Information Extraction (1)

Zhiyuan Liu

liuzy@tsinghua.edu.cn

THUNLP

Outline

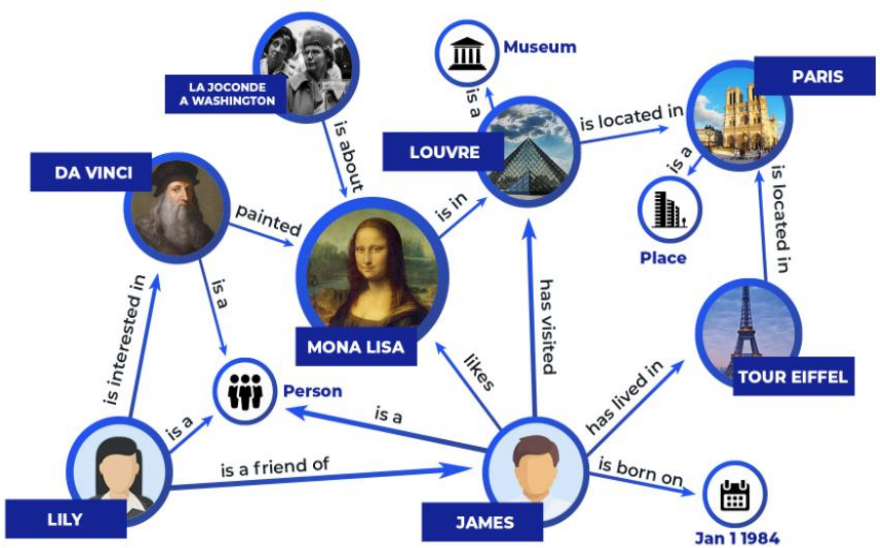
- Information Extraction (part1)
  - Information Extraction Architecture
  - Part-of-Speech Tagging
  - Sequence Labeling
  - Named Entity Recognition
  - Named Entity Typing
  - Entity Linking

Information Extraction Architecture

- Information Extraction (part1)
  - Information Extraction Architecture
  - Part-of-Speech Tagging
  - Sequence Labeling
  - Named Entity Recognition
  - Named Entity Typing
  - Entity Linking

Information Extraction Architecture

- Information source:
  - Structured data
  - Unstructured data




Mona Lisa

From Wikipedia, the free encyclopedia

This article is about the painting. For other uses,

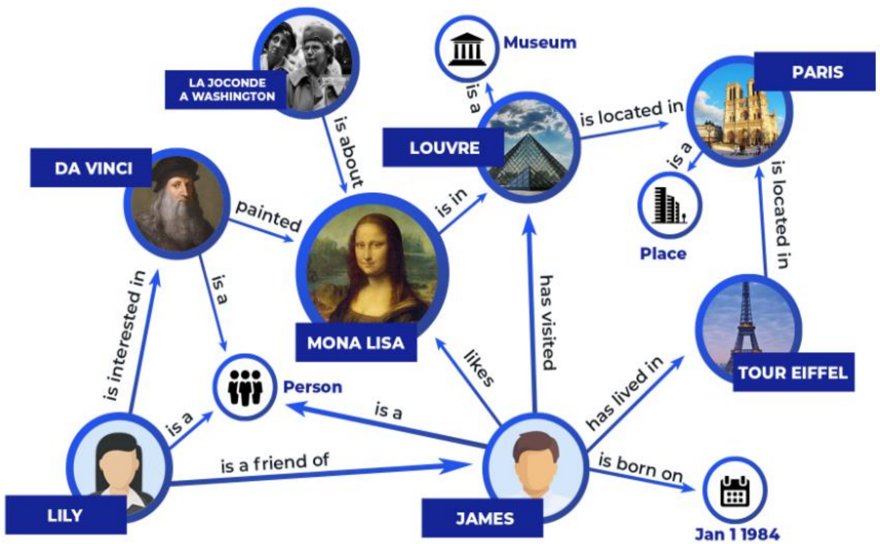
The ***Mona Lisa*** (/ˌmoʊnə ˈliːsə/; Italian: ***Monna Lisa*** portrait painting by the Italian artist Leonardo da Vi described as "the best known, the most visited, the painting's novel qualities include the subject's expre the subtle modelling of forms, and the atmospheric

Easy to store. Easy to use.

Ambiguous. Complex. Hard to use.

Information Extraction Architecture

- Information source:
  - Structured data
  - Unstructured data



Mona Lisa

From Wikipedia, the free encyclopedia

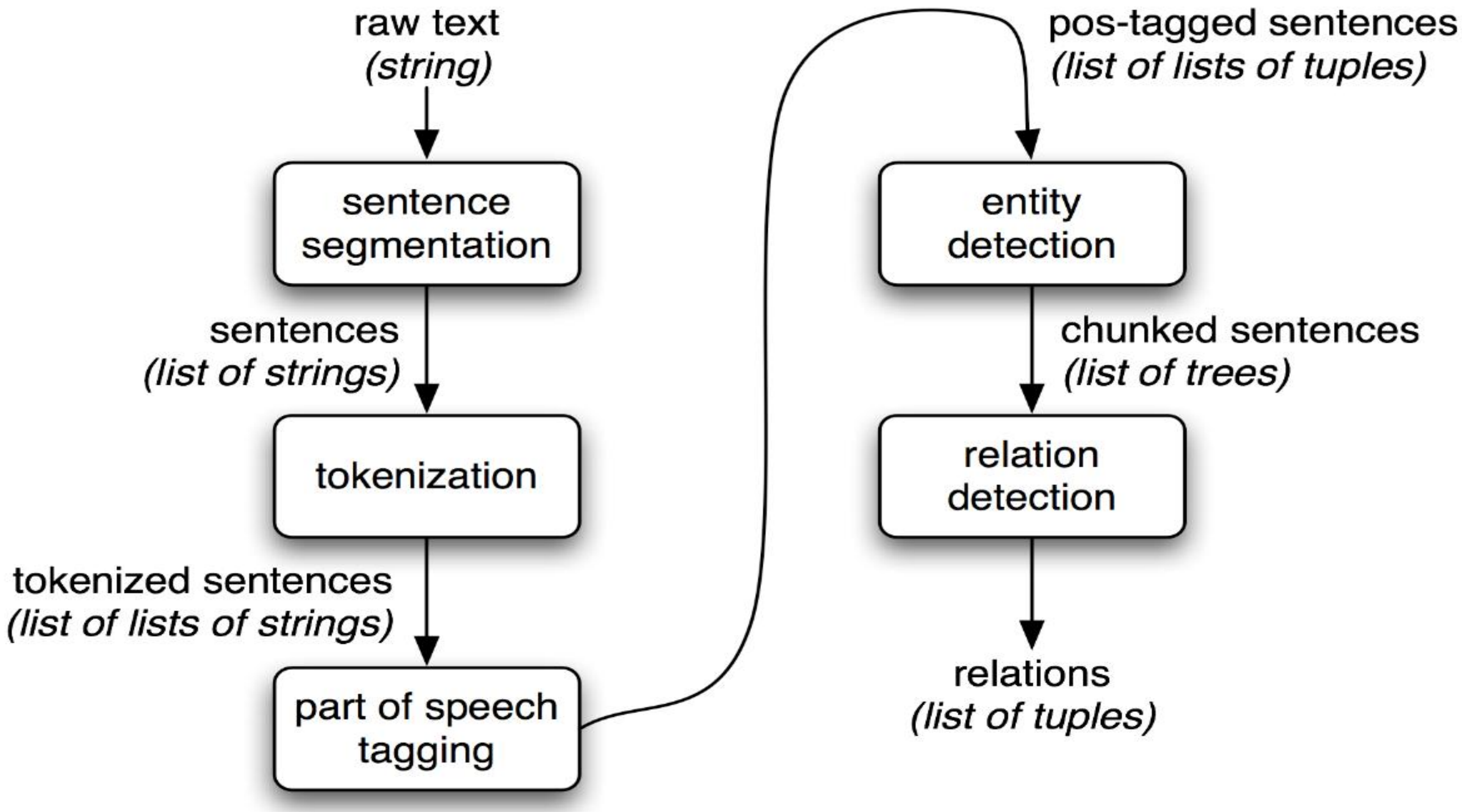
This article is about the painting. For other uses,

The ***Mona Lisa*** (/ˌmoʊnə ˈliːsə/; Italian: ***Monna Lisa***
portrait painting by the Italian artist Leonardo da Vi
described as "the best known, the most visited, the
painting's novel qualities include the subject's expre
the subtle modelling of forms, and the atmospheric

How?

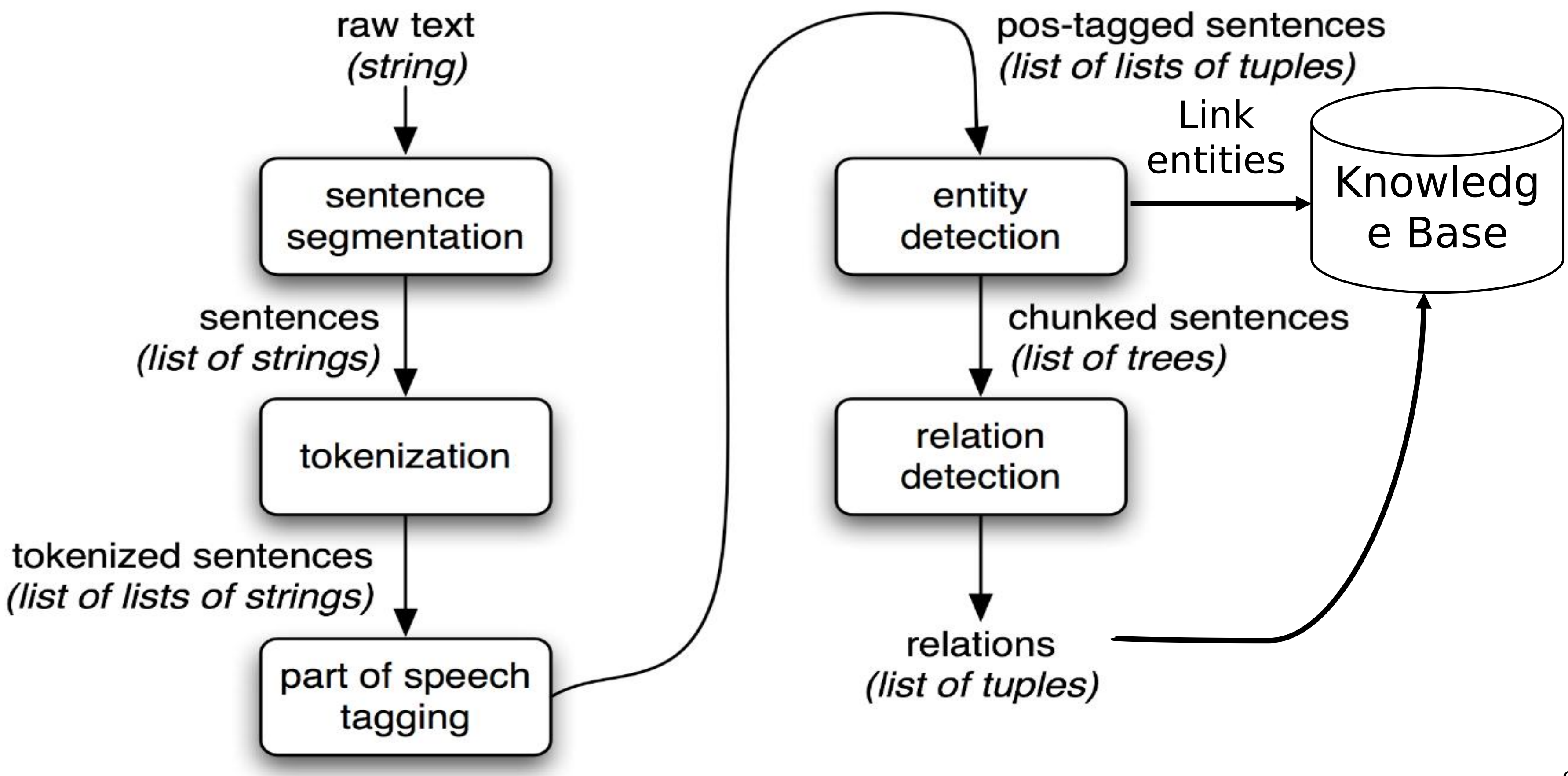
Information Extraction Architecture

- The general process of generating structured information from raw text.



Information Extraction Architecture

- The general process of generating structured information from raw text.



Part-of-Speech Tagging

- Information Extraction (part1)
  - Information Extraction Architecture
  - Part-of-Speech Tagging
  - Sequence Labeling
  - Named Entity Recognition
  - Named Entity Typing
  - Entity Linking

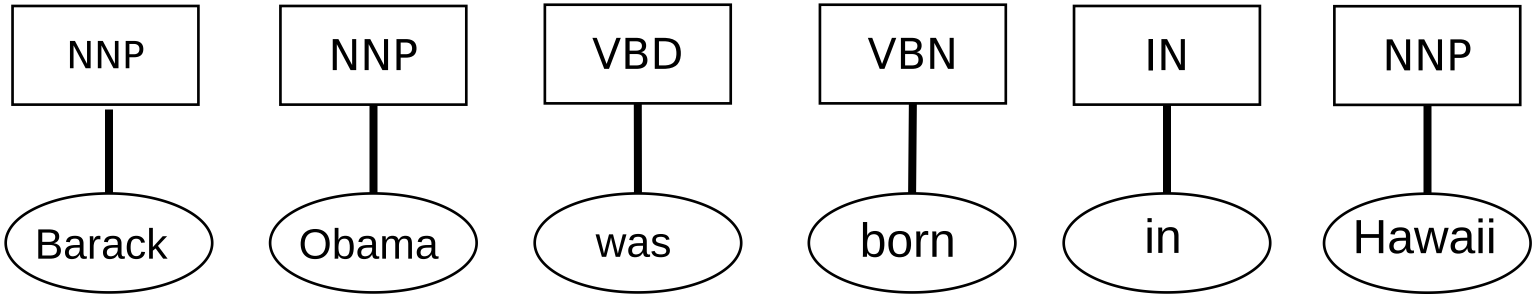
Part-of-Speech Tagging

- “Colorless green ideas sleep furiously”
  - -- Syntactically correct but semantically ill sentence.
- “I no like mathematical”
  - -- Semantically correct (have some meaning) but syntactically ill sentence.

- Part-of-Speech (POS) Tagging is helpful for machine to understand sentences syntactically.

Part-of-Speech Tagging

- Part-of-Speech: words (lexical items) that have similar grammatical properties.



Part-of-Speech Tagging

- Common datasets for POS-tagging include:
  - Brown Corpus.
  - Penn tag set .
    (Penn Treebank projects)

Part-of-Speech Tagging

- English Penn Treebank Tag set

Tag	Description	Example	Tag	Description	Example
CC	coordin. conjunction	*and, but, or*	SYM	symbol	+,%, &
CD	cardinal number	*one, two*	TO	“to”	*to*
DT	determiner	*a, the*	UH	interjection	*ah, oops*
EX	existential ‘there’	*there*	VB	verb base form	*eat*
FW	foreign word	*mea culpa*	VBD	verb past tense	*ate*
IN	preposition/sub-conj	*of, in, by*	VBG	verb gerund	*eating*
JJ	adjective	*yellow*	VBN	verb past participle	*eaten*
JJR	adj., comparative	*bigger*	VBP	verb non-3sg pres	*eat*
JJS	adj., superlative	*wildest*	VBZ	verb 3sg pres	*eats*
LS	list item marker	*1, 2, One*	WDT	wh-determiner	*which, that*
MD	modal	*can, should*	WP	wh-pronoun	*what, who*
NN	noun, sing. or mass	*llama*	WP$	possessive wh-	*whose*
NNS	noun, plural	*llamas*	WRB	wh-adverb	*how, where*
NNP	proper noun, sing.	*IBM*	$	dollar sign	$
NNPS	proper noun, plural	*Carolinas*	#	pound sign	#
PDT	predeterminer	*all, both*	“	left quote	‘ or “
POS	possessive ending	*’s*	”	right quote	’ or ”
PRP	personal pronoun	*I, you, he*	(	left parenthesis	[, (, {, <
PRP$	possessive pronoun	*your, one’s*	)	right parenthesis	],), }, >
RB	adverb	*quickly, never*	,	comma	,
RBR	adverb, comparative	*faster*	.	sentence-final punc	. ! ?
RBS	adverb, superlative	*fastest*	:	mid-sentence punc	: ; ... – -
RP	particle	*up, off*			

Part-of-Speech Tagging

- Why POS-Tagging?
  - Text-to-Speech

    "They refuse to permit us to obtain the refuse permit."
    - The two "refuse" pronounce differently because of their different POS.
  - Lemmatization

    Saw[v] → see , Saw[n] → saw

  - Helpful in Named Entity Recognition.

Part-of-Speech Tagging

- Ways to tag a corpus:
  - Use human labor:

    Penn tag set is painstakingly tagged by hand.

  - Using machine learning models of sequence labeling.

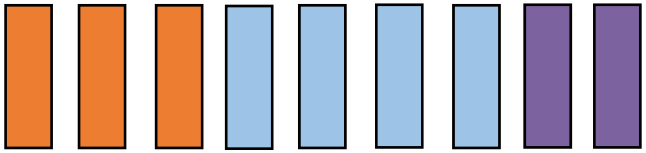
Sequence Labeling

- Information Extraction (part1)
  - Information Extraction Architecture
  - Part-of-Speech Tagging
  - Sequence Labeling
  - Named Entity Recognition
  - Named Entity Typing
  - Entity Linking

Sequence Labeling

- Sequence labeling problem

$y_{1:T} = 5 \quad 5 \quad 5 \quad 1 \quad 1 \quad 1 \quad 1 \quad 3 \quad 3 \quad \ldots$

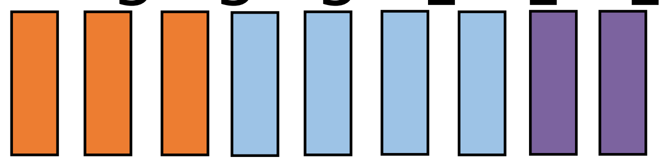


$x_{1:T} = a \quad b \quad a \quad c \quad d \quad e \quad e \quad f \quad f \quad \ldots$

Sequence Labeling

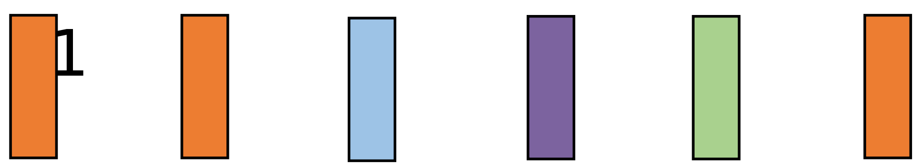
- Sequence labeling problem

$y_{1:T}$ = 5 5 5 1 1 1 1 3 3 …



$x_{1:T}$ = a b a c d e e f f …

$y_{1:T}$ = 1 1 3 2 4 1



$x_{1:T}$ = Barack Obama was born in Hawaii.

NNP : 1
VBN : 2
VBD : 3
IN : 4
…

Sequence Labeling

- Framework
  - Input: sequence of observations (feature vectors)
    $$x_{1:T} = (x_1, x_2, \ldots, x_T)$$
  - Output: sequence of labels (states)
    $$y_{1:T} = (y_1, y_2, \ldots, y_T)$$
  - The input feature set is $\{f_i\}$, the label set is $\{l_i\}$
  - In the POS-Tagging scenario, $\{f_i\}$ is the vocabulary set，$\{l_i\}$ is the tag set.
  - Label set：a finite set with discrete labels.
  - Feature set：continuous representation/ discrete token.

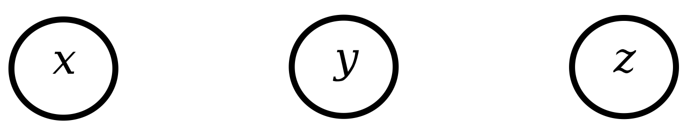
Sequence Labeling

- Classic Solutions:
  - Simple classification
  - Hidden Markov Model (HMM)
  - Conditional Random Fields (CRF)

- Can all be represented in Probabilistic Graphic Model (PGM).

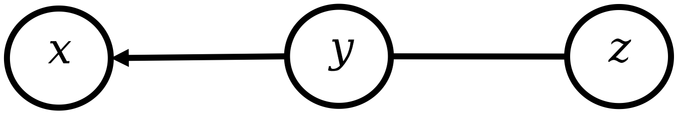
Sequence Labeling

- Brief introduction to Probabilistic Graphical Model.
  - Represent the dependency structure of random variables.
  - Each node (draw as a circle) is a random variable.



Sequence Labeling

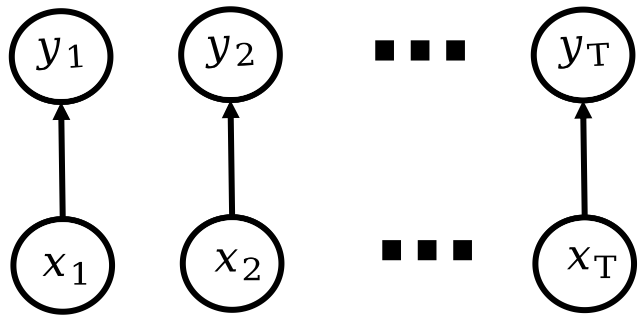
- Brief introduction to Probabilistic Graphical Model.
  - Edge between nodes is the dependency between the two random variables.
  - Undirected edge: modeling the joint probability over the two nodes.
  - Direct edges: modeling the conditional dependency between the HEAD node and the TAIL node.



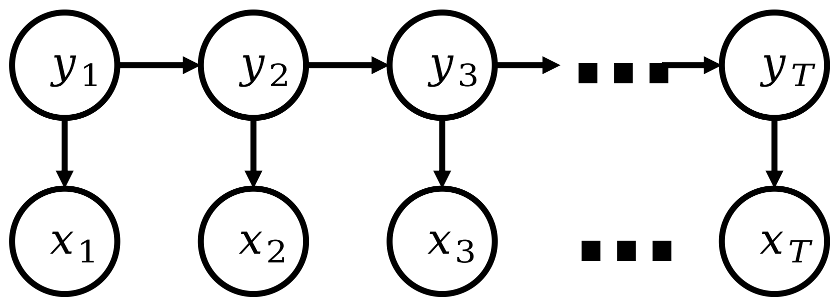
$$P(x = x_i, y = y_j, z = z_k) = P(x = x_i | y = y_j) P(y = y_j, z = z_k)$$

Sequence Labeling

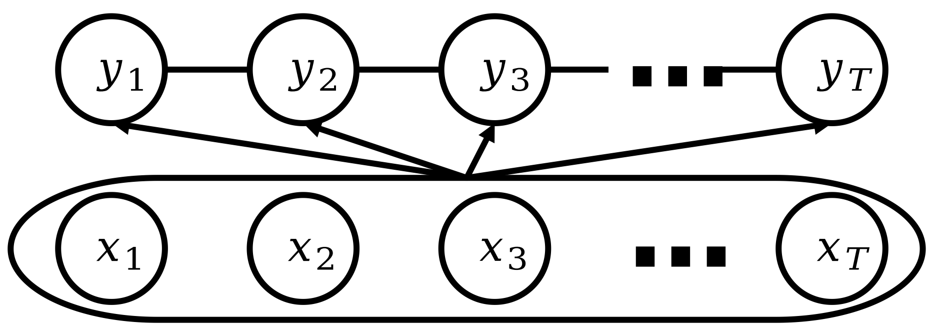
- Three classic solutions in PGM view.
  - Simple Classification:



  - Hidden Markov Model:

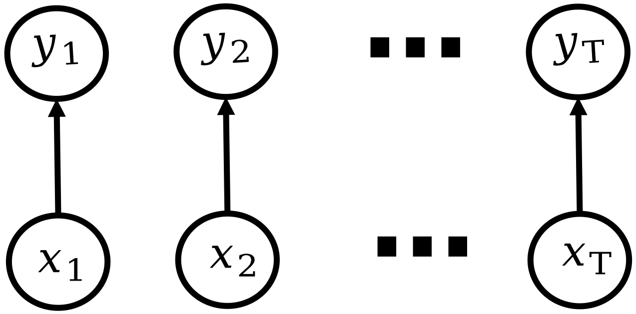


  - Conditional Random Fields



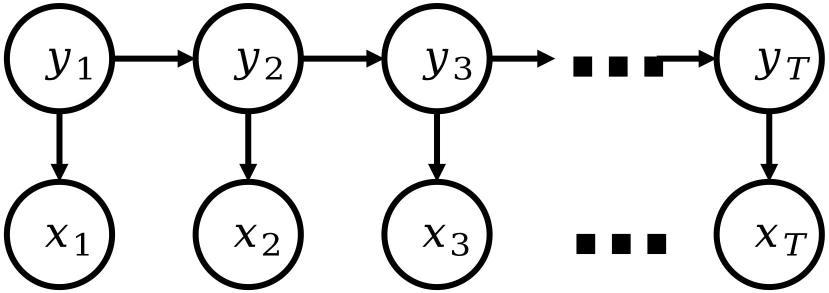
Sequence Labeling

- Simple classification:
  - Ignore the relations between neighboring states and do classification independently for each word.
  - Choose the most common tag.
  - Perform well sometimes.
    - Most words appear only in one POS in most sentences.
    - ~ 93%.
    - SOTA ~ 97+%.



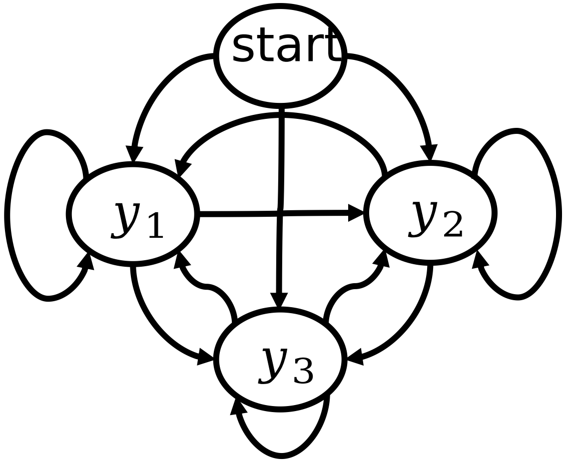
Sequence Labeling

- Hidden Markov Model
  - A generative model ($P(x, y), P(x|y)$).
  - Firstly, generate a sequence of y_t. Each y_t depends only on y_{t-1}. (Markov assumption) [transition]
  - Generate x_t based on the conditional probability $P(x_t|y_t)$. [emission]
  - The first statistical model of sequences applied to entity recognition.

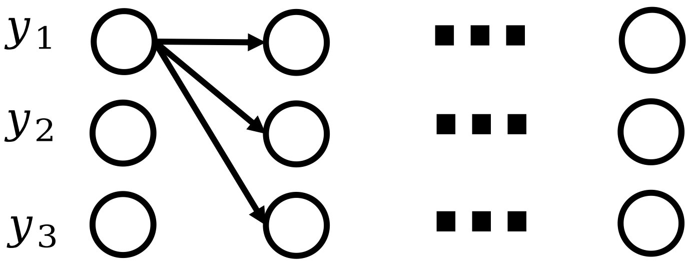


Sequence Labeling

- Hidden Markov Model
  - In state transition view



State Transition



Trellis/ lattice

Sequence Labeling

- Hidden Markov Model
  - Model the joint probability of label sequence and observed feature sequence: $P(x_{1:T}, y_{1:T})$

Transition prob Generation prob

$$P(x_{1:T}, y_{1:T}) = \Pi_{t=1}^{T} P(y_t | y_{t-1}) P(x_t | y_t)$$

Sequence Labeling

- Hidden Markov Model
  - Model the joint probability of label sequence and observed feature sequence: $P(x_{1:T}, y_{1:T})$

Transition prob Generation prob

$$P(x_{1:T}, y_{1:T}) = \Pi_{t=1}^{T} P(y_t | y_{t-1}) P(x_t | y_t)$$

  - Parameters in the model (denote by λ):
    - Transition matrix $\boldsymbol{P} = (P(y_t = l_i | y_{t-1} = l_j))$
    - Emission probability $\boldsymbol{E} = (P(x = x_i | y = y_j))$
    - Together $\lambda = \{\boldsymbol{P}, \boldsymbol{E}\}$

Sequence Labeling

- Hidden Markov Model
  - Learn parameters (supervised learning setting)
  - Training objective: Maximize $P(x_{1:T}|y_{1:T},\lambda)$
  - The maximum likelihood is reached when

$$P(y_t = l_i|y_{t-1} = l_j) = \#(l_j, l_i)/\#(l_j)$$
$$P(x_t = f_i|y_t = l_j) = \#(l_j \to f_i)/\#(l_j)$$

$\#(l_j, l_i)$: number of label l_j followed by label l_i).

$\#(l_j \to f_i)$:number of observed feature f_i generated by label l_j

  - Reduce to simple counting!

Sequence Labeling

- Hidden Markov Model
  - Learn parameters (unsupervised learning setting)
  - Training objective: Maximize $P(x_{1:T} | \cancel{y_{1:T}}, \lambda)$

$$P(x_{1:T} | \lambda) = \sum_{y_{1:T}} P(x_{1:T}, y_{1:T} | \lambda)$$

$$= \sum_{y_{1:T}} \prod_{t=1}^{T} P(y_t | y_{t-1}, \lambda) P(x_t | y_t, \lambda)$$

  - Sum over all possible $y_{1:T}$.
  - $O(N^T)$

Sequence Labeling

- Hidden Markov Model
  - Learn parameters (unsupervised learning setting)
  - Training objective: Maximize $P(x_{1:T}|\cancel{y_{1:T}}, \lambda)$

$$P(x_{1:T}|\lambda) = \sum_{y_{1:T}} P(x_{1:T}, y_{1:T}|\lambda)$$

$$= \sum_{y_{1:T}} \prod_{t=1}^{T} P(y_t|y_{t-1}, \lambda) P(x_t|y_t, \lambda)$$

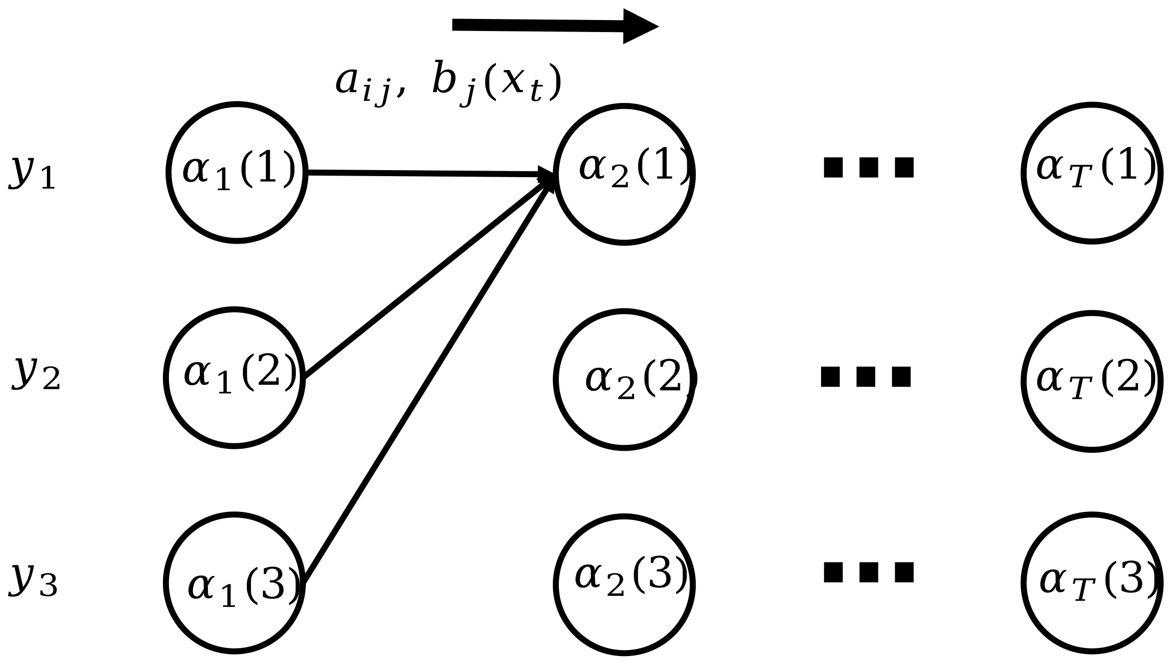
  - Forward Algorithm:
    - $\alpha_t(j) = P(x_{1:t}, y_t = j|\lambda)$
    - $P(x_{1:T}|\lambda) = \sum_{i=1}^{N} \alpha_T(i)$
    - $\alpha_t(j) = \sum_{i=1}^{N} \alpha_{t-1}(i) a_{ij} b_j(x_t)$

      Transition prob

      Generation prob
    - Use dynamic programming

Sequence Labeling

- Hidden Markov Model
  - Forward Algorithm:
    - Use dynamic programming: store $a_t(j)$



Sequence Labeling

- Hidden Markov Model
  - Learn parameters (unsupervised learning setting)
  - Training objective: Maximize $P(x_{1:T}|\cancel{y_{1:T}}, \lambda)$

$$P(x_{1:T}|\lambda) = \sum_{y_{1:T}} P(x_{1:T}, y_{1:T}|\lambda)$$

$$= \sum_{y_{1:T}} \prod_{t=1}^{T} P(y_t|y_{t-1}, \lambda) P(x_t|y_t, \lambda)$$

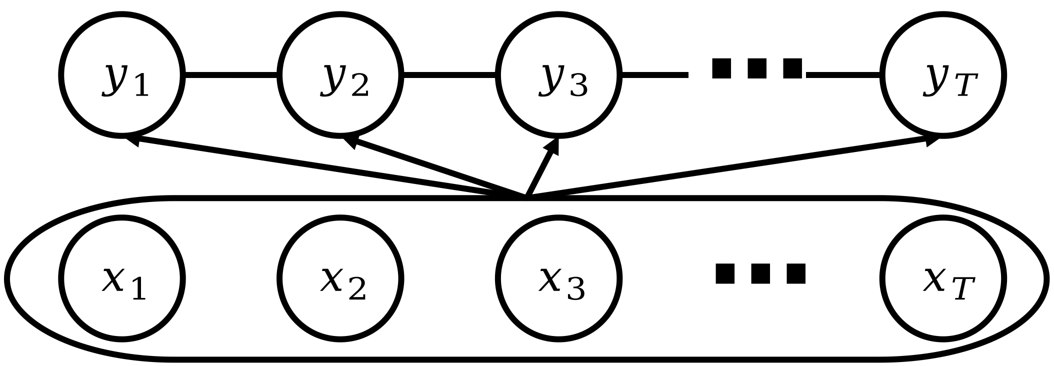
  - Optimize using E-M algorithm (Forward-Backward algorithm), basic idea:
    - E-step: use λ_{old} to get the probability of $y_{1:T}$: $P(y_{1:T}|x_{1:T}, \lambda_{old})$.
    - M-step: find the optimal λ of generating $x_{1:T}$ using $y_{1:T}$.

Sequence Labeling

- Hidden Markov Model
  - Infer (decode) with HMM.
    - compute the most probable label sequence by
      $$y^* = \arg\max_{y_1,\ldots,y_n} \Pi_{t=1}^{T} P(y_t|y_{t-1}) P(x_t|y_t)$$
    - $y_t^* = P(y_t|y_{t-1}) P(x_t|y_t) \times y_{t-1}^*$
    - Efficient algorithm for finding the maximum value route is Dynamic Programing.

Sequence Labeling

- (Linear Chain) Conditional Random Fields:
  - Discriminative model ($P(y|x)$). Doesn't consider the generating process of the observed data.
  - Directly modeling conditional probability

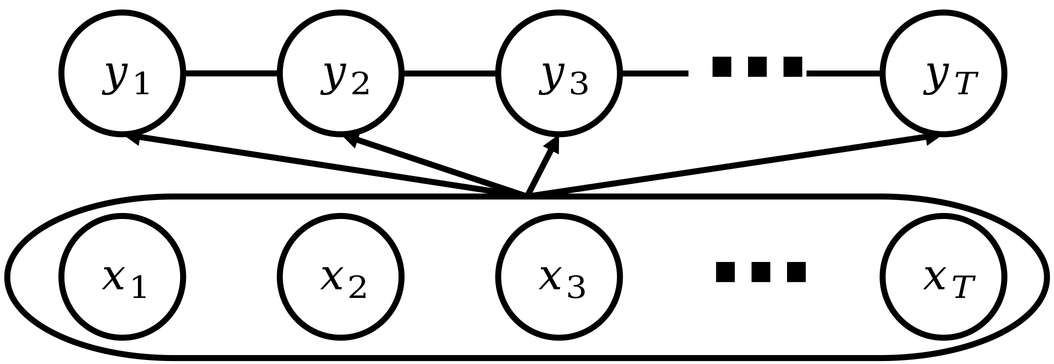


$$P(y_{1:T}|x_{1:T}) = \frac{P(x_{1:T}, y_{1:T})}{P(x_{1:T})} = \frac{P(x_{1:T}, y_{1:T})}{\sum_{y_{1:T}} P(x_{1:T}, y_{1:T})}$$

$$= \frac{1}{Z}\exp\left(S(x_{1:T}, y_{1:T})\right)$$

Sequence Labeling

- (Linear Chain) Conditional Random Fields:
  - The dependency structure in CRF.



  - So the joint probability of the whole sequence & label is:

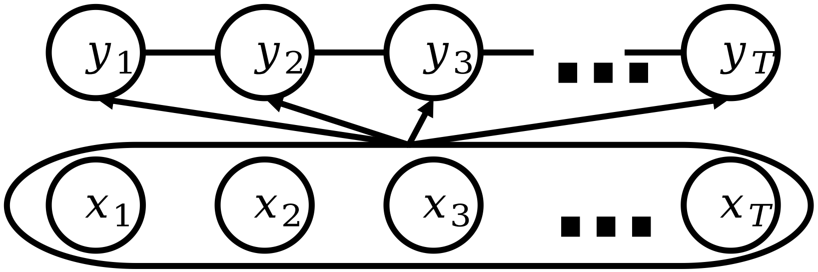
$$P(x_{1:T}) \times P(y_1, y_2 | x_{1:T}) \times \ \dots \times P(y_{T-1}, y_T | x_{1:T})$$

Sequence Labeling

- (Linear Chain) Conditional Random Fields:
  - The joint probability is:

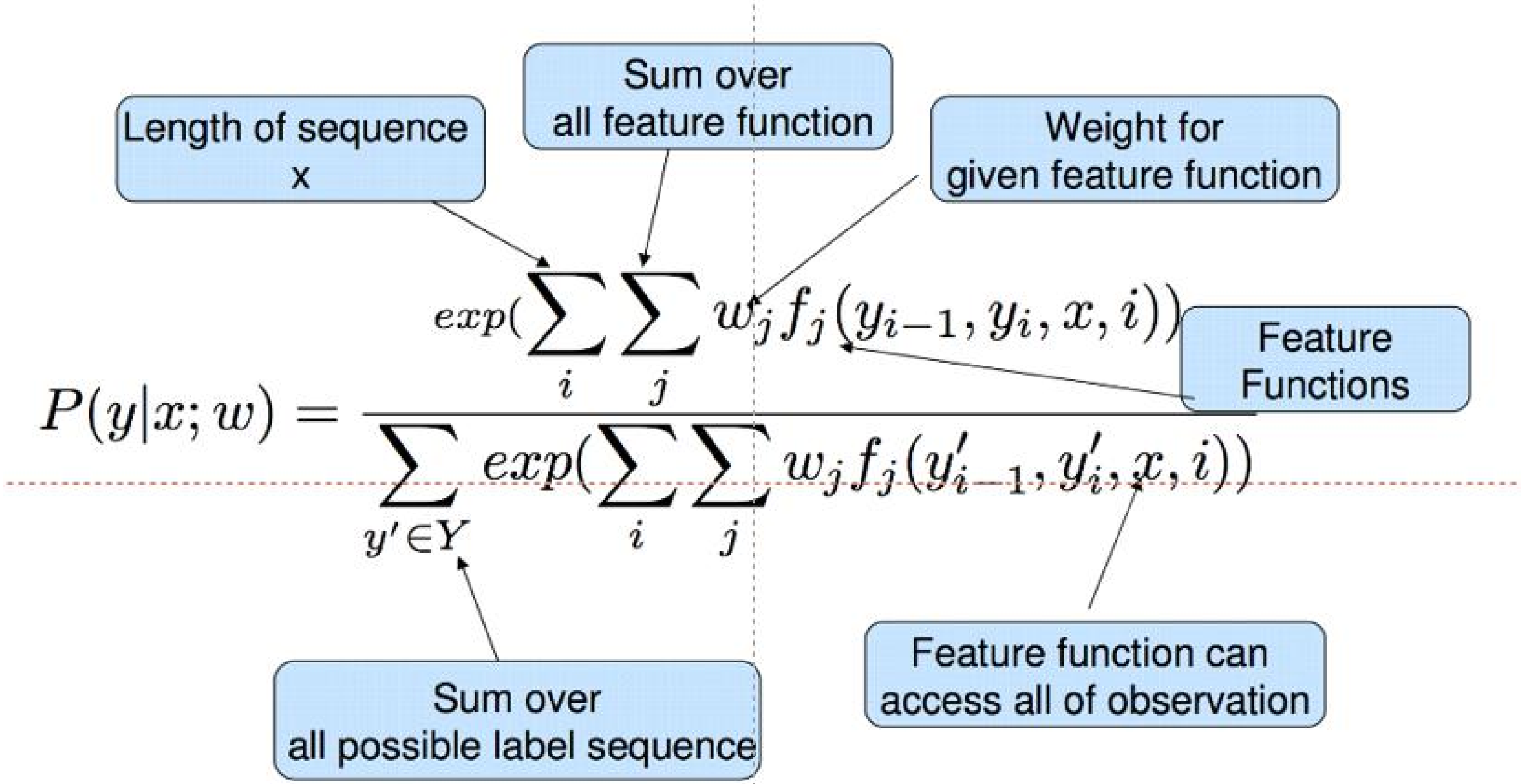
$$
\begin{aligned}
\mathrm{P}(y_{1:T}|x_{1:T}) &= \frac{1}{Z}\exp\left(S(x_{1:T}, y_{1:T})\right) \\
&= \frac{1}{\mathrm{Z}}\exp\left(\sum_{t=1}^{T} wf(y_t, y_{t-1}, x_{1:T}, t)\right) \\
&= \frac{1}{\mathrm{Z}}\exp\left(\sum_{t=1}^{T}\sum_{k} w_k f_k(y_t, y_{t-1}, x_{1:T}, t)\right)
\end{aligned}
$$

  - The combination of the random variables obeys the structure of CRF and forms the feature f_k.
  - Each feature f_k is assigned a weight w_k.



Sequence Labeling

- (Linear Chain) Conditional Random Fields:



Linear-Chain Conditional Random Field.
(taken from Sameer Maskey slides)

Sequence Labeling

- (Linear Chain) Conditional Random Fields:
  - Optimize (estimate the parameters):
  - Find the parameters w_i that best fit the training data, when given a set of labeled sentences:

$$\{(x_{1,1:T}, y_{1,1:T}), (x_{2,1:T}, y_{2,1:T}), \ldots (x_{m,1:T}, y_{m,1:T})\}$$

Sequence Labeling

- (Linear Chain) Conditional Random Fields:
  - Optimize (estimate the parameters):
    - Negative Log-Likelihood:

$$L(X, y, \{w_{\mathrm{k}}\}) = \sum_{i=1}^{m} -\log \mathrm{P}(y_{i,1:T} | x_{i,1:T}, \{w_{\mathrm{i}}\})$$

  - Take derivative of $\{w_k\}$, and perform gradient descent.

Sequence Labeling

- (Linear Chain) Conditional Random Fields:
  - Usually we add a regularization term to the trainable parameters:

$$L(X, y, \{\mathrm{w_i}\}) = \sum_{i=1}^{m} -\log \mathrm{P}(y_{i,1:T} | x_{i,1:T}) + \frac{\lambda}{2} \|w\|_2^2$$

Big weights are bad!

Sequence Labeling

- POS-Tagging using CRF
  - Specify the feature function.

$$L(y|x) = \frac{1}{Z}\exp\sum_{t=1}^{T}\sum_{k} w_k f_k(y_t, y_{t-1}, x_{1:T}, t)$$

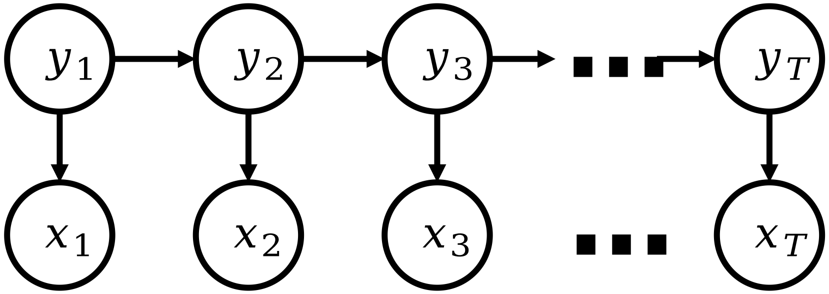
  - Commonly used features:
    - Word suffix and prefix of $x_{t-2}, x_{t-1}, x_t, x_{t+1}, x_{t+2} \dots$
    - Word length (function word~3.13, content word~6.47)
    - Contain digits or symbols?
    - In noun/adjective/verb lexicon? (if available)

Sequence Labeling

- Comparison between HMM, CRF

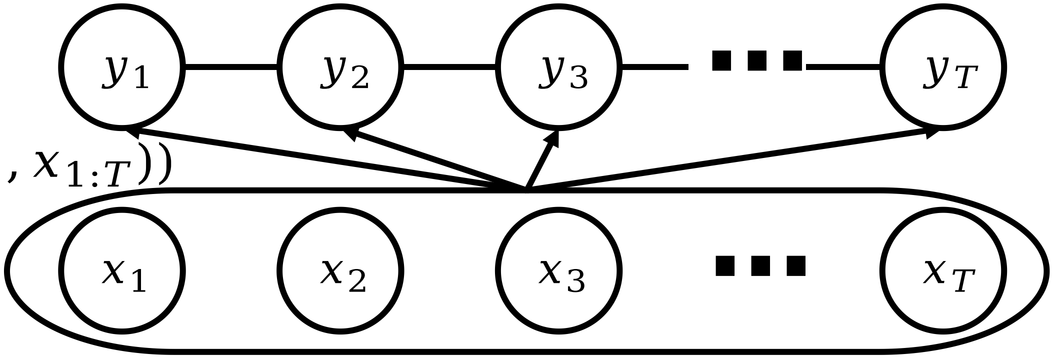
HMM

$$P(x_{1:T}, y_{1:T}) = \prod_{t=1}^{T} P(y_t | y_{t-1}) P(x_t | y_t)$$



CRF

$$P(y_{1:T} | x_{1:T}) = \frac{1}{z} \exp\left(\sum_{t=1}^{T} \sum_k w_k f(y_k, y_{k-1}, x_{1:T})\right)$$



Sequence Labeling

- A table of open-source toolkits for CRF.

Name	Description
python-crfsuite	is a python binding for CRFsuite which is a fast implementation of Conditional Random Fields written in C++.
CRF++: Yet Another CRF toolkit	is a popular implementation in C++ but there are no python bindings.
MALLET	includes implementations of widely used sequence algorithms including hidden Markov models (HMMs) and linear chain conditional random fields (CRFs), it's written in Java.
FlexCRFs	supports both first-order and second-order Markov CRFs, it's written in C/C++ using STL library.
python-wapiti	is a python wrapper for wapiti, a sequence labeling tool with support for maxent models, maximum entropy Markov models and linear-chain CRF.

Sequence Labeling

- Application of Sequence Labeling framework
  - POS-Tagging
  - Named Entity Recognition
  - Named Entity Typing
  - Speech Recognition
  - Handwriting Recognition
  - Video Analysis
  - Protein Secondary Structure Prediction

Named Entity Recognition

- Information Extraction (part1)
  - General Framework
  - Part-of-Speech Tagging
  - Sequence Labeling
  - Named Entity Recognition
  - Named Entity Typing
  - Entity Linking

Named Entity Tasks

- Named Entity:
  - A real-world object that can be denoted with a proper name.
  - E.g.

  Person names, organizations, locations,

  time expressions, quantities, medical codes,

  etc.
  - In a text: Entity mention.

Named Entity Tasks

- Three tasks about Named Entity:
  - Named Entity Recognition (NER)
    - Sequence labeling
  - Named Entity Typing (NET)
    - Classification
  - Entity Linking (EL)
    - Classification/matching

Named Entity Tasks

- Three tasks about Named Entity:
  - Named Entity Recognition (NER)
    - Locate and classify named entities by predicting a label for each word in text.

Unstructured Text Barack Obama was born in Hawaii in 1961.

Named Entity Recognition

Barack Obama PER was born in LOC Hawaii TIME
in 1961 .

Named Entity Tasks

- Three tasks about Named Entity:
  - Named Entity Typing (NET)
    - Label an entity mention in a text with detailed type
    - Different from NER:
      - Provide the boundary of entity mentions
      - Classify the entities into finer-grained classes

Named Entity Typing

Barack Obama was born in Hawaii in 1961

President/Husband

state/Island

Year

Named Entity Tasks

- Three tasks about Named Entity:
  - Entity Linking (EL)
    - Link mentions in text to their corresponding entities in a knowledge base.

Barack Obama was born in Hawaii in 1961 .

https://en.wikipedia.org/wiki/Barack_Obama
https://en.wikipedia.org/wiki/Hawaii

Named Entity Recognition

- Information Extraction (part1)
  - General Framework
  - Part-of-Speech Tagging
  - Sequence Labeling
  - Named Entity Recognition
  - Named Entity Typing
  - Entity Linking

Named Entity Tasks

- Named Entity Recognition:
  - Difficulty: not always single words.

  E.g.

  New York City, Tsinghua University,
  - Compositionality.

    E.g.

    Barack Obama, Romeo and Juliet

Named Entity Recognition

- Named Entity Recognition:
  - Annotation schemes for NER:
    - IOB
    - BIOES

Named Entity Recognition

- IOB
  - I: Inside an entity, but not the first entity
  - O: Outside an entity
  - B: Beginning of an entity

Barack	B-PER
Hussein	I-PER
Obama	I-PER
was	O
born	O
in	O
Hawaii	B-LOC

Named Entity Recognition

- BIOES: the most widely used scheme
  - B: Beginning of an entity
  - I: Inside an entity
  - O: Outside an entity
  - E: End of an entity
  - S: Single-word entity

Barack	B-PER
Hussein	I-PER
Obama	E-PER
was	O
born	O
in	O
Hawaii	S-LOC

Named Entity Recognition

- Another sequence labeling problem.

$y_{1:T}$ = 1 3 4 4
4 5

$x_{1:T}$ = Barack Obama was born
in Hawaii.

B-PER	: 1
I-PER	: 2
E-PER	: 3
O	: 4
S-LOC	: 5

- Popular methods:
  - CRF
  - Deep Neural Network combined with CRF

Named Entity Recognition

- CRF for NER
  - Specify the feature function.

$$L(y|x) = \frac{1}{Z}\exp\sum_{t=1}^{T}\boxed{\sum_k w_k f_k(y_t, y_{t-1}}, x_{1:T}, t)$$

  - Commonly used features:
    - Word feature—orthographical features of the (-2,-1,0,1,2) words.
    - POS tag —part-of-speech tag of the (-2,-1,0,1,2) words.

Named Entity Recognition

- Build CRF on top of Bi-LSTM.
  - Define the energy function as:

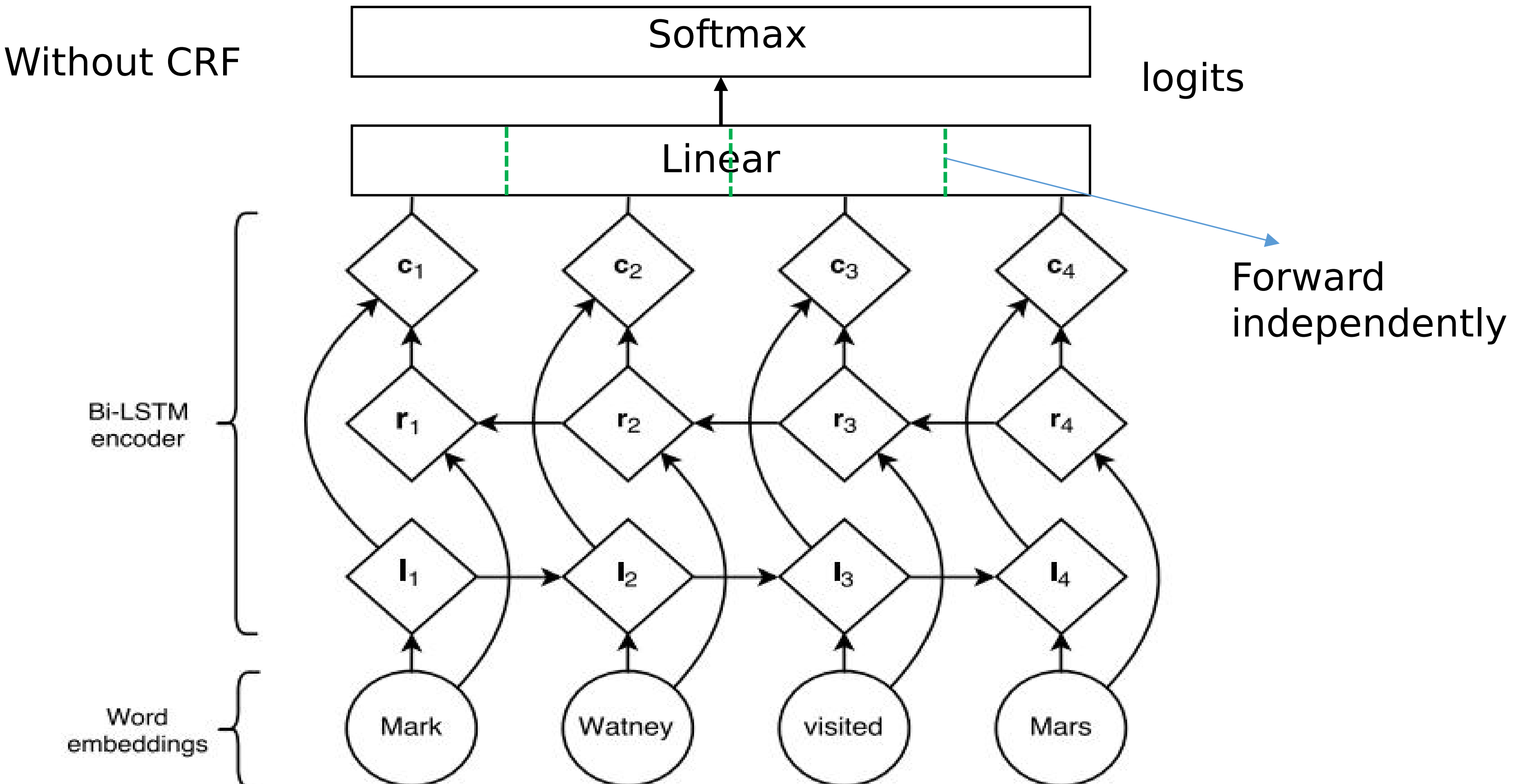
$$S(x_{1:T}, y_{1:T}) = \sum_{i=0}^{n} A_{y_i, y_{i+1}} + \sum_{i=1}^{n} P_{i, y_i}$$

  - $A_{y_i, y_{i+1}}$: the transition probability between tags.
  - P_{i, y_i} : logits output from Bi-LSTM, followed by a Softmax operator.

Neural Architectures for Named Entity Recognition. NAACL 2016.

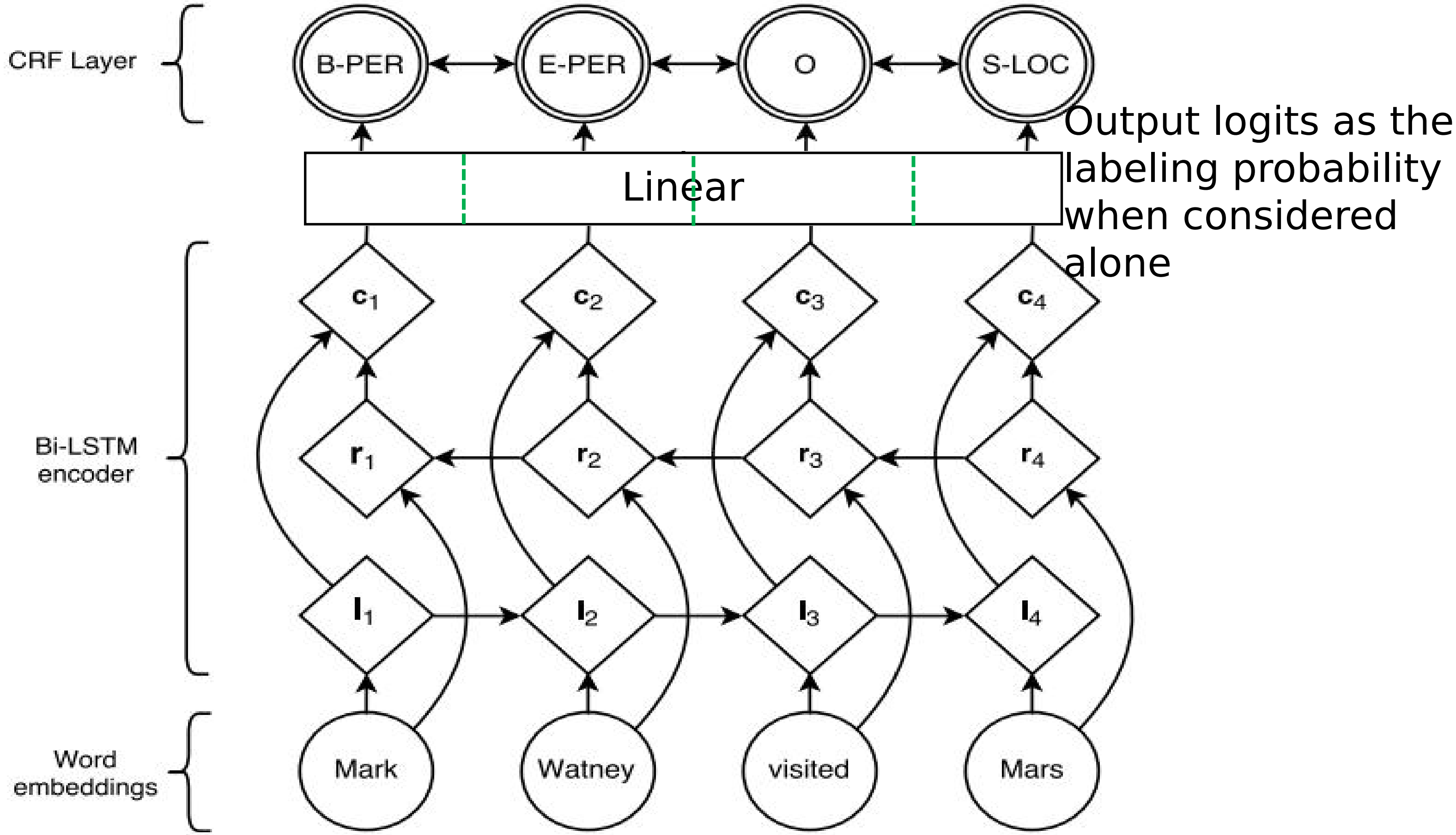
Named Entity Recognition

- Build CRF on top of Bi-LSTM.



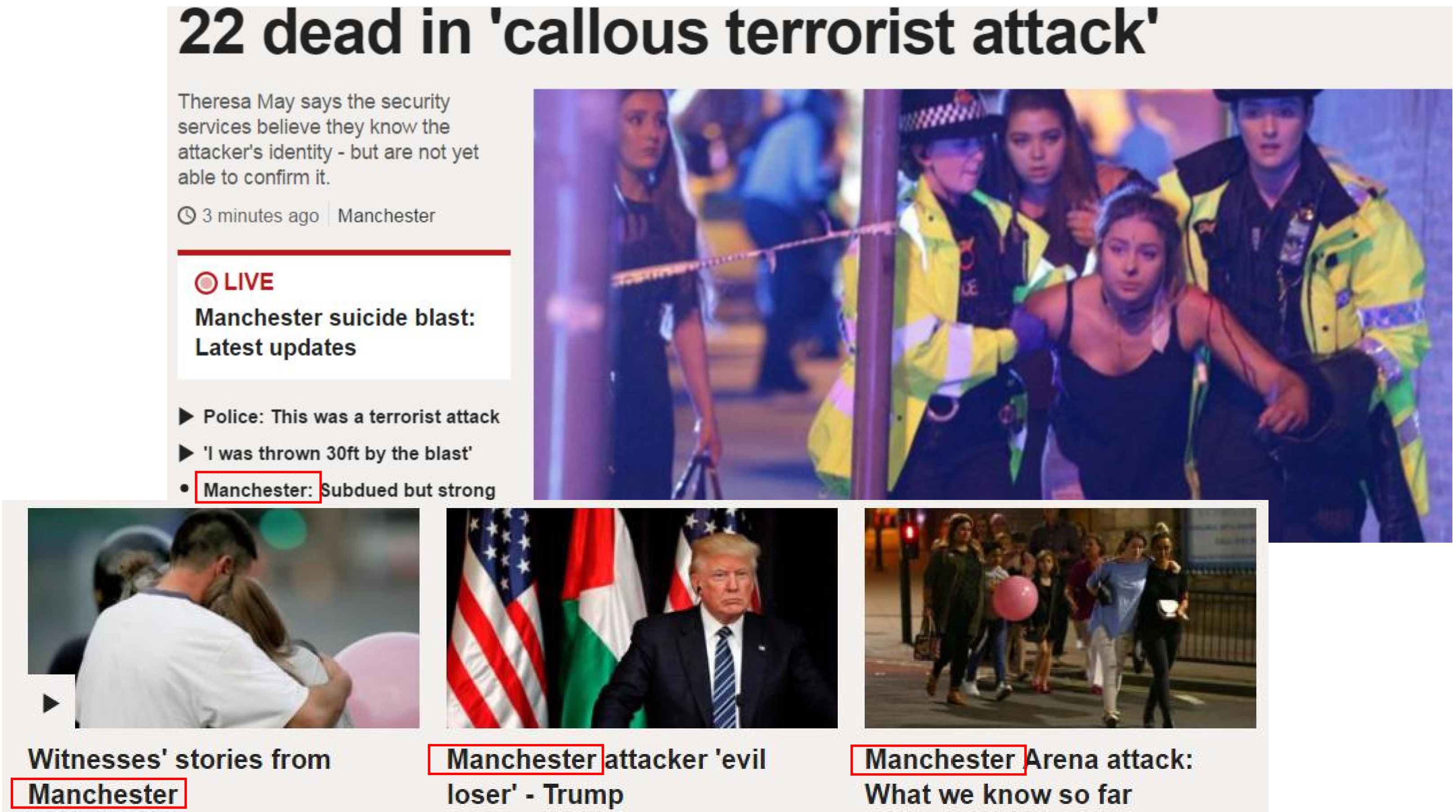
Named Entity Recognition

- Build CRF on top of Bi-LSTM.



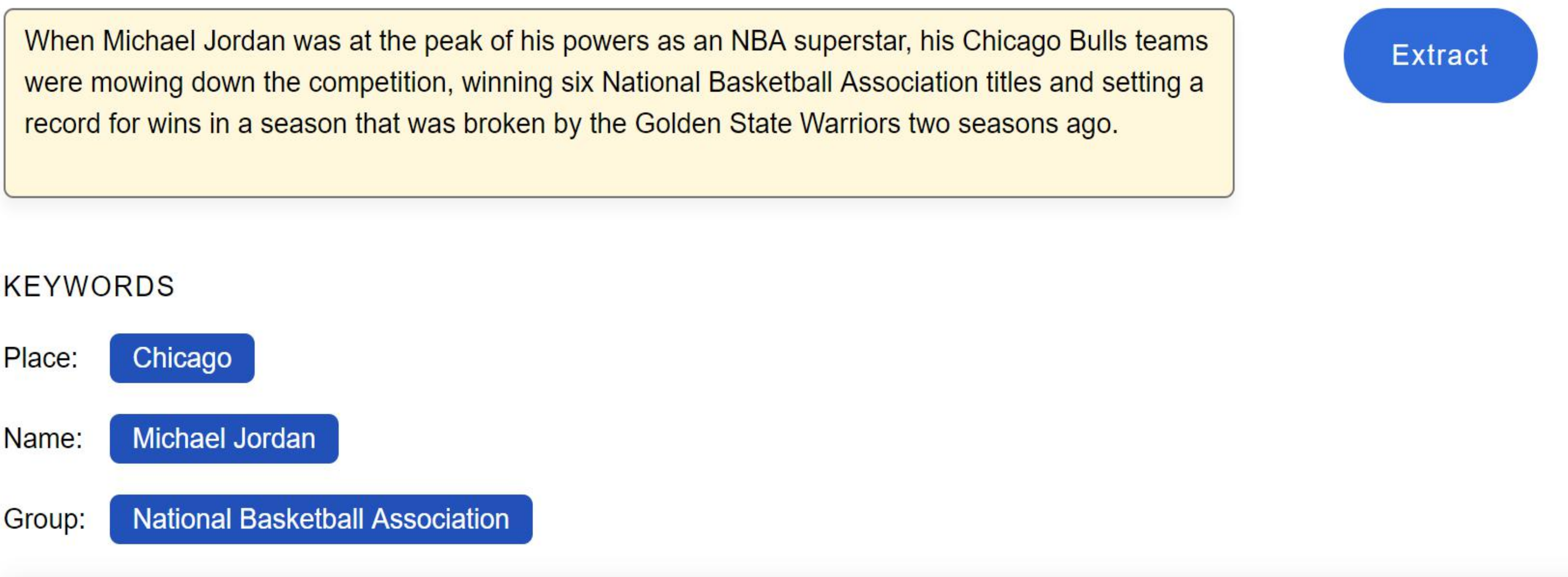
Named Entity Recognition

- Application of NER
  - Recommendation system.



Named Entity Recognition

- Application of NER
  - News classification. (+ Named Entity Typing)



Named Entity Recognition

- Application of NER
  - Efficient searching.
    Speed up matching when the content is related to some named entity.

Named Entity Recognition

- Some Demo websites for NER

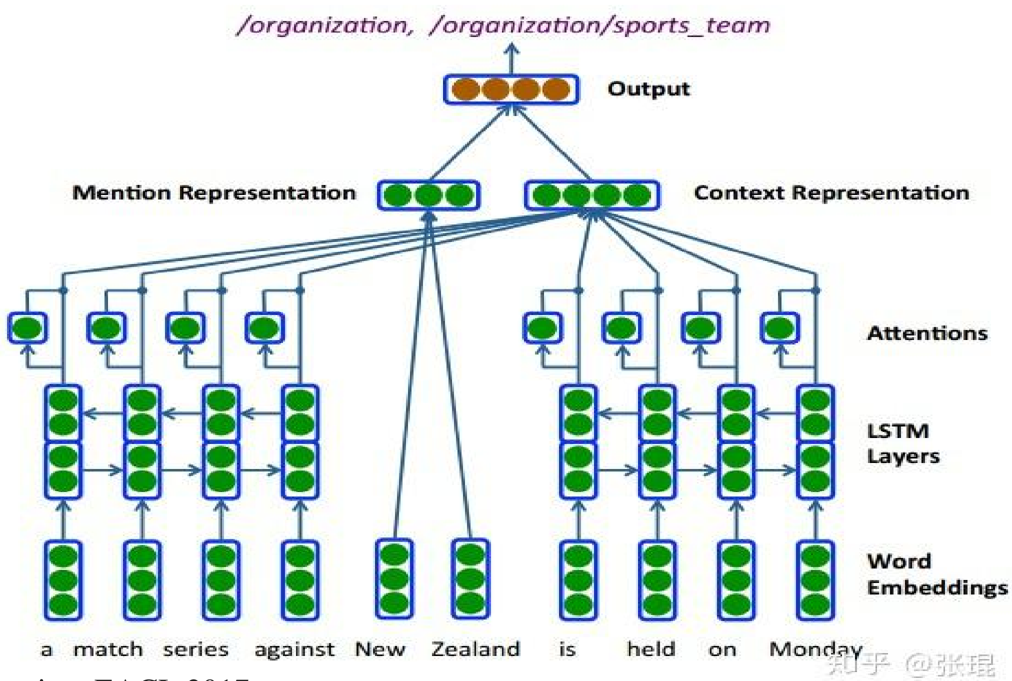
Name	Link
Allennlp	https://demo.allennlp.org/named-entity-recognition
StanfordNLP	https://nlp.stanford.edu/software/CRF-NER.html

Named Entity Typing

- Information Extraction (part1)
  - General Framework
  - Part-of-Speech Tagging
  - Sequence Labeling
  - Named Entity Recognition
  - Named Entity Typing
  - Entity Linking

Named Entity Typing

- A simple approach
  - Use LSTM to encode the entity’s context.
  - Use average word embedding as the entity embedding..



Neural Architectures for Fine-grained Entity Type Classification. EACL 2017.

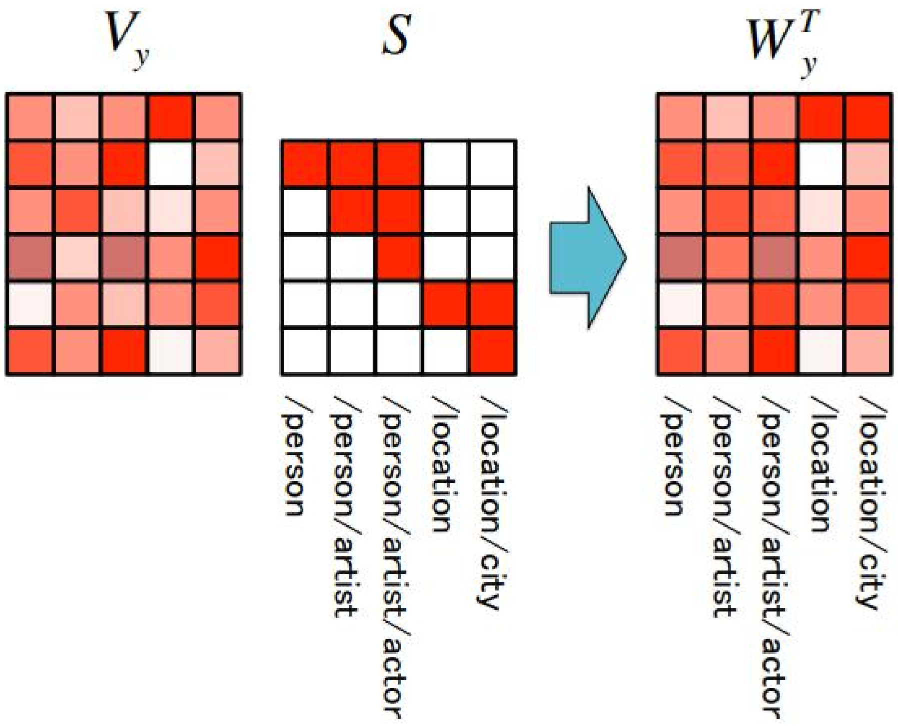
Named Entity Typing

- Multiple/Hierarchical type prediction
  - In real world, an entity often belongs to multiple types.
  - E.g.
    - *I went to Chicago this weekend*.

      Chicago can be labeled as “location” and “city”
    - *person/artist/actor*
  - Predicting the coarse-grained types may be helpful to the fine-grained types and vise versa.

Named Entity Typing

- Multiple/Hierarchical type prediction
  - Hierarchical label encoding



Neural Architectures for Fine-grained Entity Type Classification. EACL 2017.

Named Entity Typing

- Some further improvement
  - Add document-level context for entity mentions.
  - Better handle type hierarchy chains:

    Types in different granularities should be assigned with different weights.

$$P'(y|m,c) = P(y|m,c) + \beta \sum_{t \in \Gamma} P(t|m,c)$$

Named Entity Typing

- Relationship between NER and NET
  - Now: Coupled.
  - As suggested in a survey paper:
    - Considering NER as a task
      - Dedicated in detecting named entity’s boundary
      - Without considering entity typing.
    - Advantages: a more robust solution which can be shared across different domains.

A survey on deep learning for named entity recognition. 2019.

Entity Linking

- Information Extraction (part1)
  - General Framework
  - Part-of-Speech Tagging
  - Sequence Labeling
  - Named Entity Recognition
  - Named Entity Typing
  - Entity Linking

Entity Linking

- Why Entity Linking?
  - Enrich existing knowledge bases using new facts from text.
  - It can also be used in information retrieval

Entity Linking

- Challenges in Entity Linking
  - Name variations: A named entity may have multiple names.
  - E.g.

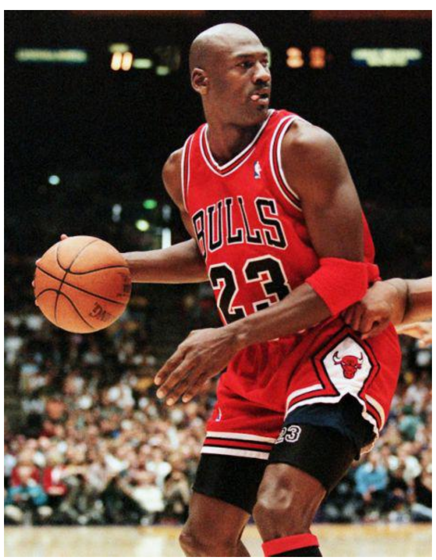
New York City = Big Apple = NYC

Tsinghua University = Tsinghua = THU

Entity Linking

- Challenges in Entity Linking
  - Entity ambiguity (more important)

  Q: What is the birthdate of the famous basketball player Michael Jordan?



OR

Entity Linking

- Challenges in Entity Linking
  - Entity ambiguity (more important)

Q: What is the birthdate of the famous basketball player Michael Jordan?

Michael Jordan (disambiguation)

From Wikipedia, the free encyclopedia

Michael Jordan (born 1963), American basketball player and businessman

Michael Jordan or **Mike Jordan** may also refer to:

People [edit]

Sports [edit]

- Michael Jordan (footballer) (born 1986), English goalkeeper
- Mike Jordan (racing driver) (born 1958), English racing driver
- Mike Jordan (baseball, born 1863) (1863–1940), baseball player
- Mike Jordan (cornerback) (born 1992), American football cornerback
- Michael Jordan (offensive lineman), American football offensive lineman
- Michael-Hakim Jordan (born 1977), American professional basketball player
- Michal Jordán (born 1990), Czech ice hockey player

Other people [edit]

- Michael B. Jordan (born 1987), American actor
- Michael I. Jordan (born 1956), American researcher in machine learning and artificial intelligence
- Michael Jordan (insolvency baron) (born 1931), English businessman

Entity Linking

- Similar Problems:
  - Coreference resolution
  - Word sense disambiguation
  - Database record linkage

Entity Linking

- Basic steps in Entity Linking
  - Candidate Entity Generation
  - Candidate Entity Ranking
  - Unlinkable Mention Prediction

Entity Linking

- Basic steps in Entity Linking
  - Candidate Entity Generation
    - Name dictionary (generated by searching engine)

TABLE 1
A part of the name dictionary D

k (Name)	$k.value$ (Mapping entity)
Microsoft	*Microsoft*
Microsoft Corporation	*Microsoft*
Michael Jordan	*Michael Jordan* *Michael I. Jordan* *Michael Jordan (footballer)* *Michael Jordan (mycologist)* ...
Hewlett-Packard Company	*Hewlett-Packard*
HP	*Hewlett-Packard*
Bill Hewlett	*William Reddington Hewlett*

Entity Linking

- Basic steps in Entity Linking
  - Candidate Entity Generation
    - Surface form expansion: using heuristic or supervised learning to generate expansion rules.
    - E.g.

      Communist Party of China = CPC

      New York City = NYC

Entity Linking

- Basic steps in Entity Linking
  - Candidate Entity Ranking
    - Useful features include:
      String similarity
      Entity type. E.g. football player Michael Jordan
      Entity popularity
      Textual context.

Entity Linking

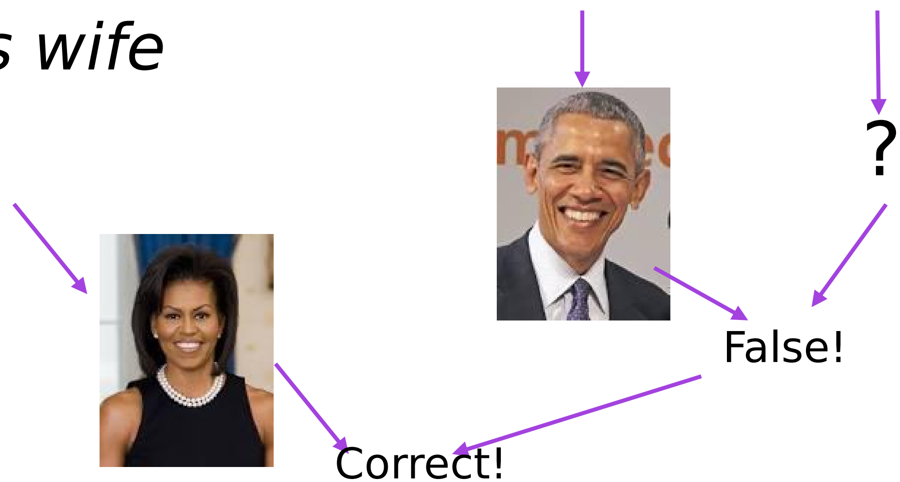
- Basic steps in Entity Linking
  - Unlinkable Mention Prediction:
    - Set a threshold for similarity score.
    - If the highest scored entity in KB is still lower than the threshold, predict as Unlinkable.

Entity Linking

- Entity Linking + NER:
  - Advantages:
    - NER may split a larger span into two mentions of less informative entities:

      E.g. *B.Obama's wife gave a speech …*
    - NER system recognizes *B.Obama & wife.*
    - Joint System:

      B.Obama's wife



Entity Linking

- Entity Linking + NER:
  - Advantages:
    - NER may choose a shorter span, referring to an incorrect entity:

      E.g. *The New York Times is a popular newspaper.*
    - NER system only recognizes *New York*

      a perfect Entity linking system will fail
    - Joint system:

      New York : False → back propagate error to NER system.

Entity Linking

- Entity Linking + NER:
  - Advantages:
    - NER may choose a longer span

      E.g. *Babies Romeo and Juliet were born hours apart*
    - NER system only recognizes *Romeo and Juliet* immediately.
    - Joint system:

      Romeo and Juliet is different from the context embedding → back propagate error to NER system.

Entity Linking

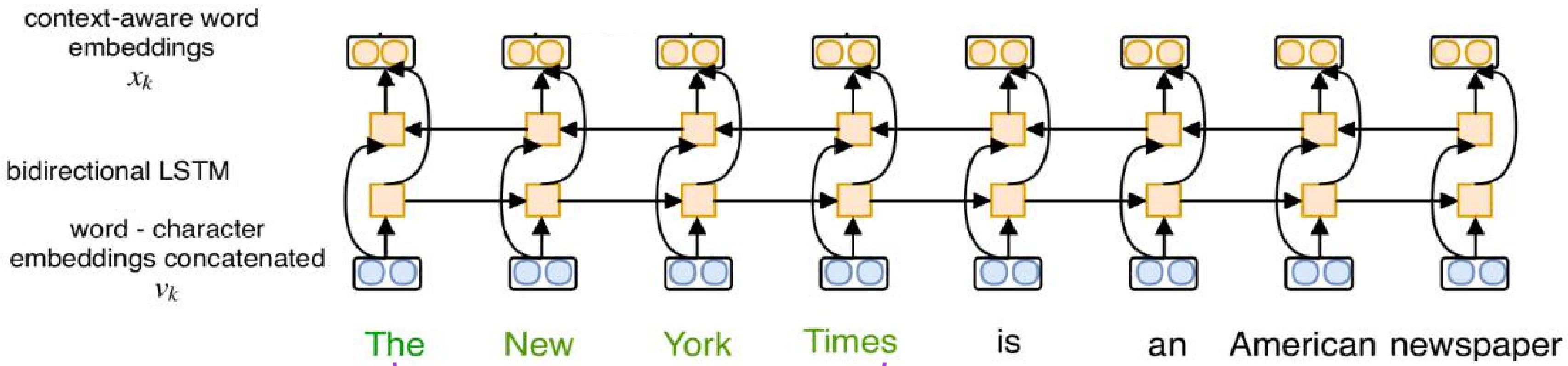
- End-to-End Neural Entity Linking (Entity Linking+NER)
  - Assume the existence of gold (ground-truth) text-link pairs.
  - Compare all possible spans with all entities in the knowledge base.
    - Span: a short sequence of words, possible entity mention.
  - Use context information to justify the pairing.

End-to-End Neural Entity Linking. ACL 2019.

Entity Linking

- End-to-End Neural Entity Linking (Entity Linking+NER)
  - Context embedding:

  Use Bi-LSTM to generate context-aware word embeddings.



A possible span

With its context

Entity Linking

- End-to-End Neural Entity Linking (Entity Linking+NER)
  - (Possible) entity mention embedding:
  how to represent spans with different lengths.

$$g^m = [x_q; x_r; \widehat{x}^m]$$

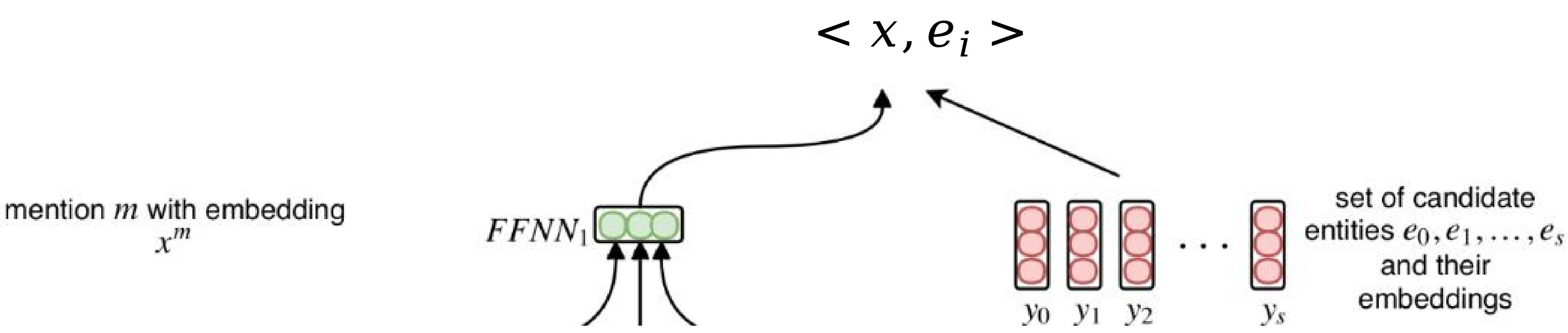
The first word

The weighted average words

The last word

Entity Linking

- End-to-End Neural Entity Linking (Entity Linking+NER)
  - Compare all possible spans with all entities in the knowledge base.



Entity Linking

- End-to-End Neural Entity Linking (Entity Linking+NER)
  - Use well-matched pairs to help uncertain pairs.
  - Well-matched pairs set in the document V.
  - Global similarity score: if the entity is close to other entities in the same document.

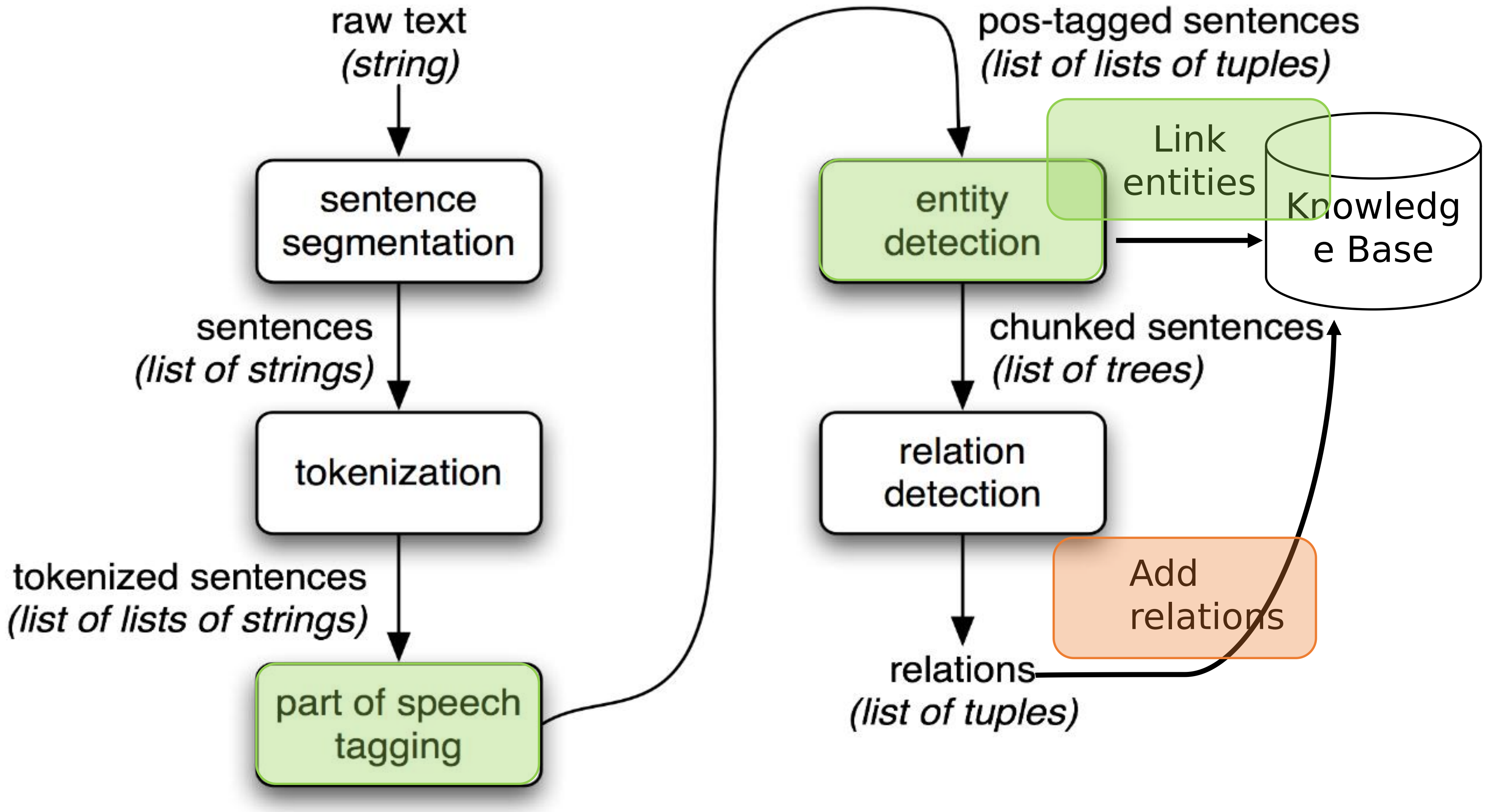
Entity in the KB

$$x_V = \sum_{e \in V} x_e, \quad Score(e_j, m) = \cos(e_j, x_V)$$

Possible entity mention

Information Extraction Architecture

- Review of the IE process (part 1)



Reading Material

a. Part-of-Speech Tagging (POS Tagging)

· Introduction from Wikipedia [link]

· Multilingual Part-of-Speech Tagging with Bidirectional Long Short-Term Memory Models and Auxiliary Loss. Plank 2016 [link]

· Blog: NLP Guide: Identifying Part of Speech Tags using Conditional Random Fields [link]

b. Sequence Labeling

· Hierarchically-Refined Label Attention Network for Sequence Labeling. EMNLP-IJCNLP 2019 [link]

· End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. ACL 2016 [link]

· Comparisons of sequence labeling algorithms and extensions. ICML 2007 [link]

For reading material recommendation of this course, please refer to our github.

Reading Material

c. Named Entity Recognition

· Blog: Named Entity Recognition Tagging, CS230 [link]

· A survey of named entity recognition and classification. David Nadeau, Satoshi Sekine. 2007 [link]

· Neural Architectures for Named Entity Recognition [link]

· Named entity recognition with bidirectional LSTM-CNNs [link]

For reading material recommendation of this course, please refer to our github

Q&A

THUNLP