機械学習の続きとJSONについて

グループI4
2008MI233 鈴木健太
2008MI214 沢田天馬

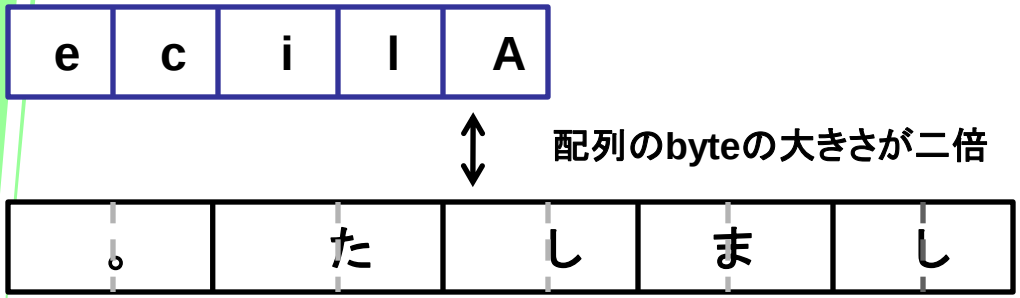
目次

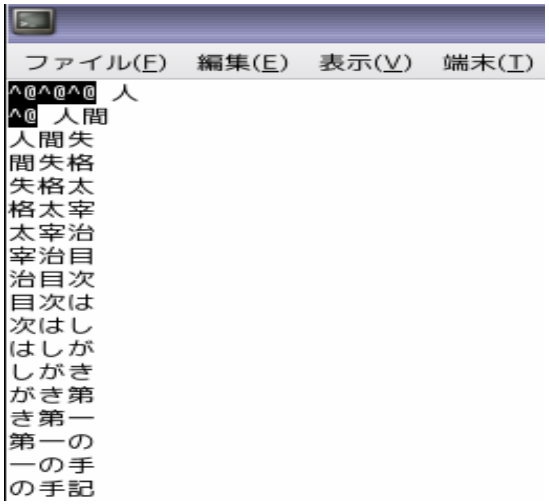
- 日本語テキストデータの学習
- 汎化の導入
- 分類知識の学習
- PredictionAPIの続き
- JSON(JavaScript Object Notation)
- JSONの記法
- PredicitonAPIの場合
- 今後の課題
- 参考文献

日本語テキストデータの学習(1/2)

テキストデータ:太宰治の「人間失格」
入手先:青空文庫

前回のngram.cを変更し、ngramsjis.cを作成





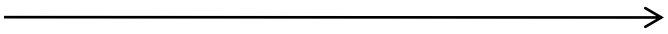
「人間失格」の3-gram

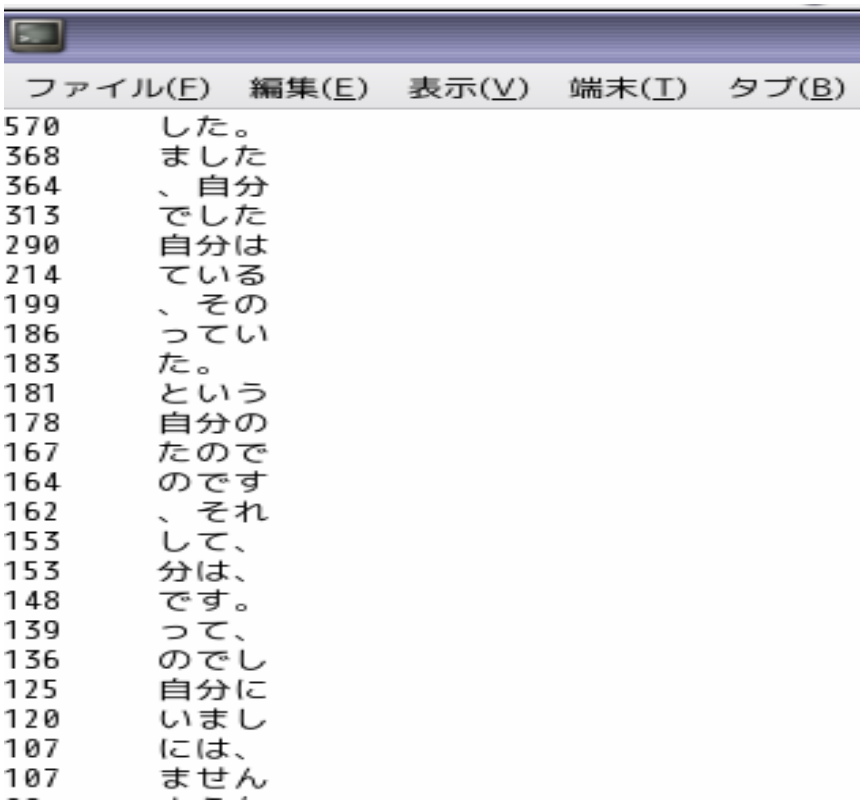
```
n=n*2;

while((chr=getchar()) != EOF){
  d =(unsigned char)chr;
  if(((d>0x7f) && (d<0xa0)) || (d>0xdf) && (d<0xf0)){
    setlastch(n,chr,lastch);
    chr=getchar();
    setlastch(n,chr,lastch);

    printngram(n,lastch);
  }
}

return 0 ;

}

void setlastch (int n,char chr,char lastch [])
{
int i;

for(i=n-2;i>=0;--i)
     lastch[i+1] =lastch[i];

     lastch[0] =chr;
     }
```

ngramsjis.c

日本語テキストデータの学習(2/2)

前回のrank.cを使い、実行





人間失格の3-gramの頻度順

実装の結果

日本語でも頻度をはかることができた
文字コードのバイト数の違いによって
作成するプログラムがかわってしまう。
これにより、英語と日本語の混ざった
データに対する適用が難しいかもしれない

さらなる利用のためには、C言語以外での実装も考えなければならない

汎化の導入

n-gramによって抽出した文書の特徴を一般化する

出現順位	頻度	5-gram
1	1483	the
2	690	and
3	459	d the
4	447	said
5	431	, and
6	423	she
7	403	said

例：tf-idf法

ある文書の中に出現する文字列が、その文書の特徴をどれだけ表しているのかを数値で表現するための手法

tf（term frequency）：
ある文字列の文書中での出現回数

×

積

idf(inverse document frequency)
ある文字列が一般の文書全体のうちのどれくらいの文書に出現するかに関連した値

分類知識の学習(1/3)

与えられたデータがどのカテゴリに該当するかを分類する知識を学習する

例：迷惑メールの判別

命題 p1　件名に“激安”という語を含むか？

命題 p2　件名に“無料”という語を含むか？

命題 p3　本文が10文字以下か？

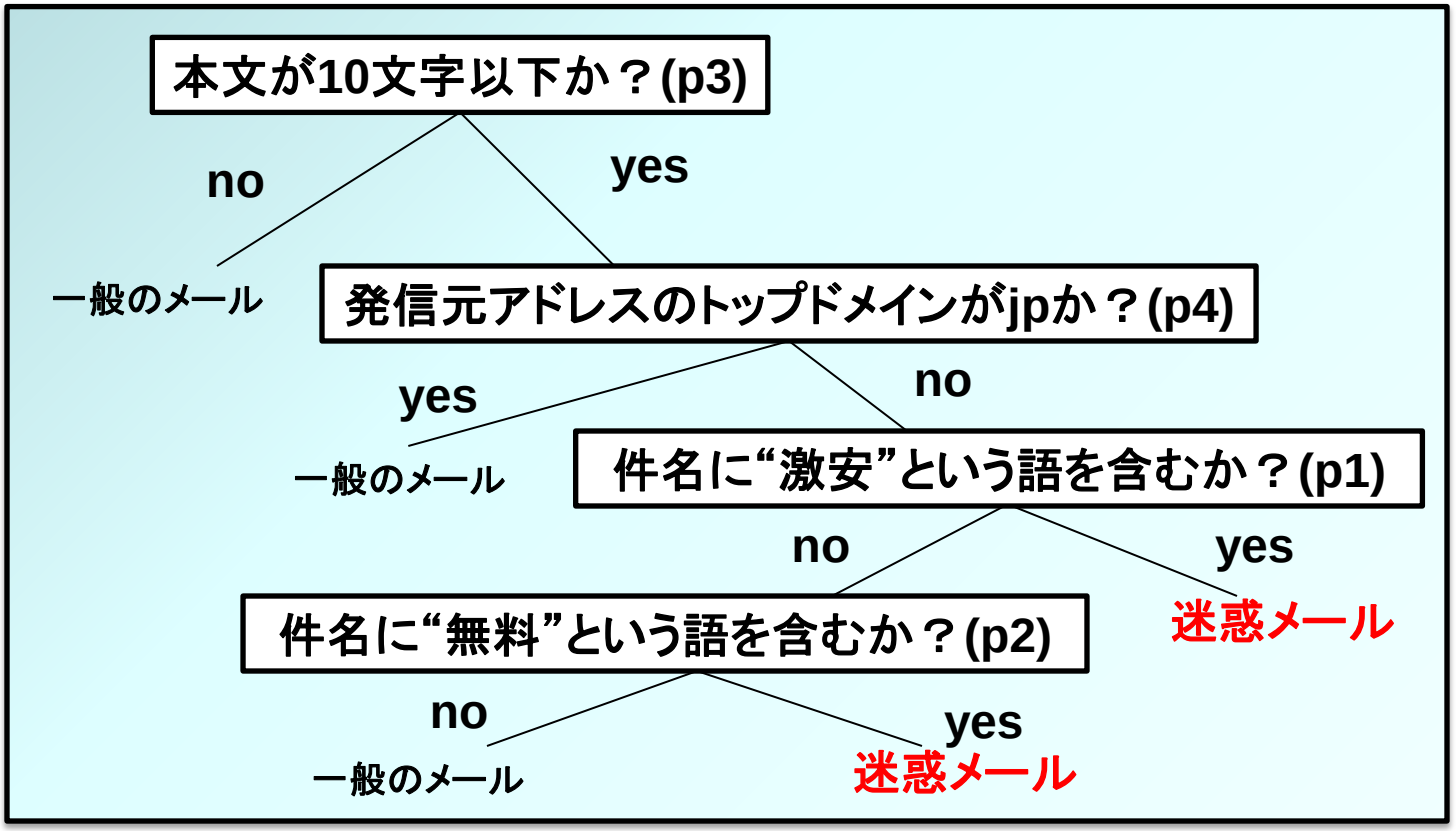
命題 p4　発信元アドレスのトップドメインがjpか？

論理式　：　$(p1 \lor p2) \land p3 \land \neg p4$

参考資料における迷惑メールの定義の一例

件名に“激安”または“無料”という語を含み、かつ、本文が10文字以下であり、かつ、発信元アドレスのトップドメインがjpでないメールは迷惑メールである。

分類知識の学習(2/3)

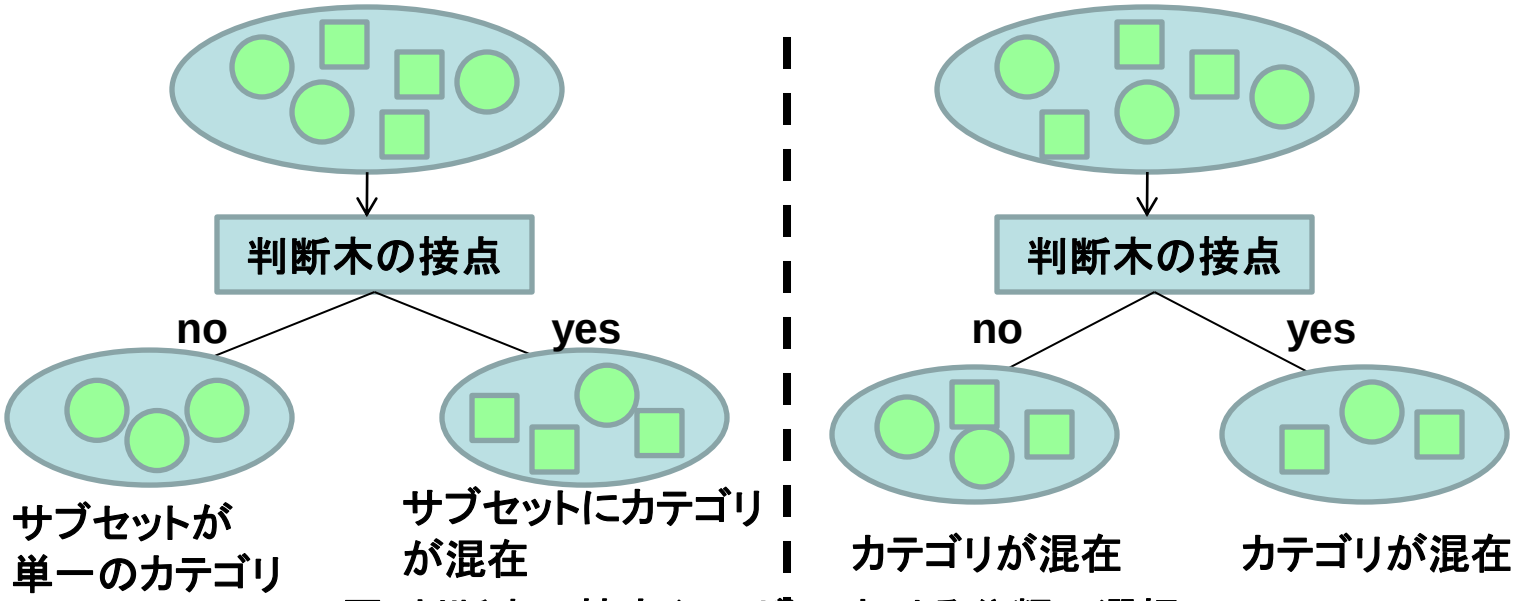


図：判断木の適用例

プロダクションシステムにおける表現方法

ルール 1： if p3 == yes then ルール2を適用する
ルール 2： if p4 == no then ルール 3を適用する
ルール 3： if p1 == no then ルール 4を適用する
ルール 4： if p2 == yes then 迷惑メールと判断する

分類知識の学習(3/3)



図：判断木の接点（ノード）における分類の選択

判断木の機械学習アルゴリズム

（1）与えられた学習セットが空ならば、学習終了

（2）与えられた学習セットの要素が全て単一のカテゴリに属していれば、学習終了

（3）学習セットを適切に分類する属性を選んで、学習セットをサブセットに分類する。それぞれのサブセットに、学習手続きを再帰的に適用する

（4）適用できる属性が無いのに分類が終わっていなければ、学習を完成せずに手続きを終了する

Prediction APIの続き

3つのステップで
PredictionAPIの利用が可能

Google Storage に
トレーニングデータを
アップロードする

アップロード

トレーニング
リクエストを行う

トレーニング

トレーニング完了の
レスポンスが存在

予測リクエストを行う

予測

予測結果の
レスポンスが存在

➢リクエストとレスポンスの形式についての理解を深めた
⇒形式としてJSONが用いられていることがわかった

JSON（JavaScript Object Notation）

データの表現と交換を目的としたテキストベースのデータフォーマット

XMLとの比較を用いた例

XML

```
<?xml version="1.0" encoding="Shift-Jis"?>
<student>
   <MI>
      <name>鈴木健太</name>
   </MI>
   <MI>
      <name>沢田天馬</name>
   </MI>
</student>
```

JSON

```
{
  "student":
  [
    {
      "name":"鈴木健太"
    },
    {
      "name":"沢田天馬"
    }
  ]
}
```

XMLと比べて簡略で、通信時のデータ量削減につながっている

JSONの記法

オブジェクトと配列で構造化されたデータを表現

オブジェクト

{}で全体を囲み、キーと値のペアを:(コロン)で区切る、,(カンマ)で複数の記述も可能

```
{“name”:”鈴木健太”,”age”:”21”}
```

配列

繰り返し項目を表現する際に使用、全体を[]で囲み、値を,(カンマ)で区切って列挙する

```
[“JavaScript”,”JSON”,”Ajax”]
```

オブジェクトと配列のネスト

```
{“languages”:[“JavaScript”,”PHP”,”XML” ] }
```

数値、文字列、真偽値、配列、オブジェクト、nullのデータ型が使用可能

Prediction APIの場合

HelloPredicitonのレスポンス


```
{
  “kind”:”prediction#output”,
                  .
                  .
                  .
"outputLabel": "Spanish",
"outputMulti": [
    {
     "label": "French", "score": 0.334130
    },
    {
     "label": "Spanish", "score": 0.418339
    },
   {
     "label": "English", "score": 0.247531
    }
 ]
}
```


赤い丸は
変更可能であると考えられる部分

データを抽出して利用ができないか

考えられるレスポンスの例


```
{

“outputLabel”:”URL2”,
 “outputMulti”:[
   {
     “label”:”URL1”,”score”:0.334130
    },
   {
     “label”:”URL2”,”score”:0.418339
    }
  ]
}
```

今後の課題

テーマ

コンテキストアウェアなサービス提供技術の提案

考えていかなくてはならないこと

対象（ナビ・携帯など）をどうするのか

機械学習を用いるとして、どのようなことを学習させるのか

与えられた情報を機械学習に利用するための仕組みはどうすればいいのか

参考文献

Think IT　http://thinkit.co.jp/

Google I/O 2011: Smart App Design
http://www.youtube.com/watch?v=FJDP_0Mrb-w&feature=player_embedded

はじめての機械学習　小高　知宏