公立はこだて未来大学 2015 年度 システム情報科学実習
グループ報告書

Future University Hakodate 2015 System Information Science Practice

Group Report

プロジェクト名

地方のための Twitter ローカライズ

Project Name

Twitter Localization

グループ名

分析班

Group Name

Analysis Group

プロジェクト番号/Project No.

19-分析班

プロジェクトリーダ/Project Leader

1013039　丸山大仁　Hirohito Maruyama

グループリーダ/Group Leader

1013169　赤坂尚衡　Naohira Akasaka

グループメンバ/Group Member

1013003　石橋笙　Sho Ishibashi
1013007　小川聖司　Seiji Ogawa
1013010　川向達也　Tatsuya Kawamukai
1013095　菅原春香　Haruka Sugawara
1013014　傳法谷強　Tsuyosi Denpouya
1012207　村尾雅都　Masato Murao

指導教員

寺沢憲吾　竹之内高志　永野清仁　片桐恭弘

Advisor

Kengo Terasawa　Takashi Takenouchi　Kiyohito Nagano　Yasuhiro Katagiri

提出日

2016 年 1 月 20 日

Date of Submission

Janualy 20, 2016

概要

近年ソーシャルネットワーキングサービス (SNS) の利用者数は増加しており、そのなかでも Twitter は 2014 年 12 月の時点で、LINE や Facebook についで 3 番目に利用者数が多い SNS である。このため、Twitter と連携しているアプリケーションも数多くあり、その中にはアプリメーカー [1] やツイートプロファイリング [2] などのツイートから性格を診断するアプリケーションも存在する。しかし既存の性格診断アプリケーションは、解析して得られたデータをさらに解析できないこと、解析するツイート数が 500 件と少ないことなど、改良の余地があると考えられる。そこで、我々分析班は既存のアプリにはない新しい性格診断の作成を目標にした。

我々分析班はツイートから性格を推測し、加えて機械学習を用いて函館の観光地をお勧めする Web アプリケーションを提案した。新しい性格診断の機能として、ユーザーの未来の性格を推測する「現在から未来診断」と、ランダムに選ばれた相互フォロワー全体の性格を診断する「集団診断」を作成した。これに加えて、ユーザーのツイートと性格診断の結果から、機械学習を用いてユーザーに函館の観光地をお勧めする機能を実装した。これらの機能を実現するために、前期は性格診断の部分に焦点を当て、Web 班、API 班、エゴグラム班に分かれて開発を行った。後期は性格診断の作成に加え、機械学習を用いて函館の観光地をお勧めする機能を実装した。

キーワード　Twitter, 性格診断, 機械学習

（※文責: 川向達也）

Abstract

The number of social networking service (SNS) users has increased in the past several years. In Japan, Twitter is the third-largest SNS next to LINE and Facebook in December, 2014. There are many Twitter applications, including personality test applications like appli-maker[1] and tweet-profiling[2]. However, the existing personality test applications have little flexibility. They cannot analyze tweets to the full extent, and they can deal with only 500 tweets for each account. The purpose of the analysis group is to provide a new personality analysis application that settles the above problems.

We proposed a personality analysis web application, which classifies the personalities of Twitter users, and determines the recommended tourist spots in Hakodate by using machine learning. The analysis results of the proposed application are "future analysis", which is the user's future personality, and "group analysis", which is the personality of a group of randomly chosen mutual followers. In addition, we developed a tourist spot recommendation system in Hakodate. The recommendation system utilizes the result of tweet analysis and machine learning methods. In the first semester, we split into Web team, API team, and Egogram team to develop the personality analysis application. In the second semester, we continued to develop the application, and added a recommendation system of tourist spots in Hakodate.

Keyword Twitter, Personality analysis, Machine learning

（※文責: 川向達也）

目次

第 1 章　プロジェクトの背景

近年 Web 上でのコミュニケーション手段として、すでに普及が進んでいる電子掲示板システム (BBS) やブログ (Weblog) に次いで、利用者が急増しているものがソーシャルネットワーキングサービス (SNS) である [3]。 SNS は、人と人とのつながりを促進・サポートする、コミュニティ型の Web サイトである [4]。さらに、SNS の利用者数は多く、そのなかでも 2006 年に米国でリリースされた Twitter は、2008 年 4 月 23 日には日本語サイト「Twitter Japan」のサービスも開始され [5]、2014 年 12 月で LINE や Facebook についで 3 番目に利用者数が多い [6]。また Twitter は世界中で 2 億 8400 万人のユーザーが利用し、国内でも 2000 万人近くが利用しているサービスである [7]。 ユーザーは 140 文字までの短文から構成されるツイートと呼ばれる投稿ができる。現在の Twitter はモバイルベースのユーザー率が全体の 80%[7] を占め、利用者は各地を移動しながらいつでも Twitter を利用することができる。そのため Twitter では位置情報を利用してツイートを投稿したり、情報を検索したりすることが可能である。

Twitter における Facebook など他の SNS と異なる特徴として、以下の 2 点を挙げることができる。1 点目はリアルタイム性が高いことである。投稿件数は 1 秒あたり約 5700 件あり、トレンド分析や口コミ分析に利用されることがある [8]。2 点目は 10 代から 20 代の若い世代の利用者が非常に多いことである。

本プロジェクト内では、Twitter を用いる場合に感じる不満な部分を話し合った際に、「アプリメーカー」や「ツイートプロファイリング」のような連携サービスを用いた分析では不明瞭な点が多いということや、Twitter で提供される検索サービスを用いて情報を入手しようとした際に他の情報検索システムを利用しなければならず手間がかかるということが挙げられた。話し合いで挙げられた不満点をまとめると以下の 2 つに分類された。

- 情報検索の手間が多く不満
  - 1 つのアプリケーションに簡潔化できないか
- 既存の性格診断アプリに対する不満
  - 既存のものにない, 独自の性格分析を実現できないか
  - より多くのツイートを用いて特徴を得ることができないか

本プロジェクトではこのような部分の改善に関する提案・調査・意見交換を行い、「Twitter の特徴を最大限に活かした新しいサービスを提供」を目的として活動することになった。また本プロジェクトは上記の 2 点の問題に対する改善を行うために、以下の 2 つの班に分かれプロジェクト活動を進めてきた。

- 分析班
  - ツイートから性格を推測し, 函館の観光地をお勧めする Web アプリケーション
- 検索班
  - カテゴリから任意の場所の飲食店に関するツイートを検索し地図に表示する Web アプリケーション

結成後、両班ともに前期は意見交換・調査を重点的に行い、Web サイト設計に必要となる知識

習得と、その得た知識を使った Web アプリケーションの実装を行った。後期は実装されたアプリケーションに対してレビューを行い、そこで挙げられた改善点に対する実装を繰り返し行った。

（※文責: 小川聖司）

第 2 章　本グループの課題の背景

分析班では、Twitter におけるツイートの内容に着目した。ツイートは 140 字までの短文で構成されており、画像や位置情報も付け加えることができる。Twitter は携帯端末から気軽に利用されることが多いため、その時のユーザーの気分や思いつきといったものが含まれることから即時性も高い。またユーザー同士の Twitter 内での会話 (リプライ) もあり、フォロワー間でのやりとりが可能である。このようなツイートはユーザーごとで内容が様々であり、その特徴も異なっている。こういった特徴をツイートから抽出しようとする場合は、ツイートプロファイリング [2] やアプリメーカー [1] のような、既存の Twitter 連携アプリを使用することで傾向を見い出すことが可能である。しかしこのようなアプリケーションによって得られる出力結果は正確に解析されていないため、そのユーザーが持つ傾向や特徴を調べることは困難である。

このような現状に対して分析班で議論した結果、いくつかの意見や不満点が挙げられた。まず、既存の Twitter 連携アプリで行う解析精度が正確でない点である。例を挙げるとツイートプロファイリングではユーザーの最近のツイートを分析及び集計して、ツイートの内容をグラフ化して表示することができる。ツイートの分析にはユーザーの過去 500 件のツイートを使用し、出力結果は画像付きツイートとして投稿することが可能である。しかしこうして得られた出力結果は精度が高いわけではない。原因として考えられるのは、取得しているユーザーのツイート数が 500 件であることから、より高い精度で解析を行うにはツイート取得数が不十分であるためと考えられる。また他の事例として、アプリメーカーが挙げられる。アプリメーカーではユーザーが自由に独自のアプリを作成することが可能であり、Twitter の情報を解析するアプリも作成できる仕様になっている。しかしアプリメーカーによって作成された Twitter 解析アプリは、表示結果がランダムである場合や毎回異なることがあり、利用するユーザーに正確な解析結果を提供することは不可能である。

（※文責: 小川聖司）

第 3 章　本グループの提案

3.1　概要

分析班は「ツイートから性格を推測し函館の観光地をお勧めする Web アプリケーション」の開発を提案した。既存の性格診断アプリケーションにはない新しい診断や、函館に関連した機能の作成を目指した。前期は、エゴグラムを用いた性格診断に焦点を当て、Web 班、API 班、エゴグラム班の 3 つに分かれて活動した。後期は、性格診断の作成に加えて、機械学習を用いて函館の観光地をお勧めする機能の開発に取り組んだ。

（※文責: 川向達也）

3.2　機能

「ツイートから性格を推測し函館の観光地をお勧めする Web アプリケーション」の機能は、TwitterAPI を使用してユーザーのツイートを取得しそのツイートを分析して性格を診断する。そして、機械学習を用いて診断結果とツイートからユーザーに合いそうな函館の観光地をお勧めする。性格診断は以下の 3 種類の機能を提案した。

- 現在の性格を診断する「通常診断」
- ユーザーのツイートを分析してユーザーの未来の性格を推測する「現在から未来診断」
- ユーザーと相互フォローしてる人のツイートを分析してユーザーの所属している集団の性格を診断する「集団診断」

函館の観光地をお勧めする機能は、性格診断の結果とユーザーのツイートを分析し、機械学習を用いて作成することを提案した。Web ページ上では、性格診断の結果とユーザーにお勧めの観光地を同時に表示することにした。これは、性格診断の結果と同時に表示することで、性格診断だけに興味を持っていた人が観光地にも興味を持つかもしれないと考えたからである。集団診断の場合は、集団に対してお勧めの函館の観光地を表示することを提案した。

（※文責: 川向達也）

第 4 章　課題解決のプロセス

4.1　グループ結成について

プロジェクト開始当初に、プロジェクトメンバー全員で現在自分たちが利用している Twitter 関連のサービスについて話し合った。その結果、以下のサービスを利用していることがわかった。

- ツイートのログを自動で記録できる「Twilog」
- Twitter に関するデータを集計してくれる「Twitter Analytics」
- ツイートを整理し、分類分けできる「TweetDeck」
- タイムラインを自由に構築できる「Krile STARRYEYES」
- リツイート直後のツイートを表示できる「RtRT」
- タイムラインが見やすくなる「Plume」

次に、Twitter の不満点について意見を出し合った。その結果、以下の不満点が出た。

- ツイートをまとめて消せない
- 写真などからの個人情報の流出が防げない
- トレンドに関心が持てない
- フォロー、フォロワーのリスト順がバラバラ
- 間違ってお気に入りしたときの対処が面倒
- 確認したツイートをタイムラインから消したい
- 広告が出る
- 過去のツイートが一覧で見れない
- 140 文字では情報を正確に伝えられない

しかし、Twitter の不満点をあげているだけでは、自分たちが作りたいサービスのイメージが湧かず行き詰ってきたので、不満点から新しい Twitter サービスを考える視点から自分たちのやりたいことから新しい Twitter サービスを考える視点へと変更し、やりたいことをホワイトボードに書き出し、意見をまとめた結果、以下の 2 つになった。

- 機械学習をしたい
- 位置情報を利用したい

その後、機械学習を中心に活動する分析班と位置情報を中心に活動する検索班の 2 つの班に分かれて活動した。

（※文責: 村尾雅都）

4.2　サービス提案までの流れについて

分析班結成当初に、自分たちがしたい事に関してメンバーでブレインストーミングを行い、以下の 4 つの意見が挙がった。

- 機械学習で何かしたい
- ツイートのテキストデータ、画像、動画を解析したいが、そのなかでもテキストデータの解析に興味がある
- 方言から地域を特定したい
- 性格分析を行いたい

そこで、これらをまとめ、機械学習によってさまざまな特徴を抽出して、「プロフィール帳」を自動で作成してくれるシステムの開発を行うことに決めた。プロフィール帳とは、個人の特徴を書いたものである。前期は、プロフィール帳の項目の中の性格を診断する Web アプリケーションの作成に焦点を当てた。それに伴い、どのような性格があるのか、ツイートから性格を判別するにはどうすればよいのかの意見を出し合った。その結果、エゴグラムを使って性格診断を行う方法に決定した。また、Twitter での性格診断アプリケーションの先行事例調査を行うと、「アプリメーカー」や「ツイートプロファイリング」などといった、自分たちが提案する案と似た性格診断アプリケーションが見つかった。そこで、既存の性格診断アプリケーションと差別化するために、3 つの新しい診断方法を考えた。

- 診断するユーザーの 3200 件のツイートから現在の性格を診断する現在診断
- ユーザーのツイートを分析して未来の性格を予測する未来診断
- ユーザーとユーザーの相互フォロワーのツイートを用いて、ユーザーが所属する集団がどのような性格なのかを診断する集団診断

前期は、3 つ診断方法の中の現在診断の作成を行った。

後期当初に、地域のためのローカライズをどのようにして組み込むか意見を出し合い、以下の 3 つの意見が出た。

- 函館のためになるアプリケーションを作りたい
- 数ある函館の観光地から自分の性格に合った観光地を見つけたい
- 観光客の満足度を高めたい

そこで、これらをまとめ、ユーザーに函館の観光地をお勧めする機能の作成をすることに決めた。また、現在診断と未来診断はまとめた方が良いのではないかと意見が出た。そこで、現在診断と未来診断を合わせ、現在から未来診断とし現在から未来の性格の推移を表示する機能の作成をすることに決めた。さらに、集団診断では、自分のグラフの形と似た人を気の合う人として表示する機能の作成を提案した。

（※文責: 村尾雅都）

4.3 具体的な構想の作成について

私たちは具体的な構成を考えるに当たり、始めに各々の考えた結果画面をホワイトボードに書いた。そこから、分析班全員の良い点をまとめた結果、画面に以下の 4 つを組み込むことに決めた。

- 性格診断を行ったアカウント名
- 結果画面に折れ線グラフで性格診断の結果を表示
- その性格についての説明

- 共有ツイートできるボタン

次に Web ページの全体の構成についてを全員で考えた。その結果、作成する Web ページは以下の 4 つになった。

- ホーム画面
- Twitter アカウントの認証・ログイン画面
- 性格タイプ診断選択画面
- 診断結果画面

性格の診断手順としては、まずユーザーは分析班が作成した Web ページのホーム画面にアクセスを行い、診断ボタンを押すと、自分の Twitter アカウントを認証・ログインを行う画面にアクセスされる。ユーザーが認証・ログインを完了すると、診断タイプ選択画面にアクセスされ、現在、未来、集団のどれか 1 つの診断タイプを選択する。診断タイプを選択すると、TwitterAPI を用いてユーザーの最新のツイートから 3200 件のツイートを取得し、それを形態素解析する。形態素解析されたツイートをエゴグラムの特徴辞書と比較して性格診断を行う。その診断結果とそれによって作成されたエゴグラムの折れ線グラフを図 4.1 のようなプロフィール帳形式で Web ページに表示する。また、これらの出力結果をフォロワー間で共有することができる共有ツイート機能もある。共有ツイート機能では、分析班が作成した Web ページのアドレスと作成されたプロフィール帳がツイートされるものを想定している。

前期はこのサービスを実現するために、Web ページの作成とサーバーに関連する作業を担当する Web 班、TwitterAPI に関連する作業と各班の作成物の連携を担当する API 班、エゴグラムを用いた性格診断を行うためのエゴグラム班の 3 つの班に分かれてシステムの開発を行った。

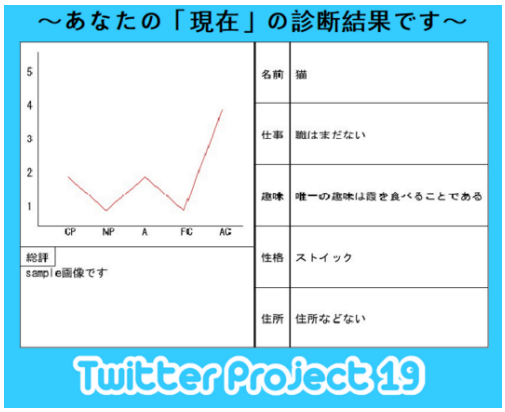

図 4.1　中間発表時の診断結果の画面

後期は上記の性格の診断手順を少し改良した。ホーム画面には診断ボタンに加え、Twitter 診断、エゴグラム、2 つの診断にアクセスでき、アクセスするとそれぞれのことについて説明されている Web ページが表示される。診断タイプ選択画面は、通常診断と集団診断のどちらか 1 つの診断タイプを選択する画面に変更した。通常診断を選択すると、現在診断と現在から未来診断が行われる。現在診断は、最新のツイートから 3200 件のツイートを取得し、それを形態素解析にかけ、エゴグラムの特徴辞書と比較して性格診断を行う。現在から未来診断は、ユーザーのツイートを時系列順に 30 等分して形態素解析し、エゴグラムの推移を用いて現在から未来への性格の推移を予測し性格診断を行う。また、現在から未来診断の表示箇所の下にはスライドバーがあり、これを動かすことで現在から未来への性格の推移を確認する事ができる。これらをまとめて通常診断の診断

結果として、図 4.2 のように Web ページに表示する。集団診断を選択すると、集団診断が行われる。集団診断は、相互フォローしているユーザーからランダムで 15 人選び、選ばれたユーザーの最新のツイートから 100 件を形態素解析し、エゴグラムを用いてユーザーの所属する集団の性格診断を行う。また、集団診断では自分のエゴグラムに基づいて作成したグラフの形と似た人を気の合う人として表示している。これらをまとめて集団診断の診断結果として、図 4.3 のように Web ページに表示する。また、それぞれの診断結果の画面ではユーザーに函館のお勧めの観光地も表示される。

後期はこのサービスを実現するために、Web ページの作成とサーバーに関連する作業を担当する Web 班、未来診断、集団診断、函館の観光地をお勧めする機能作成する API+ エゴグラム班の 2 つの班に分かれてシステムの開発を行った。

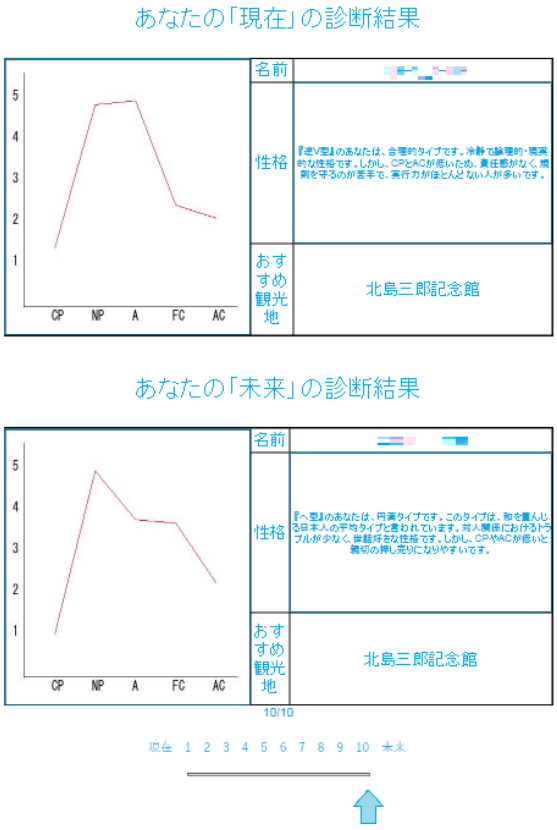

図 4.2　最終成果発表時の通常診断の結果画面

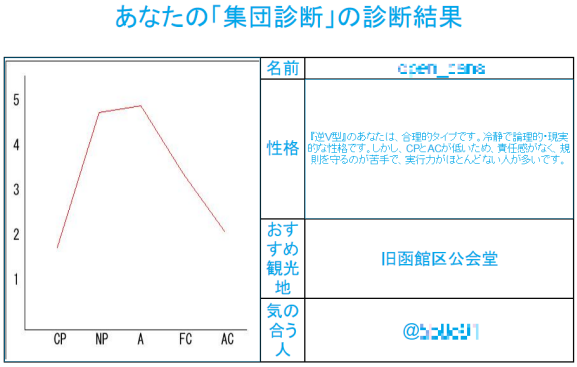

図 4.3　最終成果発表時の集団診断の結果画面

（※文責: 村尾雅都）

4.4　中間発表までの開発について

4.4.1　API 班

API 班は中間発表までに以下のことを担当した。

- ユーザーがログインする機能
- ユーザーのツイートを取得する機能
- エゴグラムの点数化プログラムの作成
- 取得したツイートを形態素解析する機能
- エゴグラムを画像として出力する機能
- エゴグラムの特徴辞書を用いてツイートからエゴグラム作成に使用する値を算出する機能
- 共有ツイートの実装
- Web 班との連携

ログイン機能は TwitterAPI を使用するためのライブラリである TwistOAuth[9] にサンプルコードが載っていたので、それを参考にして作成した。サンプルコードのログイン機能のままの状態では、他の Web ページに遷移したときにログイン情報が失われてしまった。そのため、サーバー上にログイン情報を保存して遷移先の Web ページからサーバーに保存したログイン情報を取得する方法を取った。

ユーザーのツイートを取得する機能では、TwitterAPI でユーザーのツイートを 200 件取得するものを使用した。この API ではツイートの情報が最新のものから 200 件ずつ読み込むために、そのままではツイートを 200 件しか取ることが出来なかった。しかし、ツイートに付加されているツイート ID を使用し、取得したツイートの ID よりも古いツイート 200 件を取得することにした。これによりユーザーのツイートを最大で 3200 件取得することが可能になった。

上記で取得したツイートをエゴグラム班が作成した特徴辞書を用いて、該当する語の数をカウントする機能を作成した。特徴辞書は 5 種類あり、各辞書毎に該当する語をカウントし、その数を用いてエゴグラムを作成した。

前述したツイートを点数化するプログラムを用いて得られたデータを基に、エゴグラムを作成するプログラムを作成した。ツイートを点数化したものは最大値が決まっていないため、もっとも点数が高い場所を基準に全体の比を崩さないようにグラフを作成するようにした。

（※文責: 赤坂尚衡）

共有ツイートの実装では、このプロジェクトは Twitter に関したアプリケーションを作成しているので、出力された結果を Twitter 上で共有することも大切だと考えた。よって、最初は Web ページに付けることができる Twiter 公式の共有ボタンを利用しようとしたが、この機能は画像を付けてツイートすることができなかった。診断結果の出力される表は画像形式になっているため、この画像をふまえてツイートすることが重要だった。そこで、TwitterAPI を利用して画像を付けてツイートする方法を見つけたので、参考にして実装することができた。

Web 班との連携では、Web 班がプロトタイプとして作成したページを組み込むために、eclipse 上で PHP のコードとは別に HTML を組み込むことができるので実装した。ページ遷移のリンク

や、API 班が作成した共有ツイートボタンを貼り付けてしっかり動くかなどを確認した。

（※文責: 石橋笙）

4.4.2　エゴグラム班

エゴグラム班は、性格を診断する機能の作成を課題に活動した。エゴグラムとは精神分析学者のエリック・バーンによって提唱された交流分析という心理学理論をもとに、心理学者のジョン・M・デュッセイが考案した性格診断法である。具体的には人間の心の状態を、

- CP(Critical Parent)：批判的な親。責任感や義務感が強く、他人に批判的。
- NP(Nurturing Parent)：養護的な親。他人を思いやり、世話好きで親切。
- A(Adult)：大人。物事を客観的、論理的に考え、合理的である。
- FC(Free Child)：自由奔放な子供。明るく好奇心旺盛で、ユーモアがある。
- AC(Adapted Child)：従順な子供。言いたいことを我慢し、従順で遠慮がち。

の 5 つに分け、それぞれの相対的な値によって性格を表現する性格診断方法である。[11] 診断方法としては、ユーザーのツイートを形態素解析し、CP、NP、A、FC、AC の状態ごとに事前に用意した特徴辞書と比較する。その後、一致するものを抜き出し、それぞれの項目に対して点数をつける。それをグラフで表し、その形から性格を診断する。以下に例を示す。

例えば、図 4.4 に示す女の子のツイートを形態素解析すると、図下段な結果が得られる 。それを特徴辞書と比較すると、図 4.5 のように FC が 2 つ、A が 1 つ、CP が 1 つ抜き出される。特徴辞書と一致したもの 1 つを 2 点とすると、図 4.6 のようにあらわすことができる。そして、グラフの形から大まかなパターンに分けて性格を診断する。

診断の際に用いる特徴辞書に関しては先行事例を調査した結果、機械学習を用いて作成した例があったので、特徴辞書の作成に機械学習を用いることを目標としていた。[12] 具体的には、手作業で作成した教師データをコンピュータに学習させ、実際のツイートを形態素解析したものを与えてやることで、その中からエゴグラムの 5 つの項目に対して点数の高い言葉を特徴辞書に自動で登録するというものである。しかし、まずは動くものを作るという考えから、実際は特徴辞書を手作業で作成し前期の時点で約 500 語を登録した。ツイートを特徴辞書と比較し点数化するために用いるプログラミング言語は、統計的な処理が簡単にできるということで R 言語を使うことにした。[13] しかし、実際に形態素解析されたものを特徴辞書と比較しテキストに出力することはできたが、処理速度が遅いものしか作れなかったので、その部分は API 班に処理を依頼した。点数の付け方は登録されている言葉すべてに対して 1 点とした。理想的には、出現頻度や出現回数などから言葉ごとに点数を変えるべきだったがそこまで作成することはできなかった。

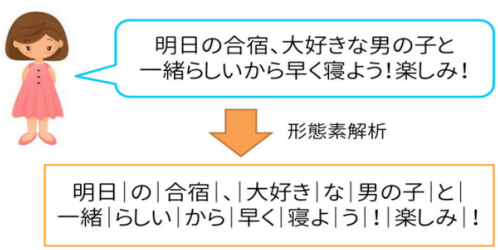

図 4.4　ツイートを形態素解析した結果

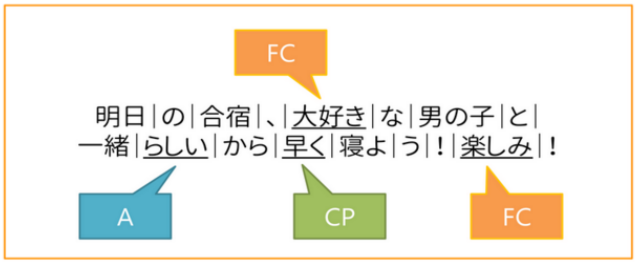

図 4.5　特徴語の抜き出し

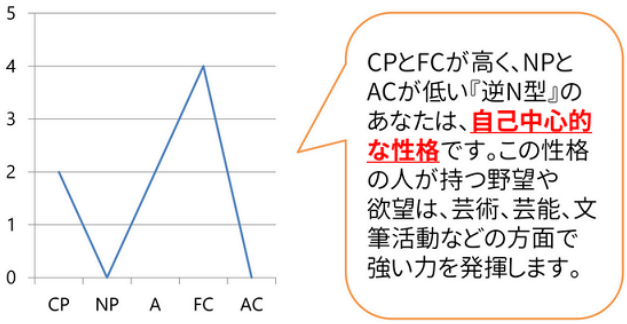

図 4.6　診断結果

（※文責: 川向達也）

4.4.3　Web 班

Web 班では Web アプリケーションを実行し、出力結果を表示するためのページを作成した。アプリケーションの実現にはホームページ、診断の形式選択のページ、診断結果を表示するページの 3 つに分類してページを作成した。ページの作成には HTML を使用し、ページ全体のデザインは Twitter らしさを表現するため、明るい青色や白を用いて文字や背景色を設定したり、診断するためのボタンを作成するといった工夫を取り入れた。これらを各ページで統一し、班のメンバー同士で共有・確認をしながら作成した。アプリケーションの出力結果を表示するためのページに関しては診断項目が「現在」、「未来」、「集団」の 3 つであるため、図 4.1 のようにそれぞれの項目ごとに表示するページを用意した。図 4.1 に示すページはユーザーが「現在」の診断形式を選択した際に、アプリケーションの出力結果を受けて、それをプロフィール帳として表示するものである。図 4.1 ではプロフィール帳にある各項目に関する出力結果の他に、エゴグラムに基づくグラフも画像出力しているが、作成するアプリケーションとの連携は実現できなかった。そのため Web ブラウザ等で作成した HTML ファイルを表示させることは可能だが、各ページとの連携が未完成の状態に終わった。

（※文責: 小川聖司）

4.5　中間発表のスライド・ポスター作成について

4.5.1　ポスター

中間発表のポスターでは、全体のポスター 1 枚・検索班のポスター 1 枚・分析班のポスター 1 枚と 3 枚あり、その中の分析班のポスターを作成した。中間発表ではポスター・スライドともに班ごとに色分けをし、分析班はオレンジ色に統一した。中間発表の準備のための作製は 1 ヶ月前に始め、検索班の方やスライド担当の方と調整を行ったり、先生方からもアドバイス頂くなど多数協力していただいた。

初めに、中間発表に用いるポスターとスライドを作成するために、提案するサービスをまとめた要件定義書を作成した。その要件定義書では問題提起・解決案 (新しいサービス案)・最終イメージ・現状と展望の 4 つの項目を作り、ポスターにまとめるときには背景・提案するサービス・現在までの取り組みと展望、という 3 つの項目を作成した。全体の文字の大きさを最小でも 28pt にし、情報量の多さよりも文字の読みやすさを優先して作成を進めた。

背景の項目では、提案するサービスを開発するにあたっての経緯を述べるために、メンバー内で話し合ったツイートの不満点やどんなことが物足りないか、ということから既存のツイート診断アプリケーションの物足りなさをまとめた。そこで代表的なツイート診断アプリケーションであるアプリメーカーとツイートプロファイリングを例に挙げて、以下のような物足りなさを挙げた。「アプリメーカー」

- 取得したツイートを解析して得られたデータはさらに解析することはできない
- 提供された解析はできるがそれ以外の解析はできない

「ツイートプロファイリング」

- 解析方法は提示されているが、解析するツイート数が 500 件とすくないためより多くの情報が欲しい
- 自ら解析してみるともっと多くのツイートを解析できることが判明

以上のことから

- ツイートから何かわかることはないのか
- ツイートから多くの特徴を知ることはできないのか
- 既存のツイート診断アプリケーションで解析出来ていない部分を解析したい

という、分析班での改善案の提案までの流れをポスターの背景に載せた。

提案するサービスの項目では、開発していた Web アプリケーションで実行できることをポスター閲覧者に理解してもらいやすく工夫した。サービスの名前では、開発していた性格診断アプリケーションでメインとなる「ツイート」と「エゴグラム」を強調し、特にエゴグラムにおいてはポスター閲覧者にとっては普段聞くことのない用語であり難解であると考えたので、注意書きを入れた。次に、分析方法について述べた後、診断できる 3 つの性格診断についての詳細を述べた。診断方法については細かい点までは言及せずにポスター制作時点でできた分析方法を述べた。そして、中間発表の時点では現在の性格診断は実行できるが、他の 2 つの診断は開発段階であり、機能の実行は不可能なものの将来的にはどのような機能が実行できるようになるかを述べた。さらに、ポスター右側には診断結果を共有したツイートを載せた。ここでは既存のツイート診断アプリケーショ

ンにも存在する診断結果を共有できる機能があることをアピールすると同時に、中間発表時点でのサービスの診断結果画面を見ていただけるようにし、ポスター閲覧者が一目でユーザのツイートからどんなことが診断できるかを理解できるようにした。

現在までの取り組みと展望の項目では、中間発表までに終わらせたこと、未完了のこと、プロフィール帳を作成できるようにする、という 3 つの取り組みと展望を挙げていった。図 4.7 が実際に中間発表にて使用した分析班ポスターである。

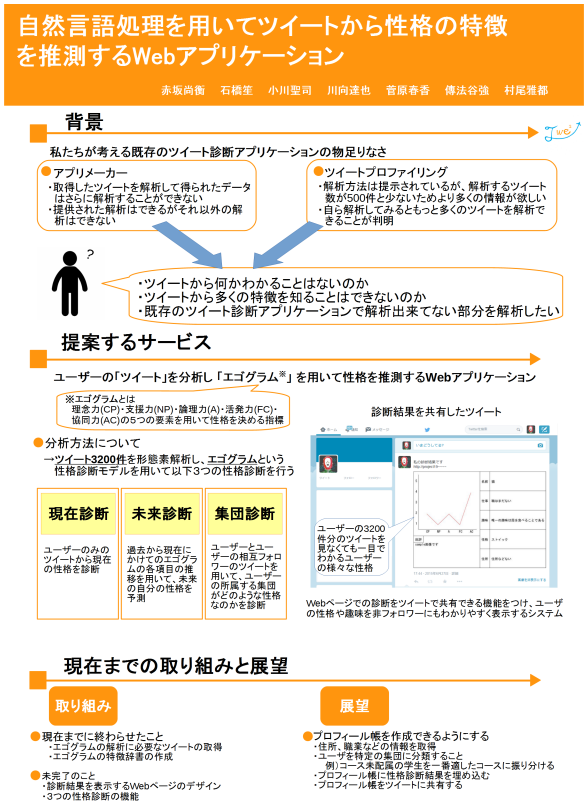

図 4.7　中間発表で使用したポスター

（※文責: 傳法谷強）

4.5.2　スライド

中間発表のスライドは全体の概要・分析班の説明・検索班の説明・まとめの大きく 4 つに分かれている。ここでは、分析班の説明のスライド作成について述べる。

中間発表で使用した分析班のスライドでは、分析班が開発している Web アプリケーションについての説明・現状の取り組み・今後の展望を説明した。スライド作成は中間発表の 1 ヶ月前から開始し、ポスターと内容をあわせるためポスター作成の担当者と繰り返し調整を行った。また、検索班のスライド担当者や先生方からも図の作り方や専門用語の説明の仕方など多くのアドバイスを頂き多数協力していただいた。

具体的に工夫した点として、分析方法の説明がある。スライド発表において、性格の分析方法を技術用語を用いて説明しても閲覧者には難解であると感じていた。そこで、具体的な例として女の子の 1 ツイートをもとに、どのような性格であるかを分析する過程をわかりやすく説明した（図 4.4、図 4.5、図 4.6）。図 4.4 は、女の子の 1 ツイート「明日の合宿、大好きな男の子と一緒らしい

から早く寝よう！楽しみ！」を形態素解析したらどのような形態素に分かれるかを記号｜で区切って示している。図 4.5 は、形態素解析したツイートから特徴語を抜き出している。この例では「大好き」「楽しみ」が FC、「らしい」が A、「早く」が CP と判定される。図 4.6 は、図 4.5 で判定した特徴語をもとに CP、NP、A、FC、AC のグラフを作成し、そのグラフの形から性格を判定している。この例では、『逆 N 型』のグラフとなり自己中心的な性格であると判定される。このように具体例を用いて説明することで閲覧者の理解の手助けになるようなスライド作成を行った。

（※文責: 菅原春香）

4.6　中間発表について

2015 年 7 月 10 日、公立はこだて未来大学内にて中間発表が行われた。中間発表に向けてポスターの作成や発表するためのプレゼンテーションの資料の作成を行った。以下に、中間発表での各人の担当・中間発表内や評価者フィードバックで寄せられた意見や質問・中間発表後の反省会の内容を示す。

（※文責: 菅原春香）

4.6.1　各人の役割

赤坂は中間発表でのメンバーの仕事の担当振り分け、進捗確認、加えてプレゼンテーションの発表を行った。

菅原は中間発表で発表するためのプレゼンテーションの資料の作成及びプレゼンテーションの発表を行った。

傳法谷は中間発表で使用するポスターの作成を行った。

小川・村尾・石橋・川向は中間発表で質疑応答の内容を記録し文書に起こした。また、中間発表で他の発表グループの発表技術に関する記録をとり他の班員に共有した。

（※文責: 菅原春香）

4.6.2　寄せられた意見・質問

中間発表では多数の意見・質問が寄せられた。まず、発表内容に関しての意見・質問内容を以下に示す。

- ローカライズと関係性があるのか。
- ツイート分析の際にフィルタリングしているのか。
- ツイートを 500 件とるのと 3200 件とることに違いはあるのか。
- エゴグラムのアルゴリズムとはどういうものなのか。
- エゴグラムの信頼性は高いのか。
- 分析班と検索班の連携を考えてみてはどうか
- 順調そうだと分かった
- 今後の展開に期待
- 簡単に飲食店を見つけるための手法の開発は良い課題だと思う

- プロフィール帳のプロトタイプのようなものがあれば見たかった
- 分析班の用語がやや多かった
- 2 つの組み合わせると良いものができそう
- アプリに名前を付けた方が良い
- いまいちゴールが見えない
- システムの評価の話がなかった
- ツイッターの利点を生かしている
- どうすれば情報を入手しやすくなるかを考えるともっとよくなる
- 技術的な言葉の説明が必要だと思う
- 現在の状況と展望がわかりやすく説明されていた
- 後期のスケジュールが示されていない
- 現在の問題と作成するソフトの良さの比較があってわかりやすかった
- 面白いことをやっている
- 問題提起と目標がしっかりしていた
- 有用性のある内容で完成に期待が持てる
- 容易に実現可能であると思えた
- 理想的かつ現実的な研究で興味が持てた
- ユーザーへのメリットがあるのか
- 20 歳を過ぎると性格はあまり変化しないと言われているが、どうなのだろう
- 3200 件を取得することは連続ではできないのか？
- エゴグラムがどれほど正確で効果があるのか知りたい
- エゴグラムの信憑性が怪しい
- ツイートプロファイリングはどのような目的として使われているかが不明瞭
- データの分析を行い、その結果を活用するのは面白いと思う
- 性格がばれるのはぶりっ子にはつらい
- 先行事例が既にあるのでは？
- 分析する数を減らすことで効率化できるのでは
- 分析班についてもっとデータが欲しい
- 分析班に関してはプライバシーに注意してもらいたい
- 分析班は既存のサービスとの差別化がされているのが良い
- 未来診断が面白そう
- 話題作りになってよさそう

次に発表技術に関しての意見・質問内容を以下に示す。

- スライドに動きがあると良いと思う
- スライドの内容をただ話すだけではなく、掘り下げてみてはどうか
- はきはきとしていて聞きやすかった
- ページ番号があってよかった
- 画像を多用していて見やすくまとめられていた
- 少し声が聞こえにくかった
- 少し単調に感じた
- 内容が理解しやすかった

- 発表が短い
- 班のタイトルが長いのでもっとシンプルなタイトルを言ってからでもよいと思った
- 分析班と検索班に分けてあるのが良いと思った
- 例があってわかりやすかった
- 例が多くてわかりやすい
- 話さない内容の文章はいらないと思う
- わかりやすい
- 聞き取りやすい
- よくリハーサルがされている

（※文責: 菅原春香）

4.6.3 中間発表後の反省会

中間発表後には、中間発表の質疑応答の際に寄せられた質問や評価者フィードバックを中心に反省会を行った。中間発表時で発表した Web アプリケーションにはローカライズ要素が欠如していることが 1 番の問題点としてあげられた。その結果分析班では、中間発表時のアイデアに加え、ローカライズ要素を含んだアイディアを夏季休業期間中に考えてくることになった。

（※文責: 菅原春香）

4.7 最終成果発表までの開発について

4.7.1 API 班 + エゴグエラム班

現在から未来診断

この機能はユーザーのツイート 3200 件から未来の性格を推測するものである。手法としては、まず 3200 件のツイートを TwitterAPI を用いて取得し、時系列順に並べてから 30 等分する。そして、30 等分したものそれぞれでエゴグラムを作成し、回帰分析を行うことでツイートから未来の性格を推測している。また、最適な回帰直線を引くために最小二乗法を用いた。この場合の最小二乗法は、データを直線で最良近似することである。この直線から未来の性格を推測している。この機能を作成するために、当初は統計的な処理が簡単にできる R 言語を用いて実装した。しかし、サーバー上で動かすためには exe ファイルにしなければならなかったのだが、その方法が見つからなかったため、exe ファイルにする方法を見つけられた python で実装し直した。

また、性格が変化していく過程を見られるようにするために、推測して得られたデータを加えて回帰分析を繰り返し行うことで、さらに未来を予測する機能も python で実装した。開発に取り掛かる前に、実際に得られたデータをプロットし確認した結果、グラフの推移がランダムウォークのような形や、株の時系列変化のようなグラフになっていることもあったため、時系列解析を行うことも考えたが、アプリケーションに組み込み、処理を自動化できなかったために、最終的には回帰分析を用いてこの機能を完成させた。

（※文責: 川向達也）

集団診断

TwitterAPI では、フォロワーの情報も参照できるため、この機能を性格診断に活用したいと考え、ユーザーの周りを取り巻くフォロワーは総じてどのような性格であるかを診断する集団診断を開発することにした。集団診断の分析方法の手順は以下の通りである。

1. 診断しているユーザーの相互にフォローしている人の ID を取得し、その中から 15 人分の ID をランダムに選出する。
2. 選出した 15 人のフォロワーからそれぞれ最新の 100 件のツイートを取得し、ひとつのテキストファイルにまとめる。
3. 通常の診断同様、テキストファイルを形態素解析にかけて特徴辞書で要素ごとに点数化を行い、性格を診断する。

この結果、選出したフォロワー全体の性格を診断し、出力することができた。

集団診断では、フォロワーのまとまりを考えて性格を診断していたが、ユーザーと一番性格の近いフォロワーを特定して診断するのも面白いという意見があり、これを気の合う人の診断として開発を行った。気の合う人の診断の手順は以下の通りである。

1. PHP において、TwitterAPI を利用し、相互フォロワーの最新のツイート 100 件ずつ配列に格納する。　それらフォロワーのツイートはそれぞれテキストファイルに保存されているので、それらをいっぺんにまとめるようにパスを設定し、Processing に送る。
2. Processing の中で渡されたツイート群を人数ごとに区切り、点数化を繰り返し行う。そして、全員分の各要素の点数をまとめて PHP に返す。
3. PHP で返ってきた点数を人数分に区切って配列に格納する。そして、ユーザー自身の各要素の点数との比較を繰り返し行い、気の合う人を算出する。

点数の比較は各要素の点数の差分の絶対値の和を計算している。この和が一番小さい値となった相互フォロワーをグラフの形の差異が小さい、すなわち性格が近いとして一番気の合う人とした。そして出力されるのは一番気の合う相互フォロワーの ID となっている。

（※文責: 石橋笙）

函館のおすすめ観光地

この機能で出力されるお勧めの観光地は金森赤レンガ倉庫群、函館山、北島三郎記念館、旧函館市公会堂、トラピスチヌ修道院の 5 つである。ユーザーのツイートがこれらのどの観光地に行った人と同じ傾向のツイートかを分析し、その観光地をお勧めとして表示する。分析には機械学習を用いた。当初の計画では、出力する観光地 (以下ラベルとよぶ) の数は金森赤レンガ倉庫群、函館山、北島三郎記念館、旧函館市公会堂、トラピスチヌ修道院、五稜郭タワー、谷地頭、湯の川、松前城、北洋資料館の 10 個であり、これらのどれか 1 ヶ所を訪れたと推測される 446 人のツイートを機械学習で学習させる際に用いるデータ (以下トレーニングデータとよぶ) とした。機械学習に用いるために、各ユーザーのツイートデータをベクトル（以下特徴ベクトルとよぶ）として表現した。ツイートに現れた単語を集めることで特徴語辞書を作成し、その辞書から各ユーザの特徴ベクトルを

定めた。特徴ベクトルの次元は辞書の単語数とする。

これらを R で作成した機械学習の技法であるサポートベクトルマシーンや線形判別分析、ランダムフォレストで識別を行おうとしたがメモリエラーと出力された。R では基本的にメモリ上でデータを保持し、計算を実行しており、オブジェクトなどは値渡しするためにメモリを大量に消費する [10]。そのため、メモリが足りずにエラーが出たものと推測される。これを改善すべく、特徴語辞書を削減をして使用するメモリを減らす方策を採った。

特徴語を削減するため、形態素解析で出来た単語のうち、名詞と識別されたものだけを使用することにした。これにより特徴語辞書の語数が 221028 語から 107944 語になった。この特徴語辞書を使い、R で作成した前述の機械学習のアルゴリズムで学習させたが、線形判別分析ではメモリエラーが発生し正答率を求めることができなかった。一方、サポートベクトルマシーンではトレーニングデータと性能を測る際に用いるデータ (以下テストデータとよぶ) を同一のものを使用したところ、正解率は 49% で実行時間は 10 分だった。また、ランダムフォレストではトレーニングデータとテストデータをサポートベクトルマシーンの時と同じにして正解率を測ったところ 27.58% であった。しかし、学習中にメモリエラーが出力されていたのにもかかわらず、正解率が出力されていたため、情報の信頼性が低いことは明らかであった。さらに、学習に 6 時間以上を要した。Web アプリケーションで 10 分以上の待ち時間が発生するのは許容出来ないため、サポートベクトルマシーンとランダムフォレストは更なる改善が必要だった。

線形判別分析とランダムフォレストに関しては、メモリ使用量の特に多い R 以外の言語であればメモリエラーを出力しないのではないかと推測し、線形判別分析を python で作成することにした。また、主成分分析で次元を削減することやランダム行列をかけて次元を削減すること、ランダムフォレストで算出することのできる特徴の重要度を利用し重要度が 0 となったものを特徴語辞書から削除する等してデータ量の縮小を図った。

python で作成したランダムフォレストで出力された特徴の重要度を使い、次元数を減らしたものは 1428 次元であった。また、主成分分析を行い、次元数を減らす方法では特徴ベクトルの次元は 107944 次元から 445 次元に削減されたが主成分分析を用いて特徴ベクトルを削減する計算に 2 時間、ユーザーのツイートを取得してから 107944 次元の特徴ベクトルにして主成分分析の結果を利用して次元を削減するのに 10 分以上かかり、Web アプリケーション上での使用を断念した。ランダム行列をかけて次元数を削減する方法では特徴ベクトルの次元数を 10 とした。特徴語辞書の削減はこれ以上減らすことは出来ないと判断し、これ以降は正解率を高めることに焦点を当てて開発を行った。

ここまでの特徴ベクトルは特徴量を単語の出現回数としていた。単語の出現回数を特徴量としたものが、ランダムで選んだ時と変わりない正解率だったときのために、別の特徴量を持つベクトルとして文書内の単語の重要度 (以下 TFIDF 値とよぶ) を計算する機能を実装し、特徴語辞書の各単語の TFIDF 値を特徴量とした特徴ベクトルを作成した。その後、表 4.1 に示す特徴ベクトルを用意して python で作成した線形判別分析、ランダムフォレストを使い正解率を計測した。この時、前述した 446 人のユーザー以外でラベルの場所に行ったと推測されるユーザーのツイートを取得して特徴ベクトルを作成した。10 個のラベルのうち、北洋資料館はこれ以上ツイートを取れなかったため、北洋資料館を除いた 9 つで新たに各 10 件ずつデータを取ってテストデータとして使用した。

以下では、単語の出現回数を特徴量にした 107944 次元の特徴ベクトルを Normal_L、単語の出現回数で作成した特徴ベクトルをランダムフォレストで次元削減したものを Normal_S、TFIDF 値を特徴量にした 107944 次元の特徴ベクトルを TFIDF_L、TFIDF 値を特徴量として作成した特

表 4.1　作成した特徴ベクトル

単語の出現回数を特徴量にした 107944 次元の特徴ベクトル
単語の出現回数で作成した特徴ベクトルをランダムフォレストで次元削減したもの
TFIDF 値を特徴量にした 107944 次元の特徴ベクトル
TFIDF 値を特徴量として作成した特徴ベクトルをランダムフォレストで次元削減したもの
107944 次元の特徴ベクトルにランダム行列を掛けたもの

徴ベクトルをランダムフォレストで次元削減したものを TFIDF_S、107944 次元の特徴ベクトルにランダム行列を掛けたものを random とする。

表 4.2　線形判別分析、ランダムフォレスト、サポートベクトルマシーンの各特徴ベクトルの正解率

特徴ベクトル	線形判別分析 (python)	ランダムフォレスト (python)	サポートベクトルマシーン (R)
Normal_L	メモリエラー	29.21%	17.58%
Normal_S	21.33%	20.00%	17.39%
TFIDF_L	メモリエラー	メモリエラー	メモリエラー
TFIDF_S	13.33%	23.33%	14.29%
random	13.04%	12.02%	14.13%

決定木を 1000 本のランダムフォレストが最高 29.21% を記録し、他の機械学習のアルゴリズムで作成した正解を判別するもの（以下識別機とよぶ）の最大正解率である 21.33% よりも 5% 以上高い精度を見せた。他の特徴ベクトルと比べて精度がよくなかった random は使用しないことにした。また、R で作成したサポートベクトルマシーンは最初に正解率を計測したときはトレーニングデータとテストデータが同じだったため、再度計測を行ったところ正解率は他の識別機よりも低かったため、R で作成したサポートベクトルマシーンも使用しないことにした。

表 4.2 の識別方法で本当に識別できているのかを検証するため、人が見ても明らかにユーザーが分けられそうなラベルを 3 つ用意して、精度を測ることにした。トレーニングデータとして各観光地に 40 件ずつ、テストデータを 10 件ずつ、ラベルとして函館競馬場、函館で行われる日本ハムの野球の試合、函館黒船イベントの 3 つで行ったところ、表 4.3 のような結果が得られた。

表 4.3　線形判別分析、ランダムフォレストの各特徴ベクトルの正解率

特徴ベクトル	線形判別分析	ランダムフォレスト
Normal_L	51.61%	83.65%
Normal_S	58.06%	80.97%
TFIDF_L	メモリエラー	メモリエラー
TFIDF_S	51.61%	80.65%

この結果より、ランダムで選ぶよりも判別できていることがわかった。目標をラベル数 5 で 50% 以上判別することと定めて正解率の向上を図ることにした。

残すラベルを選ぶ際に調べたところ、松前城と北洋資料館はこれ以上取れるツイートがないため削除して、残りのラベル 8 個の中から 5 個を選ぶことにした。8 つの中から 5 つを選ぶ組み合わせ

は 8C5 で 56 通りあり、これら全てで特徴ベクトルを作成し、識別機にかけた。上記の 3 ラベルで計測した際に線形判別分析とランダムフォレストの識別率に顕著な差があったために、識別機はランダムフォレストのみを使用して分析した。木の本数は 10000 本である。使用した特徴ベクトルは単語の出現回数を特徴量にした 107944 次元の特徴ベクトルと、単語の出現回数で作成した特徴ベクトルをランダムフォレストで次元削減したものの 2 つで、最も高い正解率は 46% であった。

その後、特徴ベクトルの末尾にエゴグラムを作成するときに使用した、5 つの指標の算出方法を用いて計算した値 (以下エゴデータとよぶ) を付加して新たな特徴ベクトルとして、再度同じ方法で計測した。その結果、目標である 50% を超えたものは 2 つあり、ラベルが「金森赤レンガ倉庫群、函館山、北島三郎記念館、旧函館市公会堂、トラピスチヌ修道院」の組み合わせのものと、「金森赤レンガ倉庫群、函館山、北島三郎記念館、トラピスチヌ修道院、谷地頭」の組み合わせのものであった。木を 20000 本にしたランダムフォレストで再度両者を計測したところ、「金森赤レンガ倉庫群、函館山、北島三郎記念館、トラピスチヌ修道院、谷地頭」が 48.9%、「金森赤レンガ倉庫群、函館山、北島三郎記念館、旧函館市公会堂、トラピスチヌ修道院」では 52% となった。

後者のラベルがトレーニングデータの組み合わせで偶然 50% を超えた可能性を考慮して、トレーニングデータとテストデータを足して 8 割をランダムに選んでトレーニングデータとし、残りをテストデータとした。これを 100 回繰り返して検証を行った結果、正解率の平均は 52.33、分散は 0.003358、標準偏差は 0.0579 であり、偶然 50% を超えたわけではないと推測した。また、特徴ベクトルにエゴデータを付加したものでは、単語の出現回数を特徴量にした 107944 次元の特徴ベクトルと単語の出現回数で作成した特徴ベクトルをランダムフォレストで次元削減したものの 2 つにおいて、t 検定を行ったところ有意差が認められ (t=14.09,df=54,p<0.01)、Web アプリケーションには正解率が高かった特徴ベクトルである、単語の出現回数で作成した特徴ベクトルをランダムフォレストで次元削減したものを用いることにした。

現在の正解率を向上させるために複数の識別機を用意し、各々で予測を行い多数決をとる方法を採用した。前述した正解率が 50% を超えたラベルの組み合わせで複数回識別をして多数決をとることで試行回数の半分が正解のラベルに、失敗のラベルは一か所に集中せずに分散して結果として正解率が高まるのではないかと推測した。前述の組み合わせで作成した、正解率が 50% を超えたものでを 3 つ識別機を作成して多数決を取ったが識別機が 1 つで多数決を取らなかったときと変化が見られなかった。多数決の内訳を確認したところ、1 つを全てのテストデータにおいて全識別機が同じラベルを選んでいた。1 つの識別機で 3 回識別を行い多数決をとったところ、正解率が 1 つの識別機で 1 回しか判別を行わなかったものと正解率は変わらず、全てのテストデータにおいて全識別機が同じラベルを選んでいた。同じトレーニングデータでは正解率の上昇が見込めないものと推測し、トレーニングデータとして用意されているデータから、ランダムに 8 割分選んだデータを使って作成した識別機を 9 個作成して多数決を行った。識別機の数を増やすほど識別率の向上は見られたが 9 個まで増やした段階でトレーニングデータを 10 割使用して作成した識別機 1 つの識別率には及ばなかった。正解率が上がらなかった理由として、ランダムフォレストでの正解率は苦手なデータと得意なデータの比であって苦手なデータでも何回もやれば 50% 正解するのではないかと推測した。以上の結果として多数決での正解率の向上はできなかった。

（※文責: 赤坂尚衡）

4.7.2 Web 班

ホーム画面の作成

ユーザーがアクセスして最初に表示されるページの作成を行った。ページを作成するにあたっては他の性格診断の Web サイトを参考にしつつ、どういったデザインにするかを話し合った。議論の結果、ページはわかりやすく見やすいものが良いという結論になったため、基本色は白と Twitter らしさを表現する水色を使用して全体が明るくなるようにした。画面構成はできるだけシンプルにするため、タイトル、メニュー、診断開始ボタンの 3 つで構成した。タイトルに使用した「Twitter 診断」の文字は公式 Twitter のフォントと同じものを採用しようとしたが、漢字も同じフォントに変更するのは困難であった。このため Twitter の箇所だけフォントを変更し、診断の箇所は別のフォントを使用した。メニュー項目は診断に関する説明をユーザーに伝える必要があったため、「Twitter 診断とは?」、「エゴグラム」、「2 つの診断」の 3 つに決定し、それぞれの項目をクリックすると、その説明ページに飛べるようにした。画面中央には「診断する」のボタンを配置し、マウスカーソルをかざすとボタンの色が明るくなり、少し浮き上がって見えるような工夫を施した。

（※文責: 小川聖司）

「Twitter 診断とは?」のページ作成

このページでは分析班で作成した Web アプリケーションの診断内容と、形態素解析についての説明を行った。画面の上半分では Twitter 診断、下半分で形態素解析の説明が表示されるようにした。それぞれの説明部分では文章だけではなく実際の動作画面といったものを図として採用し、内容がわかりやすくなるようにした。Twitter 診断の説明部分ではエゴグラムや形態素解析といった用語を使用しているため、エゴグラムの箇所をクリックすると「エゴグラム」のページに飛び、形態素解析の箇所をクリックすると形態素解析の詳細部分に飛べるようにした。

（※文責: 傳法谷強）

エゴグラムのページ作成

このページでは Twitter 診断で使用されているエゴグラムの詳細を記述した。Twitter 診断では、エゴグラムをユーザーのツイートから特徴を抽出して性格を判定する際に使用し、結果表示では説明文とグラフが同時に表示されるようになっている。しかしエゴグラムでは性格を 5 つの要素 (CP、NP、AF、C、AC) に分けているため、結果で表示されるグラフを見ただけでは、それぞれの要素が何を意味するのかを把握しにくい課題があった。そこでこのページでは、エゴグラムに関する説明がわかりやすくなるような内容にすることをテーマとした。ページの構成は画面の上半分でエゴグラムの解説を行い、下半分でエゴグラムで使われている 5 つの性格の要素を説明する画像を配置をした。エゴグラムの説明文は文章量が多いと逆にわかりづらくなると判断したため、できる限りコンパクトな内容にした。図 4.8 のように、エゴグラムの 5 つの性格要素を説明する画像は、エゴグラムの 5 つの要素である CP、NP、AF、C、AC がそれぞれ意味する内容を一目で理解できるように、CP を支配性、NP を寛容性、A を論理性、FC を奔放性、AC を順応性と表現した。またよりわかりやすくするために、各要素を言葉だけでなくイメージ画像と組み合わせるようにした。それぞれ CP に拳、NP にハート、A に考える人、FC に自由の女神、AC に子供のイメー

ジ画像を採用した。ページの最下部にはホーム画面にすぐ戻れるようにボタンを配置した。

図 4.8　エゴグラムの 5 つの性格要素

（※文責: 小川聖司）

2 つの診断のページ作成

分析班で作成した Twitter 診断が行う診断の種類の説明するページの作成を行った。Twitter 診断では現在から未来の性格の診断 (通常診断) と、集団における性格診断 (集団診断) の 2 種類があるため、このページではユーザーにこれらの内容を説明することをテーマとした。ページの構成として最初に提案されたのは、画面の左半分で通常診断の解説を行い、右半分で集団診断の解説を行うものであった。シンプルな内容にした方がユーザーに説明がわかりやすく、伝わりやすいと当初は考えていたため、このような提案がなされた。実際にこの案を基にページの実装を行い、他の班員や教員の複数人にレビューをしてもらった。しかしレビューを通して挙げられた意見は、ページにあるのは説明文だけであり、文章だけでは内容を把握しづらいことが挙げられた。またページ全体が物足りない印象があるので、工夫を施した方が良いといった意見も挙げられた。これらの意見を参考にしてページの構成について再考した結果、実際に診断を行って表示される結果の画像を診断の説明文と一緒に表示する提案がなされた。最終的なページ構成は、画面上半分で現在から未来の性格の説明とその診断結果の画像を配置し、画面下半分で集団の性格診断とその診断結果の画像を配置する形式になった。ホーム画面にもすぐ飛べるように、ページの最下部にはボタンを配置した。こうして出来上がったページを再度他の班員の複数人にレビューしてもらうと、診断結果の画像が説明文と一緒になっているため、わかりやすくなったという意見をもらうことが出来た。

（※文責: 傳法谷強）

形式選択ページの作成

Twitter 診断で行う診断形式を選択するページを作成した。Twitter 診断で行える診断は、ユーザーの現在から未来の性格の診断と、ユーザーが所属する集団の性格の診断の 2 種類である。ここではユーザーがどちらの形式で診断を行うかを選択できるように実装を行った。ユーザーの現在から未来の性格を診断する場合を通常診断とし、ユーザーのフォロワーデータから、そのユーザーが所属する集団の性格を診断する場合を集団診断とした。当初の Twitter 診断は診断する形式が 3 つあり、ユーザーの現在の性格診断と、未来の性格診断、集団の性格診断が行える予定であった。そのため、このページの初期のページ構成案は現在、未来、集団の 3 項目からユーザーが形式を選択するというものであった。ページの背景は全体に水色を使用し、現在と未来と集団の 3 つのボタンを配置しており、それぞれのボタンにマウスをかざすと、それぞれの診断の説明文がフェードアウトする仕様になっていた。しかし後に現在の性格診断と未来の性格診断を統合して、現在から未来の性格診断 (通常診断) の形式に変更されたため、「通常診断」と「集団診断」の 2 つから選択する形式となった。またレビューを通して、説明文がフェードアウトする仕様が不要であると意見が挙がったため、この要素は不採用とした。ページ全体に使用していた水色は、さらに見やすい配

慮を行えるという意見から、白へと変更した。このページからもホーム画面へ戻りやすくするために、ページ最下部には戻る用のボタンを配置した。

（※文責: 小川聖司）

結果表示のページ作成

ユーザーがホーム画面から診断形式のページに移動し、選択した形式の診断結果を表示するページの作成を行った。このページでは当初、それぞれの診断形式に応じて、ページを分けて作成する方針であった。Twitter 診断では初期の段階で現在、未来、集団の 3 つの診断形式を実装する予定であったため、結果表示のページも 3 つに分割して作成された。この段階で開発されたページの構成は 3 つとも共通しており、ユーザーの特徴を表形式で表示する内容だった。表形式で表示する内容は以下の項目であった。

- エゴグラムによる性格を表した画像形式のグラフ
- 名前 (Twitter 内で使用している ID)
- 住所
- 趣味
- 性格
- 総評

他の形式で診断を行えるように、表だけではなく、異なる形式の診断ページに移動できるボタンも配置した。また他にも診断結果を共有ツイートするボタンと、診断結果の根拠となるツイートを表示するボタンを配置していた。

しかし Twitter 診断の開発が進むにつれて、当初予定していた実装機能にいくつか変更点が加わった。まず Twitter 診断で行う現在の性格診断と、未来の性格診断が統合されて「現在から未来の性格診断」(通常診断) に変更された。これにより結果表示のページの現在のページと未来のページを 1 つにまとめることになった。異なる形式へ移動するボタンは、それぞれの診断結果のページに移動する形式ではなく、形式選択のページに移動するボタンに変更となった。診断結果の根拠となるツイートを表示するボタンでは、Twitter 診断の機能から削除になったため、このボタンも削除することとなった。結果表示に関しては住所や趣味、総評の項目が削除され、新たにお勧め観光地や気の合う人が追加された。以下が変更後に表示する項目である。

- エゴグラムによる性格を表した画像形式のグラフ
- 名前 (Twitter 内で使用している ID)
- 性格
- お勧め観光地
- 気の合う人

これらの項目で通常診断と集団診断で共通しているのは、エゴグラムによる性格を表した画像形式のグラフ、名前、性格の項目である。通常診断ではこれらに加えてお勧め観光地を表示し、集団診断では気の合う人を表示するようになっている。

新たに追加された要素として挙げられるのは、エゴグラムによるグラフの推移を確認するスライダーバーと、エゴグラムの性格を表す 5 つの要素 (CP、NP、A、FC、AC) を説明した画像の配置である。まずエゴグラムの 5 つの要素を説明した画像では、「エゴグラム」のページでも使用した

画像を採用している。ホーム画面からエゴグラムのページにアクセスしないでユーザーが診断を行う場合も予想されることから、診断結果で表示されるエゴグラムのグラフの各要素が把握しにくいケースも考えられた。よりユーザーにグラフが意味する要素をわかりやすく伝えられるように、診断結果を表示する表の下にこの画像を配置した。

エゴグラムによる推移を確認できるスライダーバーは、通常診断の結果表示のページに実装した。このページではユーザーの現在の性格の診断結果と、未来の性格の診断結果の 2 つを表示している。このうち未来の性格診断において、Twitter 診断ではユーザーの未来の性格の推移を予想したエゴグラムの画像を 10 枚出力するようになっている。スライダーバーでは、これらの画像をユーザーが自由に見たい画像に切り替えられるようにすることで、未来の性格の推移をわかりやすく確認できることを目的とした。実装に至っては Javascript を使用し、10 個の目盛りを作成して、そのメモリの下に専用のカーソルを置くというものであった。しかしカーソルや目盛りをプログラム上で実装するのは困難であると判断したため、カーソル、目盛りは共に画像で実装を行った。また診断に使用するユーザーのツイート数が少ない場合は、出力する画像が 10 枚以下になる可能性がある。出力した画像の枚数に合わせて目盛りの個数も対応するために、必要になるパターンの数の目盛りの画像の作成も行った。

共有ツイートの機能は通常診断と集団診断の 2 つのページで実装した。ユーザーが共有ツイートのボタンを押すと、新たにウィンドウが開くようになっており、このウィンドウにあるツイートのボタンを押すと、診断結果を画像付きツイートとして投稿できるようになっている。初めに提案されていた方法は、新規ウィンドウは出現せずに共有ツイートのボタンを押すとそのまま診断結果をツイートできるものであったが、公式 Twitter の共有ツイート機能を参考にして、新規ウィンドウからツイートする形式になった。

（※文責: 小川聖司）

サーバー

ここでは分析班で使用したサーバーについて述べる。前期までは分析班で作成していた Web アプリケーションはローカルサーバーでしか稼働していなかった。後期からサーバーの選定・構築を行った。

まず、10 月にサーバーの選定を行いさくらのレンタルサーバー　スタンダードプラン（以下さくらのレンタルサーバと呼ぶ）の 2 週間お試し期間を契約することにした。契約するにあたってさくらのレンタルサーバーにネットなどの資料を見て申込に関する手続きや下調べをした。さくらのレンタルサーバでは php.ini の編集、PHP のバージョン選択、アクセスログの設定などを行った。アクセスログの解析を定期的に行い、エラーがでたら随時開発しているプロジェクトメンバーに報告し修正した。

さくらのレンタルサーバではファイルのアップロードに関して困難な場面があった。さくらのレンタルサーバには、オプションでファイルをアップロードできるファイルマネージャという機能がある。予定ではファイルマネージャで開発した Web アプリケーションでファイルをアップロードする予定だった。しかし、ファイルマネージャではフォルダごとアップロードすることができなかった。この問題を解決すべく、ファイルマネージャ上で擬似的なフォルダを作りそこにデータをアップロードする方法を試してみたが失敗してしまった。また、ファイルマネージャではアップロードできるファイルの容量に限界があり、作成している Web アプリケーションはその容量を超えておりデータをアップロードすることができなかった。そこでさくらのレンタルサーバのファ

イルマネージャを使用せず、FTP でファイルをアップロードすることを試みた。FTP のソフトは WinSCP を使用した。FTP でファイルをアップロードしたところ開発している Web アプリケーションのすべてのファイルをアップロードすることに成功した。

また、さくらのレンタルサーバーでは解決できない問題があった。アップロードした後それぞれのファイルを実行してみたところ、拡張子 PHP ファイルは実行できるが拡張子 exe ファイルが実行できないことが判明した。拡張子 exe ファイルをどうにか実行できないか調べてたところさくらのレンタルサーバ上では拡張子 exe ファイルが実行できない仕様だとわかった。拡張子 exe ファイルを実行できないと Web アプリケーションそのものが動作しないため、改めてサーバーの選定に入ることになった。

新たなサーバの選定の際には、開発に使用しているプログラミング言語のバージョン・拡張子の確認・ファイルの構造など含めて綿密に調整を行った。また、料金や運用期間などもサーバーを選ぶ際に考慮した。仮想サーバの他に物理サーバの購入も検討したが、学内でしかサーバの操作を行うことができないため購入は断念した。結果として新たなサーバには、さくらの VPS Window server for VPS 8GB プランを選択した。運用期間は 2015 年 11 月から 2015 年 3 月までを予定している。さくらのレンタルサーバーの時と同様、サーバーに関する申し込みマニュアルなどを作成した。また、VPS の接続にはリモートデスクトップを用いた。使用したのは Windows server 2012 R2 であり、そこに IIS8.5 をたてた。Windows server 2012 R2 のサーバーマネージャを起動させ、Web サーバー (IIS) をインストールした。次に IIS マネージャで ISAPI および CGI の制限の編集やハンドラーマッピングの設定を行った。そして IIS8.5 に PHP5.4、Python2.7.10、Java8 の導入を行った。この際に php.ini の編集も行った。次に Python のモジュールである Numpy、scipy、sklearn を導入した。

この段階で診断した結果のページが 500 エラーとなる現象が起こった。この 500 エラーは php.ini で session.save_path を編集することで 500 エラーは修正することができたが SSL certificate problem: unable to get local issuer certificate というエラーに置き換わってしまった。このエラーはサイトの資料 [14] を参考に curl 証明書をおくことで修正できた。しかし、今度は SSL certificate problem: unable to get local issuer certificate のエラーの画面が再度 HTTP エラー 500.0 - Internal Server Error に置き換わってしまう事態が起こった。500 エラーの詳細には「c:\php5\php-cgi.exe - FastCGI プロセスは、構成されている要求タイムアウトを超過しています」と記載されていた。そこで、php.ini 及び IIS の FastCGIsetting の値を何度か変更して実行してみたもののエラーは直らなかった。その他にも VPS の再起動・ISAPI および CGI の制限の設定・ログの確認・USERS 権限の変更・php.ini の編集などを行ったがエラーの原因解明は出来なかった。しかし開発しているプロジェクトメンバーと話し合いプログラムを書き換えたところ 500 エラーは解決することができた。

また、診断結果において 1 箇所正常に表示されない部分があった。詳しく調査していくと PHP の exec 関数で問題が起きているようだった。プログラムの動作が正常に動いた場合、PHP の exec 関数に配列の値が 50 個ほど渡され、その値を表示するといった内容のプログラムである。しかし、ブラウザでは PHP の exec 関数自体は動いているものの Array() と表示され、どうやら配列の値が空白で返ってきてるようだった。試しに PHP の exec 関数のみを用いたテストケースを VPS のコンソール上で試してみたところ、同じく Array() と表示される結果となった。まず、コンソール上で正しく実行できるよう試みたところ python のパスを編集することでコンソール上で実行できるようになった。しかし、ブラウザでは依然と Array() のままでプログラムは正常に動作しなかった。ネット上の資料を参考にハンドラーマッピングの設定、システム環境変数の設定、パーミッ

ションの設定、エラーログの確認、IIS_IUSRS の権限、USERS の権限、CGI および ISAPI の制限の編集、php.ini の編集、UAC のレベル下げなどを再度確認・設定し直したがブラウザのこの問題は解決することが出来なかった。最終的には、この部分のプログラムを修正したことによりエラーは解決し、最終発表会にはサーバーの構築を完了することができた。

開発ファイルの更新についてだが、プロジェクト学習の時やメールなどを使い分析班のメンバーと定期的に連絡をとり、随時更新されたファイルに置き換えていくことで常に最新のファイルをサーバーに置くように実現した。現状、セキュリティの面で多く不安が残る結果となっている。3月までサーバーを稼働する予定であるため、サーバーの定期メンテナンスをしていくなかで改善・修正を行っていく。

（※文責: 菅原春香）

4.8 最終成果発表のスライド・ポスター作成について

4.8.1 ポスター

最終発表のポスターでは、中間発表と同様に全体のポスター 1 枚・検索班のポスター 1 枚・分析班のポスター 1 枚と 3 枚作成し、そのうち分析班のポスターを作成した。最終発表でもポスター・スライドともに班ごとに色分けをし、分析班はオレンジ色に統一した。最終発表の準備のための作製は 1 ヶ月前に始め、分析班のスライド担当の方と話し合いや調整を行ったりなどした。

最終発表に用いるポスターを作成するにあたって、中間発表のポスターを再度見直し、ポスター閲覧者によりわかりやすく直観的に内容を理解していただくための工夫を重視した。中間発表のポスターでは文章による説明が多く難解な単語も多数あったために、中間発表の際には用語の意味やシステムについての多くの質問を受けた。そこで、専門的用語を図やイラストを用いることでポスター閲覧者に直観的な理解をしていただけるのではないかと考えた。最終発表のポスターでは中間発表の形式を参考にして背景・作成したサービスの 2 項目を作り、全体の文字の大きさを最小でも 24pt にし、情報量の多さよりも文字の読みやすさを優先して作成を進めた。

背景の項目の内容は中間発表とほとんど変わりがなかったが、性格診断アプリケーションを使っていただきたいツイッター利用者には、文字だけでの説明よりも一目見るだけでどんなものかがわかるように、アプリメーカー・ツイートプロファイリングそれぞれの診断結果画面を用いて、そのツイート診断アプリケーションの物足りなさをそれぞれ一言でまとめた。それらの物足りなさから中間発表での改善案に加えてツイッターをローカライズする項目も増やし、作成したサービスで述べた。

作成したサービスの項目では中間発表での文字の多さを改善し、図での表現を用いて直観的な理解をできるような工夫をした。サービスの名前の部分は、中間発表の時に加えて観光地を勧める部分を追加し、ローカライズをしていることをアプリケーションの名前でアピールした。さらに、2015 年 12 月現在ではツイートを自動的に解析してエゴグラムを用いて性格診断をするアプリケーションは前例がないことから、分析班が開発したアプリケーションが初めての機能となることもアピールした。メインの機能である「現在～未来診断」と「集団診断」の 2 つの診断を枠で囲みそれぞれの説明と結果表示画面を強調しているが、この 2 つの診断を行うまでの流れをその下の部分にて記載した。中間発表では言葉の説明が多かったため、矢印などの図を使うことによってよりわかりやすさを重視した。オレンジ色にて説明の流れが書かれているところは実際に診断者が操作する見える部分であり、その下の青色にて説明の流れが書かれているところは診断を行うときにアプリ

ケーション内部で実行されるシステムの流れである。診断者が診断の選択画面にて診断を選択すると、それぞれ異なった分析方法でアプリ内部の操作が行われ、ツイートの分析を行った後にそれぞれの診断結果が表示されるようになっている。アプリ内部の操作で行われる形態素解析、エゴグラムの特徴辞書で点数化、機械学習によるお勧め観光地診断に使われている画像は分析班のスライド班と共有しているものである。「現在～未来診断」では作成したエゴグラムから現在～未来の性格の推移を見れるということがわかるように矢印のある画面になり、「集団診断」では矢印はないが気の合う人を表示するという項目がある画面になり、それぞれの結果表示画面に違いがあることがわかるようにしている。最後には中間発表の時にもあったツイート共有機能もついている。以上が性格診断の一連の流れであるが、どこで「現在～未来診断」と「集団診断」の分析の違いが表れるかがこの図だけではわからないという問題点もあった。

中間発表では現在までの取り組みと展望を記載していたが、分析班が目標としていたサービスの作成が完了していたので、今回は記載しないこととした。図 4.9 が実際に最終発表で使用した分析班ポスターである。

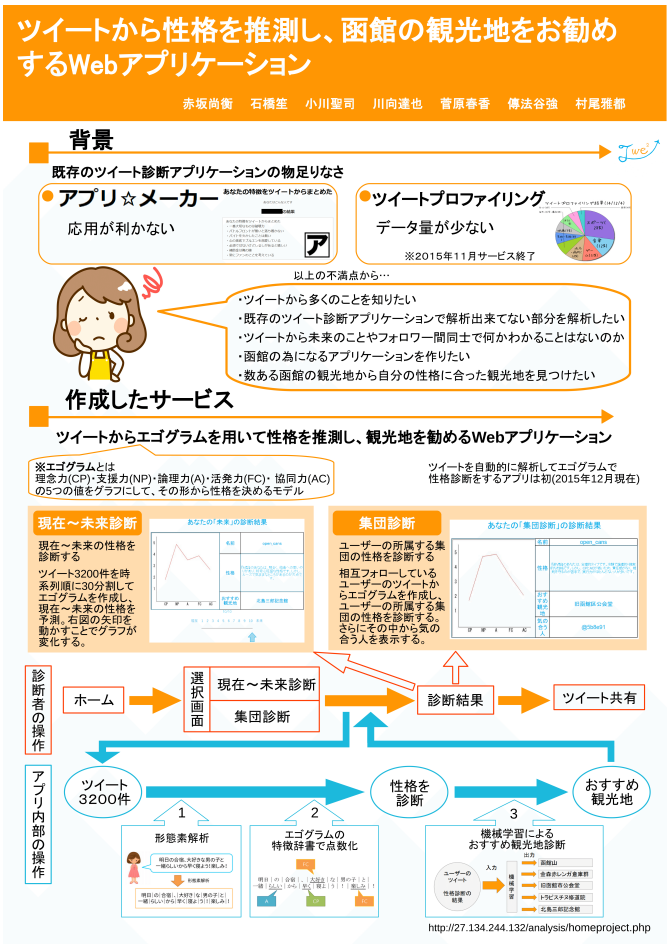

図 4.9　最終発表のポスター

（※文責: 傳法谷強）

4.8.2　スライド

最終成果発表で分析班が開発した Web アプリケーションの説明をするためのスライドを作成した。初めに作成したスライドでは、手法を説明するスライドを多く作っていた。しかし、手法が多

くどのようなものを作成したか分かりづらいという意見が出た。そのため、手法を説明するスライドを減らし、作成物を説明するスライドを増やし、作成したものがどのようなものか分かりやすくした（図 4.10、図 4.11）。また、実演を行うのであれば、実演ができなかった時のための、動画やスクリーンショットを用意しておくとよいと言われた。そのため、スライドに作成物のスクリーンショットを用意した。工夫した点は、「観光地を訪れるきっかけづくり」のスライドである（図 4.12）。ここでは、性格診断に興味を持っている人が、性格診断の結果と同時に函館のお勧め観光地を表示することで興味を持ってもらい、観光地に行くきっかけづくりになることを説明した。しかし、文章だけでは理解が困難なため、具体例を用いて説明し分かりやすくした。

最終成果発表を行った後、発表評価シートでは、ツイートからの性格診断はどうやっているのかや機械学習の部分がよく分からなかったなど、どのような手法で性格診断が行われているかよく分からないという意見が多く、手法を説明するスライドを少なくし過ぎ、わかりづらくなっていた。また、作成物の利用者の評価や性格診断の信憑性や機械学習の精度について説明してほしかったという意見があり、これらをスライドに組み込むべきだった。

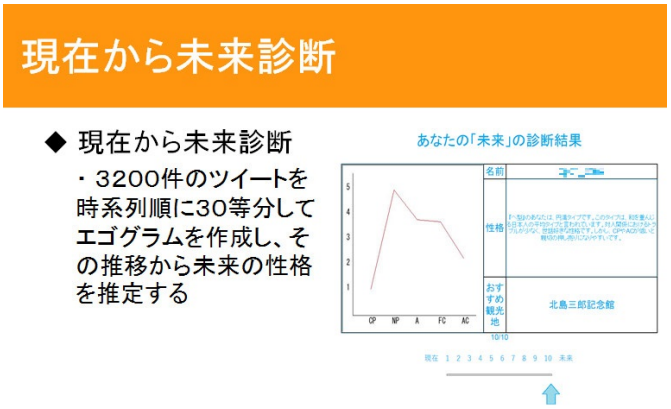

図 4.10　現在から未来診断の説明のスライド

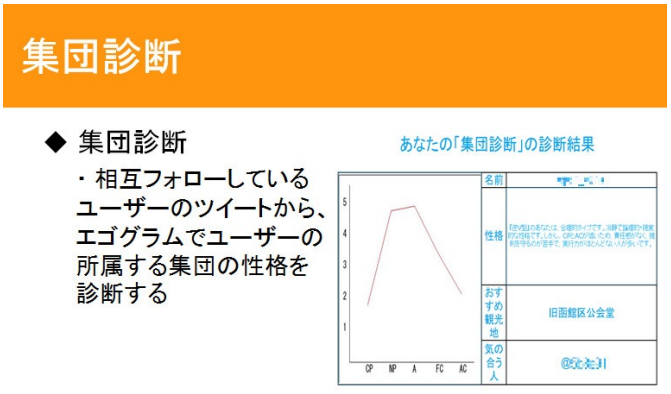

図 4.11　集団診断の説明のスライド

（※文責: 村尾雅都）

図 4.12　観光地を訪れるきっかけづくりのスライド

4.9　最終成果発表について

2015 年 12 月 11 日に行われた最終成果発表会では、作成したポスターの展示を行い、スライドを用いて説明をするため、プロジェクターとスクリーンを用意した。加えて、スライドの説明個所を分かりやすくするために指し棒を使用した。分析班の発表の流れは以下の通りである。

1. 既存のツイート診断アプリの紹介
2. 本グループの目的
3. 作成した Web アプリケーションの紹介
4. 実演
5. 性格診断の分析方法の説明
6. 観光地お勧め機能の説明
7. まとめ

実演の際は、Web アプリケーションの説明やページ遷移を時間をかけずに分かりやすく伝えるため、発表スライドを映し出すための PC と、Web アプリケーションを見せるための PC の 2 台を用いた。そして、実際に診断を行ってしまうと処理時間がかかってしまうため、結果画面をあらかじめ別のタブで開いておき、タブの移動で結果画面を次々と表示することで、スムーズにアプリケーションの説明を行うことができた。その他には、実際に Web アプリケーションを体験するための PC を用意した。

この発表では、聴衆者に発表評価シートを配り、発表技術と発表内容について評価をしていただいた。そこで寄せられた発表内容に対する意見と疑問は以下の通りである。

- エゴグラムと機械学習といったキーワードのもう少し丁寧な説明がほしい
- 観光地をおすすめする根拠がほしい
- 機械学習の判断基準が信用しにくい
- 結果画面が見にくい
- 観光地を出す必要性はあるのか
- 新しくできた観光スポットの情報がほしい
- 性格診断の信憑性が分からない

- なんで函館の観光地だけなのか
- ツイート量は診断時間にどれくらい関係してくるのか
- アプリ上でエゴグラムの要素として A や FC と出されても分からない
- 性格診断と観光地のつながりが分からない
- ツイートから性格診断するプロセスの詳しい説明がほしい
- おすすめの観光地はユーザーの興味ある地域のものを選んでほしい
- 現在から未来の診断ではなく、過去から現在ではないのか
- 現在から未来の推移とおすすめ観光地の関係が分からない
- 状況によって単語の意味が変わってくるものがあると思うが、そこは考慮しているのか
- 3200 件ツイートを溜めないと診断できないのか
- どういったワードが性格診断に大きく影響してくるのか詳しく知りたい
- ツイートと性格と観光にはどのようなつながりがあるのか分からない
- エゴグラムの正式な表記がない
- スライダーバーにした意味はあるのか

上記の意見や疑問を集約すると、以下の 2 点が多かった。

- 性格診断やお勧め観光地の結果の精度
- 性格診断とお勧め観光地の 2 つの機能の関係性

性格診断やお勧め観光地の結果の精度については、作成したアプリケーションについては前述したように発表会当日に体験できるよう PC を用意してはいたものの、一定の人数に利用させてみて、その結果についてアンケートで評価してもらうことはしていなかったため、性格診断の結果が本当に当てはまっているのかどうかの精度はまだ分かっていない。お勧め観光地の精度については、機械学習の過程で元となるデータとテストデータの掛け合わせで約 50% の正答率であったが、説明が長くなってしまうことで発表では話してはいなかった。

性格診断とお勧め観光地の 2 つの機能の関係性については、その機能の意味ついて関係は無いが、どちらもツイートから分析し、性格診断はエゴグラム、お勧め観光地には機械学習と、異なった指標を用いているため、ユーザーにはツイートの分析方法には様々あり、何を出力するのかも様々なバリエーションがあることを伝えたかった。

（※文責: 石橋笙）

第 5 章　本グループにおける各人の担当課題及び解決課題

5.1　赤坂尚衡

私は、作成物の以下の機能の作成を担当した。

- 診断に共通する機能
- 現在から未来診断機能の一部
- 集団診断機能の一部
- お勧めの観光地を推定する機能

集団診断と現在から未来診断の共通する機能のうち、以下の機能を実装した。

- ユーザーがログインする機能
- ユーザーのツイートを取得する機能
- 取得したツイートを形態素解析する機能
- エゴグラムを画像として出力する機能
- 診断結果画面を画像として出力する機能
- エゴグラムの特徴辞書を用いてツイートからエゴグラム作成に使用する値を算出する機能

ログイン機能は TwitterAPI を使用するためのライブラリである TwistOAuth にサンプルコードが載っていたので、それを参考にして作成した。サンプルコードのログイン機能のままの状態では、他の Web ページに遷移したときにログイン情報が失われてしまった。そのため、サーバー上にログイン情報を保存して遷移先の Web ページからサーバーに保存したログイン情報を取得する方法を取った。　エゴグラムを画像として出力する機能と診断結果画面を画像として出力する機能を Processing で作成したが、サーバーで使用できないことが判明し、Java で書き直した。また、ツイートを取得する機能では取得したツイートをサーバーに保存しており、2 回目以降の診断の処理を軽くするために、取得したツイートがサーバーに残っていれば再び取得しないようにした。以下の性格診断においては共通する機能以外に担当した部分を示す。

現在から未来診断では以下の機能を実装した。

- 現在の性格から未来の性格へと推移するスライダーバーの実装
- 複数枚の画像をツイートする機能の実装

画像を 1 枚だけツイートするプログラムは API 班の石橋が作成したが、現在から未来診断を作成するに当たり、画像を複数枚共有する必要があったため作成した。スライダーバーでは画像が 1〜10 枚までに対応するようにしてエラーに柔軟性のある設計にした。

集団診断では以下の機能を実装した。

- 相互にフォローしているユーザーの ID リストを取得する機能
- 相互にフォローしているユーザーのツイートを取得する機能

- 相互にフォローしているユーザーの名前取得

集団の機能を診断する方法では相互にフォローしているユーザーが多いと、ツイートを取る処理が重くなってしまうため、相互にフォローしているユーザーが 15 人以上いた場合はツイートを取得するユーザーをランダムに 15 人選ぶ方法をとった。また、TwitterAPI の「一定時間内に既定の回数以上使用できない」という利用制限に抵触しないため、API でユーザー名を取得した後、名前をグルーバル変数に保存するなどして API の使用回数をなるべく減らすようにした。

お勧めの観光地を推定する機能については以下のことを担当した。

- トレーニングデータからツイートの名詞の集合を取り出し特徴語辞書にするプログラムの作成
- ランダムフォレストで作成した特徴の重要度を用いて特徴語辞書を削減するプログラムの作成
- 任意のユーザーのツイートを出現頻度を特徴量とした特徴ベクトルにするプログラムの作成
- 任意のユーザーのツイートを TFIDF 値を特徴量とした特徴ベクトルにするプログラムの作成
- 主成分分析で特徴ベクトルの次元を削減するプログラムの作成
- ランダム行列をかけて特徴ベクトルの次元を削減するプログラムの作成
- サポートベクトルマを R で作成
- ランダムフォレストを R、python、Java で作成
- 線形判別機を R、python で作成
- 学習した結果をモデルとして保存するプログラムの作成
- 保存したモデルを使用して診断結果を出力するプログラムの作成
- 特徴ベクトルの次元削減
- 機械学習のモデルの決定
- 機械学習に使用する特徴ベクトルの選定
- 機械学習の識別精度の向上
- 未知データの取得

特徴語辞書にするプログラム、特徴語辞書を削減するプログラム、特徴ベクトルを作成するプログラム、特徴ベクトルの次元を削減するプログラム、機械学習のモデルのプログラムは使用方法を記載して共同作業者全員に配布し、全員が機械学習の正解率を確認できるようにした。

他に行ったこととして以下のことが挙げられる。

- 形態素解析に用いるライブラリの選定
- TwitterAPI を使用する際に用いるライブラリの選定
- php ファイルから他の php ファイルに値を渡す方法の確立
- 作成した関数のライブラリ化
- API 班の作成したプログラムの単体テスト
- API 班、Web 班、エゴグラム班のプログラムの連結テスト

形態素解析においては mecab が知られているが、分析班では導入が用意であり、サーバーにインストールが不要な kuromoji という形態素解析ライブラリを使用することにした。TwitterAPI ライブラリは TwistOAuth を使うことにした。Twitter4j や TwitterOAuth が使用する候補として挙げられたが、Twitter4j は Java 用のライブラリであり Web アプリケーションとしては不適格

だと判断し、TwitterOAuth は php 用のライブラリであったが改良版の TwistOAuth が存在したため使用しなかった。また、他の開発者が見やすいように作成した関数群を別のファイルにまとめることやリファクタリング、コメントの追加なども随時行った。

（※文責: 赤坂尚衡）

5.2 石橋笙

私が API 班として前期に取り組んだ課題は 2 つある。それは以下の通りである。

- 診断結果を共有ツイートするための機能実装
- Web 班の作成した HTML ページとアプリケーションの連携作業

共有ツイートの機能実装では、はじめは Twitter が提供している Web ページに貼り付けるタイプの共有ボタンを取り入れようと考えたが、実際に使用してみると画像を付けてツイートすることができないことが分かった。この Web アプリケーションの診断結果の表は画像として出力されるため、この方法とは違う方法をとることにした。そこで検索した結果、TwitterAPI を利用して画像を付きのツイートを行えるサンプルプログラムを見つけたので利用して組み込んだ。

HTML ページとアプリケーションの連携作業では、PHP を書いている eclipse 内に HTML を組み込むことができるため、ページ遷移に気を付けて実装を行った。

次に、後期に取り組んだ課題は 2 つある。それは以下の通りである。

- 集団診断の診断処理の実装
- 集団診断の中に存在する気の合う人の診断処理の実装

集団診断では、TwitterAPI を用いてユーザーの相互フォロワーの情報を扱うのだが、まず API では、相互フォロワーの TwitterID を直接取得することはできない。よって、まずユーザーがフォローをしている人の ID を抜き出し、次にユーザーをフォローしている人の ID を抜き出し、それら 2 つを参照してどちらにも存在している ID を取得することで相互フォロワーの ID 取得を実装した。相互フォロワーの人数は利用するユーザーによって様々であり、人数が多ければ多いほど処理に時間がかかってくる。よって、相互フォロワーの人数に応じて可変長に人数を選出する考えになったが、実装において困難だったため、固定の人数を選出することにした。その結果、多すぎずも少なすぎずもない人数として 15 人の相互フォロワーをランダムに選出することにした。加えて、各フォロワーから 3200 件のツイートを持ってくるとなると膨大なテキストデータとなってしまうので、各フォロワーの最新の 100 件のツイートを用いた。

気の合う人の診断では、初めに 1 人目の相互フォロワーについてツイート情報を PHP から Processing に渡し、点数化を行ってから PHP に返してユーザー本人の要素ごとの点数と比較を行い、2 人目に移って繰り返し処理していく工程で実装したが、PHP と Processing を何回も行き来することになり、処理が途中で停止してしまった。そこで、行き来する回数を減らすために、渡す情報を一気にまとめて渡し、受取先でひとりひとりの情報に区切って繰り返し処理する方法にすることにした。ここでは、取得したフォロワーのツイートはそれぞれテキストファイルに格納して保存しているため、Processing の性格診断を行うための exe ファイルが対象のテキストファイルを

正しく読み込むことができるように、ディレクトリの場所やパスの指定に注意をして実装を行った。この方法を取るにあたって、集団診断で使っていた関数をそのまま使ってしまうと、相互フォロワーから取得したツイート群をひとつのテキストファイルにまとめてしまうことや、先ほど記述した PHP と Processing の行き来の際にデータをまとめたり区切る処理が必要なので、気の合う人を診断するための専用の新しい関数を PHP と Processing の両方に作成した。この方法で実装すると、PHP と Processing を往復する回数は 1 回だけになり、処理を最後まで終えることができた。

この 2 つの課題に取り組んだ結果として、ランダムに一定数選出した相互フォロワー全体の性格診断と、その相互フォロワーの中から一番気の合う人が結果の画面に表示され、機能の実装ができた。しかし、気の合う人の診断結果は相互フォロワーをランダムに 15 人取得しているため、相互フォロワーの人数が多いユーザーにとっては診断後またすぐに診断を行うと直前の結果とは異なった気の合う人が表示される場合がある。

その他には、最終成果発表において発表者として活動した。発表用のスライドを基に作成された原稿をメンバーに向けて読み上げ、そこで得られた意見から原稿を繰り返し改稿した。そこでは GoogleDrive 等を活用して、もうひとりの発表者の方と相談しながら聞いている人たちに伝わりやすく、まとまった文章を作ることを意識した。また、分析班の発表では実際の Web アプリケーションの動きを見せるための実演も行った。スムーズな画面遷移の仕方を心がけるため、検索班の発表者の方に手伝ってもらい、診断までの操作方法を工夫して説明することが出来た。質疑応答の際には、回答が曖昧になってしまった部分もあったが、どうにか分かりやすく伝えることを心がけた。

（※文責: 石橋笙）

5.3　小川聖司

私が主に行った担当課題は分析班で作成した Web アプリケーション (Twitter 診断) を表示する Web ページの実装であった。ページの構成はホームページ、診断形式選択、診断結果表示の 3 つからなり、ページ全体のデザインは公式 Twitter を連想できるものにする結論に至った。私はこれらのページの作成や、分析班で作成するアプリケーションの機能と Web ファイルの連携・調整を担当した。Web ページの作成には HTML、CSS、Javascipt の 3 つを採用し、共有ツイートの機能の実装には Javascipt、ページの大まかな内容は HTML、CSS で行った。

ホームページではユーザーがまず最初にアクセスするページであるため、ユーザーに Twitter 診断で行える診断の内容の説明や、診断時に Twitter の ID とパスワードを使用することを伝える必要があった。Twitter 診断ではエゴグラムや形態素解析を行ってユーザーの性格やお勧めの観光地を推測することから、エゴグラムや形態素解析についての解説と、Twitter 診断の診断内容を説明しなければならなかった。これらよりホームページの画面では診断開始用のボタンの他に Twitter 診断、エゴグラム、2 つの診断の 3 つの項目のボタンを作成することで、それぞれの説明用のページにアクセスできるようにした。Twitter 診断の項目からはこのアプリケーションがどのような診断になっているかの説明と、形態素解析について解説しているページに飛び、エゴグラムの項目からはエゴグラムとは何かについての詳細ページに飛び、2 つの診断の項目からは Twitter 診断で行う現在から未来の診断と、集団診断の 2 つの診断を紹介しているページに飛べるように実装した。

診断方法の形式選択を行うページでは、Twitter 診断で行う現在から未来の診断と、集団診断の 2 つの形式を選択するためのページであった。そのため現在から未来の診断を通常診断とし、通常診断と集団診断の 2 つのボタンとそれぞれの診断の説明文を合わせて作成した。また形式選択のページからもホームページに飛べるように、ホーム画面へ戻るボタンを作成した。

診断結果を表示するページにおいては通常診断と集団診断の 2 つの形式があることから、それぞれの診断形式に合わせた結果表示のページが必要であった。

まず通常診断のページではユーザーのツイートから判定された現在の診断と未来の診断の 2 つの結果を、エゴグラムの性格推移のグラフと合わせて表示しなけてばならなかった。このため現在と未来のそれぞれの結果を、表でまとめた形で表示されるように実装した。加えて未来の診断結果ではユーザーの未来の性格を推測したグラフが 10 枚あるため、未来の診断結果の表の下にスライダーバーを配置した。スライダーバーにあるカーソルを移動すると表にあるグラフが次々に切り替わるので、性格の推移が確認できるようになっている。またグラフには AC、CP といった項目が使用されているため、それぞれの項目の意味するものを大まかに説明した画像を作成した。これらの診断結果を共有ツイートするためのボタンの他に、ホーム画面へ戻るボタンと選択形式のページへ戻るボタンを作成した。

集団診断のページで表示するのは、ユーザーのフォロワーのデータより、そのユーザーの所属している集団の性格診断結果と、性格の傾向が類似しているユーザーであった。このうち後者を気の合う人として、表の 1 項目として表示できるように組み込んだ。通常診断同様、診断結果は表形式で表示するため、エゴグラムの項目を説明した画像、共有ツイート用のボタン、ホーム画面へ戻るボタンと選択形式のページへ戻るボタンの実装を行った。

（※文責: 小川聖司）

5.4　川向達也

前期の主な担当課題は、エゴグラムを用いた性格診断を行うための特徴辞書の作成と R 言語を使って性格を分析することだった。特徴辞書に関しては、現在登録されている 500 語のうち手作業で作成し 250 語程度を私が担当した。R 言語に関しては、形態素解析されたものを API 班から提供してもらい、実際に特徴語を抜き出しテキストに出力することはできたが、処理速度が遅いものしかつくれなかったため、その部分を API 班に処理を依頼した。

夏休みは開発中の性格診断以外に、一捻り案を考えてくることと、性格診断の作成を進めるという課題がでた。そこで私は、機械学習を用いて函館に関連するものを作りたいと考え、ユーザーに函館の観光地をお勧めする機能の作成を提案した。そして、実際にこの機能を作成することになった。さらに、未来の性格を回帰分析を用いて推測する機能を R 言語で実装した。しかし、exe ファイルにする方法が見つからなかったために、このプログラムを組み込むことはできなかった。

後期の主な担当課題は、現在から未来診断と観光地をお勧めする機能の作成、分析班の最終成果発表とその準備だった。 現在から未来診断に関しては python で回帰分析を用いて作成した。しかし、実装する際に数値計算ライブラリを用いたものを exe ファイルにできなかったので、数値計算ライブラリを用いることなく実装した。具体的には、numpy がうまく動かなかったので、最小二乗法の数式を入力して実装した。機能としては、PHP から 30 等分されたエゴグラムのデータを受け取り、回帰分析によって未来の性格を推測し、その結果を返すというものである。さらに、追加の機能として、性格の推移をみられるようにした。これは、推測して得られたデータを加えても

う一度回帰分析を行うことでさらに未来の性格を推測できるというものだ。実際のアプリケーションでは、Web 班と API 班が作成したスライダーバーを動かすことで時間の経過とともにどのように変化しているのかが見られる。実際に 30 等分したデータをプロットして確認したときに、ランダムウォークや株の時系列変化のような形に似ていると感じたこともあり、時系列解析を取り入れればより良い予測ができるのではないかと考えたが、処理をすべて自動化できなかったので、そこまで作成することはできなかった。そのため、この機能は最終的に回帰分析を用いて実装した。

観光地をお勧めする機能に関しては、実際に観光地に行った人のツイートの収集や、python でサポートベクターマシンを実装し、どの程度正しく分類できているのかテストを行った。実際に北島三郎記念館と函館山で 2 値分類を行った結果、正答率が 49% とランダムと変わらない結果になった。次に 10 個に分類することをやったが正答率は最高で 15% 程度だった。サポートベクターマシンのパラメータを変更してテストすると正答率が 4% 程度上がることもあったが、最適なパラメータがわからなかった。その後、API 班が実装したランダムフォレストが最も正答率が高かったためそちらが実装された。その後はランダムフォレストを用いて 4 値分類や 5 値分類のテストを行った。

最終成果発表の準備については、スライド班として活動した。まず原稿を作成しそれをもとにスライドを作成した。初期のスライドは、分析方法の説明ばかりでつまらない、わかりにくい点が多い、機械学習の説明についての間違いなど多くの指摘を受けた。そこで、他のメンバーとともに、画像を使ってなるべくわかりやすくしたり、分析の手法を入れすぎないことなどを意識して改良を重ねた。また、実演ができない場合の対応や、実演の流れについても検索班と打ち合わせを重ね、スムーズな発表を心がけた。分析班の発表時間が長かったので、時間を計測しての練習も重ね、目標である 6 分 30 秒を確実に切れるようにした。最終成果発表当日は、後半 3 回の分析班の部分を発表した。

（※文責: 川向達也）

5.5　菅原春香

前期はホーム画面の Web ページのデザインと中間発表のスライド、後期はサーバーの選定・構築を担当した。

まず、ホーム画面の Web ページのデザインを説明する。私はホーム画面のページを担当した。ホーム画面のデザインは、Twitter をモチーフにした青色を基調とするプレーンなデザインのページを作成した。ページを作成するにあたって Web 班はじめプロジェクトメンバーの方にアドバイスを多数いただいた。

次に、中間発表のスライド発表について説明する。中間発表のスライドの作成及び発表を行った。スライドの作成に関して検索班の発表の方にも手伝っていただき、非常に伝わりやすいスライドを作成することができた。スライドの発表では、プロジェクトメンバーや先生方に何度か練習をみてもらいアドバイスを頂いた。結果として、中間発表の評価者フィードバックで、スライド発表の声が大きくてよかった・聞き取りやすかった・スライわかりやすかったといった評価を頂くことができた。

最後に後期に担当したサーバーの選定・構築について説明する。そこで、まずサーバーの選定を行った。10 月にさくらのレンタルサーバー　スタンダードプランの 2 週間お試し期間を契約することにした。契約するにあたって、さくらのレンタルサーバーに申込に関する手続きや下調べなど

をした。さくらのレンタルサーバでは php.ini の編集、PHP のバージョン選択、アクセスログの設定・確認などを行った。しかし、さくらのレンタルサーバーでは一部表示できない部分があり、改めてサーバーの選定に入ることになった。新たなサーバの選定の際には、開発に使用しているプログラミング言語のバージョン・拡張子の確認・ファイルの構造などを調べて綿密に調整を行った。結果として新たなサーバには、さくらの VPS Window server for VPS 8GB プランを選択した。運用期間は 2015 年 11 月から 2015 年 3 月までを予定している。さくらのレンタルサーバーの時と同様、サーバーに関する申し込みマニュアルなどを作成した。使用したのは Windows server 2012 R2 であり、そこに IIS8.5 をたてた。次に IIS マネージャで ISAPI および CGI の制限の編集やハンドラーマッピングの設定を行った。そして IIS8.5 に PHP5.4、Python2.7.10、Java8 の導入を行いどちらも成功した。そして python のモジュールである Numpy、scipy、sklearn を導入した。開発ファイルの更新についてだが、プロジェクト学習の時やメールなどを使い開発している班員と調整しながら随時更新されたファイルに置き換えていくことで常に最新のファイルをサーバーに置くように実現した。現状、セキュリティの面で多く不安が残る結果となっている。3 月までサーバーを稼働する予定であるためサーバーの定期メンテナンスを行っていくなかで改善・修正を行っていく。

（※文責: 菅原春香）

5.6　傳法谷強

私はプロジェクト内で分析班の Web 班、中間発表と最終発表のポスター作成を主に担当した。

Web 班の担当課題として、Web ページ作成、ページレイアウト、デザインなどといった、目に見える部分を主に担当した。ページを作るうえで HTML、CSS を使用した。初めにページ遷移図を作っていき、そこからページの作成を行っていった。その際、同じ Web 班の人や分析班の人と話し合いながら、どこにどんな機能を置いていくかの話し合いと調整を何度も行った。その結果、中間発表までにはページが遷移する最低限の Web ページの機能と、ページ全体を Twitter らしい外見に実装をするところまで完成した。そこから最終発表までは、実際にサーバーに公開するためのページ作成作業のため、既に存在する診断サイトに限らない様々な Web ページを閲覧し、分かりやすく面白い性格診断ページの作成のためにはどうするかを考えた。最終発表までの開発の途中でポスター作業に移ったが、それまではホーム画面・2 つの診断のページを主に作成した。ツイートとエゴグラムを用いての性格診断を、誰でもわかりやすく使えるようにと心掛けて作業した。しかし、機能面での説明が不足していたり、説明の手順があまりよくなかったという私自身のレビューと、ページについての閲覧者からのレビューがあまりなかったことで、分析班が目指していたわかりやすく面白いページになっているかどうかがわからなかった。

中間発表の担当課題ではポスター作成を行った。中間発表では 3 枚のポスターがあり、そのうち分析班のポスター 1 枚を作成した。ポスター作成は中間発表の 1 ヶ月前から行い、提案するサービスをまとめた要件定義書をもとにポスターを作り始めた。成果発表のポスターを作るのはこれが初めてで、周りの人の意見や学会のポスター等を参考にしながら、両班で話し合い同じような構成になるようにポスターを作成していった。同じ分析班内でも開発中のサービスについて理解することが難しく、ポスタースライド作りはうまく進められなかった。中間発表ではポスター閲覧者にわかりやすくするということを意識して作業に取り組み、できるだけ図やイラスト、実際に Web で表示される結果表示画面を使っていく方針でいた。しかし、中間発表に出来上がったポスターは文字

が多くなってしまい、私自身でもポスターの内容をしっかりと読み込まないと分かりにくい内容になってしまった。その中でも、文字サイズを全体的に大きめにして読みやすさを重視したり、専門的な言葉が少なくとも存在するのでしっかりと説明書きを書くといった、少しでも読み手に伝わるような努力をした。

最終発表の担当課題でも引き続きポスター作成を行った。最終発表の 1 ヶ月前から取り掛かり、分析班のポスター 1 枚を担当した。中間発表では文章による説明が多く、開発しているサービスに対しての疑問を抱く人が多かった。そのため、わかりやすさを追求してデザイン関連の本を読み込んだりデザインの学生にアドバイスを頂いたりして、伝えたい情報を一目でいかにわかりやすく伝えるかを重視した。中間発表の背景ではほとんどが文字の部分であったが、班内の全員の人がツイート診断アプリケーションの画像を見れば想像できるということで、その診断サイトの代表的な画像をポスターの背景の部分に記載した。作成したサービスの部分では、必要最低限の文章に抑え、できるだけ図や画像を使った説明になるような工夫をした。2015 年 12 月現在では、ツイートを自動的に解析してエゴグラムを用いて性格診断をするアプリケーションは調べた限りでは存在しないため、分析班が開発したアプリケーションが初めての機能となることは文で説明した。後になりこの部分をもっと強調していくべきだと感じた。メインとなる 2 つの診断の機能を色付けした枠で囲み、結果画面表示をすることで閲覧者に作成した性格診断アプリケーションをイメージして頂き、興味を持って頂けるような工夫をした。その診断結果表示をするために、診断者の操作 (フロントエンド) とアプリ内部の操作 (バックエンド) を分けて書くことでシステムの流れを理解していただけるようにした。これにより、診断者は最大 4 つのステップを踏むだけで性格診断からツイート共有までできるということをわかりやすくした。アプリ内部の操作は自動的に行われるのだが、最終発表において、そのことが書かれていないこと、性格診断を選択してから診断結果が出るまでに要する時間などを記入していない点が指摘された。さらに、機械学習によるお勧め観光地診断の説明があいまいで、質問にうまく答えれないような書き方をしてしまったので、改善する必要があると感じた。

開発した性格診断 Web アプリケーションはパソコン向けのアプリケーションであるので、QR コードを作成してポスターの右下に記載するかどうかの議論も分析班ないしプロジェクト内であったが、サーバーの問題や本来は Android に対応していない点から考えて、QR コードは載せずに URL のみの記入とした。

（※文責: 傳法谷強）

5.7　村尾雅都

前期の行った課題は形態素解析されたツイートから性格診断を行うために必要な特徴辞書の作成とエゴグラムについてである。特徴辞書は手作業で作成を行った。現在、特徴辞書は約 500 語程度になっている。エゴグラムは基本的なパターンについて学び、そのパターン別の性格の特徴をまとめた文章を、中間発表のスライドで使用するエゴグラムの例として API 班に提供した。

後期に行った課題は、エゴグラムについてと機械学習とスライドである。エゴグラムは前期の続きとしてエゴグラムのパターン別の性格の特徴をまとめた文章の作成に加えて、エゴグラムのパターン分けのプログラムの作成を行った。また、各項目の実数値 0〜5 までを組み合わせた時のグラフと結果として表示される性格の特徴の文章が一致しているかの確認を行い、一致していない場合は、該当するプログラムを追加、修正を行った。現在、エゴグラムのパターン数は 12 種類となっ

ており、今後もパターン数を増やす予定となっている。機械学習はユーザーに函館の観光地をお勧めする機能を作成するために機械学習について学び、機械学習で使う実際に函館の観光地に行った人の Twitter アカウントを正解データと未知データとして収集した。また、函館の観光地についてを紹介する文章の作成も行った。スライドは最終成果発表で使ったスライドの作成を行った。

（※文責: 村尾雅都）

第 6 章　成果物の現状

分析班の成果物としては、ツイートから性格を推測し、同時に函館の観光地をお勧めする Web アプリケーションを作成した。このアプリケーションにおいてできることは、以下の通りである。

- 現在から未来の性格の推移を推測して診断する
- 相互フォロワー全体の性格を診断する
- 相互フォロワーの中から最も気の合う人を診断する
- 機械学習を用いて函館の観光地をお勧めする

この Web アプリケーションの動きの流れを説明すると、作成した URL を入力することでアプリケーションのホームページに移動する。ホーム画面（図 6.1）では、診断に関わる 3 つの説明を見るためのリンクがある。まず、1 つ目は Twitter を利用した性格診断の基本的な診断手法の説明を見ることができる。2 つ目は、性格を診断するために利用したエゴグラムの詳しい説明を見ることができる。最後に、この Web アプリケーションにおいて独自の性格診断として取り入れた「現在から未来診断」と「集団診断」の概要と、どのような結果が得られるのかの例を見ることができる。ホーム画面にある「診断する」のボタンを押すと、アプリケーションと Twitter の連携のため、ID とパスワードの入力画面に移り、ユーザー自身の Twitter の ID とパスワードを入力する必要がある。入力を終えると診断タイプの選択画面へ移動する（図 6.2）。

図 6.1　ホーム画面

図 6.2　診断選択画面

ここでは、通常診断と集団診断が選択できる。通常診断を選択すると、現在の性格診断の結果と現在から未来の性格の推移を表した診断結果の 2 つが表示される（図 6.3）。現在から未来診断の結果画面の表示箇所の下にはスライダーバーがあり、動かすことで現在から未来への性格の移り変わりを 10 段階で見ることができる。次に、診断タイプの選択画面で集団診断を押すと、相互フォロワー全体の性格の診断結果が表示される（図 6.4）。この診断では、相互フォロワーとユーザーに気の合う人の TwitterID も表示される。そして、通常診断と集団診断の結果にはユーザーにお勧

めの観光地が表示され、同時に結果の表を画像として Twitter に共有ツイートすることが可能である。これにより、他の診断を行ったユーザーの結果を見ることができ、例えば仲の良い相互フォロワー同士でお互いの性格についてレビューを行い、この部分は合っているなどの話題作りになると考える。

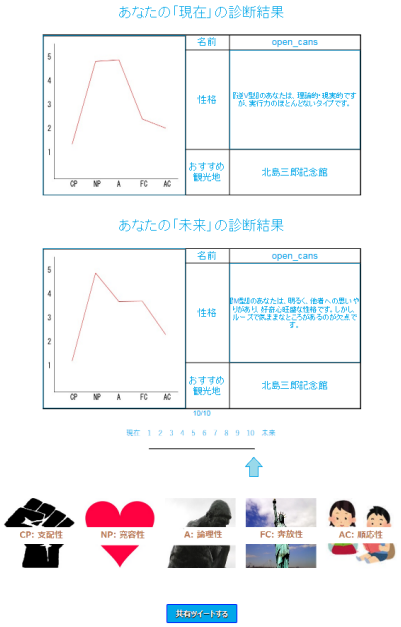

図 6.3　通常診断の結果画面

図 6.4　集団診断の結果画面

動作確認済みのブラウザは Firefox42.0 と IE11 となっている。このアプリケーションのサーバーは Windows server を使用しており、サーバーの稼働期間の関係で、2015 年 3 月までの運用となっている。

（※文責: 石橋笙）

第 7 章　本グループの展望

本グループの展望としては、まず 2016 年 2 月に秋葉原において課外成果発表会があるため、そこに向けて Web アプリケーションを改良していくことを考えている。具体的な改良点は以下の 4 点である。

- 観光地をお勧めする機能の正答率の向上
- 集団診断における相互フォロワー取得の調整
- 対応するブラウザの調整
- ウェブでも使いやすくスマホでも準拠するような画面の設定

観光地をお勧めする機能の正答率の向上では、機械学習による決定木の改良を行い、テストデータを掛け合わせた際の正答率がより向上させる必要がある。

集団診断における相互フォロワー取得の調整とは、現在は 15 人という固定の人数を取得しているが、相互フォロワーの総人数に応じて取得するフォロワー数を可変長に対応させることである。そしてある程度多い数でも処理時間に影響しないようにしたい。

対応するブラウザの調整とは、現在は正常に動作するブラウザが Firefox42.0 と InternetExplorer11 となっている。Chrome で動作させようとすると文字化けが発生してしまい、アプリ内の説明が読み取れないため、Chrome でも正常に動作するよう調整をしている。

現在のアプリケーションは Web 上での動作を想定しているが、スマートフォンでも動作する方がもっと手軽に体験してもらえると考える。現在は URL をスマートフォンのブラウザで入力すると画面が小さくなってしまい、文字が読み取りづらいので、利用する機器に対応した画面の大きさで表示することが必要である。

（※文責: 石橋笙）

第 8 章　まとめ

前期では、自然言語処理を用いてツイートから個人の特徴を推測する Web アプリケーションの作成を行った。そのシステムを実現するために、Web 班、API 班、エゴグラム班の 3 つに分かれて必要な機能の作成を行った。Web 班は、大まかな Web デザインを決定し、診断結果を表示するページやユーザーにツイートを取得することの許可を求めるページを作成した。API 班は、ユーザー認証や性格診断の結果のサンプル画像を共有ツイートする機能を完成させた。また、ユーザーのツイートを取得し、ツイートをエゴグラムで扱えるようにデータを形態素解析した。エゴグラム班は、形態素解析されたテキストデータから性格推定に用いる特徴語を抜き出す辞書を作成した。中間発表では、これらの内容についてや後期への展望について発表した。中間発表後の反省会では、プロジェクト始動後から現時点に至るまでの良かった点と悪かった点を話し合った。良かった点としては以下の 2 点が挙げられ、これらに関しては、今後も積極的に継続していくことが必要だと感じた。

- 作業が早い点
- グループメンバー同士のサポートを行っていた点

悪かった点として以下の 2 点が挙げられ、これらを改善することが今後の課題となった。

- スケジューリングが行えていなかった点
- 地方のためのローカライズから離れている点

後期では、自然言語処理を用いてツイートから個人の特徴を推測する Web アプリケーションの作成に加えて、機械学習を用いてユーザーに函館の観光地をお勧めする機能の作成を行った。そのシステムを実現するために、Web 班、API+ エゴグラム班の 2 つに分かれて必要な機能の作成を行った。Web 班は、Web デザインを決定し、ホームページの作成とサーバーの選定・構築を行った。API+ エゴグラム班は現在から未来診断の作成、集団診断の作成、ユーザーに函館の観光地をお勧めする機能の作成、集団診断で気の合う人を表示させる機能の作成を行った。最終成果発表ではこれらの内容をまとめ発表した。最終成果発表までに分析班が最終的な目標にしてきた新しい性格診断の Web アプリケーションの作成を完了することができた。しかし、性格診断や機械学習の精度向上、エゴグラムのパターン数の増加、サーバーの問題など改善する点は多く存在している。

（※文責: 村尾雅都）

参考文献

[1] ウェブサービス研究会, “アプリメーカー”, http://appli-maker.jp/ (2016/1/8 参照).

[2] ktty1220,“ツイートプロファイリング”, http://tweet-profiling.ktty1220.me/ (2016/1/8 参照).

[3] 大向　一輝, 情報処理, vol.47, no.8, pp.993-1000, September. 2006.

[4] IT 用語辞典, “SNS 【 Social Networking Service 】 ソーシャルネットワーキングサービス”, http://e-words.jp/w/SNS.html (2015/12/18 参照).

[5] “若者における SNS 利用行動およびリスク認知の検討 —LINE と Twitter を中心に—”, 苅野 正美, 2014.

[6] nielsen, “スマートフォン利用者の 92% が SNS を利用 ～ ニールセン, SNS の最新利用動向を発表 ～”, http://www.netratings.co.jp/news_release/2015/01/Newsrelease20150127.html (2016/7/24 参照).

[7] Twitter, Inc., “Twitter Reports Third Quarter 2014 Results”, https://investor.twitterinc.com/releasedetail.cfm?ReleaseID=878170 (2015/7/24 参照).

[8] “Twitter を情報源とした発話ロボットシステムの開発 Development of Speech Function for Communication Robot using Twitter ”, 藤原 裕樹, 山下 晃弘, 2014.

[9] mpyw,“TwistOAuth/README_EXAMPLES.md”,https://github.com/mpyw/TwistOAuth/blob/master/README_EXAMPLES.md,(2015/7/19 参照).

[10] @sfchaos, “統計解析言語 R における大規模データ管理のための boost.interprocess の活用”,http://www.slideshare.net/sfchaos/rboostinterprocess (2015/11/8 参照).

[11] エゴグラム 243 パターン全解説, 講談社, 1995, (ブルーバックス ; B-1063 . 自分がわかる心理テスト ; part 2).

[12] CiNii Articles,“ブログからの性格推定手法に関する一考察”, http://ci.nii.ac.jp/naid/110008123560 (2015/7/22 参照).

[13] R 言語逆引きハンドブック石田基広著. シーアンドアール研究所, 2012.

[14] Various Program Blog, “windows コマンド「cURL」のダウンロードと設定”,http://piji.daiwa-hotcom.com/wordpress/?p=1005 (2016/1/20 参照).