コーパスから見えることば・文化・社会　　第6回 2017/11/14 担当：望月 源

「日本語：数億語のコーパスを作って調べるとみえてくる頻出語,頻出表現」

1. コーパスとは？　（おさらい）

私たちが普段使っている生の言葉を大量に集めた言語データ

- 新聞記事,雑誌記事,小説, 教科書,Web ページ... 書き言葉コーパス（written language corpus）
- 会話,講演, TV 字幕... 話し言葉コーパス（spoken language corpus）
- blog, twitter, SNS, 製品レビュー...中間的なコーパス

→大規模に収集した言語データ（コーパス）を調査すると，言語の特徴が見えてくる（はず）

機械可読(コンピュータで扱える形)の形式であることが最大の利点

- 検索が楽... コーパスの中である「語」や［表現］などが出現する場所をすばやく突き止めることができる.
- 語の統計データなどを取りやすい
  - ある語が何回出現しているか（出現頻度）
  - ある語とよく一緒に使われる語（共起語）
  - どのくらいよく一緒に使われるか（共起頻度）

  →集計したり一覧表にしたり，確率などの計算が可能になりさまざまに応用可能
- 大量のデータを相手にしても比較的容易に処理ができる.

  例：　今日の新聞記事の中で「車」という文字が何回でてくるか？

  手作業で調べようと思ったら気が遠くなるが，コンピュータの力を借りれば作業は楽.

  では，一年分の記事に「車」が何回出てくるか？

  手作業ではちょっと無理を感じるが，コンピュータの力を借りれば，十分可能な作業

  ちなみに，毎日新聞 1995 年には「車」は 7085 回，2000 年は 7690 回.

2. コーパスの入手

- 既存のコーパスを利用
  - 研究目的利用での無償提供されたコーパス

    国立国語研究所提供のコーパス　（http://pj.ninjal.ac.jp/corpus_center/）

    Wikipedia の公開データ
  - レビューなど研究用に提供されているデータ

    楽天データ（https://rit.rakuten.co.jp/data_release_ja/）

ライブドアデータ（https://github.com/livedoor）
リクルートオープンデータ（http://atl.recruit-tech.co.jp/opendata/）
など.

- 市販されているコーパスを購入
  新聞記事データ（毎日新聞，朝日新聞，読売新聞，日本経済新聞...）　非常に高価
- パブリックドメインのテキスト集を利用
  青空文庫（http://www.aozora.gr.jp/）を利用

既存のコーパスを利用すると
○まとまったデータが比較的容易に利用可能
○著作権処理がされているので使用しやすい
×有料だと高価なことも
×自分の目的に合う内容だとは限らない

- 自分でコーパスを作成して利用
  例： WWW ページ，　TV 字幕

3. WWW のホームページをコーパスにする

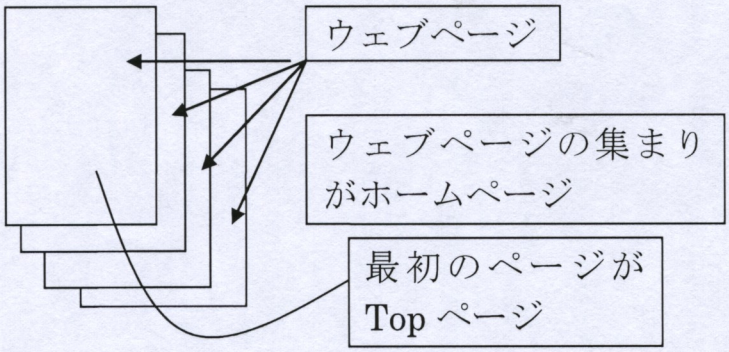
www の利点は，大量に存在する，ただで利用できる（研究目的）

- 作成手順
  1. 元になるデータの収集（web から収集することを web スクレイピングや web クローリングという）
  2. 収集したデータの加工
  3. 単語や品詞などの言語的情報の付加，整備

WWW (World Wide Web)の仕組みについて

- インターネット上に分散している情報やサービスをハイパーリンクで関連付けたもの
- アンカー（リンクの付いた語句）やアイコンなどをクリックするだけで情報にアクセス
- WWW に関連するアプリケーション→　Web ブラウザ（インターネットエクスプローラ，Google Chrome, FireFox，Opera など）

Web ページの構造



- 文字のみからなるファイルがベースとなる.
  - ページ内で使われている画像，音声，プログラムなどの情報は文字で書かれている
- ベースとなるファイル
  - HTML（HyperText Markup Language） という言語で書かれている HTML ファイル. 例： index.html
- HTML ファイルは HTML というマークアップ言語で書かれている.
  - マークアップ言語... テキスト中に記号（タグ）を挿入することで，語や文などに属性を与える言語規格のこと．代表例：HTML, SGML, TEX, XML など
  - HTML では，＜と＞で囲まれた記号を挿入する.
    例：見出しは<h1>見出し</h1>
    <h1> …… </h1> こういうのを HTML タグと呼ぶ.

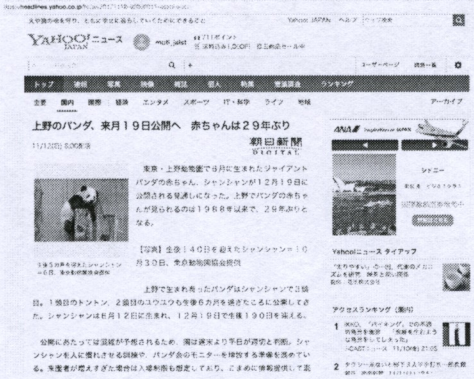
Web ページを集める

- URL を頼りにページデータを連続的に呼び出して，ファイルとして保存する
  - たくさん集めてコーパスの元データにすることが可能
  - ただし，html のタグがついているので収集した元データを加工する必要がある
  - サイトごとに html のタグの使い方は異なるので，個別に判断する必要がある.

サイトのページを自動的に収集するプログラムを利用することが多い(例 wget や, curl とか)

Web ページを加工してテキスト部分を取り出す

その前に www のデータがどんなものか確認 →web ブラウザでソースの表示
(yahoo ニュースの例)



右クリックしてページのソースを表示


```
保護された通信 view-source:https://headlines.yahoo.co.jp/hl?a=20171112-00000011-asahi-soci
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html lang="ja"
   xmlns:og="http://ogp.me/ns#"
   xmlns:fb="http://ogp.me/ns/fb#">
<head prefix="article: http://ogp.me/ns/article#">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<meta http-equiv="Content-Style-Type" content="text/css">
<meta http-equiv="Content-Script-Type" content="text/javascript">
<title>上野のパンダ、来月19日公開へ 赤ちゃんは29年ぶり（朝日新聞デジタル） - Yahoo!ニュース</title>
<meta name="description" content="東京・上野動物園で6月に生まれたジャイアントパンダの赤ちゃん、シャンシャンが1">
<meta name="keywords" content="ニュース,国内,社会">
<meta name="robots" content="noarchive">
<link type="text/css" rel="stylesheet" href="https://s.yimg.jp/images/jpnews/cre/common/pc/css/common_pc_v2.css?20170130" media="all">
<link rel="stylesheet" href="https://s.yimg.jp/images/jpnews/cre/article/pc/css/article_pc_v3.css?20170817" type="text/css" media="all">
<script type="text/javascript" src="https://s.yimg.jp/images/ds/rapid/1.5.0/ult.inc.js"></script>
<meta property="al:ios:app_name" content="Yahoo!ニュース" />
<meta property="al:ios:app_store_id" content="407906756" />
<meta property="al:ios:url" content="yjtrend://d/hdl/20171112-00000011-asahi-soci" />
<meta property="al:android:app_name" content="Yahoo!ニュース" />
<meta property="al:android:package" content="jp.co.yahoo.android.news" />
<meta property="al:android:url" content="yjnews://d/hdl/20171112-00000011-asahi-soci" />
<meta property="og:type" content="article" />
<meta property="og:title" content="上野のパンダ、来月19日公開へ 赤ちゃんは29年ぶり（朝日新聞デジタル） - Yahoo!ニュース" />
<meta property="og:description" content="東京・上野動物園で6月に生まれたジャイアントパンダの赤ちゃん、シャンシャンが12 - Yahoo!ニュース(朝日新聞デジタル)" />
<meta property="og:image" content="https://lpt.c.yimg.jp/amd/20171112-00000011-asahi-000-view.jpg" />
<meta property="og:image:width" content="640" />
<meta property="og:image:height" content="424" />
<meta property="og:url" content="https://headlines.yahoo.co.jp/hl?a=20171112-00000011-asahi-soci"/>
<meta property="og:site_name" content="Yahoo!ニュース">
<meta property="og:locale" content="ja_JP" />
<meta property="fb:app_id" content="276725822409153"/>
<meta name="twitter:card" content="summary_large_image">
<meta name="twitter:title" content="上野のパンダ、来月19日公開へ 赤ちゃんは29年ぶり（朝日新聞デジタル） - Yahoo!ニュース">
<meta name="twitter:description" content="東京・上野動物園で6月に生まれたジャイアントパンダの赤ちゃん、シャンシャンが12 - Yahoo!ニュース(朝日新聞デジタル)">
<meta name="twitter:image" content="https://lpt.c.yimg.jp/amd/20171112-00000011-asahi-000-view.jpg">
<meta name="twitter:image:width" content="640">
<meta name="twitter:image:height" content="424">
<meta name="twitter:site" content="@YahooNewsTopics">
<meta name="twitter:app:country" content="jp">
<meta name="twitter:app:name:iphone" content="Yahoo!ニュース" />
<meta name="twitter:app:id:iphone" content="407906756" />
```


www のページは HTML (HyperText Markup Language)という言語で書かれている
その中に，コーパスのデータとなる内容が含まれている
該当部分を取り出す→該当部分を特定する→プログラムを書いて該当部分を取り出す

- 元データのソースを読むと...
  - 　からニュース本文が 始まる
  - <h1> 見出し </h1> として書かれている
  - 記事本文は， の中の に記述されている
  - より厳密には，<p class= "ynDetailText"> の中に記述されている
  - <!-- /.paragraph -->が記事の終了箇所と一致する.

  などの手がかりを得て，プログラムを書く.

単語や品詞などの言語的情報の付加，整備

- 取り出したテキストの中にどんな単語があるか，どんな品詞があるかなどを解析して，情報を付与する．こうした情報がつくことでコーパスの使い道が広がる．
- 単語の取り出しには，形態素解析という処理をする
  - 日本語では，mecab，茶筌，JUMAN の 3 つが特に有名

4. TV 字幕からのコーパス作成

- 日本の地上波デジタルテレビ放送は字幕放送に対応(東京では全体の約 60%)
- EPG(Electronic Program Guide)電子番組表で [字]のマークが付いているのが字幕放送

作成手順

1. EPG（番組表）データを自動的に更新し，[字]の番組を自動的に録画予約
2. 予約された番組を実際に録画
3. 定期的に録画データから字幕データを作成

2 台のサーバを 24 時間毎日動かしつづけて記録

- ASS という形式の字幕データの元ファイルが抽出されるので，そのファイルを加工して，TV 字幕コーパスを作成している．

1	0:00:02.95	0:00:11.64	マサラタウンに向かう サトシたちは最初の島に到着しようとしていた>(汽笛)
2	0:00:11.64	0:00:14.91	(サトシ)お～ 見えてきたぜ!
3	0:00:11.64	0:00:14.91	(ピカチュウ)ピカピカチュ～!
4	0:00:14.91	0:00:18.62	(アイリス)何ていう島ですか?
5	0:00:14.91	0:00:18.62	(パーカー)ハニー島でございます。
6	0:00:18.62	0:00:20.99	(デント)ハニー島?
7	0:00:18.62	0:00:20.99	(パーカー)はい。
8	0:00:20.99	0:00:25.77	ミツハニーが集める 甘い蜜ハニーミツで 有名な島でございます。
9	0:00:25.77	0:00:27.77	ミツハニー。
10	0:00:25.77	0:00:27.77	ピカ?

- 規模の話
  - NHK, e テレ, TBS, NTV, Fuji, TV Asahi, TV Tokyo
    - 東京のキー局 7 局の字幕放送は 月間約 4,200
  - 2012 年 12 月から収集開始
    - 2017 年 3 月の時点で 約 232,000 番組分の蓄積
    - 約 89,000,000 文のデータ

    さて，この中に何単語があるでしょうか？

- テレビでよく使われる語は何か？
  - 単語頻度統計
- テレビによく出てくる表現は何か？
  - 定型表現 (formulaic sequences)の出現
  - 単語レベルの *n*-gram を作って数を数える

  実際に数えて，集計してみます．想像と比較してみましょう．

- *n*-gram：1 単語ずつではなく，複数(n 個)の単語単位で出現頻度を数えて統計をとる

単語レベルの n-gram という考え方．例：

1-gram 幾重にも

2-gram 幾重にも 広がる

3-gram 幾重にも 広がる 芳じゅん

4-gram 幾重にも 広がる 芳じゅん な

5-gram 幾重にも 広がる 芳じゅん な 香り

…

- コーパス全体でお互いに関連の強い語や表現を探す
  - word2vec (https://code.google.com/archive/p/word2vec/)
    - ✧ 非常に有名な自然言語処理ツール
    - ✧ コーパスをデータとして使って，各単語をベクトル表現
    - ✧ distance ツールを使うと，与えられた単語と非常に関係の高い単語を計算することができる

「こんにちは」に近い表現の例

Word	Cosine distance
こんばんは	0.796703
お疲れさまです	0.778086
おかえりなさい	0.762287
おかえり	0.722781
ありがとうございます	0.694695
ごめんください	0.684179
さようなら	0.677760
お邪魔します	0.667552
はーい	0.663395
ごきげんよう	0.654722
おはよう	0.642613
すいませんありがとうございます	0.642137
さようでございますか	0.639013
お疲れさまでした	0.637674
ご苦労さま	0.637072
そういえば…	0.634369
どういたしまして	0.629217
少々お待ちくださいませ	0.628805
ただいま…	0.628740
バイバ～イ	0.628297
いただきまーす	0.626046
はじめまして	0.624270