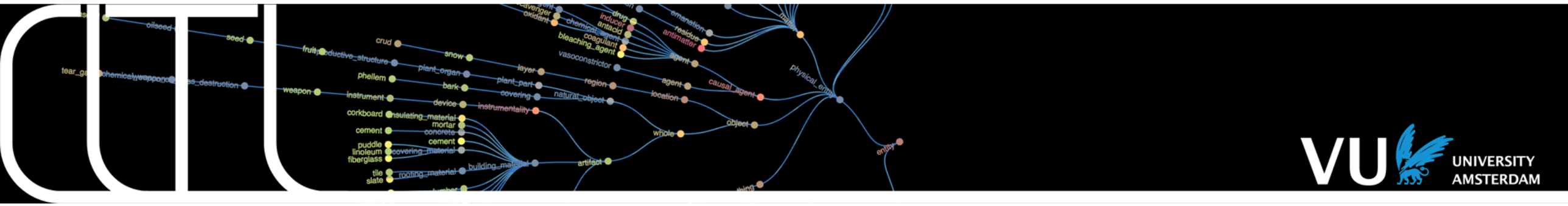
Text Mining CBS

Lecture 4: Named entity detection and classification

Piek Vossen



What is Named Entity Recognition?

- Named Entity Recognition is the task of finding and classifying names in text.

Abraham Lincoln Listeni/ˈeɪbrəhæm ˈlɪŋkən/ (February 12, 1809 – April 15, 1865) was the 16th President of the United States, serving from March 1861 until his assassination in April 1865. Lincoln led the United States through its Civil War—its bloodiest war and its greatest moral, constitutional and political crisis.[1][2] In doing so, he preserved the Union, abolished slavery, strengthened the federal government, and modernized the economy.

Lincoln grew up on the western frontier in Kentucky and Indiana. Largely self-educated, he became a lawyer in Illinois, a Whig Party leader, and a member of the Illinois House of Representatives, where he served from 1834 to 1846. Elected to the United States House of Representatives in 1846, Lincoln promoted rapid modernization of the economy through banks, tariffs, and railroads. Because he had originally agreed not to run for a second term in Congress, and his opposition to the Mexican–American War was unpopular among Illinois voters, Lincoln returned to Springfield and resumed his successful law practice. Reentering politics in 1854, he became a leader in building the new Republican Party, which had a statewide majority in Illinois. In 1858, while taking part in a series of highly publicized debates with his opponent and rival, Democrat Stephen A. Douglas, Lincoln spoke out against the expansion of slavery, but lost the U.S. Senate race to Douglas.

source: http://en.wikipedia.org/wiki/Abraham_Lincoln

What is Named Entity Recognition?

- Named Entity Recognition is the task of finding and classifying names in text.

Abraham Lincoln Listeni/ˈeɪbrəhæm ˈlɪŋkən/ (February 12, 1809 – April 15, 1865) was the 16th President of the United States, serving from March 1861 until his assassination in April 1865. Lincoln led the United States through its Civil War—its bloodiest war and its greatest moral, constitutional and political crisis.[1][2] In doing so, he preserved the Union, abolished slavery, strengthened the federal government, and modernized the economy.

Lincoln grew up on the western frontier in Kentucky and Indiana. Largely self-educated, he became a lawyer in Illinois, a Whig Party leader, and a member of the Illinois House of Representatives, where he served from 1834 to 1846. Elected to the United States House of Representatives in 1846, Lincoln promoted rapid modernization of the economy through banks, tariffs, and railroads. Because he had originally agreed not to run for a second term in Congress, and his opposition to the Mexican–American War was unpopular among Illinois voters, Lincoln returned to Springfield and resumed his successful law practice. Reentering politics in 1854, he became a leader in building the new Republican Party, which had a statewide majority in Illinois. In 1858, while taking part in a series of highly publicized debates with his opponent and rival, Democrat Stephen A. Douglas, Lincoln spoke out against the expansion of slavery, but lost the U.S. Senate race to Douglas.

People

source: http://en.wikipedia.org/wiki/Abraham_Lincoln

What is Named Entity Recognition?

- Named Entity Recognition is the task of finding and classifying names in text.

Abraham Lincoln Listeni/ˈeɪbrəhæm ˈlɪŋkən/ (February 12, 1809 – April 15, 1865) was the 16th President of the United States, serving from March 1861 until his assassination in April 1865. Lincoln led the United States through its Civil War—its bloodiest war and its greatest moral, constitutional and political crisis.[1][2] In doing so, he preserved the Union, abolished slavery, strengthened the federal government, and modernized the economy.

Lincoln grew up on the western frontier in Kentucky and Indiana. Largely self-educated, he became a lawyer in Illinois, a Whig Party leader, and a member of the Illinois House of Representatives, where he served from 1834 to 1846. Elected to the United States House of Representatives in 1846, Lincoln promoted rapid modernization of the economy through banks, tariffs, and railroads. Because he had originally agreed not to run for a second term in Congress, and his opposition to the Mexican–American War was unpopular among Illinois voters, Lincoln returned to Springfield and resumed his successful law practice. Reentering politics in 1854, he became a leader in building the new Republican Party, which had a statewide majority in Illinois. In 1858, while taking part in a series of highly publicized debates with his opponent and rival, Democrat Stephen A. Douglas, Lincoln spoke out against the expansion of slavery, but lost the U.S. Senate race to Douglas.

People

Locations

source: http://en.wikipedia.org/wiki/Abraham_Lincoln

What is Named Entity Recognition?

- Named Entity Recognition is the task of finding and classifying names in text.

Abraham Lincoln Listeni/ˈeɪbrəhæm ˈlɪŋkən/ (February 12, 1809 – April 15, 1865) was the 16th President of the United States, serving from March 1861 until his assassination in April 1865. Lincoln led the United States through its Civil War—its bloodiest war and its greatest moral, constitutional and political crisis.[1][2] In doing so, he preserved the Union, abolished slavery, strengthened the federal government, and modernized the economy.

Lincoln grew up on the western frontier in Kentucky and Indiana. Largely self-educated, he became a lawyer in Illinois, a Whig Party leader, and a member of the Illinois House of Representatives, where he served from 1834 to 1846. Elected to the United States House of Representatives in 1846, Lincoln promoted rapid modernization of the economy through banks, tariffs, and railroads. Because he had originally agreed not to run for a second term in Congress, and his opposition to the Mexican–American War was unpopular among Illinois voters, Lincoln returned to Springfield and resumed his successful law practice. Reentering politics in 1854, he became a leader in building the new Republican Party, which had a statewide majority in Illinois. In 1858, while taking part in a series of highly publicized debates with his opponent and rival, Democrat Stephen A. Douglas, Lincoln spoke out against the expansion of slavery, but lost the U.S. Senate race to Douglas.

People

Locations

Organisations

source: http://en.wikipedia.org/wiki/Abraham_Lincoln

What is Named Entity Recognition?

- Named Entity Recognition is the task of finding and classifying names in text.

Abraham Lincoln Listeni/ˈeɪbrəhæm ˈlɪŋkən/ (February 12, 1809 – April 15, 1865) was the 16th President of the United States, serving from March 1861 until his assassination in April 1865. Lincoln led the United States through its Civil War—its bloodiest war and its greatest moral, constitutional and political crisis.[1][2] In doing so, he preserved the Union, abolished slavery, strengthened the federal government, and modernized the economy.

Lincoln grew up on the western frontier in Kentucky and Indiana. Largely self-educated, he became a lawyer in Illinois, a Whig Party leader, and a member of the Illinois House of Representatives, where he served from 1834 to 1846. Elected to the United States House of Representatives in 1846, Lincoln promoted rapid modernization of the economy through banks, tariffs, and railroads. Because he had originally agreed not to run for a second term in Congress, and his opposition to the Mexican–American War was unpopular among Illinois voters, Lincoln returned to Springfield and resumed his successful law practice. Reentering politics in 1854, he became a leader in building the new Republican Party, which had a statewide majority in Illinois. In 1858, while taking part in a series of highly publicized debates with his opponent and rival, Democrat Stephen A. Douglas, Lincoln spoke out against the expansion of slavery, but lost the U.S. Senate race to Douglas.

People

Time

Locations

Organisations

source: http://en.wikipedia.org/wiki/Abraham_Lincoln

What is Named Entity Recognition?

- Named Entity Recognition is the task of finding and classifying names in text.

Abraham Lincoln Listeni/ˈeɪbrəhæm ˈlɪŋkən/ (February 12, 1809 – April 15, 1865) was the 16th President of the United States, serving from March 1861 until his assassination in April 1865. Lincoln led the United States through its Civil War—its bloodiest war and its greatest moral, constitutional and political crisis.[1][2] In doing so, he preserved the Union, abolished slavery, strengthened the federal government, and modernized the economy.

Lincoln grew up on the western frontier in Kentucky and Indiana. Largely self-educated, he became a lawyer in Illinois, a Whig Party leader, and a member of the Illinois House of Representatives, where he served from 1834 to 1846. Elected to the United States House of Representatives in 1846, Lincoln promoted rapid modernization of the economy through banks, tariffs, and railroads. Because he had originally agreed not to run for a second term in Congress, and his opposition to the Mexican–American War was unpopular among Illinois voters, Lincoln returned to Springfield and resumed his successful law practice. Reentering politics in 1854, he became a leader in building the new Republican Party, which had a statewide majority in Illinois. In 1858, while taking part in a series of highly publicized debates with his opponent and rival, Democrat Stephen A. Douglas, Lincoln spoke out against the expansion of slavery, but lost the U.S. Senate race to Douglas.

People

Time

Locations

Events

Organisations

source: http://en.wikipedia.org/wiki/Abraham_Lincoln

Document Forensics
Network Institute project VU - Deloitte

- Investigating unsavoury business practices (e.g., slavery, fraud, bribery) can involve processing large numbers of contracts, yearly reports and external (news) sources that may reflect on a company's reputation and relations.

- Labour intensive task mainly using text search to identify relevant documents that are then manually processed.

- Project goals:

  - Extract the relevant concepts from unstructured texts (e.g. news) as well as semi-structured (e.g., contracts and financial) documents:

    - name of suppliers; the type of relationship between companies, executive management

  - Populate knowledge graphs and link them to publicly available knowledge graphs.

  - Knowledge graphs should reflect the temporal binding and provenance of the extracted relations and properties.

  - Enable automated reasoning about companies and their relationships such as structure of ownership or supply chains and their dynamics

Diligence detection

Auzina and Kim, 2019, Automated Due Diligence: Building Knowledge Graphs from News, Network Institute, VU University

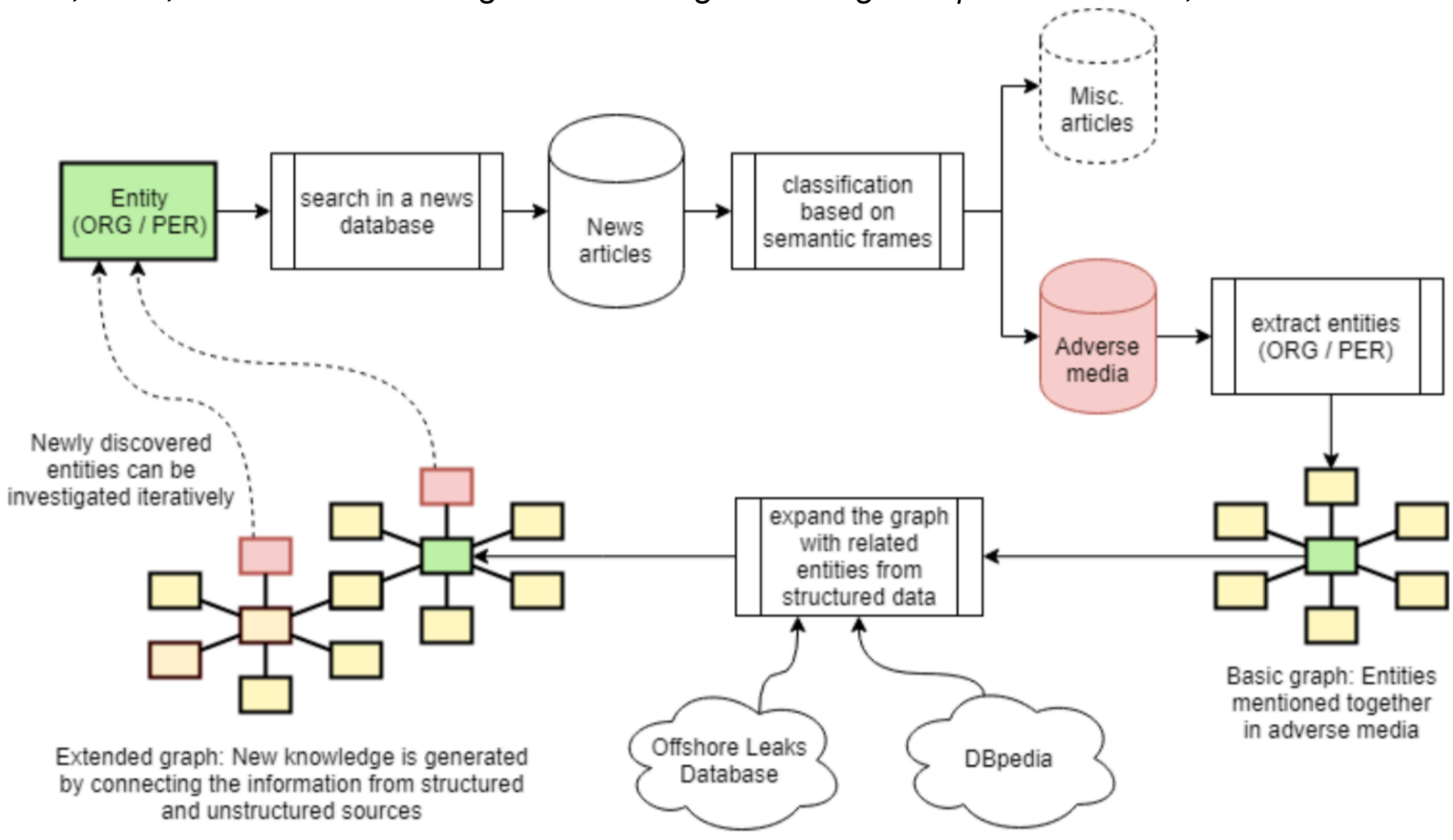


Figure 1: Overview of the proposed automated due diligence solution.

Adverse media classifier

Port of Moerdijk

Mitsubishi Materials

Kobe Steel

	MM dataset	KS dataset
Source	Nexis Uni[7]	Nexis Uni
Search term	Mitsubishi Materials	Kobe Steel
Content type	news	news
Language	English	English
Dates	06/91-02/19	01/00-04/19
# articles	707	1,774
# unique frames	460	540

Table 1: Datasets overview

Topic	# articles
MM dataset	
data falsification	65
forced labor during WWII	36
groundwater contamination	2
condos on contaminated soil	2
factory blast	1
KS dataset	
data falsification	115
tax evasion	2
asbestos-related employee death	1
employee embezzlement	1
safety and health violations	1

Table 2: Adverse media topics encountered during annotation

Model	Test set	Class	Precision	Recall	F1-score	Support
MM_10	KS: active learning sample (N=300)	0	0.65	0.89	0.75	177
		1	**0.67**	**0.31**	**0.42**	**123**
MM_10	KS: random sample (N=300)	0	0.90	0.95	0.92	242
		1	**0.73**	**0.55**	**0.63**	**58**
KS_10	MM: active learning sample (N=300)	0	0.82	0.91	0.86	194
		1	**0.80**	**0.63**	**0.71**	**106**
KS_10	MM: random sample (N=300)	0	0.91	0.97	0.94	240
		1	**0.82**	**0.62**	**0.70**	**60**

Table 3: Quantitative Evaluation Results (class 1: adverse media)

Auzina and Kim, 2019

Entity relationship graph

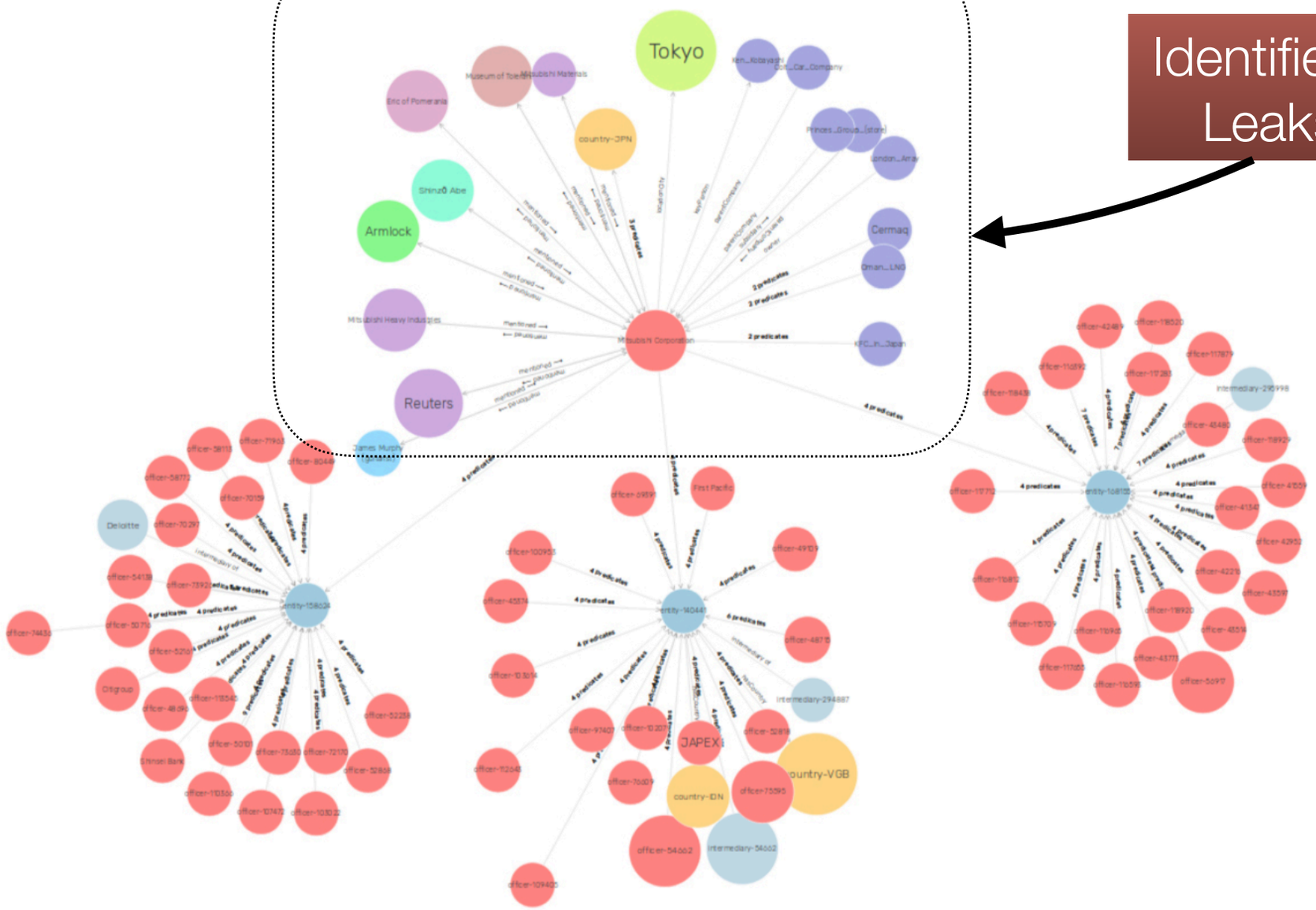


Figure 7: Extracted Officers and Intermediaries of the 3 identified entities

Auzina and Kim, 2019

Named entity detection and linking (NERC-D/L)

- NER(Recognition): detecting the phrase that is the name of an entity
- NEC(Classification): assigning an entity type to the phrase
- NEL(Linking) or NED(Disambiguation): establishing the identity of the entity in a given reference database (Wikipedia, DBPedia, YAGO)
- Coreference: any phrase that makes reference to an entity instance, including pronouns, noun phrases, abbreviations, acronyms, etc…
- Preprocessing: tokenisation, sentence splitting, Part-of-speech tagging, lookup, grammar rules, coreference

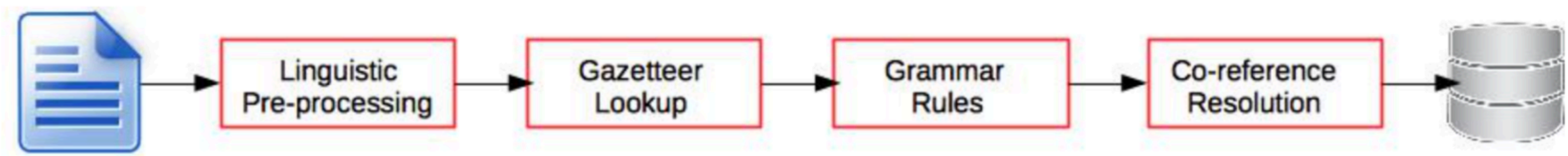


Figure 3.1: Typical NERC pipeline

What makes it a hard task?

- **variation** (IBM, The Big Blue, New York, NY, The Big Apple) and **ambiguity** **(**distinguish named entities and entities):
  - *MAY MAY RULE IN MAY*
  - *Austin Reed, Parkinson's disease, Pythagoras' Theorem*
- **extent**: *Sir Robert Walpole; [Abraham Lincoln, [the 16th President of the [United States]]]* <— nested entities
- **types**: e.g. *Criminal* as a subclass of *Person*, http://nerd.eurecom.fr/ontology, Fine-grained entity typing (e.g. FIGER uses 112 types from Freebase)
- **Time**: 8am, yesterday, last week, this month/year (TimeML types DAY, TIME, DURATION, SET)
- **Metonymy**: US, Holland, The Netherlands, Ford, Volkswagen

NERC feature engineering

- Word level features
- Digit patterns
- Common word endings
- Functions over words: non-alphabetic (A.T.&T.), n-grams
- Lookup features
- Document & Corpus features

Word level features

Table 1. Word-level features

Features	Examples
Case	– Starts with a capital letter – Word is all uppercased – The word is mixed case (e.g., ProSys, eBay)
Punctuation	– Ends with period, has internal period (e.g., St., I.B.M.) – Internal apostrophe, hyphen or ampersand (e.g., O'Connor)
Digit	– Digit pattern (*see Section 3.1.1*) – Cardinal and ordinal – Roman number – Word with digits (e.g., W3C, 3M)
Character	– Possessive mark, first person pronoun – Greek letters
Morphology	– Prefix, suffix, singular version, stem – Common ending (*see Section 3.1.2*)
Part-of-speech	– proper name, verb, noun, foreign word
Function	– Alpha, non-alpha, n-gram (*see Section 3.1.3*) – lowercase, uppercase version – pattern, summarized pattern (*see Section 3.1.4*) – token length, phrase length

From Nadeau, D., & Sekine, S. (2007)

Gazetteers & lexicons

Table 2. List lookup features.

Features	Examples
General list	– General dictionary (see Section 3.2.1) – Stop words (function words) – Capitalized nouns (e.g., January, Monday) – Common abbreviations
List of entities	– Organization, government, airline, educational – First name, last name, celebrity – Astral body, continent, country, state, city
List of entity cues	– Typical words in organization (see 3.2.2) – Person title, name prefix, post-nominal letters – Location typical word, cardinal point

From Nadeau, D., & Sekine, S. (2007)

Document features

Table 3. Features from documents.

Features	Examples
Multiple occurrences	– Other entities in the context – Uppercased and lowercased occurrences (see 3.3.1) – Anaphora, coreference (see 3.3.2)
Local syntax	– Enumeration, apposition – Position in sentence, in paragraph, and in document
Meta information	– Uri, email header, XML section, (see Section 3.3.3) – Bulleted/numbered lists, tables, figures
Corpus frequency	– Word and phrase frequency – Co-occurrences – Multiword unit permanency (see 3.3.4)

From Nadeau, D., & Sekine, S. (2007)

CONLL: Computational Natural Language Learning
https://www.conll.org

Every token on a separate line followed by TAB separated columns with annotations (TSV)

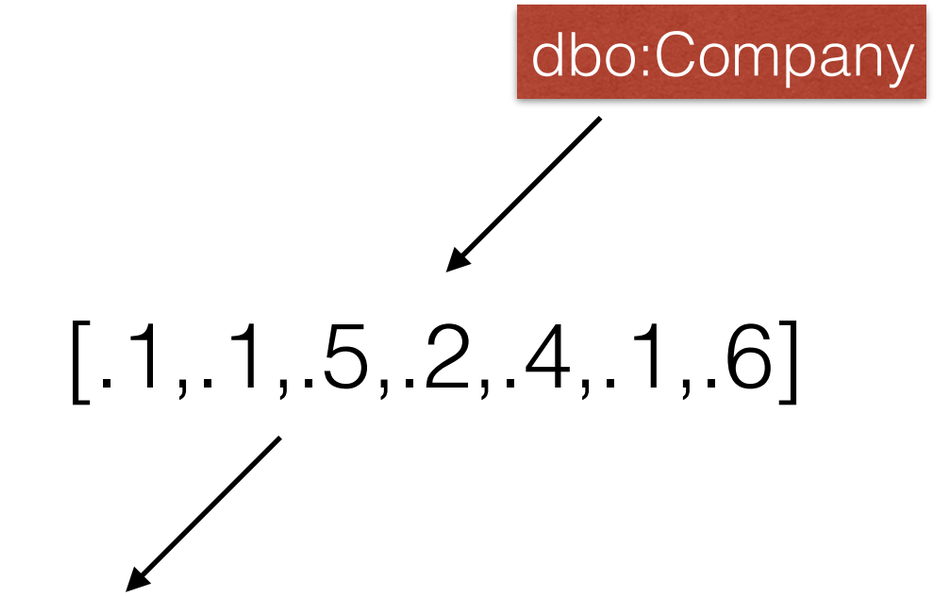
ID	FORM	LEMMA	UPOSTAG	XPOSTAG	FEATS	HEAD	DEPREL	DEPS	NER
# newdoc url = http://www.poweredbyosteons.org/2012/01/brief-history-of-bioarchaeological.html									
# newdoc s3 = s3://aws-publicdatasets/common-crawl/crawl-data/CC-MAIN-2016-07/segments...									
...									
# sent_id = http://www.poweredbyosteons.org/2012/01/brief-history-of-bioarchaeological.html#60									
# text = The American Museum of Natural History was established in New York in 1869.									
0	The	the	DT	DT	_	2	det	2:det	O
1	American	American	NNP	NNP	_	2	nn	2:nn	B-Organization
2	Museum	Museum	NNP	NNP	_	7	nsubjpass	7:nsubjpass	I-Organization
3	of	of	IN	IN	_	2	prep	_	I-Organization
4	Natural	Natural	NNP	NNP	_	5	nn	5:nn	I-Organization
5	History	History	NNP	NNP	_	3	pobj	2:prep_of	I-Organization
6	was	be	VBD	VBD	_	7	auxpass	7:auxpass	O
7	established	establish	VBN	VBN	_	7	ROOT	7:ROOT	O
8	in	in	IN	IN	_	7	prep	_	O
9	New	New	NNP	NNP	_	10	nn	10:nn	B-Location
10	York	York	NNP	NNP	_	8	pobj	7:prep_in	I-Location
11	in	in	IN	IN	_	7	prep	_	O
12	1869	1869	CD	CD	_	11	pobj	7:prep_in	O
13	.	.	.	.	_	7	punct	7:punct	O
...									

https://www.clips.uantwerpen.be/conll2003/ner/

IO(B) style
I = insight
O = outside
B = beginning

Entity embeddings

- **Light entity:** Groupon (dbo:Company)
  - **Met de korting**sbonnen van Groupon kunt u tegen hoge **korting kennismaken** met de diensten van een professionele fotograaf.
  - **Met de** Groupon **app** kan je nu ook onderweg de deals bekijken, kopen en inwisselen.
- **Dark Entity**: Scoupy
  - Via de gratis **app** voor iPhone of Android of via de website www.scoupy.nl kun je met **hoge korting** of zelfs helemaal gratis **kennismaken** met allerlei producten.
  - **Met de** Scoupy **app** kan je op zoek naar **korting**scoupons voor winkels en restaurants bij jou in de buurt.



[.1,.1,.5,.2,.4,.1,.6]

- Groupon
  [.3,.1,.2,.3,.7,.2,.3]

  Scoupy
  [.2,.1,.3,.2,.5,.1,.1]

Word embeddings

300-500 dimensions

Converting features to one-hot-vectors & embeddings

- "The president of Groupon eats an apple"
  - Groupon $[1]_{Case}$ +$[7]_{Length}$ + $[0,0,1,0,0,0]_{PoS}$ +

    $[0,0,00,0,0,0,0,0,0,0,1]_{Word}$ + $[1]_{Gazetteer}$ + $[.1,.3,.2,.6,.7.,2]_{Embedding}$
- "A manager of Scoupy swallows the banana"
  - Scoupy $[1]_{Case}$ + $[6]_{Length}$ + $[0,0,1,0,0,0]_{PoS}$ +

    $[0,0,00,0,0,0,0,0,0,1,0]_{Word}$ + $[0]_{Gazetteer}$ +$[.1,.2,.3,.4,.1.,1]_{Embedding}$

NERC as a sequence tagging task

- Sentences exhibit strong predictive probabilities for sequences of words and their tags, including IOB entity tags:
    - Abraham (B-PER) Lincoln (I-PER) ((O) February (B-T) 12 (I-T) , (I-T) 1809 (I-T)
- CRFs (Conditional Random Fields) are one of the most widely used algorithms for NERC
    - Graph models view NERC as a sequence classification task
    - Strong dependence between features and predictions in a sequence, e.g. I-LOC never occurs immediately after B-PER
- https://www.quora.com/What-are-the-pros-and-cons-of-these-three-sequence-models-MaxEnt-Markov-Model-Conditional-random-fields-and-recurrent-neural-networks

Sequence tagging problems in NLP
Part of speech tagging using Hidden Markov Models (HMM)

- http://www.cse.unsw.edu.au/~billw/cs9414/notes/nlp/ambiguity/ambiguity-2009.html

Large collection of text annotated with Part-of-Speech tags

[I, do, not, like, flies, on, my, food]
[Pr, V, Av, V, N, P, PR, N]
[She, flies, to, China, after, the, meeting]
[Pr, V, P, N, P, ART, N]

Table 7.6 The lexical generation probabilities

Pr(the | ART)	0.54		Pr(a | ART)	0.360
Pr(flies | N)	0.025		Pr(a | N)	0.001
Pr(flies | V)	0.076		Pr(flower | N)	0.063
Pr(like | V)	0.1		Pr(flower | V)	0.05
Pr(like | P)	0.068		Pr(birds | N)	0.076
Pr(like | N)	0.012			

Sequence tagging problems in NLP
Part of speech tagging using Hidden Markov Models (HMM)

- http://www.cse.unsw.edu.au/~billw/cs9414/notes/nlp/ambiguity/ambiguity-2009.html

Viterbi algorithm

[flies, like, a, flower]

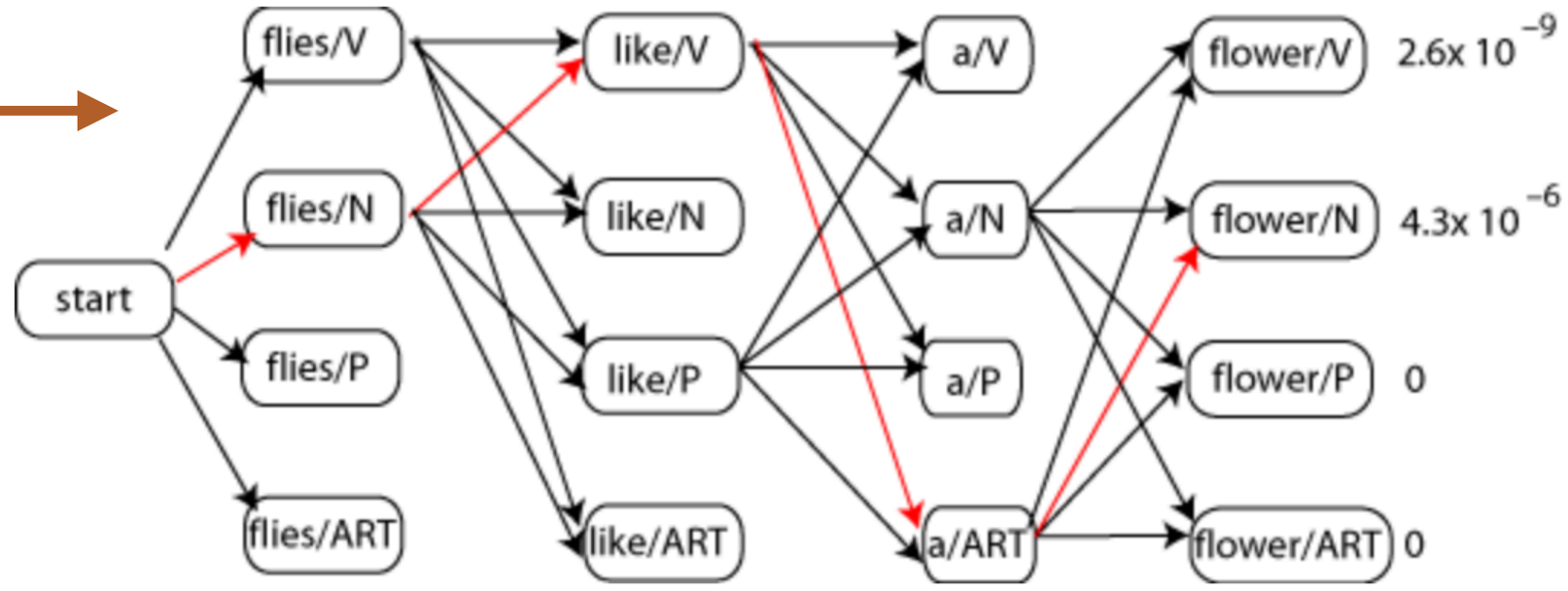
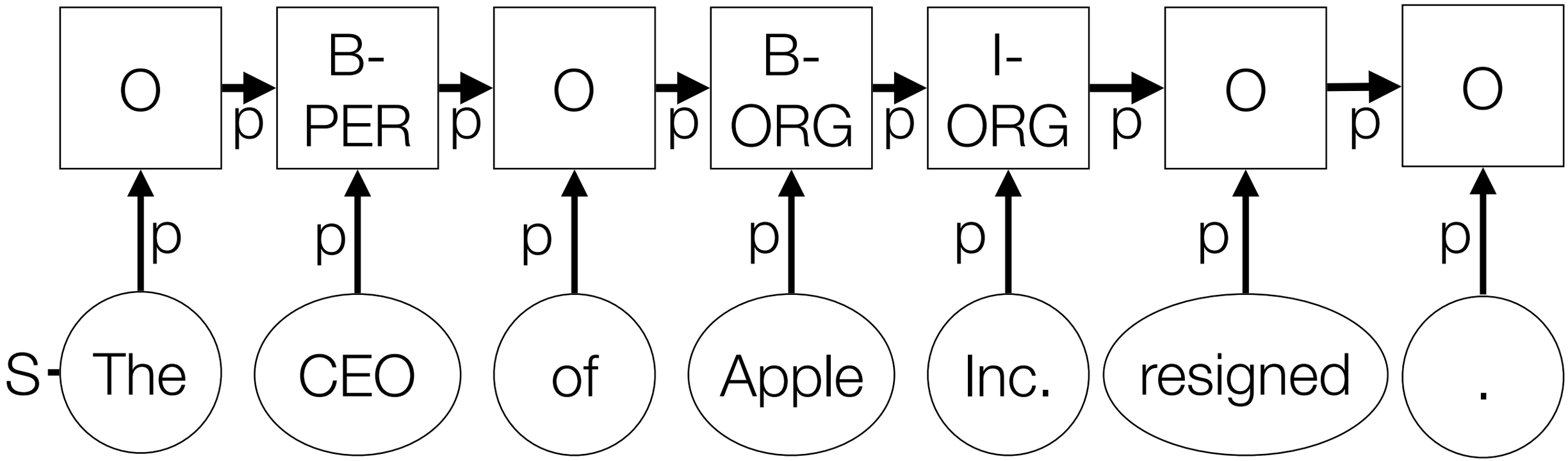


Table 7.6 The lexical generation probabilitie

Pr(the | ART)	0.54		Pr(a | ART)	0.3
Pr(flies | N)	0.025		Pr(a | N)	0.0
Pr(flies | V)	0.076		Pr(flower | N)	0.0
Pr(like | V)	0.1		Pr(flower | V)	0.0
Pr(like | P)	0.068		Pr(birds | N)	0.0
Pr(like | N)	0.012			

lexcat	SeqScore (lexcat,1)	SeqScore (lexcat,2)	SeqScore (lexcat,3)	SeqScore (lexcat,4)	BackPtr (lexcat,4)
V	7.6×10^{-6}	0.00031	0	2.6×10^{-9}	ART
N	0.00725	1.3×10^{-5}	1.2×10^{-7}	4.3×10^{-6}	ART
P	0	0.00022	0	0	∅
ART	0	0	7.2×10^{-5}	0	∅

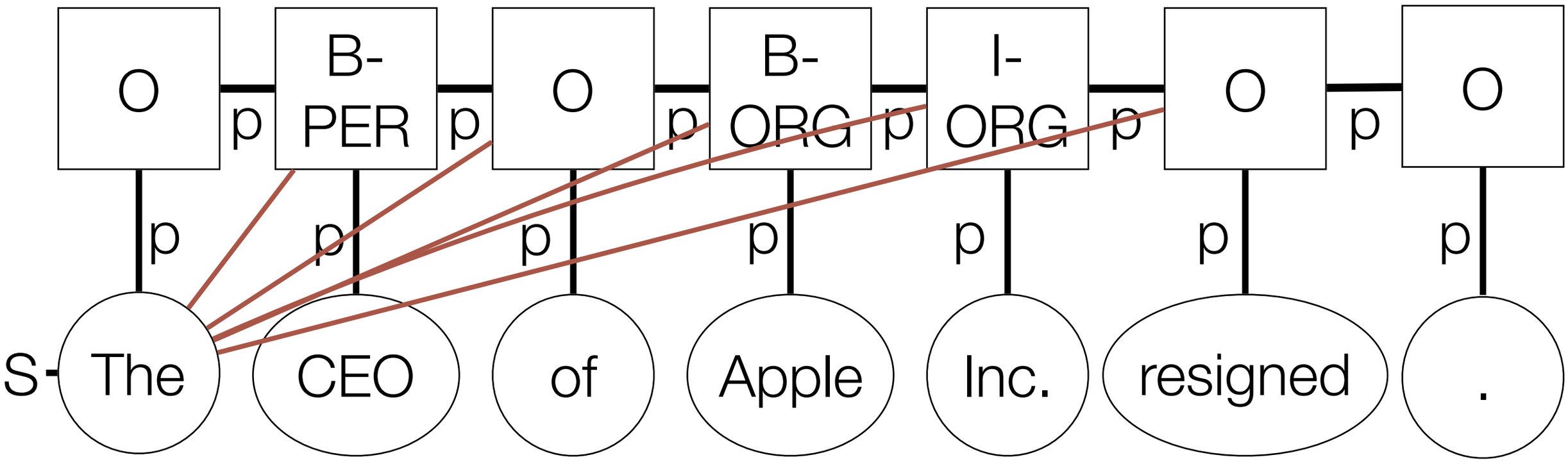
Conditional Random Field for IOB sequences



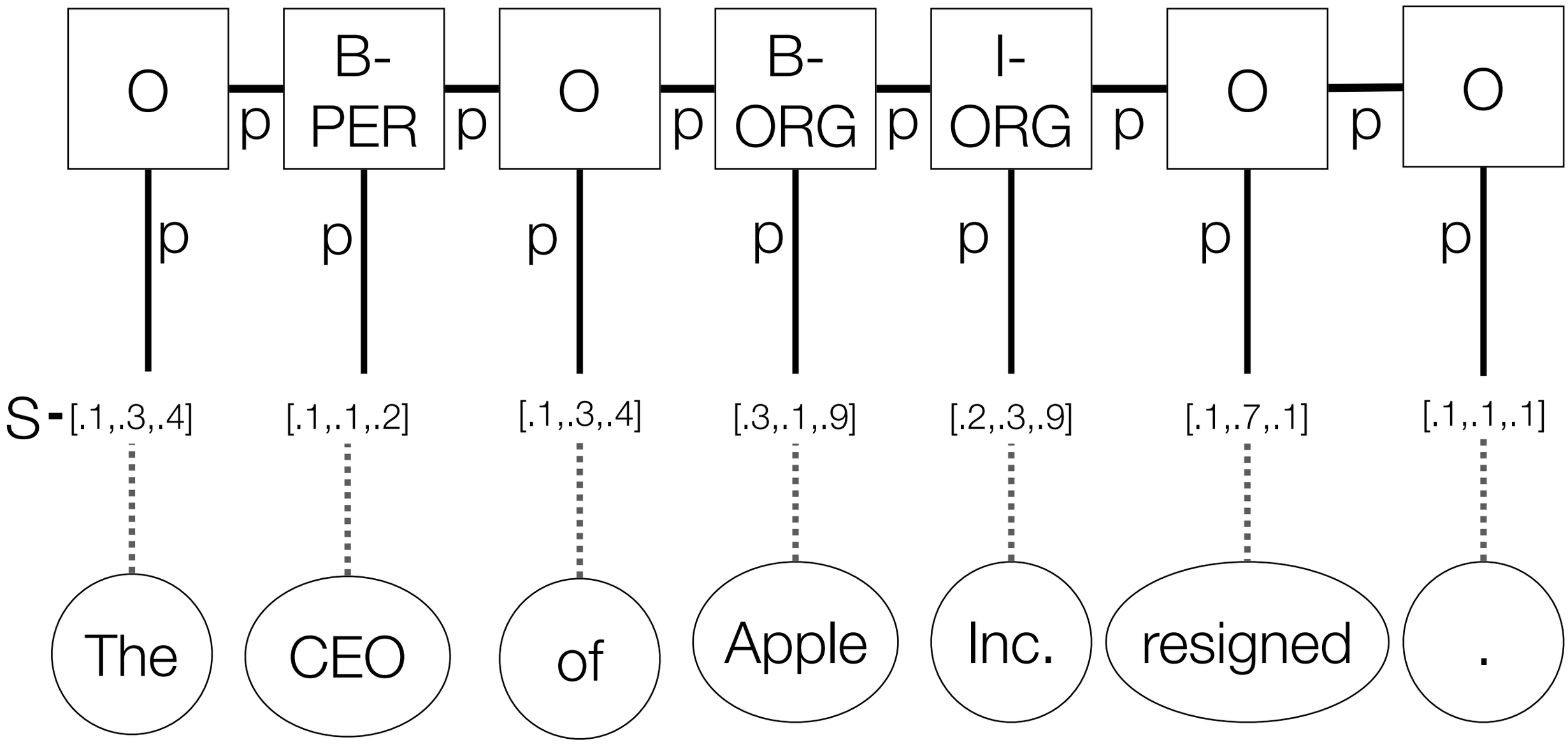
if token has Initial Capital & preceded by O then B

if token has Initial Capital followed by “Inc.” then ORG

Conditional Random Field for IOB sequences



Conditional Random Field for IOB sequences



NERC performance

Feature-engineered machine learning systems	Dict	SP	DU	EN	GE
Carreras et al. (2002) binary AdaBoost classifiers	Yes	81.39	77.05	-	-
Malouf (2002) - Maximum Entropy (ME) + features	Yes	73.66	68.08	-	-
Li et al. (2005) SVM with class weights	Yes	-	-	88.3	-
Passos et al. (2014) CRF	Yes	-	-	90.90	-
Ando and Zhang (2005a) Semi-supervised state of the art	No	-	-	89.31	75.27
Agerri and Rigau (2016)	Yes	**84.16**	**85.04**	**91.36**	**76.42**
Feature-inferring neural network word models					
Collobert et al. (2011) Vanilla NN +SLL / Conv-CRF	No	-	-	81.47	-
Huang et al. (2015) Bi-LSTM+CRF	No	-	-	84.26	-
Yan et al. (2016) Win-BiLSTM (English), FF (German) (Many fets)	Yes	-	-	88.91	**76.12**
Collobert et al. (2011) Conv-CRF (SENNA+Gazetteer)	Yes	-	-	89.59	-
Huang et al. (2015) Bi-LSTM+CRF+ (SENNA+Gazetteer)	Yes	-	-	**90.10**	-
Feature-inferring neural network character models					
Gillick et al. (2015) – BTS	No	**82.95**	**82.84**	**86.50**	**76.22**
Kuru et al. (2016) CharNER	No	82.18	79.36	84.52	70.12
Feature-inferring neural network word + character models					
Yang et al. (2017)	Yes	85.77	**85.19**	91.26	-
Luo (2015)	Yes	-	-	91.20	-
Chiu and Nichols (2015)	Yes	-	-	**91.62**	-
Ma and Hovy (2016)	No	-	-	91.21	-
Santos and Guimaraes (2015)	No	82.21	-	-	-
Lample et al. (2016)	No	85.75	81.74	90.94	**78.76**
Bharadwaj et al. (2016)	Yes	**85.81**	-	-	-
Dernoncourt et al. (2017)	No	-	-	90.5	-
Feature-inferring neural network word + character + affix models					
Re-implementation of Lample et al. (2016) (100 Epochs)	No	85.34	85.27	90.24	78.44
Yadav et al. (2018)(100 Epochs)	No	86.92	87.50	90.69	78.56
Yadav et al. (2018) (150 Epochs)	No	**87.26**	**87.54**	90.86	**79.01**

Table 1: Comparison of NER systemsin four languages: CoNLL 2002 Spanish (SP), CoNLL 2002 Dutch (DU), CoNLL 2003 English (EN), and CoNLL 2003 German (GE). Dict indicates whether or not the approach makes use of dictionary lookups. Best performance in each category is highlighted in bold.

Yadav, Vikas, and Steven Bethard. "A survey on recent advances in named entity recognition from deep learning models." In *Proceedings of the 27th International Conference on Computational Linguistics*, pp. 2145-2158. 2018.

Feature-inferring NN systems outperform feature-engineered systems, despite the latter's access to domain specific rules, knowledge, features, and lexicons

Neural network (LSTM) and CRF

https://github.com/guillaumegenthial/tf_ner

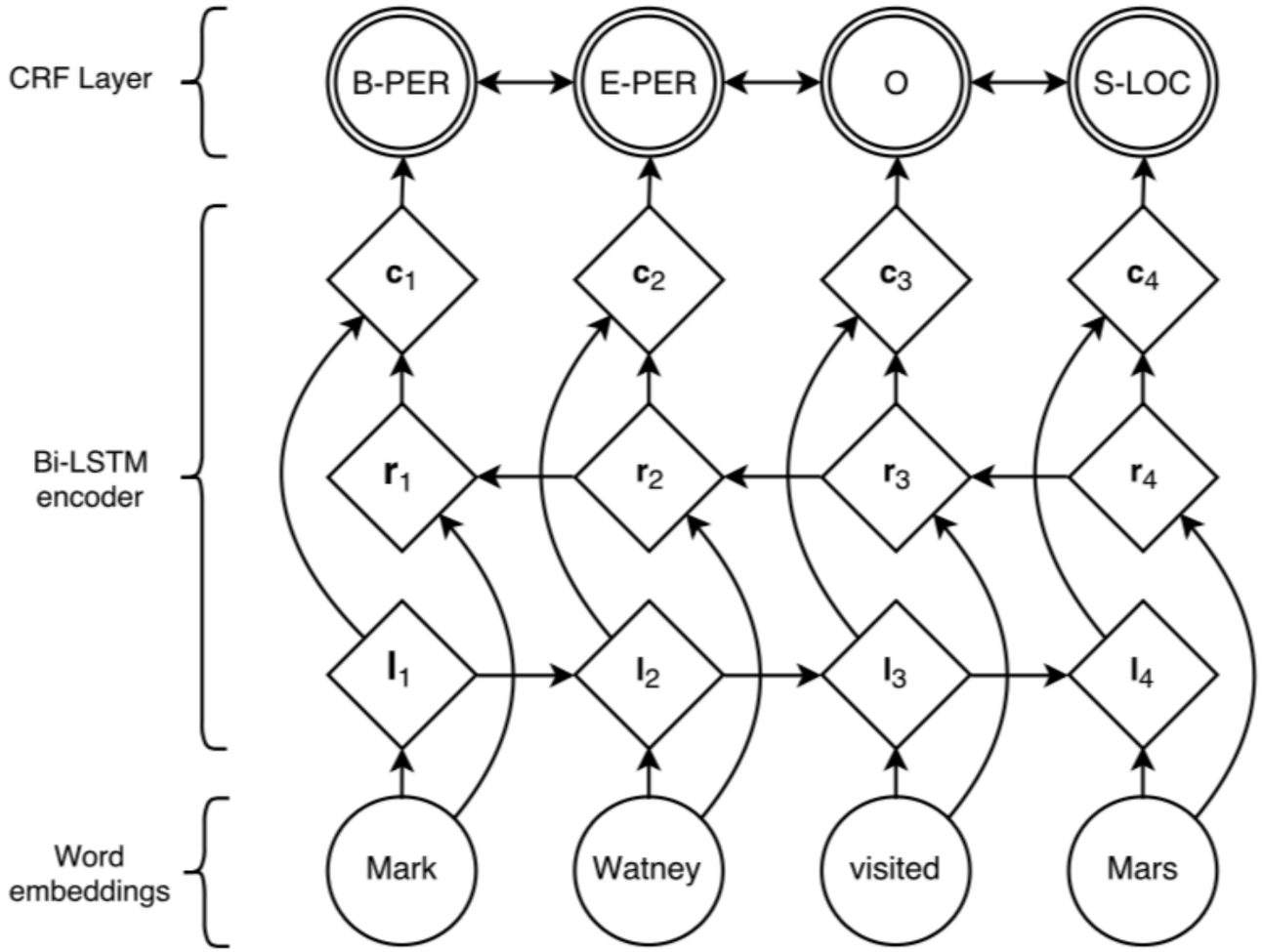


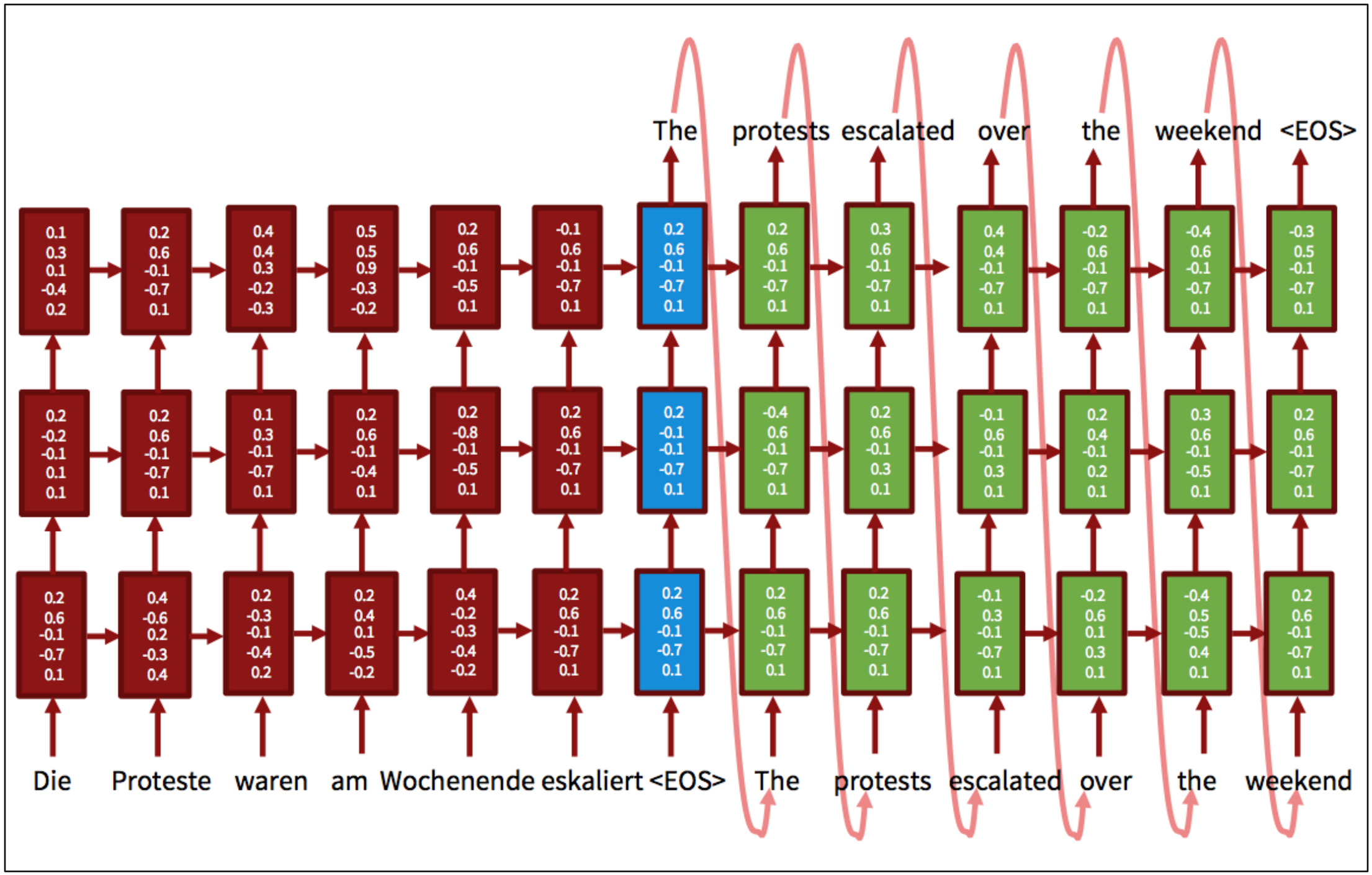
Figure 1: Main architecture of the network. Word embeddings are given to a bidirectional LSTM. $\mathbf{l}_i$ represents the word i and its left context, $\mathbf{r}_i$ represents the word i and its right context. Concatenating these two vectors yields a representation of the word i in its context, $\mathbf{c}_i$.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami and Chris Dyer, 2016, Neural Architectures for Named Entity Recognition, NAACL.

Long Short-Term Memory LSTM

Zhiheng Huang, Wei Xu, Kai Yu 2015, Bidirectional LSTM-CRF Models for Sequence Tagging, arXiv.1508.01991v1

Machine translation: long-short-term-memory



Google Neural MT

Bidirectional LSTM models for NERC

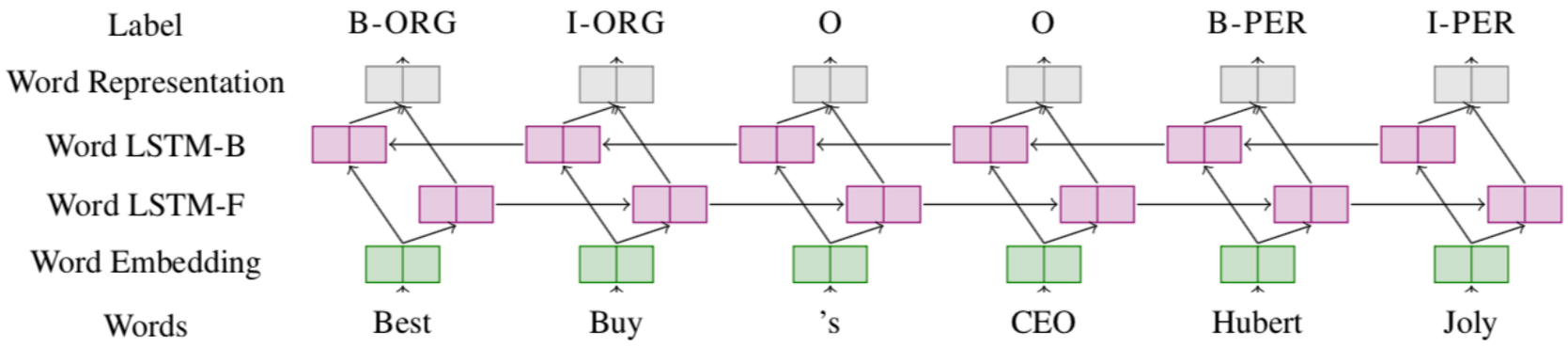


Figure 1: Word level NN architecture for NER

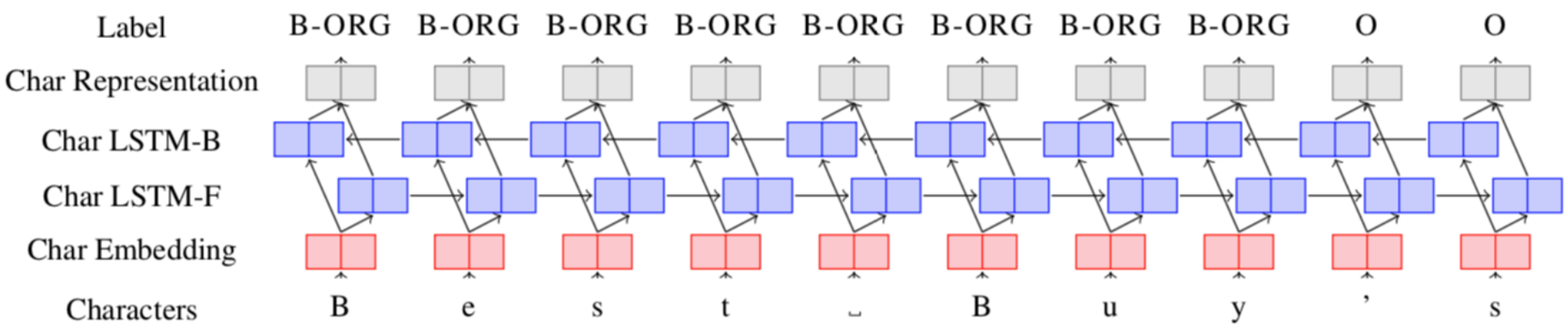


Figure 2: Character level NN architecture for NER

Yadav & Bethard 2018

https://towardsdatascience.com/besides-word-embedding-why-you-need-to-know-character-embedding-6096a34a3b10

Bidirectional LSTM models for NERC

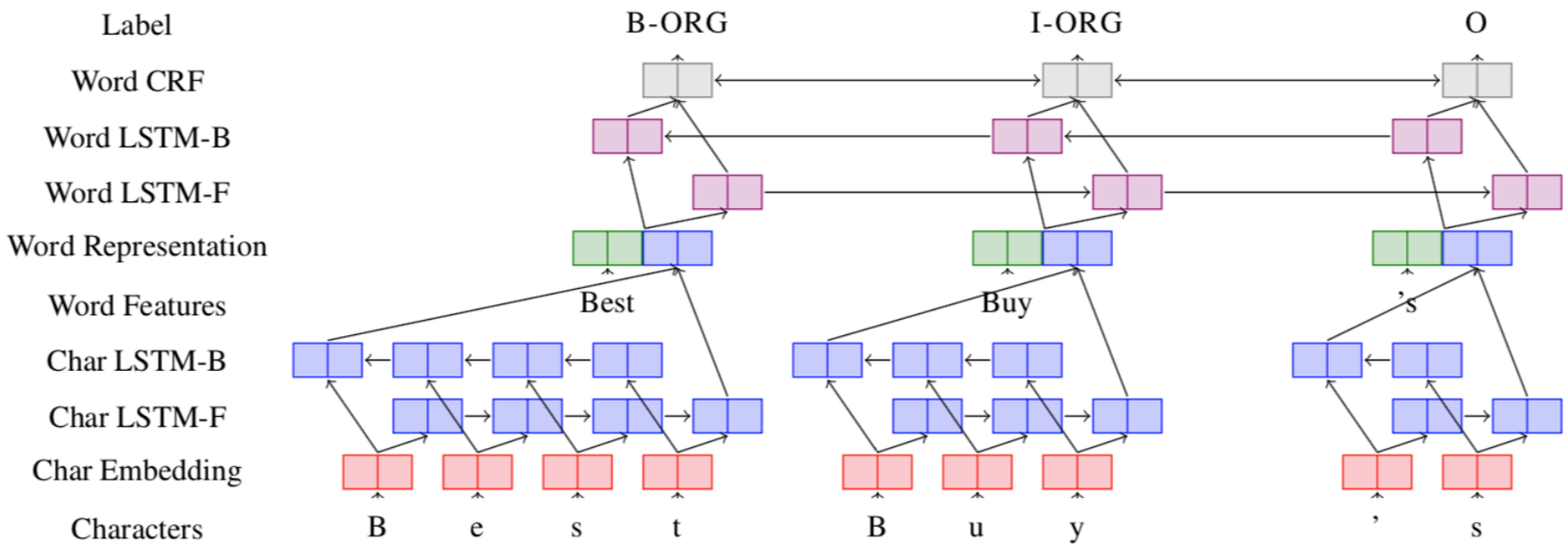


Figure 3: Word+character level NN architecture for NER

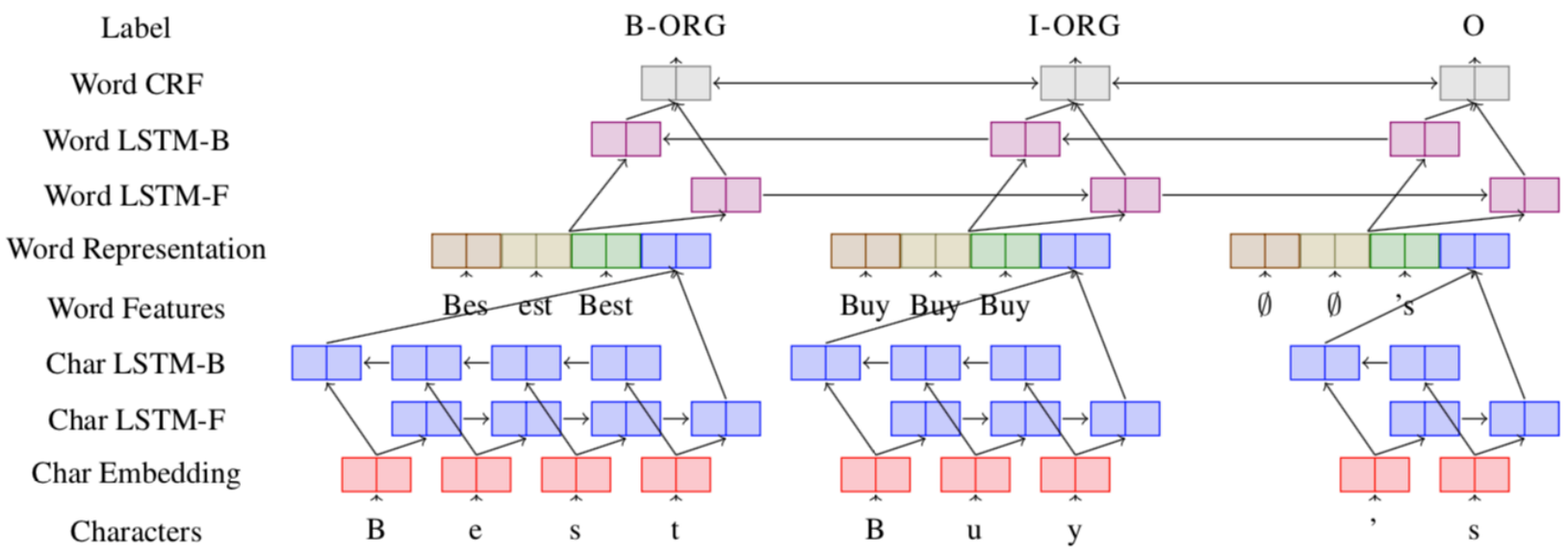


Figure 4: Word+character+affix level NN architecture for NER

Affix embeddings from all n-gram prefixes and suffixes of words in the training corpus

Yadav & Bethard 2018

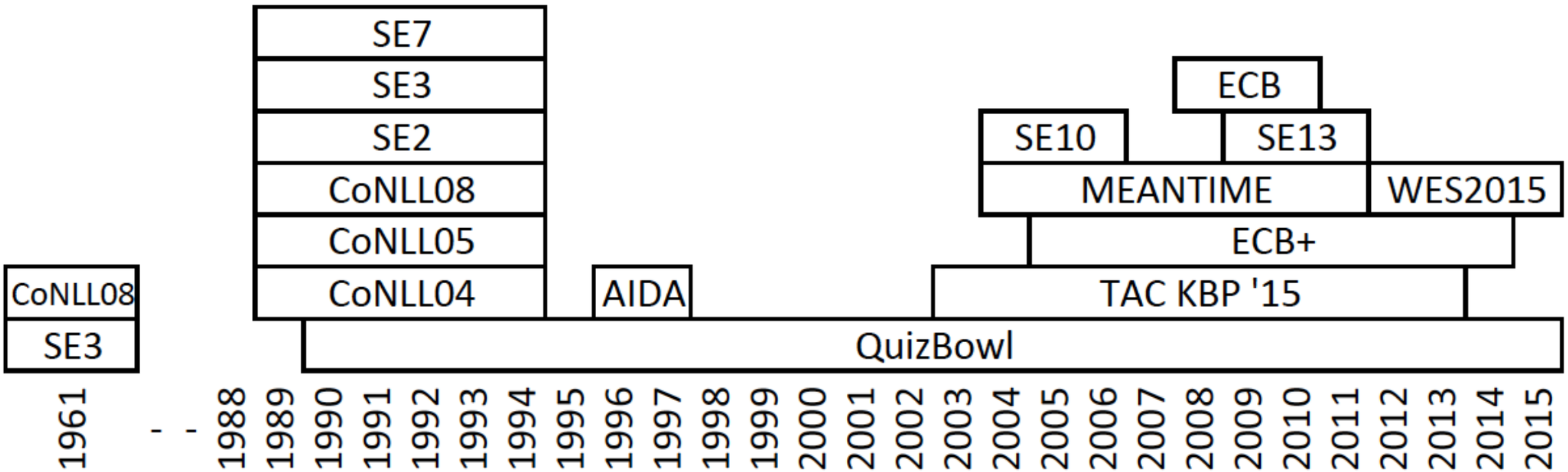
Factors that impact performance for NERC

- The annotation of the spans, annotation of nesting
  - [[[White House] [press] secretary] Scott McClellan]
  - [The [CEO] of the [US]-Based company [Facebook]]
- Type of text: news or tweets/ social media
- Entity types: people, organisations, amounts, dates, events
- Amount of training data
- Difference between training data and test data:
  - domain dependency of entities
  - training data rapidly becomes obsolete

Measuring performance for entities

- Is simple precision and recall enough?
- Neil Young & Crazy Horse
- Score per chunk (only give a true positive score if the entire NE is correctly classified)
    - Here “Neil” gets a false negative score, “&” gets a false positive score, “Horse” again a false negative.
- Some other metrics exist (e.g., MUC) that give partial credit (complex rules)

How specific is our data?



Ilievski, Postma, and Vossen, Semantic overfitting: what `world' do we consider when evaluating disambiguation of text? COLING 2016.

What is the effect of gazetteers on data over time?

Performance drops when shifting data

Agerri and Rigau 2016

Table 6: NERC CoNLL 2003 testb results.

	Precision	Recall	F1
Newsreader (ixa-pipe-nerc)	91.64	90.21	**90.92**
Stanford NER (CRF)	-	-	88.08
Ratinov et al. (2009)	-	-	90.57
Passos et al. (2014)	-	-	90.90

Table 7: NERC Intra-document Benchmarking with Wikinews.

System	mention extent	Precision	Recall	F1
Newsreader (ixa-pipe-nerc)	Inner phrase-based	62.15	76.06	**68.41**
Stanford NER (all english crf distsim)	Inner phrase-based	63.53	68.21	65.79
Newsreader (ixa-pipe-nerc)	Inner token-based	72.17	79.31	**75.57**
Stanford NER (all english crf distsim)	Inner token-based	77.14	71.77	74.36
Newsreader (ixa-pipe-nerc)	Outer phrase-based	53.01	68.03	**59.59**
Stanford NER (all english crf distsim)	Outer phrase-based	52.86	59.51	55.99
Newsreader (ixa-pipe-nerc)	Outer token-based	73.40	67.20	**70.16**
Stanford NER (all english crf distsim)	Outer token-based	78.22	60.63	68.31

P. Vossen, R. Agerri, I. Aldabe, A. Cybulska, M. van Erp, A. Fokkens, E. Laparra, A. Minard, A. P. Aprosio, G. Rigau, M. Rospocher, and R. Segers, “NewsReader: Using knowledge resources in a cross-lingual reading machine to generate more knowledge from massive streams of news”, Special issue knowledge-based systems, elsevier, 2016. dx.doi.org/10.1016/j.knosys.2016.07.013

Domain specific NER

	Dict	MedLine (80.10%)			DrugBank (19.90%)			Complete dataset		
		P	R	F1	P	R	F1	P	R	F1
Feature-engineered machine learning systems										
Rocktäschel et al. (2013)	Yes	60.70	55.80	58.10	88.10	87.50	87.80	73.40	69.80	71.50
Liu et al. (2015) (baseline)	No	-	-	-	-	-	-	78.41	67.78	72.71
Liu et al. (2015) (MED. emb.)	No	-	-	-	-	-	-	82.70	69.68	75.63
Liu et al. (2015) (state of the art)	Yes	78.77	60.21	68.25	90.60	88.82	**89.70**	84.75	72.89	**78.37**
NN word model										
Chalapathy et al. (2016) (relaxed performance)	No	52.93	52.57	52.75	87.07	83.39	85.19	-	-	-
NN word + character model Yadav et al. (2018)	No	73	62	67	87	86	87	79	72	75
NN word + character + affix model Yadav et al. (2018)	No	74	64	**69**	89	86	87	81	74	77

91+ on CoNLL 2003

90+ on CoNLL 2003

Table 2: DrugNER results on the MedLine and DrugBank test data (80.10% and 19.90% of the test data, respectively). The Yadav et al. (2018) experiments report no decimal places because they were run after the end of shared task, and the official evaluation script outputs no decimal places.

Yadav & Bethard 2018

NERC References

- Yadav, Vikas, and Steven Bethard. "A survey on recent advances in named entity recognition from deep learning models." In Proceedings of the 27th International Conference on Computational Linguistics, pp. 2145-2158. 2018.
- Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami and Chris Dyer, 2016, Neural Architectures for Named Entity Recognition, NAACL.
- P. Vossen, R. Agerri, I. Aldabe, A. Cybulska, M. van Erp, A. Fokkens, E. Laparra, A. Minard, A. P. Aprosio, G. Rigau, M. Rospocher, and R. Segers, “NewsReader: Using knowledge resources in a cross-lingual reading machine to generate more knowledge from massive streams of news”, Special issue knowledge-based systems, elsevier, 2016. dx.doi.org/10.1016/j.knosys.2016.07.013
- Zhiheng Huang, Wei Xu, Kai Yu 2015, Bidirectional LSTM-CRF Models for Sequence Tagging, arXiv.1508.01991v1
- Ilievski, Postma, and Vossen, Semantic overfitting: what `world' do we consider when evaluating disambiguation of text? COLING 2016.
- Rodrigo Agerri and German Rigau. 2016. Robust multilingual named entity recognition with shallow semi- supervised features. Artificial Intelligence, 238:63–82.
- https://github.com/guillaumegenthial/tf_ner
- https://www.depends-on-the-definition.com/named-entity-recognition-conditional-random-fields-python/
- https://towardsdatascience.com/besides-word-embedding-why-you-need-to-know-character-embedding-6096a34a3b10
- https://machinelearningmastery.com/develop-character-based-neural-language-model-keras/
- https://www.kaggle.com/abhinavwalia95/entity-annotated-corpus

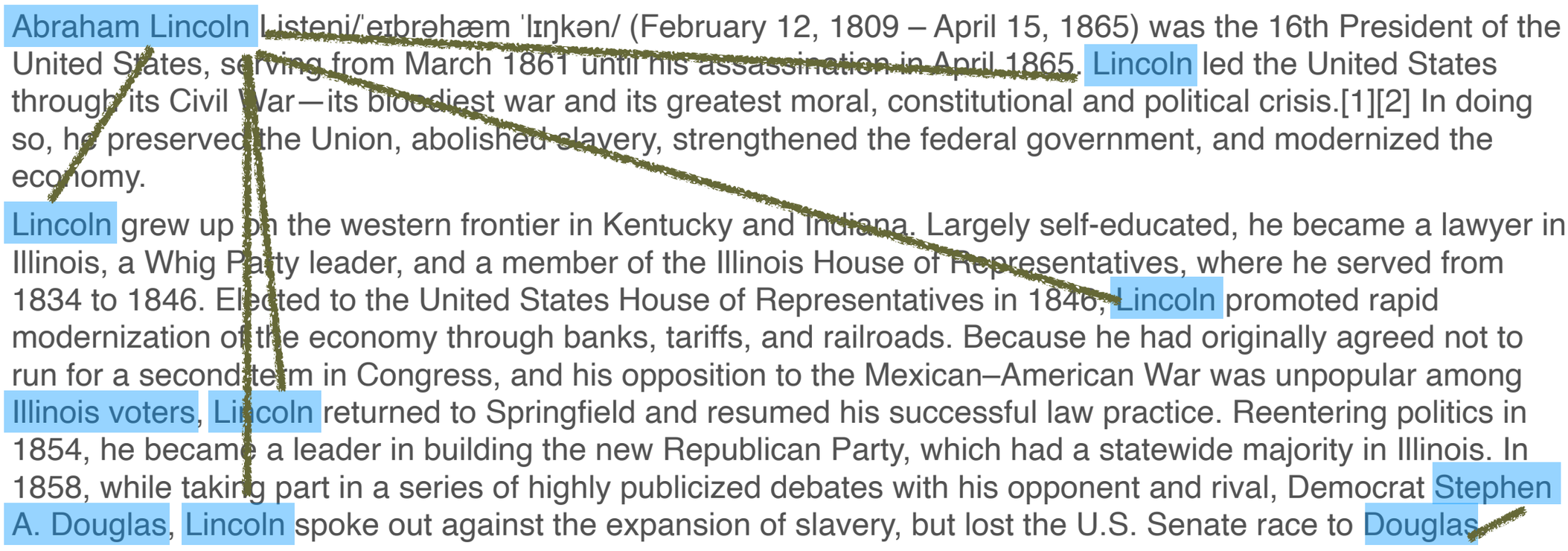
There is more than named entity expressions

- Identities: people with the same name (Joe Smith) are not necessarily people with the same identity

- Coreference: also phrases (“the president”) and pronouns (“he”, “she”) can make reference to the same entity

What is Coreference Resolution

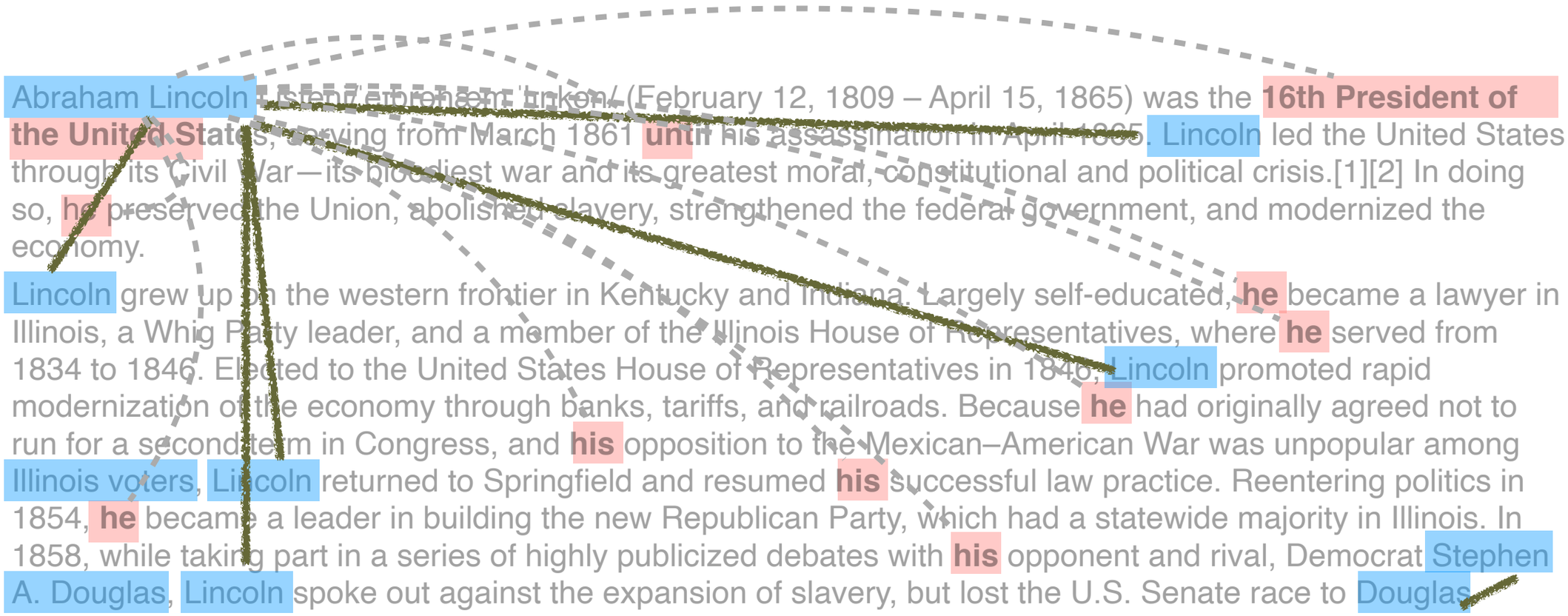
- Coreference resolution is the task of finding out which words/phrases refer to the same entity



source: http://en.wikipedia.org/wiki/Abraham_Lincoln

But it's actually more complicated

- Coreference resolution is the task of finding out which words/phrases refer to the same object



source: http://en.wikipedia.org/wiki/Abraham_Lincoln

But it’s actually more complicated

- Coreference resolution is the task of finding out which words/phrases refer to the same object

Abraham Lincoln Listeni/ˈeɪbrəhæm ˈlɪŋkən/ (February 12, 1809 – April 15, 1865) was the 16th President of the United States, serving from March 1861 until his assassination in April 1865. Lincoln led the United States through its Civil War—its bloodiest war and its greatest moral, constitutional and political crisis.[1][2] In doing so, he preserved the Union, abolished slavery, strengthened the federal government, and modernized the economy.

Lincoln grew up on the western frontier in Kentucky and Indiana. Largely self-educated, he became a lawyer in Illinois, a Whig Party leader, and a member of the Illinois House of Representatives, where he served from 1834 to 1846. Elected to the United States House of Representatives in 1846, Lincoln promoted rapid modernization of the economy through banks, tariffs, and railroads. Because he had originally agreed not to run for a second term in Congress, and his opposition to the Mexican–American War was unpopular among Illinois voters, Lincoln returned to Springfield and resumed his successful law practice. Reentering politics in 1854, he became a leader in building the new Republican Party, which had a statewide majority in Illinois. In 1858, while taking part in a series of highly publicized debates with his opponent and rival, Democrat Stephen A. Douglas, Lincoln spoke out against the expansion of slavery, but lost the U.S. Senate race to Douglas.

source: http://en.wikipedia.org/wiki/Abraham_Lincoln

How to do coreference resolution

Antecedent	Anaphor	Corefers?
Abraham Lincoln	He$_1$	yes
16th president of the United States	He$_1$	yes
Lincoln	His$_5$	yes
Stephen A. Douglas	He$_4$	no
Abraham Lincoln	Lincoln$_6$	yes
Member of the Illinois House of Representatives	He$_2$	yes

Lincoln grew up on the western frontier in Kentucky and Indiana. Largely self-educated, he$_1$ became a lawyer in Illinois, a Whig Party leader, and a member of the Illinois House of Representatives, where he$_2$ served from 1834 to 1846. Elected to the United States House of Representatives in 1846, Lincoln$_3$ promoted rapid modernization of the economy through banks, tariffs, and railroads. Because he$_4$ had originally agreed not to run for a second term in Congress, and his$_5$ opposition to the Mexican–American War was unpopular among Illinois voters, Lincoln$_6$ returned to Springfield and resumed his$_7$ successful law practice.

Why is coreference resolution important?

- Coreference is a frequently used natural language phenomenon
- Coreference resolution is essential to aggregate all knowledge and properties of the entities that a text makes reference to
- Coreference resolution is difficult because it stretches across sentences (syntax) and involves semantics and discourse

How to do coreference resolution (1)

Rule-based (Stanford multi-sieve, Lee eval 2013)

1. String matching

   - Will help you with proper names (*Smith* & *Smith*), common NPs (risky): *a man*, *another man*, *the man*
   - Partial matching is a problem (with/without titles?)
   - Fails on abbreviations and acronyms (and anything that doesn't use the same strings, e.g. *Lincoln, he*)

2. Agreement heuristics (anaphora must agree with their antecedents in name, gender and animacy)

3. Scoping (identify a text region where you expect to find the antecedent): Most recent matching subject is the most likely antecedent:

   - *John gave Bill a book. He ….*
   - *John gave Mary a book, She…*
   - *John gave Bill a book. Bill did not read it/ He did not read it/He asked him about the title*

How to do coreference resolution (2)

Machine Learning

- Annotated data marked up with co-reference chains
- Supervised technique to identify antecedents to anaphora
    - Clustering to merge pairwise coreference decisions into coreference chains
- Varied features used: part-of-speech tags, parse information, named entities, semantic class lookup, NP chunks, proximity, aliases, number, gender

Featured used for entity-coreference resolution

Type	Features
Mention	String match, part-of-speech, alias, number, gender (Soon, Ng, and Lim 2001), appositive, animacy, speaker (Lee et al. 2011), WordNet relation (Culotta, Wick, and McCallum 2007), modifier (Culotta, Wick, and McCallum 2007), overlap, quotation (Ng and Cardie 2002b), syntax subtree (Versley et al. 2008), dependency label (Björkelund and Nugues 2011), dependency path (Bergsma and Lin 2006), named-entity type (Denis and Baldridge 2009), semantic class (Soon, Ng, and Lim 2001), selectional preference (Dagan, Dagan, and Itai 1990), semantic roles (Ponzetto et al. 2006)
Textual context	Saliency (Lappin and Leass 1994), recency McCarthy (1996, pp. 87), narrative chain (Rahman and Ng 2012; Peng and Roth 2016)
Entity linking	Wikipedia (Ponzetto et al. 2006), Freebase attribute (Hajishirzi et al. 2013)

Table 1: A non-comprehensive list of features used in the literature. Each feature can be instantiated in many ways and sometimes one system contains more than one version.

Minh, Fokkens, Vossen, (under review), A Content-based Analysis of Coreference Resolution

Coreference performance

- CoNLL-2012 (Pradhan et al. 2012) standard benchmark in entity coreference resolution in recent years.:
    - 2,385 annotated English documents, totaling at 1.6M words, from various genres such as newswire, weblogs, and telephone conversations.
    - Highest reported result (after six years) is only 73% (Lee, He, and Zettlemoyer 2018).
- Some genres get much lower performance than others
    - Stanford Sieve (Lee et al. 2013) is lowest for newswire (55%) and highest for bible (67%).
    - The neural model of Clark and Manning (2016b) displays more than 10% difference between broadcast conversations (64%) and bible (78%).
- References:

    - Clark, Kevin and Christopher D. Manning. 2016a. Deep Reinforcement Learning for Mention-Ranking Coreference Models. Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP-16), pages 2256–2262.
    - Clark, Kevin and Christopher D. Manning. 2016b. Improving Coreference Resolution by Learning Entity-Level Distributed Representations. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pages 643–653.
    - Lee, Heeyoung, Angel Chang, Yves Peirsman, Nathanael Chambers, Mihai Surdeanu, and Dan Jurafsky. 2013. Deterministic Coreference Resolution Based on Entity-Centric, Precision-Ranked Rules. Computational Linguistics, 39(4):885–916.
    - Lee, Kenton, Luheng He, and Luke Zettlemoyer. 2018. Higher-order Coreference Resolution with Coarse-to-fine Inference. pages 687–692.
    - Pradhan, Sameer, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. 2012. CoNLL-2012 Shared Task: Modeling Multilingual Unrestricted Coreference in OntoNotes. EMNLP-CoNLL 2012.
