g.get_connections(coke_id, 'links')
```

```
{
  "picture": "https://scontent.xx.fbcdn.net/v/...",
  "story": "Gerri Matveenko DeMarshall shared a video to Pepsi's Timeline.",
  "from": {
    "name": "Gerri Matveenko DeMarshall",
    "id": "10203993640987476"
  },
  "name": "NowThis Future",
  "is_hidden": false,
  "to": { "data": [
    {
      "category": "Food/Beverages",
      "name": "Pepsi",
      "id": "56381779049"
    }
  ]
},
  "link": "https://www.facebook.com/NowThisFuture/videos/1131815580192954/",
  "created_time": "2016-05-24T12:33:06+0000",
  "message": "If a worldwide company like Pepsi started using these biodegradable
              six-pack rings, think of the good it would do for the environment,
              not to mention the wildlife that would be saved!",
  "type": "video",
  "id": "56381779049_10209650374482278",
  "status_type": "shared_story",
  "description": "These edible six pack rings promise to save marine life and reduce pollution",
  ...
}
```

```
{
  "message": "They just don't make Pepsi commercials or James Bond 007 movies like they used to",
  "likes": {
    "paging": { ... },
    "data": [
      {
        "id": "10204290862174412",
        "name": "Steve Spring"
      },
      ...
    ]
  },
  "comments": {
    "paging": { ... },
    "data": [
      {
        "from": {
          "name": "Douglas Philpott",
          "id": "724802164312140"
        },
        "like_count": 0,
        "can_remove": false,
        "created_time": "2016-05-24T02:43:53+0000",
        "message": "Just what was Pepsi Max anyway?",
        "id": "10154042880435168_10154042932730168",
        "user_likes": false
      },
      ...
    ]
  },
  ...
}
```


Starting Point for

Frequency Analysis

+

Text Analysis



Summary & Follow-Up

Summary

- Understand the Facebook Graph API
 - how to access the API
 - how to analyse Facebook pages
 - how to access page feeds and shared links

Follow-Up

- Create a Facebook account
- Exercise Sheet 5:
 - get familiar with the Facebook Graph API