

チーム課題

<<パターンA：記事の自動分類>>

1. wikipediaや星空文庫,ニュースサイト,Q&Aサイトなど、カテゴリ情報があるテキストを複数カテゴリ・複数テキストをあつめる。
2. カテゴリ毎に特徴語（単語出現回数,TF-IDF値が高いTop N）を抽出し、比較
3. 集めたデータの何割かを教師あり学習で学習したのち、学習に使用しなかったデータのカテゴリを推定し、その結果について考察する

実験例)

- ・ニュースを自動分類できるか

<<パターンB：分類と特徴抽出>>

1. wikipediaや星空文庫,ニュースサイト,Q&Aサイトなどで、1カテゴリまたは2カテゴリからテキストを集める
2. 集めたデータをクラスタリングし、クラスタ毎の特徴語（単語出現回数,TF-IDF値が高いTop N）を抽出し、比較

実験例)

- ・大学を分類し、どのような特徴で分類されているか
- ・スポーツのニュースを分類し、どのような特徴で分類されているか

チーム課題スケジュール



発表について

<<発表方法>>

プロジェクタにスライド等を投影しながら発表

<<発表時間>>

発表：15分 質疑：10分

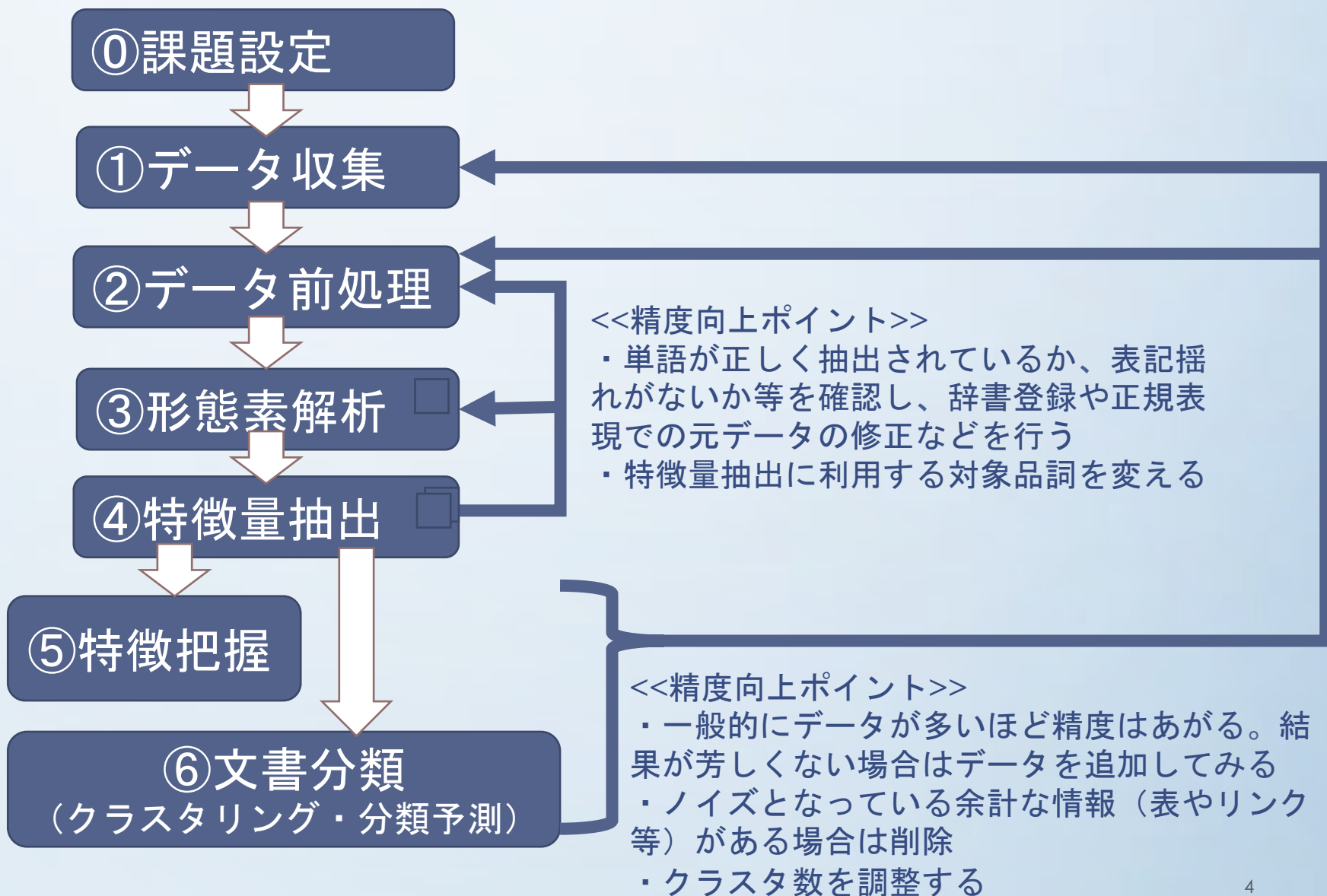
※積極的に質問を。(質疑も評価します)

<<発表内容>>

- ・ チーム名/メンバーと主な担当箇所
- ・ 実験の目的と使用データ
- ・ 手法 (利用辞書, 利用品詞, 前処理/形態素解析等の工夫点, クラスタリング/学習メソッド 等)
- ・ 結果と考察
- ・ 苦労点/考えられる今後の改善点

※特徴把握や考察などの集計/グラフ化にはExcel等を用いても良い

テキストマイニングの基本フローとポイント



補足

- クラスタリングや機械学習ではデータ量はなるべく多い方が良い。
小説は1文書が多くなるので、Wikipwdiaやニュースなどの記事で数を多くした方がよい
(小説を章毎にわかるなどの工夫をしても良い)
- Webからテキストを集めるのに「Webクローラ」がある(wgetなど)が、サーバーに負荷をかけないように、クローリング間隔は十分に(3秒以上)とる。
 - ※ クローリング&Webページから本文を抽出するツールWebstemmer
<http://www.unixuser.org/~euske/python/webstemmer/index-j.html>
 - ※ Wikipwdiaはクローリングが禁止されている
 - ※ 数十~100記事 程度なら 手分けしてコピーのほうが確実
- ロコミは非常に有益なテキスト情報だが、精度を本気であげるためには、誤字脱字や顔文字など前処理に手間がかかる
 - ・ Twitterは短文のため扱いが難しい(クラスタリング等が目的の場合は特に)

<発表後の提出物>

発表スライド・使用データ・プログラム