

ネットワーク解析のための FacebookAPI の試用

谷 研究室 鈴木 悠也

概要

Facebook におけるネットワークの相関関係を解析する準備として, FacebookAPI を用いたアプリケーションを試作した.

1 はじめに

急激なインターネットの普及により自宅に居ながら世界中の人と繋がりをもてるようになった. そして SNS(ソーシャル・ネットワーキング・サービス)といわれるコミュニティ型のサイト普及により, 遠くの一度も会ったことのないような人とも密接な繋がりを持てるようになった. 主な SNS としては mixi や Myspace などが挙げられ, その中でも近年もっとも急成長し世界最大のユーザー数を持つのが Facebook([1, 2, 3]) である.

一方, 様々なネットワークを対象に, 様々な方法でネットワークの解析が行われている ([4]). 代表的なものとして, ダンカン・ワッツとスティーブン・ストロガッツが提案したスモールワールドや, アルバート＝ラズロ・バラバシとレカ・アルバートが提案したスケールフリー性などがある. スモールワールド現象とは知り合い関係を辿っていくと比較的に世界中の誰にでも行き着くという仮説である. 社会心理学者スタンレー・ミルグラムは, 1960 年代にこの仮説を検証する実験を行った. その結果, 人と人は平均して 6 人の知り合いを介して間接的な知り合いになるということが検証された.

このような現象が Facebook のネットワークにおいても成り立つのか検証したいと考えた. その準備として, Facebook API(Application Program Interface) を利用したアプリケーションを試作した.

2 章では Facebook について, 3 章では Facebook API について, 4 章ではアプリケーション開発について, 5 章では実験と考察を述べてまとめとする.

2 Facebook について

2004 年にハーバード大学の学生だったマック・ザッカーバーグが創業し, 当初はハーバード大学の学生のみだったが要望があり他大学へも開放した. 2006 年初頭に全米の高校生に開放し, 2006 年 9 月に一般に開放された. そして 2008 年に日本語版が一般公開され, 2009 年に MySpace を抜き世界最大の SNS となり, 2010 年

7 月現在では全世界に 5 億人以上のユーザーを持っている. 大多数の人が実名登録をしており, かなり事細かプロフィールの設定ができることから他の SNS より密接なつながりを持てると言える.

以下でいくつか Facebook の特徴的だと思う部分を紹介する.

2.1 プロフィール

基本的にはどこのサイトでもあるものばかりだが, 元々学生向けだったことや世界中に公開していることから Facebook ならではの箇所もある.

- 宗教・信仰の欄...世界中にユーザーがいる Facebook だからこそあるといえる. 日本だとあまり気にされないためかあまり見かけない.
- プロフィール写真の欄...実名登録のところからわかるように大多数の人が自分の顔写真を載せている.
- 学歴・経歴の欄...よくある出身高校や出身大学を記入する欄だが, もとが学生向けだったこともあり次で紹介するコネクションサーチや母校の友達検索などに役立つ.

2.2 コネクションサーチ

コネクションサーチとは自分の在籍している学校と自分の志望している業界にいる同じ学校の就活生や内定者, OB・OG を探すことができ, 図 1 ように表示される. こういったサービスはもとが大学生向けであったことからきている. これを利用することにより学部・学科の違う人と知り合ったり, 就職活動がやりやすくなり, 内定者も事前に交流を持つことができる.



図 1 コネクションサーチ

2.3 友達検索

友達検索画面 (図 2) でアドレスやパスワードなどを入力することにより入力したサーバにアクセスし、そこから登録している友達に招待のメッセージを送ることができる。

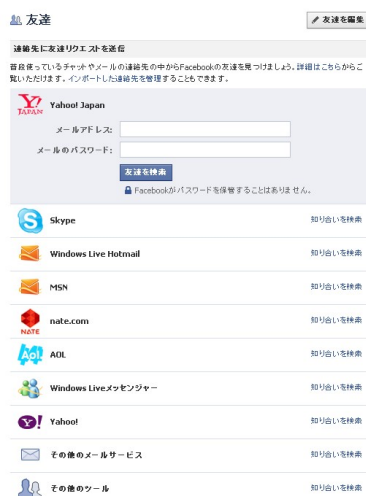


図 2 友達検索

3 Facebook API について

Facebook の API は 8 つに分かれており、

- 管理方法
- ログイン/認証方法
- データ探索方法
- 公開方法
- ダッシュボード API 方法

- イベント API 方法
- カスタムタブ API 方法
- 広告方法

がある。特に使うであろうデータ探索方法の API からいくつかを紹介する ([5])。

- `users_getInfo(A,'uid,name,etc')...A` というユーザーの ID, 名前などのデータを出力する
- `friends_get()...` 自分の友達の ID を取得する
- `friends_areFriends(B,C)...` B と C が友達関係にあるかを判定する
- `notes_get...` ノートに書いてある情報を取得する

4 アプリケーション開発

4.1 アプリケーションを実行

Facebook では Facebook サーバがアプリケーションを表示するまでに製作者サーバに通信する必要がある。それは製作者サーバにアプリケーションが入っているからで、ここでは表示までの流れを説明する。

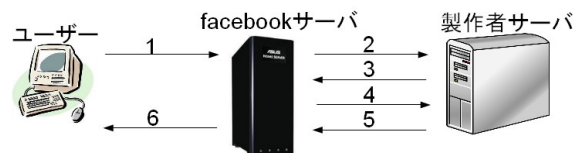


図 4 アプリケーション表示までの流れ

1. ユーザーがある製作者のアプリケーションをリクエスト
2. Facebook サーバから製作者サーバにリクエストを送る
3. Facebook サーバに必要なデータをリクエスト
4. リクエストされたデータを返す
5. 製作者サーバから HTML・JavaScript で返す
6. 返された HTML・JavaScript をユーザに返す

こうした流れにより Facebook 上でアプリケーションが表示される ([6])。

4.2 サンプルコードの実行

以下は Facebook におけるアプリケーション開発までの簡単な流れでサンプルコードを使い説明する．

1. まず Facebook developeres にアクセスし新しいアプリケーションをセットアップする
2. アプリケーション名などを入力し必要な情報 (Application ID , API key , Secret key) を取得する
3. キャンバスページとキャンバスページ URL を入力
4. サンプルコードに従い , Facebook の PHP ライブラリをダウンロード
5. Application ID , Secret key , require_once を変更
6. キャンバスページにアクセスし実行すると , 1 部ではあるが図 3 のような自分のユーザー情報が表示される

Your User Object

```
Array
(
    [id] => 100001869363827
    [name] => Yuva Suzuki
    [first_name] => Yuva
    [last_name] => Suzuki
    [link] => http://www.facebook.com/profile.php?id=100001869363827
    [hometown] => Array
        (
            [id] => 107454992625669
            [name] => Japan
        )
    [location] => Array
        (
            [id] => 109807622370958
            [name] => Kazo-shi, Saitama, Japan
        )
)
```

図 3 アプリケーションの表示の一部

ここで重要なのはキャンバスページとキャンバスページ URL である．これは 4.1 で説明したように Facebook サーバにアクセスし , アプリケーションを動かす画面がキャンバスページで製作者サーバにアクセスし , アプリケーションの実行プログラムがある場所をキャンバスページ URL という．

参考文献

- [1] Facebook(日本語)
<http://ja-jp.facebook.com/>
- [2] 横田 真俊 著 今すぐ使えるかんたん mini facebook 基本&便利技 (2010)
- [3] Wayne Graham 著 Facebook API Developers Guide (2008)
- [4] 今野 紀雄 , 井手 勇介 著 複雑ネットワーク入門 (2008)
- [5] Facebook DEVELOPERS
<http://developers.facebook.com/>
- [6] Facebook Platform × Flash Platform で開発するアプリケーションのアーキテクチャーの理解
http://www.adobe.com/jp/devnet/facebook/articles/facebook_architecture_overview.html

5 実験

本来は *friends_get()* を用いて友達の友達を表示していき , 6 次の隔たり検証やソーシャルグラフの可視化をする予定だった．しかし , *friends_get()* は自分の友達しか表示できなかったため友達の友達が取得できなかった．これは実名登録が多数なためプライバシー設定が複雑なためだと思われる．

なので 1 人のユーザーを選び *friends_areFriends()* を使い全ユーザーと友達関係かを調べ , 友達関係だったユーザーをまた *friends_areFriends()* を使い全ユーザーとの友達関係を調べていくことにより理論上は 6 次の隔たり検証やソーシャルグラフの可視化はできる．しかし , それは膨大な時間と API の実行制限により不可能なため 500 万分の 1 というかなり低い確率だが , 全ユーザーからランダムに 1 人を選び出し , ランダムで選んだ 100 人と友達関係であるかを調べるアプリケーションを作った．

6 考察

100 回ほど繰り返した結果 , 一度も友達と表示されなかった．これは確率的にもかなり低かったことも原因だが , 最初に選び出した 1 人の友達の数が何人かを数名見てみたら 500 人を超える人もいれば 10 人いない人もいた．

7 終わりに

Facebook API を利用し 1 次の隔たりは可能なので , そこからネットワークへのアプローチの模索や人との繋がりを考察するとともに , 今回の経験をもとに他の SNS(mixi, Myspece, グリー, etc) の API を使い 6 次の隔たり検証やソーシャルグラフの可視化を行いネットワーク上の情報の伝達や拡散を考察する．