1.コーパスとテキストマイニング

九州大学大学院言語文化研究院
内田　諭

1.1. コーパスとは？

- コーパス＝電子化された言語資料

- 汎用コーパス（general corpus）

→ランダムサンプリングに基づいて均衡化されたもの

例：BNC（British National Corpus）、BCCWJ（現代日本語書き言葉コーパス）

- 特殊コーパス（specialized corpus）

→特定の目的に沿って作られたもの

例：医学英語コーパス、英語学習者コーパス

1.1. コーパスとは？

サンプルコーパス（静的コーパス）

BNC、BCCWJなど

モニターコーパス（動的コーパス）

COCA (Corpus of Contemporary American English)、Bank of Englishなど

1.1.テキストマイニングとは？

ウェブ≒電子化された言語資料

※コーパスとは異なり、構造化されていない

テキストマイニング

→構造化されていないデータから情報を抽出する

1.1.テキストマイニングとは？

（ウェブから）データの収集
↓
クリーニング
↓
タグ（情報）付け（形態素解析）
↓
分析（視覚化、統計の利用）

1.2. テキストマイニングの準備

- STEP1
- テキストの収集

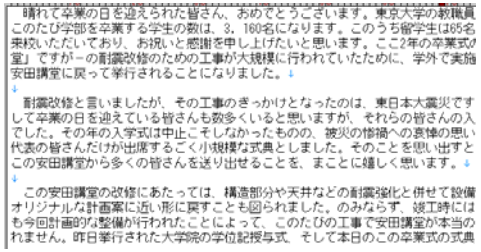
→【実例】総長のスピーチ
http://www.u-tokyo.ac.jp/gen01/b_message26_10_j.html

平成26年度東京大学
平成26年度京都大学
平成25年度九州大学
平成26年度九州大学

1.2. テキストマイニングの準備

- クリーニング

→タグの削除、ふりがなの削除、etc.
テキストファイルとして保存：
Todai_H26.txt
Kyodai_H26.txt
Kyudai_H25.txt
Kyudai_H26.txt



晴れて卒業の日を迎えられた皆さん、おめでとうございます。東京大学の教職員のたび学部を卒業する学生の数は、3，160名になります。このうち留学生は65名来校いただいており、お祝いと感謝を申し上げたいと思います。ここ2年の卒業式の会場」ですが－の耐震改修のための工事が大規模に行われていたために、学外で実施安田講堂に戻って挙行されることになりました。

耐震改修と言いましたが、その工事のきっかけとなったのは、東日本大震災ですして卒業の日を迎えている皆さんも数多くいると思いますが、それらの皆さんの入でした。その年の入学式は中止こそしなかったものの、被災の惨禍への哀悼の思い代表の皆さんだけが出席するごく小規模な式典としました。そのことを思い出すとこの安田講堂から多くの皆さんを送り出せることを、まことに嬉しく思います。

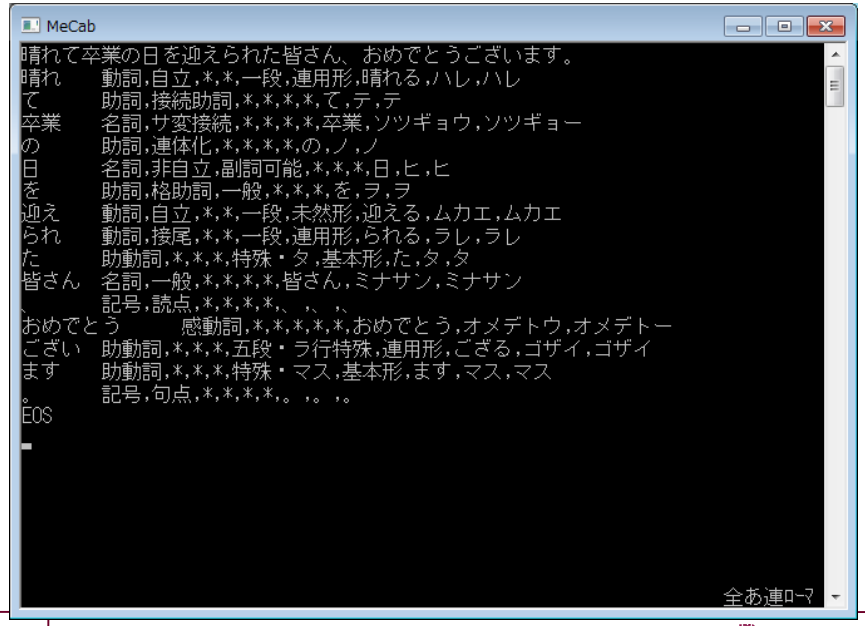
この安田講堂の改修にあたっては、構造部分や天井などの耐震強化と併せて設備オリジナルな計画案に近い形に戻すことも図られました。のみならず、竣工時には今回計画的な整備が行われたことによって、このたびの工事で安田講堂が本当のれません。昨日挙行された大学院の学位記授与式、そして本日のこの卒業式の式典

1.2.テキストマイニングの準備

- テキストの処理と加工

A：形態素解析
→MeCab
日本語の場合、単語の切れ目がわからないため、「単語の頻度を数える」という単純な作業でも形態素解析が必要。見出し語形での検索、品詞情報なども分析の上で有用。

B：構文解析
→CaboCha、Stanford Parser



1.3テキストの構造化

- N-gramモデル

N個の隣接する文字・単語・文節などの集計し、パターンを分析する（cf.前回の中藤先生のご発表）

- 共起語（コロケーション）の分析

例：

- tallの直後にくる名詞

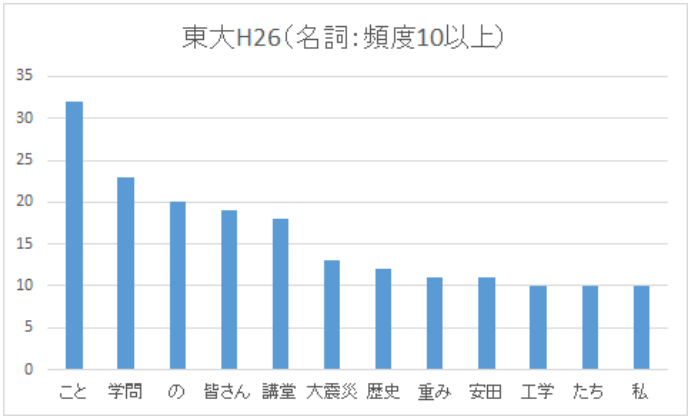
chimney, tree, building, man, tower, etc.

- highの直後にくる名詞

speed, quality, level, temperature, etc.

1.4.テキストマイニングの主な方法

1.4.1. 視覚化：単純なグラフ



1.4.テキストマイニングの主な方法

1.4.1. 視覚化：ワードクラウド（Todai_H26）

1.4.テキストマイニングの主な方法

1.4.1. 視覚化：ワードクラウド(Kyodai_H26)

1.4.テキストマイニングの主な方法

1.4.1. 視覚化：ワードクラウド(Kyudai_H25)

1.4.テキストマイニングの主な方法

1.4.1. 視覚化：ワードクラウド(Kyudai_H26)

合わせて質的な分析が必須

1.4.テキストマイニングの主な方法

1.4.1. 視覚化：Rでワードクラウドを作る【参考】

(1)MeCabをコンピュータにインストール
(2)RMeCabをインストール
install.packages("RMeCab", repos = "http://rmecab.jp/R")
(3)"tm"と"wordcloud"をpackagesのメニューからインストール
(4)パッケージの読み込み
library(wordcloud)
library(RMeCab)
(5)RMeCabで頻度を解析
data.freq=RMeCabFreq("data.txt")
(6)品詞のフィルタリング
data.sub=subset(data.freq, Info1=="名詞" & Info2!="数", c(Term, Freq))
#名詞を指定、数を除外
(7)描画
wordcloud(data.sub[,1], freq=data.sub[,2], min.freq=3,random.order=F)

1.4テキストマイニングの主な方法

1.4.2. 特徴を抽出するための指標

TF-IDF

カイ二乗統計量

共起強度

1.4テキストマイニングの主な方法

1.4.2. 特徴を抽出するための指標

TF＝Term Frequency
→たくさん出現する単語＝重要！
Q：the, a, of, atなどは頻度が高いが重要か？
↓
IDF=Inverted Document Frequency
→いろんなテキストで使われていれば特徴的ではない！

1.4テキストマイニングの主な方法

1.4.2. 特徴を抽出するための指標

TF＝Term Frequency

テキスト中の単語の頻度　OR　$\frac{\text{テキスト中の単語の頻度}}{\text{テキストの総語数}}$

IDF=Inverted Document Frequency

$$\text{Log}\frac{\text{総テキスト数}}{\text{単語の出現テキスト数}}+1$$

TF-IDF=TF（頻度 OR 相対頻度）×IDF（重み付け）

総長スピーチでのTF-IDF

- 例：

単語	Kyudai_H25.txt	Kyudai_H26.txt	Kyodai_H26.txt	Todai_H26.txt
アクティブ・ラーナー	3	0	0	0

TF=3
IDF=$\log_2(4/1)+1$
TF-IDF=3×3＝9
→IDFによって重み付けがされた！

総長スピーチでのTF-IDF（九大H25がプラスのもの）

単語	Kyudai_H25.t	Kyudai_H26.t	Kyodai_H26.t	Todai_H26.tx
教育	48	8	0	0
等	34	4	0	0
九州大学	32	26	0	0
平成	28	8	0	0
プログラム	27	0	0	0
推進	21	0	0	0
事業	20	2	0	0
学府	15	0	0	0
基幹	15	0	0	0
育成	14	2	0	0
リーディング	12	0	0	0
博士	12	0	0	0
本学	12	4	0	0
様々	12	4	0	0
記念	10	6	0	0
人材	10	4	0	0
椎木	10	10	0	0
アクティブ・ラーナー	9	0	0	0
アジア	9	0	0	0
スタート	9	0	0	0
課程	9	0	0	0
企業	9	0	0	0
高い	9	0	0	0
最近	9	0	0	0
省	9	0	0	0
世代	9	0	0	0
文部	9	0	0	0
要請	9	0	0	0
理工	9	0	0	0

1.4テキストマイニングの主な方法

1.4.3. テキストの特徴分析

対応分析（correspondence analysis）
→変数の圧縮＋視覚化

テキストの種類（東大、京大、九大×2）と単語を対応付けさせ、それぞれの特徴を把握することが可能となる

対応分析の前処理

- 名詞頻度のクロス集計表を作成

	Kyudai_H25	Kyudai_H26	Kyodai_H26	Todai_H26
アクティブ・ラーナー	3	0	0	0
アジア	3	0	0	0
エネルギー	0	4	0	0
カルチュア	0	0	4	0
キャンパス	6	10	0	0
グローバル	6	0	0	0
スタート	3	0	0	0
ステージ	0	3	0	0
ストレス	0	19	0	0
テーマ	0	0	3	0
プロ	0	0	3	0
プログラム	9	0	0	0
リーダー	3	0	0	0
リーディング	4	0	0	0
安田	0	0	0	11
伊都	4	7	0	0
意見	0	0	4	0
意味	0	0	0	5

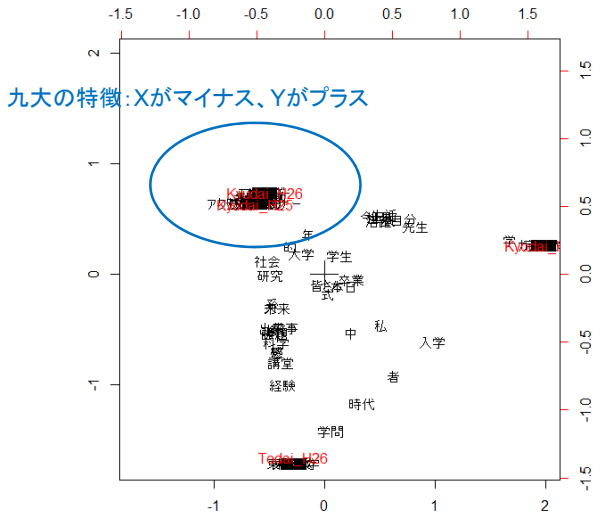
- データのクリーニング

→こと、もの、よう、などひらがなの名詞を削除

Rで対応分析を行う【参考】

- Library(MASS)
- greetings <- read.table("clipboard", header=TRUE)
- greetings.corresp=corresp(greetings, nf=3)
- plot(greetngs.corresp)

対応分析の結果(plot)



九州大学

対応分析の結果(固有値)

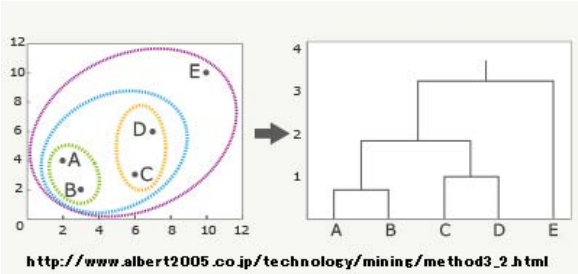
	[,1]	[,2]	[,3]	九大特徴
アクティブ・ラーナー	−0.76736	0.809275	1.504989	○
アジア	−0.76736	0.809275	1.504989	○
グローバル	−0.76736	0.809275	1.504989	○
スタート	−0.76736	0.809275	1.504989	○
プログラム	−0.76736	0.809275	1.504989	○
リーダー	−0.76736	0.809275	1.504989	○
リーディング	−0.76736	0.809275	1.504989	○
育成	−0.76736	0.809275	1.504989	○
課程	−0.76736	0.809275	1.504989	○
開催	−0.76736	0.809275	1.504989	○
学府	−0.76736	0.809275	1.504989	○
企業	−0.76736	0.809275	1.504989	○
基幹	−0.76736	0.809275	1.504989	○
基本	−0.76736	0.809275	1.504989	○
期待	−0.76736	0.809275	1.504989	○
記	−0.76736	0.809275	1.504989	○
牽引	−0.76736	0.809275	1.504989	○
最近	−0.76736	0.809275	1.504989	○
姿	−0.76736	0.809275	1.504989	○
支援	−0.76736	0.809275	1.504989	○
事業	−0.76736	0.809275	1.504989	○
自律	−0.76736	0.809275	1.504989	○

九州大学

1.4.テキストマイニングの主な方法

1.4.4. テキストのグループ分け

→クラスター分析



1.4.5. テキストの自動分類

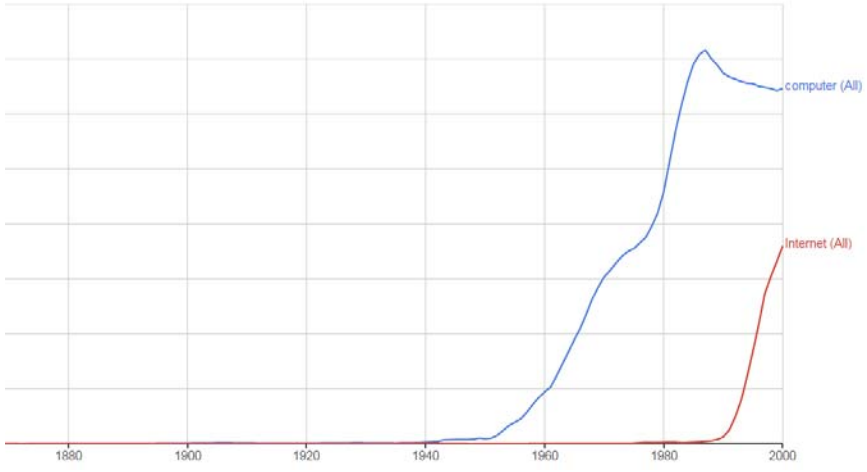
→ランダムフォレスト、SVMなど。cf. 中藤先生のご発表

1.4.6. テキストの時系列分析

→Google Books Ngram Viewer

九州大学

Google Books Ngram Viewer: computer vs Internet

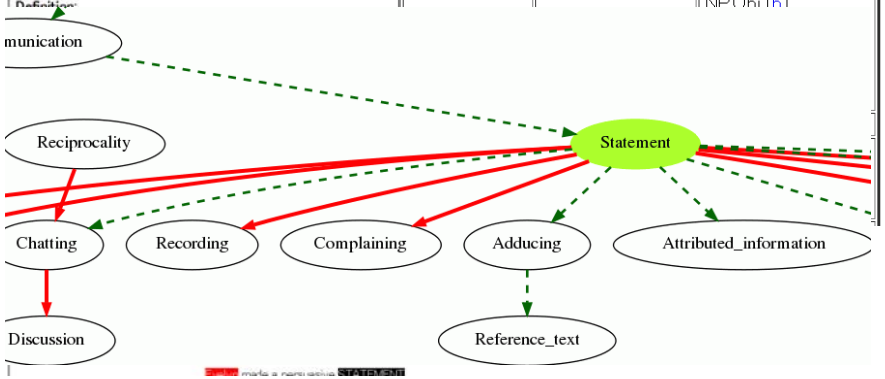


九州大学

1.5 意味解析と辞書

→WordNet, FrameNet

Frame Element	Number Annotated	Realization(s)
Addressee	(4)	PP[to] Dep (4)
Manner	(8)	AVP Dep (6) PP[with] Dep (2)
		INI.-- (3)



Thank you for your attention!

【発表者】
内田　諭
言語文化研究院