Fundamentals — from data to visualisation

Big Data Business Academy

Finn Årup Nielsen

DTU Compute
Technical University of Denmark

September 21, 2016

Getting my hands dirty with:

DBC library loan data.

Twitter retweet study.

Library information.

Art depictions data mining.

Danish Business Authority (Erhvervsstyrelsen).

Wikipedia citations mining.

Using tools such as: Python, Perl, R, sklearn, statsmodels, Matplotlib, D3, command-line, Semantic Web, Wikidata, Wikipedia.

Example: Library loans data

Library loans data

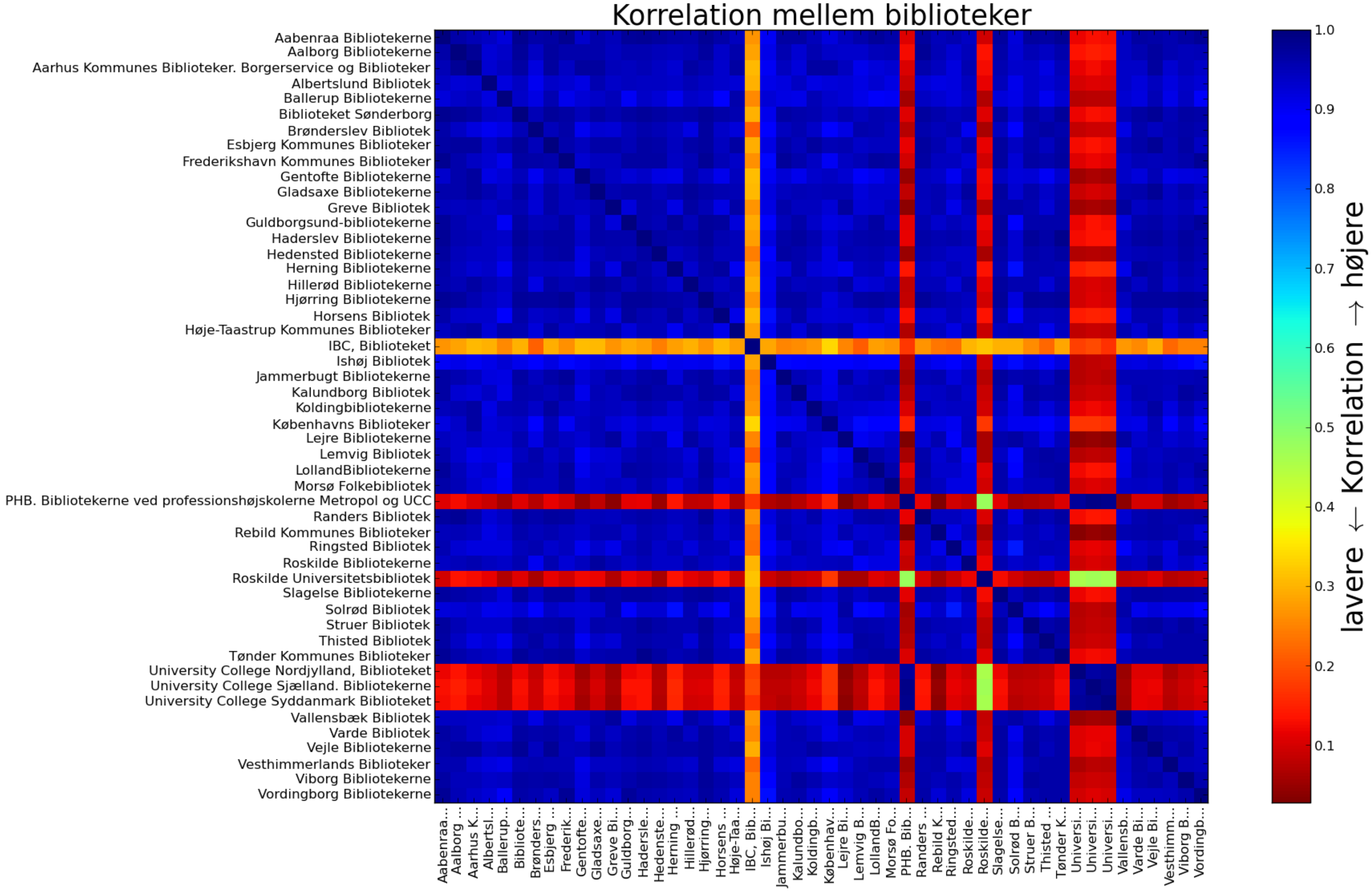
47 million loan data collected from Danish library users by DBC ("Dansk Bibliotekscenter").

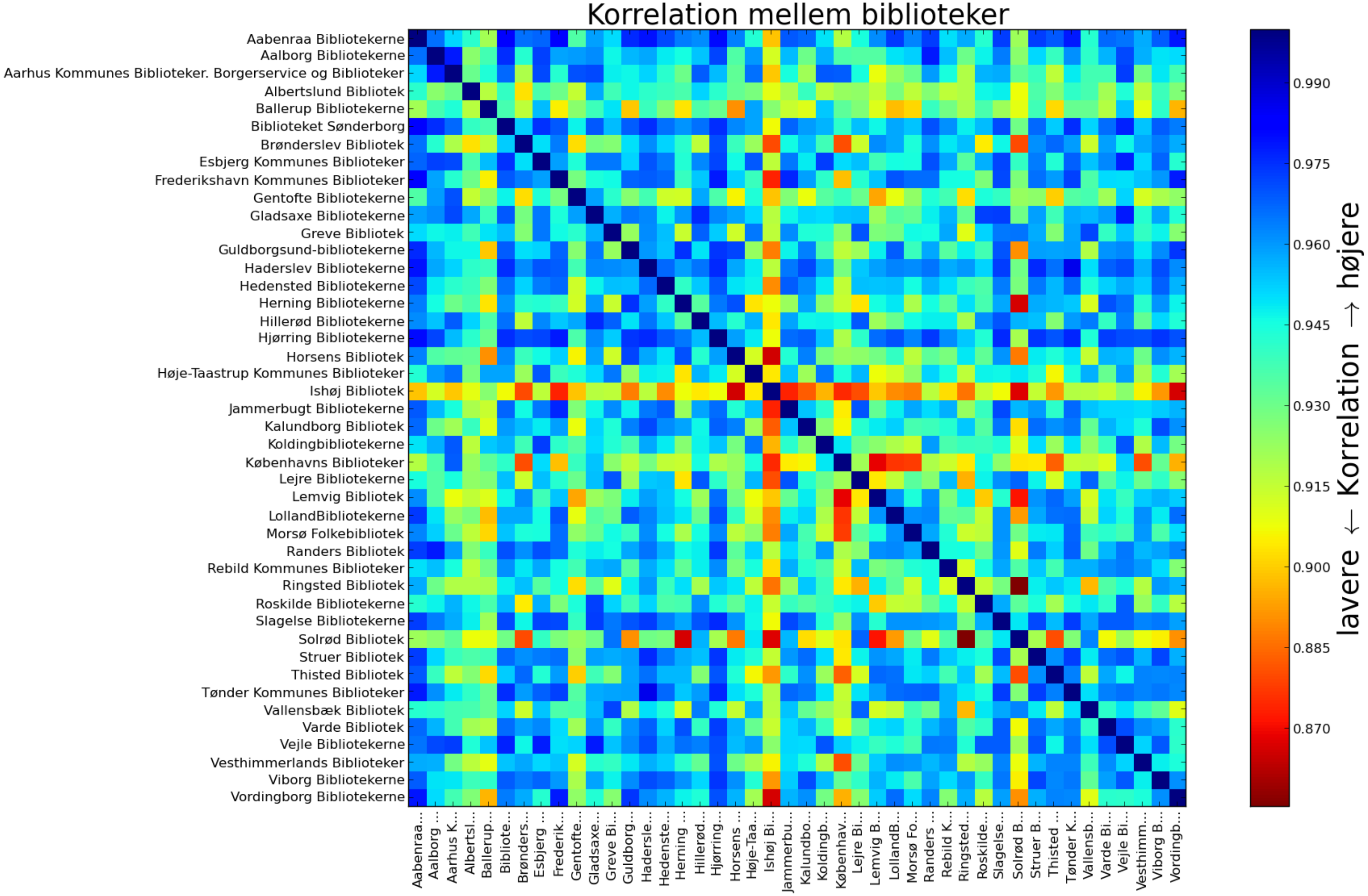
Anonymized structured data in the format of comma-separated values dataset with name with the size of 5.8 GB: One loan, one line.

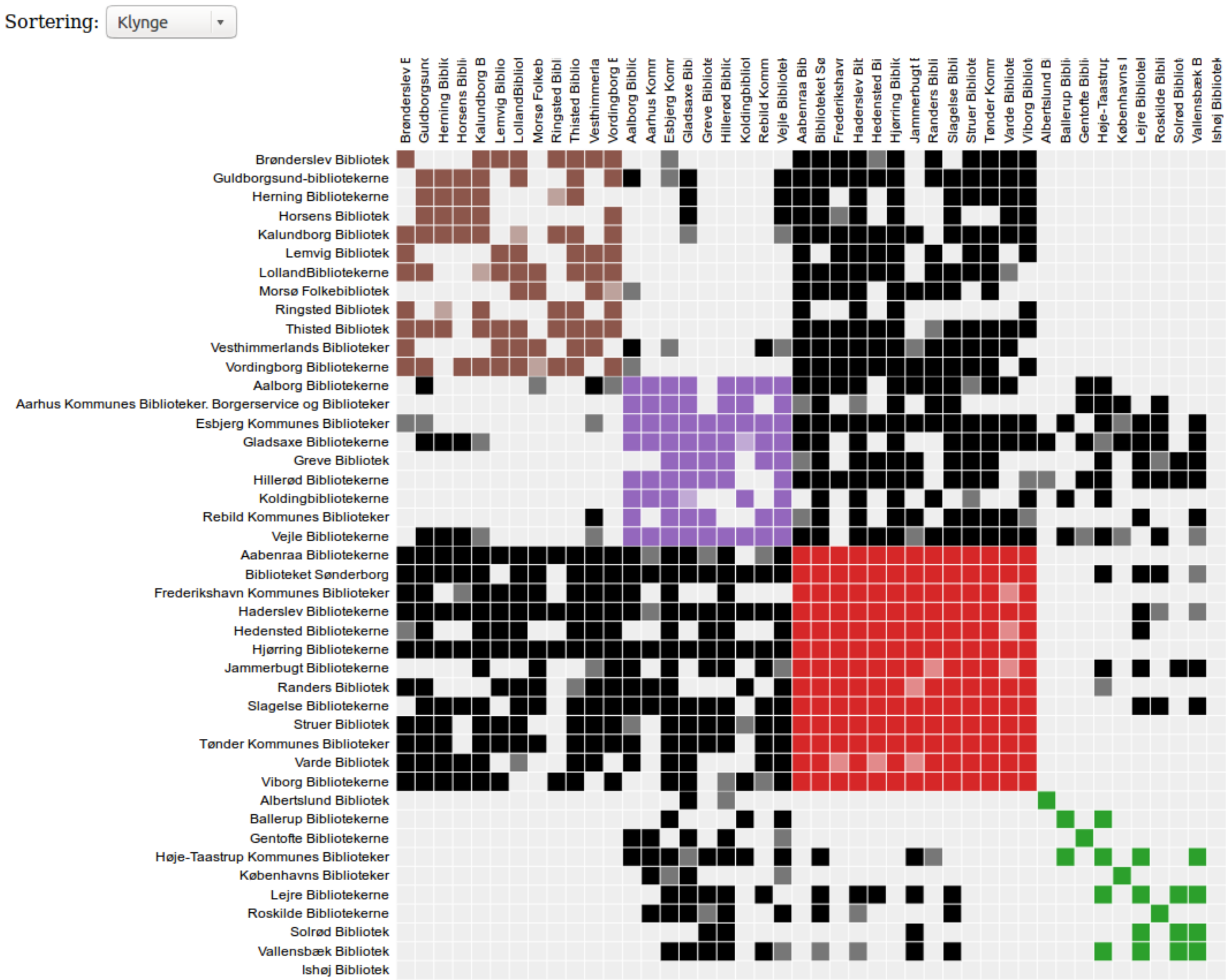
Extraction of title words wrt. to each of the 50 library system ("biblioteksvæsen", e.g., municipality). Streaming processing over lines in 5 to 10 minutes to build:

Medium-sized data matrix of size words-by-library-system.

(with help from David Tolnem and Søren Vibjerg at HACK4DK)


Korrelation mellem biblioteker
Aabenraa Bibliotekerne
Aalborg Bibliotekerne
Aarhus Kommunes Biblioteker. Borgerservice og Biblioteker
Albertslund Bibliotek
Ballerup Bibliotekerne
Biblioteket Sønderborg
Brønderslev Bibliotek
Esbjerg Kommunes Biblioteker
Frederikshavn Kommunes Biblioteker
Gentofte Bibliotekerne
Gladsaxe Bibliotekerne
Greve Bibliotek
Guldborgsund-bibliotekerne
Haderslev Bibliotekerne
Hedensted Bibliotekerne
Herning Bibliotekerne
Hillerød Bibliotekerne
Hjørring Bibliotekerne
Horsens Bibliotek
Høje-Taastrup Kommunes Biblioteker
IBC, Biblioteket
Ishøj Bibliotek
Jammerbugt Bibliotekerne
Kalundborg Bibliotek
Koldingbibliotekerne
Københavns Biblioteker
Lejre Bibliotekerne
Lemvig Bibliotek
LollandBibliotekerne
Morsø Folkebibliotek
PHB. Bibliotekerne ved professionshøjskolerne Metropol og UCC
Randers Bibliotek
Rebild Kommunes Biblioteker
Ringsted Bibliotek
Roskilde Bibliotekerne
Roskilde Universitetsbibliotek
Slagelse Bibliotekerne
Solrød Bibliotek
Struer Bibliotek
Thisted Bibliotek
Tønder Kommunes Biblioteker
University College Nordjylland, Bibliotekeт
University College Sjælland. Bibliotekerne
University College Syddanmark Biblioteket
Vallensbæk Bibliotek
Varde Bibliotek
Vejle Bibliotekerne
Vesthimmerlands Biblioteker
Viborg Bibliotekerne
Vordingborg Bibliotekerne
Aabenraa...
Aalborg ...
Aarhus K...
Albertsl...
Ballerup...
Bibliote...
Brønders...
Esbjerg ...
Frederik...
Gentofte...
Gladsaxe...
Greve Bi...
Guldborg...
Hadersle...
Hedenste...
Herning ...
Hillerød...
Hjørring...
Horsens ...
Høje-Taa...
IBC, Bib...
Ishøj Bi...
Jammerbu...
Kalundbo...
Koldingb...
Københav...
Lejre Bi...
Lemvig B...
LollandB...
Morsø Fo...
PHB. Bib...
Randers ...
Rebild K...
Ringsted...
Roskilde...
Roskilde...
Slagelse...
Solrød B...
Struer B...
Thisted ...
Tønder K...
Universi...
Universi...
Universi...
Vallensb...
Varde Bi...
Vejle Bi...
Vesthimm...
Viborg B...
Vordingb...
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
lavere ← Korrelation → højere


Korrelation mellem biblioteker
Aabenraa Bibliotekerne
Aalborg Bibliotekerne
Aarhus Kommunes Biblioteker. Borgerservice og Biblioteker
Albertslund Bibliotek
Ballerup Bibliotekerne
Biblioteket Sønderborg
Brønderslev Bibliotek
Esbjerg Kommunes Biblioteker
Frederikshavn Kommunes Biblioteker
Gentofte Bibliotekerne
Gladsaxe Bibliotekerne
Greve Bibliotek
Guldborgsund-bibliotekerne
Haderslev Bibliotekerne
Hedensted Bibliotekerne
Herning Bibliotekerne
Hillerød Bibliotekerne
Hjørring Bibliotekerne
Horsens Bibliotek
Høje-Taastrup Kommunes Biblioteker
Ishøj Bibliotek
Jammerbugt Bibliotekerne
Kalundborg Bibliotek
Koldingbibliotekerne
Københavns Biblioteker
Lejre Bibliotekerne
Lemvig Bibliotek
LollandBibliotekerne
Morsø Folkebibliotek
Randers Bibliotek
Rebild Kommunes Biblioteker
Ringsted Bibliotek
Roskilde Bibliotekerne
Slagelse Bibliotekerne
Solrød Bibliotek
Struer Bibliotek
Thisted Bibliotek
Tønder Kommunes Biblioteker
Vallensbæk Bibliotek
Varde Bibliotek
Vejle Bibliotekerne
Vesthimmerlands Biblioteker
Viborg Bibliotekerne
Vordingborg Bibliotekerne
Aabenraa...
Aalborg ...
Aarhus K...
Albertsl...
Ballerup...
Bibliote...
Brønders...
Esbjerg ...
Frederik...
Gentofte...
Gladsaxe...
Greve Bi...
Guldborg...
Hadersle...
Hedenste...
Herning ...
Hillerød...
Hjørring...
Horsens ...
Høje-Taa...
Ishøj Bi...
Jammerbu...
Kalundbo...
Koldingb...
Københav...
Lejre Bi...
Lemvig B...
LollandB...
Morsø Fo...
Randers ...
Rebild K...
Ringsted...
Roskilde...
Slagelse...
Solrød B...
Struer B...
Thisted ...
Tønder K...
Vallensb...
Varde Bi...
Vejle Bi...
Vesthimm...
Viborg B...
Vordingb...
0.990
0.975
0.960
0.945
0.930
0.915
0.900
0.885
0.870
lavere ← Korrelation → højere



Interactive version

Summary: Library loan data

Fairly small "big data": No need for specialized big data tools.

Stream processing on the big data to get manageable medium-sized data.

Simple natural language processing: splitting, stopwords, counting

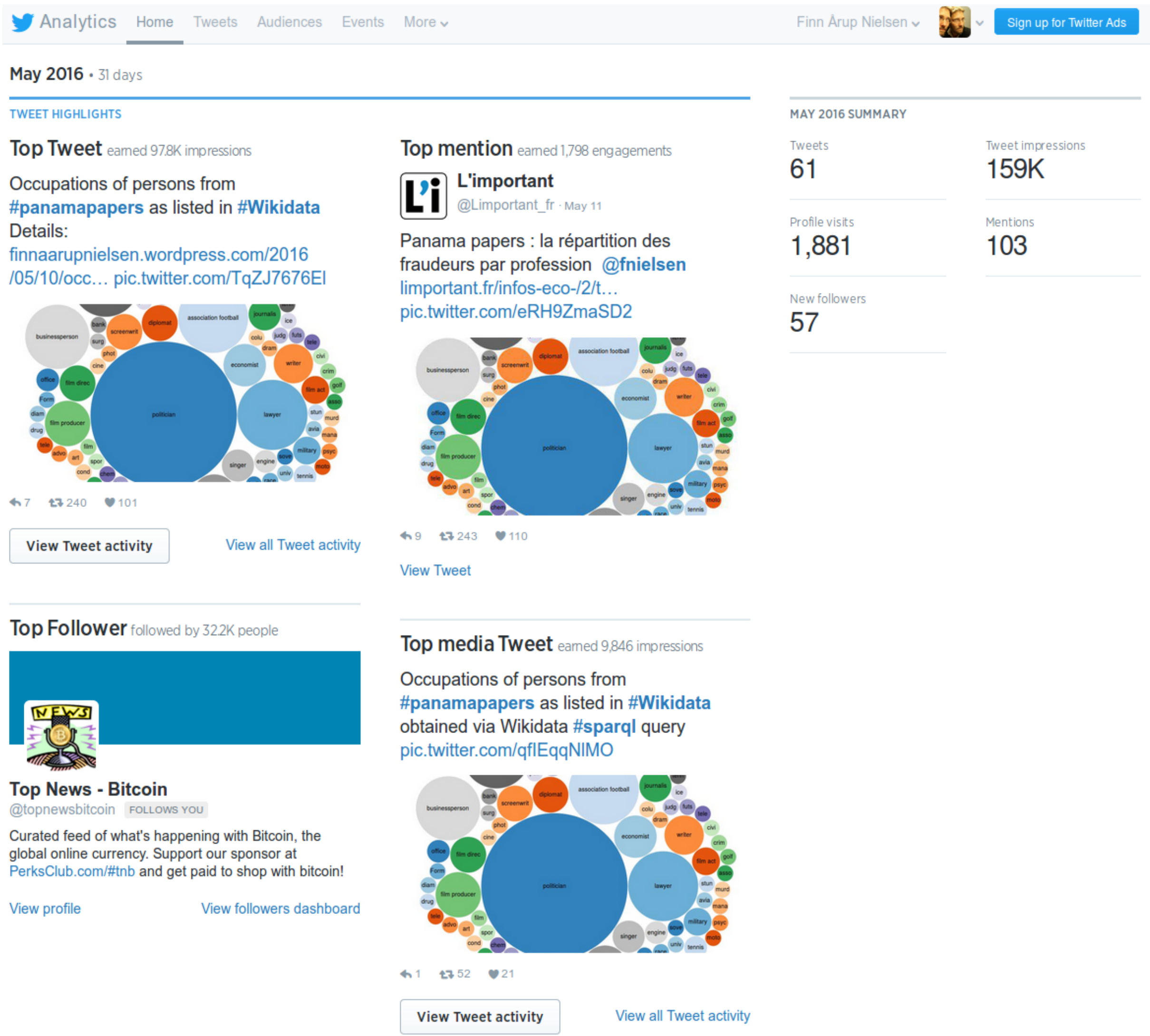
Little issue with feature processing. The analyzed data is count data of words.

One-shot research analysis with clustering and correlation analysis using standard Python tools: IPython Notebook, Pandas, sklearn, ...

Visualization with Python's Matplotlib and JavaScript's NVD3 and D3.

Example: Twitter retweet analysis

Twitter retweet analysis question



Research question: What determines whether a Twitter post will be retweeted?

"Good Friends, Bad News — Affect and Virality in Twitter" (Hansen et al., 2011)

Collect a lot of tweets, extract features, build statistical model and determine feature importance.

Twitter retweet analysis data


version2

DTU-forsker afkoder Twitter-beskeder med 1.200 linjer Python-kode

Twitter-beskeder og blog-indlæg har stor betydning for, hvordan virksomheders omdømme ser ud online. Danske forskere arbejder på at skabe et digitalt stemningsbarometer ud fra syndfloden af oplysninger online.

Mikkel Meister
Tirsdag, 29. december 2009 - 6:59

0

FØLG VERSION2

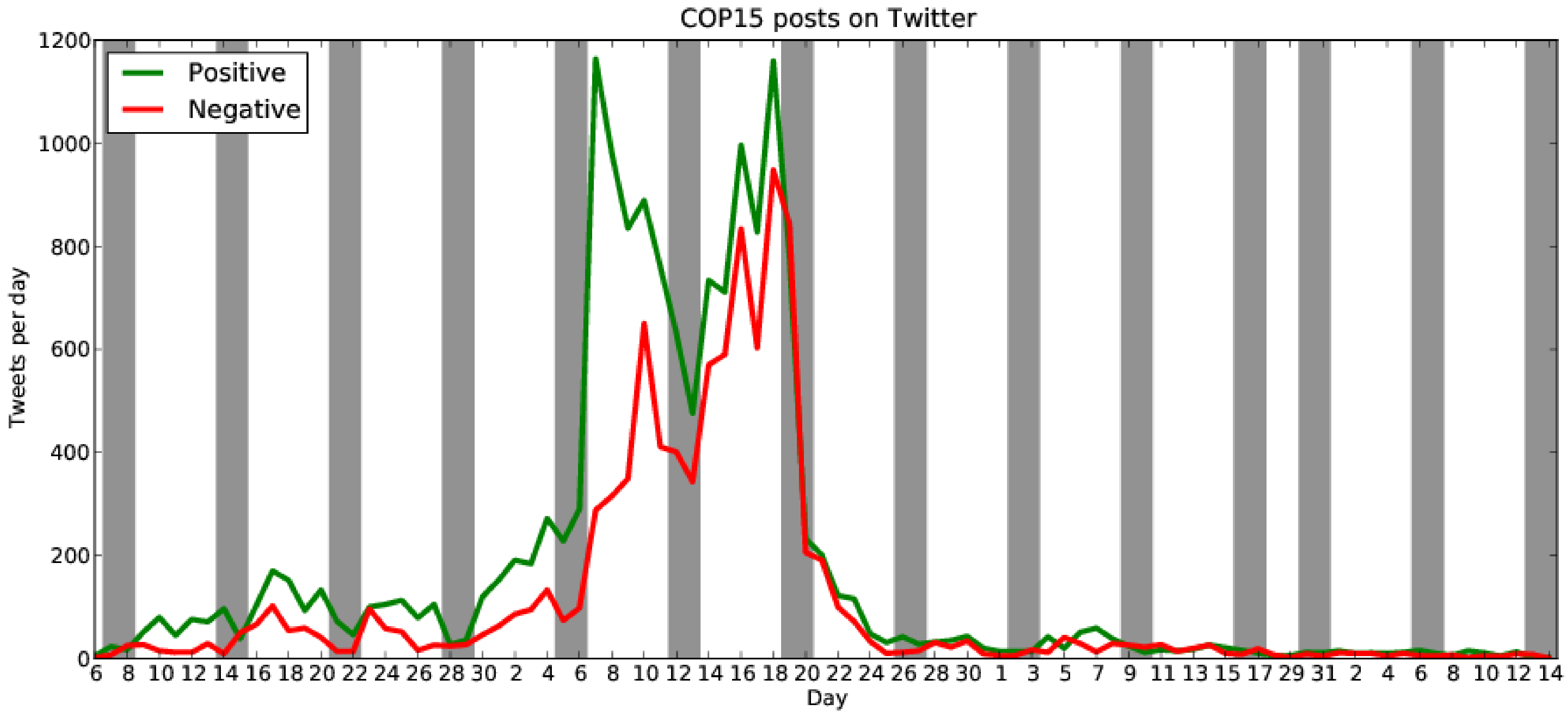
Collection of Twitter data in two ways:

1) Attach to streaming API and store the returned (unstructured) JSON data in the MongoDB nosql database. A one-liner!

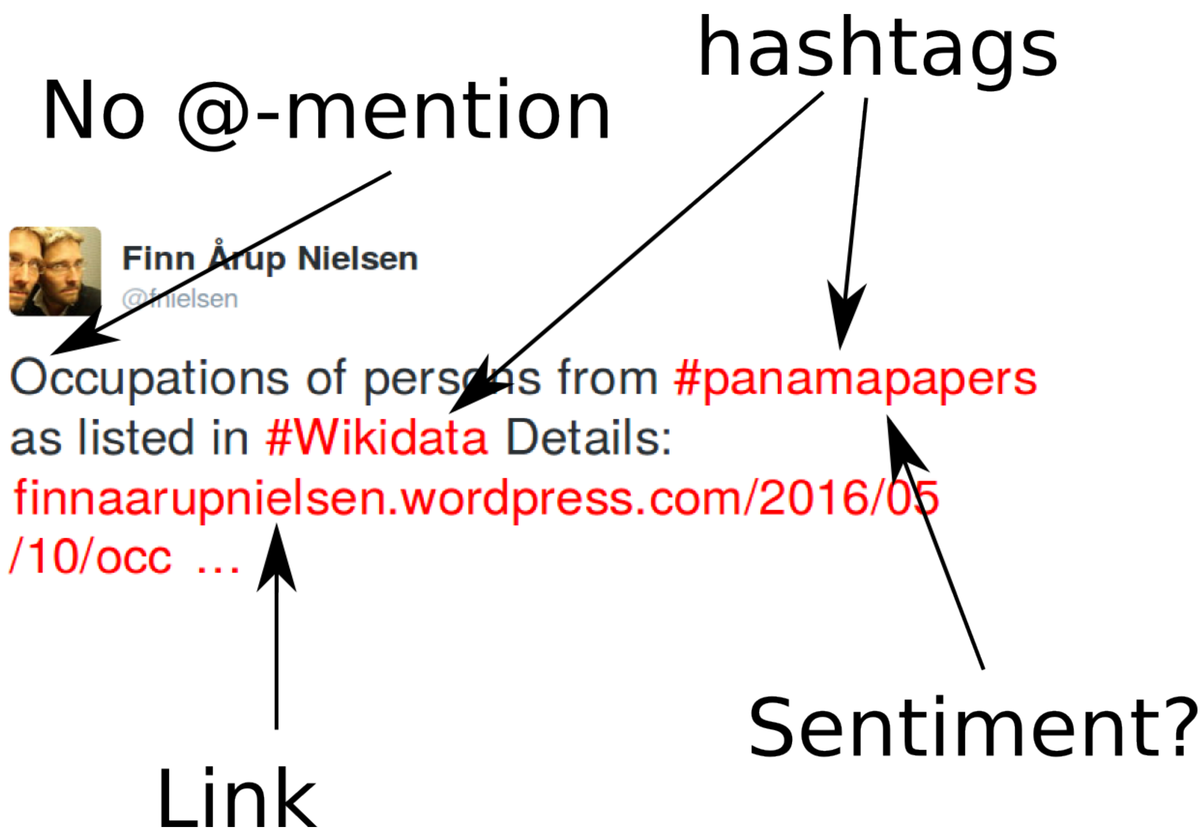
2) Query Twitter search API regularly searching on COP15.

Getting around half a million tweets.

Twitter sentiment through time



Twitter retweet analysis feature extraction



Extracted features:

Occurence of hash tag

Occurence of @-mention

Occurence of link

"Newsiness" from trained Naïve Bayes classifier

Sentiment via AFINN word list

Twitter retweet analysis summary

Stream processing for extraction of features written to a medium-sized comma-separated values file.

Twitter features analyzed with logistic regression over 100'000s tweets in R.

Investigated the interaction between newsiness and sentiment, particularly negative sentiment. An R one-liner.

Various statistical tests support that negative newsy tweets are retweeted more ("bad news is good news") as is positive non-news ("friends") tweets.

Example: Library information

Library information

DBC ("Dansk Bibliotekscenter") competition in 2015/2016.

"How can data science be used to provide library users with new and better experiences?"

Library information

DBC (“Dansk Bibliotekscenter”) competition in 2015/2016.

“How can data science be used to provide library users with new and better experiences?”

DBC made loan data available.

Recommendation system based on loan data?

Library information

DBC (“Dansk Bibliotekscenter”) competition in 2015/2016.

“How can data science be used to provide library users with new and better experiences?”

DBC made loan data available.

Recommendation system based on loan data?

1st and 3rd prize did that.

Library information

DBC ("Dansk Bibliotekscenter") competition in 2015/2016.

"How can data science be used to provide library users with new and better experiences?"

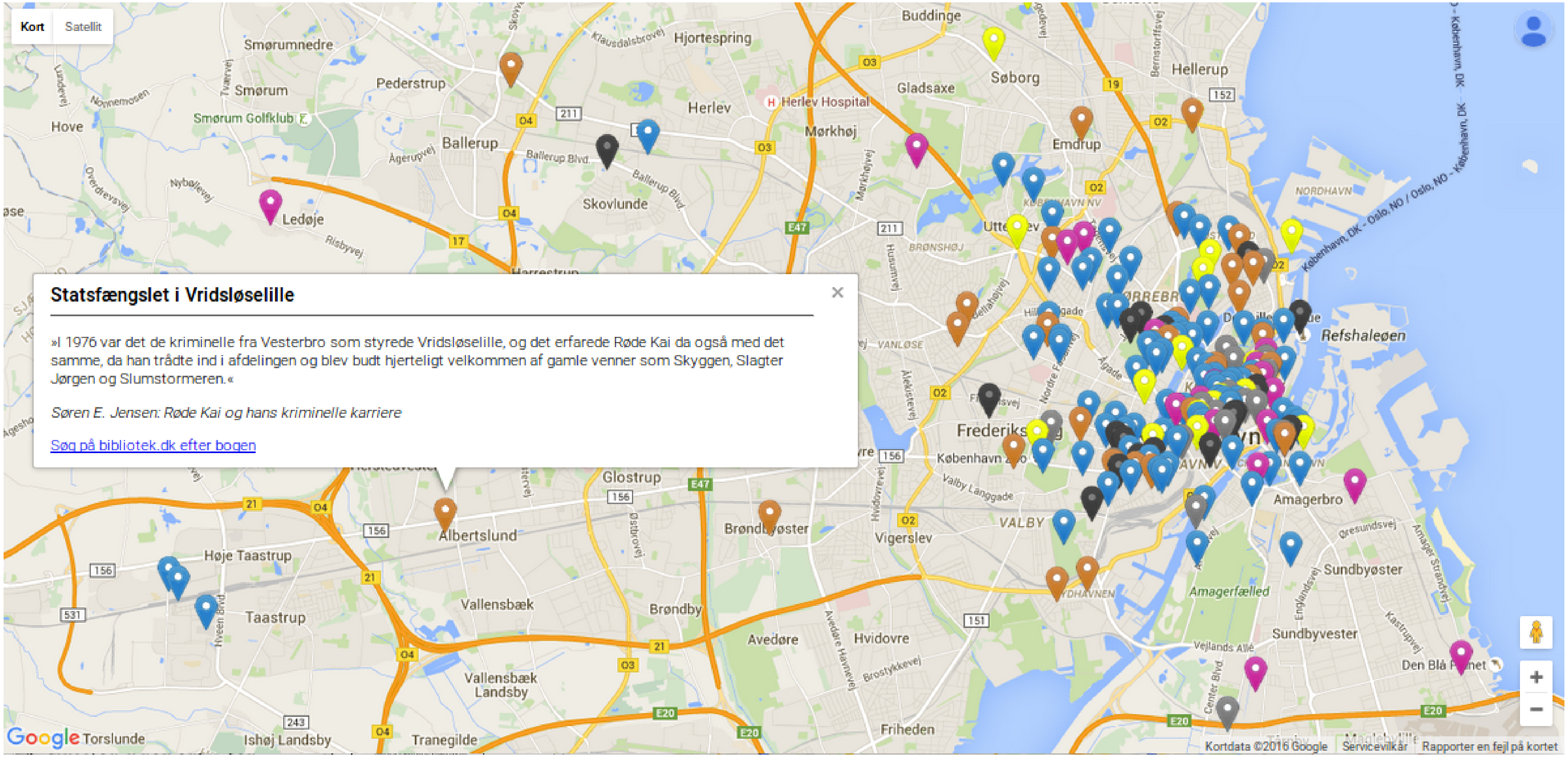
DBC made loan data available.

Recommendation system based on loan data?

1st and 3rd prize did that.

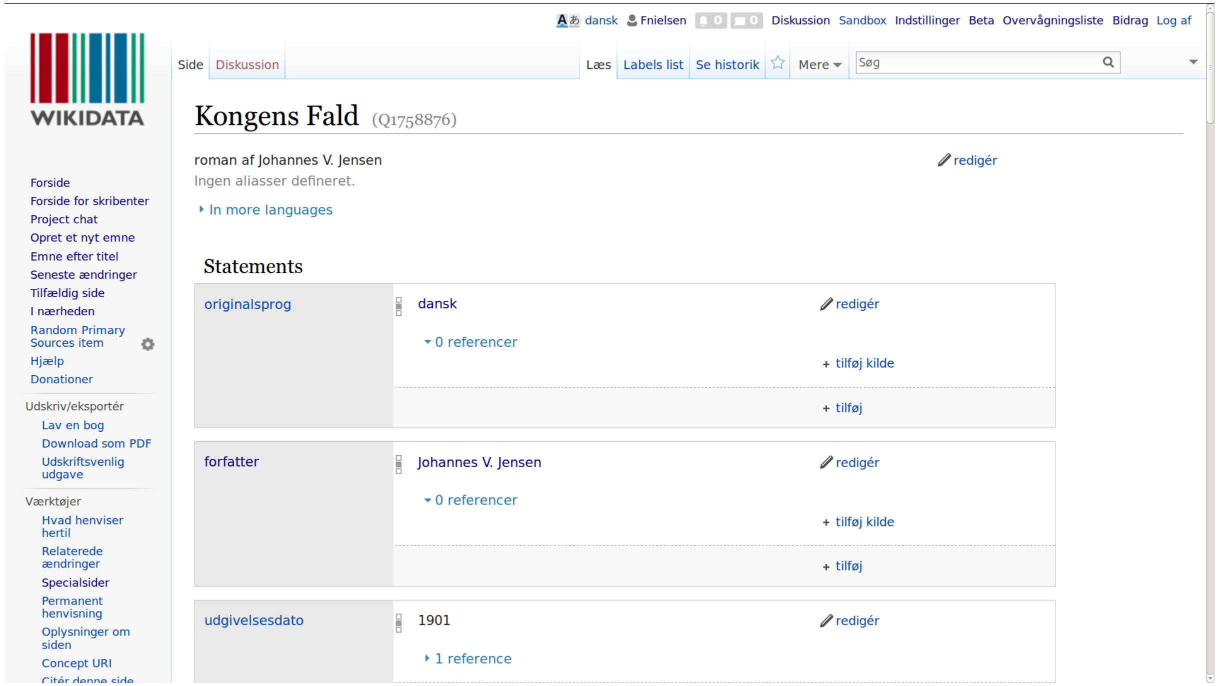
New approach to search library information via **geolocation**.

Littar



Geolocatable narrative locations from literary works in Wikidata plotted on a map available at http://fnielsen.github.io/littar.

So where is the data from? Wikidata!



Wikidata = Wikipedia's sister site with semi-structured data.

Over 20 million items. For instance, over 180'000 literary works.

Each may be described by one or more of over 2700 properties.

Crowdsourced from over 15'000 "active users" and a total of over 370 million edits.

Semantic Web: Example triples

Subject	Verb	Object
neuro:Finn	a	foaf:Person
neuro:Finn	foaf:homepage	http://www.imm.dtu.dk/~fn/
dbpedia:Charlie_Chaplin	foaf:surname	Chaplin
dbpedia:Charlie_Chaplin	owl:sameAs	fbase:Charlie Chaplin

Table 1: Triple structure

where the the so-called "prefixes" are

```
PREFIX foaf:    <http://xmlns.com/foaf/0.1/>
PREFIX neuro:   <http://neuro.imm.dtu.dk/resource/>
PREFIX dbpedia: <http://dbpedia.org/resource/>
PREFIX owl:     <http://www.w3.org/2002/07/owl#>
PREFIX fbase:   <http://rdf.freebase.com/ns/type.object.>
```

Semantic Web search engine

SPARQL search engines:

BlazeGraph (formerly called "Bigdata"), "supports up to 50 Billion edges on a single machine"

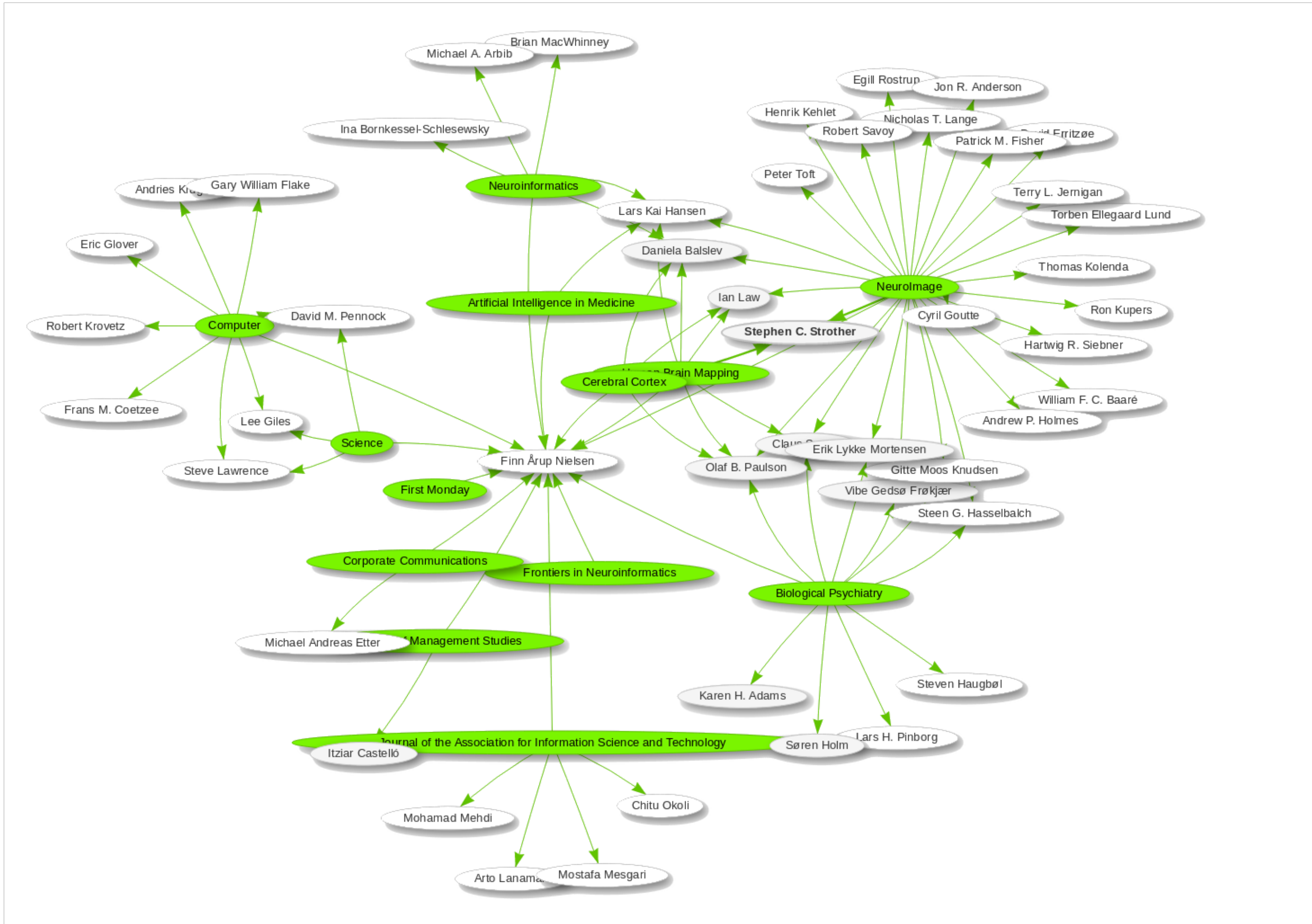
Virtuoso Universal Server from Openlink Software

Apache Jena

RDF4J/Sesame

The Wikidata Query Service presently uses BlazeGraph. It is available from https://query.wikidata.org and includes, e.g., graph and map visualizations.

Example query: coauthor-journal network



Example query: coauthor-journal network

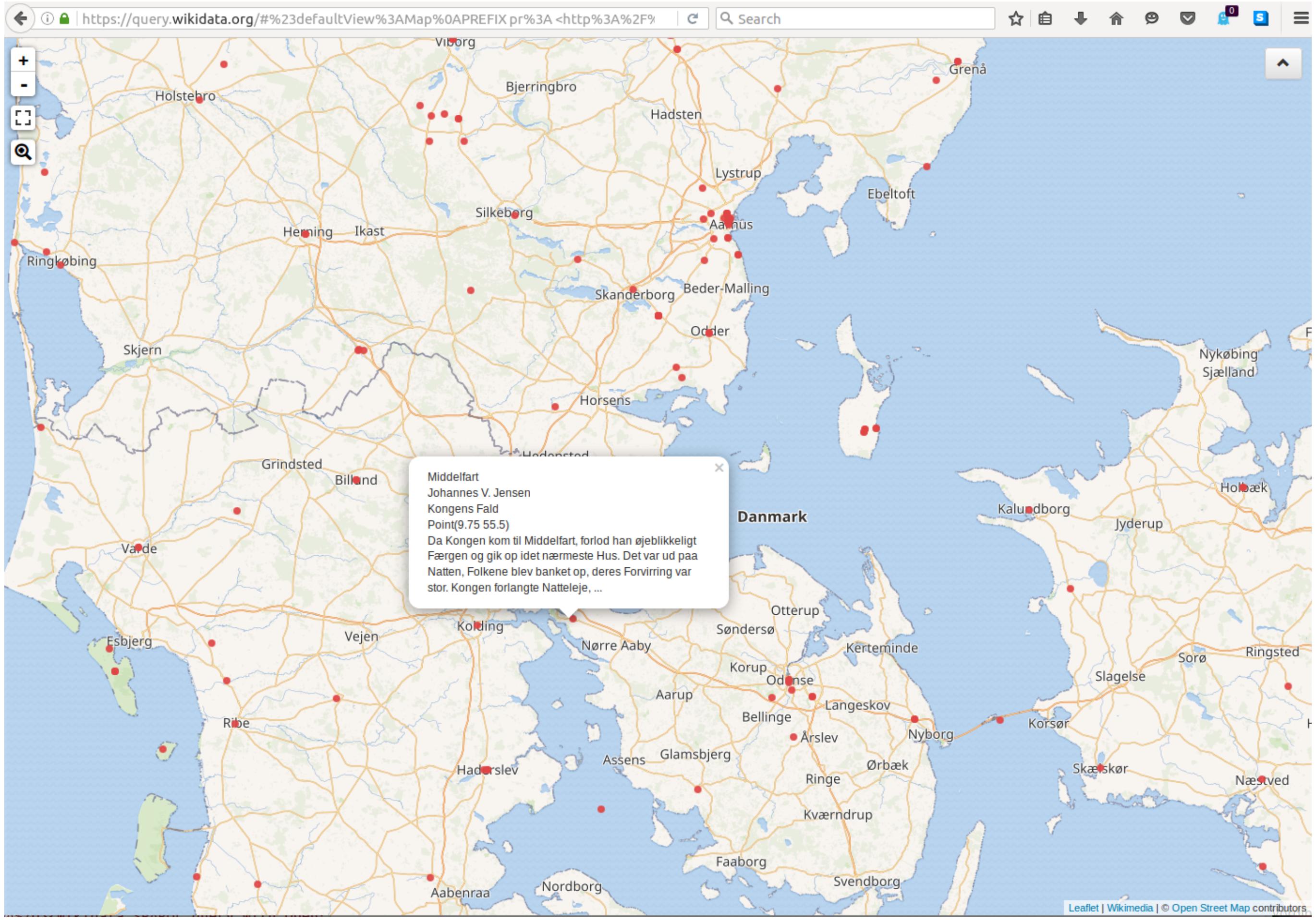
Query on Wikidata Query Service with graph visualization for data with scientific articles, their authors and journals over more than 100 million statements.

```
#defaultView:Graph
SELECT DISTINCT ?journal ?journalLabel
                (concat("7FFF00") as ?rgb)
                ?coauthor ?coauthorLabel

WHERE {
  ?work wdt:P50 wd:Q20980928 .
  ?work wdt:P50 ?coauthor .
  ?work wdt:P1433 ?journal .
  SERVICE wikibase:label {
    bd:serviceParam wikibase:language "en". }
}
```

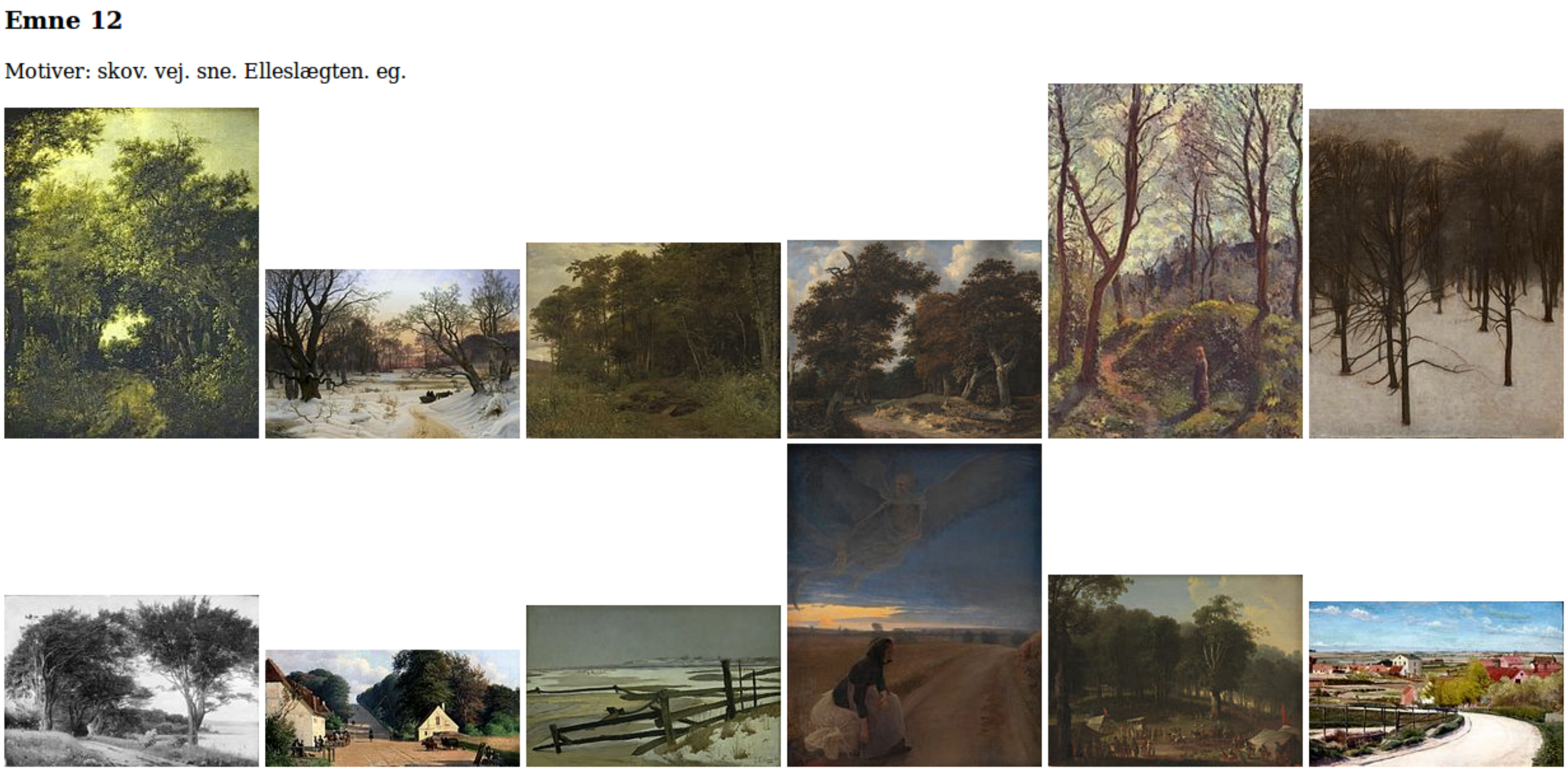
Try it! or one for drug-disease interaction (of Dario Taraborelli).

Example: Wikidata query on book data



Wikidata SPARQL query with OpenStreetMap and Leaflet map

One step further: Data mining Wikidata data



Unsupervised learning (Non-negative matrix factorization) on a 896-by-576-sized matrix of depictions in paintings as described on Wikidata.

Example: Company information

Company information for novelty detection

Extract features from 43 GB JSONL file from Erhvervsstyrelsen.

Feature: antal penheder, branche ansvarskode, nyeste antal ansatte, nyeste virksomhedsform, reklamebeskyttet, sammensat status, sidste virksomhedsstatus, stiftelsesaar.

Features imputed and scaled.

Novelty here: Distance from company to each cluster center after K-means clustering.

Technical: Python, Pandas, unsupervised learning with MiniBatchKMeans from Scikit-learn (sklearn) implemented in a Python module called cvr-miner and an IPython Notebook

Company information novelty

FIHINSEA-DENAMRK A/S

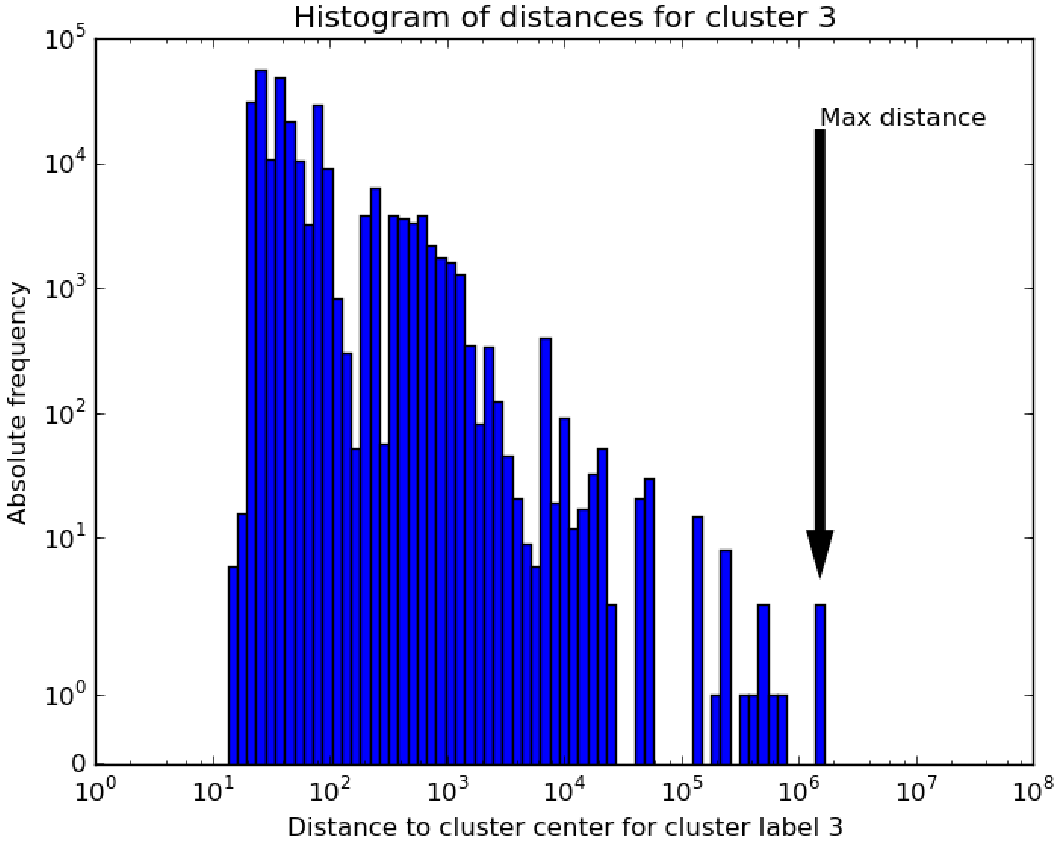
CVR-nummer	15706538
Adresse	Bådehavnsgade 48
Postnummer og by	2450 København SV
Startdato	30.10.2014
Virksomhedsform	Aktieselskab
Reklamebeskyttelse	Nej
Status	Underreasummation
	Alle enheder på adressen

The most unusual company listing in the present analysis (with $K = 8$ clusters).

"Sammensat status" is unusual: "Underreasummation". There is only a single instance of this category.

Other examples: "Medarbejderinvesteringsselskab" (one of this kind), SAS DANMARK A/S (large number of employees compared to p-sites?)

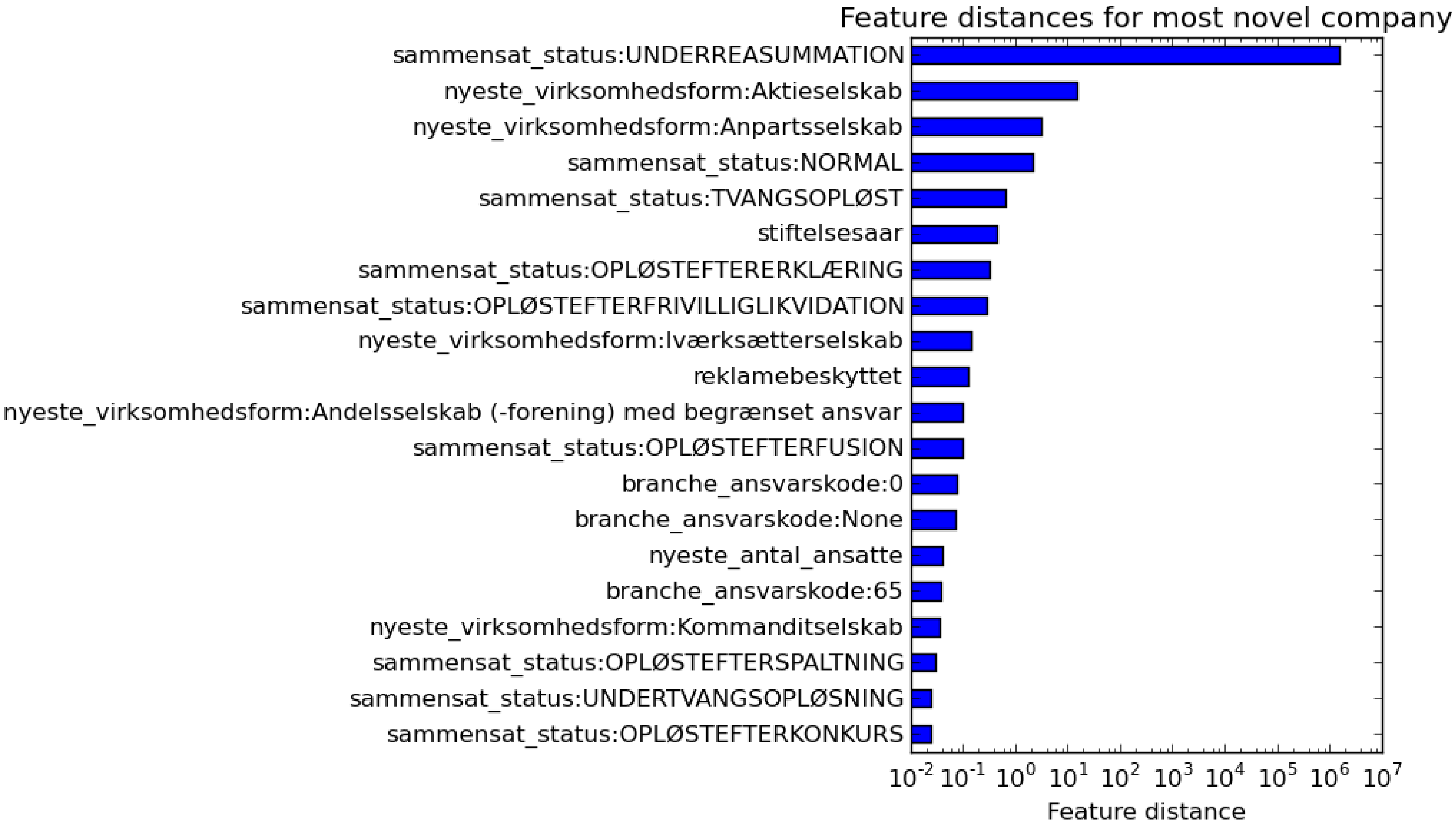
Company novelty distances



Histogram of distances from company features to their estimated cluster centers

Here for the companies assigned to the cluster with the most novel/outlying company.

Company feature distances



Company information for bankruptcy detection

Extract features from 43 GB JSONL file from Erhvervsstyrelsen.

Features extracted with indexing and regular expressions: antal penheder, branche ansvarskode, nyeste antal ansatte, (nyeste virksomhedsform), reklamebeskyttet, sammensat status, (nyeste statuskode), stiftelsesaar.

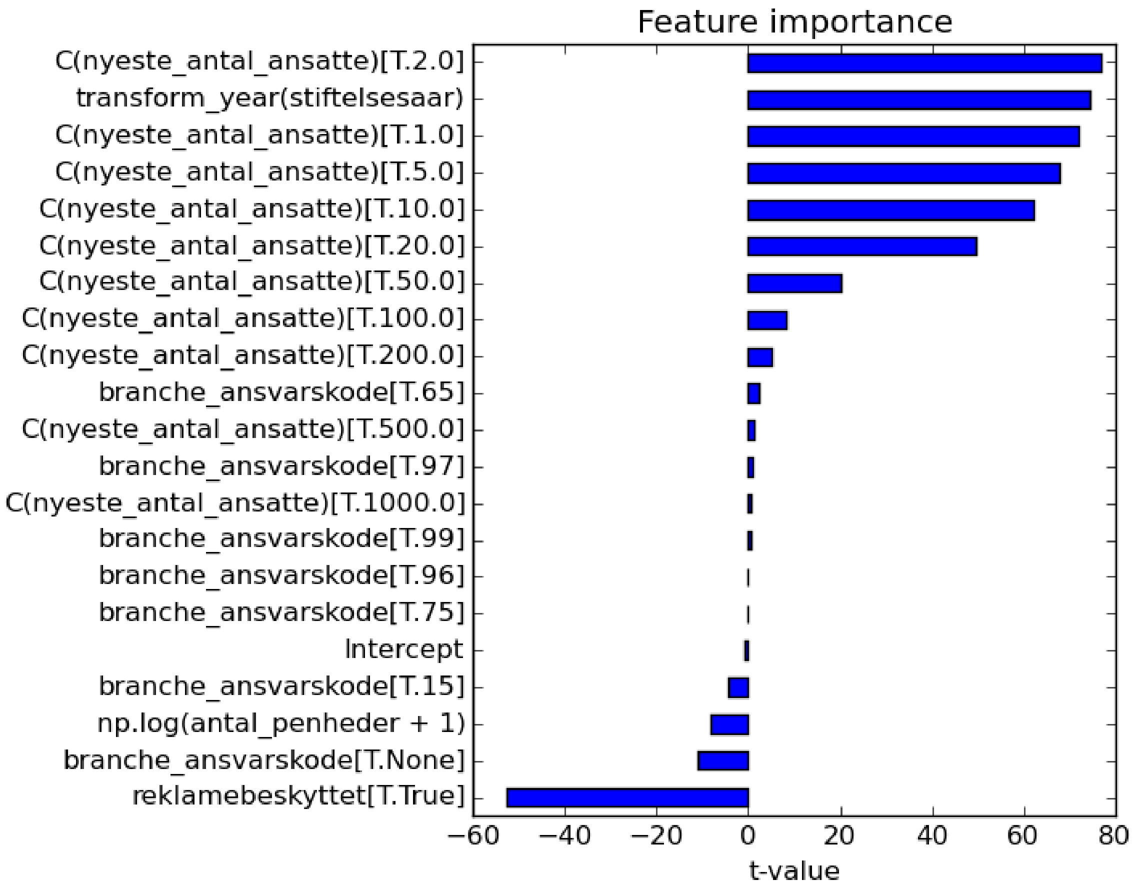
Focus on companies with 'Aktiv' or 'OPLØSTEFTERKONKURS' in "sammensat status".

Technical: Python, Pandas, supervised learning with generalized linear model from statsmodels implemented in a Python module called cvrminer and an IPython Notebook.

Initial bankruptcy detection feature results

	coef	std err	z	P>|z|
Intercept	-0.1821	0.187	-0.976	0.329
C(nyeste_antal_ansatte)[T.1.0]	1.3965	0.019	71.879	0.000
C(nyeste_antal_ansatte)[T.2.0]	1.4391	0.019	76.948	0.000
C(nyeste_antal_ansatte)[T.5.0]	1.6605	0.025	67.751	0.000
C(nyeste_antal_ansatte)[T.10.0]	1.9545	0.032	62.028	0.000
C(nyeste_antal_ansatte)[T.20.0]	2.1077	0.043	49.589	0.000
C(nyeste_antal_ansatte)[T.50.0]	1.8773	0.093	20.237	0.000
C(nyeste_antal_ansatte)[T.100.0]	1.2759	0.157	8.126	0.000
C(nyeste_antal_ansatte)[T.200.0]	1.4266	0.274	5.206	0.000
C(nyeste_antal_ansatte)[T.500.0]	1.0133	0.752	1.347	0.178
C(nyeste_antal_ansatte)[T.1000.0]	0.7364	1.051	0.701	0.484
branche_ansvarskode[T.15]	-4.5699	1.034	-4.421	0.000
branche_ansvarskode[T.65]	0.4971	0.209	2.381	0.017
branche_ansvarskode[T.75]	-24.7808	1.42e+04	-0.002	0.999
branche_ansvarskode[T.96]	28.5924	2.16e+05	0.000	1.000
branche_ansvarskode[T.97]	0.5545	0.614	0.903	0.366
branche_ansvarskode[T.99]	0.2416	0.542	0.446	0.656
branche_ansvarskode[T.None]	-1.9593	0.180	-10.896	0.000
reklamebeskyttet[T.True]	-2.6928	0.051	-52.787	0.000
np.log(antal_penheder + 1)	-0.5775	0.072	-8.058	0.000
transform_year(stiftelsesaar)	0.0498	0.001	74.561	0.000

Bankruptcy detection observation



"reklamebeskyttelse" is surprisingly indicating an "active" company.

The age of the company is important (in our present analysis)

The size of the company is important cf. "antal penheder" og "antal ansatte".

Example: Wikipedia citations mining

Wikipedia citations mining

13 GB compressed XML file with English Wikipedia dump:

```
bzcat enwiki-20160701-pages-articles.xml.bz2 | less
```

Output from command-line streaming decompression:

```
<mediawiki xmlns="http://www.mediawiki.org/xml/export-0.10/" ...
  <siteinfo>
    <sitename>Wikipedia</sitename>
    <dbname>enwiki</dbname>
    <base>https://en.wikipedia.org/wiki/Main_Page</base>
    <generator>MediaWiki 1.28.0-wmf.8</generator>

  ...

  <page>
    <title>AccessibleComputing</title>
    <ns>0</ns>
    <id>10</id>
    <redirect title="Computer␣accessibility" />
```

Wikipedia citations mining

Iterate over pages and use a regular expression in Perl (does not match all instances):

```perl
$INPUT_RECORD_SEPARATOR = "<page>";

@citejournals = m/({{\s*cite journal.*?}})/sig;
@titles      =  m|<title>(.*?)</title>|;
```

We are after these parts in the wiki text:

```
<ref name=Dapson2007>{{Cite journal |last1= Dapson |first1= R.
 |last2= Frank |first2= M. |last3= Penney |first3= D. |last4= Kiernan
 |first4= J. |title= Revised procedures for the certification of carmine
 (C.I. 75470, Natural red 4) as a biological stain |doi=
 10.1080/10520290701207364 |journal= Biotechnic & Histochemistry
 |volume= 82 |pages= 13 |year= 2007 }}</ref>
```

Wikipedia citations mining

Article | Talk | Read | Edit | View history | More | Search

Carmine

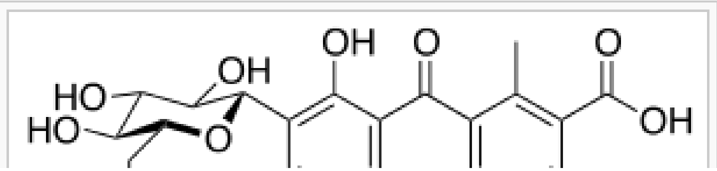
From Wikipedia, the free encyclopedia

This article is about the pigment. For other uses, see

Dapson, R.; Frank, M.; Penney, D.; Kiernan, J. (2007). "Revised procedures for the certification of carmine (C.I. 75470, Natural red 4) as a biological stain". *Biotechnic & Histochemistry*. **82**: 13. doi:10.1080/10520290701207364.

Carmine (/ˈkɑːrmɪn/ or /ˈkɑːrmaɪn/), also called **cochineal, cochineal extract**, **crimson lake** or **carmine lake**, **natural red 4**,[1] **C.I. 75470**,[1] or **E120**, is a pigment of a bright-red color obtained from the aluminium salt of carminic acid; it is also a general term for a particularly deep-red color. The pigment is produced from some scale insects such as the cochineal scale and certain *Porphyrophora* species (Armenian cochineal and Polish cochineal). Carmine is used in the manufacture of artificial flowers, paints, crimson ink, rouge, and other cosmetics, and is routinely added to food products such as yogurt, candy and certain brands of juice, the most notable ones being those of the ruby-red variety.

Carmine



Wikipedia citations mining

To help match different variation of journal names a manually-built XML file was setup:

```
...
<Jou>
  <wojou>7</wojou>
  <name>The Journal of Neuroscience</name>
  <abbreviation>JNeurosci</abbreviation>
  <namePubmed>J Neurosci</namePubmed>
  <type>jou</type>
  <variation>Journal of Neuroscience</variation>
  <variation>j. neurosci.</variation>
  <variation>J Neurosci</variation>
  <wikipedia>Journal of Neuroscience</wikipedia>
</Jou>

...
```

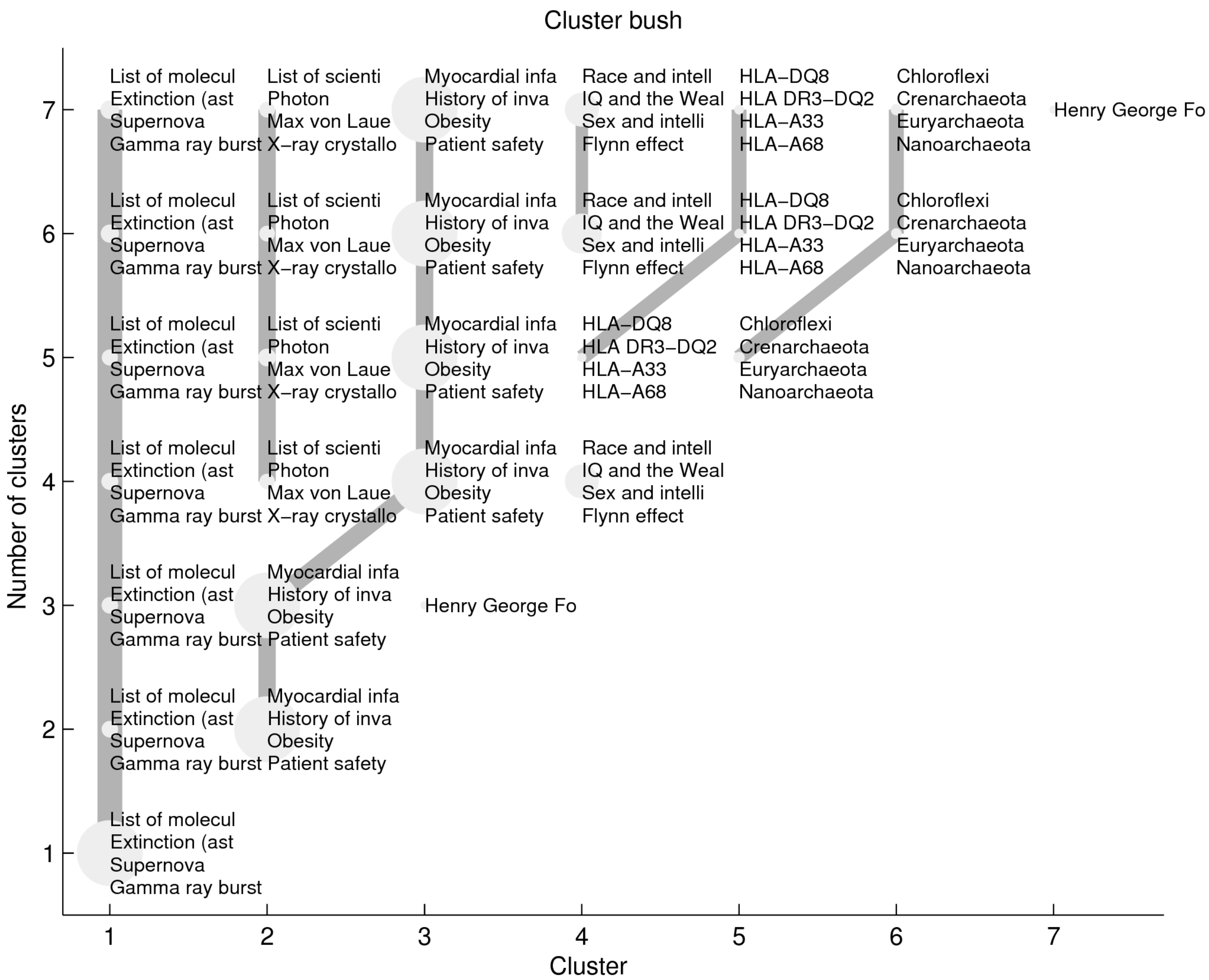
Wikipedia citations mining

	Science	Nature	JBC	JAMA	AJ	...
Evolution	3	1	1	0	1	...
Bacteria	1	3	0	1	0	...
Sertraline	0	0	4	2	0	...
Autism	0	0	0	2	0	...
Uranus	1	0	0	0	3	...
⋮	⋮	⋮	⋮	⋮	⋮	⋱

Begin with (Wikipedia articles × journals)-matrix.

Topic mining with non-negative matrix factorization. This algorithm is, e.g., implemented in sklearn.

Wikipedia citations mining



Wikipedia citations mining

Cluster 21

#	Cites	Load	Wikipedia hub article	#	Cites	Load	Authoritative journal
1	67	2.561	Multiple sclerosis signs and symptoms	1	232	5.036	Neurology
2	119	1.937	Alzheimer's disease	2	72	1.292	Archives of Neurology
3	65	1.936	Multiple sclerosis	3	79	1.063	Annals of Neurology
4	44	1.596	Familial hemiplegic migraine	4	24	0.834	j neurol
5	33	0.986	Episodic ataxia	5	61	0.773	Brain
6	56	0.961	Parkinson's disease	6	24	0.695	j neurol sci
7	23	0.892	Restless legs syndrome	7	19	0.689	mult scler
8	43	0.890	Migraine	8	42	0.675	j neurol neurosurg psychiatr
9	25	0.769	Benign familial neonatal convulsions	9	8	0.358	european archives of psychiatry and clinical neuroscience
10	24	0.666	Therapies under investigation for multiple sclerosis	10	12	0.318	lancet neurology

Example of cluster with Wikipedia articles and scientific journals

Summing up

Structured and unstructed data

Structured data: Data that can be represented in a table and "easily" converted to numerical data and with a fixed number of columns. Represented in CSV, SQL databases, spreadsheet. Most machine learning/statistical algorithms need a fixed size input.

Unstructed data: Data with no fixed number of columns/fields. Free-format text, . . .

Semi-structured data: Data not in column format:

Semi-structured data **I**: Representation in XML, JSON, JSONL (lines of JSON), NoSQL databases, . . .

Semi-structured data **II**: Semi-structured data easy to convert to structured data, e.g., Semantic Web. Represented in triple format, SPARQL engine, . . .

Machine learning

Supervised learning (regression, classification, . . .)

- Python now has a range of of-the-shelve data analysis packages: machine learning (sklearn), statistics (statsmodels) and deep learning
- Linear models also available in R.

Unsupervised learning (clustering, topic mining, density modeling . . .)

- Novelty detection, detection of anormalies
- Topic mining, e.g., of text corpora

Background knowledge from Semantic Web (Wikidata et al.)

Streaming data processing

Operations that can be performed using streaming processes:

- Counting, mean, . . .
- Feature extraction for large datasets for conversion to "medium-sized" data for in-memory data analysis.

Operations which is not so efficient with streaming because of data reload: many machine learning algorithms. Streamining machine learning solutions,

- Batch processing, e.g., partial fit of sklearn in Python, deep learning.
- Spark's MLlib/ML

References

Hansen, L. K., Arvidsson, A., Nielsen, F. Å., Colleoni, E., and Etter, M. (2011). Good friends, bad news — affect and virality in Twitter. In Park, J. J., Yang, L. T., and Lee, C., editors, *Future Information Technology*, volume 185 of *Communications in Computer and Information Science*, pages 34–43, Berlin. Springer.