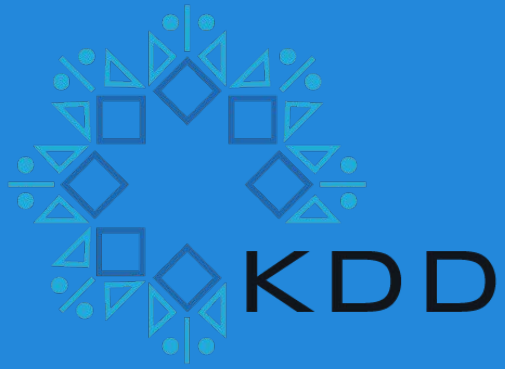
Introduction to Big Natural Language Processing (NLP)



Gabor Melli, Matt Seal

Aug.15, 2016

goo.gl/m4MVsT

Schedule (9AM - 12PM)

- Overview
- Text Preprocessing

} 40 mins.

- Syntactic Parsing
- Shallow Semantic Analysis

} 30 mins.

- Big NLP Framework Demos
- Text Classification & Clustering

} 40 mins.

- Information Extraction
- Word Relatedness Functions

} 20 mins.

About us...

- PhD thesis on "*Supervised Ontology to Document Interlinking*".
- 7+ years of applied NLP.
- 14+ years of applied data science.
- linkedin.com/in/Melli

- Masters focus on AI and NLP
- 4 years of NLP and data modeling at OpenGov
- https://databythebay2016.sched.org/speaker/matthew_seal.1uvtb8mk
- linkedin.com/in/matthew-seal-70a07932

OPENGOV

OpenGov is a software as a service (SaaS) company that provides financial data management, visualization, and intelligence software for local governments across the US and Canada.

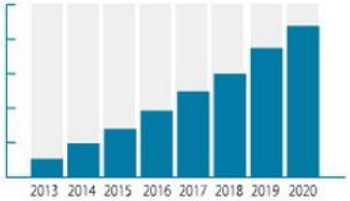
We currently serve over 1,000 governments and have been around for 4 years.

Global Text Analytics Market

Segmentation and Forecast, 2013 - 2020

Global Text Analytics Market

Global Air Compressor Market is Expected to reach $6.5 Billion by 2020



Growing at a CAGR of 25.2% (2014-2020)

GLOBAL TEXT ANALYTICS MARKET BY END USER

- Fast Moving Consumer Goods (FMCG)
- Banking Financial Services and Insurance (BFSI)
- Healthcare and Pharmaceuticals
- Telecommunication
- Government
- Retail

The comprehensive view on the % share of End user segment (2020)

GLOBAL TEXT ANALYTICS MARKETS BY GEOGRAPHY

North America is expected to be highest revenue generating region by 2020

Asia Pacific, Europe,LAMEA

GLOBAL TEXT ANALYTICS MARKET BY APPLICATIONS

- Competitive Intelligence
- Customer Relationship management
- Predictive Analytics
- Fraud detection
- Brand Reputation

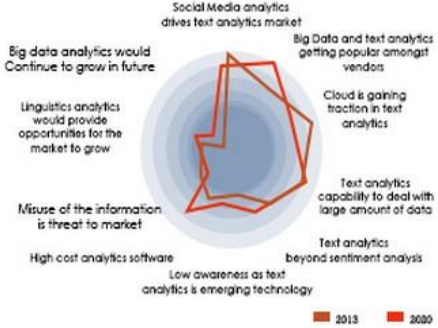
The comprehensive view on the % share of application segment (2020)

GLOBAL TEXT ANALYTICS MARKET BY DEPLOYMENT MODEL

- On-Premise Model
- Cloud Based Model

The comprehensive view on the % share of Deployment model segment (2020)

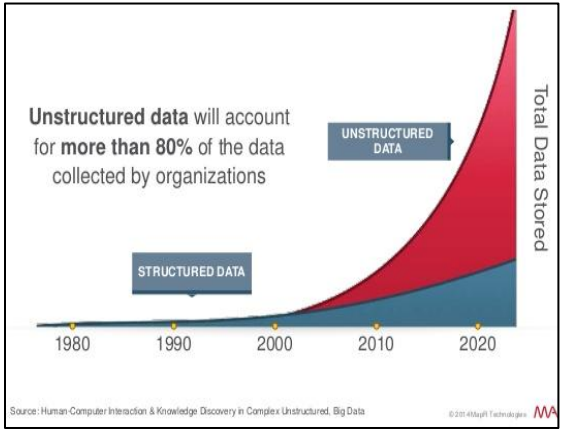
Top Impacting Factors



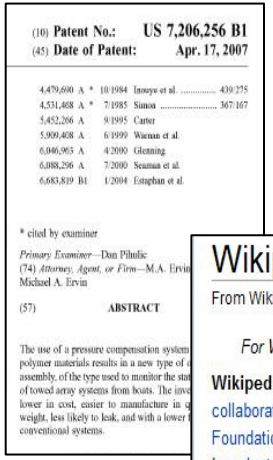
For More Details See Table of Contents

“Big” NLP?

Volume

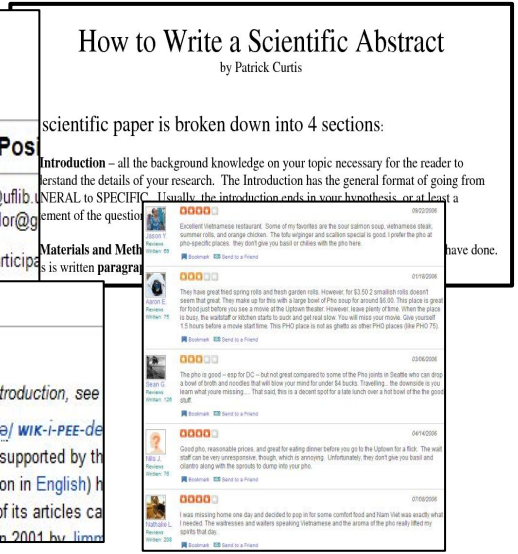


Variety



Velocity

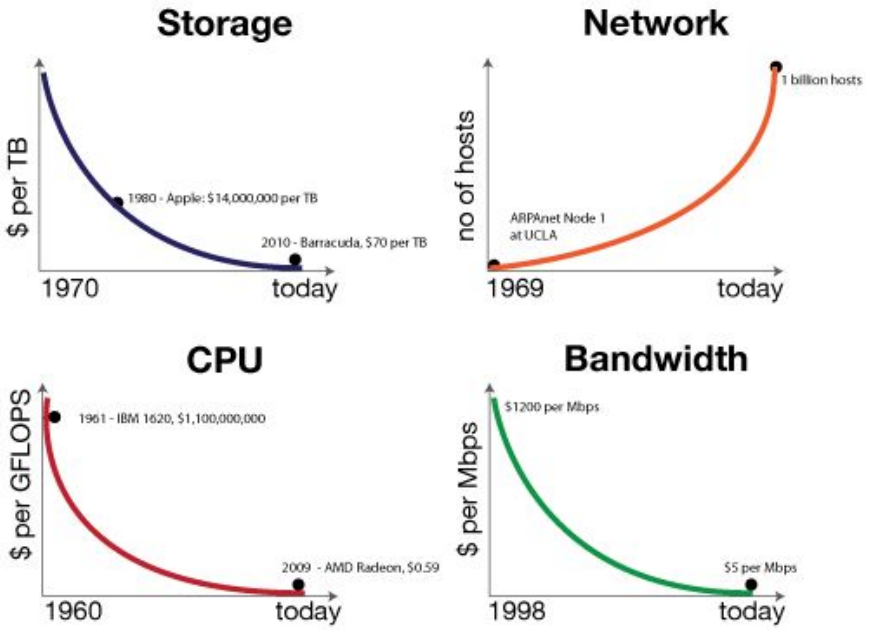
Veracity



Growth in Computing and Storage

AWS EC2 General Purpose	
m3.medium	$0.0261 per Hour
m3.large	$0.0522 per Hour
m3.2xlarge	$0.3243 per Hour

AWS S3 Standard Storage	
First 1 TB / month	$0.0300 per GB
Next 49 TB / month	$0.0295 per GB
Next 450 TB / month	$0.0290 per GB



After this tutorial you will know:

- How to create best-practice baseline solutions to many key (Big) NLP tasks.
- What some state-of-the-art methods approaches are.
- More ways to learn about the topic.
- And ...

One hour of offline assistance. @gmelli / @mseal

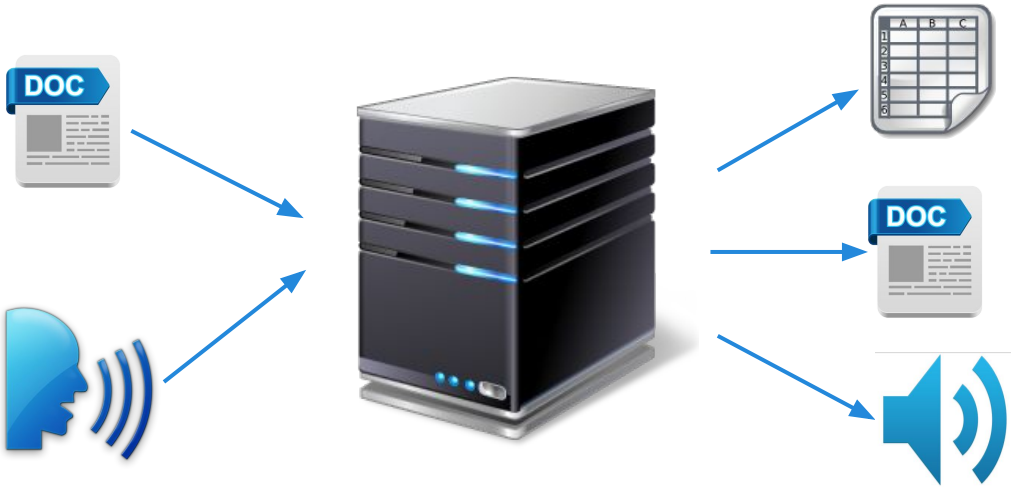
Year 1000
Langues d'oïl
Franco - Pro
Occitan
Galician
Leonese
Castilian
Basque
Aragonese
Catalan
Arabic & Mozarabic
Ethnic-Li
Southwe

We are here

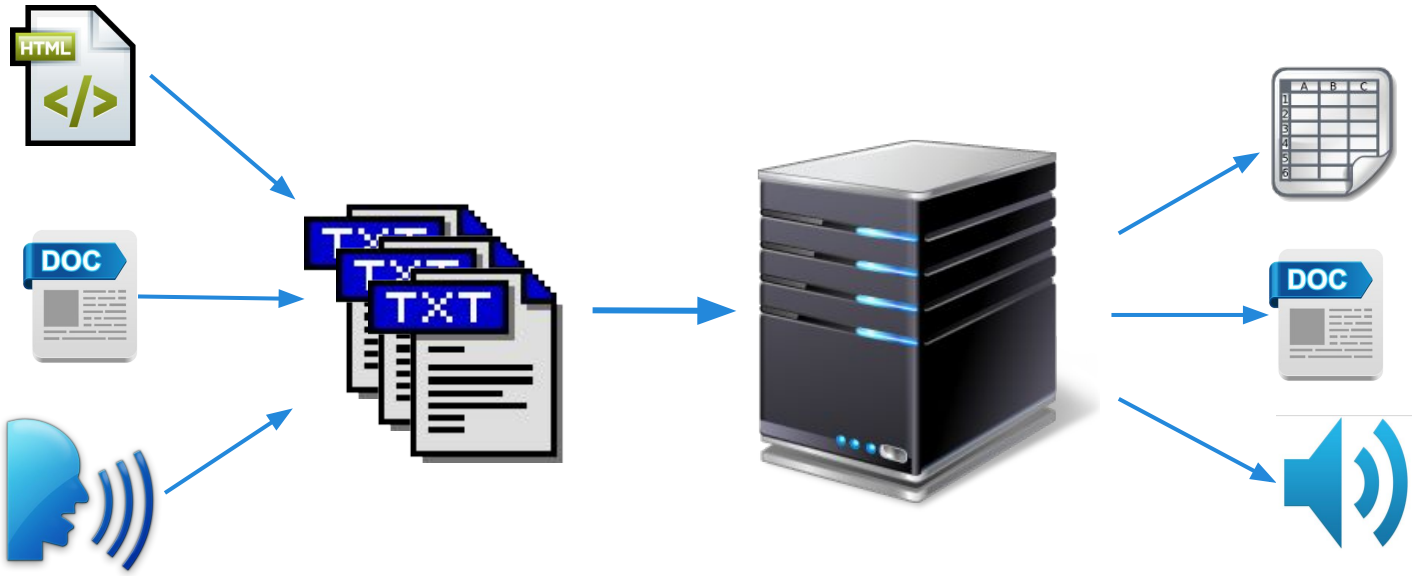
- **> Overview <**
- Text Preprocessing
- Syntactic Parsing
- Big NLP Framework Demos
- Shallow Semantic Analysis
- Text Classification & Clustering
- Information Extraction
- Word Relatedness Functions

What is NLP?

Machining that can process linguistic data

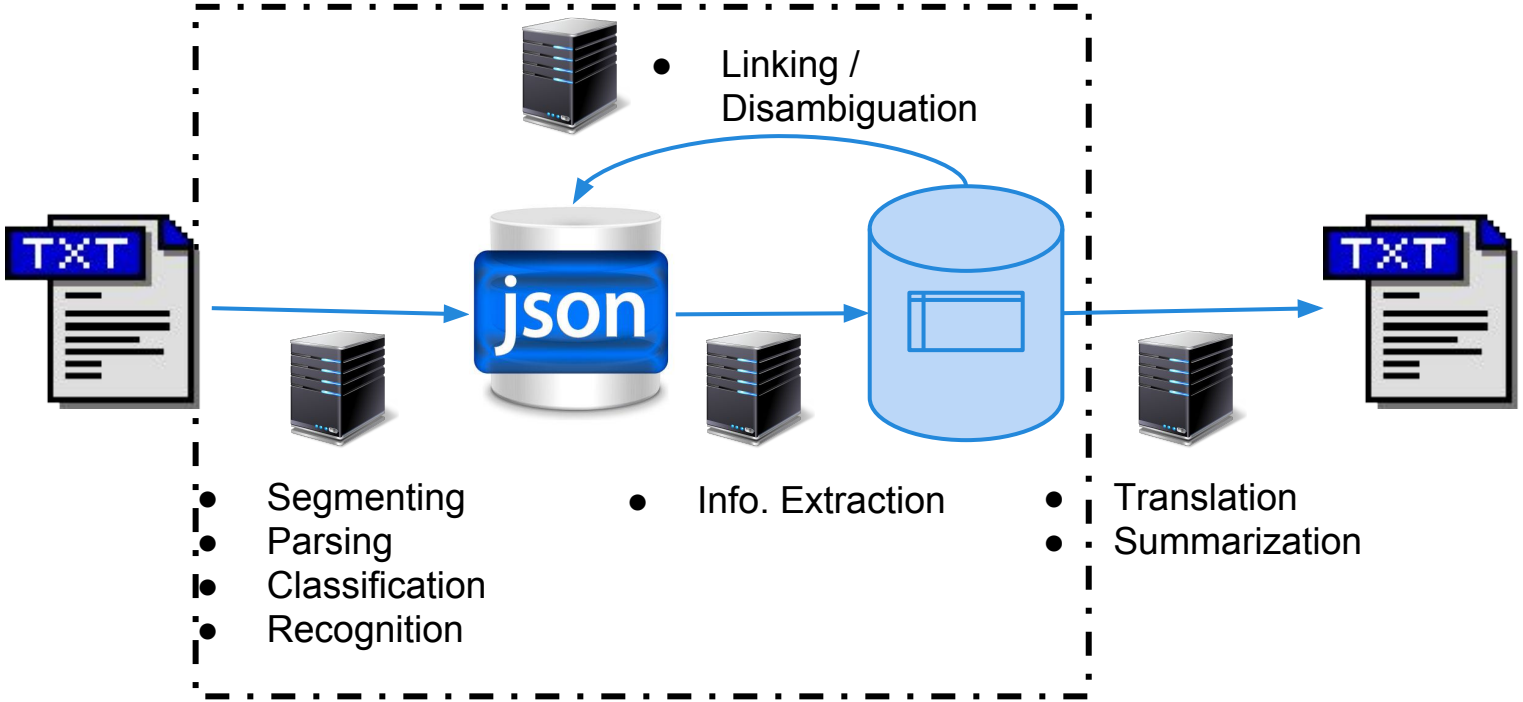


NLP often assumes “text” input



NLP often assumes “text” input

NLP Task Steps



coursera .org/nlp/lecture

Week 1 - Basic Text Processing
- Regular Expressions (11:25)
- Regular Expressions in Practical NLP (6:04)
- Word Tokenization (14:26)
- Word Normalization and Stemming (11:47)
- Sentence Segmentation (5:31)

Week 1 - Edit Distance
- Defining Minimum Edit Distance (7:04)
- Computing Minimum Edit Distance (5:54)
- Backtrace for Computing Alignments (5:55)
- Weighted Minimum Edit Distance (2:47)
- Minimum Edit Distance in Computational Biology

Week 2 - Language Modeling
- Introduction to N-grams (8:41)
- Estimating N-gram Probabilities (9:38)
- Evaluation and Perplexity (11:09)
- Generalization and Zeros (5:15)
- Smoothing: Add-One (6:30)
- Interpolation (10:25)
- Good-Turing Smoothing (15:35)
- Kneser-Ney Smoothing (8:59)

Week 2 - Spelling Correction
- The Spelling Correction Task (5:39)
- The Noisy Channel Model of Spelling (19:30)
- Real-Word Spelling Correction (9:19)
- State of the Art Systems (7:10)

Week 3 - Text Classification
- What is Text Classification? (8:12)
- Naive Bayes (3:19)
- Formalizing the Naive Bayes Classifier (9:28)
- Naive Bayes: Learning (5:22)
- Naive Bayes: Relationship to Language Modeling
- Multinomial Naive Bayes: A Worked Example
- Precision, Recall, and the F measure (16:16)
- Text Classification: Evaluation (7:17)
- Practical Issues in Text Classification (5:56)

Week 3 - Sentiment Analysis
- What is Sentiment Analysis? (7:17)
- Sentiment Analysis: A baseline algorithm (13:27)
- Sentiment Lexicons (8:37)
- Learning Sentiment Lexicons (14:45)
- Other Sentiment Tasks (11:01)

Week 4 - Discriminative classifiers: MaxEntropy classifiers
- Generative vs. Discriminative Models (7:49)
- Making features from text for discriminative NLP models
- Feature-Based Linear Classifiers (13:34)
- Building a Maxent Model: The Nuts and Bolts (8:04)
- Generative vs. Discriminative models:
- Maximizing the Likelihood (10:29)

Week 4 - Named entity recognition and MaxEntropy Models
- Introduction to Information Extraction (9:18)
- Evaluation of Named Entity Recognition (6:34)
- Sequence Models for Named Entity Recognition (15:05)
- Maximum Entropy Sequence Models (13:01)

Week 4 - Relation Extraction
- What is Relation Extraction? (9:47)
- Using Patterns to Extract Relations (6:17)
- Supervised Relation Extraction (10:51)
- Semi-Supervised and Unsupervised Relation Extraction (9:53)

Week 5 - Advanced Maximum Entropy Models
- The Maximum Entropy Model Presentation (12:14)
- Feature Overlap/Feature Interaction (12:51)
- Conditional Maxent Models for Classification (4:11)
- Smoothing/Regularization/Priors for Maxent Models

Week 5 - POS Tagging
- An Intro to Parts of Speech and POS Tagging (13:19)
- Some Methods and Results on Sequence Models for POS Tagging

Week 5 - Parsing Introduction
- Syntactic Structure: Constituency vs Dependency (8:46)
- Empirical/Data-Driven Approach to Parsing (7:11)
- The Exponential Problem in Parsing (14:30)

Week 6 - Probabilistic Parsing
- CFGs and PCFGs (15:29)
- Grammar Transforms (12:05)
- CKY Parsing (23:25)
- CKY Example (21:52)
- Constituency Parser Evaluation (9:45)

Week 6 - Lexicalized Parsing
- Lexicalization of PCFGs (7:03)
- Charniak's Model (18:23)
- PCFG Independence Assumptions (9:44)
- The Return of Unlexicalized PCFGs (20:53)
- Latent Variable PCFGs (12:07)

Week 6 - Dependency Parsing (Optional)
- Dependency Parsing Introduction (10:25)
- Greedy Transition-Based Parsing (31:05)
- Dependencies Encode Relational Structure (7:20)

Week 7 - Information Retrieval
- Introduction to Information Retrieval (9:16)
- Term-Document Incidence Matrices (8:59)
- The Inverted Index (10:42)
- Query Processing with the Inverted Index (6:43)
- Phrase Queries and Positional Indexes (19:45)

Week 7 - Ranked Information Retrieval
- Introducing Ranked Retrieval (4:27)
- Scoring with the Jaccard Coefficient (5:06)
- Term Frequency Weighting (5:59)
- Inverse Document Frequency Weighting (10:16)
- TF-IDF Weighting (3:42)
- The Vector Space Model (16:22)
- Calculating TF-IDF Cosine Scores (12:47)
- Evaluating Search Engines (9:02)

Week 8 - Semantics
- Word Senses and Word Relations (11:50)
- WordNet and Other Online Thesauri (6:23)
- Word Similarity and Thesaurus Methods (16:17)
- Word Similarity: Distributional Similarity I (13:14)
- Word Similarity: Distributional Similarity II (8:15)

Week 8 - Question Answering
- What is Question Answering? (7:28)
- Answer Types and Query Formulation (8:47)
- Passage Retrieval and Answer Extraction (6:38)
- Using Knowledge in QA (4:25)
- Advanced: Answering Complex Questions (4:52)

Week 8 - Summarization
- Introduction to Summarization
- Generating Snippets
- Evaluating Summaries: ROUGE
- Summarizing Multiple Documents

“Big?” Natural Language Processing

- Volume
  - > 10 GB (e.g. English Wikipedia ~43 GB)
- Velocity
  - > 1 GB/day
- Variety
  - e.g. HTML, PDF, ...
- Veracity
  - Formal to informal (tweets).

NLP Capability Framework

DS Capability		Focus	Examples
Explore	Modeling Task	What is the basic modeling task?	clustering, classif., ranking, estim., forecast, ...
	Modeling Method	What are the techniques being used?	k-Means, PCA, k-NN, DTree, SVMs, NNet, CRFs, ...
	Data Domain	What king of data is involved in the task?	financial transaction, click streams, ...
	Data Source	Where did the data come from?	which database tables, data source, ...
	Dev. Environment	What was the modeling environment?	laptop, DBMS server, Spark cluster, ...
	Visualization	How can patterns and models be visualized?	Excel, maplotlib, ...
	Intrinsic Metric	What is the intrinsic metric?	accuracy, AUC, precision, F1, RMSE, ..., subjective
	Benchmark Data	What data was used to benchmark?	which dataset used to benchmark against?
	Retrain(able)	Is the model retrainable with additional data?	model training and evaluation push-button, ...
DS Capability		Focus	Examples
Exploit	Model Format	How is the model instantiated?	table lookup, SQL, Executable object, PMML, ...
	Service Env.	In what environment is the model hosted and used?	where is the model, what system instantiates it?
	Source Data	what data source supports inferences?	what data is reference when the model is applied?
	Business Metric	how is business value measured?	Revenue, clicks, downloads, ... subjective.

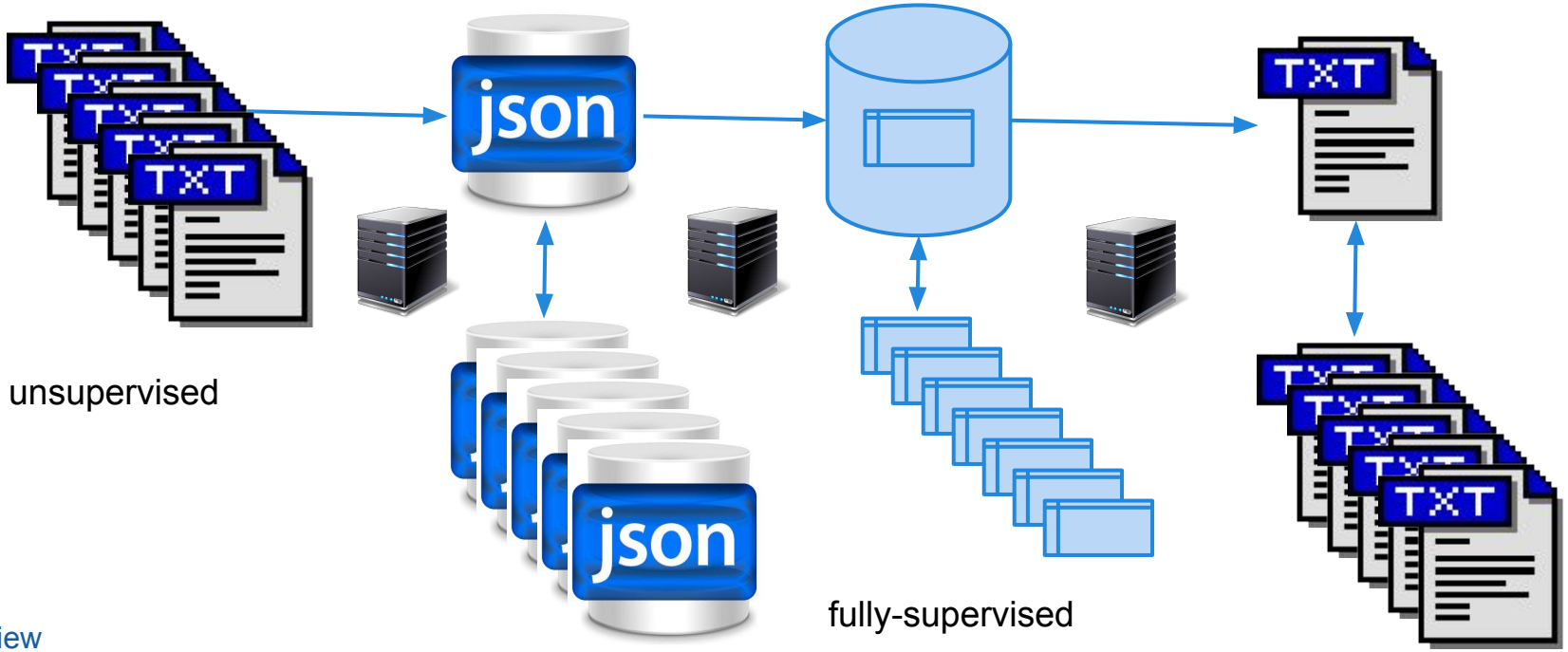
Heuristic NLP (vs. Data-Driven)

Many NLP task can be solved with *good-enough* accuracy using rule-of-thumb/ heuristic approaches.

For example:

- Heuristic end-of-sentence detection.
- Dictionary-based named entity recognition.
- Hearst-patterns: "*cities such as LA and NYC*"

Data-Driven NLP (Input)

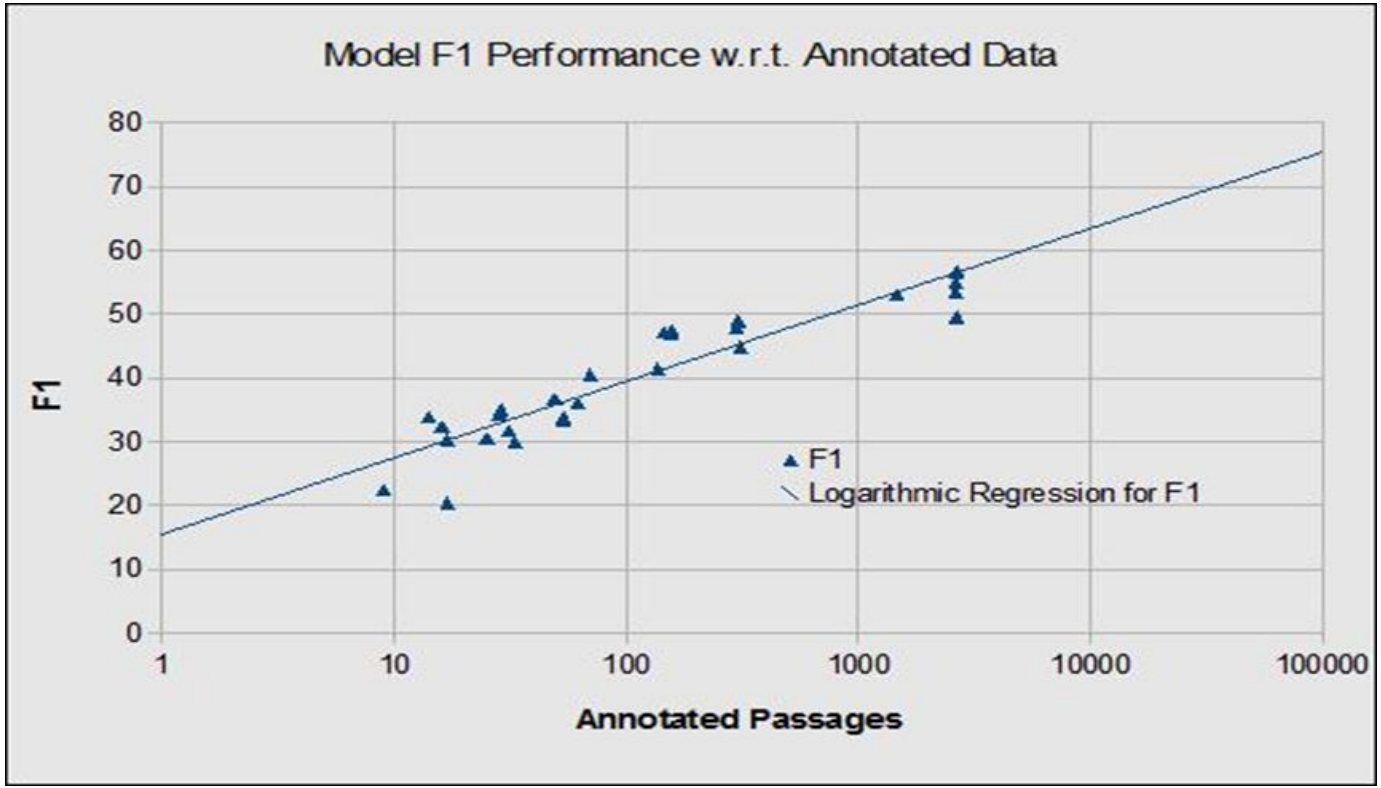

TXT
TXT
TXT

Heuristic vs. Data-Driven NLP

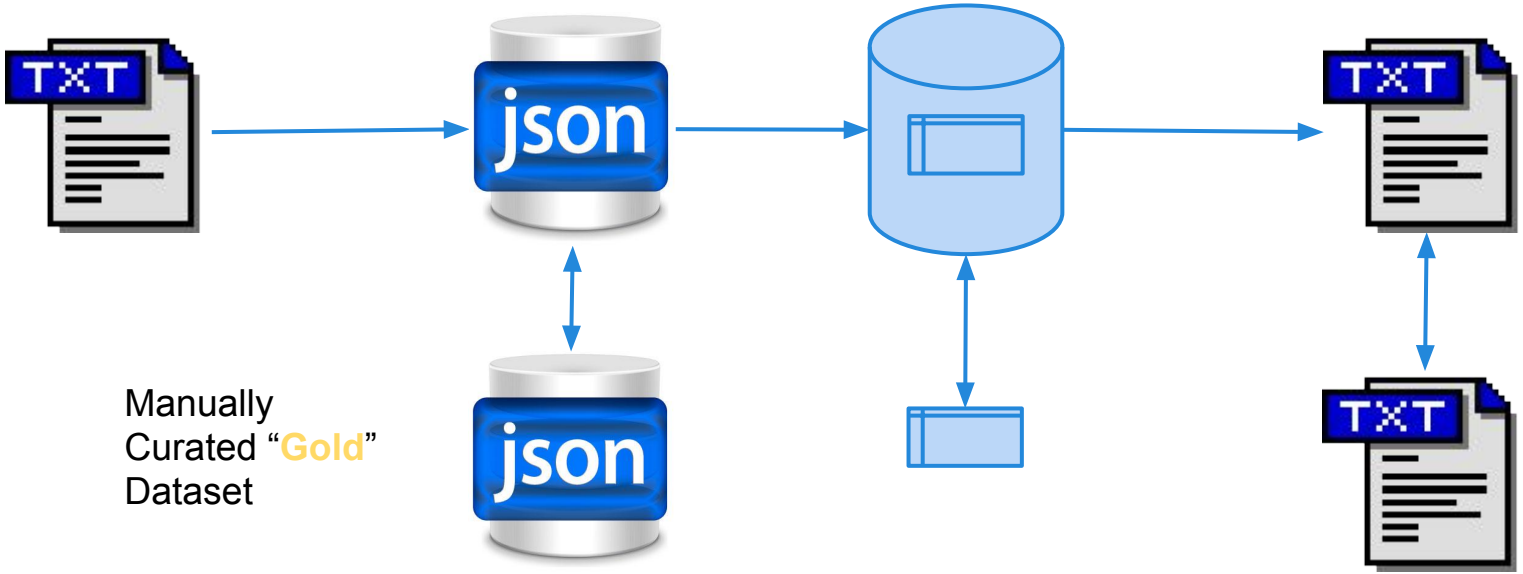
Heuristic approaches are often good ‘baselines’ that can deliver relatively high accuracy on many tasks (so important for Big NLP).

Data-Driven approaches can often complement heuristic approaches to significantly boost accuracy.

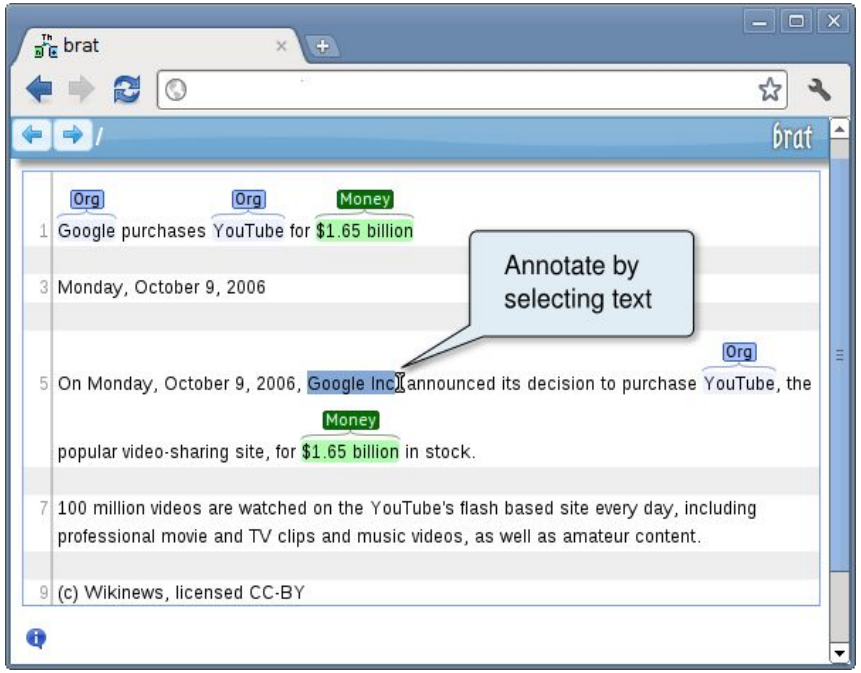
Performance w.r.t. to Data



NLP Evaluation - Benchmark Data



Custom Annotation



APC XS BX1000G 1000VA Tower UPS	Computing Hardware & Accessories
Outwitting the Devil: The Secret to Freedom and Success	Books & eBooks
OmniMount 1 B - 1 Shelf Wall Furniture (Black / Dark Glass)	OTHER Electronics
Shi no fuchi wo mita otoko (Masao Yoshida, man and saw the edge of death of Sun five hundred Fukushima Daiichi nuclear power plant)	Books & eBooks
Logitech Wireless Headset F540 with Stereo Game Audio	Computing Hardware & Accessories
Winegard Distribution Amplifier 5-1000 MHz 15dB	Home Electronics
Rachael Ray Oven Lovin ' Nonstick Bakeware 9-Inch-by-13-Inch Cake Pan, Orange	Kitchen, Dining & Bar
Worship Leaders, We Are Not Rock Stars	Books & eBooks
I Love My Doctor, But...: A Lighthearted Look at a Serious Problem	Books & eBooks

Annotation Guidelines

Product-Linking Annotation Guidelines

Table of Contents

- Annotation in BRAT
- Mention Phrase (yellow)
- Offer Candidates (white)
- Offer Information
- Helper Annotations
- Annotations and Attributes for Offer Candidates
- Validity of Offer Candidate
- Centrality of Mention Phrase in Offer Candidate
- Accessories and Parts
- Context-Property
- Aspects to be always considered for contextuality
- Product category-specific aspects
- Some example annotations
- Annotation Rules and Tips
- ...

Product-related Mention Annotation Guidelines

This document provides guidelines for the annotation of mentions of product related terms that is updatable via brat.

First time readers will benefit from first skimming the entire document to understand its structure.

You may also benefit from review the separate documents on: Dictionary Management, Marketable Item management and Product Title processing.

<SNIP>

General vs. Domain-specific NLP

↑:Overview

- General
  - Newswire, Wikipedia, ..., Twitter

- <u>Domain-Specific</u>
  - Bio-Medicine; Legal; Product Offers; Recipes; ...

Case Study 1: Hyperlink Insertion (CPROD1)

Task: Recognize consumer product mentions within user generated postings, and link to a large product offer catalog.

VigLink

viglink.com/products/insert/

IEEE ICDM
BRUSSELS
10–13 Dec
2012

kaggle

kaggle.com/c/cprod1

CPROD1 Task - Example

- Given a passage such as:

  ...you will drop over $800 for a apple 24" cinema display. ...

- and a large product catalog with entries such as:

2258624	Apple LED Cinema Display 24-Inch MB382LL/A	$619.94	CE
3828027	Viewsonic's VA2448M-LED 24-Inch Widescreen LED Monitor - Black	$174.99	CE
8742810	Apple Cinema 24" Widescreen LCD Monitor - Black	$589.00	CE

- then <u>recognize</u> each product mention

  ...you will drop over $800 for an <u>apple 24" cinema display</u>. ..

- and <u>associate</u> the mention with the correct product id(s):

  `2258624,8742810`

Crowdsouring model development

Benefits

- Forces creation of:
  - a high-quality dataset.
  - very-clear problem description and measure(s).
- Results in algorithmic ideas

Cons

- Time consuming = money

CPROD1 - data

{

"TextItem": {

"abb942017f73da8f522b46fca366f178": [
"Quote", ":", "Originally", "Posted", "by", "kingpcgeek", "I", "just", "signed", "up", "for", "HBO", "GO", ",", "I'm", "in", "the", "AZ", "market", ".", "<s>", "Talk", "about", "a", "quiet", "launch", ".", "<s>", "Very", "convoluted", "setup", ".", "<s>", "Go", "to", "the", "HBO", "GO", "site", "and", "sign", "in", ".", "<s>", "You", "choose", "Cox", "(", "Beta", ")", "and", "it", "takes", "you", "to", "the", "Cox.com", ".", "<s>", "Sign", "in", "on", "cox.com", "and", "get", "no", "indication", "how", "to", "get", "back", "to", "HBO", ".", "<s>", "I", "played", "with", "it", "at", "home", "for", "5", "minutes", "before", "leaving", "for", "work", ".", "<s>", "I", "just", "tried", "to", "play", "it", "here", "at", "work", "and", "I", "can't", "because", "I'm", "logged", "in", "at", "home", "30", "minutes", "ago", ".", "<s>", "Can't", "help", "with", "HBO", "but", "here", "'s", "a", "direct", "link", "that", "works", "for", "me", "after", "I've", "signed", "into", "my", "Arizona", "Cox", "account", "."
],

"063756291aad06d08f7ce44b684aafbf": [
"wow", "90000", "for", "a", "new", "explorer", "?", "<s>", "here", "a", "brand", "new", "one", "with", "everything", "and", "aftermarket", "stuff", "too", "would", "still", "be", "under", "40000", ".", "<s>", "i", "say", "come", "hear", "and", "get", "it", "shipped", "back", "there", ".", "<s>", "well", ",", "welcome", "to", "the", "site", "and", "i", "hope", "you", "will", "find", "everything", "you", "need", ".", "<s>", "under", "the", "word", "explorer", "on", "the", "back", "tailgate", "it", "should", "have", "some", "designation", "like", "xl", "or", "xlt", "on", "a", "chrome", "plastic", "badge", "type", "thing", "like", "the", "word", "explorer", ".", "<s>", "i", "hope", "this", "helps", "."],

"b0831c8859683071f6ead1421e1777fe": [
"Kiefer", "Sutherland", "charged", "with", "assault", "in", "NY", "http://www.reuters.com/article/enter..", ".5460FN20090508"
],

…

}

}

Case Study 2: VigLink Product Offer Parsing

Polk Audio Blackstone TL1600 Speaker sys - home theater - 5.1-CH - wired

[Polk Audio]^{BN} Blackstone^{PL} TL1600^{PI} [Speaker sys]^{PC} [home theater]^{PF} [5.1-CH] ^{PF} [wired]^{PF}

Brand Name (BN): *Polk Audio*
Product Line (PL): *Blackstone*
Product (PI): *TL 1600*
Product Category (PC): *speaker sys*
Product Feature (PF): *home theater*
Product Feature (PF): *5.1-CH*
Product Feature (PF): *wired*

http://gabormelli.com/Projects/PTPv1/

Case Study of Protein Prokaryote Localization Relation Extraction (PPLRE)

Can you spot the **subcellular localization event** mention?

Identification of a TcpC-TcpQ Outer Membrane Complex Involved in the Biogenesis of the Toxin-Coregulated Pilus of *Vibrio cholerae*

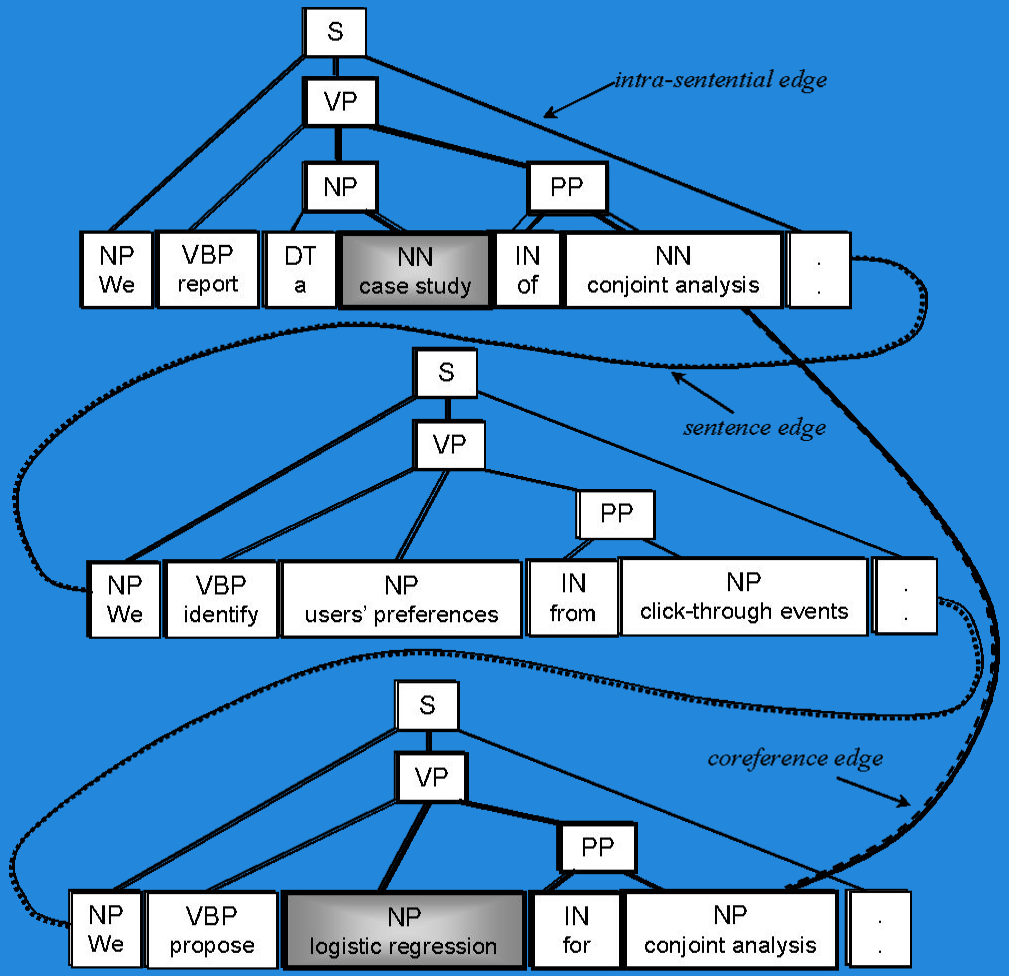
By Niranjan Bose and Ronald K. Taylor

The toxin-coregulated pilus (TCP) of *Vibrio cholerae* and the soluble TcpF protein that is secreted via the TCP biogenesis apparatus are essential for intestinal colonization. The TCP biogenesis apparatus is composed of at least nine proteins but is largely uncharacterized. TcpC is an outer membrane lipoprotein required for TCP biogenesis that is a member of the secretin protein superfamily. In the present study, analysis of TcpC in a series of strains deficient in each of the TCP biogenesis proteins revealed that TcpC was absent specifically in a *tcpQ* mutant. TcpQ is a predicted periplasmic protein required for TCP biogenesis.

Case Study 3: PPLRE after Segmentation and NER ...

"The pilus(location) of V. cholerae(organism) is essential for intestinal colonization.
The pilus(location) biogenesis apparatus is composed of nine proteins.
TcpC(protein) is an outer membrane(location) lipoprotein required for pilus(location) biogenesis."

Organism	Protein	Subcellular Location
V. cholerae	*TcpC*	*outer membrane*


S
VP
NP
PP
intra-sentential edge
NP We
VBP report
DT a
NN case study
IN of
NN conjoint analysis
. .
S
VP
PP
sentence edge
NP We
VBP identify
NP users' preferences
IN from
NP click-through events
. .
S
VP
PP
coreference edge
NP We
VBP propose
NP logistic regression
IN for
NP conjoint analysis
. .

Case Study 4: GM-RKB

Gabor Melli's Research Knowledge Base

Gmelli Talk Preferences Watchlist Contributions

Special page

Search

Parse Text Form

Main page
Recent changes
Random page
Help

Tools
Upload file
Special pages
Printable version

Parse text

Text Content: This paper introduces the problem of combining multiple partitionings of a set of objects into a single consolidated clustering without accessing the features or algorithms that determined these partitionings. We first identify several application scenarios for the resultant `knowledge reuse' framework that we call cluster ensembles. The cluster ensemble problem is then formalized as a combinatorial optimization problem in terms of shared mutual information. In addition to a direct maximization approach, we propose three effective and efficient techniques for obtaining high-quality combiners (consensus functions). The first combiner induces a similarity measure from the partitionings and then reclusters the objects. The second combiner is based on hypergraph partitioning. The third one collapses groups of clusters into

Debug Level: 0

Submit

GM-RKB (cont.): Output

[[Strehl & Ghosh, 2002b|*This paper*]] *introduces the problem of combining* [[*multiple partitionings*]] *of a* [[cluster|*set of object*]]*s into a single* [[*consolidated clustering*]] *without accessing the* [[record feature|*feature*]]*s or* [[clustering algorithm|*algorithm*]]*s that determined these* [[data cluster|*partitioning*]]*s.* </s>

[[Strehl & Ghosh, 2002b|*We*]] *first identify several* [[*application scenario*]]*s for the resultant* [[*`knowledge reuse' framework*]] *that we call* [[*cluster ensemble*]]*s.* </s>

The [[cluster ensemble problem]] is then formalized as a [[combinatorial optimization problem]] in terms of shared [[mutual information]].

In addition to a [[direct maximization approach]], we propose three [[effective]] and [[efficient technique]]s for obtaining [[high-quality combiner]]s ([[consensus function]]s). </s>

The first [[combiner]] [[induces a similarity measure]] from the [[partitionings]] and then [[reclusters the objects]]. </s>

The second [[combiner]] is based on [[hypergraph partitioning]]. </s>

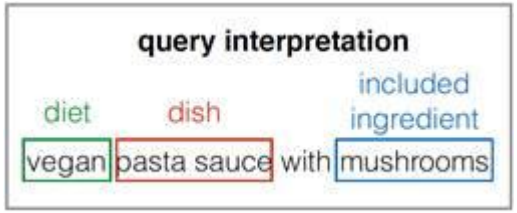
The third one [[collapse]]s [[groups of clusters]] into [[meta-cluster]]s which then compete for each object to determine the combined [[clustering]]. </s>

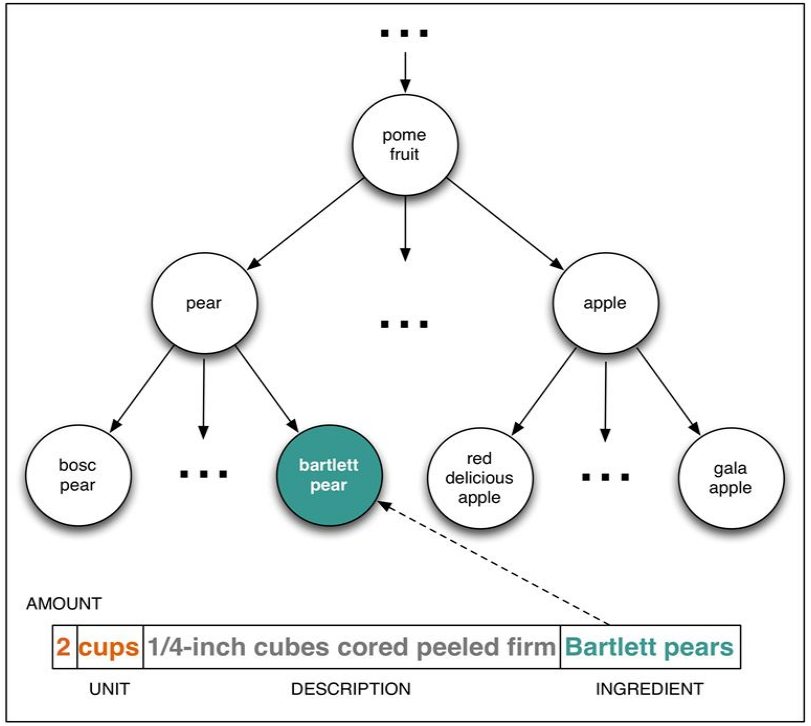
Due to the [[low computational cost]]s of [[our technique]]s, it is quite feasible to use a [[supra-consensus function]] that evaluates all three approaches against [[the objective function]] and picks the best solution for a given situation. </s>

[[Strehl & Ghosh, 2002b|We]] evaluate the [[clustering effectiveness|effectiveness]] of [[cluster ensemble]]s in three qualitatively different [[application scenario]]s: (i) where the original clusters were formed based on [[non-identical sets of features]], (ii) where the original [[clustering algorithm]]s worked on [[non-identical sets of object]]s, and (iii) where a [[common data-set]] is used and the main purpose of combining multiple [[clustering]]s is to improve the quality and [[robustness of the solution]]. </s>

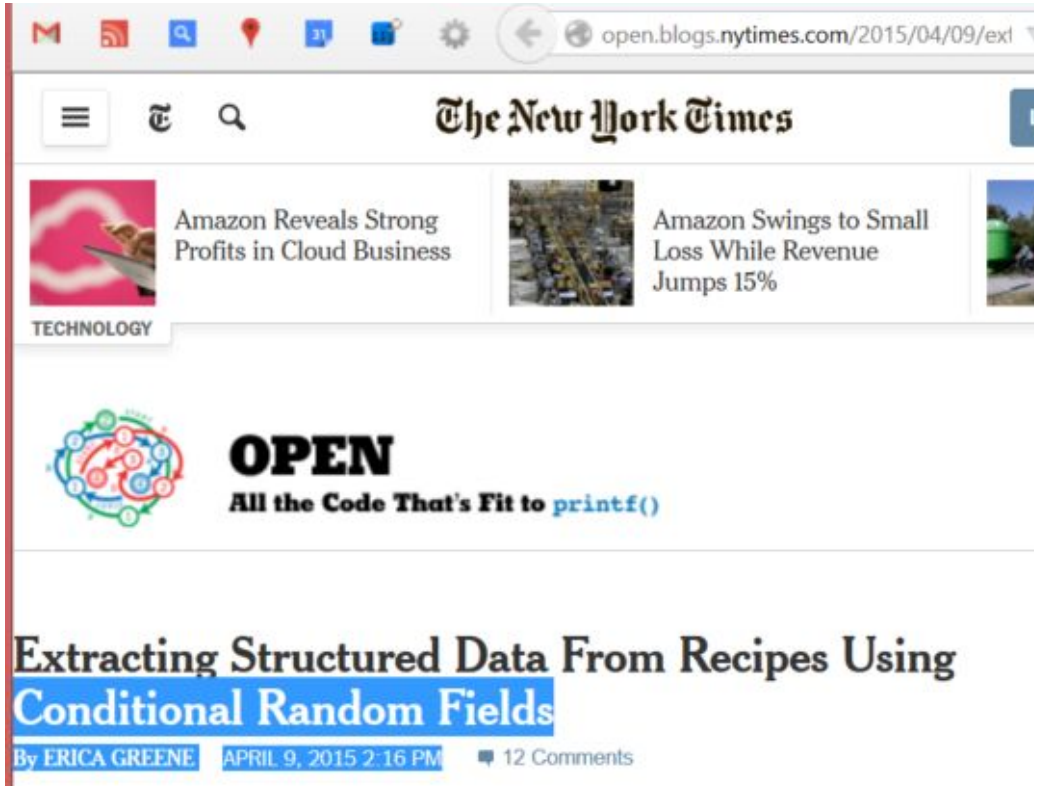
[[Promising result]]s are obtained in all three [[situation]]s for [[synthetic dataset|synthetic]] as well as [[real data-set]]s. </s>

Case Study 5: Parsing Recipes





Recipes at New York Times



… We now have a treasure trove of structured data to play with. As of yesterday, the database contained 17,507 recipes, 67,578 steps, 142,533 tags and 171,244 ingredients broken down by name, quantity and unit. ...

$$f_3(y_t, y_{t-1}, x) = \begin{cases} 1 \text{ if } x_t \text{ is “cup” and} \\ 0 \text{ otherwise} \end{cases}$$

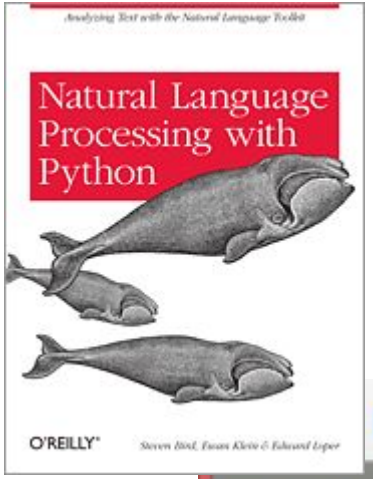
$$f_4(y_t, y_{t-1}, x) = \begin{cases} 1 \text{ if } x_t \text{ is “flour” and} \\ 0 \text{ otherwise} \end{cases}$$

Case Study: OPENGOV

Capability highlights

- Gov. Budget PDF Collection
- Wordnet Synsets: Local Gov. Information Content using Stanford NLP Parser
- HAC Clustered Topics
- OpenGov Label Associations

Natural Language Toolkit (NLTK)



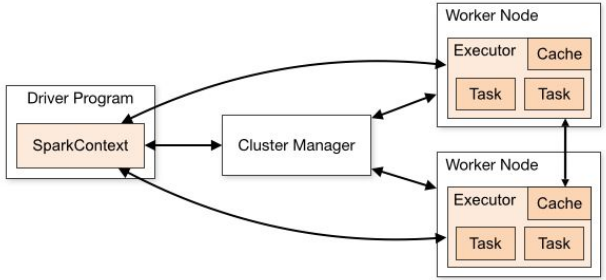
Library highlights

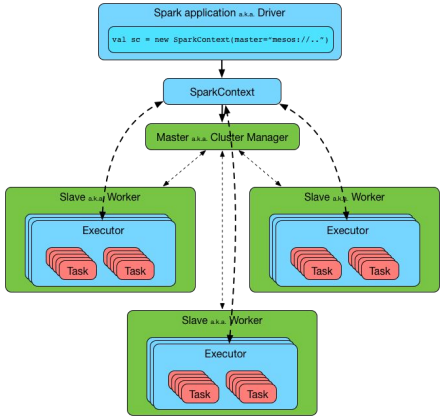
- Lexical analysis: Word and text tokenizer
- n-gram and collocations
- Part-of-speech tagger
- Tree model and Text chunker for capturing
- Named-entity recognition

NLP on Spark (via AWS EMR)

http://Goo.gl/85BIkI

This document describes how to create and use an Apache Spark cluster for Python-based (and iPython) natural language processing tasks using the NLTK library. The Spark cluster is based on Amazon's (AWS) Elastic Map Reduce (EMR) service. A PDF conversion step is also included.

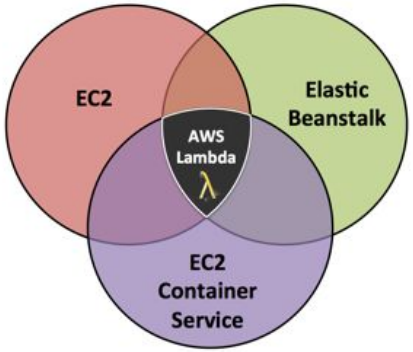




NLP on AWS Lambda

http://Goo.gl/DJ9dbc

This document describes a AWS Lambda function-based approach to perform the PDF-to-text plus NLP process on AWS (instead of, say, on Spark).



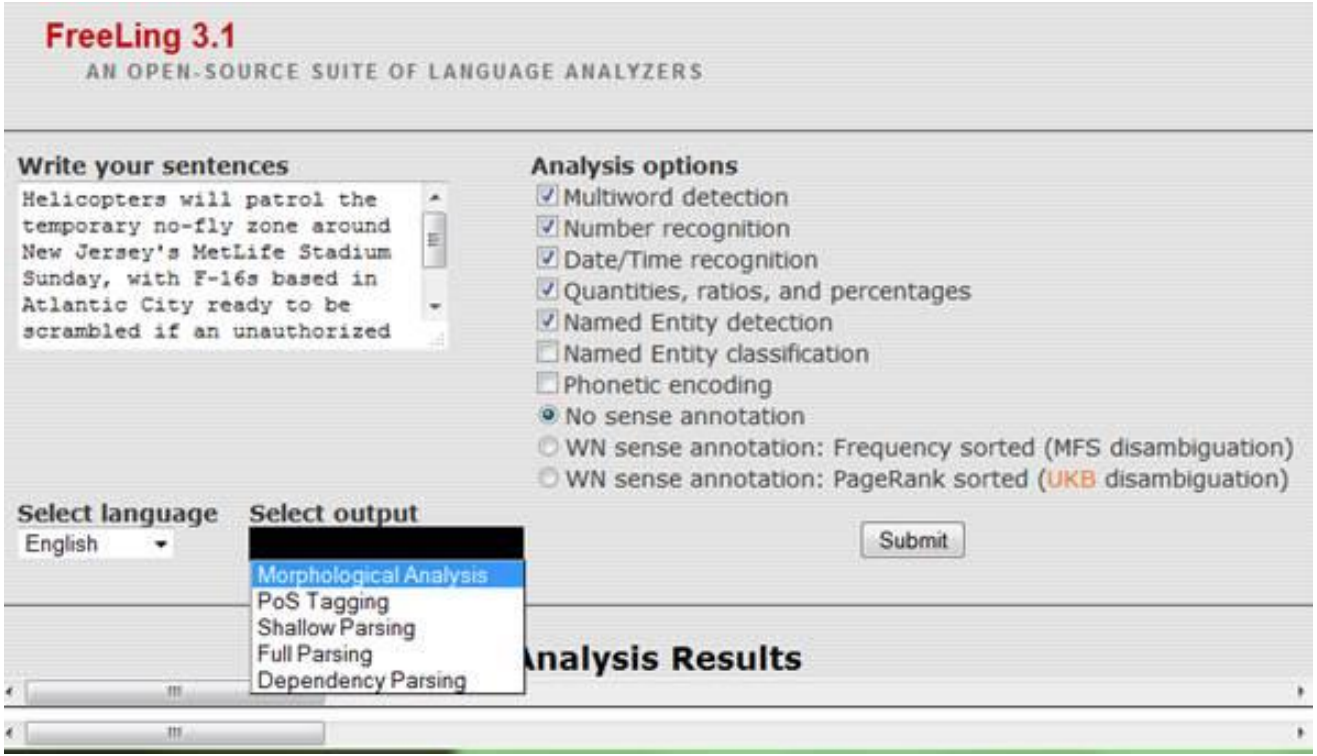
We are here

- ~~Overview~~
- **>> Text Preprocessing <<**
- Syntactic Parsing
- Big NLP Framework Demos
- Shallow Semantic Analysis
- Text Classification & Clustering
- Information Extraction
- Word Relatedness Functions

Text Preprocessing

- Text Segmentation Tasks
  - End-of-Sentence Detection
  - Text Tokenization ($ex\text{-}general's \Rightarrow ex\text{-} \mid general \mid 's$)
  - Word Segmentation ($ice\ cream \Rightarrow ice_cream$)
  - Stemming ($wolves \Rightarrow wolv$)
  - Lemmatization ($wolves \Rightarrow wolf$)
- Text Segmentation Evaluation
  - Alignment
  - Agreement

nlp.lsi.upc.edu/freeling/demo/demo.php



String Segmentation

```
.......___.....___............
```

partial

```
[.][__][___][.][....][__][..]
```

complete

```
x1=5.1; ⇒ [x1][=][5.1][;]

ex-wife's ⇒ [ex-][wife]['s]

...TACGATCGTATCGTAGCACCCCACAGCATC...
```

Intron Intron

Analysis of Misclassifications

a) … on **user-generated** content sites …

b) … , **transfer learning** employs knowledge from ….

c) … in **recommender system** settings …

d) … novel **semi-supervised** boosting algorithms that …

e) … All **Netflix Prize** algorithms proposed …

f) … to **predictive-modeling** holdout evaluation …

g) … on **eigen equation** compression , …

h) … useful **accurately labeled** instances . …

i) … **collaborative filtering** recommendation approaches …

j) … in **historical search session** logs .

k) … than **approximate** spectral clustering based on …

l) … of **prequential** error estimators …

m) … the **multi-relational** data mining framework is …

n) … the **UCI** machine learning repository .

Binary Performance Measures

$$\text{Precision(P)} = \text{TPs} / (\text{TPs}+\text{FPs})$$

$$\text{Recall (R)} = \text{TPs} / (\text{TPs}+\text{FNs})$$

$$F_1 = 2\text{PR} / (\text{P} + \text{R})$$

$$= 2\text{TP} / (2\text{TP} + \text{FN} + \text{FP})$$

$$F_b = (1+b^2)\text{TP} / ((1+b^2)\text{TP} + b^2\text{FN} + 1\text{FP})$$

$$F_{0.5} = (1.25)\text{TP} / ((1.25)\text{TP} + 0.25\text{FN} + 1\text{FP})$$

		Predicted	
		TRUE	FALSE
O	TRUE	TP	FN
	FALSE	FP	TN

Calculate Binary Classifier Performance

case#	Decision Predicted by Model	Optimal Decision
1	Act=YES	Act=YES
2	Act=YES	Act=YES
3	Act=NO	Act=YES
4	Act=YES	Act=NO
5	Act=YES	Act=YES
6	Act=NO	Act=NO
7	Act=NO	Act=NO
8	Act=YES	Act=YES
9	Act=YES	Act=NO
10	Act=YES	Act=YES

		Predicted		SUM
		TRUE	FALSE	
O	TRUE	5	2	7
	FALSE	1	2	3
SUM		6	4	10

Model Accuracy	=	
Model Precision	=	
Model Recall	=	
Model F1	=	

Relationship between R & P

F1				R			
P	0.00	0.25	0.50	0.75	0.90	0.95	1.00
0.00	**0.00**	0.00	0.00	0.00	0.00	0.00	0.00
0.25	0.00	**0.25**	0.33	0.38	0.39	0.40	0.40
0.50	0.00	0.33	**0.50**	0.60	0.64	0.66	0.67
0.75	0.00	0.38	0.60	**0.75**	0.82	0.84	0.86
0.90	0.00	0.39	0.64	0.82	**0.90**	0.92	0.95
0.95	0.00	0.40	0.66	0.84	0.92	**0.95**	0.97
1.00	0.00	0.40	0.67	0.86	0.95	0.97	**1.00**

Inter-Annotation Agreement (*κ*appa)

case#	A's Predictions	B's Predictions
1	Act=YES	Act=YES
2	Act=YES	Act=YES
3	Act=NO	Act=YES
4	Act=YES	Act=NO
5	Act=YES	Act=YES
6	Act=NO	Act=NO
7	Act=NO	Act=NO
8	Act=YES	Act=YES
9	Act=YES	Act=NO
10	Act=YES	Act=YES

		B		SUM	%
		Yes	No		
A	Yes			7	0.7
	No			3	0.3
SUM		6	4	10	
%		0.6	0.4		

$$\kappa = \frac{Pr(a)-Pr(e)}{1-Pr(e)}$$

$$Pr(a)=(5+2)/10$$
$$=0.70$$

$$Pr(e)=(.7\times.6)+(.3\times.4)$$
$$=0.54$$

$$\kappa = \frac{0.70-0.54}{1-0.54}$$
$$=0.348$$

End-of-Sentence Detection

EoSD("*I saw E. coli under the microscope with Dr. Smith. They were moving.*")

⇒ *I saw E. coli under the microscope with Dr. Smith.*</SENT>
They were moving.</SENT>

EoSD("*Mr. Toro "announced" Sony's new Vaio (VPC-F223FX/S) today. It compares well to the iPad 2.*")

⇒ *Mr. Toro "announced" Sony's new Vaio (VPC-F223FX/S) today.* </SENT>
It compares well to the iPad 2.</SENT>

Text Tokenization - Task

TTT("*I'm coming home*")

⇒ `[I] ['m] [coming] [home].`

TTT("*Der Lebensversicherungsgesellschaftsangestellte kam gestern mit seinem Deutscher Schäferhund.*" (~*The life insurance company employee came yesterday with their German Shepherd*)

⇒ `[Der] [Lebensversicherungsgesellschaftsangestellte] [kam] [gestern] [mit] [seinem] [Deutscher] [Schäferhund] .`

TTT("*The ex-governor general's sisters-in-law saw the wolves' den near Mr. Smith's home in Sault Ste. Marie.*")

⇒ `[The] [ex-] [governor] [general] ['s] [sisters-in-law] [saw] [the] [wolves] ['] [den] [near] [Mr.] [Smith] ['s] [home] [in] [Sault] [Ste.] [Marie] [.].`

Text Tokenization - NLTK

text-processing.com/demo/tokenize

TreebankWordTokenizer

1. Famous | notaries | public | include | ex-attorney | generals | .

WordPunctTokenizer

1. Famous | notaries | public | include | ex | - | attorney | generals | .

PunktWordTokenizer

1. Famous | notaries | public | include | ex-attorney | generals.

WhitespaceTokenizer

1. Famous | notaries | public | include | ex-attorney | generals.

pattern

1. Famous | notaries | public | include | ex-attorney | generals | .

NLTK on IPython

jupyter NLTK demo

File Edit View Insert Cell Kernel Help | Python 2

Code

Cell Toolbar: None

demo of NLTK

```
In [11]: import nltk
         nltk.__version__
Out[11]: '3.0.2'
```

```
In [8]: sentence = """At eight o'clock on Thursday morning
        ... Arthur didn't feel very good."""
        sentence
Out[8]: "At eight o'clock on Thursday morning\nArthur did
        n't feel very good."
```

```
In [10]: tokens = nltk.word_tokenize(sentence)
         tokens
Out[10]: ['At',
          'eight',
          "o'clock",
          'on',
          'Thursday',
          'morning',
          'Arthur',
          'did',
          "n't",
          'feel',
          'very',
          'good',
          '.']
```

```
In [7]: tagged = nltk.pos_tag(tokens)
        tagged[0:6]
Out[7]: [('At', 'IN'),
         ('eight', 'CD'),
         ("o'clock", 'JJ'),
         ('on', 'IN'),
         ('Thursday', 'NNP'),
         ('morning', 'NN')]
```

Word Segmentation - Task

- WST("*I'm buying a real time operating system*")
  ⇒ `[I] ['m] [buying] [a] [real time] [operating system] .`

- WST("*Famous notaries public include ex-attorney generals.*")
  ⇒ `[Famous] [notaries public] [include] [ex-] [attorney generals] .`

- WMST("*Der Lebensversicherungsgesellschaftsangestellte kam gestern mit seinem Deutscher Schäferhund.*" (~*The life insurance company employee came yesterday with their German Shepherd)*
  ⇒ `[Der] [Lebensversicherungs] [gesellschafts] [angestellte] [kam] [gestern] [mit] [seinem] [Deutscher Schäferhund]`

- WST("日文章魚怎麼說")
  ⇒ [日文] [章魚] [怎麼] [說]) (i.e. ~[*Japanese*] [*octopus*] [*how*] [*say*])
  ⇒ [日][文] [章][魚] [怎][麼] [說]) (i.e. ~[*Japan*] [*culture*] [*chapter*] [*fish*] ...)

Bulk PDF Parsing

PDFs, and often individual pages of PDFS, are easily separable and independently processable. This means tokenizing the sentences or words in these documents is stupidly parallelizable and easy to distribute.

To handle this task one can either stream PDF files individually, or read them from a distributed store. The latter is easy to setup and easily accessible through Amazon S3.

Multi-Word Expression -Types

Collocations: multiword expressions whose component words can be syntactically rearranged but (generally) cannot be substituted.

- *heavy smoker* ≈ "*He smoked heavily*.", but ≠ "*He was a <u>furious</u> smoker*".
- *strong accent* ≈ "*His accent is strong*", but ≠ "*His has a <u>rigid</u> accent*."

Idioms: multiword expressions that appears to be a phrase but whose meaning differs from the literal meaning.

- "*He will <u>kick the bucket</u> soon.*" and "*He <u>kicked the bucket</u>*". (contrast "He *dreads the bucket*")
- "*They <u>keep tabs on</u> him.*", "*<u>Tabs were kept on</u> him*". (contrast: "*put tabs on him*")

Word Segmentation - Approaches

Heuristic

- manually create a lexicon: *blue ray disks, ice cream, cream*; *snow shoes, shoes*; ...

Data-Driven

- mutual information - `word2phrase.c`; `nltk.collocations`

phrasing (eg.)

…

orlando_magic hair_bow on headband baby toddler nbafree shipping

washington_nationals hair_bow on headband baby toddler ***mlbfree_shipping***

john_deere hair_bow on headband baby toddler **free_shipping**

los_angeles_kings hair_bow on headband baby nhl **free_shipping**

sun_shade hat ***spf_upf*** microfiber removable face **neck_flap hat_cap** white

diesel leather betal belt in dk brown size **made_in** italy

women's x **kim_rogers** white knit shrug sweater cardigan lines

lane_bryant women s stretch satin dress with belt wot

nwts **vera_bradley_priscilla** pink clare shoulder crossbody hipster

nike_air_zoom_turf sz white silver photo blue black **barry_sanders**

...

identify multi-token terms

```
1: sterling silver multi color cz flower ring size 8
2: 14k yellow gold 0 01 ct diamond accent heart locket with chain
3: sterling silver 4mm carnelian bead necklace 16
4: 14k gold round wedding band sz 7 5
5: celtic irish clover design real pewter pendant necklace
6: horse buggy lapel pin
7: 925 sterling silver southwestern style ring size 8
8: rembrandt charms cowboy boot charm 14k yellow gold
```

multi-token terms w/ word2phrase

1: **sterling_silver multi_color** cz flower ring **size_8**

2: 14k yellow gold 0 01 ct diamond accent heart locket with chain

3: **sterling_silver** 4mm carnelian bead necklace 16

4: 14k gold round **wedding_band** sz 7 5

5: celtic **irish_clover** design real pewter pendant necklace

6: **horse_buggy lapel_pin**

7: 925 **sterling_silver southwestern_style** band mens ring **size_8**

8: rembrandt charms cowboy boot charm 14k yellow gold

code.google.com/p/**word2vec**/

```
/word2vec/trunk$ wc -l *.c
  143 compute-accuracy.c
  142 distance.c
  292 word2phrase.c
  702 word2vec.c
  144 word-analogy.c
```

Pointwise Mutual Information

```
>>> from nltk.collocations import *
>>> bigram_measures = nltk.collocations.BigramAssocMeasures()
>>> finder=BigramCollocationFinder.from_words(nltk.corpus.genesis.words
('english-web.txt'))
>>> finder.apply_freq_filter(3)
>>> finder.nbest(bigram_measures.pmi, 10)
[(u'gray', u'hairs'),
 (u'Most', u'High'),
 (u'ewe', u'lambs'),
 (u'many', u'colors'),
 (u'burnt', u'offering'),
 (u'east', u'wind'),
 (u'living', u'creature')]
```

Replace terms with gazetteer label

```
cat AMA-2015-02-02.JW.tsv | \
   perl -sne \
     'open DICT, "< $dict" ; \
      while (<DICT>)  {($new, $old) = split ;
         $regex_raw{$old} = $new;}\

      $olds = join "|", keys %regex ;  \
      while (<STDIN>) {
         s/\b($olds)\b/ $regex{$1} /g; \

      }' \
      -- -dict=lexical_SEMANTIC_GROUPS.150219.tsv \
  > AMA-2015-02-02.JW.$dstamp.tsv
```

Replace terms with gazetteer label

```
cat AMA-2015-02-02.JW.tsv | \
   perl -sne \
     'open DICT, "< $dict" ; \
      while (<DICT>) {($new, $old, $tokens) = split ;
         $regex_raw{$tokens}{$old} = $new;}\
         $tok1olds = join "|", keys %{$regex_raw{1}} ;  \
         $tok2olds = join "|", keys %{$regex_raw{2}} ;  \
         while (<STDIN>) {
            s/\b($tok2olds)\b/ $regex_raw{2}{$1} /g ;
            s/\b($tok1olds)\b/ $regex_raw{1}{$1} /g ;
         }' \
         -- -dict=lexical_SEMANTIC_GROUPS.150224.tsv \
  > AMA-2015-02-02.JW.$dstamp.tsv
```

Prebuilt Lexicons

- WordNet (next slide)
- Wiktionary
  - wiktionary.org/wiki/Category:en:Gems
- english common words
- BabelNet: a **multilingual encyclopedic dictionary**, with lexicographic and encyclopedic coverage of terms
- Metathesaurus

WordNet-based Inventory

wordnetweb.princeton.edu/perl/webwn?s=**rounding**

- S: (n) **rounding**, rounding error ((mathematics) a miscalculation that results from rounding off numbers to a convenient number of decimals) *"the error in the calculation was attributable to rounding";* …

wordnetweb.princeton.edu/perl/webwn?s=**integer**

- S: (n) **integer**, whole number (any of the natural numbers (positive or negative) or zero) *"an integer is a number that is not a fraction"*

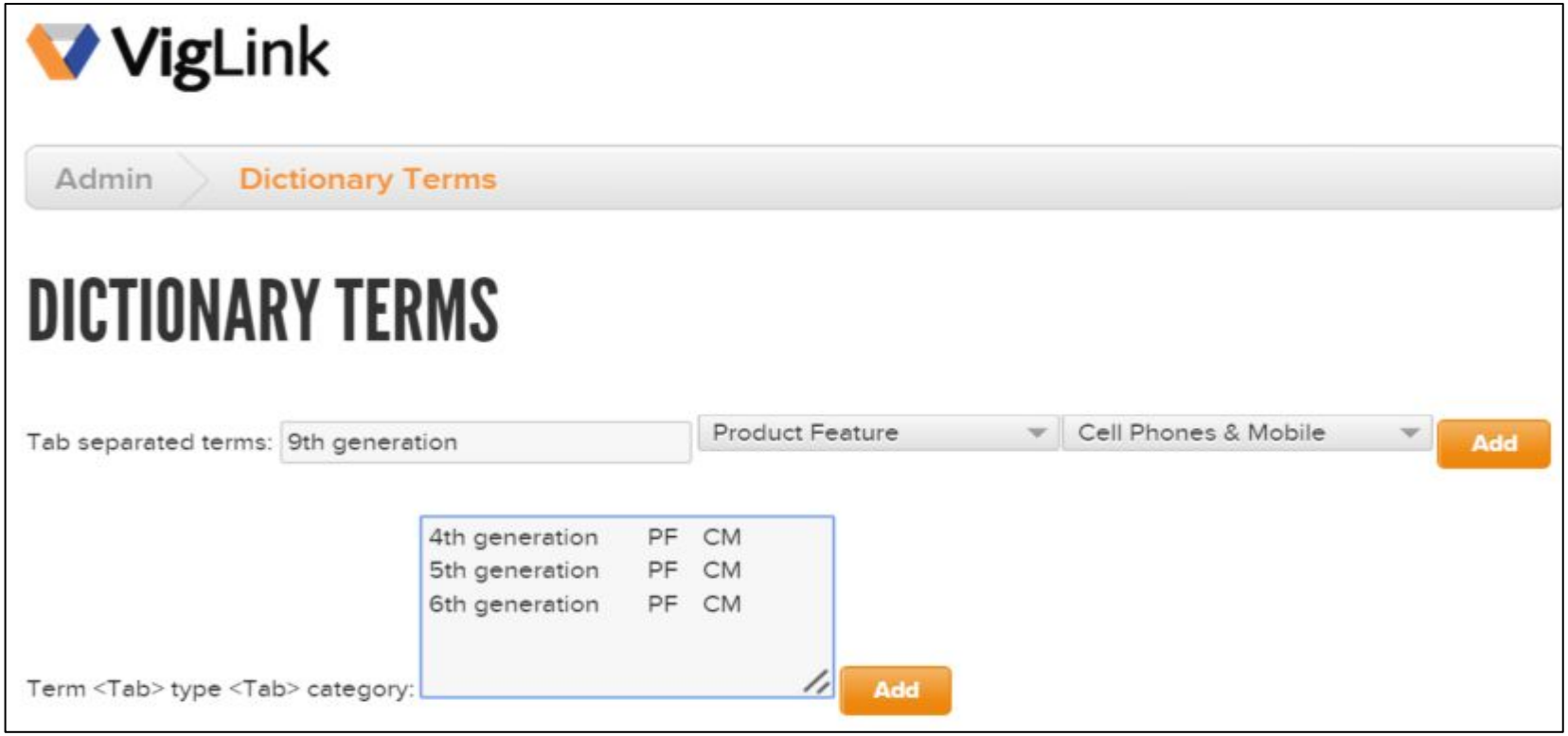
wordnetweb.princeton.edu/perl/webwn?s=**linear**

- S: (adj) **linear**, additive (designating or involving an equation whose terms are of the first degree)
- S: (adj) **linear**, one-dimensional (of or in or along or relating to a line; involving a single dimension) *"a linear measurement"*
- S: (adj) **linear**, running (measured lengthwise) *"cost of lumber per running foot"*

wordnetweb.princeton.edu/perl/webwn?s=**programs**

- S: (n) plan, program, programme (a series of steps to be carried out or goals to be accomplished) *"they drew up a six-step plan"; "they discussed plans for a new bond issue"*
- S: (n) program, programme (a system of projects or services intended to meet a public need) *"he proposed an elaborate program of public works"; "working mothers rely on the day care program"*
- S: (n) program, programme, computer program, computer programme ((computer science) a sequence of instructions that a computer can interpret and execute) *"the program required several hundred lines of code*

Maintaining a Custom Lexicon



CPROD1 Lexicon

#Text	Type
Assault Racing	BRANDNAME
CellShock	BRANDNAME
Lebez	BRANDNAME
Mango Tango	BRANDNAME
discourage	ENCOMMONWORD
marquisette	ENCOMMONWORD
skies	ENCOMMONWORD
whole	ENCOMMONWORD
if	GRAMMATICALWORD
last	GRAMMATICALWORD
then	GRAMMATICALWORD

Discounttiredirect	MERCHANT
ebay	MERCHANT
TireRack	MERCHANT
hdtv	PRODUCTCAT
memory stick	PRODUCTCAT
motorcycle	PRODUCTCAT
timing belt	PRODUCTCAT
rechargable	PRODUCTFEAT
refurbed	PRODUCTFEAT
usb3	PRODUCTFEAT

github.com/CPROD/cprod-baselines/tree/master/resources/data

Word Stemming - Task

- WSIT(*wolves, went*)
  ⇒ `wolv went.`

- WSIT(*The wolves went to her ex-sisters-in-law's pseudocottage*)
  ⇒ `the wolv went to her exsistersinlaw s pseudocottag`

- WSIT(*The ex-governor general 's sisters-in-law saw the wolves' den near Mr. Smith's home in Sault, Ste. Marie)*
  ⇒ `the exgovernor general s sistersinlaw saw the wolv den near mr smith s home in,sault ste mari.`

Word Stemming - Approaches

- Regular Expressions
- Porter Stemmer.
- Snowball Stemmer.
- Lingua Stem.
- Natural Language Tool Kit.

Lemmatisation

- Lemmatise("*wolves*") ⇒ *wolf*.
- Lemmatise("*went*") ⇒ *go*
- Lemmatise("*lobos*") ⇒ *lobo*.
- Lemmatise("*He is running*") ⇒ *run*.

Text Segmentation Systems

- NLTK
- OpenNLP
- LingPipe
- SRILM
- MALLET
- Stanford Core NLP Toolkit
- ...

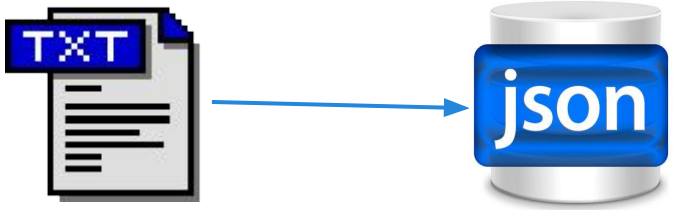
We are here

- ~~Overview~~
- ~~Text Preprocessing~~
- **>> Syntactic Parsing <<**
- Big NLP Framework Demos
- Shallow Semantic Analysis
- Text Classification & Clustering
- Information Extraction
- Word Relatedness Functions

Syntactic Parsing

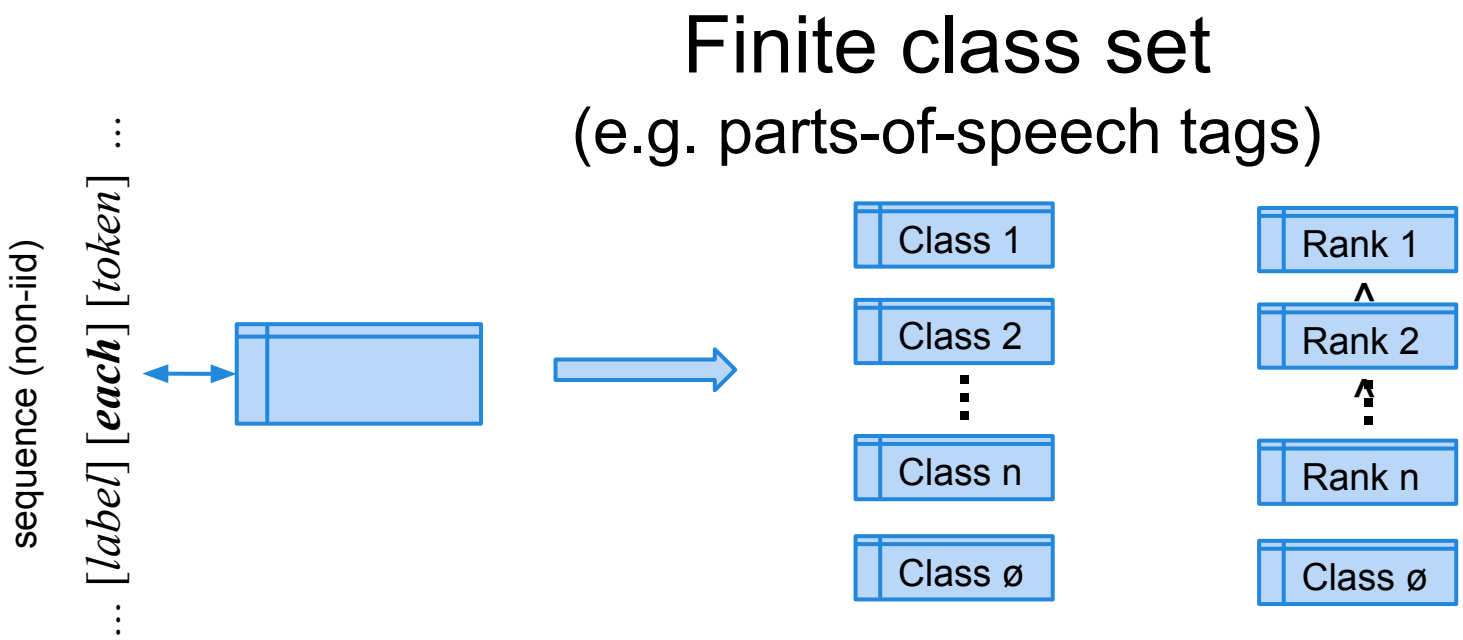
- Part-of-Speech Tagging
- Phrase Chunking
- Syntactic Tree Parsing
- Dependency Tree Parsing

Reminder of the NLP Pipeline



- ~~Segmenting~~
- **Parsing**
- Classification
- Recognition

Classification (vs. Scoring/Estimation)



Part-of-Speech Tags (English)

- CC - Coordinating conjunction
- CD - Cardinal number
- DT - Determiner
- EX - Existential there
- FW - Foreign word
- IN - Preposition
- JJ - Adjective
- JJR - Adjective, comparative
- JJS - Adjective, superlative
- LS - List item marker
- MD - Modal
- NN - Noun, singular or mass
- NNS - Noun, plural
- NNP - Proper noun, singular
- NNPS - Proper noun, plural
- PDT - Predeterminer
- POS - Possessive ending
- PRP - Personal pronoun
- PRP$ - Possessive pronoun
- RB - Adverb
- RBR - Adverb, comparative
- RBS - Adverb, superlative
- RP - Particle
- SYM - Symbol
- TO - to
- UH - Interjection
- VB - Verb, base form
- VBD - Verb, past tense
- VBG - Verb,present participle
- VBN - Verb, past participle
- VBP - Verb, non-3rd person ..
- VBZ - Verb, 3rd person ...
- WDT - Wh-determiner
- WP - Wh-pronoun
- WP$ - Possessive wh-pronoun
- WRB - Wh-adverb

Part-of-Speech Tagging

PoSTT("*Colorless green ideas sleep furiously.*")

⇒ `Colorless/JJ green/JJ ideas/NNS sleep/VBP furiously/RB ./.`

PoSTT("*This automatic teller machine is a work in progress.*")

⇒ `This/DT automatic_teller_machine/NN is/AUX a/DT work_in_progress/NN ./.`

PoSTT("*The duchess was entertaining last night.*")

⇒ `The/DT duchess/NN was/AUX entertaining/JJ|VBN last/JJ night/NN ./.`

```
I/PRP 'm/VBP buying/VBG a/DT real/JJ time/NN operating/VBG system/NN
I/PRP 'm/VBP buying/VBG a/DT real_time/JJ operating_system/NN
```

Shallow Parsing - Phrase Chunking

NP Noun Phrase
VP Verb Phrase
PP Prepositional Phrase
ADVP Adverb Phrase
SBAR Clause introduced by subordinating conjunction

Phrase Chunking

cogcomp.cs.illinois.edu/page/run_demo/ShallowParse

[NP Helicopters] [VP will] [NP patrol]
[NP the temporary no-fly zone] [PP around]
[NP New Jersey] [NP 's MetLife Stadium Sunday] ,
[PP with] [NP F-16s] [VP based] [PP in]
[NP Atlantic City] [ADJP ready] [VP to be scrambled]
[SBAR if] [NP an unauthorized aircraft]
[VP does enter] [NP the restricted airspace] .

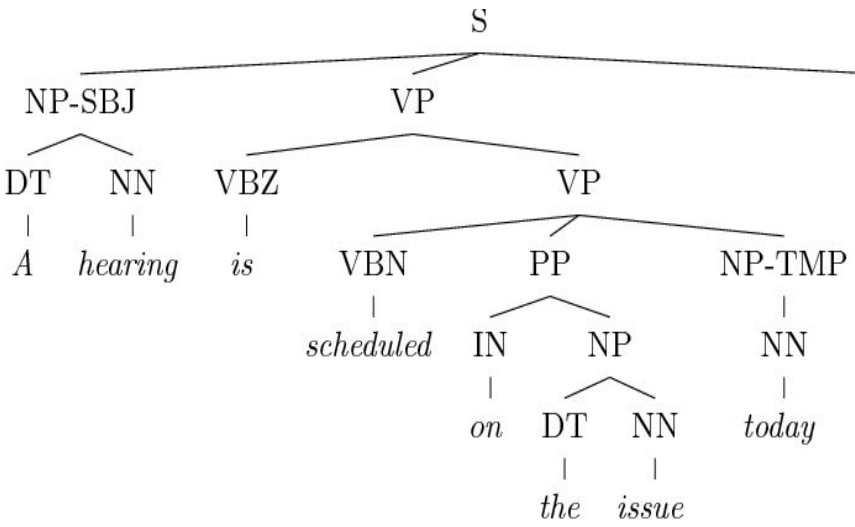
Chunking Data

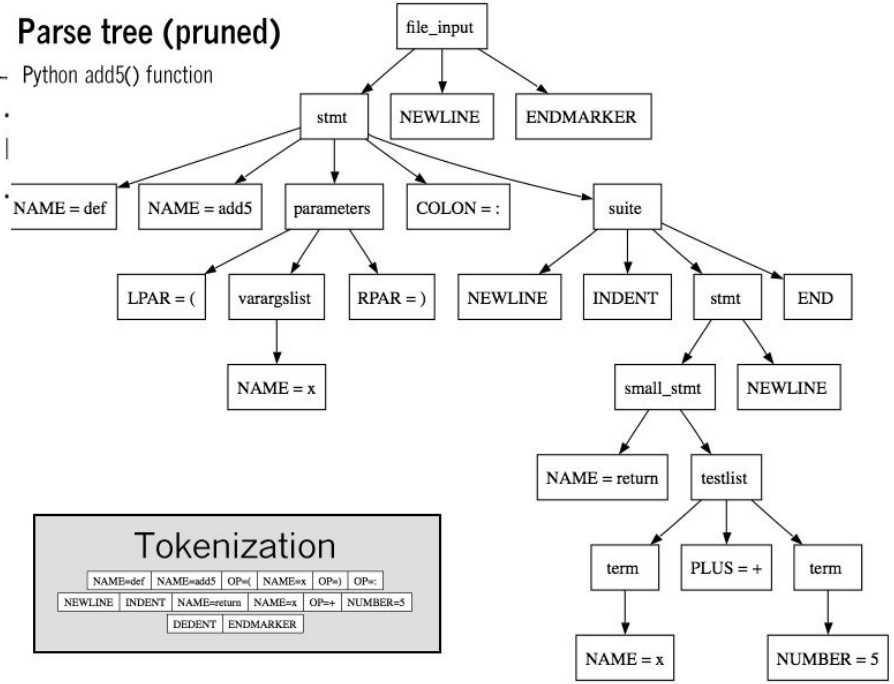
www.cnts.ua.ac.be/conll2000/chunking/

```
He          PRP   B-NP
reckons     VBZ   B-VP
the         DT    B-NP
current     JJ    I-NP
account     NN    I-NP
deficit     NN    I-NP
will        MD    B-VP
narrow      VB    I-VP
```

```
to          TO    B-PP
only        RB    B-NP
#           #    I-NP
1.8         CD    I-NP
billion     CD    I-NP
in          IN    B-PP
September   NNP   B-NP
.           .    O
```

Deep Syntactic Tree Parsing





Parsing with NLTK

```
sentence_tokens_ishot = [ 'I', 'shot', 'an', 'elephant', 'in', 'my', 'pajamas']

parser = nltk.ChartParser(groucho_grammar)

for tree in parser.parse(sentence_tokens_ishot ):
   print(tree)
```

```
(S
  (NP I)
  (VP
    (VP (V shot) (NP (Det an) (N elephant)))
    (PP (P in) (NP (Det my) (N pajamas)))))
(S
  (NP I)
  (VP
    (V shot)
    (NP (Det an) (N elephant) (PP (P in) (NP (Det my) (N pajamas))))))
```

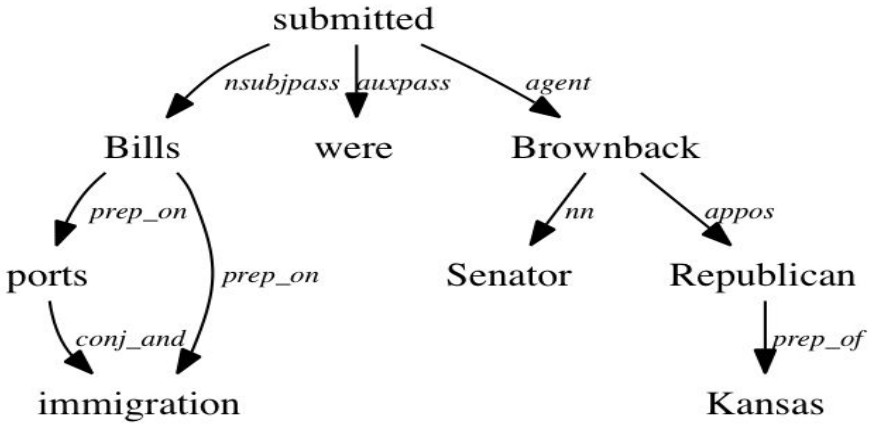
nlp.stanford.edu:8080/parser/index.jsp

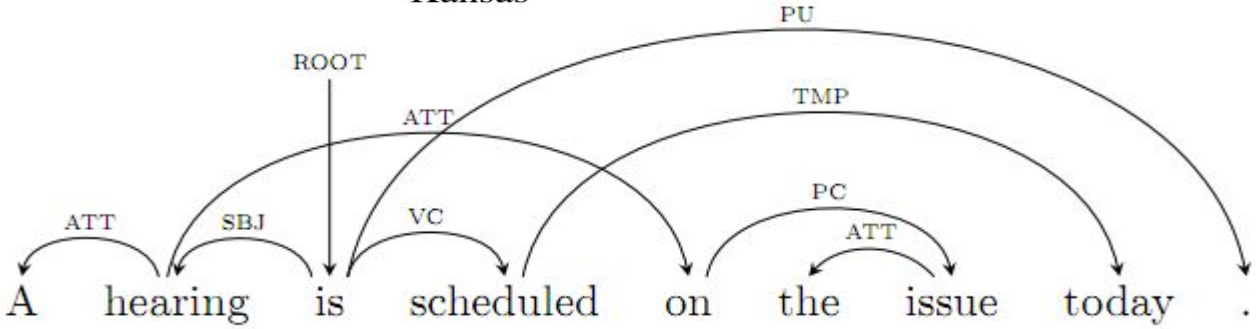
```
(ROOT
  (S
    (NP (PRP I))
    (VP (VBP 'm)
      (VP (VBG buying)
        (NP (DT a) (JJ real_time) (NN operating_system))))))
```

NLSPT(*"Alexander went to the park."*, *English Syntax*, *English Lexicon*)

⇒ `(S (NP (NNP Alexander))(VP (VBD went)(PP (TO to) (NP (DT the) (NN park)))) (. .)))`

Dependency Tree Parsing





SyntaxNet: *Parsey McParseface* model

THE WALL STREET JOURNAL.

Home World U.S. Politics Economy Business Tech Markets Op

Updated May 12, 2016 7:13 p.m. ET

Meet Parsey McParseface, a whimsically named computer program that helps machines u

Alphabet Inc. subsidi
SyntaxNet—the softw
for anyone to distribu
on Thursday.

"Our hope is that peo
own," said Dave Orr, S

models/syntaxnet at mast...

GitHub, Inc. (US) https://github.com/ter

Search

SyntaxNet: Neural Models of Syntax.

A TensorFlow implementation of the models described in Andor et al. (2016).

At Google, we spend a lot of time thinking about how computer systems can read and understand human language in order to process it in intelligent ways. We are excited to share the fruits of our research with the broader community by releasing SyntaxNet, an open-source neural network framework for TensorFlow that provides a foundation for Natural Language Understanding (NLU) systems. Our release includes all the code needed to train new SyntaxNet models on your own data, as well as *Parsey McParseface*, an English parser that we have trained for you, and that you can use to analyze English text.

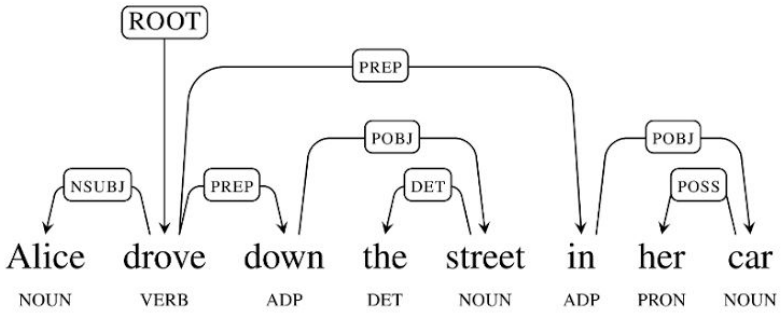
SyntaxNet: *Parsey McParseface* model

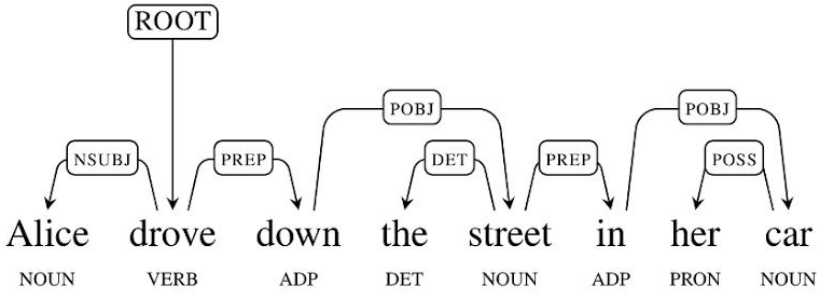
So, how accurate is Parsey McParseface? For this release, we tried to balance a model that runs fast enough to be useful on a single machine (e.g. ~600 words/second on a modern desktop) and that is also the most accurate parser available. Here's how Parsey McParseface compares to the academic literature on several different English domains: (all numbers are % correct head assignments in the tree, or unlabelled attachment score)

Model	News	Web	Questions
Martins et al. (2013)	93.10	88.23	94.21
Zhang and McDonald (2014)	93.32	88.65	93.37
Weiss et al. (2015)	93.91	89.29	94.17
Andor et al. (2016)*	94.44	90.17	95.40
Parsey McParseface	94.15	89.08	94.77

We see that Parsey McParseface is state-of-the-art; more importantly, with SyntaxNet you can train larger networks with more hidden units and bigger beam sizes if you want to push the accuracy even further: Andor et al. (2016)* is simply a SyntaxNet model with a larger beam and network. For futher information on the datasets, see that paper under the section "Treebank Union".

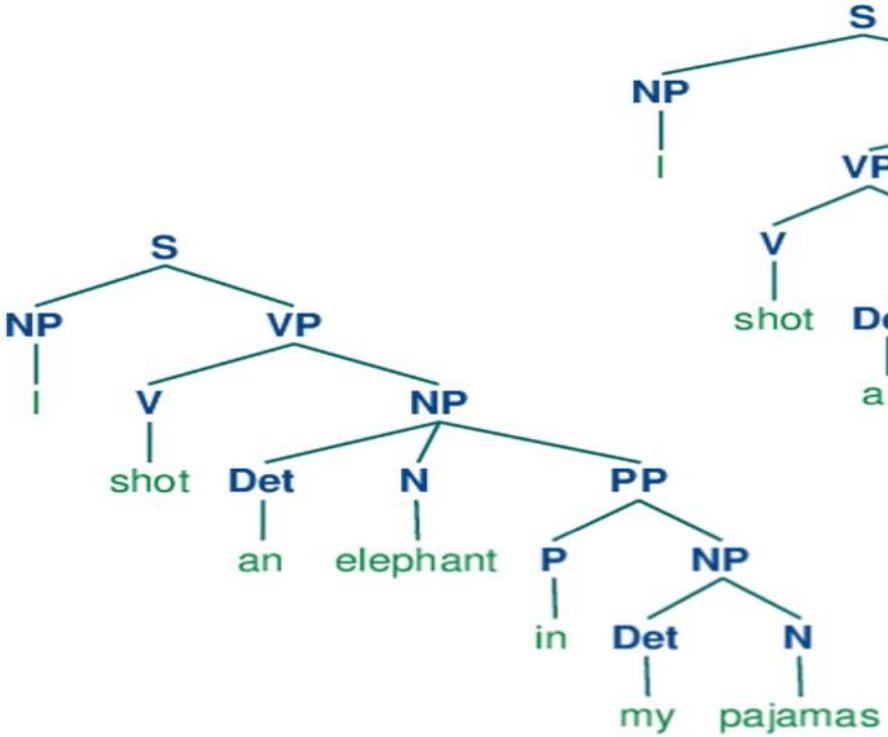
Prepositional Phrase Attachment

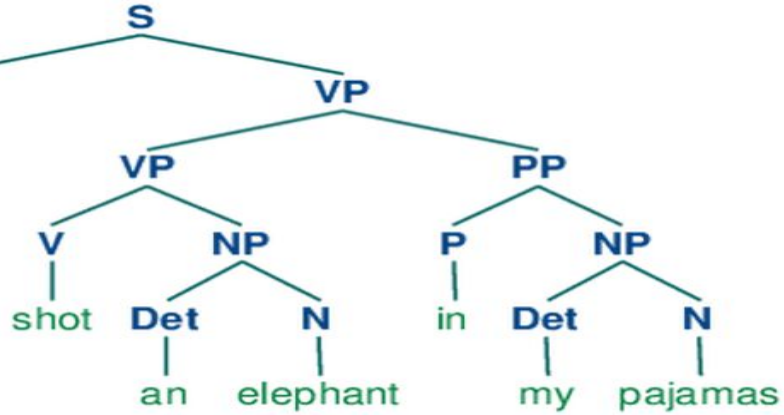


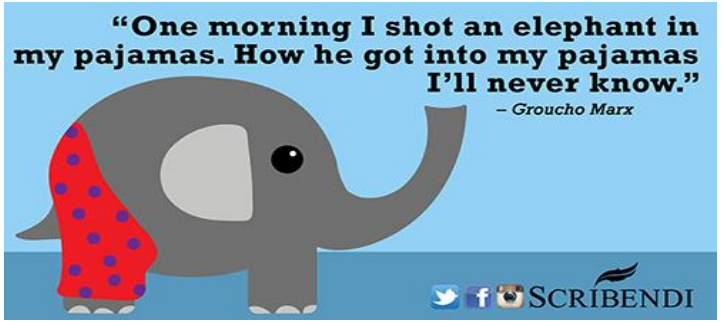


"The first corresponds to the (correct) interpretation where Alice is driving in her car; the second corresponds to the (absurd, but possible) interpretation where the street is located in her car ... , it is very important to use beam search in the model. Instead of simply taking the first-best decision at each point, multiple partial hypotheses are kept at each step, with hypotheses only being discarded when there are several other higher-ranked hypotheses under consideration. …"

I shot an elephant in my pajamas







We are here

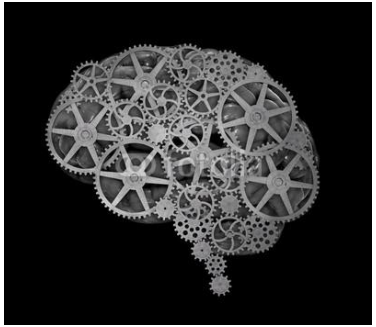
- ~~Overview~~
- ~~Text Preprocessing~~
- ~~Syntactic Parsing~~
- ~~Big NLP Frameworks~~
- **>> Shallow Semantic Analysis <<**
- Text Classification & Clustering
- Information Extraction
- Word Relatedness Functions

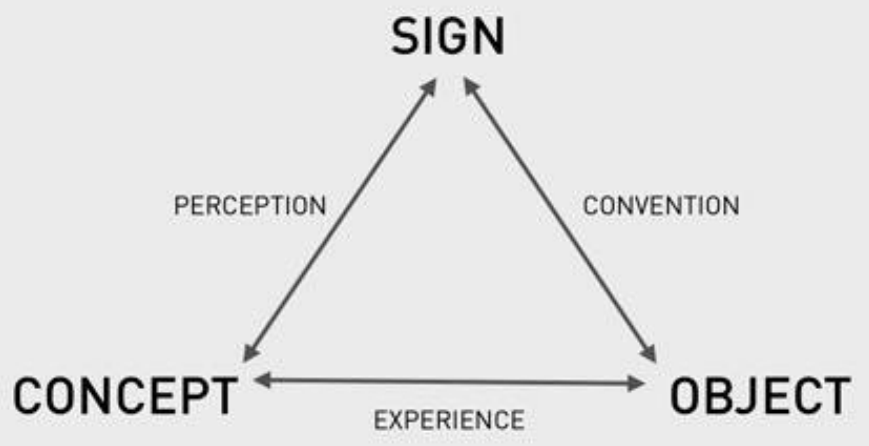
Shallow Semantic Analysis

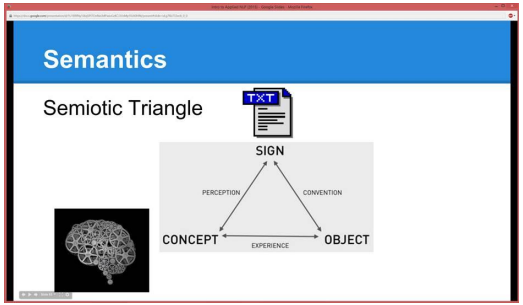
- Semiotic Triangle
- Named Entity Recognition
- Number Quantification
- Semantic Role Labeling
- Relation Mention Recognition
- Product Offer Parsing

Semiotic Triangle

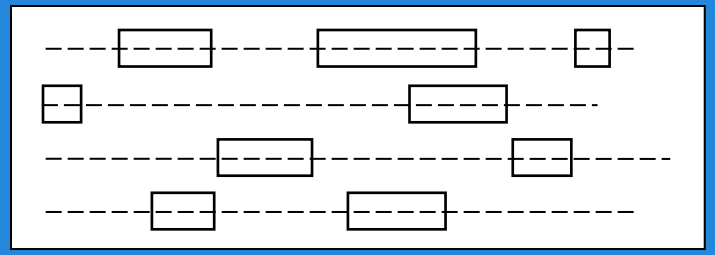
(Odgen & Richards, 1923), …, Aristotle

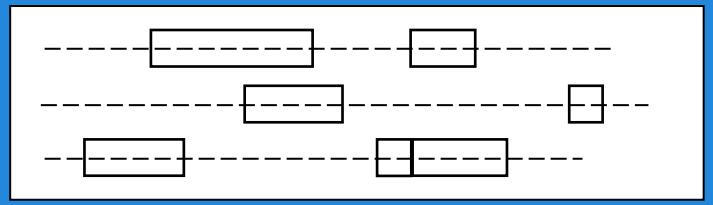






Entity Mention Recognition





Named-Entity Recognition

MUC: PERSON/LOCATION/ORGANIZATION

- *The ex-governor general's sisters-in-law saw the wolves' den near Mr. Smith's home in Sault Ste. Marie.*
- *The ex-governor general's sisters-in-law saw the wolves' den near Mr. Smith's home in Sault Ste. Marie.*
- *The ex-governor general's sisters-in-law saw the wolves' den near Mr. Smith*$_{\text{PERSON}}$*'s home in Sault Ste. Marie*$_{\text{LOCATION}}$.

Predictor Features (for text)

A *feature* is a function intended to provide a signal to a prediction task.

Text features take a text item as input.

- *f*(`hasCapitalL`("*Markov*")) ⇒ 1
- *f*(`equals`("*Markov*","*Jordan*")) ⇒ 0
- *f*(`charPattern`("*Machine-223*") ⇒ `Aaaaaaa-000`
- *f*("*rko*", "*Markov*") ⇒ true

Simple Features

- `FCUC:` First character type (UC/lc/num/oth)?
- `UCCNT:` Count of upper-case letters in token
- `LCCNT:` Count of lower-case letters in token
- `NUMCNT:` Count of numbers in token
- `DSHCNT:` Count of dashes in token
- `GRWRDCNT:` Grammatical word count
- `BRWRDCNT:` Brand word count
- `ENWRDCNT:` English word count.

Token	POS	CAP	sChar	fourC	O_1	O_2	$O_{>2}$	Label
$f_1(x)$	$f_2(x)$	$f_3(x)$	$f_4(x)$	$f_5(x)$	$f_6(x)$	$f_7(x)$	$f_8(x)$	$f(y)$
Collaborative	JJ	1	0	0	0	1	1	**B**
filtering	VBG	0	0	0	1	1	1	**I**
is	AUX	0	0	1	0	0	1	O
the	DT	0	0	1	0	0	1	O
most	RBS	0	0	0	0	0	0	O
popular	JJ	0	0	0	0	0	0	O
approach	NN	0	0	0	1	1	1	**B**
to	TO	0	0	1	0	0	0	O
build	VB	0	0	0	0	0	0	O
recommender	NN	0	0	0	0	1	1	**B**
systems	NNS	0	0	1	1	1	1	**I**
and	CC	0	0	1	0	0	1	O
has	AUX	0	0	1	0	0	0	O
been	AUX	0	0	0	0	0	0	O
successfully	RB	0	0	0	0	0	0	O
employed	VBN	0	0	0	0	0	0	O
in	IN	0	0	1	0	0	0	O
many	JJ	0	0	0	0	0	0	O
applications	NNS	0	0	0	1	0	0	**B**
.	.	0	1	1	0	0	0	O

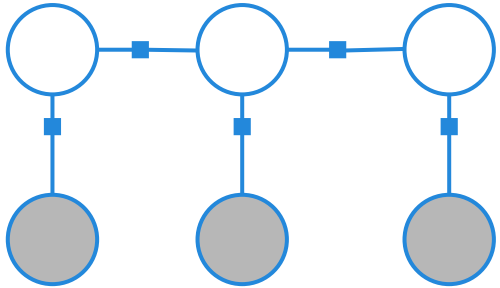
The "Label" Feature

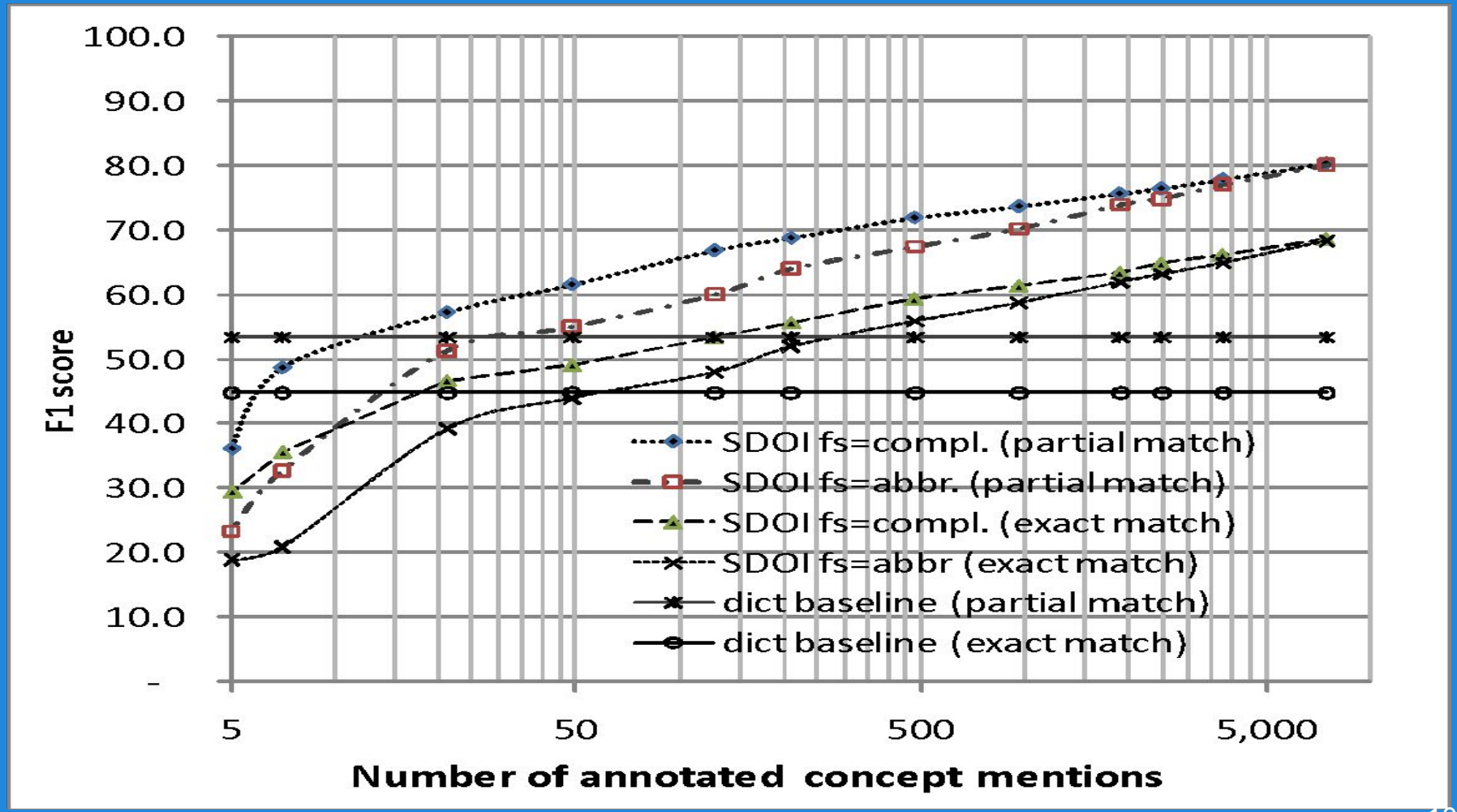
Token			
CNET	no	O	B-ORG
tested	no	O	O
the	no	O	O
Epson	yes	B	B-PROD
83500	yes	I	I-PROD
,	no	O	O
TW32000	yes	B	B-PROD
and	no	O	O
VS-400	yes	B	B-PROD
.	no	O	O
However	no	O	O
,	no	O	O
...	...	...	...

t_i	Token	$\Rightarrow f()$?	$\Rightarrow f()$?	$\Rightarrow f()$?	$\Rightarrow f()$?	$\Rightarrow f()$?	$\Rightarrow f()$?
1	Mr.	3	1	0	1	0	O
2	Toro	4	1	0	0	1	O
3	"	1	0	0	1	1	O
4	announced	9	0	0	0	0	O
5	"	1	0	0	1	0	O
6	Sony	4	1	0	0	0	O
7	's	2	0	0	1	1	O
8	new	3	0	0	0	0	O
9	Vaio	4	1	0	0	0	O
10	(	1	0	0	1	1	O
11	VPC-F223FX/S	12	1	1	1	0	B
12	)	1	0	0	1	1	O
13	today	5	0	0	0	0	O
14	.	1	0	0	1	0	O
15		0	0	0	0	0	O
16	It	2	1	0	0	0	O
17	compares	8	0	0	0	1	O
18	well	4	0	0	0	0	O
19	to	2	0	0	0	0	O
20	the	3	0	0	0	0	O
21	iPad	4	0	0	0	0	B
22	2	1	0	1	0	0	I
23	.	1	0	0	1	0	O
24		0	0	0	0	0	O

Conditional Random Fields

- MALLET
- CRFpp




100.0
90.0
80.0
70.0
60.0
50.0
40.0
30.0
20.0
10.0
-
F1 score
5
50
500
5,000
Number of annotated concept mentions
SDOI fs=compl. (partial match)
SDOI fs=abbr. (partial match)
SDOI fs=compl. (exact match)
SDOI fs=abbr (exact match)
dict baseline (partial match)
dict baseline (exact match)

Shallow Semantic Parsing

- Semantic Role Labeling.
- Product Title Shallow-Parsing.
- Relation Extraction from Text.

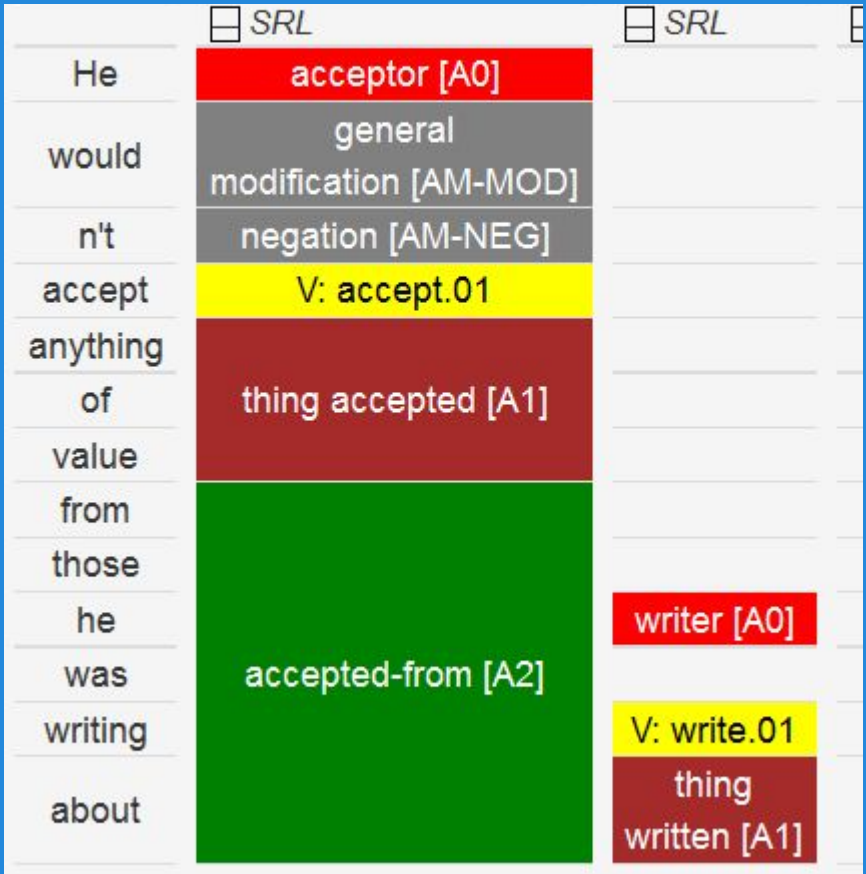
Semantic Role Labeling

- SRL("*He **is** Swiss and she **is** Dutch.*")

  ⇒ ([A0 He] **[V is]** [A1 Swiss]); ([A0 she] **[V is]** [A1 Dutch]).

- SRL("*He wouldn't **accept** anything of value from those he was writing about.*")

  ⇒ ([A0 He] [AM-MOD would] [AM-NEG n't] **[V accept]** [A1 anything of value] from [A2 those he was writing about]).



Verb

V verb

Arguments

A0 subject

A1 object

A2 indirect object

Other

C-arg continuity of an argument/adjunct of type

R-arg reference to an actual argument/adjunct

Adjunct

AM-ADV adverbial modification

AM-DIR direction

AM-DIS discourse marker

AM-EXT extent

Synset Corpus Extraction

Given text data in sentence structures, we can extract Synset Distributions which are translatable into Information Content.

$$IC(c) = -\log(P(c))$$

```
[(u'parser.n.01', 1.0),
 (u'work.n.01', 0.02404719589263042),
 (u'work.n.02', 0.01827586887839912),
 (u'employment.n.02', 0.035145901689229084),
 (u'study.n.02', 0.02404719589263042),
 (u'work.n.05', 0.028556045122498628),
 (u'workplace.n.01', 0.030459781463998536),
 (u'oeuvre.n.01', 0.028556045122498628),
```

This is useful for many semantic similarity methods (Resnik, Lin, Conrath, etc).

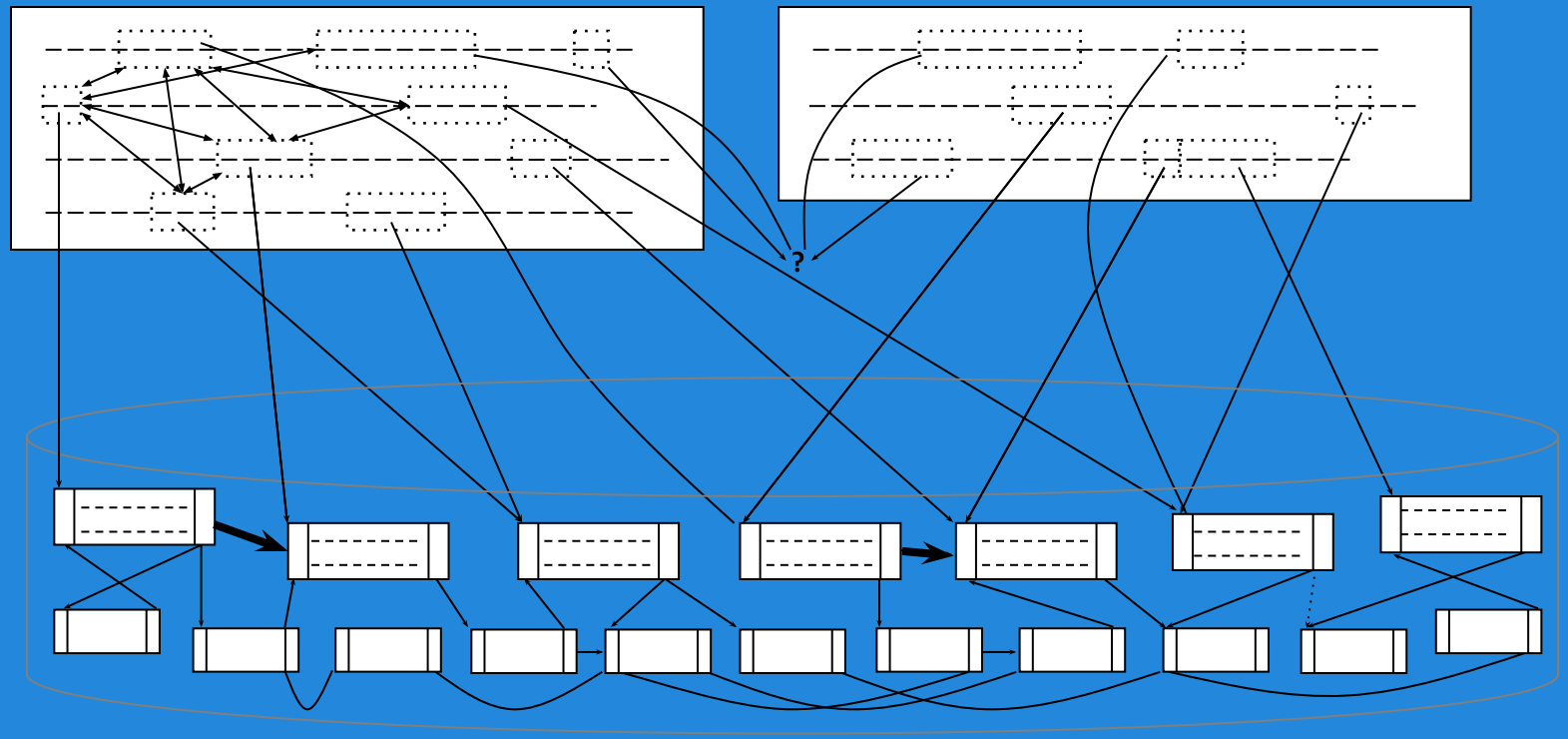
Relation Mention Recognition

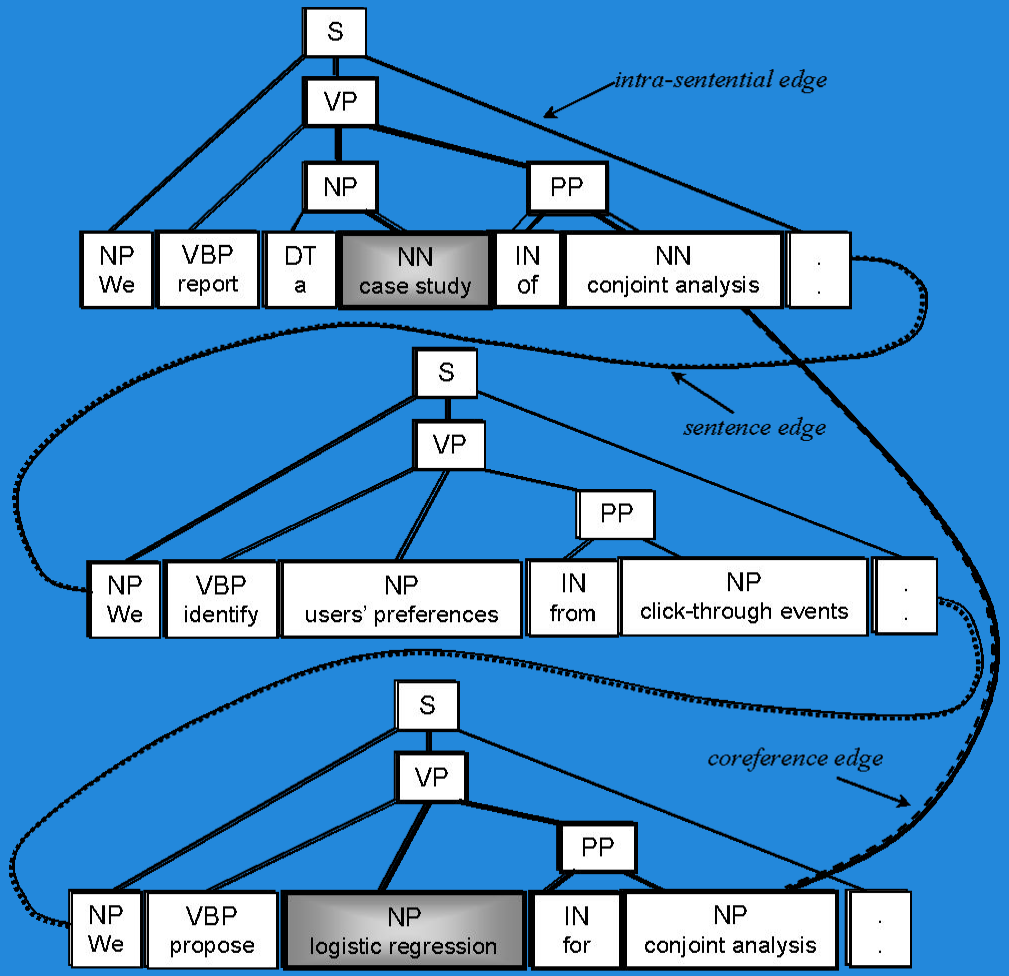
- Are two concepts mentioned in a document also linked within the ontology: True or False?

- A binary relation mention recognition task

- Repurpose TeGRR algorithm (Melli & al, 2007)
- Makes use of a Text-Graph representation

Relation Mention Recognition




S
VP
NP
PP
intra-sentential edge
NP We
VBP report
DT a
NN case study
IN of
NN conjoint analysis
. .
S
VP
PP
sentence edge
NP We
VBP identify
NP users' preferences
IN from
NP click-through events
. .
S
VP
PP
coreference edge
NP We
VBP propose
NP logistic regression
IN for
NP conjoint analysis
. .

Training Dataset

- Each candidate relation mention is associated with a feature vector based on (Melli & al, 2007) which generalizes (Jiang and Zhai, 2007)'s proposal for within sentence relations.

Relation Mention			TeGRR Text-Graph based	Concept Mention (CM) Linking based		Ontology based	label
doc_d	m_i	m_j		CM_a	CM_b		
							T
			Feature Space				F
							...

Concept mentioned	Concept mentioned	Abstracts that mention the pair	Present in Ontology
Research Paper	Algorithm	61	FALSE
Research Paper	Experiment Outcome	33	FALSE
Task	Algorithm	32	TRUE
Algorithm	Experiment Outcome	29	FALSE
Research Paper	Data Set	29	FALSE
Algorithm	Data Set	28	FALSE
Research Paper	Task	24	FALSE
Data Value Set	Algorithm	23	FALSE
Algorithm	Real-World Dataset	21	FALSE
Research Paper	Real-World Dataset	21	FALSE
Research Paper	Efficient Algorithm	20	FALSE
Experiment Outcome	Algorithm	19	FALSE
Algorithm	Efficient Algorithm	18	TRUE
Task	Experiment Outcome	18	FALSE
Data Mining Task	Algorithm	17	FALSE
Research Paper	Experimental Evaluation	17	FALSE
Algorithm	State-of-the-Art-Algorithm	13	TRUE
Dataset	Real-World Dataset	11	TRUE
Synthetic Dataset	Real-World Dataset	11	TRUE
...	...	...	...
Clustering Algorithm	Cluster Set	8	TRUE
Classification Model	Accuracy Measure	8	TRUE
...	...	...	...

Listing of some of the more common pairs of concepts mentioned in `kdd09cma1`

Case Study: VigLink Product Offer Parsing

Polk Audio Blackstone TL1600 Speaker sys - home theater - 5.1-CH - wired

[Polk Audio]^{BN} Blackstone^{PL} TL1600^{PI} [Speaker sys]^{PC} [home theater]^{PF} [5.1-CH] ^{PF} [wired]^{PF}

Brand Name (BN): *Polk Audio*
Product Line (PL): *Blackstone*
Product (PI): *TL 1600*
Product Category (PC): *speaker sys*
Product Feature (PF): *home theater*
Product Feature (PF): *5.1-CH*
Product Feature (PF): *wired*

Offer Feeds

ebay
amazon -->

~ 330M offer

Table 1: Examples of offering titles annotated using our framework. Square brackets indicate multi-token terms

Annotated product offering title	Industry
Tripp^{PI} *LS606M*^{PF} [*600 watt*]^{PC} [*Line Conditioner*]^{PF} [*6 Outlet*]^{PF} [*120 volt*]^{QQ}	Electronics
Samsung^{PF} *51-Inch*^{PF} *720p*^{PF} *600Hz*^{PF} *Plasma*^{PF} *HDTV*^{PC} (*Black*^{PF})	Electronics
[*Wheelskins*]^{PF} [*Genuine Deerskin*]^{PC} [*DRIVING GLOVES*]^{PC} - *Tan*^{PF} ([*Size Large*]^{PF})	Automotive
[*1990-1997*]^{PF} [*Mazda*]^{BN} [*Miata*]^{PL} *AxleBack*^{PL} [*Exhaust Bolt*]^{PC} *on*^{FT} *Muffler*^{PC}	Automotive
[*Cloudy with a Chance of Meatballs*]^{PI} (*Two-Disc*^{PF} [*Blu-ray / DVD Combo*]^{PF})	Entertainment
[*A Pius Man : A Holy Thriller*]^{PI} ([*The Pius Trilogy*]^{PL}) ([*PF|Volume 1*]^{PF})	Books
[*Amazing Herbs*]^{BN} [*Black Seed*]^{PF} [*Cold-Pressed Oil*]^{PC} - *32oz*^{PF}	Health & Beauty
([*180 days wrty.*]^{OF}) *ZeroLemon*^{BN} *LG*^{BN} *Nexus*^{PL} *4*^{PI} *Juicer*^{PL} [*Battery Case*]^{PC}	Mobile
[*WORLDS ONLY*]^{OF} *NEXUS*^{PL} *4*^{PI} [*BATTERY CASE*]^{PC} (*LG-N4-BattCase-black*^{PI})	Mobile
[*Kenneth Cole*]^{BN} *REACTION*^{PL} *men's*^{PF} *Cufflinks*^{PC}, *Silver*^{PF}, [*One Size*]^{PF}	Fashion

totken	First Char	Char Count	...	Brand Word?	Prod Cat Word?	Label
ViewSonic	FRSTCHR_UC	CHARCNT_9	...	BRNDWRDCNT_2	PRODCATWRDCNT_0	B-BN
3D	FRSTCHR_num	CHARCNT_2	...	BRNDWRDCNT_2	PRODCATWRDCNT_24	B-PF
HD	FRSTCHR_UC	CHARCNT_2	...	BRNDWRDCNT_4	PRODCATWRDCNT_18	B-PF
Video	FRSTCHR_UC	CHARCNT_5	...	BRNDWRDCNT_2	PRODCATWRDCNT_21	B-PF
Processor	FRSTCHR_UC	CHARCNT_9	...	BRNDWRDCNT_0	PRODCATWRDCNT_1	I-PF
(	FRSTCHR_oth	CHARCNT_1	...	BRNDWRDCNT_0	PRODCATWRDCNT_0	O
VP3D1	FRSTCHR_UC	CHARCNT_5	...	BRNDWRDCNT_0	PRODCATWRDCNT_0	B-P
)	FRSTCHR_oth	CHARCNT_1	...	BRNDWRDCNT_0	PRODCATWRDCNT_0	O
			...			
Netgear	FRSTCHR_UC	CHARCNT_7	...	BRNDWRDCNT_1	PRODCATWRDCNT_0	B-BN
85Mbps	FRSTCHR_num	CHARCNT_6	...	BRNDWRDCNT_0	PRODCATWRDCNT_0	B-PF
Powerline	FRSTCHR_UC	CHARCNT_9	...	BRNDWRDCNT_0	PRODCATWRDCNT_1	B-PL
Network	FRSTCHR_UC	CHARCNT_7	...	BRNDWRDCNT_4	PRODCATWRDCNT_23	B-PC
Adapter	FRSTCHR_UC	CHARCNT_7	...	BRNDWRDCNT_0	PRODCATWRDCNT_74	I-PC
Kit	FRSTCHR_UC	CHARCNT_3	...	BRNDWRDCNT_0	PRODCATWRDCNT_2	B-PF
=	FRSTCHR_oth	CHARCNT_1	...	BRNDWRDCNT_0	PRODCATWRDCNT_0	O
XETB1001	FRSTCHR_UC	CHARCNT_8	...	BRNDWRDCNT_0	PRODCATWRDCNT_0	B-P
			...			

Training Set	Test Set Category										
training set	All	CE	BK	HG	CM	HO	JW	HB	FS	AU	AE
All	**57.7**	**64.3**	41.2	**54.6**	**58.3**	48.0	**60.3**	**51.6**	55.0	46.7	48.8
CE	47.1	63.3	6.6	46.6	48.7	34.8	35.8	34.6	36.6	42.7	27.2
BK	11.1	4.0	**82.0**	2.8	4.2	2.6	2.1	3.5	3.8	2.8	37.5
HG	34.9	41.4	7.5	50.0	28.9	32.8	36.1	32.3	32.3	32.9	28.1
CM	41.2	48.9	9.1	39.7	56.5	29.8	32.5	33.1	31.0	35.7	31.5
HO	27.3	30.4	4.1	37.3	22.3	43.5	28.7	29.6	25.7	27.7	18.9
JW	27.1	30.2	7.2	31.8	19.5	23.9	56.2	29.3	31.9	26.1	15.9
HB	27.9	30.9	4.7	35.1	23.3	29.2	27.3	45.8	28.1	24.3	23.6
FS	27.0	32.0	5.8	31.7	22.1	23.1	32.1	27.6	52.3	25.0	27.8
AU	31.1	40.0	4.8	33.9	24.8	26.8	31.4	25.0	26.7	38.2	19.2
AE	13.3	9.0	59.2	6.3	7.5	7.4	4.3	9.7	11.7	4.6	61.3
BK&AE	12.9	6.2	65.2	3.6	6.0	4.0	3.0	5.2	4.9	3.3	**65.3**
not BK|AE	55.8	**64.6**	18.1	**54.3**	**59.0**	**49.3**	58.4	**52.1**	**56.3**	**49.2**	34.9

	graphics card	directional antenna	sleeping bag	am radio	messenger bag	gaming mouse	inkjet printer
9/1/2013	6						
9/2/2013	10						
9/3/2013	3						
9/4/2013	7						
9/5/2013	5						
9/6/2013	4						
9/7/2013	6						
9/8/2013	7						
9/9/2013	9			1			
9/10/2013	9					1	
9/11/2013	6						
9/12/2013	8						
9/13/2013	4	2					
9/14/2013	8						
9/15/2013	7	2	2				1
9/16/2013	5	6		1			
9/17/2013	5		3				1
9/18/2013	16	6	6	3	3		2
9/19/2013	28	2	3		3		2
9/20/2013	28	4	5	1	2	1	
9/21/2013	33	10	3	1	1	2	1
9/22/2013	35			1	2	4	
9/23/2013	16	6	6	3	1	3	

Demo to the OTP Product Offer Title Parser

This form demonstrates a product offering title parser.

There are two pre-populated product offering titles, either of which can be rewritten.

Sample 1: PYLE 300 Watt Stereo Amplifier

Submit Request

Sample 2: Sony VAIO Pro SVP13213CXS 13.3-Inch Core i5 Touchscreen Ultrabook (Carbon Silver

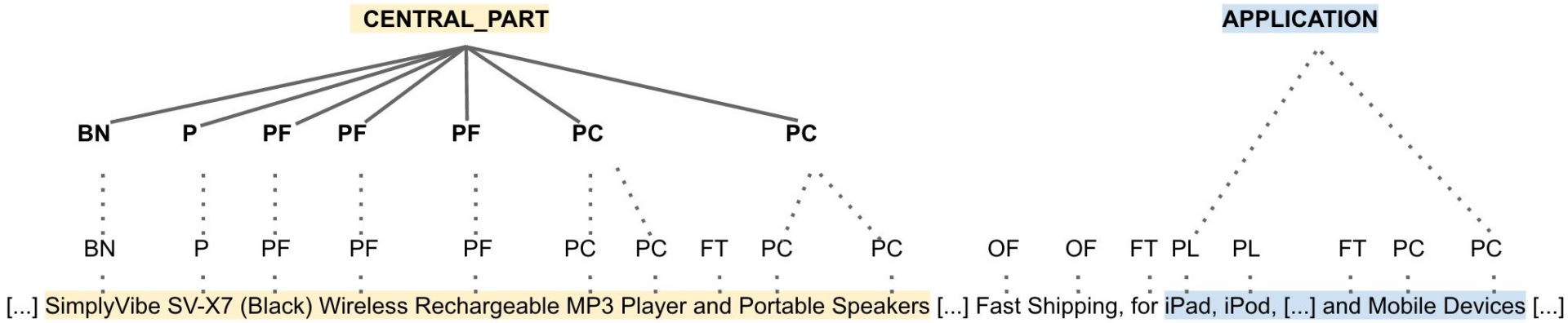
Submit Request

Manual Entry:

Submit Request

Homepage: http://www.gabormelli.com/Projects/PTPv1
Source Code: https://dropbox.com/sh/q8cyv0wfuyg0han/sXZ95rOUC8

Baseline



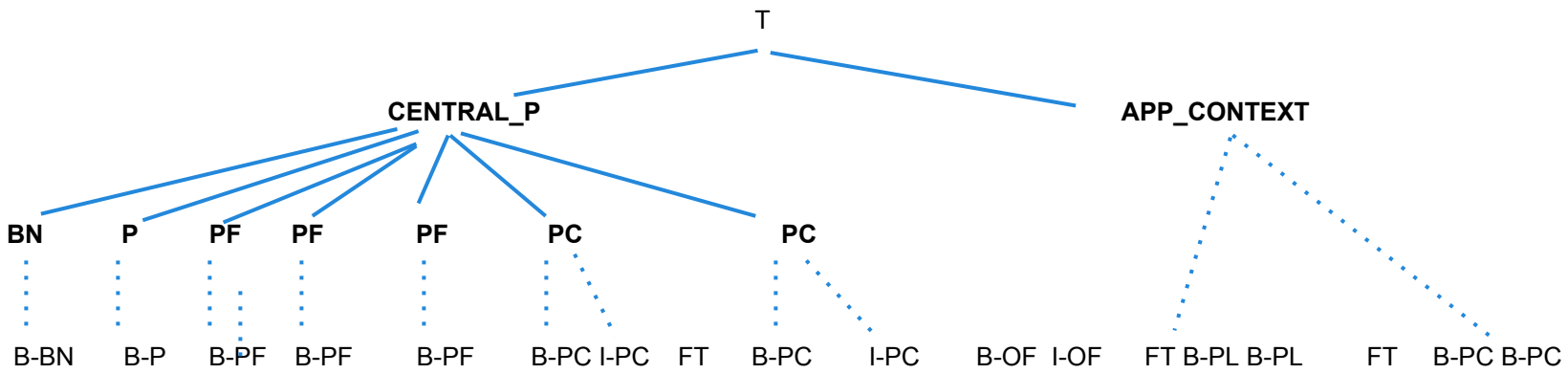
- based on tagging from shallow parser
- repeatedly split phrase into central part and app context
  - based on FT phrases (“for”, “works with”...)
  - exclude OF phrases (“free shipping” ...)
- concatenate and extract BN, P, PC, PF phrases

Towards Structured Offers

(Refurbished) SimplyVibe SV-X7 (Black) Premium Wireless Rechargeable FM, MP3 Player and Portable Speakers w/ extension cable, fast Shipping US Seller with 6 Month Warranty, for iPad, iPod, iTouch, iPhone, Laptops (Mac/ PC) and Mobile Devices - Accepts USB/Micro SD

product/id	SV-X7
product category	fm, mp3 player, portable speaker
brand name	SimplyVibe
attributes	black, rechargeable, wireless
central phrase	SimplyVibe SV-X7 (Black) Premium Wireless Rechargeable FM, MP3 Player and Portable Speakers
app context	iPad, iPod, iTouch, iPhone, Laptops (Mac/ PC) and Mobile Devices

Step 2: Phrase Identification



[...] SimplyVibe SV-X7 (Black) Wireless Rechargeable MP3 Player and Portable Speakers [...] Fast Shipping, for iPad, iPod, [...] and Mobile Devices [...]

Heuristic for Phrase Extraction

- repeatedly split phrase into `central` and `app` part
  - rules based on occurrence of FT terms (e.g. “for”)
  - remove OF sequences (“free shipping”)
- extract all BNs, Ps, PCs, PFs from `central`
- Jeep Cherokee Replacement Fog Light Assembly - 1-Pair, Free Shipping

Step 3: Normalization and Sub-Categorization

- base product categories (PCs)
  - extracted PC can be very specific
  - reduce to base form
    - domain-specific dictionary of valid/invalid base forms
    - heuristics base on head noun

rec	norm
fiberglass boat	→ boat
lcd display	→ display
baking kit	→ baking kit
cake baking kit	→ baking kit

Overall Workflow

- offers stored in ES
- separate parser per feed
  - Ebay, Amazon,...
- offer structure stored in features field
  - field values
  - conf values

```
"_index": "ama_offer_20150417_x3ugt",
"_type": "offer",
"_id": "l15xOKyoP2w",
"_score": 16.271683,
▼ "_source": {
    "offerId": "l15xOKyoP2w",
    "displayName": "Berryman 0517 Diesel Cetane Booster - 15 oz.",
    "name": "berryman 0517 diesel cetane booster - 15 oz",
    "targetUrl": "http://www.amazon.com/dp/B0002JM868",
    "imageUrl": "http://ecx.images-amazon.com/images/I/41DfezUvtWL.jp
    "expirationDate": "201507200000",
    "source": "AMA",
    ▼ "features": {
        ▶ "attributes": [ "15 oz.","diesel"],
        ▶ "brandname": [ "Berryman"],
        ▶ "brandname_conf": [ 0.9940250834270589],
        ▶ "central_phrase": [ "Berryman 0517 Diesel Cetane Booster - 15 o
        ▶ "central_words_P": [ "0517"],
        ▶ "central_words_P_conf": [ 0.9047638183549801],
        ▶ "central_words_P_num": [ 1],
        ▶ "central_words_all_PC": [ "booster","cetane booster"],
        ▶ "central_words_all_PC_conf": [ 0.8101465476905569],
        ▶ "central_words_all_PC_num": [ 1],
        ▶ "central_words_base_PC": [ "booster"],
        ▶ "central_words_base_PC_conf": [ "p"],
        ▶ "is_accessory": [ true],
        ▶ "parsing_conf": [ 0.9144932796206553]
```

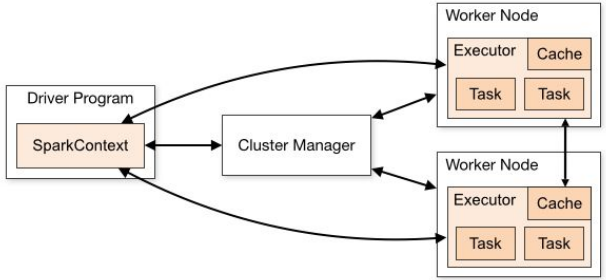
We are here

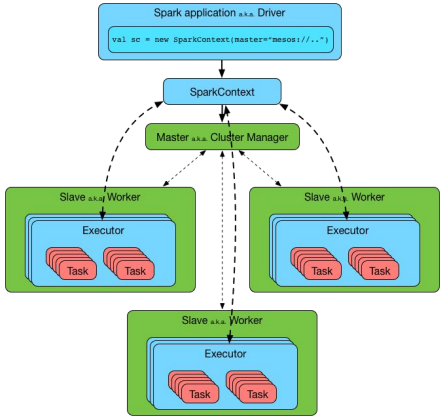
- ~~Overview~~
- ~~Text Preprocessing~~
- ~~Syntactic Parsing~~
- ~~Shallow Semantic Analysis~~
- **>> Big NLP Frameworks <<**
- Text Classification & Clustering
- Information Extraction
- Word Relatedness Functions

NLP on Spark (via AWS EMR)

http://Goo.gl/85BIkI

This document describes how to create and use an Apache Spark cluster for Python-based (and iPython) natural language processing tasks using the NLTK library. The Spark cluster is based on Amazon's (AWS) Elastic Map Reduce (EMR) service. A PDF conversion step is also included.

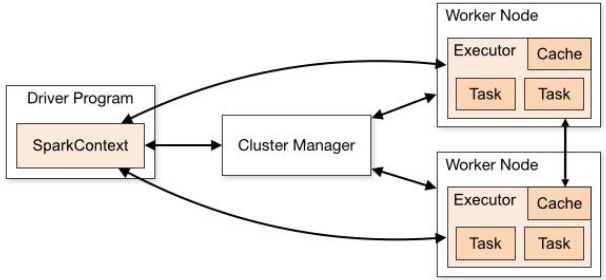


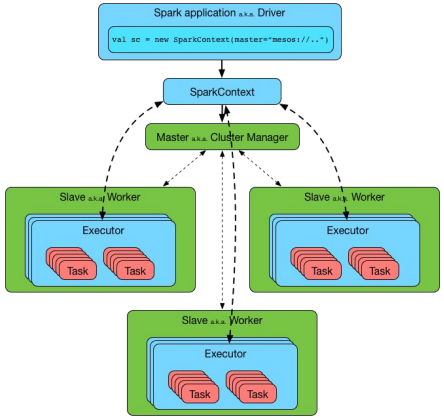


NLP on Spark (via AWS EMR)

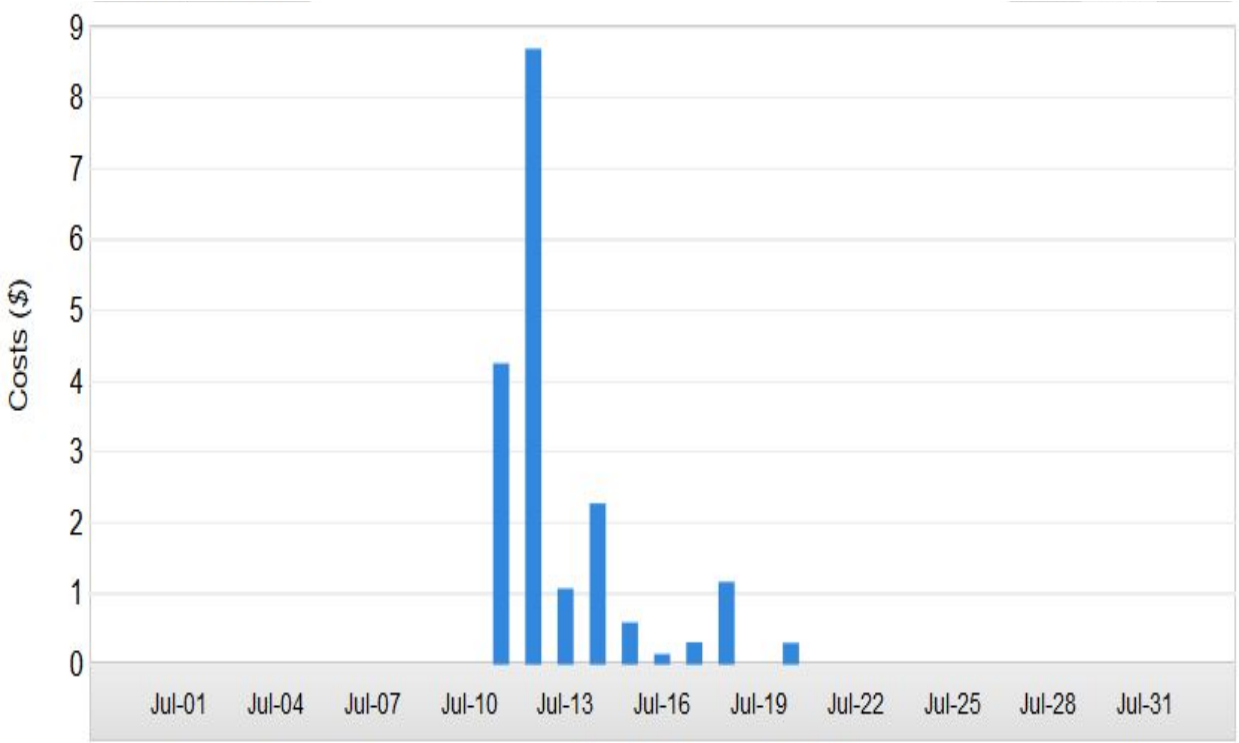
http://Goo.gl/85BIkI

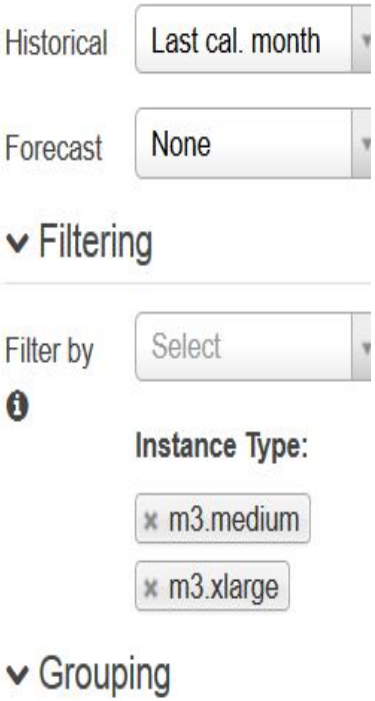
- S3 Storage Setup
- Set-up EMR Cluster
- Connect to the Spark Cluster
- Using iPython
- Examples of Performing Jobs
- APPENDIX A - Monitoring the Cluster
- APPENDIX B - pyspark jobs via CLI command line





Spark's (EMR's) cost structure

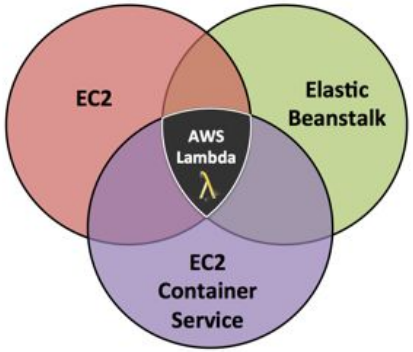




NLP on AWS Lambda

http://Goo.gl/DJ9dbc

This document describes a AWS Lambda function-based approach to perform the PDF-to-text plus NLP process on AWS (instead of, say, on Spark).



NLP on AWS Lambda

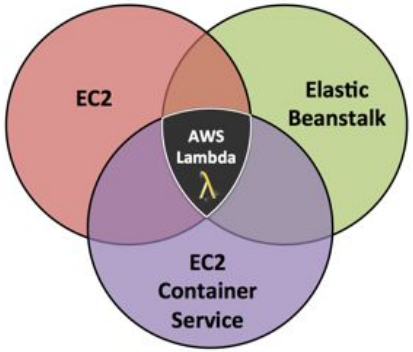
http://Goo.gl/DJ9dbc

[Part 1 - Lambda function code]

[Part 2 - Packaging Lambda function]

[Part 3 - Deploying Lambda function]

[Part 4 - Running and monitoring]



AWS Lambda - for (Unpredictable) High-Velocity Data

Lambda is an event-driven “serverless” compute service for custom code.

Lambda's cost advantage

Lambda		$0.00
US East (Northern Virginia) Region	**Usage**	
AWS Lambda Lambda-GB-Second		
AWS Lambda - Compute Free Tier - 400,000 GB-Seconds - US East (Northern Virginia)	189.250 seconds	$0.00
Total:		**$0.00**
AWS Lambda Request		
AWS Lambda - Requests Free Tier - 1,000,000 Requests - US East (Northern Virginia)	4 Requests	$0.00
Total:		**$0.00**
Region Total:		**$0.00**
Registrar		$24.00

Demo

gmelli-web.s3-website-us-east-1.amazonaws.com/

S3 uploader

AWS Details

ACCESS KEY	enter access key
SECRET KEY	enter secret key
BUCKET	lambda-s3-trigger

Upload PDF File

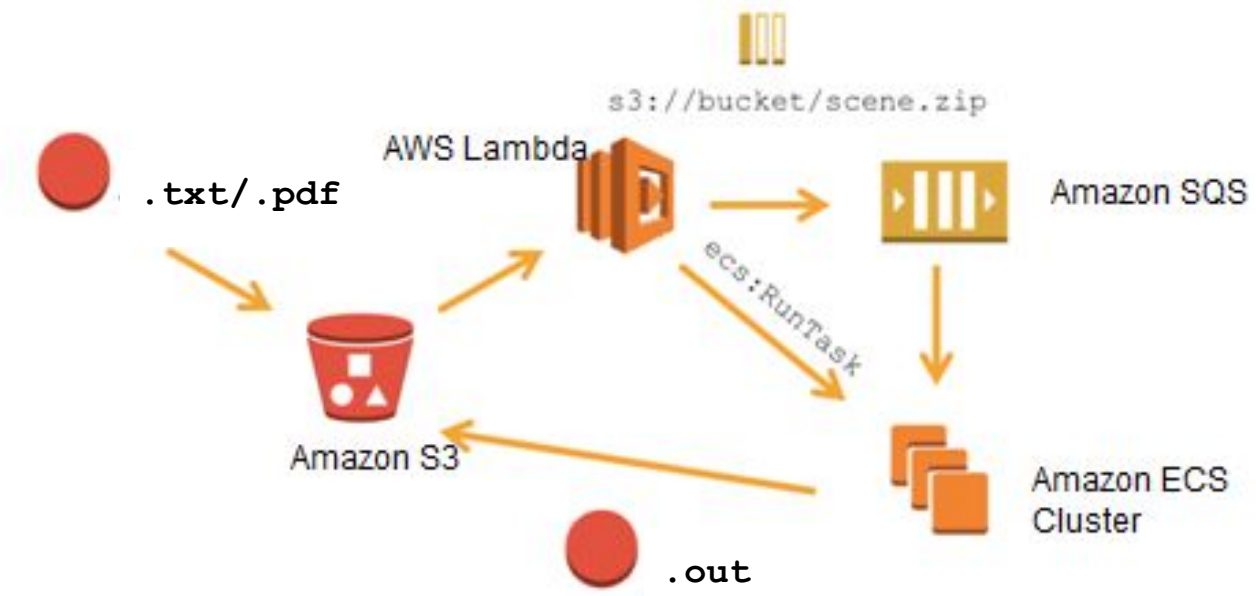
Browse... No file selected.

5MB Upload Limit
10MB Upload Limit
15MB Upload Limit

Upload

Last Modified	Size	Key
2016-08-02T01:22:09.000Z	424.0 kB	aberdeen_SC_13.nlp
2016-08-02T01:21:02.000Z	883.4 kB	aberdeen_SC_13.pdf
2016-08-09T03:33:39.000Z	1.3 MB	lrec2016-workshop_boa_1.nlp
2016-08-09T03:29:18.000Z	5.7 MB	lrec2016-workshop_boa_1.pdf
2016-08-09T03:33:02.000Z	1.3 MB	lrec2016-workshopboa_1.nlp
2016-08-09T03:28:41.000Z	5.7 MB	lrec2016-workshopboa_1.pdf

For more intensive NLP jobs



We are here

- ~~Overview~~
- ~~Text Preprocessing~~
- ~~Syntactic Parsing~~
- ~~Shallow Semantic Analysis~~
- ~~Big NLP Frameworks~~
- **>> Text Classification & Clustering <<**
- Information Extraction
- Word Relatedness Functions

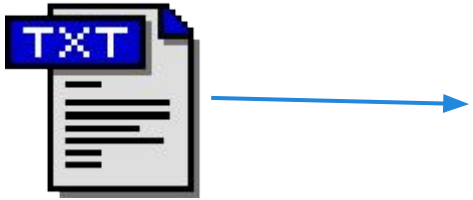
Text Classification & Clustering

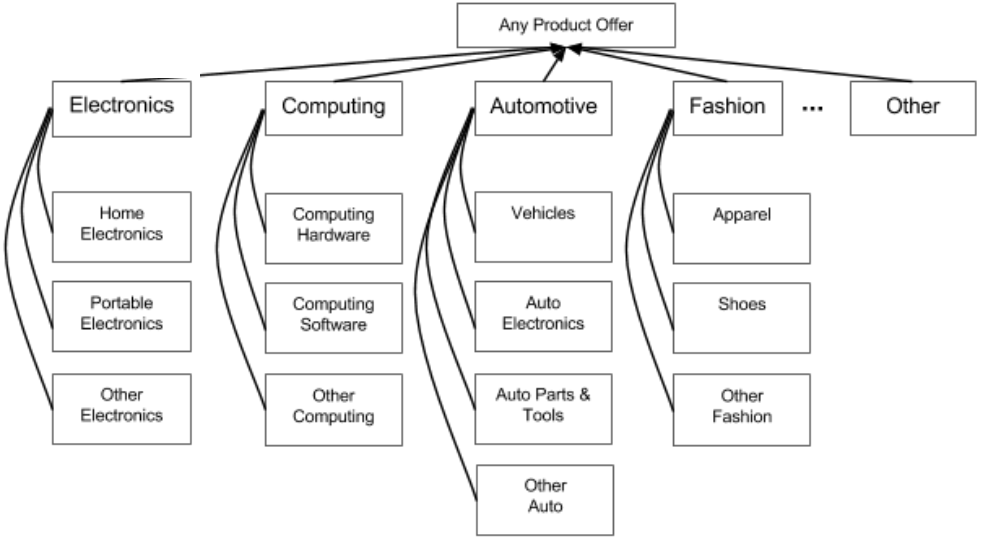
- Text Classification
  - Binary Text Classification
  - Taxonomy Node-Directed Classification
  - Extra-Propositional Semantic Analysis, such as: Sentiment Recognition, Propositional Hedging Recognition, Propositional Factuality Recognition, Propositional Negation Recognition, Irony Recognition, Sarcasm Recognition, Spam Filtering.
- Text Clustering
  - Topic discovery

Binary/n-ary Text Classification

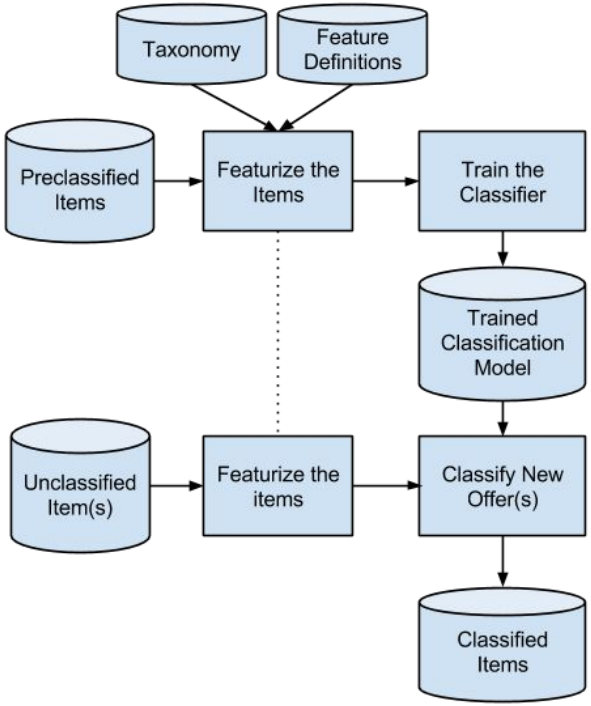
- Spam filtering, discern E-mail spam messages from legitimate emails
- Email routing, direct a receive email to a mailbox depending on topic[12]
- Language identification, automatically determine the language of a text
- Genre classification, automatically determining the genre of a text[13]
- Readability assessment, automatically determine the degree of readability of a text, either to find suitable materials for different age groups or reader types or as part of a larger text simplification system
- Sentiment analysis, determining the attitude of a speaker or a writer with respect to some topic or the overall contextual polarity of a document.

Taxonomy Node-directed Text Item Classification





Workflow: Train and Predict/Classify



ClassifyToTaxoNode.Predict.ipynb

```
import pandas as pd
import numpy as np
from re   import split
print "numpy version: " + np.__version__ "pandas version: " + pd.__version__
# pandas version: 0.14.1    numpy version: 1.8.1

from sklearn import preprocessing, svm, cross_validation  # labelEncoder
from sklearn.metrics import accuracy_score
from sklearn.feature_extraction.text import CountVectorizer
import sklearn.feature_extraction.text
from sklearn.externals import joblib
print "sklearn version: " + sklearn.__version__
# sklearn version: 0.15.0
```

Text Clustering

Hierarchical Clustering

- Top-Down Partitional Clustering
  - a k-Means clustering algorithm.
  - a k-Medoids clustering algorithm.
- Bottom-Up Agglomerative Clustering
  - a CURE clustering algorithm.
  - a Single-Linkage clustering algorithm.
  - a Complete-Linkage clustering algorithm.
  - a UPGMA clustering algorithm.

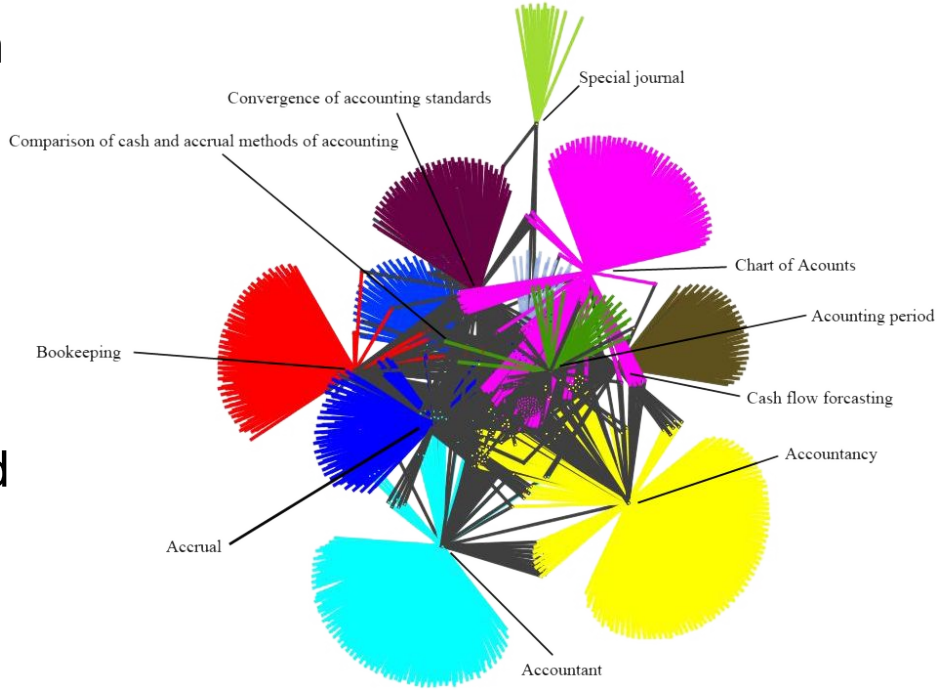
As simple as

```
from nltk.cluster import
    KMeansClusterer,
    GAAClusterer,
    euclidean_distance

cluster = KMeansClusterer(7, euclidean_distance)
cluster = GAAClusterer(7)
```

Scaling Topic Clustering

Given text from new sources, what can we say about the data we're viewing?

Let's use some unsupervised techniques to find out.



How can we grow this into a distributed problem?

Split the preparation work onto separate machines.

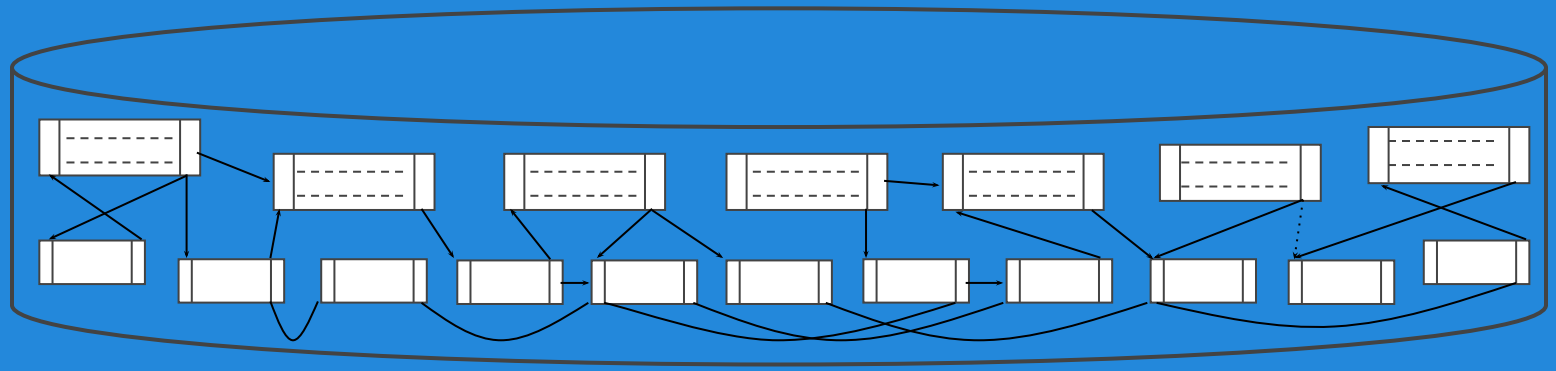
We are here

- ~~Overview~~
- ~~Text Preprocessing~~
- ~~Syntactic Parsing~~
- ~~Big NLP Frameworks Demo~~
- ~~Shallow Semantic Analysis~~
- ~~Text Classification & Clustering~~
- **>> Information Extraction <<**
- Word Relatedness Functions

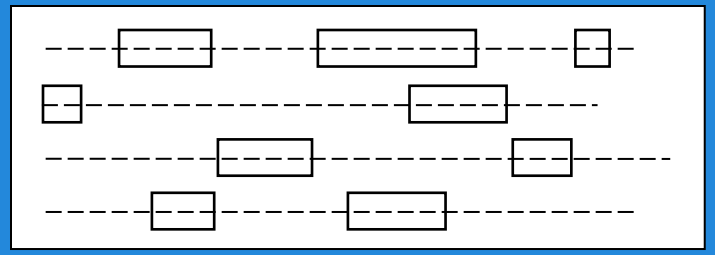
Inputs

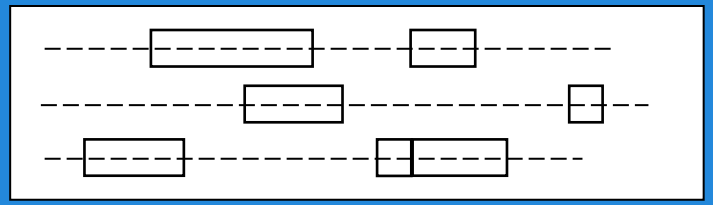
text corpus

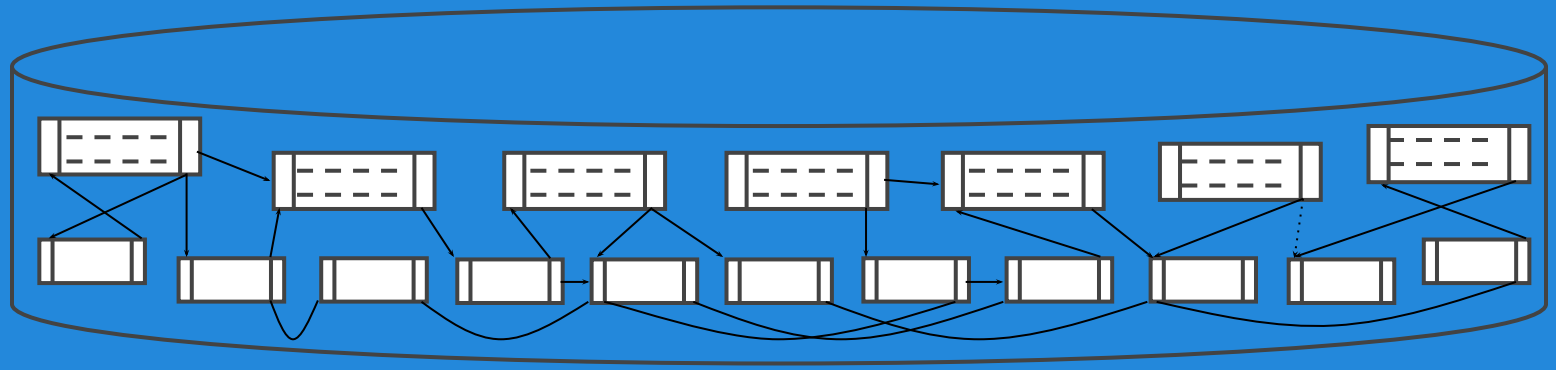
ontology



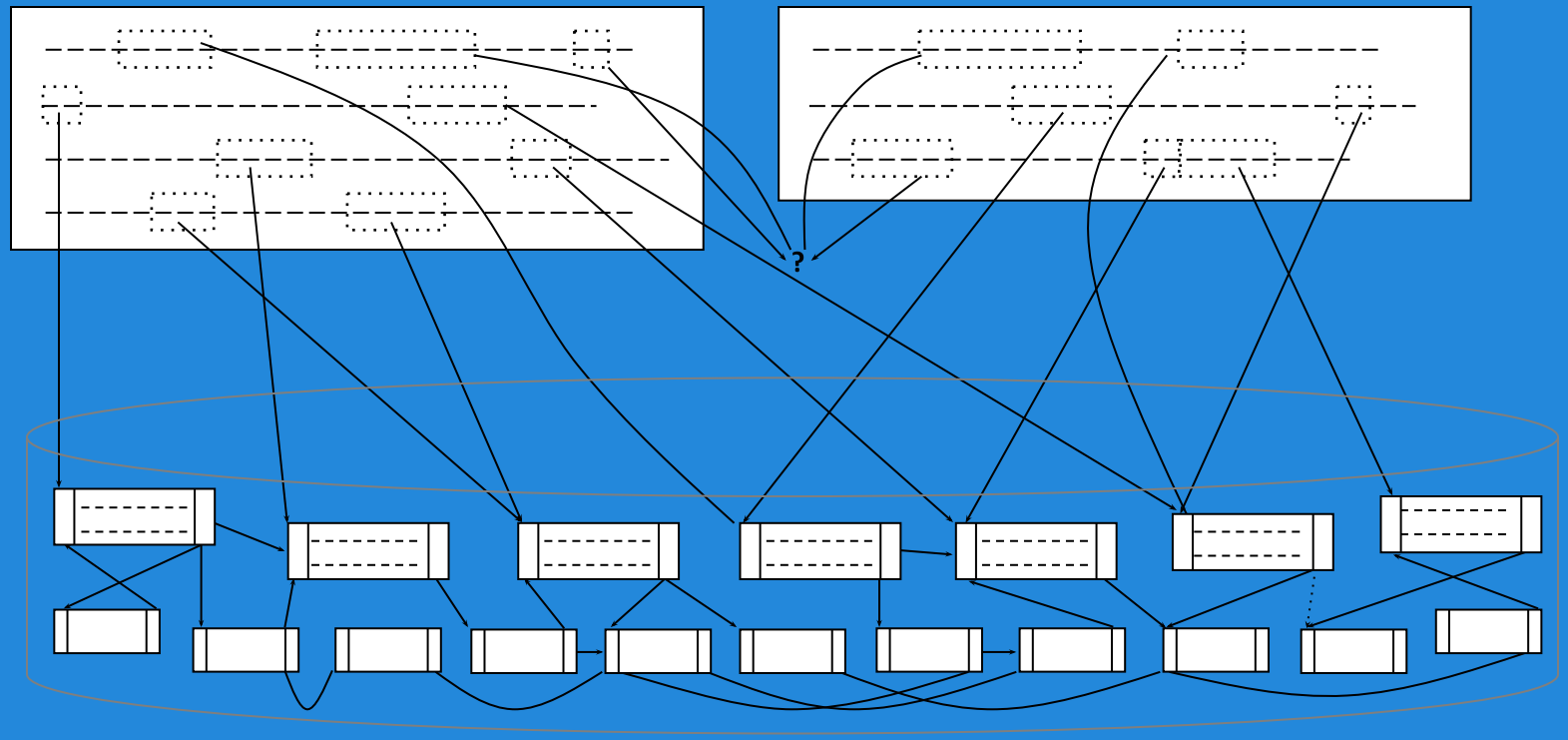
Entity Mention Recognition



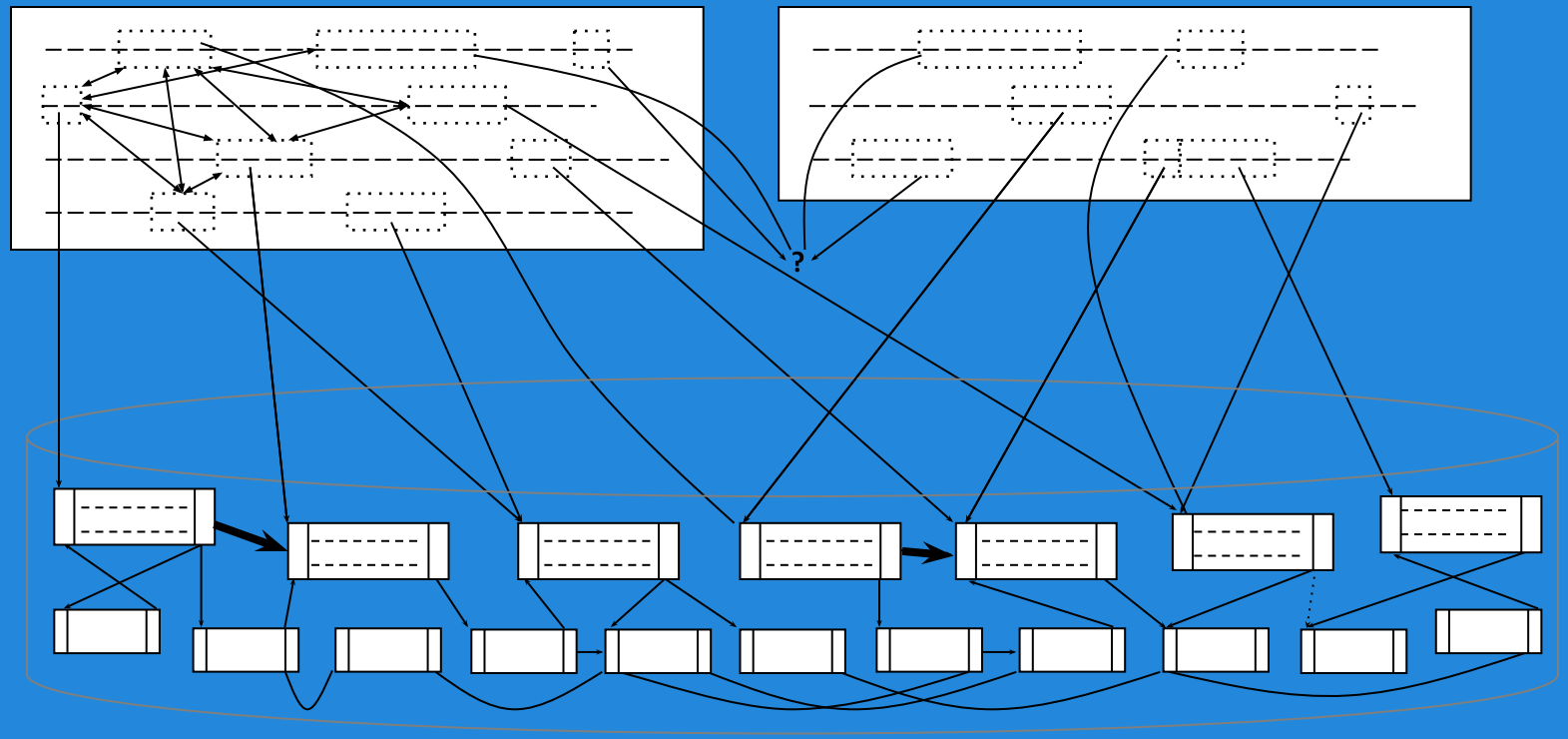




Entity Mention Linking



Relation Mention Recognition



Information Extraction

- Restricted Information Extraction
  - from PDF: Resumes
  - from Web: Recipes
  - from Abstracts: Protein Locale. Rel. Extr. (PPLRE).
- Open Information Extraction
  - from Web+Wikipedia: Google Knowledge Vault
  - from Web: NELL

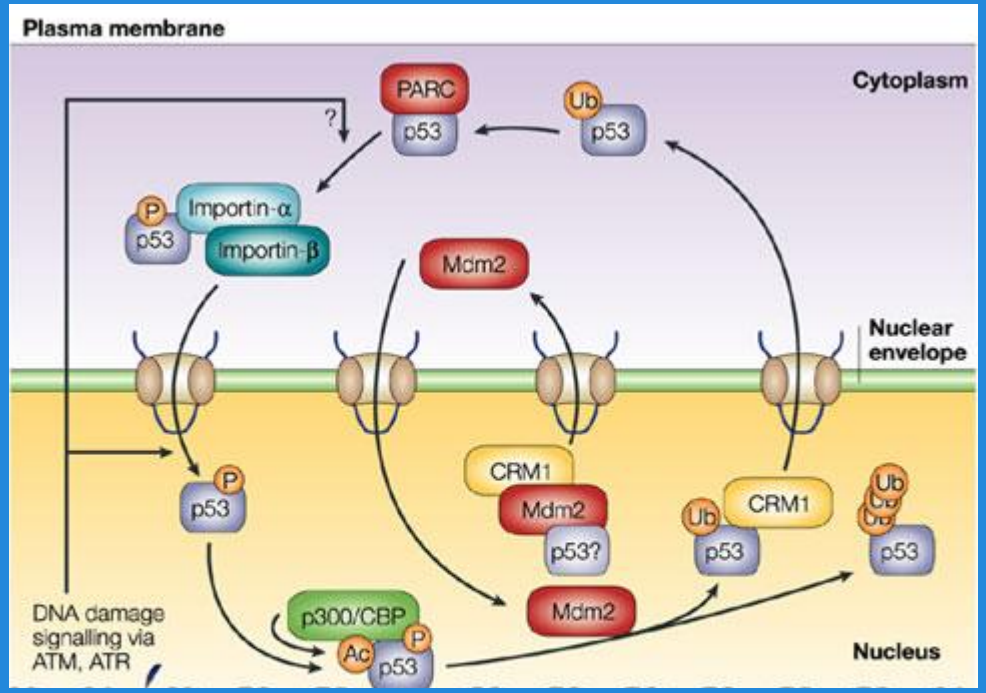
Protein Prokaryote Localization Relation Extraction (PPLRE)

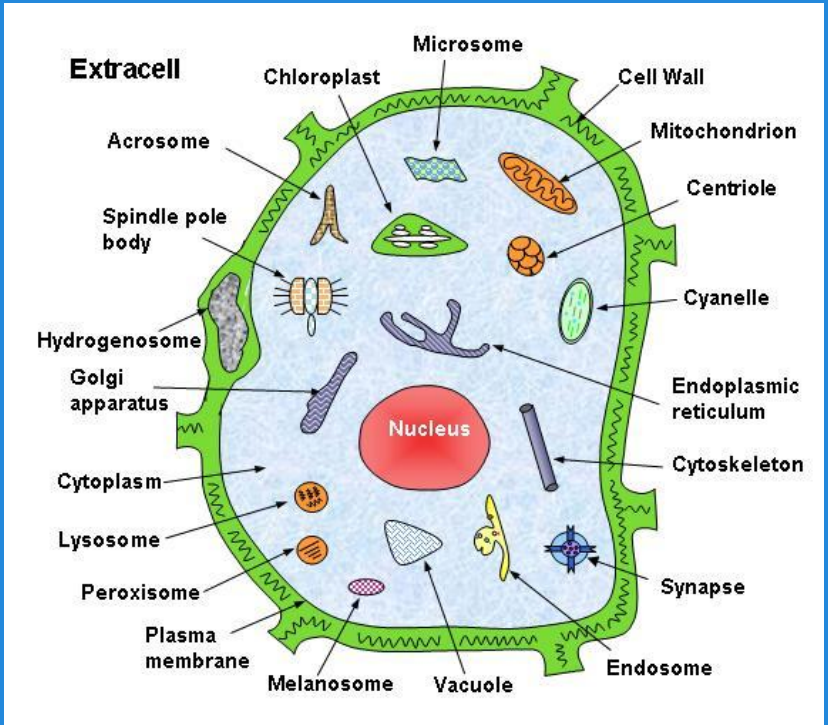
Can you spot the **subcellular localization event** mention?

Identification of a TcpC-TcpQ Outer Membrane Complex Involved in the Biogenesis of the Toxin-Coregulated Pilus of *Vibrio cholerae*

By Niranjan Bose and Ronald K. Taylor

The toxin-coregulated pilus (TCP) of *Vibrio cholerae* and the soluble TcpF protein that is secreted via the TCP biogenesis apparatus are essential for intestinal colonization. The TCP biogenesis apparatus is composed of at least nine proteins but is largely uncharacterized. TcpC is an outer membrane lipoprotein required for TCP biogenesis that is a member of the secretin protein superfamily. In the present study, analysis of TcpC in a series of strains deficient in each of the TCP biogenesis proteins revealed that TcpC was absent specifically in a *tcpQ* mutant. TcpQ is a predicted periplasmic protein required for TCP biogenesis.





Organism	Protein	Subcellular Location
???	???	???

After Segmentation and NER ...

"The pilus(location) of V. cholerae(organism) is essential for intestinal colonization.
The pilus(location) biogenesis apparatus is composed of nine proteins.
TcpC(protein) is an outer membrane(location) lipoprotein required for pilus(location) biogenesis."

Organism	Protein	Subcellular Location
V. cholerae	*TcpC*	*outer membrane*

| Rel. Case | | | | Feature Space | | | |

PPLRE Annotation Tool

Form1

RELATION_CASEID
240

DOC_ID (PubMed ID)
3290200

ORGANISM_NAME
S. marcescens

PROTEIN_NAME
hemolysin

PASSAGE/SENTENCE

The <PROT>shlB gene product</PROT> was localized to the outer membrane of <ORG>Escherichia coli</ORG> cells harboring the <ORG>S. marcescens</ORG> <PROT>hemolysin</PROT> determinant .

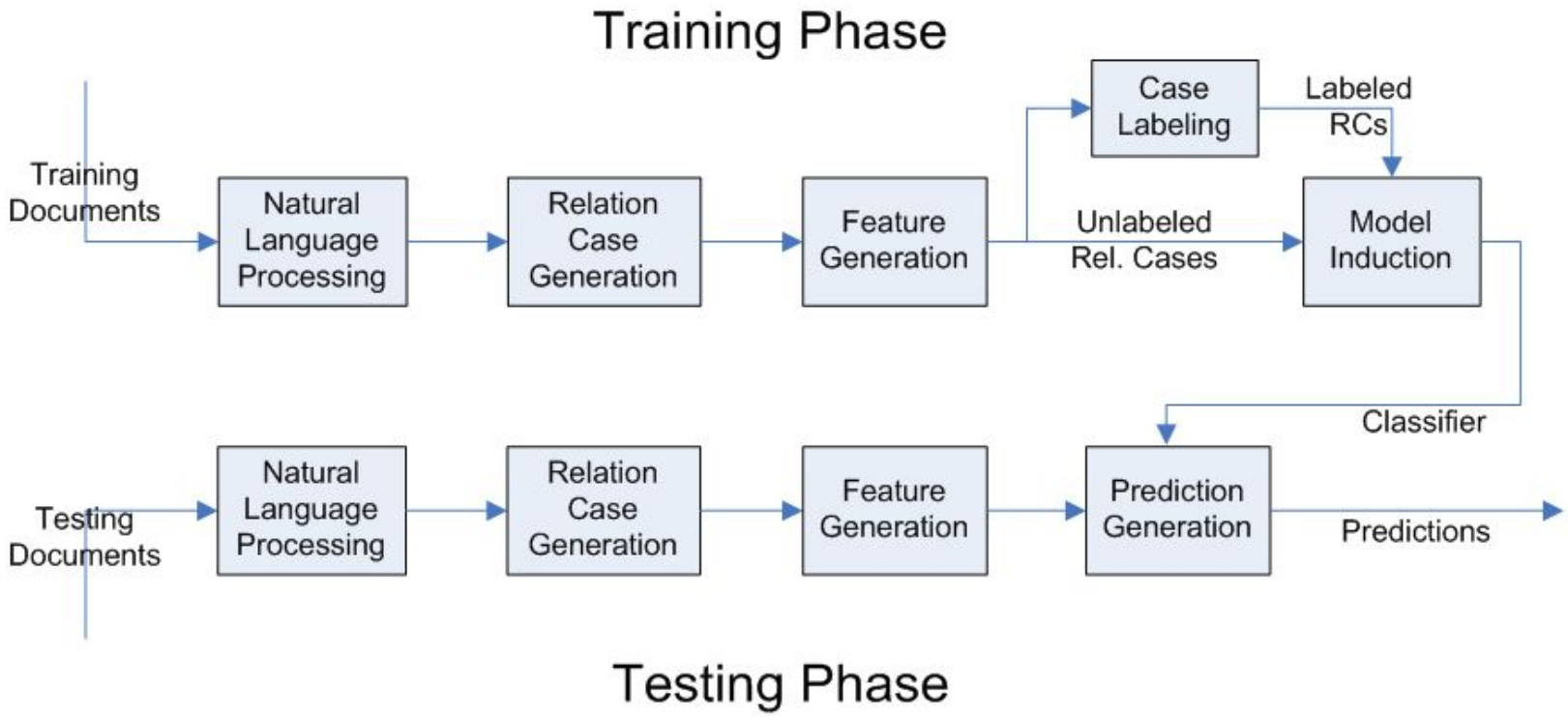
Is the relation case above a valid relation?

☐

COMMENTS

Record: 12 of 185

PPLRE project process



Record Canonicalization

An important non-NLP post-processing step to Information Extraction is record merging.

author	Brian Milch et al.
title	BLOG: Probabilistic Models with Unknown Objects
venue	IJCAI
editor	
pages	

(a)

author	B. Milch et al.
title	BLOG: Probabilistic Models with Unknown Objects
venue	Intl. Conf. on AI
editor	
pages	1352-1359

(b)

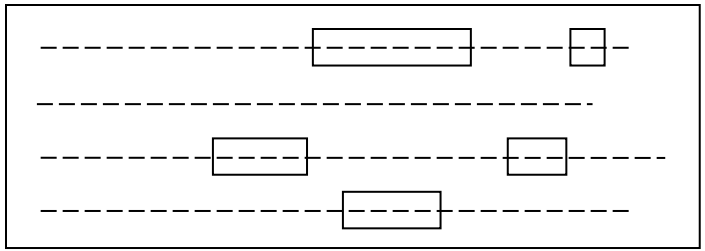
author	Brian Milch et al.
title	BLOG: Probabilistic Models with Unknown Objects
venue	
editor	L. Getoor and B. Taskar
pages	

(c)

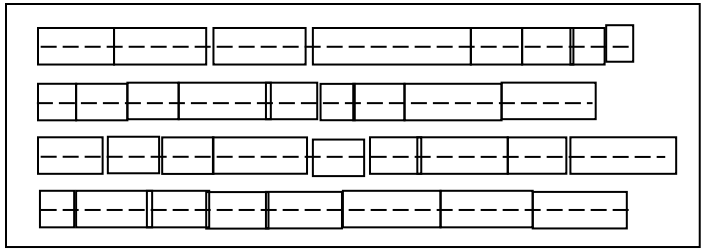
Word Sense Disambiguation (WSD)

- Solved by a WSD System using a WSD Algorithm.
- Heuristic WSD vs. Data-Driven WSD.
- All-Words WSD vs. Target-Word WSD.
- Course-Grained WSD vs. Fine-Grained WSD.
- Completed Sense Dictionary vs. Incomplete Sense Dictionary.

Target-Words vs. All-Words WSD



- PEOPLE, ORG., LOCATION,

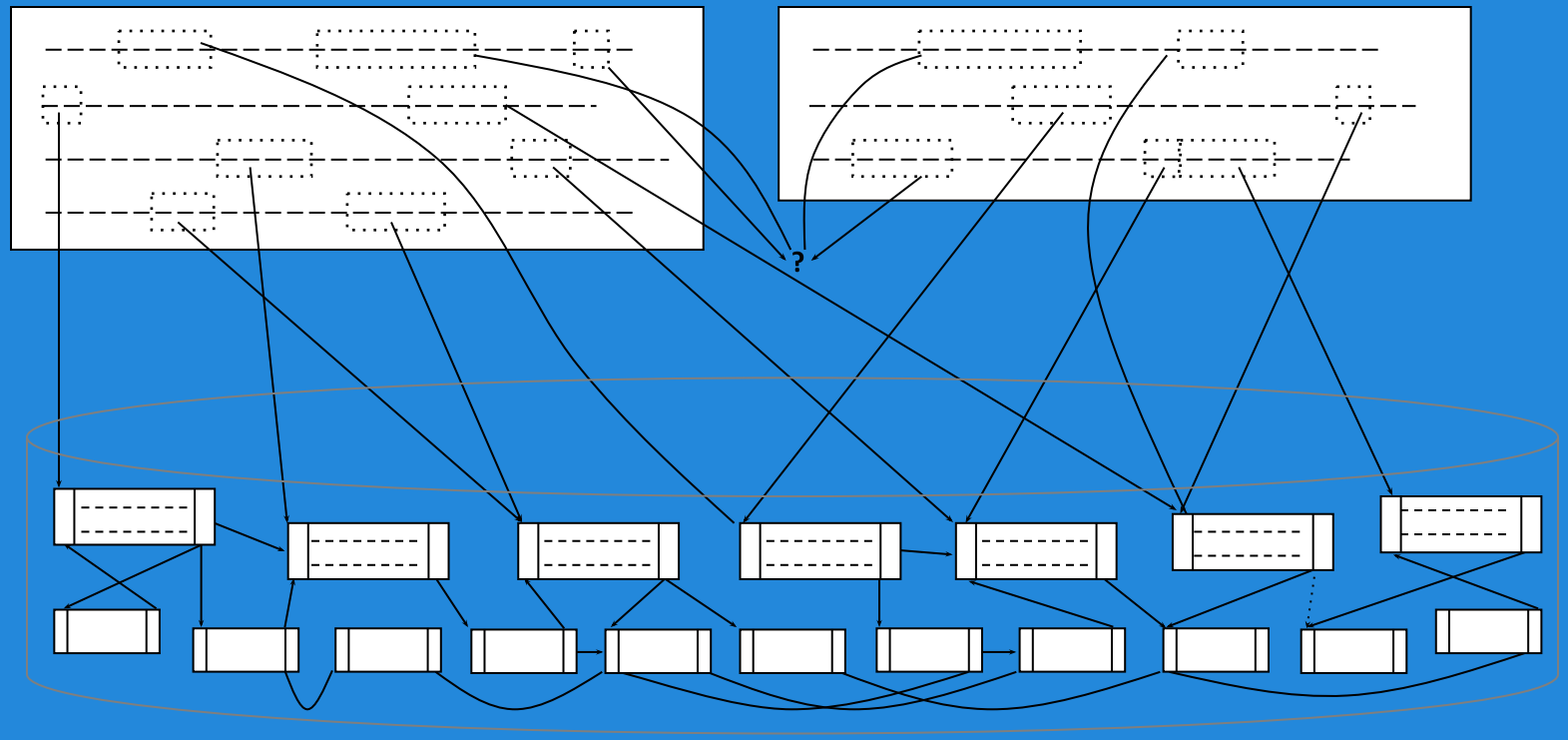


- Dictionary words

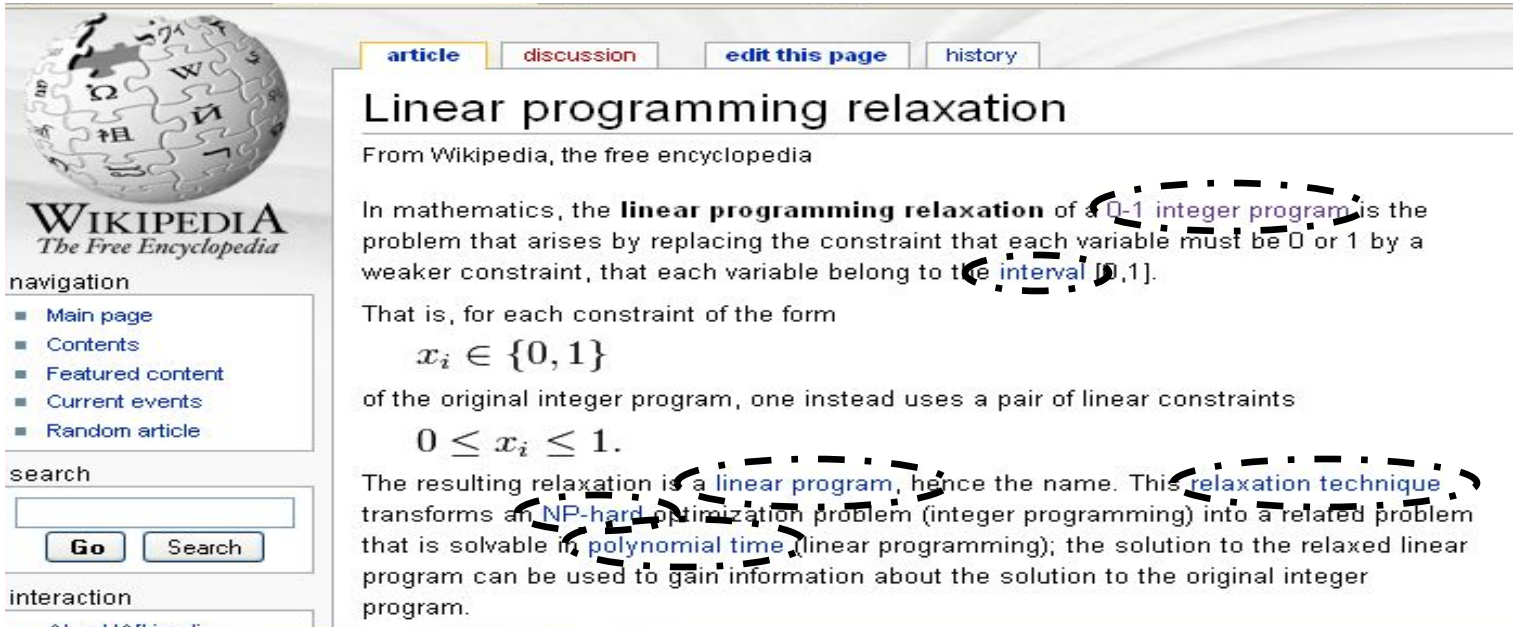
babelfy.org/

SemEval-2015 Task 13
Multilingual All-Words Sense Disambiguation and Entity Linking

Entity Mention Linking



Manually Hyperlinked Pages



WIKIPEDIA
The Free Encyclopedia

navigation
- Main page
- Contents
- Featured content
- Current events
- Random article

search
Go Search

interaction

article | discussion | edit this page | history

Linear programming relaxation

From Wikipedia, the free encyclopedia

In mathematics, the **linear programming relaxation** of a 0-1 integer program is the problem that arises by replacing the constraint that each variable must be 0 or 1 by a weaker constraint, that each variable belong to the interval [0,1].

That is, for each constraint of the form

$$x_i \in \{0, 1\}$$

of the original integer program, one instead uses a pair of linear constraints

$$0 \leq x_i \leq 1.$$

The resulting relaxation is a linear program, hence the name. This relaxation technique transforms an NP-hard optimization problem (integer programming) into a related problem that is solvable in polynomial time (linear programming); the solution to the relaxed linear program can be used to gain information about the solution to the original integer program.

The kdd09cma1 Annotated Corpus

- Composed of the 139 abstracts for KDD-2009
  http://portal.acm.org/toc.cfm?id=1557019

Documents	139	Per Document (min/med/max)
Sentences	1,185	3/8/17
Tokens	30,450	105/220/367
Concept Mentions	(100%) 5,449	25/57/95
Single Token	(~68%) 3,712	7/25/57
Multi Token	(~31%) 1,737	2/12/34

The kdd09cma1 Annotated Corpus

DOCUMENTS		139	(min/median/max)
LINKED MENTIONS	51.7%	3,920	10 | 26 | 66
UNLINKED MENTIONS	48.3%	3,660	3 | 25 | 49
DISTINCT CONCEPTS LINKED TO BY CORPUS		820	9 | 19 | 50
CONCEPTS UNIQUELY LINKED TO BY A SINGLE DOCUMENT			0 | 2 | 17

Evaluations

- Concept Mention Identification
  - F1 (Precision, Recall)
- Concept Mention Linking
  - Accuracy
    - on Manually Curated (Gold) Mentions
    - on Predicted Mentions
- Relation Mention Recognition
  - F1 (Precision, Recall)

Evaluations (cont.)

- Partial Credit
  - “Near” Concept Mention Identification
  - “Near” Concept Mention Linking
- Extrinsic Measure: Annotation Time Savings
  - Time to perform manually (e.g. 34 min)
  - Time to fix automated annotation (e.g. 19 min)
  - Ratio (e.g. 34/19 ≈1.8)

Concept Mention Linking Features

Mention	Concept	Predictor Features														Label
		Anchor Txt.		Text Wind.	Document		Concept		Corpus		Cand. Set		Collective			
m_m	o_c	a_m, o_c		t_m, o_c	d_m, o_c		o_c		$D\text{-}d_m$		C_{mc}		o_c, S_m			T/F
1	1903	0	...	0.01	0.03	...	3	...	0	...	15	...	4	...		F
1	1021	0	...	0	0.01	...	1	...	2	...	30	...	1	...		F
1	829	1	...	0.02	0.02	...	12	...	1	...	15	...	7	...		T
2	4028	0	...	0.08	0.08	...	5	...	11	...	30	...	9	...		T
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...		...

Collective Features

- Use knowledge about some of the links' labels
- For example:
  - Assume unambiguous mention of "*Indonesia*"
  - Predict link for the ambiguous mention of "*Java*"
  - The ontology can now be used to provide some background knowledge based on relatedness.

Collective Feature Handling via Iterative Classification

- Based on (Neville & Jensen, 2000)
  1. Train model ($M_{\cancel{col}}$) without the collective features
  2. Train a model (M_{col}) with the collective features
  3. For each iteration of i from 1 to m
     - Calculate the value for the collective features
     - Apply model $M_{\cancel{col}}$ to the test set, if i is 1
  4. otherwise, apply model M_{col} to the test set.
     - Select the k most probable links, where $k = N\,(i\,/m)$.

We are here

- ~~Overview~~
- ~~Text Preprocessing~~
- ~~Syntactic Parsing~~
- ~~Big NLP Frameworks Demo~~
- ~~Shallow Semantic Analysis~~
- ~~Text Classification & Clustering~~
- ~~Information Extraction~~
- **>> Word Relatedness Functions <<**

Semantically Related Words

`moreLikeThis`(*dress shoes, oxford shoes, ...*)

`notLikeTheOther`(*shoes, computer, dress*)

`analogical` ([*dress shoes, sneaker*],
[*high heels, ?*])

Distributional (*co-occurence*) Lexical Semantics Heuristic

Words of a feather flock together

I was reading a **magazine** today

The **magazine** published an article

She buys this **magazine** every day

I was reading a **newspaper** today

The **newspaper** published an article

She buys this **newspaper** every day

based)
Lexical Semantics

- ...
- words that occur in the same contexts tend to have similar meanings (Harris, 1954)
- “*a word is characterized by the company it keeps*" (1957, Firth).
- “*words which are similar in meaning occur in similar contexts*” (Rubenstein & Goodenough, 1965).
- ...

context window

~~She~~ buys this **newspaper** every day

vocab id		1	2	3	4	5	6	7	8	9	...	671,202
term window		*an*	*article*	*buys*	*day*	*every*	*published*	*she*	*the*	*this*	...	*zzz*
1	*newspaper*	f	f	**T**	**T**	**T**	f	f	f	**T**	f	f
2	*newspaper*	**T**	f	f	f	f	**T**	f	**T**	f	f	f
...	*newspaper*	**T**	f	f	f	f	**T**	f	**T**	f	f	f
8,027,192	*zzz*	**f**	f	f	f	f	**f**	f	**f**	f	...	n/a

Words represented as dense continuous vectors

"*chrome-plated*" =

```
 0.286
 0.792
-0.177
   ...
-0.542
 0.349
 0.271
```

projections to an n-dimensional latent space, where n is often > 200

More Like This: “*france*” , “*china*”

spain	0.678515
belgium	0.665923
netherlands	0.652428
italy	0.633130
switzerland	0.622323
luxembourg	0.610033
portugal	0.577154
russia	0.571507
germany	0.563291
catalonia	0.534176

https://code.google.com/p/**word2vec**/

```
/word2vec/trunk$ wc -l *.c
  143 compute-accuracy.c
  142 distance.c
  292 word2phrase.c
  702 word2vec.c
  144 word-analogy.c
```

Semantic Word Models (of Offers)

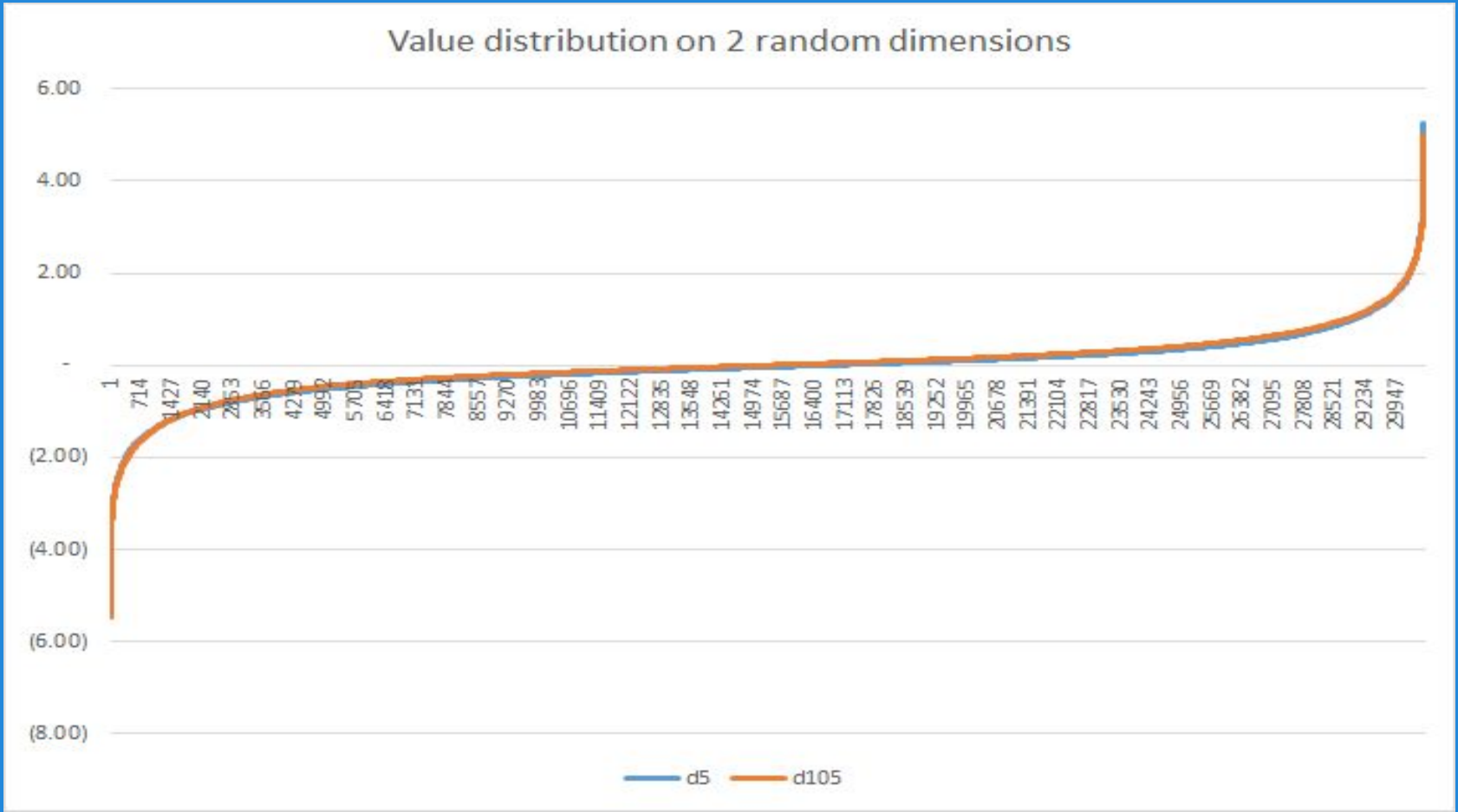
`model.bin` := `word2vec`(OfferCorpus)

`model.bin.distance`(*x*,...)

`model.bin.analogy`(*x*,*y*,*w*)

```
$ ./word2vec -train fashion.2014-06.txt -output clothing.2014-06.bin \
       -window 8 -size 200 \
       -min-count 10 \
       -cbow 1 -negative 25 -hs 0 -sample 1e-4 -threads 40 -iter 15 -binary 1

Starting training using file fashion.2014-06.txt
Vocab size: 1,239,058
Words in train file: 362,247,010
Alpha: 0.048747  Progress: 2.51%  Words/thread/sec: 70.65k
Alpha: 0.048364  Progress: 3.27%  Words/thread/sec: 70.67k
Alpha: 0.047480  Progress: 5.04%  Words/thread/sec: 70.68k
Alpha: 0.028223  Progress: 43.55%  Words/thread/sec: 70.79k
Alpha: 0.024451  Progress: 51.10%  Words/thread/sec: 70.78k
Alpha: 0.008861  Progress: 82.28%  Words/thread/sec: 70.79k
Alpha: 0.004163  Progress: 91.67%  Words/thread/sec: 70.77k
Alpha: 0.000173  Progress: 99.65%  Words/thread/sec: 70.74k
Alpha: 0.000021  Progress: 99.96%  Words/thread/sec: 70.77k
Alpha: 0.000005  Progress: 100.00%  Words/thread/sec: 70.78k


$ ./distance clothing.2014-06.bin
Enter word or sentence (EXIT to break): _____
```

row	term	dim1	dim2	dim3	dim4	...	dim248	dim249	dim250
1	watch	-1.121	0.232	0.704	-0.181	...	1.611	-1.037	-0.978
2	ruby	0.396	-0.721	-1.035	-0.178	...	-0.986	-0.187	0.157
3	black	-0.287	-0.092	-1.034	-0.315	...	-0.366	0.014	-0.468
4	green	-0.076	-0.114	-0.144	-1.081	...	-0.484	0.650	-0.567
5	silver	0.558	-0.702	0.314	-1.270	...	-0.309	0.521	0.083
6	-	0.366	-0.312	-0.139	-1.004	...	-0.319	0.042	0.176
7	ring	0.555	0.424	-1.203	0.340	...	-0.658	0.820	1.033
8	sterling	-0.594	-0.961	1.106	-2.857	...	-1.341	1.476	0.687
9	gold	0.976	0.207	-0.678	0.180	...	-1.271	1.100	0.303
10	with	-0.877	-0.158	-0.624	-1.077	...	-0.458	0.408	0.004
...	...	...	...	...	...	...	...	...	...
30,624	necklace	0.704	1.202	-2.102	-2.386	...	-0.339	2.992	0.463
30,625	steel	-1.014	-0.083	0.905	-0.332	...	1.034	-1.002	-0.418
30,626	men's	0.028	0.024	0.227	-0.470	...	1.404	-2.450	-1.286
30,627	LENGTH>LINEAR>VERY_SMALL>METRIC	1.834	0.727	1.445	-0.606	...	0.627	0.625	-0.128
30,628	leather	-1.306	-0.445	-0.162	-0.477	...	1.941	-0.859	-1.094
30,629	bracelet	-0.396	0.640	-2.301	-1.292		0.769	0.874	0.396


Value distribution on 2 random dimensions
6.00
4.00
2.00
-
(2.00)
(4.00)
(6.00)
(8.00)
1
714
1427
2140
2853
3566
4279
4992
5705
6418
7131
7844
8557
9270
9983
10696
11409
12122
12835
13548
14261
14974
15687
16400
17113
17826
18539
19252
19965
20678
21391
22104
22817
23530
24243
24956
25669
26382
27095
27808
28521
29234
29947
d5
d105

distance(*Reebok*)

Word: **Reebok** Position in vocabulary: 571

Word	Cosine distance
Adidas	**0.551837**
Nike	**0.491554**
Fila	**0.481622**
Skora	**0.471142**
Puma	**0.460984**
K-swiss	**0.445765**
Balance	**0.440028**
Techy	**0.425781**
...	

distance(*Gucci*)

Word: **Gucci** Position in vocabulary: 1244

Word	Cosine distance
Dsquared2	**0.504137**
Ysl	**0.497998**
Burberry	**0.477078**
Prada	**0.448514**
Balenciaga	**0.447502**
Gg	**0.444650**
Horsebit	**0.441435**
...	

distance (**blue**, **red**, **yellow**, **green**)

term	AMA			AMA	EUS			EUS	sum
	FS	HG	JW	sum	FS	HG	JW	sum	
orange	1	1	1	3	1	1	1	3	6
magenta	1	1	1	3	1	1	1	3	6
pink	1	1	1	3	1	1	1	3	6
purple	1	1	1	3	1	1	1	3	6
teal	1		1	2	1	1	1	3	5
black	1	1	1	3	1		1	2	5
brown	1	1	1	3		1	1	2	5
gray	1	1	1	3	1	1		2	5
white	1	1	1	3	1		1	2	5
lime	1	1		2	1	1	1	3	5
violet	1	1		2	1		1	2	4
multicolor	1		1	2	1	1		2	4
peach			1	1	1	1	1	3	4
grey	1	1	1	3	1			1	4
turquoise	1	1		2	1	1		2	4
lavender	1		1	2	1		1	2	4
burgundy	1	1		2	1	1		2	4
aqua	1			1	1	1	1	3	4
maroon	1	1		2	1	1		2	4
multi-color			1	1	1	1	1	3	4
navy	1			1	1	1		2	3
colorful		1	1	2		1		1	3
periwinkle	1		1	2	1			1	3
...	...	...	...	...	...	...	...	...	...

Lexical Vector Models (cont.)

distance: red

Word	Cosine distance
white	0.545141
black	0.516964
purple	0.439816
pink	0.390980
yellow	0.380564
orange	0.330515
carmine	0.328415
green	0.318634
spiderweb	0.316952
gold	0.309333

distance: small

Word	Cosine distance
xx-large	0.548240
x-small	0.540055
x-large	0.538336
large	0.530152
medium	0.513127
xxx-large	0.444760
xxxx-large	0.392090
xl	0.384409

distance: wool

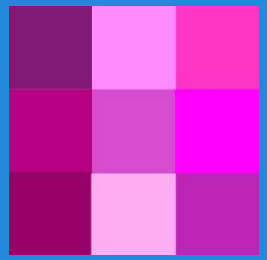
Word	Cosine distance
aran	0.532747
alpaca	0.510891
merino	0.490692
cashmere	0.472697
dahlia	0.447400
lambswool	0.442171
herringbone	0.441842

word-analogy: small s large XX

```
Enter three words (EXIT to break): small s large

Word: small  Position in vocabulary: 92

Word: s  Position in vocabulary: 280

Word: large  Position in vocabulary: 67
```

Word	Distance
l	0.422970
m	0.418687

distance(pink)



Word: **pink** Position in vocabulary: 17

term	AMA				EUS				sum
	FS	HG	JW	AMA sum	FS	HG	JW	EUS sum	
fuchsia	1		1	2	1	1	1	3	5
purple	1		1	2	1	1	1	3	5
orange	1	1	1	3			1	1	4
red	1		1	2			1	1	3
turquoise		1		1	1	1		2	3
lavender			1	1	1		1	2	3
teal						1	1	2	2
blue			1	1			1	1	2
white	1		1	2					2
fuschia					1	1		2	2
green			1	1			1	1	2
magenta						1	1	2	2
...	...	...	...	...	...	...	...	...	...

Word	Cosine distance
purple	0.729254
lavender	0.666517
fuchsia	0.660774
aqua	0.650429
turquoise	0.639061
lime_green	0.630319
coral	0.616357
polka_dot	0.610636
yellow	0.607237
peach	0.602049
fuschia	0.597598
zebra	0.597136

Clustering - kmeans

```
word2vec$ ./word2vec -train AMA...JW.2015-02-02.txt -output AMA...JW.2015-02-02.  classes.txt \
       -classes 500
       -window 8 -size 250 \
       -min-count 10 \
       -cbow 1 -negative 25 -hs 0 -sample 1e-4 -threads 40 -iter 15
```

```
word2vec$ head -14  AMA-2015-02-02.JW.500.250.8.1.150219.classes.txt
watch      21
silver     170
-          481
ring       363
sterling   170
gold       12
black      21
with       276
size       363
band       363
necklace   149
steel      17
```

ales	bier	blatz	bock	buzzed	drinker	friederike	heineken	ipa	
32b	34a	34c	36a	36b	36c	36d	38a	38c	
canucks	leafs	oilers	predators	senators	thrashers	whalers			
1950s	1960s	40's	50s	60's	60s	sixties			
1950's	1950s	1960s	40's	50s	60's	60s	sixties		
1970s	funk-fu	groovy	heartthrob	seventies					
coolest	greatest	sweetest							
happiness	merely	overrated							
azeri	bulgarian	indonesian	latvian	slovenian	tongan	tunisian	vietnamese		
chilean	ecuadorian	ghanaian	honduran	israeli	nigerian	slovene	spaniard	uruguayan	...
eggplant	emerald	mauve	plum	teal					
aquaman	batarang	batwoman	bitman	catwoman	dastardly	dco	gotham	gotham's	...
alzheimer's	alzheimers	autism	cancer	cerebral	cystic	disease	epilepsy	fibromyalgia	...
ariel	aristocats	aurora	bambi	cadabby	cinderella	disney	disney's	dora	...

Beer?	ales	bier	blatz	bock	buzzed	drinker	friederike	heineken	ipa	
brassieres > size	32b	34a	34c	36a	36b	36c	36d	38a	38c	
team > hockey	canucks	leafs	oilers	predators	senators	thrashers	whalers			
style	1950s	1960s	40's	50s	60's	60s	sixties			
style	1950's	1950s	1960s	40's	50s	60's	60s	sixties		
style > 70s	1970s	funk-fu	groovy	heartthrob	seventies					
subjective	coolest	greatest	sweetest							
subjective	happiness	merely	overrated							
region > demonym	azeri	bulgarian	indonesian	latvian	slovenian	tongan	tunisian	vietnamese		
region > demonym	chilean	ecuadorian	ghanaian	honduran	israeli	nigerian	slovene	spaniard	uruguayan	...
color	eggplant	emerald	mauve	plum	teal					
comics	aquaman	batarang	batwoman	bitman	catwoman	dastardly	dco	gotham	gotham's	...
disease	alzheimer's	alzheimers	autism	cancer	cerebral	cystic	disease	epilepsy	fibromyalgia	...
disney	ariel	aristocats	aurora	bambi	cadabby	cinderella	disney	disney's	dora	...

region>country>europe	albania
region>country>europe	andorra
region>country>europe	armenia
region>country>europe	austria
region>country>europe	azerbaijan
region>country>europe	belarus
region>country>europe	belgium
region>country>europe	bosnia

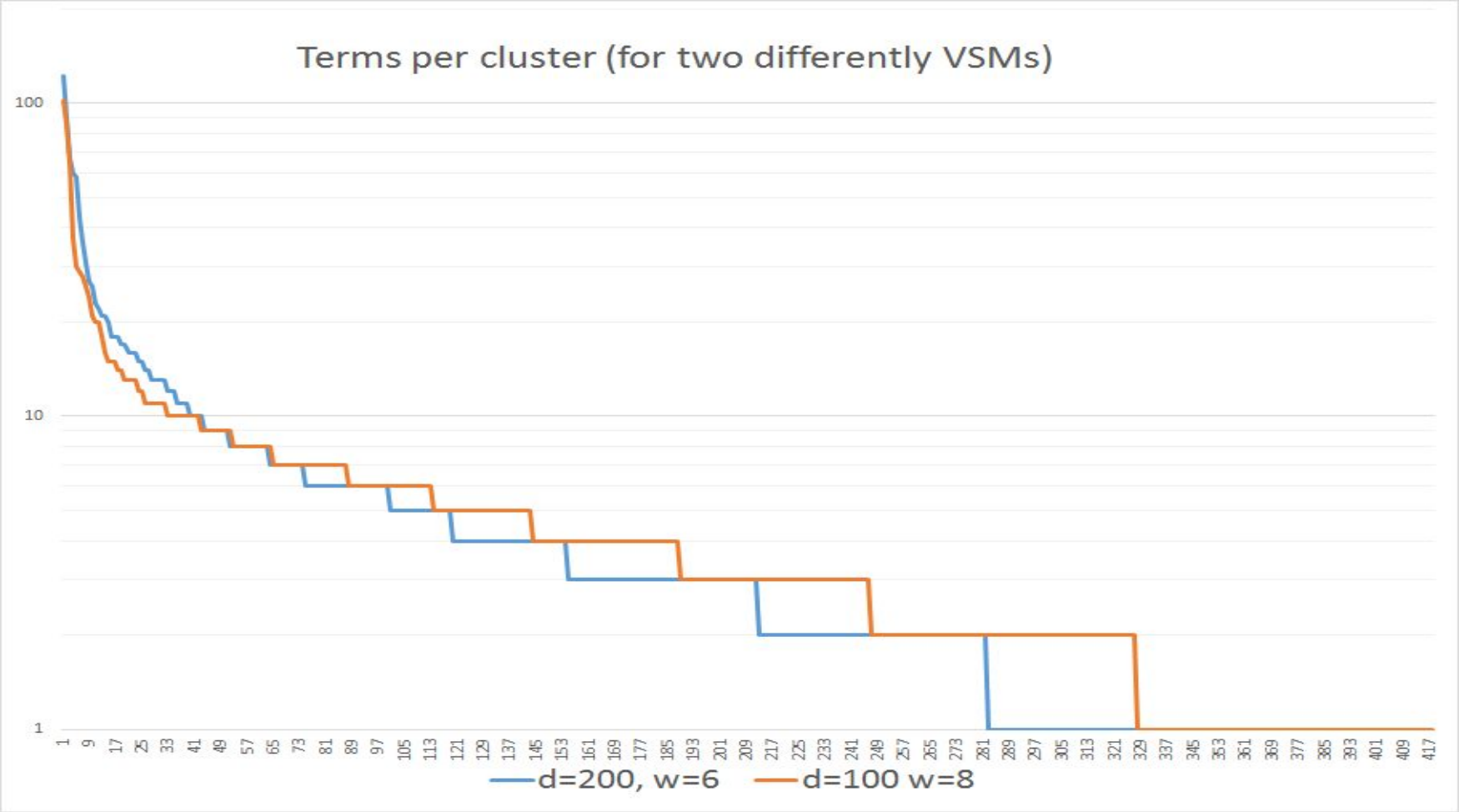
gemstone	amazonite
gemstone	azurite
gemstone	chrysocolla
gemstone	chrysoprase
gemstone	jade
gemstone	labradorite
gemstone	malachite
gemstone>crystal	bloodstone
gemstone>crystal	fluorite
gemstone>crystal	goldstone
gemstone>crystal	nephrite
gemstone>crystal	obsidian
gemstone>crystal	opalite
gemstone>crystal	rhodochrosite

month	august
month	december
month	feb
month	february
month	febuary
month	january
month	july
month	march
month	april
month	may
month	june
month	september
month	october
month	november
zodiac	aries
zodiac	capricorn
zodiac	gemini
zodiac	leo
zodiac	libra
zodiac	sagittarius
zodiac	scorpio
zodiac	pisces
zodiac	taurus
zodiac	cancer
zodiac	virgo
zodiac	aquarius

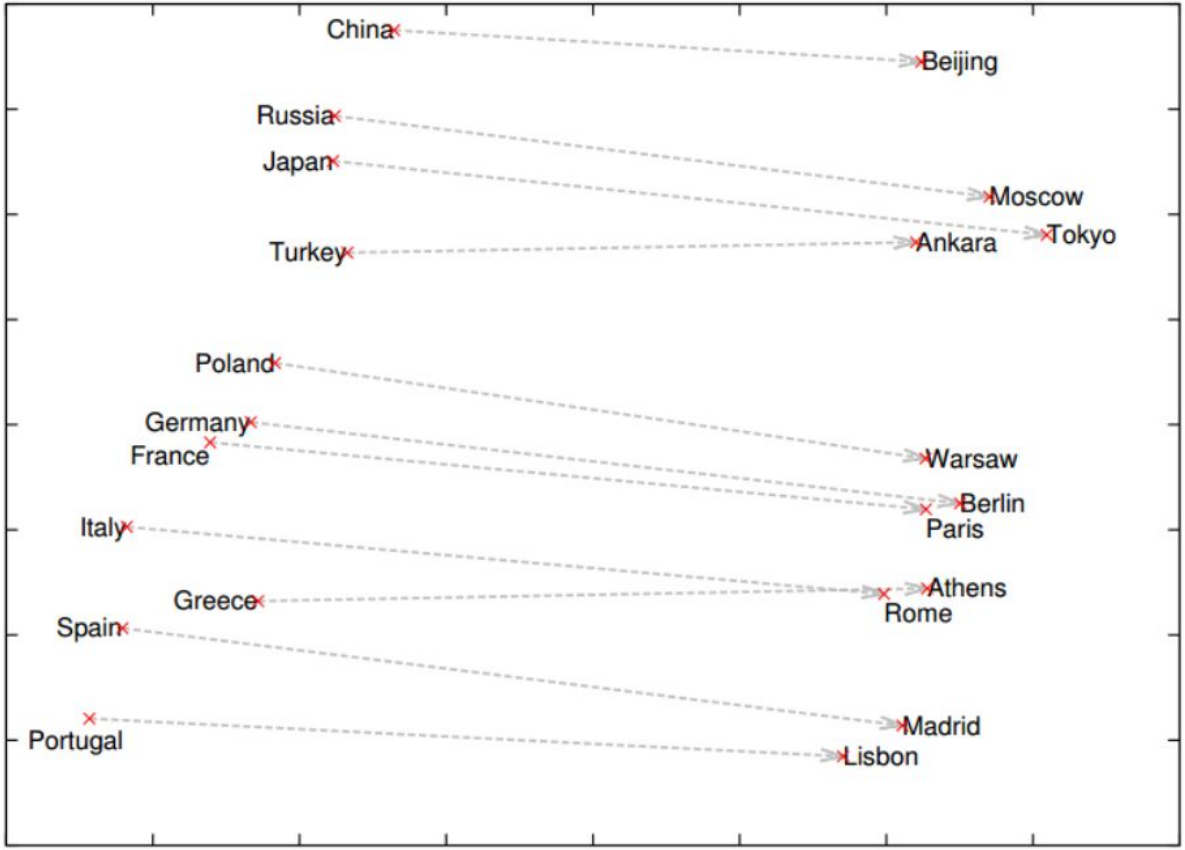
Cordlessphone-8pk 25
Cordlesss 25
Cpb-400 25
Cpb-400b 25
Cpb-400c 25
Cpb-403b 25
Cpb-466 25
Cpb-483 25
Cpb-486b 25
Cpb-487 25
Cph-400j 25
Cph-429 25
Cph-454 25
Cph-479b 25
Cph-485 25
Cph-489 25
Cph-496 25
Cph-514 25
Cph-516b 25
Cpl-536 25
Cpp-529z 25
Ct-11 25
Cx7525 25
Cxai5198 25
Cxai5698 25
D1364 25
D1384-2bk 25
D1660-2 25
D1660 25
D1660-3 25

Pwrc51 33
Pwrc61 33
Pwrc62 33
Pwrc81 33
Pwrc82 33
Pyle-home 33
Quam 33
Quintet 33
R-1650-c 33
R-1650-w 33
R-1800-c 33
R-2502-w 33
R-2502-wii 33
R-2650-c 33
R-2650-w 33
R-2650-wii 33
R-5502-w 33
R-5650-s 33
R-5650-w 33
R5800 33
R-5800-w 33
R5800wii 33
R800 33
Rb-41 33
Rb-51 33
Rb-61 33
Rc-52 33
Rc60i 33

Electronics

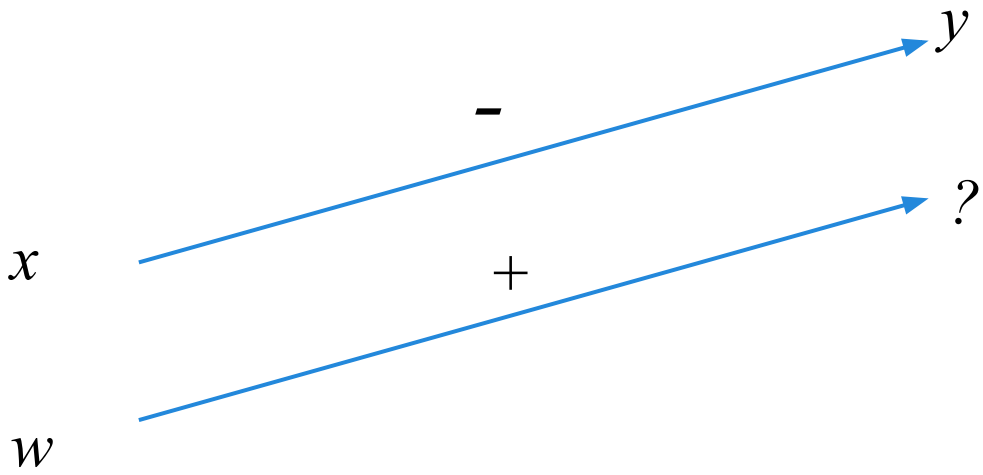

Terms per cluster (for two differently VSMs)
100
10
1
1 9 17 25 33 41 49 57 65 73 81 89 97 105 113 121 129 137 145 153 161 169 177 185 193 201 209 217 225 233 241 249 257 265 273 281 289 297 305 313 321 329 337 345 353 361 369 377 385 393 401 409 417
d=200, w=6
d=100 w=8

Word vectors – visualization using PCA



Compositional Queries (x - y + w)

```
$ ./word-analogy clothing.phrase-2014-06-16.bin


Enter three words (EXIT to break): x y w
```



Analogy - well known examples

Expression	*Nearest token*
Paris - France + Italy	Rome
bigger - big + cold	colder
sushi - Japan + Germany	bratwurst
Cu - copper + gold	Au
Windows - Microsoft + Google	Android
Montreal Canadiens - Montreal + Toronto	Toronto Maple Leafs

king - queen + man = ?

```
Enter three words: king queen man
                    Word                Distance
                   woman                0.609809
                    girl                0.501354
                    baby                0.441947
                    lady                0.415232
                    sexy                0.411406
                   loner                0.411260
```

word2vec.googlecode.com/svn/trunk/questions-words.txt

```
king queen man woman
king queen husband wife
dad mom king queen
father mother king queen
grandpa grandma king queen
grandson granddaughter king queen
king queen policeman policewoman
...
```

Often not the hoped for semantics

bird	bird - fish + salmon	bird - red + crimson	bird - treatment + therapy
bird	salmon	bird	bird
mammal	bird	crimson	therapy
same-sized	**goose**	long-winged	mammal
reptile	**tern**	flightless	hedgehog
butterfly	**pheasant**	**moorhen**	sambar
wader	**plover**	reptile	reptile
lizard	**pipit**	lizard	lizard
insect	**gull**	**sea-bird**	**moorhen**
long-winged	**warbler**	babirusa	butterfly
tern	smoked	butterfly	frugivorous

```
/mnt/word2vec/trunk$ ./demo-phrases.sh
--2015-02-06 08:03:12--  http://www.statmt.org/wmt14/training-monolingual-news-crawl/news.2012.en.shuffled.gz
2015-02-06 08:05:08 (6.46 MB/s) - `news.2012.en.shuffled.gz' saved [786717767/786717767]
Starting training using file news.2012.en.shuffled-norm0
Words processed: 296900K     Vocab size: 33198K
Vocab size (unigrams + bigrams): 18838711
Words in train file: 296901342
Words written: 296900K
      real    6m36.109s user    6m26.504s sys     0m9.569s
Starting training using file news.2012.en.shuffled-norm0-phrase0
Words processed: 280500K     Vocab size: 38761K
Vocab size (unigrams + bigrams): 21728781
Words in train file: 280513979
Words written: 280500K
      real    6m22.489s user    6m12.515s sys     0m9.933s
Starting training using file news.2012.en.shuffled-norm1-phrase1
Vocab size: 681320
Words in train file: 283545447
Alpha: 0.048486  Progress: 3.03%  Words/thread/sec: 125.85k
```

You Are Here

- ~~Overview~~
- ~~Text Preprocessing~~
- ~~Syntactic Parsing~~
- ~~Shallow Semantic Analysis~~
- ~~Text Classification & Clustering~~
- ~~Information Extraction~~
- ~~Word Relatedness Functions~~

Other NLP Tasks

- Sentiment Analysis
- Textual Entailment
- cogcomp.cs.illinois.edu/demo/NESim
- cogcomp.cs.illinois.edu/demo/WNSim
- ...

Other Introductions to NLP

- hakkalabs.co/articles/introduction-natural-language-processing
- slideshare.net/Nicolas_Nicolov/machine-learning-14528792

- Paco Nathan's text analytics at scale with PySpark and notebooks
  - http://youtube.com/watch?v=iVbTFIdZ1Io
  - http://slideshare.net/pacoid/microservices-containers-and-machine-learning-50862677

Fin!

Fin!

Fin!

Annotator Enhancements

- Semantic Relatedness: WordNet
  - Prerequisite: POS, WSD
  - Concepts
  - Synonyms, is-a, part-of
- SRL: Assert
  - Predicate-argument relationship
- SBD, NER, Co-ref: Lingpipe

Relatedness between concepts in the documents and the question.

Notice how the concept ‘score’ change as the distance of relatedness increased.

WordNetID	ConceptName	depth=3		depth=2		depth=1		depth=0	
		rank	score	rank	score	rank	score	rank	score
2064081	dog(*)	1	2453	1	2453	1	2453	1	2453
8096606	police	2	1695	2	1695	2	1695	6	1095
18345	kind of	3	1400	3	1400	3	1400	2	1400
8234204	law enforcement agency(*)	4	1268	4	1268	4	1268	3	1268
784188	project	5	1268	5	1268	5	1268	4	1268
1417230	breed(*)	6	1233	6	1233	6	1233	5	1233
10292737	Police Officer	7	1225	7	1225	NA	0	NA	0
2086825	Police dog	8	1191	8	1191	NA	0	NA	0
15002401	year	NA	0	9	897	7	897	7	897
2086633	German shephed	9	963	NA	0	NA	0	NA	0
2262856	two	NA	0	NA	0	8	893	8	893
10163106	officer	NA	0		0	9	769	9	769

Huffing Coding

```
void TrainModel() {
  if (vocab_file) ReadVocab(); else LearnVocabFromTrainFile();
  if (save_vocab_file) SaveVocab();
  InitNet();
  if (negative) InitUnigramTable();
…
```

	word	frequency	codelen	code[0]	code[1]	code[2]	code[3]	code[4]	code[5]	code[6]	code[7]	...	code[19]	code[20]	code[21]
1	</s>	1182871	4	31095	31094	31092	31087								
2	watch	399047	5	31095	31093	31090	31083	31070							
3	silver	299594	6	31095	31094	31092	31088	31079	31062						
4	-	279078	6	31095	31094	31091	31086	31077	31058						
5	ring	237899	6	31095	31094	31091	31085	31074	31053						
6	sterling	200634	6	31095	31093	31090	31083	31070	31045						
7	gold	187886	6	31095	31093	31089	31082	31068	31042						
8	black	170488	6	31095	31093	31089	31081	31066	31038						
9	with	161824	7	31095	31094	31092	31088	31080	31065	31035					
10	size	161334	7	31095	31094	31092	31088	31080	31064	31034					
	...	...	...	...	...	...	...	...	...	...					
31,094	favored	5	22	31095	31094	31092	31088	31080	31064	31034	30981	...	11415	5438	4
31,095	weissbraeu	5	22	31095	31094	31092	31088	31080	31064	31034	30981	...	11415	5438	4
31,096	dylb-151	5	22	31095	31094	31092	31088	31080	31064	31034	30981	...	11415	5438	3
31,097	bierce	5	22	31095	31094	31092	31088	31080	31064	31034	30981	...	11415	5438	3

id	recordTextContent	v1Cat
1	Monty Python 's The Meaning of Life 30th Anniversary Edition (Blu-ray + Digital Copy + UltraViolet)	AE
2	Rain-X 800002243 Glass Treatment - 7 oz.	AU
3	Wheelskins Genuine Deerskin DRIVING GLOVES - Tan (Size Large)	AU
4	Rear Bumper Top Surface Protector Cover Fits 2011 2012 2013 2014 BMW X3	AU
5	Outwitting the Devil: The Secret to Freedom and Success	BK
6	Shi no fuchi wo mita otoko (Masao Yoshida, man and saw the edge of death of Sun five hundred Fukushima Daiichi nuclear power plant)	BK
7	Kill Your Friends: A Novel	BK
8	APC XS BX1000G 1000VA Tower UPS	CE
9	OmniMount 1 B - 1 Shelf Wall Furniture (Black / Dark Glass)	CE
10	Logitech Wireless Headset F540 with Stereo Game Audio	CE
11	Belkin Surge Protector	CE
12	AE-SELECT Replacement Part TV Power Supply	CE
13	NEW Sony Xperia Honami Z1 16gb Black 20mp Camera 5" 4g LTE ★ Factory Unlocked Best Gift Fast Shipping Ship All the World	CM
14	Call of Duty: Ghosts - Xbox One	GM
15	Xbox 360 4GB	GM
16	Lemon Tea Tree Deodorant 2.50 Ounces	HB
17	Rachael Ray Oven Lovin ' Nonstick Bakeware 9-Inch-by-13-Inch Cake Pan, Orange	HG
18	DC Comics Classic TV Series Batman and Robin Action Figure, 2-Pack	HO
19	Sterling Silver "Life Is A Journey, Not a Destination, Enjoy The Moments" Cuff Bracelet	JW
...	...	...
335,037,291	Timex Expedition WS4 Widescreen 4-Function Watch (Black/Green)	SF

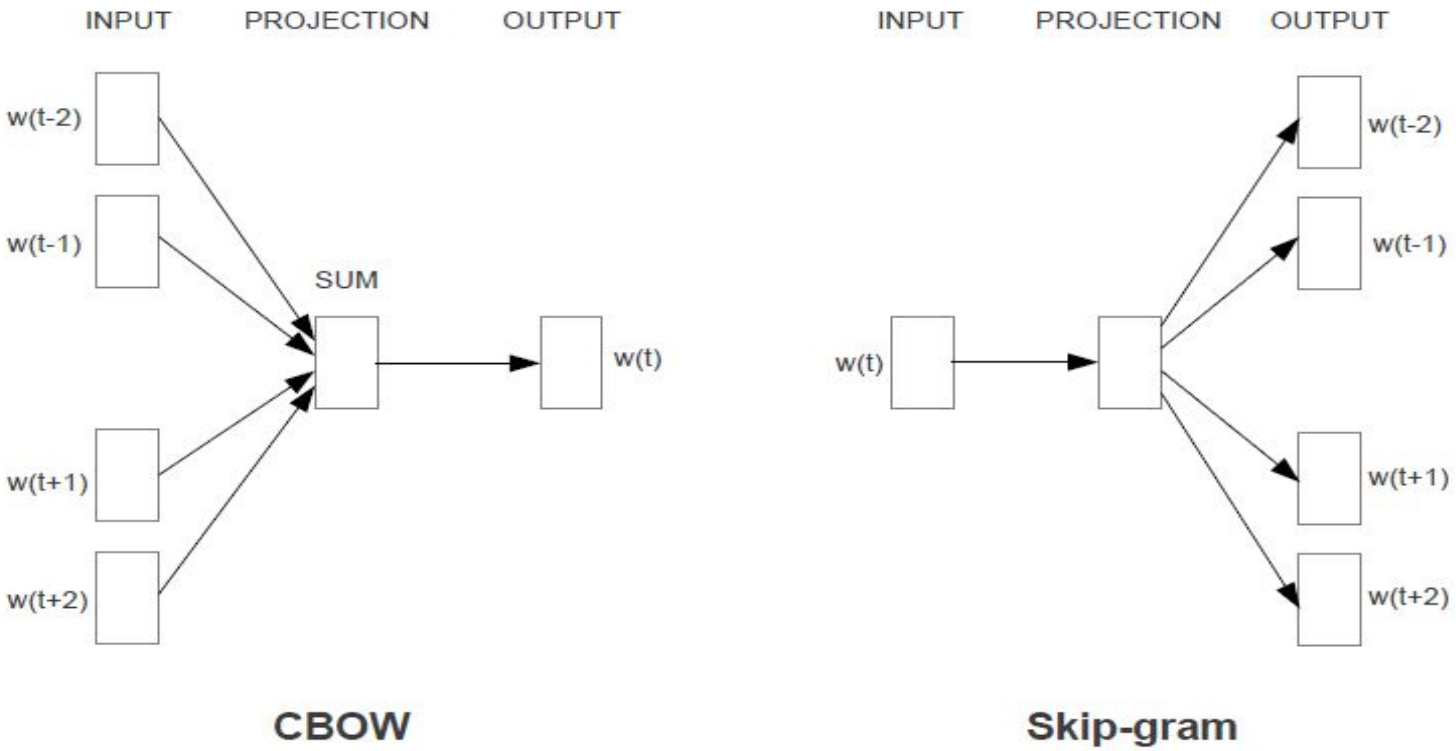
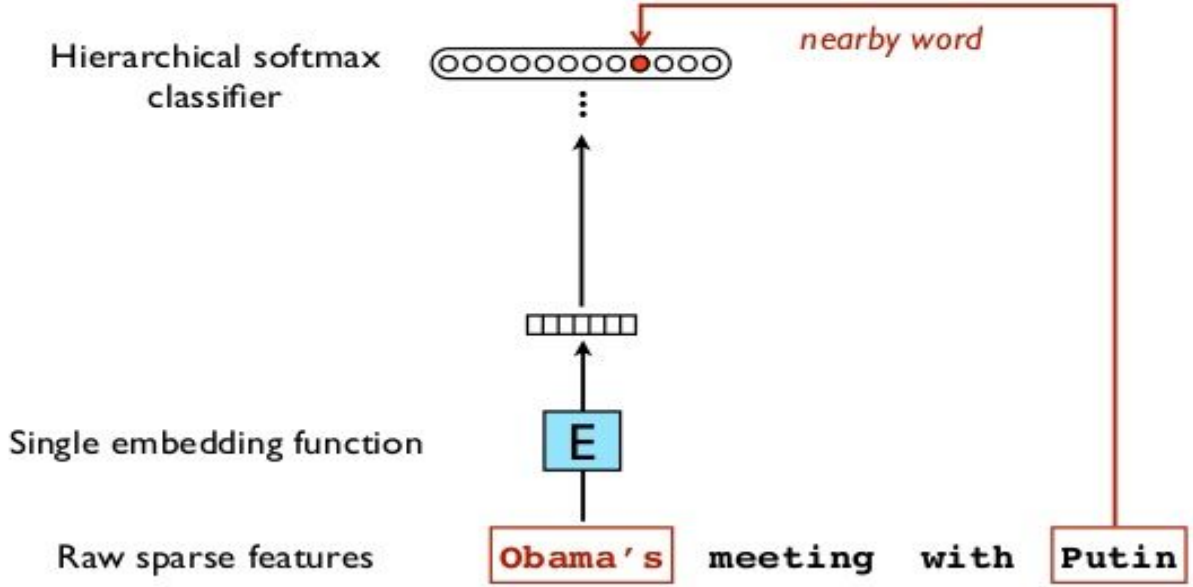


Figure 1: New model architectures. The CBOW architecture predicts the current word based on the context, and the Skip-gram predicts surrounding words given the current word.

How Can We Learn the Embeddings?

Skipgram Text Model



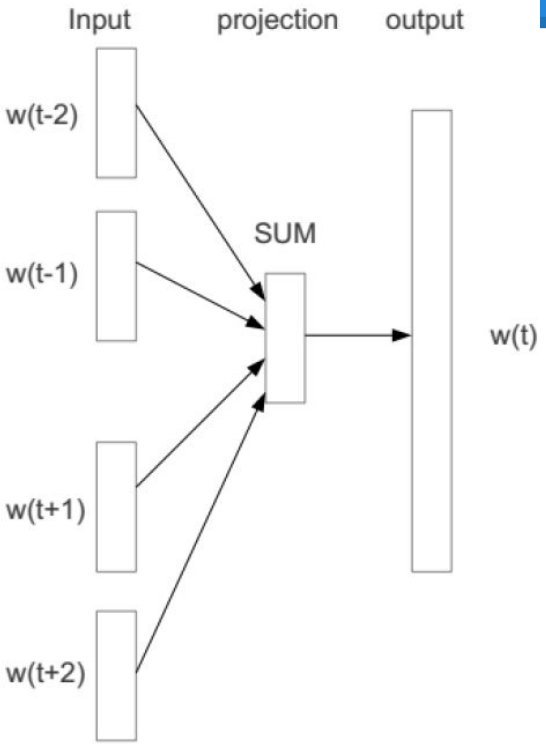
Mikolov, Chen, Corrado and Dean. *Efficient Estimation of Word Representations in Vector Space*, http://arxiv.org/abs/1301.3781.

Google

```
    next_random = next_random * (unsigned long long)25214903917 + 11;  wikipedia …
Linear_congruential_generator
    b = next_random % window;
    if (cbow) {  //train the cbow architecture
...
    } else {  //train skip-gram
      for (a = b; a < window * 2 + 1 - b; a++) if (a != window) {
        c = sentence_position - window + a;
        if (c < 0) continue;
        if (c >= sentence_length) continue;
        last_word = sen[c];
        if (last_word == -1) continue;
        l1 = last_word * layer1_size;
        for (c = 0; c < layer1_size; c++) neu1e[c] = 0;
        // HIERARCHICAL SOFTMAX
        if (hs) for (d = 0; d < vocab[word].codelen; d++) {
          f = 0;
          l2 = vocab[word].point[d] * layer1_size;
          // Propagate hidden -> output
          for (c = 0; c < layer1_size; c++) f += syn0[c + l1] * syn1[c + l2];
          if (f <= -MAX_EXP) continue;
```

Word vectors – various architectures

- The 'continuous bag-of-words model' (CBOW) adds inputs from words within short window to predict the current word
- The weights for different positions are shared
- Computationally much more efficient than normal NNLM
- The hidden layer is just linear

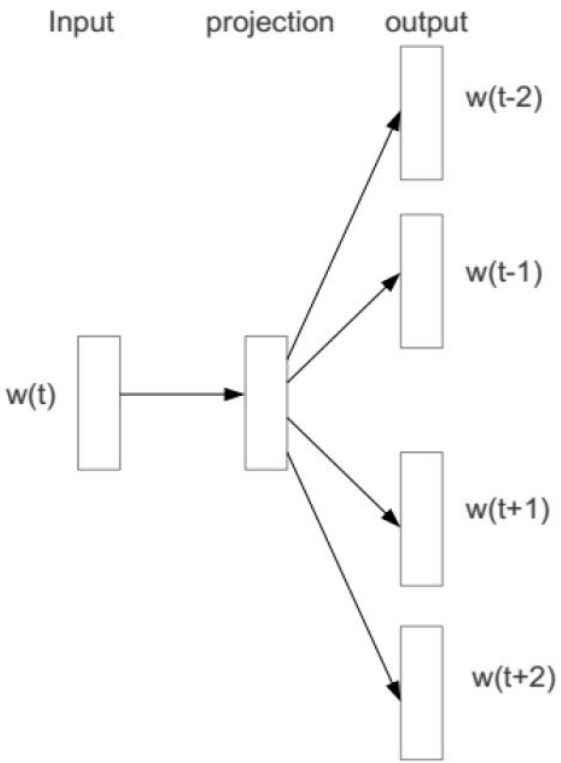


Skip-gram - predict the context given a word

- Input: w_i,
- Output: w_{i-1}, w_{i-2}, w_{i+1}, w_{i+2}.
- Training instances can be created by *skipping* a constant number of words in its context, so for example, w_{i-3}, w_{i-4}, w_{i+4}, w_{i+3}
-

Word vectors – various architectures

- We can reformulate the CBOW model by predicting surrounding words using the current word
- This architectures is called 'skip-gram NNLM'
- If both are trained for sufficient number of epochs, their performance is similar



Negative Sampling

```
printf("\t\tNumber of negative examples; default is 5, common values are 3 - 10 (0 = not used)\n");
if (negative > 0) InitUnigramTable();

a = posix_memalign((void **)&syn1neg, 128, (long long)vocab_size * layer1_size * sizeof(real));
for (a = 0; a < vocab_size; a++) for (b = 0; b < layer1_size; b++)     syn1neg[a * layer1_size + b] = 0;
…
for (d = 0; d < negative + 1; d++) {
        if (d == 0) {            target = word;           label = 1;
        } else {
          next_random = next_random * (unsigned long long)25214903917 + 11;
          target = table[(next_random >> 16) % table_size];
          if (target == 0) target = next_random % (vocab_size - 1) + 1;
          if (target == word) continue;
          label = 0;
        }
```

```
    next_random = next_random * (unsigned long long)25214903917 + 11;  wikipedia …
Linear_congruential_generator
    b = next_random % window;
    if (cbow) {  //train the cbow architecture
...
    } else {  //train skip-gram
      for (a = b; a < window * 2 + 1 - b; a++) if (a != window) {
        c = sentence_position - window + a;
        if (c < 0) continue;
        if (c >= sentence_length) continue;
        last_word = sen[c];
        if (last_word == -1) continue;
        l1 = last_word * layer1_size;
        for (c = 0; c < layer1_size; c++) neu1e[c] = 0;
        // HIERARCHICAL SOFTMAX
        if (hs) for (d = 0; d < vocab[word].codelen; d++) {
          f = 0;
          l2 = vocab[word].point[d] * layer1_size;
          // Propagate hidden -> output
          for (c = 0; c < layer1_size; c++) f += syn0[c + l1] * syn1[c + l2];
          if (f <= -MAX_EXP) continue;
```

publishers.viglink.com/admin/dictionary

Apps vpn gm PT cal prod Git adm dict mltms gmAcct UPM IP[y] » Other b

☑ Exact match

Show 25 entries

Search: topaz

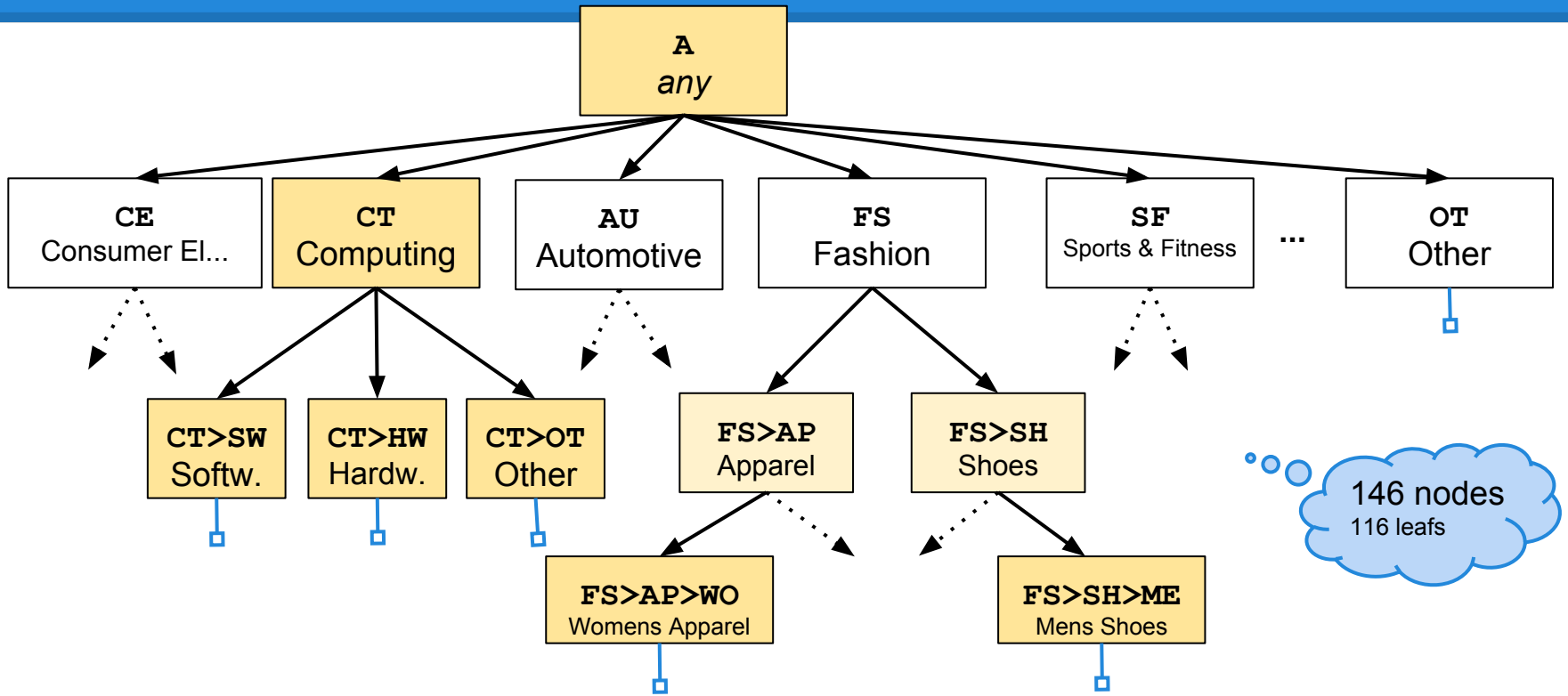
Showing 1 to 2 of 2 entries (filtered from 590,246 total entries)

			Jewelry and Watches		
Id	**Term**	**Dictionary Term Type**	**Industry/Category**	**Added By**	**Date**
382797	topaz	Product Feature	Jewelry and Watches	gabor@viglink.com	05/08/2013
433769	topaz	Black Listed Term	Jewelry and Watches	gabor@viglink.com	05/17/2013

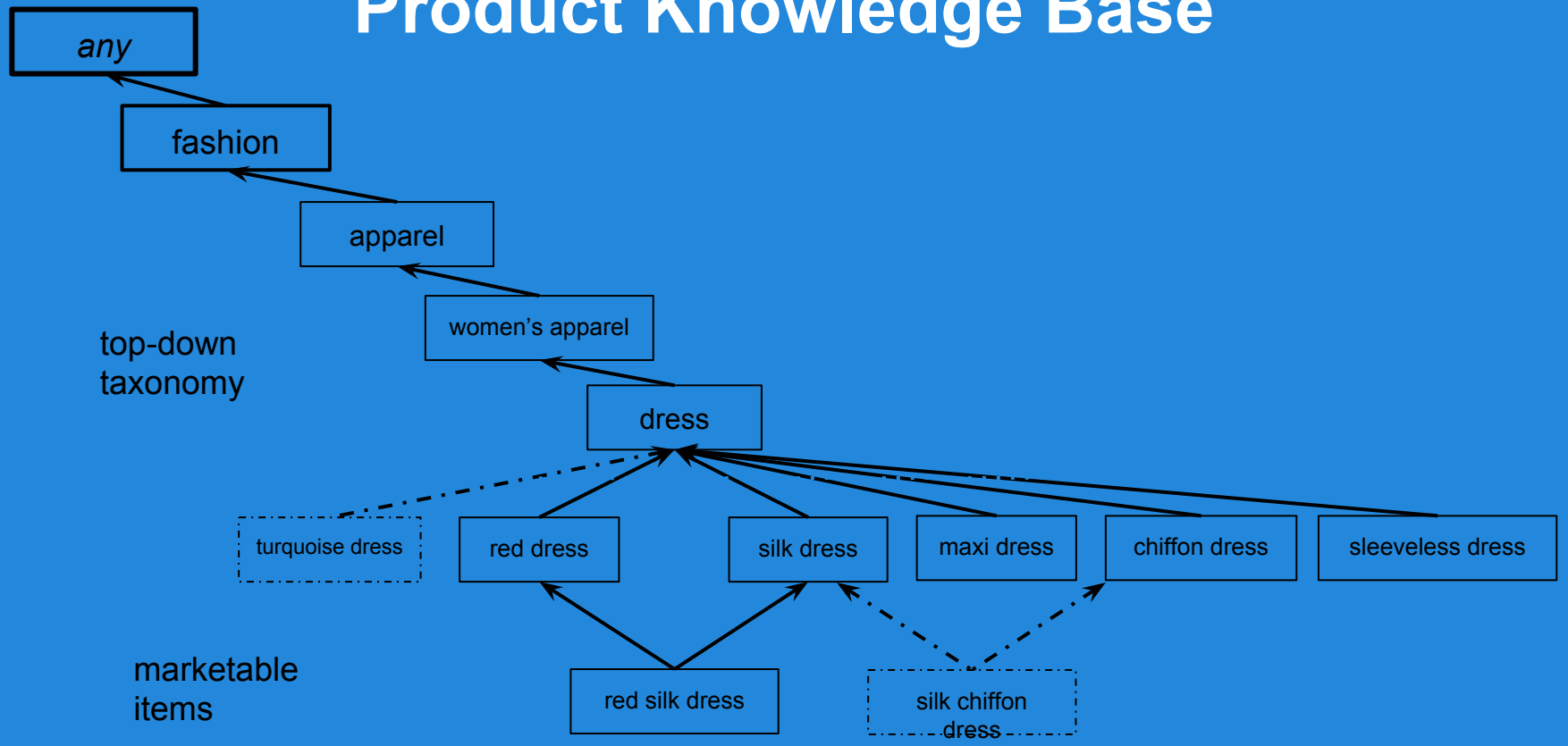
Showing 1 to 2 of 2 entries (filtered from 590,246 total entries)

	term	category
1	block heater	AU
2	powertrain control module	AU
3	crankshaft position sensor	AU
4	firestone firehawk tires	AU
5	in line rca potentiometer	CE
6	bose bluetooth headset	CE
7	bose computer music monitor	CE
8	de cecco pasta	FD
9	barbour beaufort jacket	FS
10	camel gloverall duffle coat	FS
11	ralph lauren button-down shirt	FS
12	eyeshadow palette	HB
13	polka-dot towels	HG
...	...	...
185,675	multi-strand necklace	JW

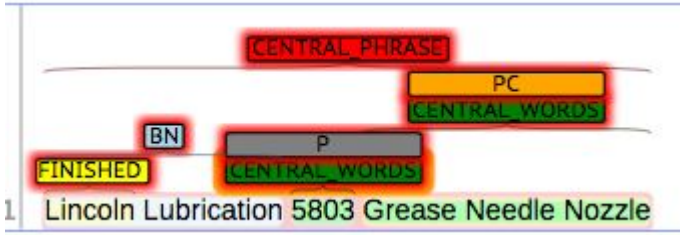
Product Category Hierarchy

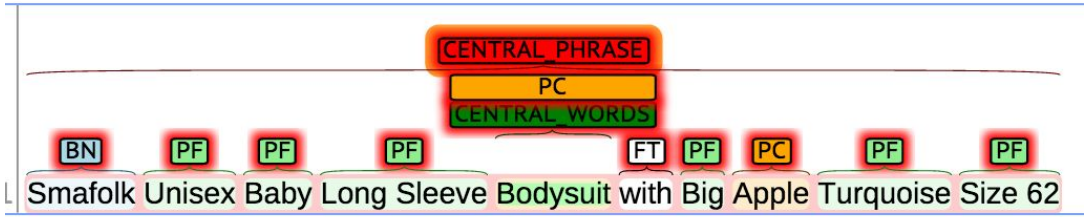


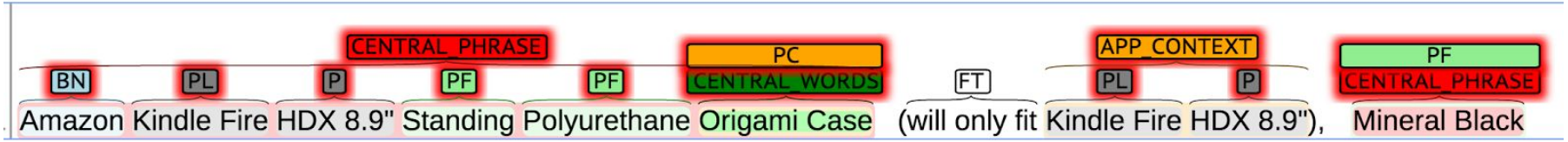
Product Knowledge Base



Case Study 2: Product Offer Parsing







Number Quantification

cogcomp.cs.illinois.edu/demo/Quantities