# Social Media & Text Analysis

## lecture 4 - Twitter NLP Pipeline

## Tokenization, Normalization, POS/NE Tagging

**CSE 5539-0010 Ohio State University**
**Instructor: Alan Ritter**
**Website: socialmedia-class.org**

# How to read a paper?

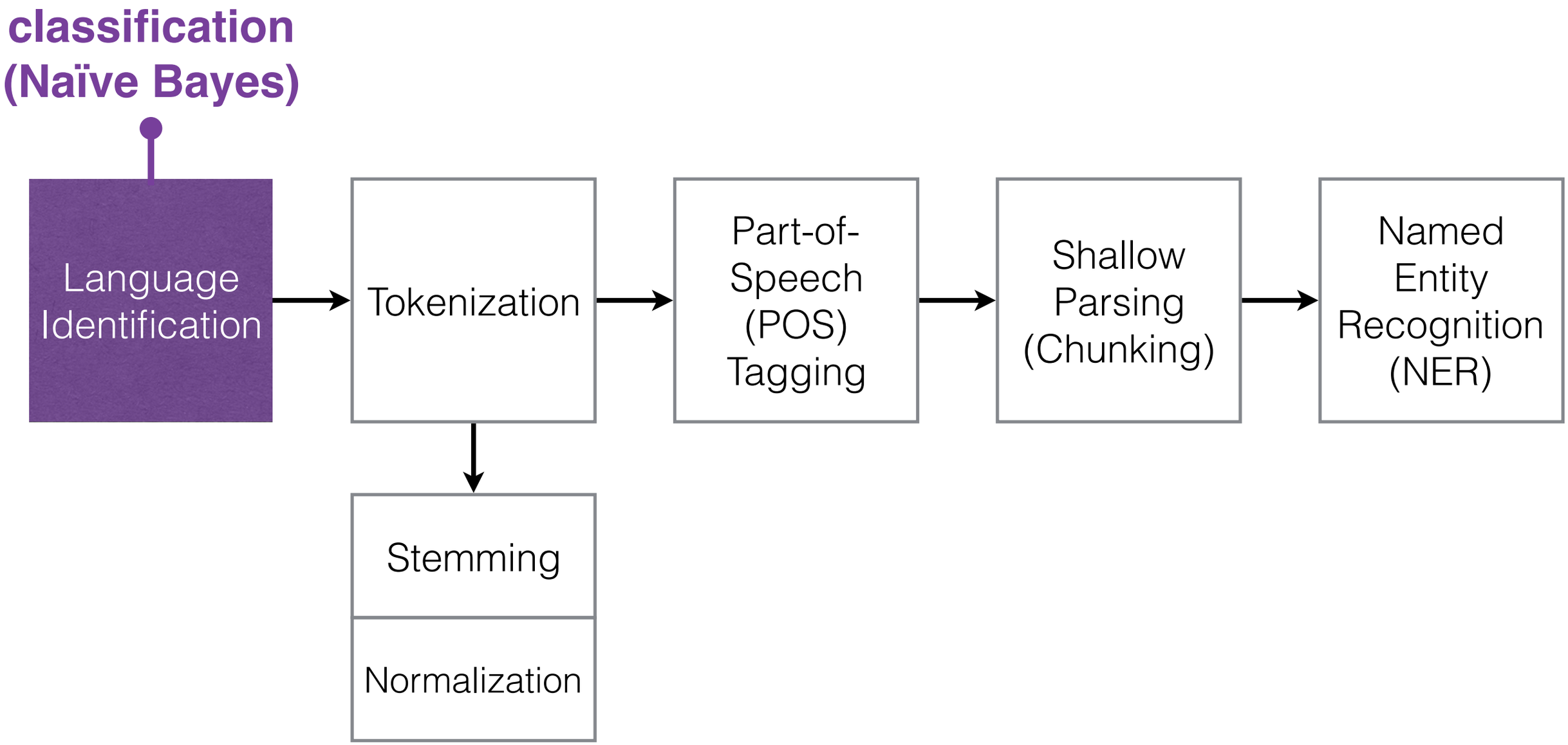
Read *creatively*: Reading a paper critically is easy, in that it is always easier to tear something down than to build it up. Reading creatively involves harder, more positive thinking.

What are the good ideas in this paper? Do these ideas have other applications or extensions that the authors might not have thought of? Can they be generalized further? Are there possible improvements that might make important practical differences? If you were going to start doing research from this paper, what would be the next thing you would do?

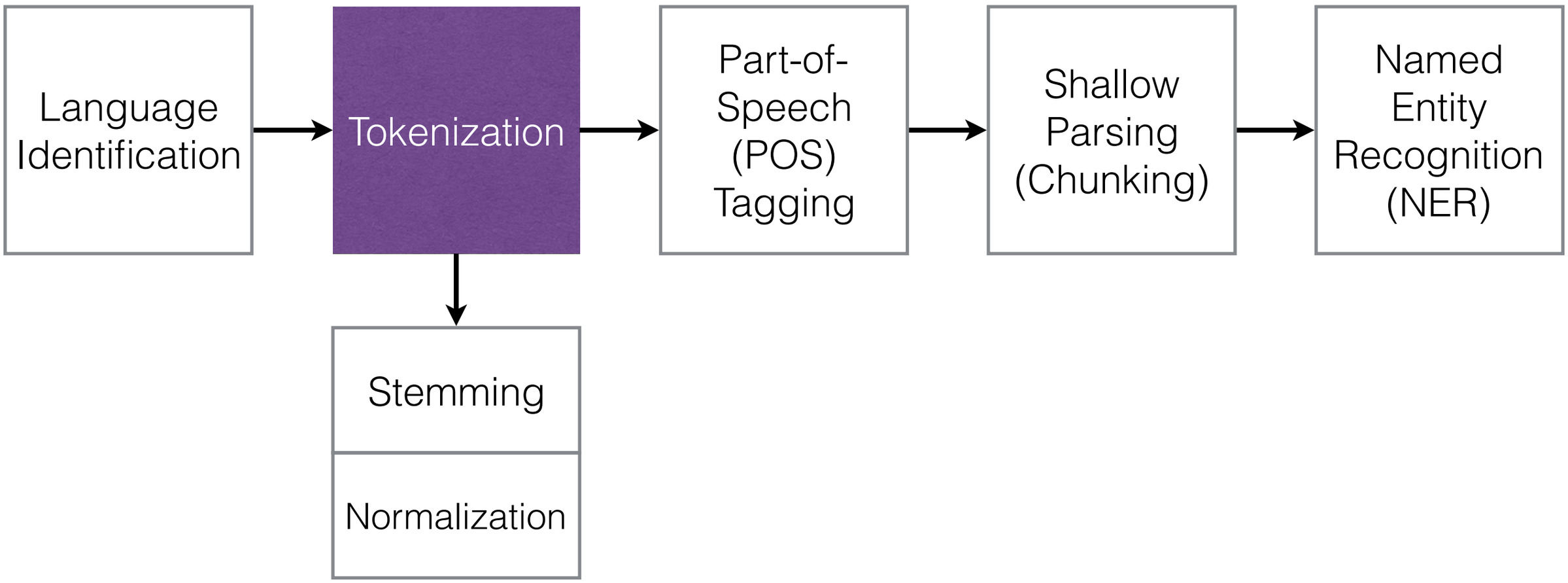
— Michael Mitzenmacher

https://www.eecs.harvard.edu/~michaelm/postscripts/ReadPaper.pdf

# Summary


classification (Naïve Bayes)
Language Identification
Tokenization
Stemming
Normalization
Part-of-Speech (POS) Tagging
Shallow Parsing (Chunking)
Named Entity Recognition (NER)

# NLP Pipeline



# Tokenization

- breaks up the string into words and punctuation
- need to handle:
  - abbreviations (“jr.”), number (“5,000”) …

```
seas479:training weixu$ ./penn-treebank-tokenizer.perl
Tokenizer v3
Language: en


Ms. Hilton last year called Mr. Rothschild “the love of my life.”
Ms. Hilton last year called Mr. Rothschild “ the love of my life . ”
```

input

output

# Tokenization

- for Twitter, additionally need to handle:
  - emoticons, urls, #hashtags, @mentions …

```
>>> import twokenize
>>> input = "Clowns are pretty gross tho O.o (I'm afraid of clow
ns :p) ask.fm/a/cc301167"
>>> twokenize.tokenizeRawTweetText(input)
['Clowns', 'are', 'pretty', 'gross', 'tho', 'O.o', '(', "I'm", '
afraid', 'of', 'clowns', ':p', ')', 'ask.fm/a/cc301167']
```

input

output

# Tool: twokenize.py

GitHub, Inc. [US] https://github.com/myleott/ark-twokenize-py/blob/master/twokenize.py

This repository Search Pull requests Issues Gist

myleott / ark-twokenize-py

branch: master ▾ ark-twokenize-py / twokenize.py

myleott on Apr 29, 2013 Initial commit

1 contributor

Executable File | 300 lines (247 sloc) | 12.993 kB Raw

```
1  # -*- coding: utf-8 -*-
2  """
3  Twokenize -- a tokenizer designed for Twitter text in English and some other European languages.
```

# Tool: twokenize.py

```
Twokenize -- a tokenizer designed for Twitter text in English and some other European languages.
This tokenizer code has gone through a long history:

(1) Brendan O'Connor wrote original version in Python, http://github.com/brendano/tweetmotif
       TweetMotif: Exploratory Search and Topic Summarization for Twitter.
       Brendan O'Connor, Michel Krieger, and David Ahn.
       ICWSM-2010 (demo track), http://brenocon.com/oconnor_krieger_ahn.icwsm2010.tweetmotif.pdf
(2a) Kevin Gimpel and Daniel Mills modified it for POS tagging for the CMU ARK Twitter POS Tagger
(2b) Jason Baldridge and David Snyder ported it to Scala
(3) Brendan bugfixed the Scala port and merged with POS-specific changes
    for the CMU ARK Twitter POS Tagger
(4) Tobi Owoputi ported it back to Java and added many improvements (2012-06)

Current home is http://github.com/brendano/ark-tweet-nlp and http://www.ark.cs.cmu.edu/TweetNLP
```

# Tokenization

- main techniques:
  - hand-crafted rules as regular expressions

# Regular Expression

- a pattern matching language
- invented by American Mathematician Stephen Kleene in the 1950s
- used for search, find, replace, validation … (very frequently used when dealing with strings)
- supported by most programming languages
- easy to learn, but hard to master

# Regular Expression

```
147    Hashtag = "#[a-zA-Z0-9_]+"
```

- [] indicates a set of characters:
  - [amk] will match ‘a’, ‘m’, or ‘k’
  - [a-z] will match any lowercase letter (‘abcdefghijklmnopqrstuvwxyz’)
  - [a-zA-Z0-9_] will match any letter or digit or ‘_’
- + matches 1 or more repetitions of preceding RE

# Regular Expression

```
147    Hashtag = "#[a-zA-Z0-9_]+"
```

- will match strings that:
  - start with a ‘#’
  - follow with one or more letters/digits/‘_’

# Regular Expression

```
147    Hashtag = "#[a-zA-Z0-9_]+"
```

```
>>> import re
>>> Hashtag = "#[a-zA-Z0-9_]+"
>>> hashtagpattern = re.compile(Hashtag)
>>> hashtagpattern.findall("So that's what #StarWars")
['#StarWars']
```

# Regular Expression

```
133    Hearts = "(?:<+/?3+)+"
```

- will match strings that:
  - start with one or more ‘<‘
  - then maybe a ‘/’
  - then one or more ‘3’
  - and maybe repetitions of the above

# Regular Expression

```
133    Hearts = "(?:<+/?3+)+"
```

- ‘+’ matches 1 or more repetitions of the preceding RE
  - ‘<+’ matches ‘<’, ‘<<’, ‘<<<’ …
  - ‘3+’ matches ‘3’, ‘33’, ‘333’ …
- ‘?’ matches 0 or 1 repetitions of the preceding RE
  - ‘/?’ matches ‘/’ or nothing (so handles ‘</3’)
- (?: …) is a non-capturing version of ( … )
- ( … ) matches whatever RE is inside the parentheses

# Regular Expression

Now, if I apply the regex below over it...

```
(https?|ftp)://([^/\r\n]+)(/[^\r\n]*)?
```

... I would get the following result:

```
Match "http://stackoverflow.com/"
     Group 1: "http"
     Group 2: "stackoverflow.com"
     Group 3: "/"

Match "https://stackoverflow.com/questions/tagged/regex"
     Group 1: "https"
     Group 2: "stackoverflow.com"
     Group 3: "/questions/tagged/regex"
```

But I don't care about the protocol -- I just want the host and path of the URL. So, I change the regex to include the non-capturing group `(?:)`.

```
(?:https?|ftp)://([^/\r\n]+)(/[^\r\n]*)?
```

Now, my result looks like this:

```
Match "http://stackoverflow.com/"
     Group 1: "stackoverflow.com"
     Group 2: "/"

Match "https://stackoverflow.com/questions/tagged/regex"
     Group 1: "stackoverflow.com"
     Group 2: "/questions/tagged/regex"
```

See? The first group has not been captured. The parser uses it to match the text, but ignores it later, in the final result.



https://stackoverflow.com/questions/3512471/what-is-a-non-capturing-group-in-regular-expressions

# Regular Expression

```
133    Hearts = "(?:<+/?3+)+"
```

```
>>> import re
>>> Hearts = "(?:<+/?3+)+"
>>> heartspattern = re.compile(Hearts)
>>> heartspattern.findall("I <3 u <3<333333")
['<3', '<3<333333']
>>> heartspattern.findall("sooo sad </3")
['</3']
```

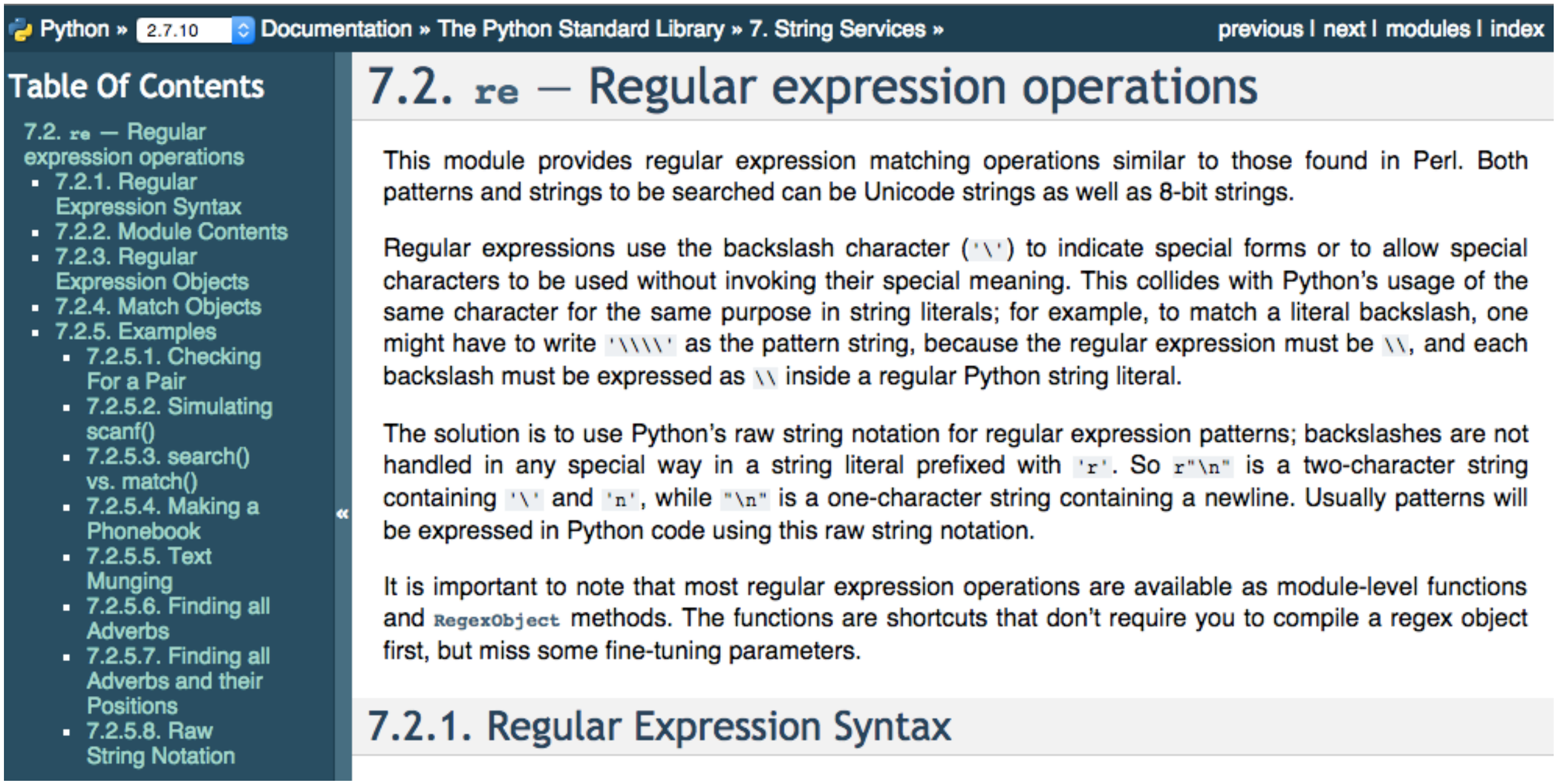
# Regular Expression

```
133    Hearts = "(?:<+/?3+)+"
```

```
Python 2.7.10 (default, Feb  7 2017, 00:08:15)
[GCC 4.2.1 Compatible Apple LLVM 8.0.0 (clang-800.0.34)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> import re
>>> heart1 = "(<+/?3+)+"
>>> heartpattern1 = re.compile(heart1)
>>> heartpattern1.findall("I <3 u <3<333")
['<3', '<333']
>>>
>>> heart2 = "(?:<+/?3+)+"
>>> heartpattern2 = re.compile(heart2)
>>> heartpattern2.findall("I <3 u <3<333")
['<3', '<3<333']
>>>
>>>
```

# Regular Expression

- learn more (https://docs.python.org/2/library/re.html)



# Tokenization

- for Twitter, additionally need to handle:
  - emoticons, urls, #hashtags, @mentions …

```
>>> import twokenize
>>> input = "Clowns are pretty gross tho O.o (I'm afraid of clow
ns :p) ask.fm/a/cc301167"
>>> twokenize.tokenizeRawTweetText(input)
['Clowns', 'are', 'pretty', 'gross', 'tho', 'O.o', '(', "I'm", '
afraid', 'of', 'clowns', ':p', ')', 'ask.fm/a/cc301167']
```

input

output

# Emoticons

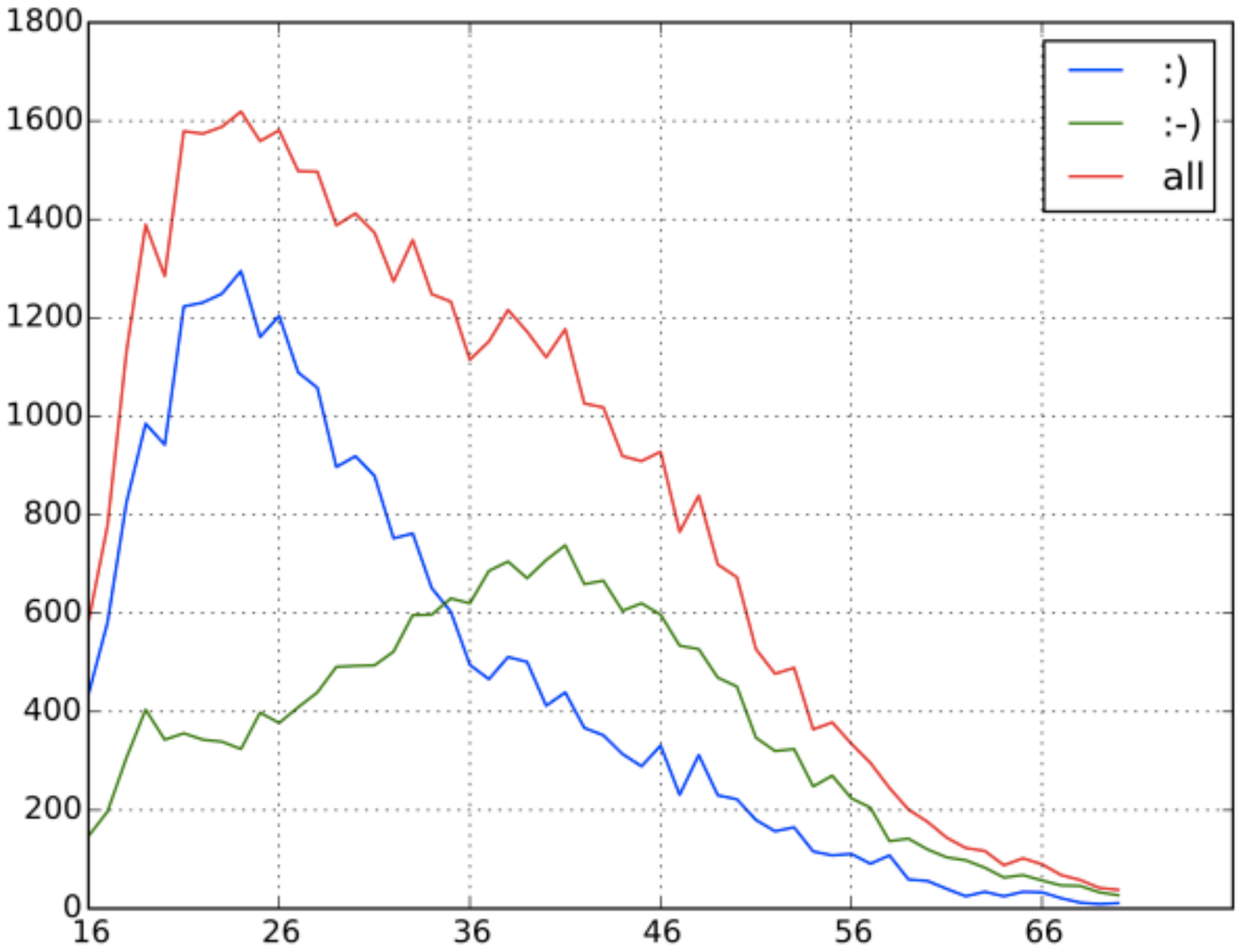



**Figure 3: Usage of emoticons with and without nose by age group, aggregated over all countries**

Dirk Hovy, Anders Johannsen, and Anders Søgaard.
User review sites as a resource for large-scale sociolinguistic studies. WWW, 2015

# Emoticons

With respect to gender, we find that women tend to use the noseless variant significantly more than men, except for France, where the difference between genders is not statistically significant at the chosen level.

| country | AGE | | GENDER |
|---|---|---|---|
| | Spearman $\rho$ | significant | significant |
| Denmark | 0.89 | yes | yes |
| France | 0.63 | yes | no |
| Germany | 0.83 | yes | yes |
| UK | 0.83 | yes | yes |
| US | 0.82 | yes | yes |

Dirk Hovy, Anders Johannsen, and Anders Søgaard.
User review sites as a resource for large-scale sociolinguistic studies. WWW, 2015

# Tokenization

- language dependent

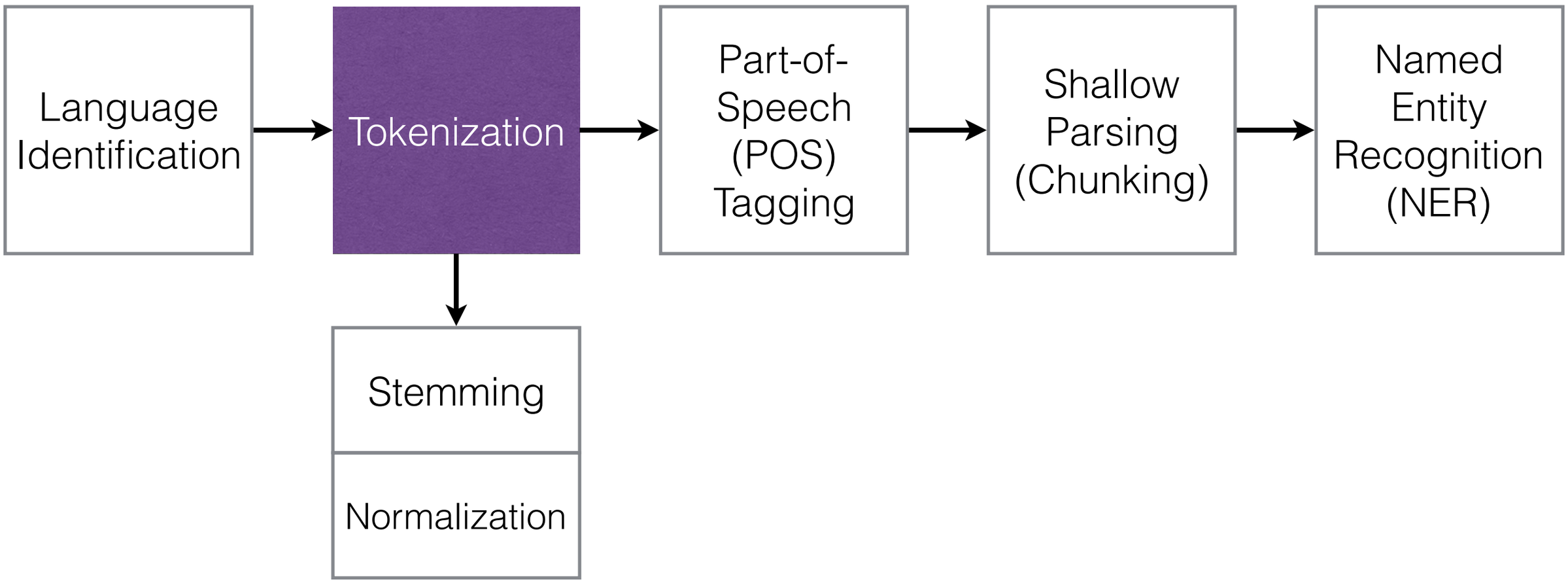
| | |
|---|---|
| 下雨天留客天留我不留 | Unpunctuated Chinese sentence |
| 下雨、天留客。天留、我不留！ | *It is raining, the god would like the guest to stay. Although the god wants you to stay, I do not!* |
| 下雨天、留客天。留我不？留！ | *The rainy day, the staying day. Would you like me to stay? Sure!* |

| | |
|---|---|
| 我喜欢新西兰花 | Unsegmented Chinese sentence |
| 我　喜欢　新西兰　花 | *I like New Zealand flowers* |
| 我　喜欢　新　西兰花 | *I like fresh broccoli* |

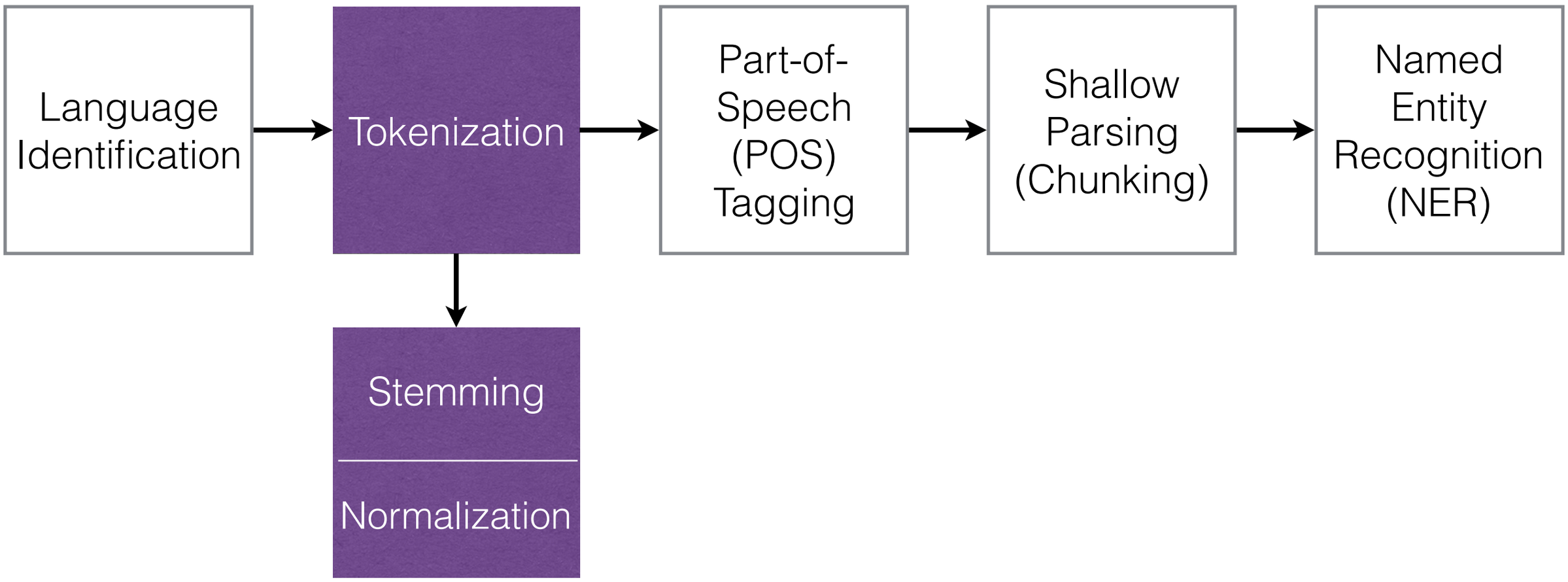


Source: http://what-when-how.com

# NLP Pipeline



# NLP Pipeline



# Stemming

- reduce inflected words to their word stem, base or root form (not necessarily the morphological root)

- studied since the 1960s

```
>>> from nltk.stem.porter import PorterStemmer
>>> stemmer = PorterStemmer()
>>> stemmer.stem('automate')
'autom'
>>> stemmer.stem('automates')
'autom'
>>> stemmer.stem('automation')
'autom'
```

# Stemming

- different steamers: Porter, Snowball, Lancaster, Morpha, etc…

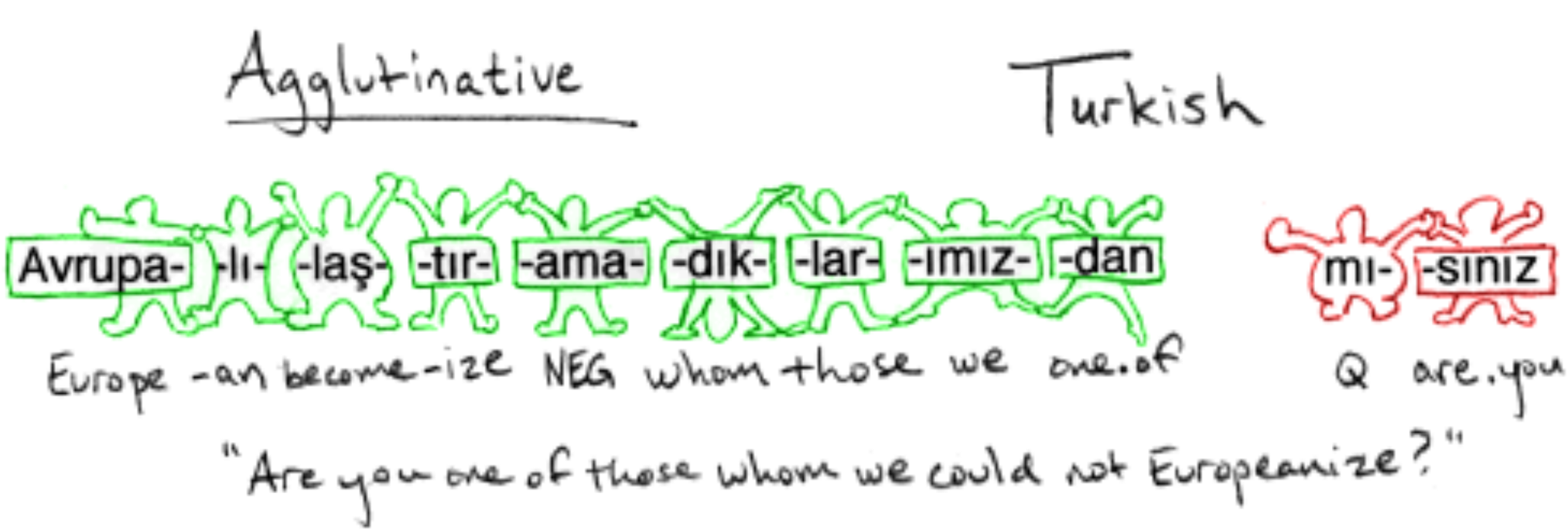
- WordNet's built-in lemmatized (dictionary-based)

```
>>> from nltk.stem import WordNetLemmatizer
>>> wordnet_lemmatizer = WordNetLemmatizer()
>>> wordnet_lemmatizer.lemmatize('leaves', pos='n')
'leaf'
>>> wordnet_lemmatizer.lemmatize('leaves', pos='v')
'leave'
```

# Stemming

- language dependent

.

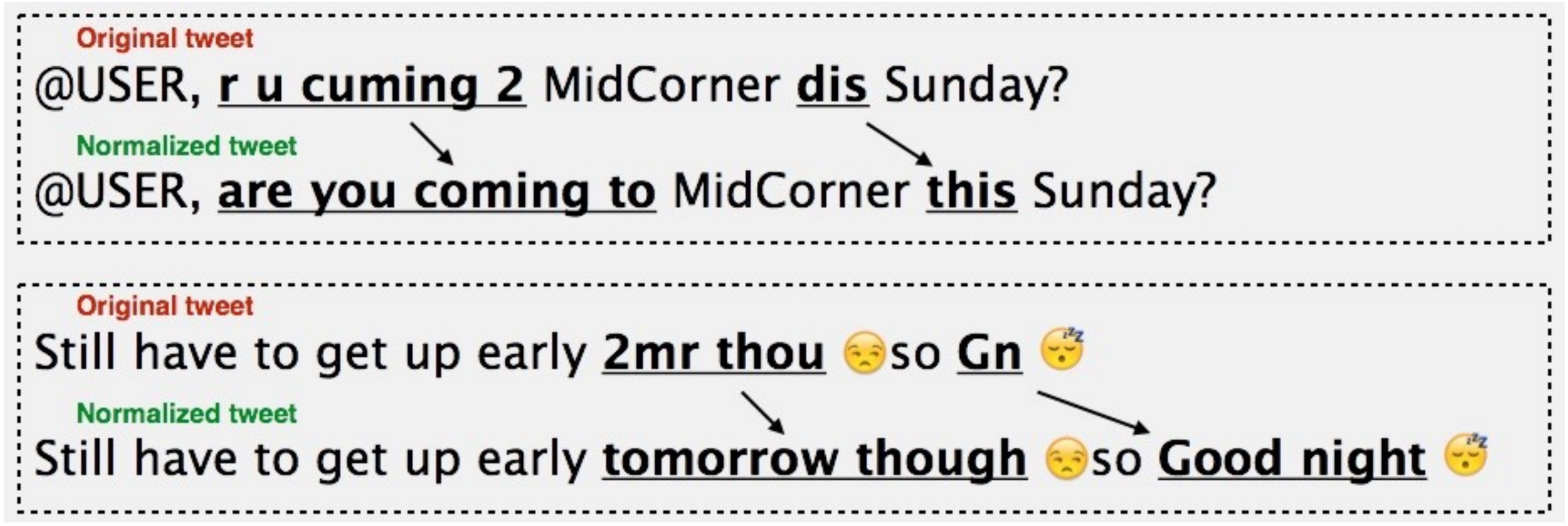





Source: All Things Linguistic

# Text Normalization

- convert non-standard words to standard




Source: Tim Baldwin, Marie de Marneffe, Han Bo, Young-Bum Kim, Alan Ritter, Wei Xu
Shared Tasks of the 2015 Workshop on Noisy User-generated Text:
Twitter Lexical Normalization and Named Entity Recognition

# Text Normalization

- types of non-standard words in 449 English tweets:

| Category | Ratio | Example |
|---|---|---|
| letter&numer | 2.36% | b4 → before |
| letter | 72.44% | shuld → should |
| number substitution | 2.76% | 4 → for |
| slang | 12.20 | lol → laugh out loud |
| other | 10.24% | sucha → such a |

most non-standard words are morphophonemic “errors”

Source: Bo Han and Timothy Baldwin
“Lexical normalisation of short text messages: Makn sens a #twitter” ACL 2011

# A Normalization Lexicon

- automatically derived from Twitter data + dictionary

```
41169    costumess   costumes
41170    nywhere anywhere
41171    sandwhich   sandwich
41172    aleksander  alexander
41173    juns    jun
41174    showi   showing
41175    washinq washing
41176    jscript script
41177    fundin  funding
41178    itxted  fitted
41179    cheeeap cheap
41180    fawesome    awesome
41181    untalented  talented
41182
```

Performance

Precision = 0.847

Recall = 0.630

F1-Score = 0.723

Source: Bo Han, Paul Cook and Timothy Baldwin
"Automatically Constructing a Normalisation Dictionary for Microblogs" EMNLP-CoNLL 2012

# Phrase-level Normalization

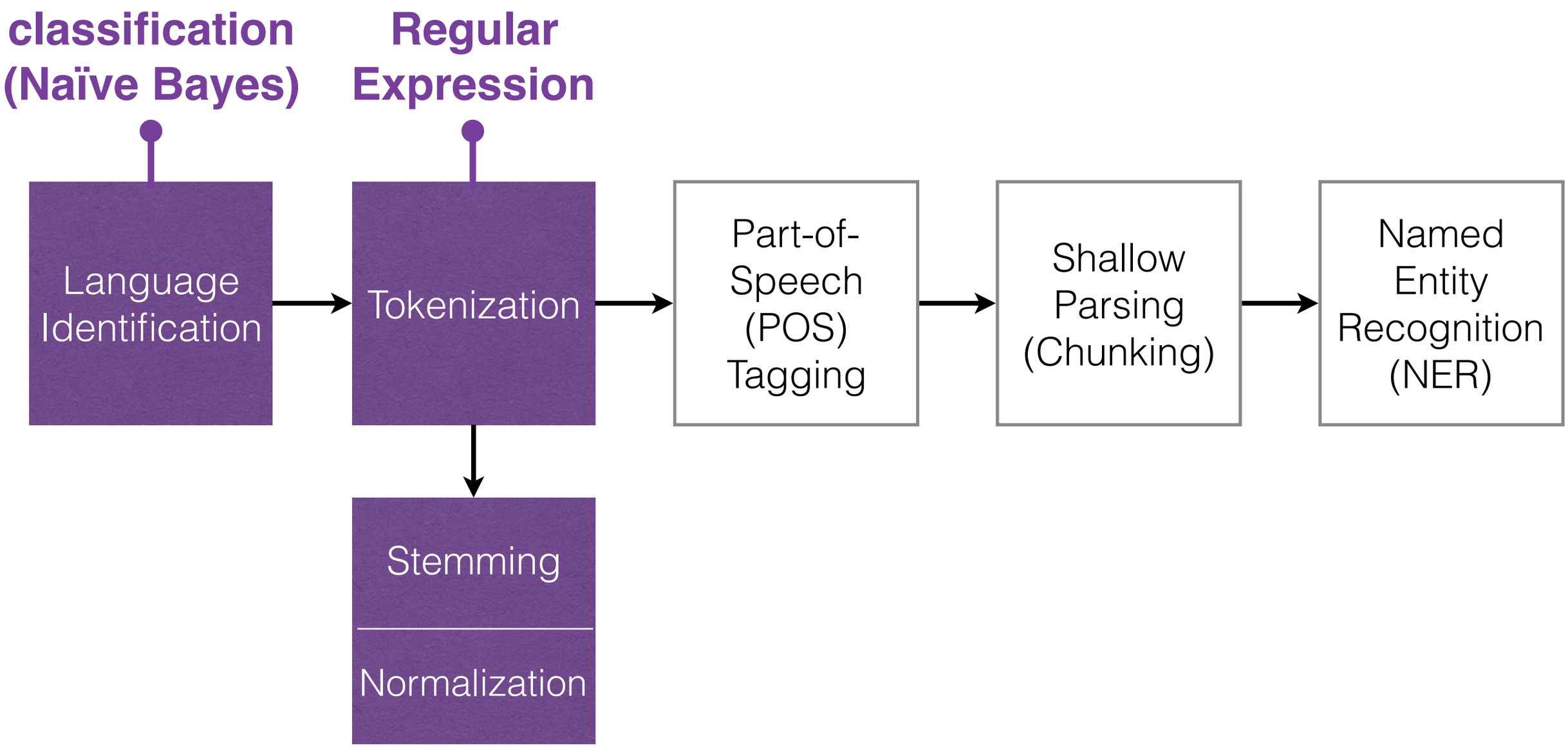
- word-level normalization is insufficient for many cases:

in-vocabulary words

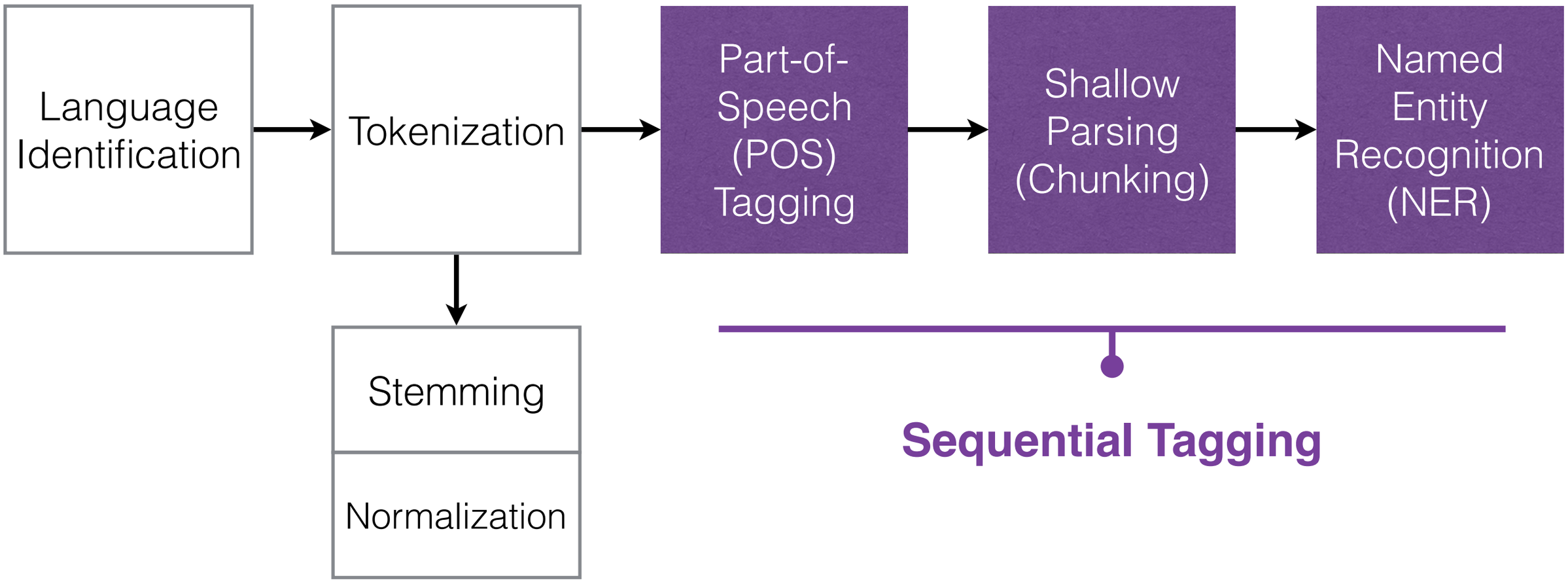
| Category | Example |
|---|---|
| 1-to-many | everytime → every time |
| incorrect IVs | can't want for → can't wait for |
| grammar | I'm going a movie → I'm going to a movie |
| ambiguities | 4 → 4 / 4th / for / four |

Source: Wei Xu, Alan Ritter, Ralph Grishman "Gathering and Generating Paraphrases from Twitter with Application to Normalization" BUCC 2013

# NLP Pipeline (summary so far)



# NLP Pipeline (next)



# Part-of-Speech (POS) Tagging

| | |
|---|---|
| Cant | MD |
| wait | VB |
| for | IN |
| the | DT |
| ravens | NNP |
| game | NN |
| tomorrow | NN |
| … | : |
| go | VB |
| ray | NNP |
| rice | NNP |
| !!!!!!! | . |

# Penn Treebank POS Tags

| | | | | | |
|---|---|---|---|---|---|
| 1. | CC | Coordinating conjunction | 25. | TO | *to* |
| 2. | CD | Cardinal number | 26. | UH | Interjection |
| 3. | DT | Determiner | 27. | VB | Verb, base form |
| 4. | EX | Existential *there* | 28. | VBD | Verb, past tense |
| 5. | FW | Foreign word | 29. | VBG | Verb, gerund/present participle |
| 6. | IN | Preposition/subordinating conjunction | 30. | VBN | Verb, past participle |
| 7. | JJ | Adjective | 31. | VBP | Verb, non-3rd ps. sing. present |
| 8. | JJR | Adjective, comparative | 32. | VBZ | Verb, 3rd ps. sing. present |
| 9. | JJS | Adjective, superlative | 33. | WDT | *wh*-determiner |
| 10. | LS | List item marker | 34. | WP | *wh*-pronoun |
| 11. | MD | Modal | 35. | WP$ | Possessive *wh*-pronoun |
| 12. | NN | Noun, singular or mass | 36. | WRB | *wh*-adverb |
| 13. | NNS | Noun, plural | 37. | # | Pound sign |
| 14. | NNP | Proper noun, singular | 38. | $ | Dollar sign |
| 15. | NNPS | Proper noun, plural | 39. | . | Sentence-final punctuation |
| 16. | PDT | Predeterminer | 40. | , | Comma |
| 17. | POS | Possessive ending | 41. | : | Colon, semi-colon |
| 18. | PRP | Personal pronoun | 42. | ( | Left bracket character |
| 19. | PP$ | Possessive pronoun | 43. | ) | Right bracket character |
| 20. | RB | Adverb | 44. | " | Straight double quote |
| 21. | RBR | Adverb, comparative | 45. | ‘ | Left open single quote |
| 22. | RBS | Adverb, superlative | 46. | “ | Left open double quote |
| 23. | RP | Particle | 47. | ’ | Right close single quote |
| 24. | SYM | Symbol (mathematical or scientific) | 48. | ” | Right close double quote |

# Part-of-Speech (POS) Tagging

- Words often have more than one POS:
  - The back door = JJ
  - On my back = NN
  - Win the voters back = RB
  - Promised to back the bill = VB

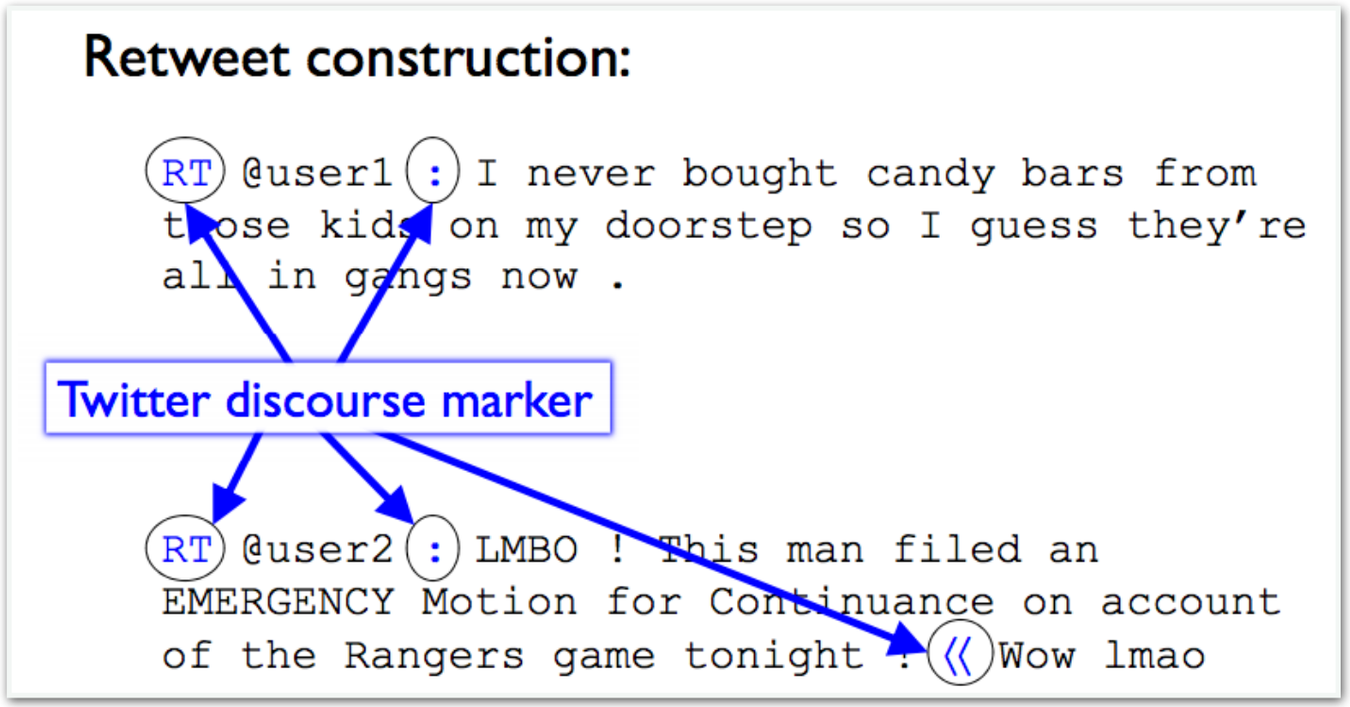
- POS tagging problem is to determine the POS tag for a particular instance of a word.



Source: adapted from Chris Manning

# Twitter-specific Tags

- #hashtag
- @metion
- url
- email address
- emoticon
- discourse marker
- symbols
- …




Source: Gimpel et al.
"Part-of-Speech Tagging for Twitter : Annotation, Features, and Experiments" ACL 2011

# Chunking

| | |
|---|---|
| Cant | VP |
| wait | |
| for | PP |
| the | |
| ravens | NP |
| game | |
| tomorrow | NP |
| … | |
| go | VP |
| ray | NP |
| rice | |
| !!!!!!! | |

# Chunking

- recovering phrases constructed by the part-of-speech tags

- a.k.a shallow (partial) parsing:

  - full parsing is expensive, and is not very robust

  - partial parsing can be much faster, more robust, yet sufficient for many applications

  - useful as input (features) for named entity recognition or full parser

# BIO tag encoding

| | | | |
|---|---|---|---|
| Cant | VP | | |
| wait | | | |
| for | PP | | |
| the | | | |
| ravens | NP | | |
| game | | | |
| tomorrow | NP | | |
| … | | | |
| go | VP | | |
| ray | NP | | |
| rice | | | |
| !!!!!!! | | | |

# BIO tag encoding

| | | |
|---|---|---|
| Cant | VP | VP |
| wait | | VP |
| for | PP | PP |
| the | | NP |
| ravens | NP | NP |
| game | | NP |
| tomorrow | NP | NP |
| … | | O |
| go | VP | VP |
| ray | NP | NP |
| rice | | NP |
| !!!!!!! | | O |

# BIO tag encoding

| | | | |
|---|---|---|---|
| Cant | VP | VP | B-VP |
| wait | | VP | I-VP |
| for | PP | PP | B-PP |
| the | | NP | B-NP |
| ravens | NP | NP | I-NP |
| game | | NP | I-NP |
| tomorrow | NP | NP | B-NP |
| … | | O | O |
| go | VP | VP | B-VP |
| ray | NP | NP | B-VP |
| rice | | NP | I-VP |
| !!!!!!! | | O | O |


I: Inside

O: outside

B: Begin

BIO allows separation of adjacent chunks/entities

# Named Entity Recognition(NER)

| | |
|---|---|
| Cant | |
| wait | |
| for | |
| the | |
| ravens | ORG |
| game | |
| tomorrow | |
| … | |
| go | |
| ray | PER |
| rice | |
| !!!!!!! | . |


ORG: organization

PER: person

LOC: location

# NER: Basic Classes

| | |
|---|---|
| Cant | |
| wait | |
| for | |
| the | |
| ravens | ORG |
| game | |
| tomorrow | |
| … | |
| go | |
| ray | PER |
| rice | |
| !!!!!!! | . |


ORG: organization

PER: person

LOC: location

# Noisy Text: NLP breaks

**POS:**

NNP/ Yess ./ ! NNP/ Yess ./ ! PRP$/ Its JJ/ official NNP/ Nintendo VBD/ announced NN/ today IN/ that PRP/ they MD/ Will VB/ release DT/ the NNP/ Nintendo NN/ 3DS IN/ in RB/ north NNP/ America NN/ march CD/ 27 IN/ for NN/ $250

**Chunk:**

[NP Yess] ! [NP Yess] ! [NP Its offcial Nintendo] [VP announced]
[NP today] [SBAR that] [NP they] [VP Will release] [NP the Nintendo 3DS]
[PP in] [NP north America march] [NP 27] [PP for] [NP $250]

**NER:**

[**ORG Yess**] ! [**ORG Yess**] ! Its official [**ORG Nintendo**] announced today that they Will release the [**ORG Nintendo**] 3DS in north [**LOC America**] march 27 for $250

**Noisy Style**

# Noisy Text: NLP breaks

**Sohaib Athar**
@ReallyVirtual

**LOCATION**

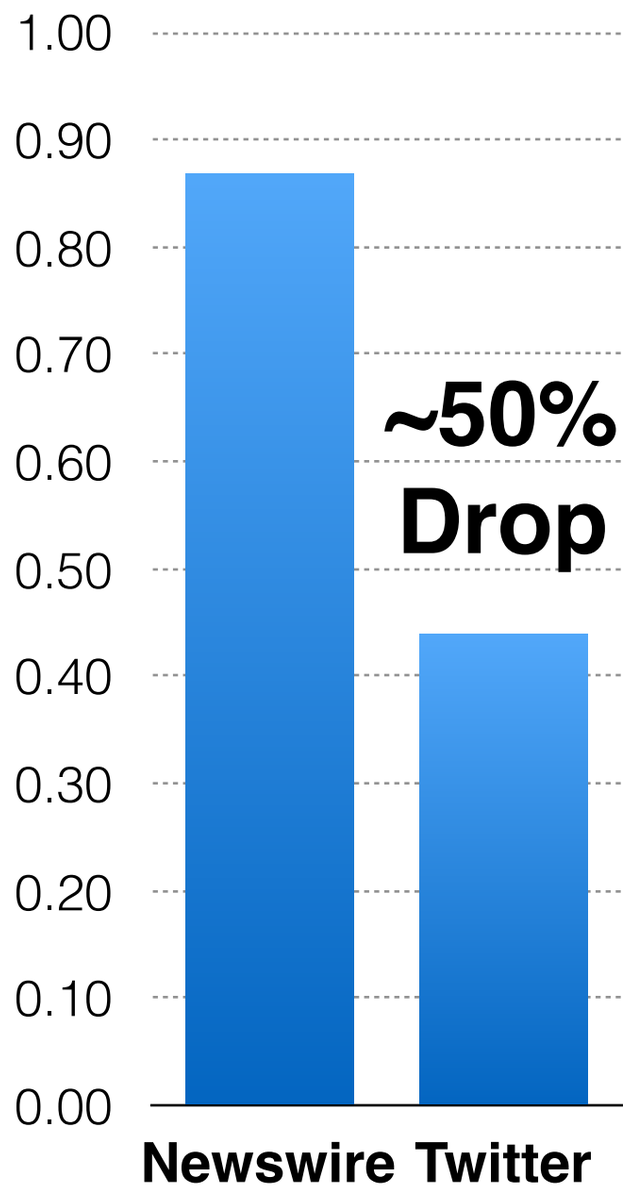
Helicopter hovering above Abbottabad at 1AM (is a rare event).

**Keith Urbahn**
@keithurbahn

**PERSON**

So I'm to table person they have killed Osama Bin Laden. Hot damn.

**Stanford NER:**



# Evaluation Metric

| | | Predicted: Negative | Predicted: Positive |
|---|---|---|---|
| Actual | Negative | True Negative | False Positive |
| | Positive | False Negative | True Positive |

True Positive + False Positive = Total Predicted Positive

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

$$= \frac{True\ Positive}{Total\ Predicted\ Positive}$$

# Evaluation Metric

$$\text{Recall} = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

$$= \frac{True\ Positive}{Total\ Actual\ Positive}$$

| | | Predicted | |
|---|---|---|---|
| | | **Negative** | **Positive** |
| **Actual** | **Negative** | True Negative | False Positive |
| | **Positive** | False Negative | True Positive |

True Positive + False Negative = Actual Positive

# Evaluation Metric

$$F1 = 2 \times \frac{Precision * Recall}{Precision + Recall}$$

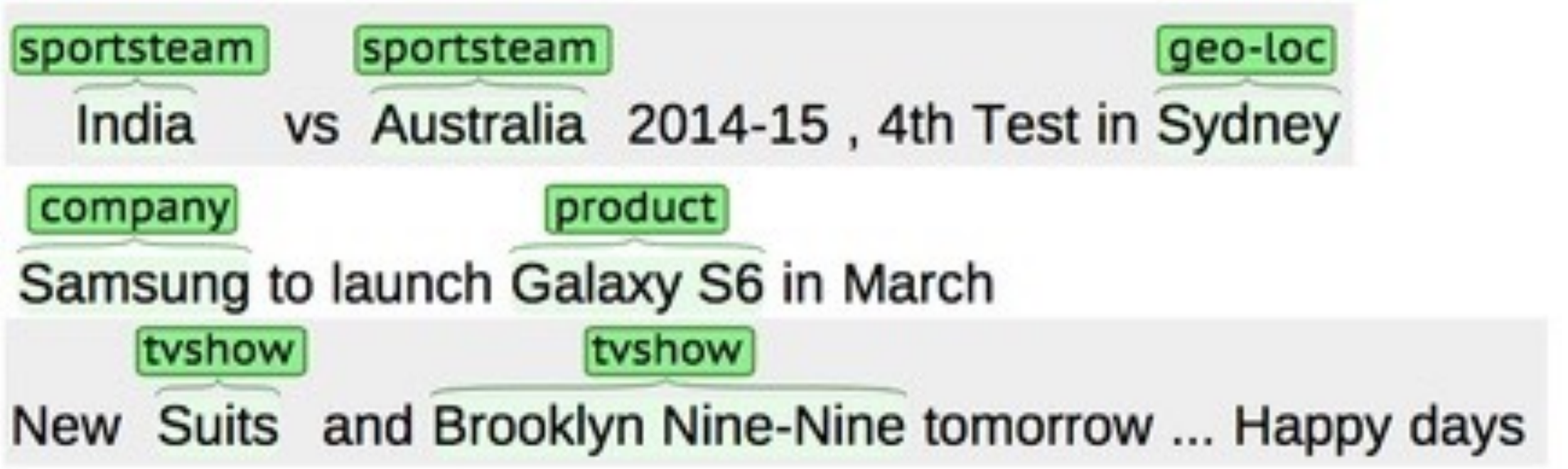
| | |
|---|---|
| Foreign | ORG |
| Ministry | ORG |
| spokesman | O |
| Shen | PER |
| Guofang | PER |
| told | O |
| Reuters | ORG |
| : | : |

} Standard evaluation is per entity, *not* per token

# Noisy Text: Challenges

- Lexical Variation (misspellings, abbreviations)
  - `2m', `2ma', `2mar', `2mara', `2maro', `2marrow', `2mor', `2mora', `2moro', `2morow', `2morr', `2morro', `2morrow', `2moz', `2mr', `2mro', `2mrrw', `2mrw', `2mw', `tmmrw', `tmo', `tmoro', `tmorrow', `tmoz', `tmr', `tmro', `tmrow', `tmrrow', `tmrrw', `tmrw', `tmrww', `tmw', `tomaro', `tomarow', `tomarro', `tomarrow', `tomm', `tommarow', `tommarrow', `tommoro', `tommorow', `tommorrow', `tommorw', `tommrow', `tomo', `tomolo', `tomoro', `tomorow', `tomorro', `tomorrw', `tomoz', `tomrw', `tomz‘
- Unreliable Capitalization
  - “The Hobbit has FINALLY started filming! I cannot wait!”
- Unique Grammar
  - “watchng american dad.”

# NER: Rich Classes




Source: Strauss, Toma, Ritter, de Marneffe, Xu
Results of the WNUT16 Named Entity Recognition Shared Task (WNUT@COLING 2016)

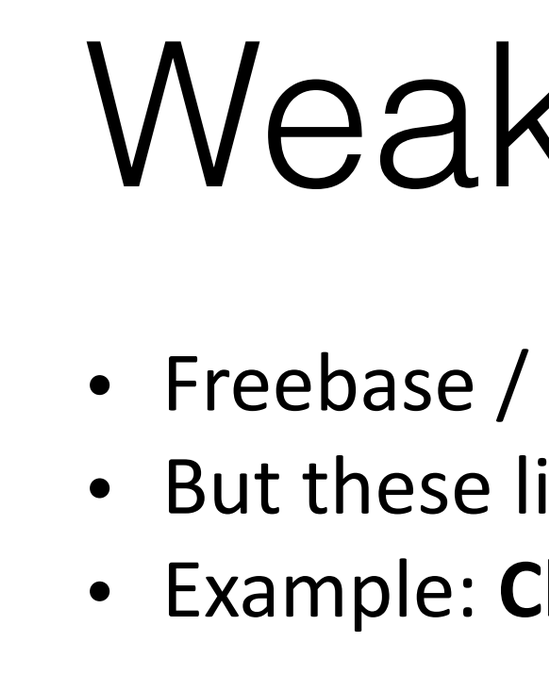
# NER: Genre Differences

| | News | Tweets |
|---|---|---|
| PER | Politicians, business leaders, journalists, celebrities | Sportsmen, actors, TV personalities, celebrities, names of friends |
| LOC | Countries, cities, rivers, and other places related to current affairs | Restaurants, bars, local landmarks/areas, cities, rarely countries |
| ORG | Public and private companies, government organisations | Bands, internet companies, sports clubs |

Source: Kalina Bontcheva and Leon Derczynski
"Tutorial on Natural Language Processing for Social Media" EACL 2014

# Weakly Supervised NER

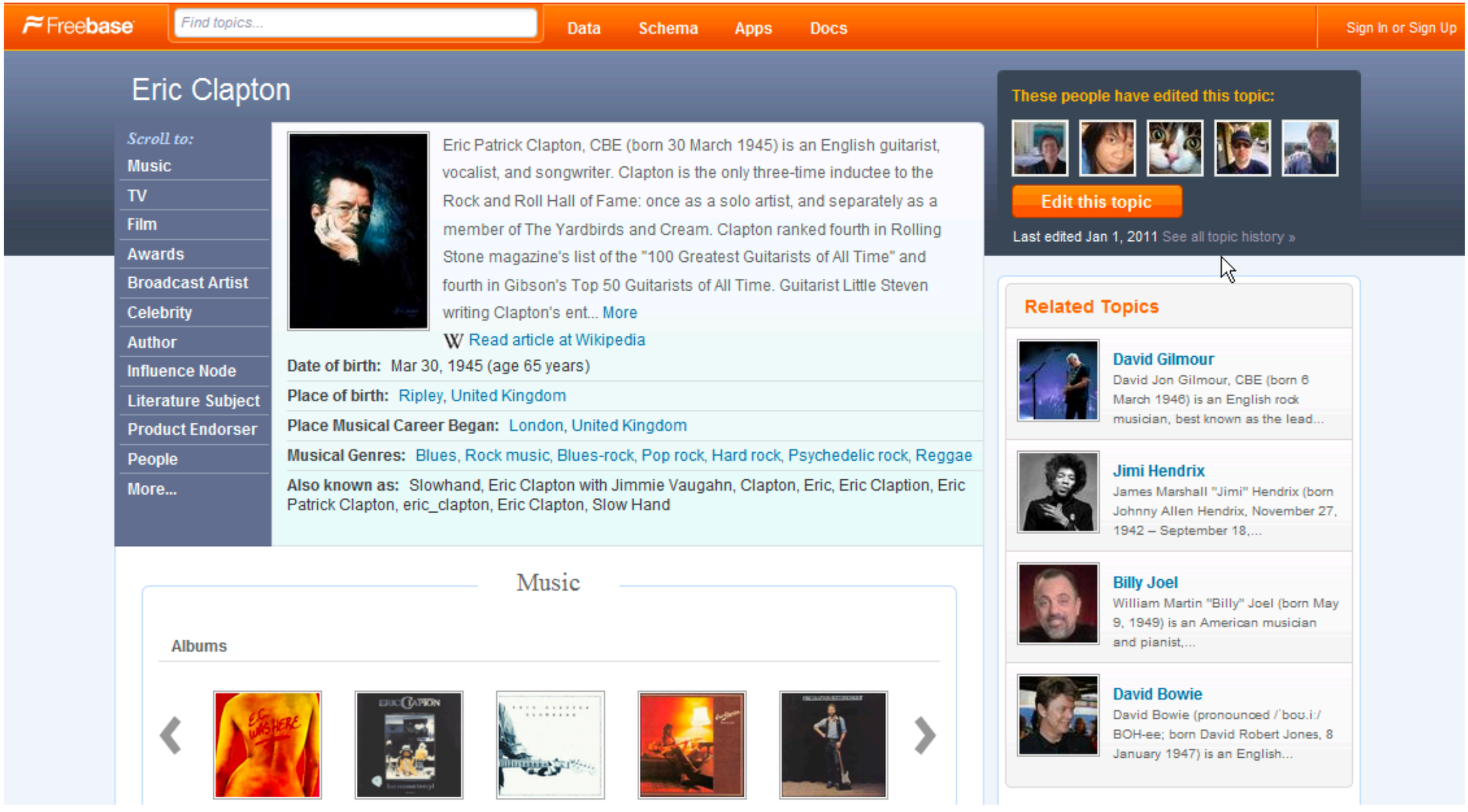
- Freebase / Wikipedia lists provide a source of supervision
- But these lists are highly ambiguous
- Example: **China**






...

# Freebase


Freebase
Find topics...
Data
Schema
Apps
Docs
Sign In or Sign Up
Eric Clapton
Scroll to:
Music
TV
Film
Awards
Broadcast Artist
Celebrity
Author
Influence Node
Literature Subject
Product Endorser
People
More...
Eric Patrick Clapton, CBE (born 30 March 1945) is an English guitarist, vocalist, and songwriter. Clapton is the only three-time inductee to the Rock and Roll Hall of Fame: once as a solo artist, and separately as a member of The Yardbirds and Cream. Clapton ranked fourth in Rolling Stone magazine's list of the "100 Greatest Guitarists of All Time" and fourth in Gibson's Top 50 Guitarists of All Time. Guitarist Little Steven writing Clapton's ent... More
Read article at Wikipedia
Date of birth: Mar 30, 1945 (age 65 years)
Place of birth: Ripley, United Kingdom
Place Musical Career Began: London, United Kingdom
Musical Genres: Blues, Rock music, Blues-rock, Pop rock, Hard rock, Psychedelic rock, Reggae
Also known as: Slowhand, Eric Clapton with Jimmie Vaughan, Clapton, Eric, Eric Claption, Eric Patrick Clapton, eric_clapton, Eric Clapton, Slow Hand
These people have edited this topic:
Edit this topic
Last edited Jan 1, 2011 See all topic history »
Related Topics
David Gilmour
David Jon Gilmour, CBE (born 6 March 1946) is an English rock musician, best known as the lead...
Jimi Hendrix
James Marshall "Jimi" Hendrix (born Johnny Allen Hendrix, November 27, 1942 – September 18,...
Billy Joel
William Martin "Billy" Joel (born May 9, 1949) is an American musician and pianist,...
David Bowie
David Bowie (pronounced /ˈbuːi/ BOH-ee; born David Robert Jones, 8 January 1947) is an English...
Music
Albums

# Wikidata

Metaweb Acquired by Google - Inc.

https://www.inc.com › news › articles › 2010/07 › google-acquires-metaweb

Aug 19, 2019 - The five-year-old San Francisco startup maintains **Freebase**, an open ... (Curiously, **Google** made the official **acquisition** announcement right ...

## From Freebase to Wikidata: The Great Migration

Thomas Pellissier Tanon*
Google, San Francisco, USA
thomas@pellissier-tanon.fr

Denny Vrandečić
Google, San Francisco, USA
vrandecic@google.com

Sebastian Schaffert
Google, Zürich, Switzerland
schaffert@google.com

Thomas Steiner
Google, Hamburg, Germany
tomac@google.com

Lydia Pintscher
Wikimedia, Berlin, Germany
lydia@pintscher.de

### ABSTRACT

Collaborative knowledge bases that make their data freely available in a machine-readable form are central for the data strategy of many projects and organizations. The two major collaborative knowledge bases are Wikimedia's Wikidata and Google's Freebase. Due to the success of Wikidata, Google decided in 2014 to offer the content of Freebase to the Wikidata community. In this paper, we report on the ongoing transfer efforts and data mapping challenges, and provide an analysis of the effort so far. We describe the *Primary Sources Tool*, which aims to facilitate this and future data migrations. Throughout the migration, we have gained deep insights into both Wikidata and Freebase, and share and discuss detailed statistics on both knowledge bases.

### General Terms

Human Factors, Documentation

One such collaborative knowledge base is Freebase, publicly launched by Metaweb in 2007 and acquired by Google in 2010. Another example is Wikidata, a collaborative knowledge base developed by Wikimedia Deutschland since 2012 and operated by the Wikimedia Foundation. Due to the success of Wikidata, Google announced in 2014 their intent to shut down Freebase and help the community with the transfer of Freebase content to Wikidata [10].

Moving data between two knowledge bases that do not share a similar design is usually a problematic task and requires the careful mapping between their structures. The migration from Freebase to Wikidata was no exception to this rule: we encountered a number of to-be-expected *structural* challenges. However, even more demanding was the *cultural* difference between the two involved communities. The Freebase and Wikidata communities have a very different background, subtly different goals and understandings of their tasks, and different requirements regarding their data.

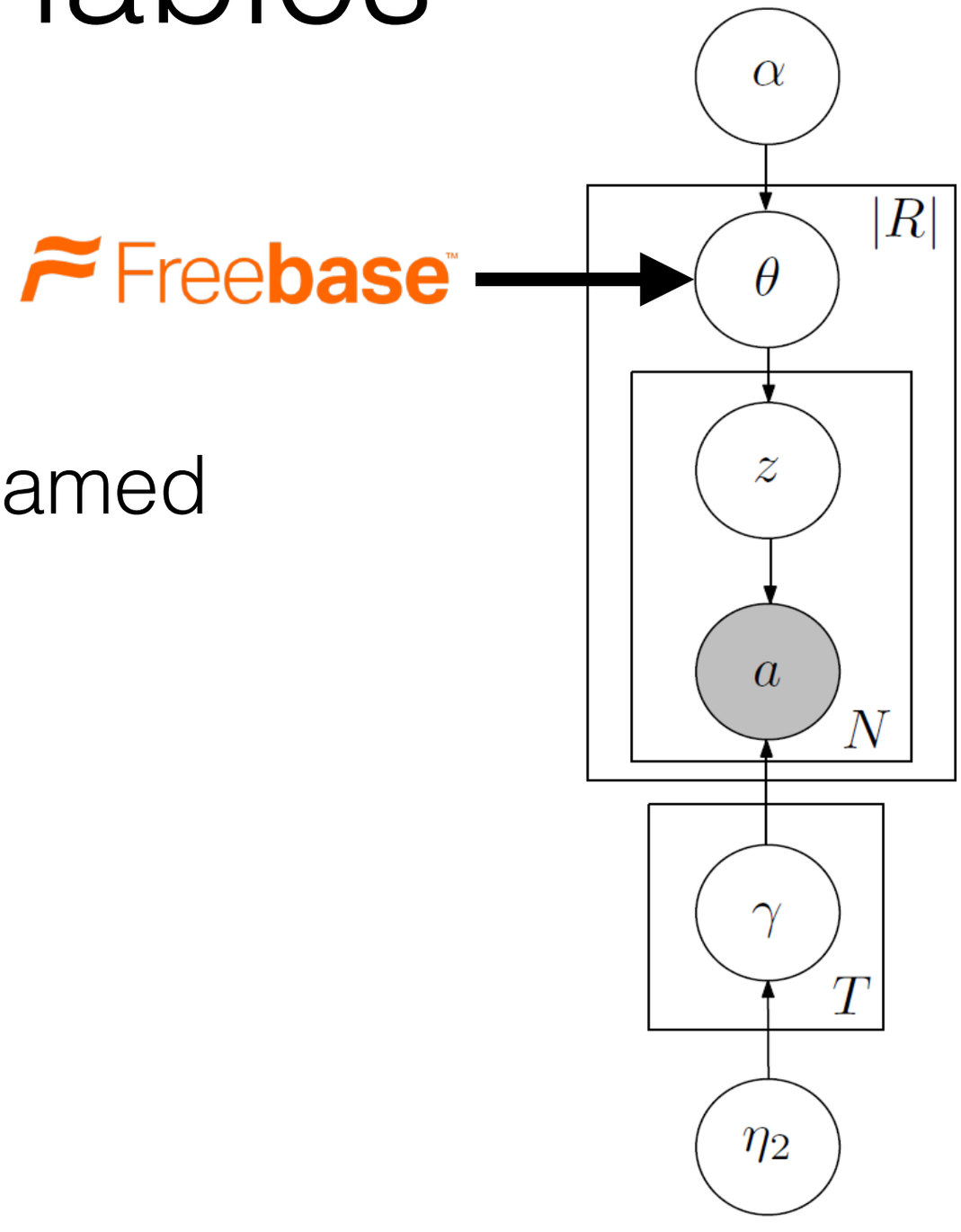
In this paper, we describe how we support the Wikidata

(WWW 2016)

[Ritter, et. al. EMNLP 2011]

# Distant Supervision with Latent Variables

Latent variable model for Named Entity Categorization **with constraints**



[Ritter, et. al. EMNLP 2011]

Freebase™

Obama

Apple

**JFK**

On my way to **JFK** early in the…

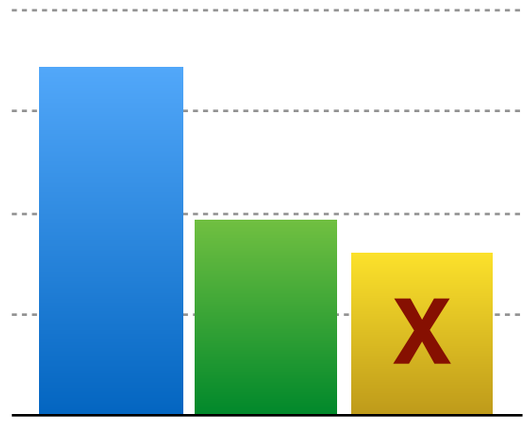
**JFK** 's bomber jacket sells for…

**JFK** Airport's Pan Am Worldport…

Waiting at **JFK** for our ride…

When **JFK** threw first pitch on…

⋮ ⋮




| | |
|---|---|
| 's | 0.04 |
| threw | 0.02 |
| jacket | 0.01 |
| … | |

**PERSON**

| | |
|---|---|
| waiting | 0.04 |
| ride | 0.03 |
| way | 0.02 |
| … | |

**FACILITY**

| | |
|---|---|
| announced | 0.04 |
| new | 0.03 |
| release | 0.02 |
| … | |

**PRODUCT**

[Ritter, et. al. EMNLP 2011]

Freebase™

Obama

Apple

JFK

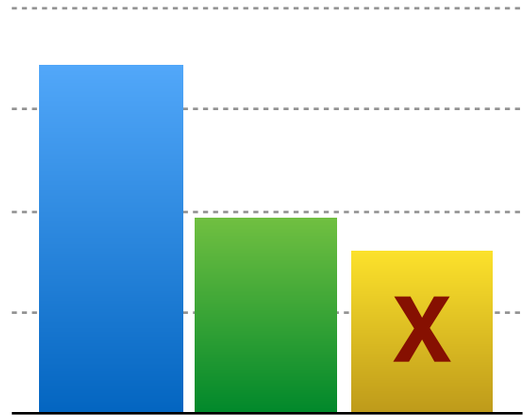
On my way to JFK early in the…

JFK 's bomber jacket sells for

JFK Airport's Pan Am Worldport…

Waiting at JFK for our ride…

When JFK threw first pitch on…




| | |
|---|---|
| 's | 0.04 |
| threw | 0.02 |
| jacket | 0.01 |
| … | |

PERSON

| | |
|---|---|
| waiting | 0.04 |
| ride | 0.03 |
| way | 0.02 |
| … | |

FACILITY

| | |
|---|---|
| announced | 0.04 |
| new | 0.03 |
| release | 0.02 |
| … | |

PRODUCT

# Data/Inference

- Gather entities and words which co-occur
  - Extract Entities from about 60M status messages
- Used a set of 10 types from Freebase
  - Commonly occur in Tweets
  - Good coverage in Freebase
- Inference: Collapsed Gibbs sampling:
  - Constrain types using Freebase
  - For entities not in Freebase, don't constrain

[Ritter, et. al. EMNLP 2011]

# Example Type Lists

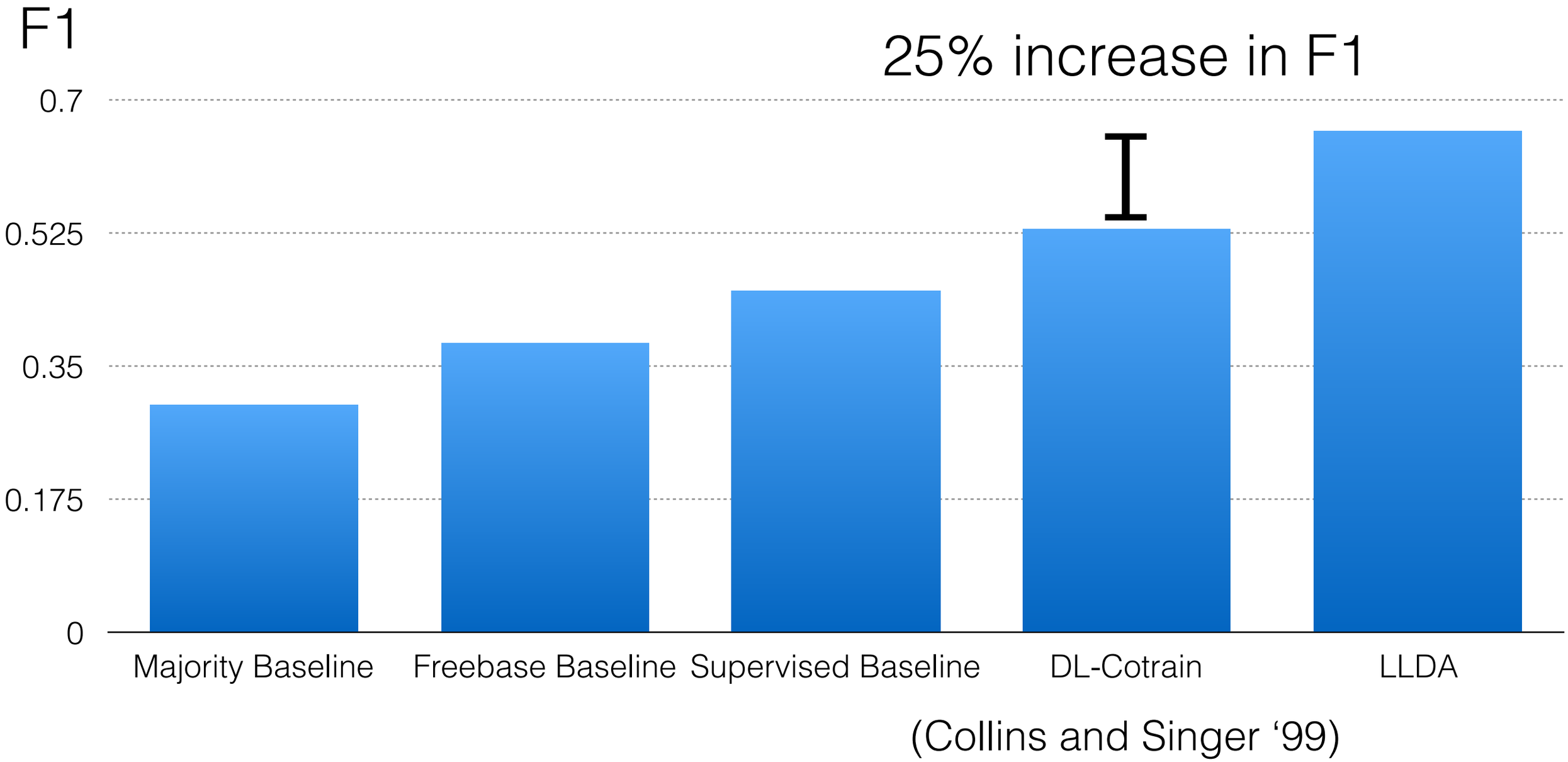
| Type | Top 20 Entities not found in Freebase dictionaries |
|---|---|
| *PRODUCT* | nintendo ds, n black, ipod nano, apple iphone, verizon media, mac app store, kde, phone/ipod, galaxy tab, samsung gal, nintendo ds, vpn |
| *TV-SHOW* | pretty little, svu, greys, kktny, rhobh, parks & recr, n 's creek, big fat gypsy weddings, bi, r wipeout, jersey shores, idiot abroad, royle, jerseyshore, mr ., sunshine, hawaii five-0, new jersey shore |
| *FACILITY* | voodoo loun, mansion, sullivan hall, memorial un, d music hall, amway center, el moca, gestone arena, cat club, le poisson ro, y bay, broadway bar, ritz carlton, mgm grand, olympia theatre, consol energy center |



KKTNY = Kourtney and Kim Take New York
RHOBH = Real Housewives of Beverly Hills

[Ritter, et. al. EMNLP 2011]

# Twitter NER: Classification Results




(Collins and Singer '99)

# Tool: twitter_nlp

https://github.com/aritter/twitter_nlp

This repository Search Pull requests Issues Gist

aritter / twitter_nlp Watch 71

Twitter NLP Tools

55 commits 2 branches 0 releases 1 contributor

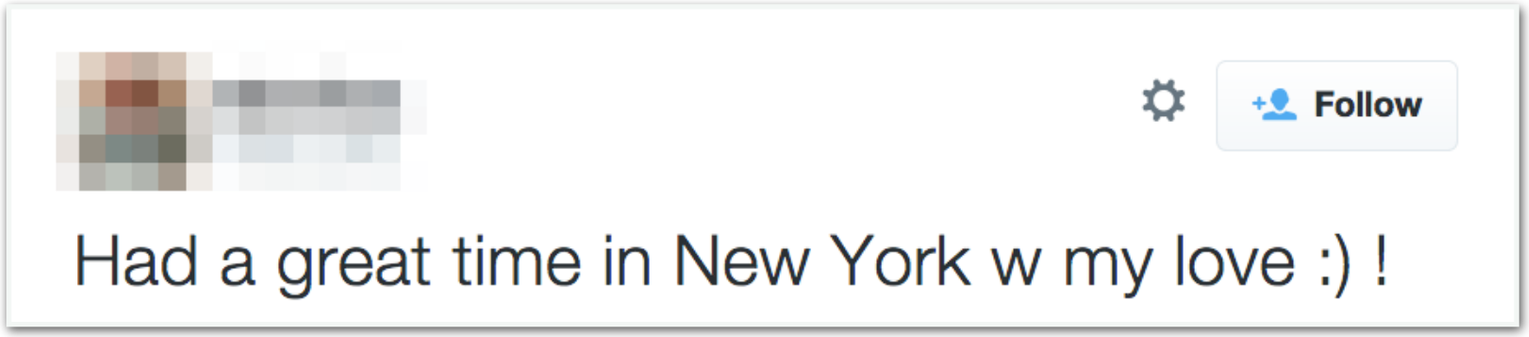
branch: master twitter_nlp / +

a few corrections to the NER annotation from Brendan 1 comment

aritter authored on Nov 8, 2014 latest commit 27c8190084

| | | |
|---|---|---|
| data | a few corrections to the NER annotation from Brendan | 8 months ago |
| hbc | added labels for weakly supervised NE categorization | 2 years ago |
| lib | added README.md | 3 years ago |
| mallet-2.0.6 | re-importing to blow away some large files in the history | 4 years ago |
| models | Fixed a bug in computing brown clusters reported by Yiye Ruan and Lu … | a year ago |

# Tool: twitter_nlp




```
xuwei@proteus100[twitter_nlp]$ export TWITTER_NLP=./
xuwei@proteus100[twitter_nlp]$
xuwei@proteus100[twitter_nlp]$ echo "Had a great time in New York w my
love :) ! " | python python/ner/extractEntities2.py

Had/O a/O great/O time/O in/O New/B-ENTITY York/I-ENTITY w/O my/O love/
O :)/O !/O
Average time per tweet = 3.04769945145s
xuwei@proteus100[twitter_nlp]$
xuwei@proteus100[twitter_nlp]$ echo "Had a great time in New York w my
love :) ! " | python python/ner/extractEntities2.py --pos --chunk

Had/O/VBD/B-VP a/O/DT/B-NP great/O/JJ/I-NP time/O/NN/I-NP in/O/IN/B-PP
New/B-ENTITY/NNP/B-NP York/I-ENTITY/NNP/I-NP w/O/IN/B-PP my/O/PRP$/B-NP
 love/O/NN/I-NP :)/O/UH/B-INTJ !/O/./I-INTJ
Average time per tweet = 5.49846148491s
xuwei@proteus100[twitter_nlp]$
```

# Results of the WNUT16 Named Entity Recognition Shared Task

Benjamin Strauss, Bethany Toma, Alan Ritter, Marie-Catherine de Marneffe and Wei Xu

http://noisy-text.github.io/

# Need for Shared Evaluations

- **Fast Moving Area:** Papers published in the same year use different datasets and evaluation methodology
- Performance still behind what we would like
  - ~0.6 - 0.7 F1 score (**much lower than news**)
  - Explore new ideas & approaches

# Related NER Evaluations

- MUC Newswire
  - http://www.itl.nist.gov/iaui/894.02/related_projects/muc/muc_data/muc_data_index.html
- CONLL Newswire
  - http://www.cnts.ua.ac.be/conll2002/ner/
  - http://www.cnts.ua.ac.be/conll2003/ner/
- ACE Newswire
  - https://catalog.ldc.upenn.edu/LDC2005T09
- Named Entity rEcognition and Linking (NEEL) Challenge Microblogs
  - #Microposts workshop at WWW
  - http://microposts2016.seas.upenn.edu/challenge.html

# Data

**Training + Dev Data:**

- All training, dev, test data from 2015
- **Training:** 2,394 tweets, **Dev:** 1,420 tweets

**Test Data**

- 3,856 tweets
- Later Time Period
    - No overlap in time period with Training/Dev data

# 2016 Test Data Annotation

- Simple Annotation Guidelines:
  - http://bit.ly/1FSP6i2
- Re-annotated 100 tweets from 2015 data
  - Good agreement - 0.68 F-Score
- Frequent questions, discussion among the group
- BRAT annotation tool
  - (http://brat.nlplab.org/)

# Annotation Interface

1 company Zendesk Security Breach Affects company Twitter , company Tumblr and company Pinterest : http://bit.ly/12UtbI8

2 company Snapchat Responds to other New Year 's Eve Security Breach http://bit.ly/1kd1mUK | #TECH - @HCP520

3 tvshow Today Show @Snapchat CEO talks about breach on . #smsportschat #snapchat http://www.nbcnews.com/id/21134540/vp=53971379&#53971379 ...

4 company #Twitter , company #Pinterest and company #Tumblr company Notify of Security Breach After #Zendesk #Hack http://goo.gl/H8sGj

5 other White House Hiding other Pentagon Report On geo-loc Russia's Breach Of Nuclear Treaty http://ln.is/dailycaller.com/2015/oMO52 ... via @dailycaller

6 company ICANN resets passwords after website breach http://dlvr.it/BlyZSX

7 company St . Joseph Health notifies 33,000 of potential data breach http://dlvr.it/5zPJ8C - #Imaging

8 company TalkTalk data breach hit 155,000 customers #TCSITWiz

9 company Final TalkTalk breach tally : 4% of customers affected : company TalkTalk continues with its practice ... http://bit.ly/1HpzbhD #infosec #security

10 company #Snapchat Breach Exposes Weak Security http://nyti.ms/Knelmn | Photo by person J . Emilio Florespic .twitter.com/8GTCH4VAsY

# 10 Participating Teams

| Team ID | Affiliation |
|---|---|
| CambridgeLTL | University of Cambridge |
| Talos | Viseo R&D |
| akora | University of Manchester |
| NTNU | Indian Institute of Technology Patna |
| ASU | Ain Shams University, Cairo, Egypt |
| DeepNNNER | Honda Research Institute Japan |
| DeepER | University of Illinois at Urbana-Champaign |
| hjpwhu | Wuhan University |
| UQAM-NTL | Université du Québec à Montréal |
| LIOX | The Hong Kong Polytechnic University |

# Approaches

| | POS | Orthographic | Gazetteers | Brown clustering | Word embedding | ML |
|---|---|---|---|---|---|---|
| BASELINE | – | ✓ | ✓ | – | – | CRFsuite |
| CambridgeLTL | – | ✓ | – | – | – | LSTM |
| akora | – | – | – | – | – | LSTM |
| NTNU | ✓ | ✓ | ✓ | – | – | CRF |
| Talos | ✓ | ✓ | ✓ | ✓ | GloVe | L2S |
| DeepNNNER | – | – | – | – | Multiple | LSTM-CNN |
| ASU | – | – | ✓ | ✓ | – | LSTM |
| UQAM-NTL | ✓ | ✓ | ✓ | – | – | CRF |

- All teams use some form of machine supervised ML
- Many LSTM-based approaches as compared to previous year
- **Unique approaches:** CambridgeLTL, Talos

# Results (10 types)

| | Precision | Recall | F1 |
|---|---|---|---|
| CambridgeLTL | 60.77 | 46.07 | 52.41 |
| Talos | 58.51 | 38.12 | 46.16 |
| akora | 51.70 | 39.48 | 44.77 |
| NTNU | 53.19 | 32.13 | 40.06 |
| ASU | 40.58 | 37.58 | 39.02 |
| DeepNNNER | 54.97 | 28.16 | 37.24 |
| DeepER | 45.40 | 31.15 | 36.95 |
| hjpwhu | 48.90 | 28.76 | 36.22 |
| UQAM-NTL | 40.73 | 23.52 | 29.82 |
| LIOX | 40.15 | 12.69 | 19.26 |

# Results (No Types)

| | Precision | Recall | F1 |
|---|---|---|---|
| CambridgeLTL | 73.49 | 59.72 | 65.89 |
| NTNU | 64.18 | 62.28 | 63.22 |
| Talos | 70.53 | 52.58 | 60.24 |
| akora | 64.75 | 54.28 | 59.05 |
| ASU | 57.55 | 52.98 | 55.17 |
| DeepER | 63.17 | 43.31 | 51.38 |
| DeepNNNER | 70.66 | 36.14 | 47.82 |
| hjpwhu | 63.00 | 37.06 | 46.66 |
| UQAM-NTL | 53.21 | 37.95 | 44.30 |
| LIOX | 58.18 | 31.33 | 40.73 |

# Domain-Specific Data

Cybersecurity (350 Tweets)

HACKMAGEDDON

Information Security Timelines and Statistics

Gun Violence (500 Tweets)

GUN VIOLENCE Archive

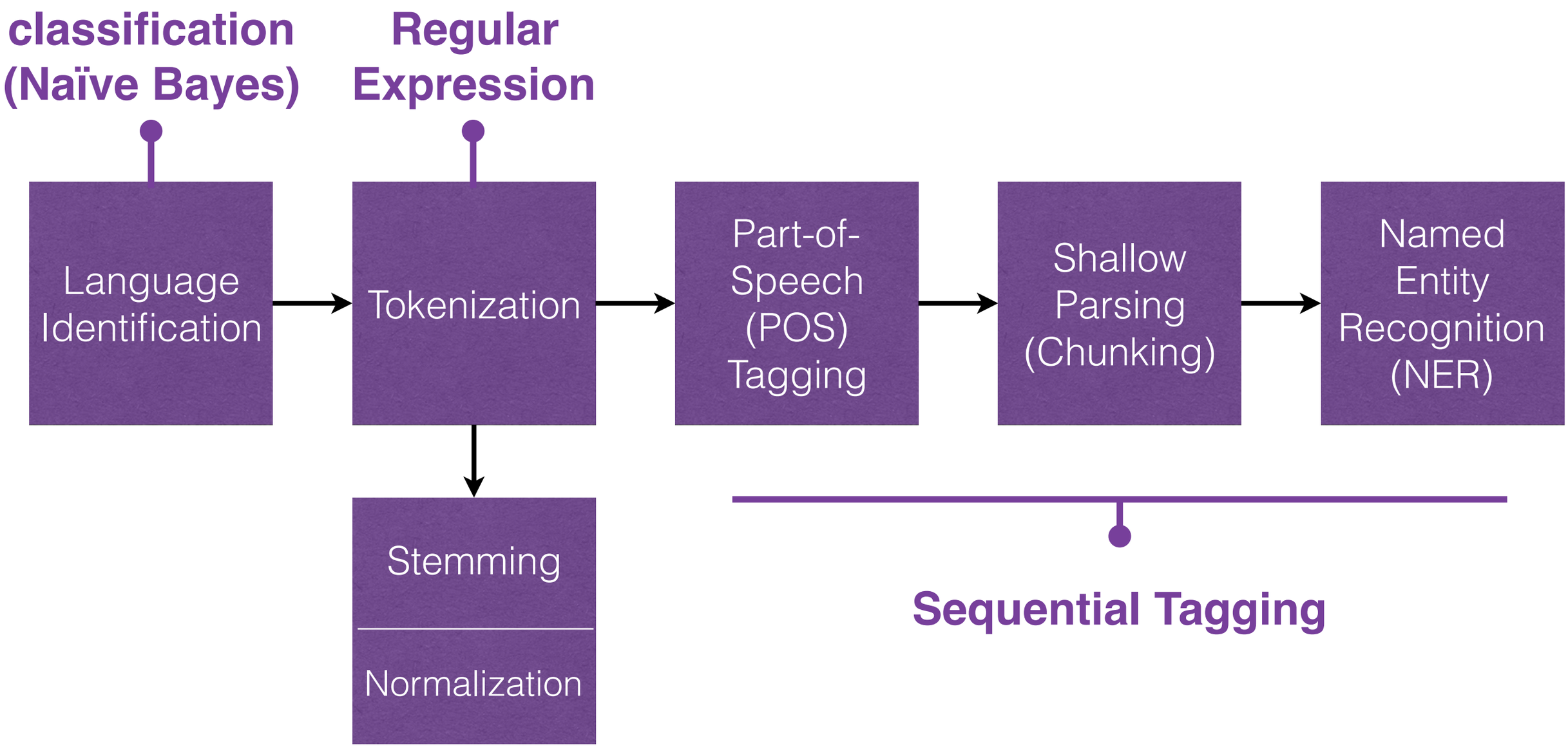
# Results (Cyber-Domain)

| | Acc | P | R | F1 |
|---|---|---|---|---|
| CambridgeLTL | 90.57 | 69.75 | 51.24 | 59.08 |
| Talos | 89.36 | 60.49 | 41.13 | 48.96 |
| akora | 88.42 | 54.21 | 36.32 | 43.50 |
| hjpwhu | 88.21 | 59.79 | 28.86 | 38.93 |
| ASU | 87.76 | 42.22 | 32.84 | 36.94 |
| NTNU | 87.72 | 51.89 | 27.36 | 35.83 |
| DeepNNNER | 87.66 | 62.88 | 23.88 | 34.62 |
| DeepER | 84.32 | 40.23 | 22.89 | 29.18 |
| UQAM-NTL | 85.64 | 37.97 | 16.75 | 23.25 |
| LIOX | 84.41 | 30.08 | 6.63 | 10.87 |

# Results (Shooting-Domain)

| | Acc | P | R | F1 |
|---|---|---|---|---|
| CambridgeLTL | 93.00 | 66.25 | 56.72 | 61.12 |
| Talos | 92.03 | 68.53 | 49.00 | 57.14 |
| DeepER | 91.96 | 64.01 | 51.40 | 57.02 |
| akora | 91.54 | 58.89 | 49.40 | 53.73 |
| NTNU | 91.14 | 61.36 | 42.08 | 49.92 |
| DeepNNNER | 91.22 | 59.88 | 41.15 | 48.78 |
| hjpwhu | 90.83 | 53.71 | 41.41 | 46.77 |
| ASU | 90.74 | 45.40 | 47.94 | 46.63 |
| UQAM-NTL | 89.38 | 45.80 | 33.42 | 38.65 |
| LIOX | 88.35 | 55.77 | 23.17 | 32.74 |

# Summary


classification (Naïve Bayes)
Regular Expression
Language Identification
Tokenization
Part-of-Speech (POS) Tagging
Shallow Parsing (Chunking)
Named Entity Recognition (NER)
Stemming
Normalization
Sequential Tagging