Base SASとオープンソースだけで行う テキストマイニングの検討 （MeCab, CaboCha及びWord2Vecの連携について）

吹谷 芳博
株式会社エスアールディ データマネジメント統計解析室

Consideration of text mining only by means of Base SAS and Open Source. (Coordination of MeCab, CaboCha and Word2Vec)

Yoshihiro Fukiya
Data Management/Biostatistics, SRD Co., Ltd.

要旨:

ビックデータではテキストマイニングの重要性は高い.
今回は医療事故／ヒヤリ・ハット事例データを用いてMeCab, CaboCha及びWord2Vecと連携して分析した事例を紹介する.

キーワード:テキストマイニング, MeCab, CaboCha, Word2Vec

目次

- 目的
- 背景と導入
  - MeCab, CaboCha, Word2Vecの説明
- 方法
  - 解析環境（使用データ、辞書等）
  - ツールの連携方法とプログラムの例示
  - テキストマイニングのフロー
- 結果と考察
- まとめ

目次

- 目的
- 背景と導入
  - MeCab, CaboCha, Word2Vecの説明
- 方法
  - 解析環境（使用データ、辞書等）
  - ツールの連携方法とプログラムの例示
  - テキストマイニングのフロー
- 結果と考察
- まとめ

目的

1. Base SAS + オープンソースの連携をすれば
   テキストマイニングが可能である.
2. テキストマイニング手法を用いて得られた結果から
   リスク評価の指標などに使用できる.

導入コストを低くして安全性データなどから
シグナル検出し, リスクの指標として可能では!!

目次

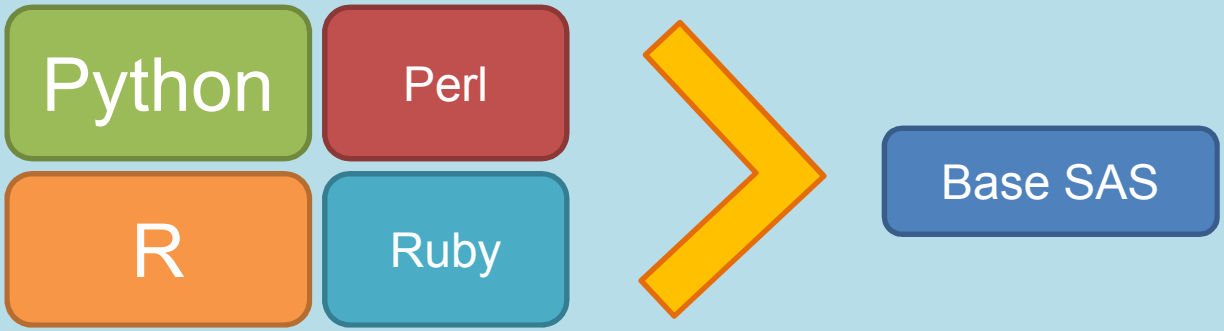
- 目的
- **背景と導入**
  - **MeCab, CaboCha, Word2Vecの説明**
- 方法
  - 解析環境（使用データ、辞書等）
  - ツールの連携方法とプログラムの例示
  - テキストマイニングのフロー
- 結果と考察
- まとめ

背景

医薬品表示からテキストマイニングで
安全性情報等の抽出を行っている.

しかし・・・・

テキストマイニング事例：（日本語において）



一つの要因は・・・

MeCabなどの連携の事例がないから・・・・？

導入（1）MeCabとは？

形態素解析

文を単語レベルに分けて品詞を判別する作業.

例）吹谷さんは冷たいからモテない。

吹谷　名詞,固有名詞,地域,一般,*,*,吹谷,フキダニ,フキダニ
さん　名詞,接尾,人名,*,*,*,さん,サン,サン
は　助詞,係助詞,*,*,*,*,は,ハ,ワ
冷たい　形容詞,自立,*,*,形容詞・アウオ段,基本形,冷たい,ツメタイ,ツメタイ
から　助詞,接続助詞,*,*,*,*,から,カラ,カラ
モテ　名詞,一般,*,*,*,*,*
ない　形容詞,自立,*,*,形容詞・アウオ段,基本形,ない,ナイ,ナイ
。　記号,句点,*,*,*,*,。,。,。

導入（2） CaboChaとは？

係り受け解析
　文節間の関係性を構造化する作業.

例）吹谷さんは変だからモテない。

吹谷さんは---D
　変だから-D
　　モテない。

吹谷さんは => モテない。

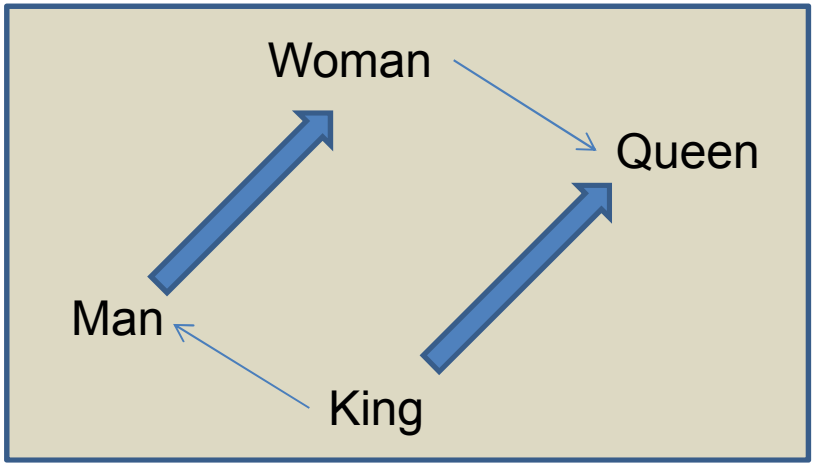
変だから => モテない。

導入（3）Word2Vecとは？

単語をニューラルネットワークによって
ベクトル変換させるプログラムライブラリ.

例）“King” – “Man” + “Woman” = “Queen”

ベクトル変換後の模式図



例）“医師”との類似度

単語	類似度
Dr	0.7496
主治医	0.6988
担当医	0.6688
DR	0.6175
当直医	0.6039
当直医師	0.6022

目次

- 目的
- 背景と導入
  - MeCab, CaboCha, Word2Vecの説明
- **方法**
  - **解析環境（使用データ、辞書等）**
  - **テキストマイニングのフロー**
  - **ツールの連携方法とプログラムの例示**
- 結果と考察
- まとめ

解析環境(1)

Base SAS Ver9.2（OS:Windows 7）

使用データ

日本医療機能評価機構のデータベース

ヒヤリ・ハット報告事例のみ使用

recode数：30833（重複除去）、総文字数： 4521211

MeCabの辞書

- ipadic（デフォルト）
- MeCab-ipadic-neologd（ipadic拡張版）
- ComeJisyo（医療従事者用辞書：Ver5.1）

解析環境(2)

Stop wordのデータ（不要な単語を除くため）

Google Code (Project: Stop-words) + Custom

Pythonの環境（Word2Vecの利用のため）

Anacondaをインストール.

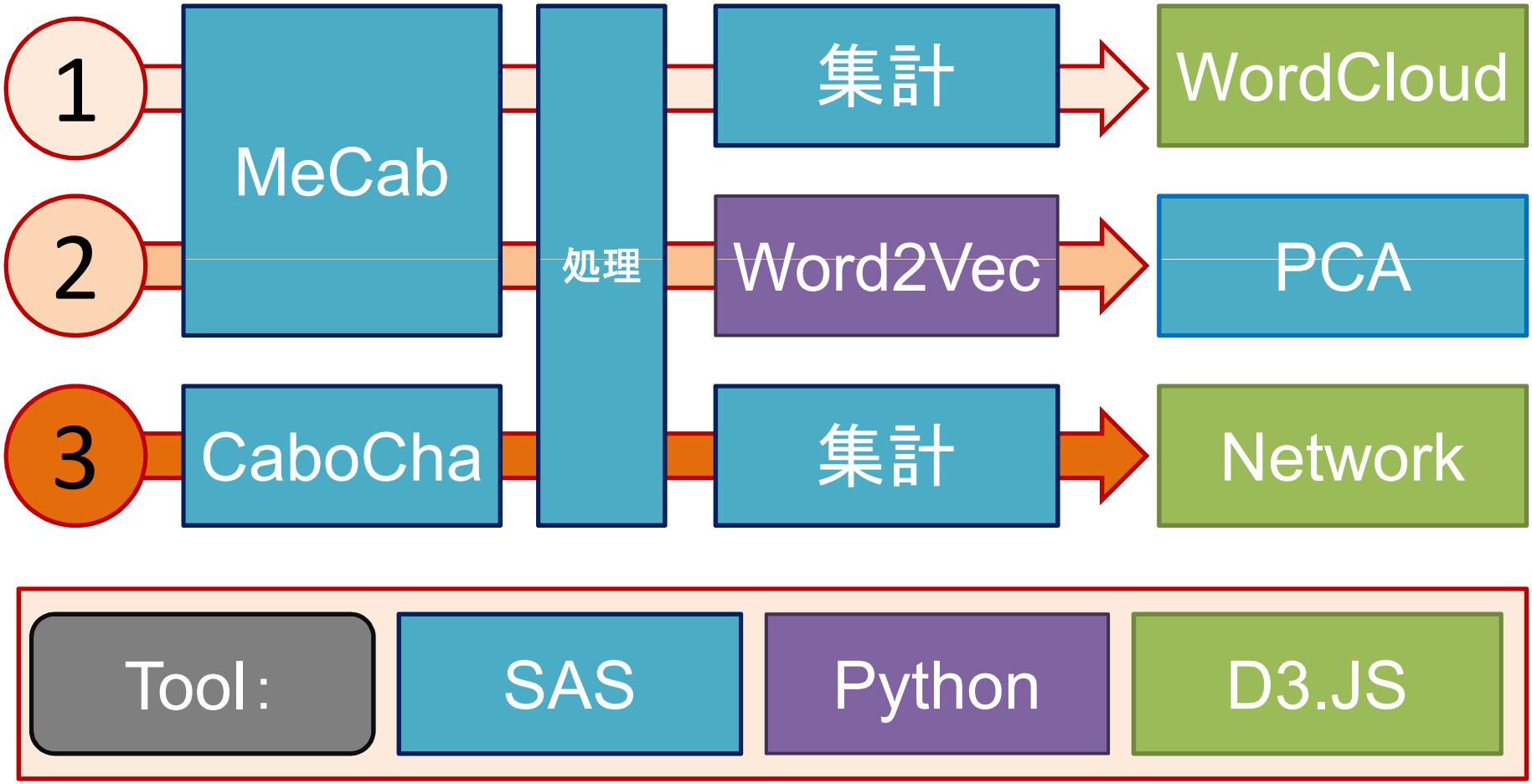
Command: easy_install -U gensim

Command: pip install cython

D3.JS（Graphの表示等で利用のため）

URLを指定してJavaScript libraryを読み込む

テキストマイニングのフロー



連携方法(1)MeCab, CaboCha

•Proc ProtoによりMeCab libraryを読込み、関数を定義（一部抜粋）.

```
proc proto package=FUNCLIB.MYLIB.MeCab label="Package of MeCab";
   link "C:¥Program Files¥MeCab¥bin¥libMeCab.dll";
   char * MeCab_sparse_tostr(int m, char * str);
run; quit;
```

•Proc Fcmpにより上記で定義した関数を基にSAS関数定義（一部抜粋）.

```
proc fcmp inlib =FUNCLIB.MYLIB outlib=FUNCLIB.MYLIB.MeCabF;
   *--- MeCab Sparse ---*;
   subroutine MeCab_spa(OUTUNITS, STR1 $, STR2 $);
      array   A_STR1{1} $30000 STR1 ; outargs STR2;
      if not missing(A_STR1[1]) then do;
         STR2 = MeCab_sparse_tostr(OUTUNITS, A_STR1[1]);
      end;
      return;
   endsub;
run; quit;
```

CaboChaも同様の方法で関数を指定。

連携方法(2)Word2Vec（その1）

•W2V_Run.py（Word2Vecを実行して単語からベクトル変換）

```
# -*- coding: utf-8 -*-
import codecs, sys, string, os.path
from gensim.models import word2vec
argvs = sys.argv; txtpath = argvs[1]; outpath = os.path.dirname(argvs[2]) + '¥¥'
data = word2vec.Text8Corpus(txtpath.replace('¥¥', '¥¥¥¥'))
model= word2vec.Word2Vec(data, size=200)
kwds = codecs.open(argvs[2], 'r', encoding='utf-8').read().split()
csvf = codecs.open(outpath + 'result.csv', 'w', encoding='shift-jis')
for x in kwds:
  val = model[x]
  csvf.write("" + x + "" + ',' ); n = 0
  for v in val:
    if n==0:
      csvf.write(str(v))
    else:
      csvf.write(','+str(v))
    n += 1
  csvf.write('¥n')
csvf.close()
```

連携方法(2)Word2Vec（その2）

Pythonの実行

```
X 'python "C:¥W2V_Run.py" "C:¥corpus.txt" "C:¥List.txt"';
```

- corpus.txt：モデル化するコーパスファイル
- List.txt：ベクトル変換させる単語リスト

SAS実行

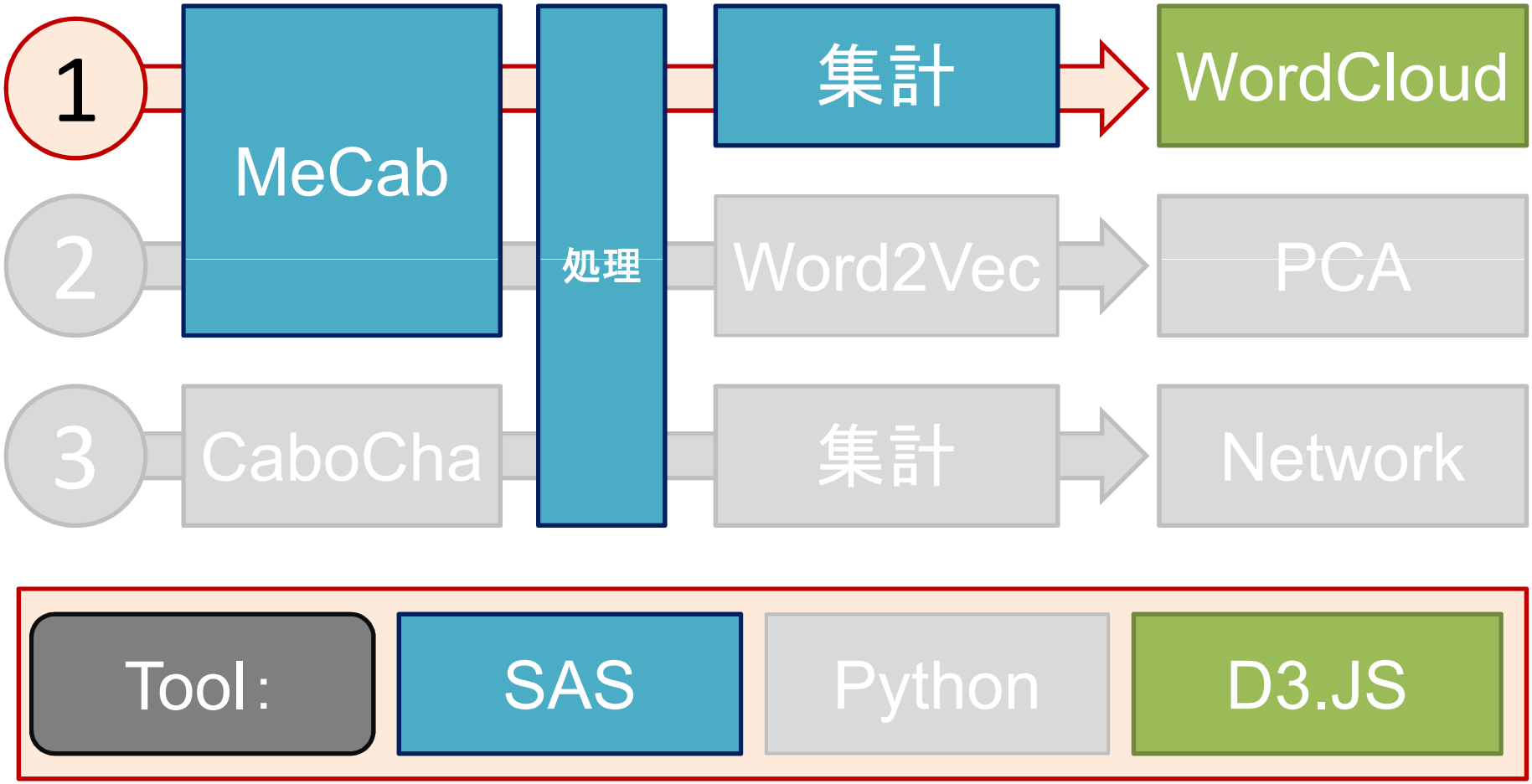
```
proc import out= WORK.VecData
  datafile="C:¥result.csv" dbms=CSV REPLACE;
  getnames=No; datarow=1; guessingrows=200;
run;

ods graphics on / width=10in height=8in;
proc princomp data=VecData plots= score(ncomp=3);
  id  VAR1; var VAR2-VAR201;
run; quit;
ods graphics off;
```

目次

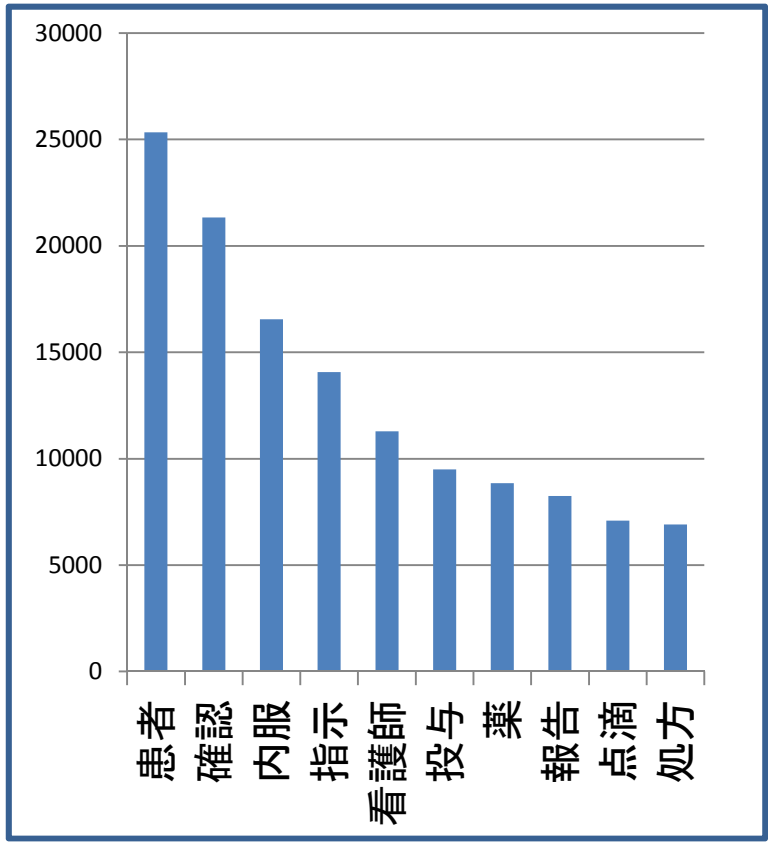
- 目的
- 背景と導入
  - MeCab, CaboCha, Word2Vecの説明
- 方法
  - 解析環境（使用データ、辞書等）
  - ツールの連携方法とプログラムの例示
  - テキストマイニングのフロー
- **結果と考察**
- まとめ

テキストマイニングのフロー

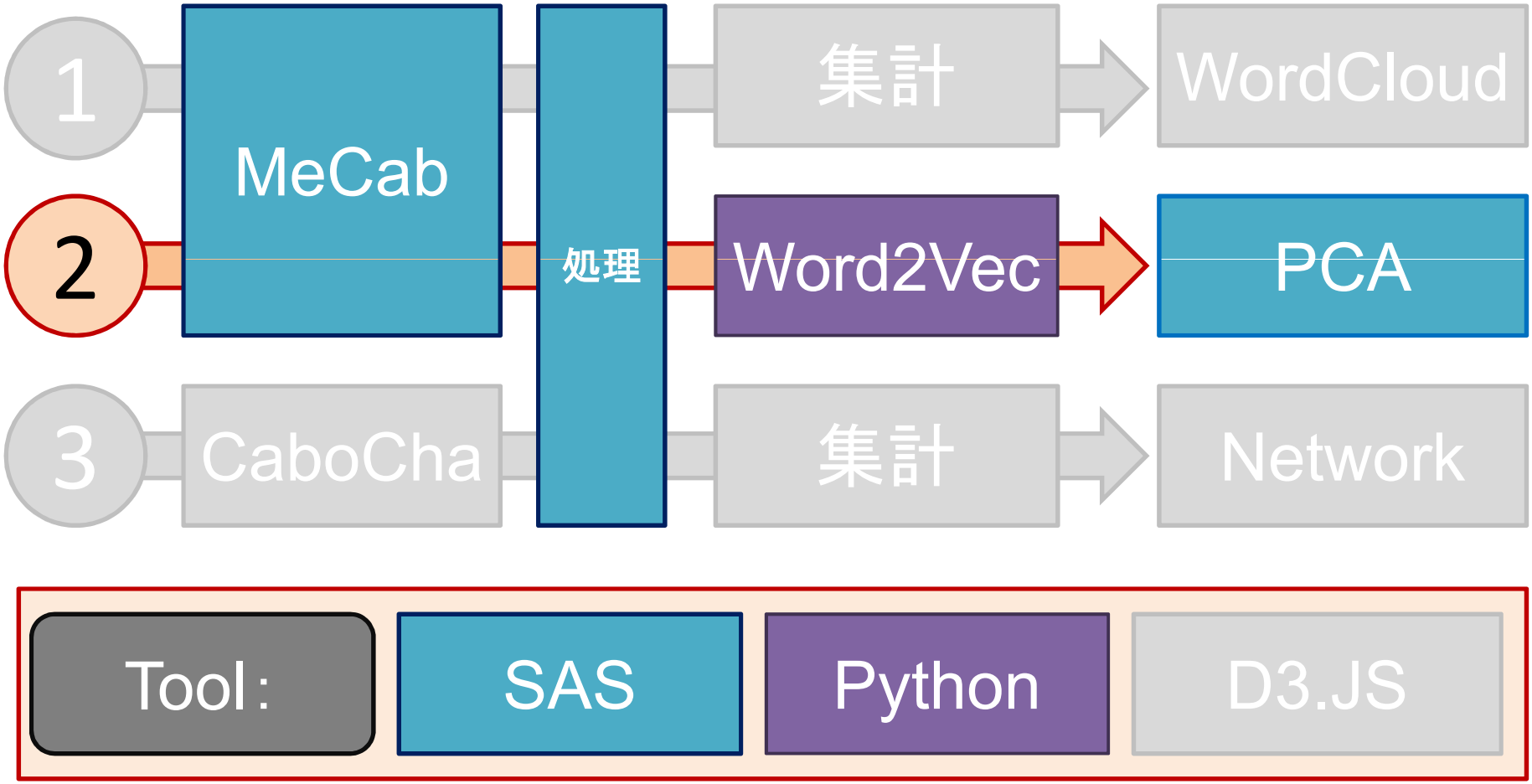


結果（1）集計結果とWordCloudの表示

ヒヤリ・ハット事例の内容を特徴づける単語が抽出された.



テキストマイニングのフロー

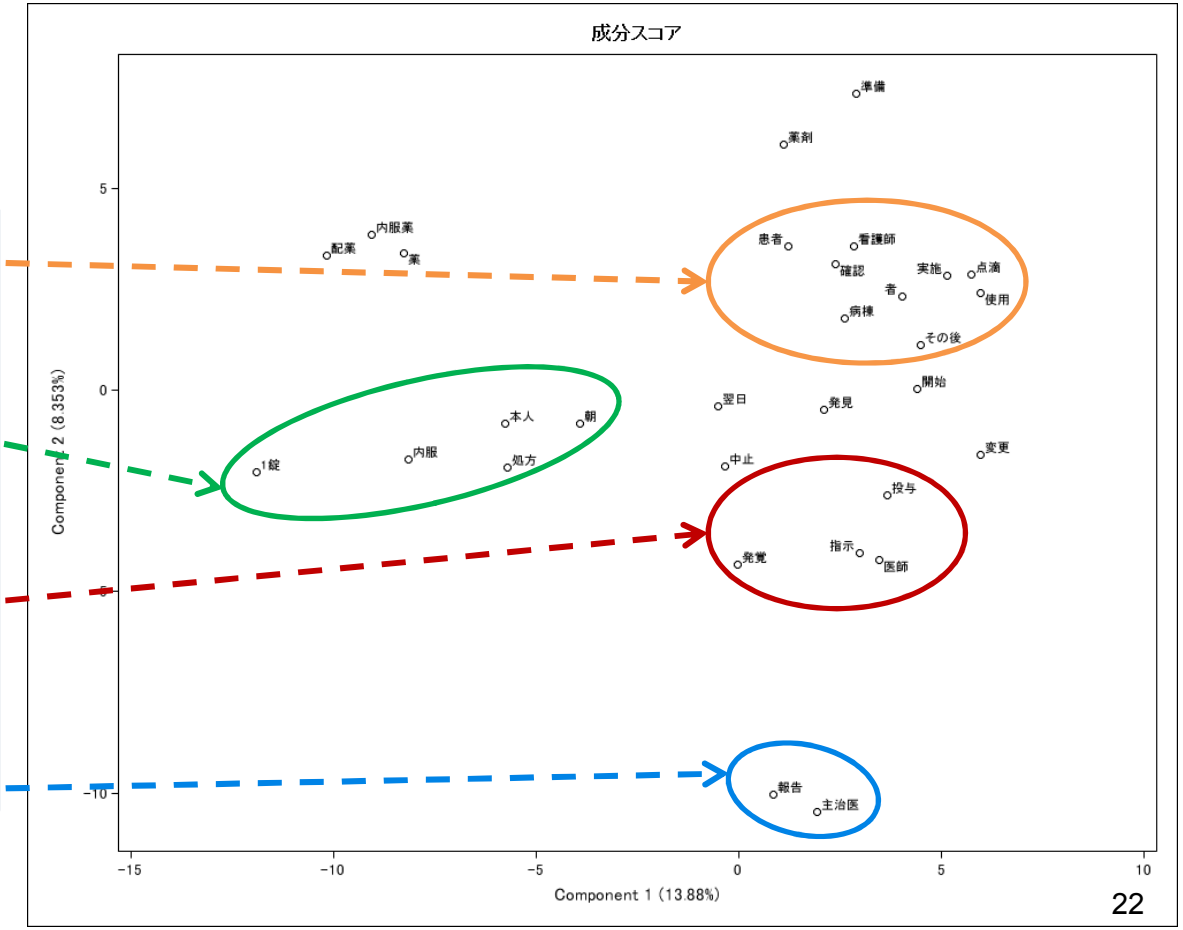


結果(2)PCAによる分類

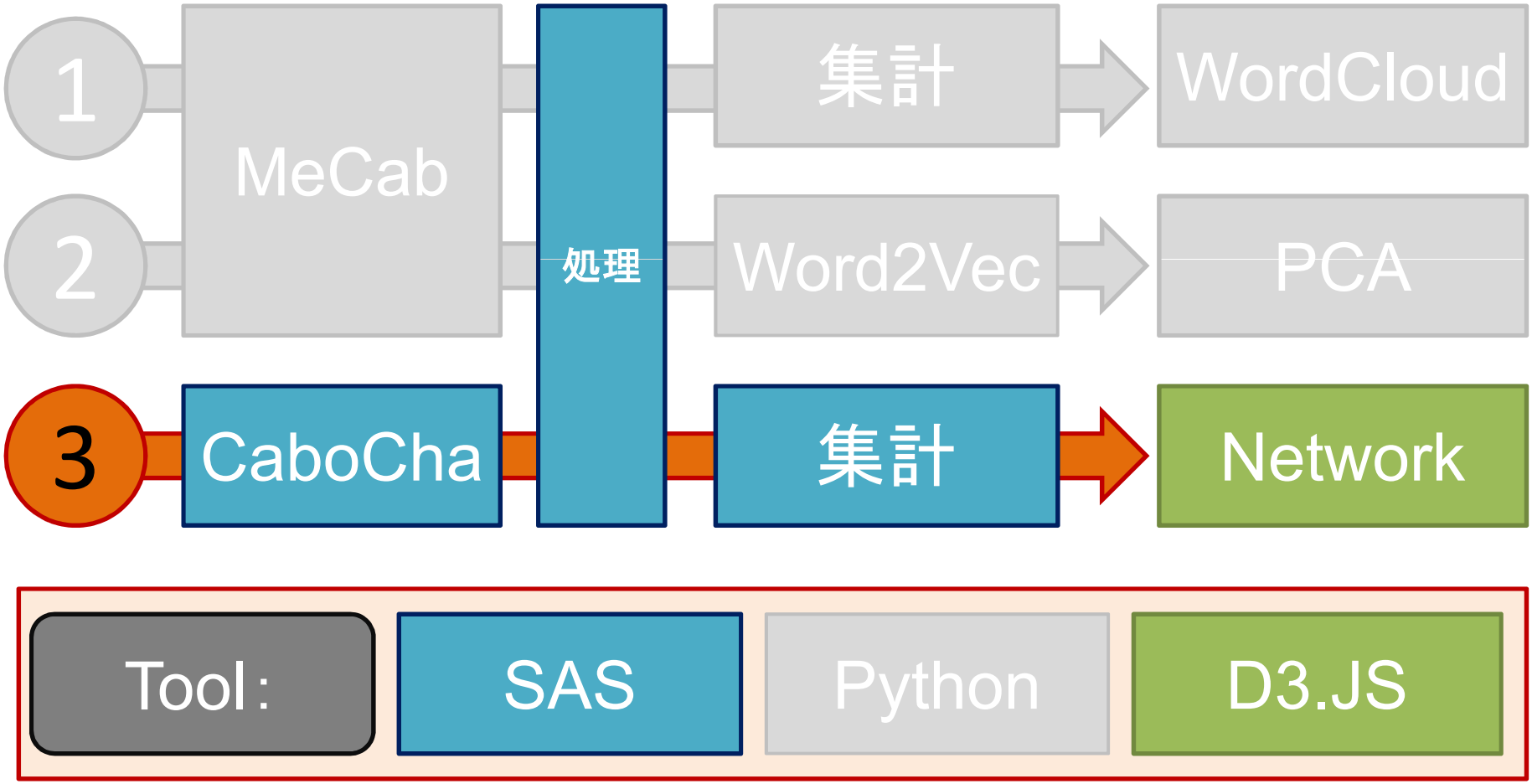
単語間の関連性を特徴づける結果が出た.

主なグループ

- 患者/看護師
- 本人
- 医師
- 主治医



テキストマイニングのフロー

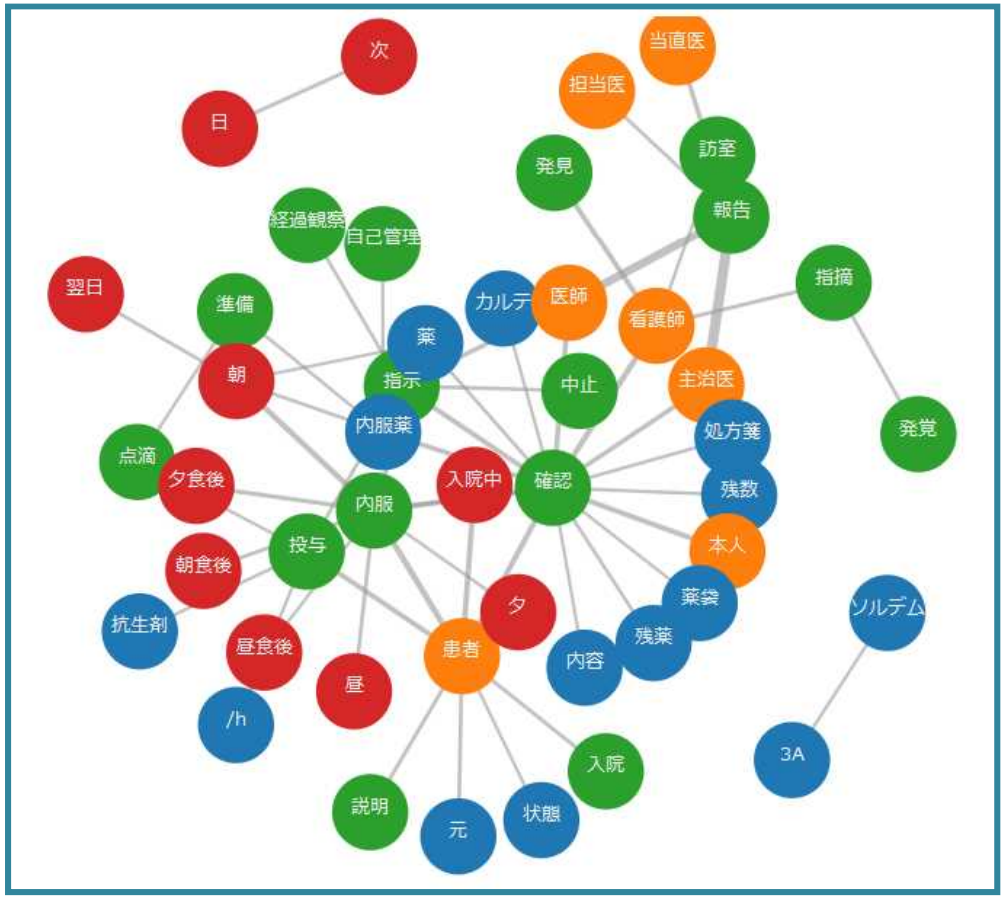


結果(3)Network解析(単語の関連性)

“確認”を中心とした単語間の関連性が示唆された.

Nodeの色分け

- 人物
- 動詞
- 時期
- その他



考察

- 一通りのテキストマイニングが可能.
- 同義語や類似語の処理を強化すれば精度上昇.
  - MeCabの辞書のカスタマイズ.
  - 機会学習との組み合わせ.

- 身近なテキストデータが活用可能.
  - モニタリング報告書.
  - 安全性のデータ.

目次

- 目的
- 背景と導入
  - MeCab, CaboCha, Word2Vecの説明
- 方法
  - 解析環境（使用データ、辞書等）
  - ツールの連携方法とプログラムの例示
  - テキストマイニングのフロー
- 結果と考察
- まとめ

まとめ

- Base SASとオープンソースツールの連携で容易にテキストマイニングが可能.

- 公共のデータベースや社内のテキストデータを使い，臨床試験におけるリスク評価等にも活用できる可能性も.

参考

テキストマイニングに関する文献

- 木村 昌臣,大倉 典子,土屋 文人 (2006)
  テキストマイニングによる医薬品投薬ヒヤリ・ハット事例の解析. 人間工学 Vol.42 P 232-233
- Halil B, Zhichao L, Hong F, Xiaowei X (2011) Mining FDA drug labels using an unsupervised learning technique - topic modeling. BMC Bioinformatics. 12(Suppl 10): S11.

Word2Vecに関する参考図書

- 西尾泰和 (2014)「Word2Vecによる自然言語処理」, オライリージャパン.

MeCabの辞書

- MeCab-ipadic-neologd. https://github.com/neologd/MeCab-ipadic-neologd
- ComeJisyo http://osdn.jp/projects/comedic/

D3.js (JavaScript library)

- Wordcloud: https://gist.github.com/emeeks/3361332
- Network: http://bl.ocks.org/mbostock/4062045