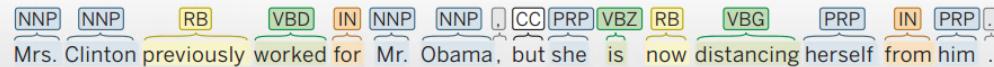
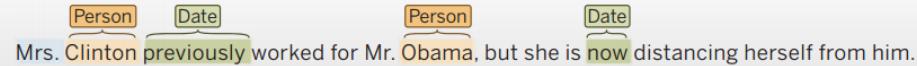
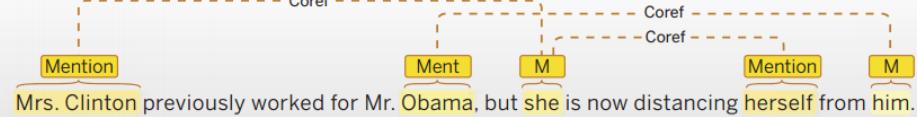
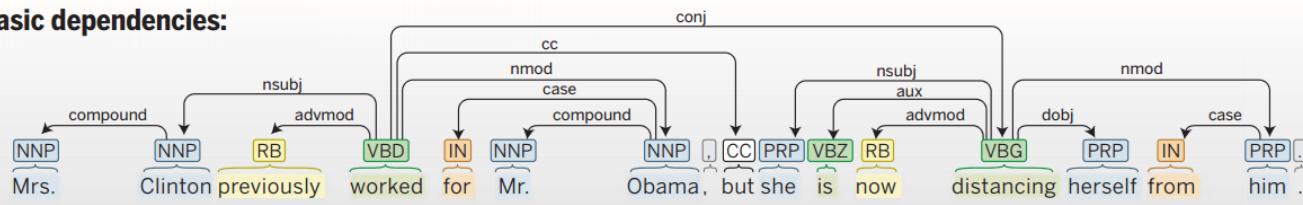


Part of speech:**Named entity recognition:****Co-reference:****Basic dependencies:**

Lecture 3: Natural Language Processing

Pilsung Kang

School of Industrial Management Engineering

Korea University

AGENDA

01 **Introduction to NLP**

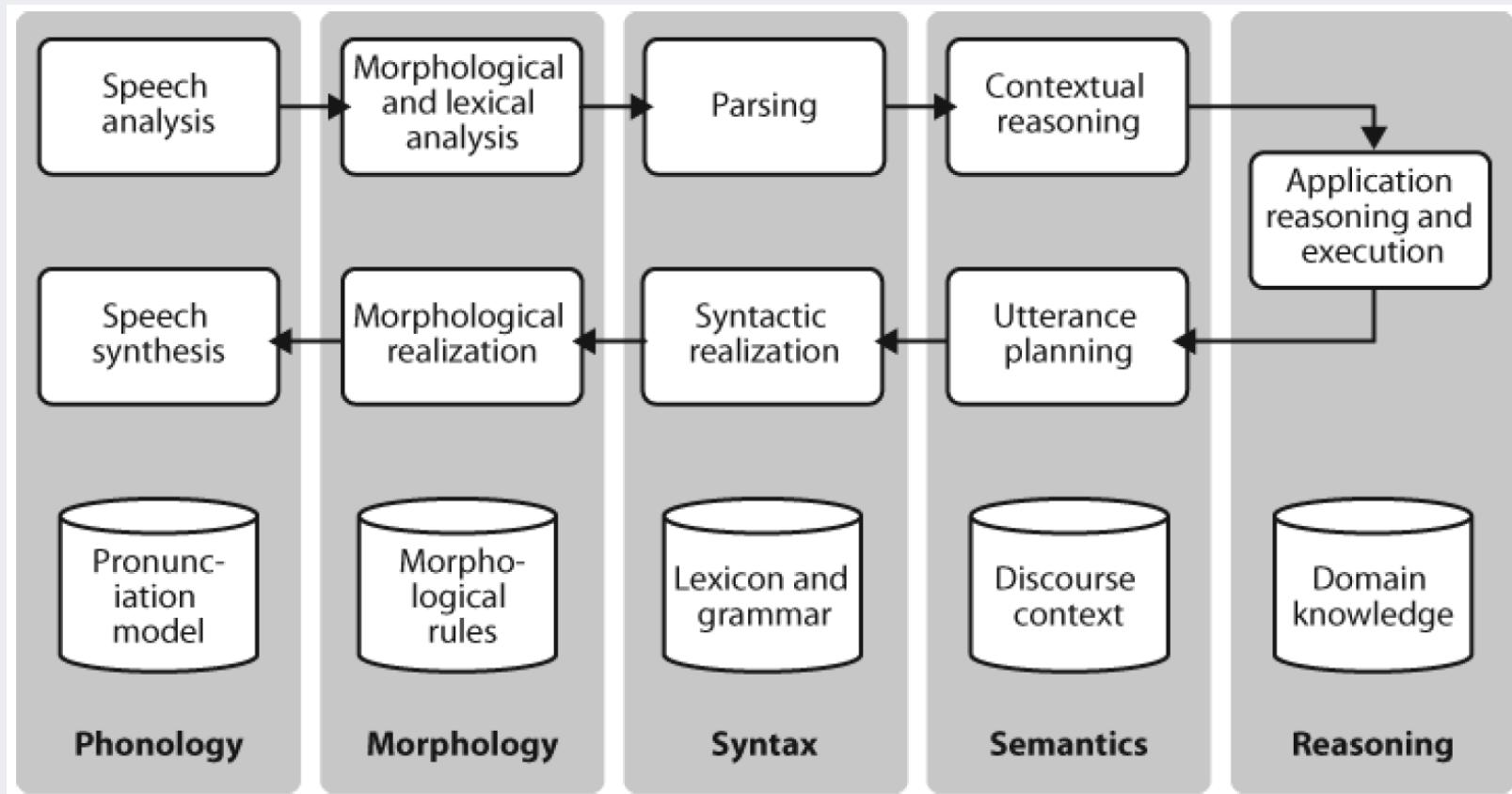
02 **Lexical Analysis**

03 **Syntax Analysis**

04 **Other Topics in NLP**

Natural Language Processing

- Natural language processing sequence



Natural Language Processing

- Natural language processing example



Natural Language Processing

Witte (2006)

- Classical categorization of NLP

Classical Categorization

To deal with the complexity of natural language, it is typically regarded on several levels (cf. Jurafsky & Martin):

Phonology the study of linguistic sounds

Morphology the study of meaningful components of words

Syntax the study of structural relationships between words

Semantics the study of meaning

Pragmatics the study of how language is used to accomplish goals

Discourse the study of larger linguistic units

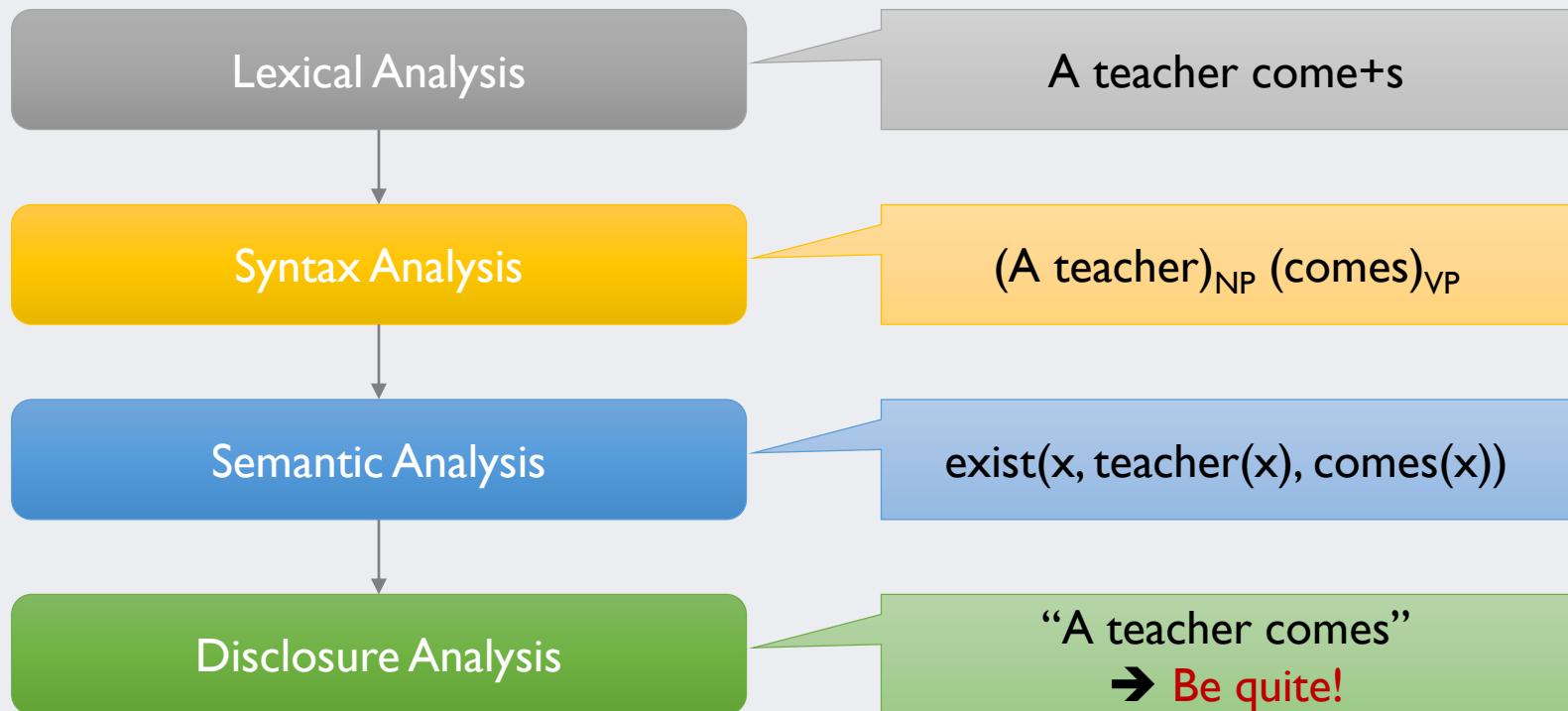
Importance for Text Mining

- Phonology only concerns spoken language
- Discourse, Pragmatics, and even Semantics is still rarely used

Natural Language Processing

Witte (2006)

- An example of NLP



Is NLP Easy? No!

Witte (2006)

- Why is NLP hard?

Difference to other areas in Computer Science

Computer scientist are used to dealing with precise, closed, artificial structures

- e.g., we build a “mini-world” for a database rather than attempting to model every aspect of the real world
- programming languages have a simple syntax (around 100 words) and a precise semantic

This approach does not work for natural language:

- tens of thousands of languages, with more than 100 000 words each
- complex syntax, many ambiguities, constantly changing and evolving

A corollary is that a TM system will never get it “100% right”

Is NLP Easy? No!

Witte (2006)

- Ambiguity of a natural language

Ambiguity appears on every analysis level

The classical examples:

- *He saw the man with the telescope.*
- *Time flies like an arrow. Fruit flies like a banana.*

And those are simple...

This does not get better with real-world sentences:

- *The board approved [its acquisition] [by Royal Trustco. Ltd.] [of Toronto] [for \$27 a share] [at its monthly meeting].*

(cf. Manning & Schütze)

Is NLP Easy? No!

Witte (2006)

- Complex and subtle relationship between concepts in texts
 - ✓ “AOL merges with Time-Warner”
 - ✓ “Time-Warner is bought by AOL”
- Ambiguity and context sensitivity
 - ✓ automobile = car = vehicle = Hyundai



vs.



Research Trends in NLP

Witte (2006)

- From rule-based approaches to statistical approaches

The classical way: until late 1980's

Rule-based approaches:

- are too rigid for natural language
- suffer from the *knowledge acquisition bottleneck*
- cannot keep up with changing/evolving language
ex. “*to google*”

The statistical way: since early 1990's

“Statistical NLP” refers to all quantitative approaches, including Bayes' models, Hidden Markov Models (HMMs), Support Vector Machines (SVMs), Clustering, ...

- more robust & more flexible
- need a *Corpus* for (supervised or unsupervised) learning

But real-world systems typically combine both.

Research Trends in NLP

Collobert et al. (2011)

- From statistical approaches to machine-learning (deep-learning) approaches

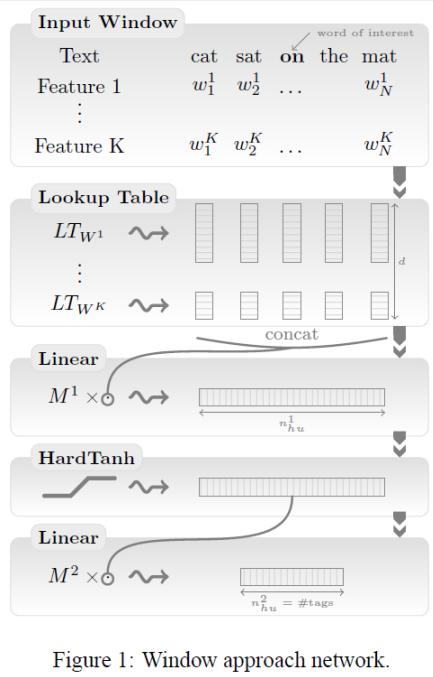


Figure 1: Window approach network.

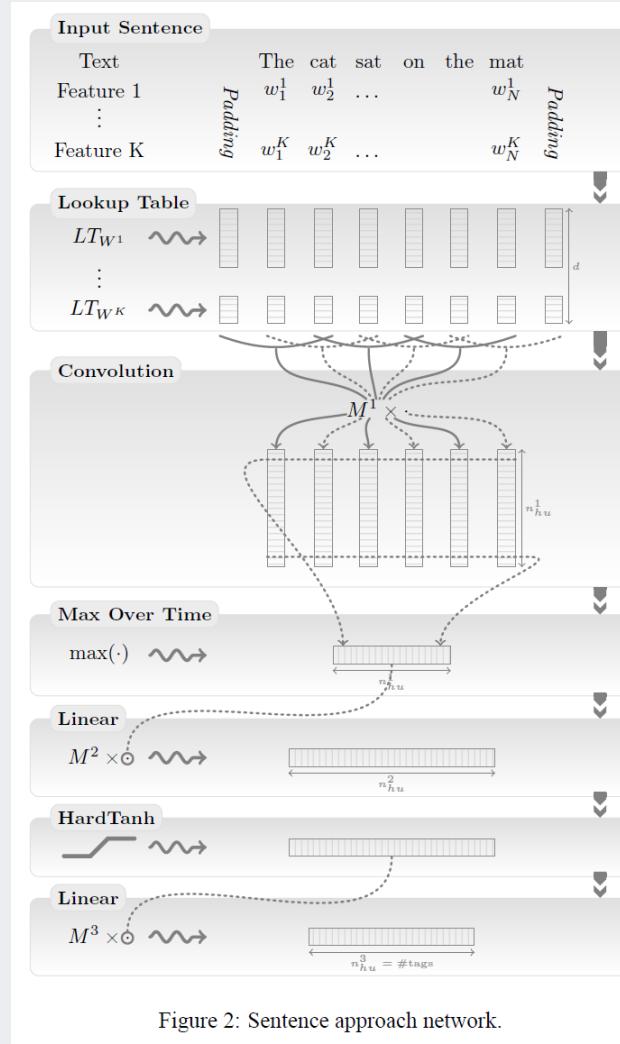


Figure 2: Sentence approach network.

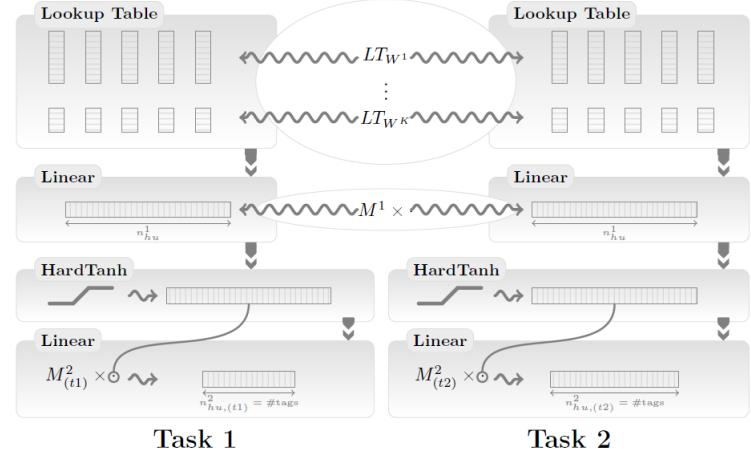


Figure 5: Example of multitasking with NN. Task 1 and Task 2 are two tasks trained with the window approach architecture presented in Figure 1. Lookup tables as well as the first hidden layer are shared. The last layer is task specific. The principle is the same with more than two tasks.

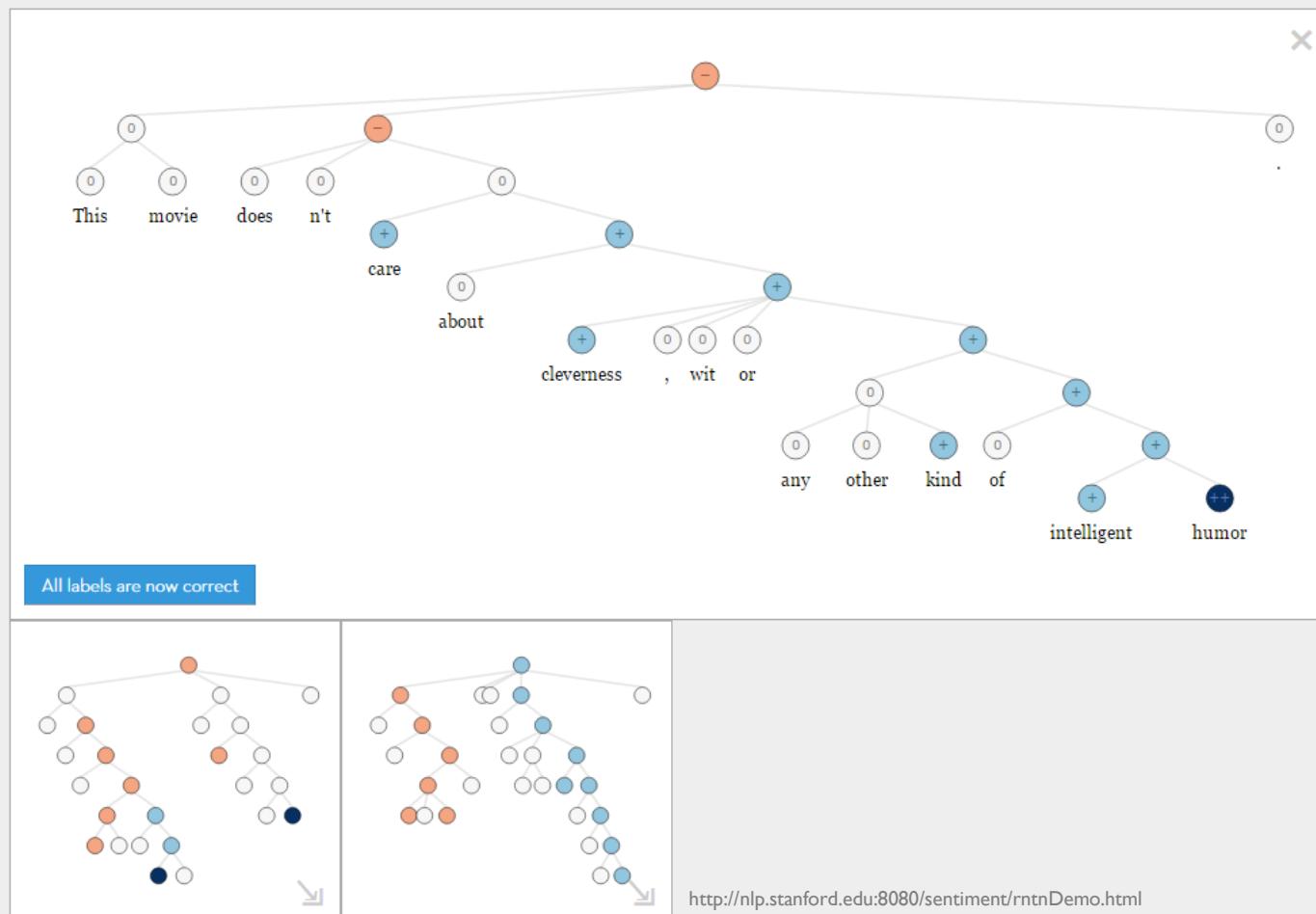
Research Trends in NLP

Socher et al. (2013)

- From statistical approaches to machine-learning (deep-learning) approaches

Sentiment Trees

You can double-click on each tree figure to see its expanded version with greater details. There are 5 classes of sentiment classification: very negative, negative, neutral, positive, and very positive.



Research Trends in NLP

Wu et al. (2016)

- From statistical approaches to machine-learning (deep-learning) approaches

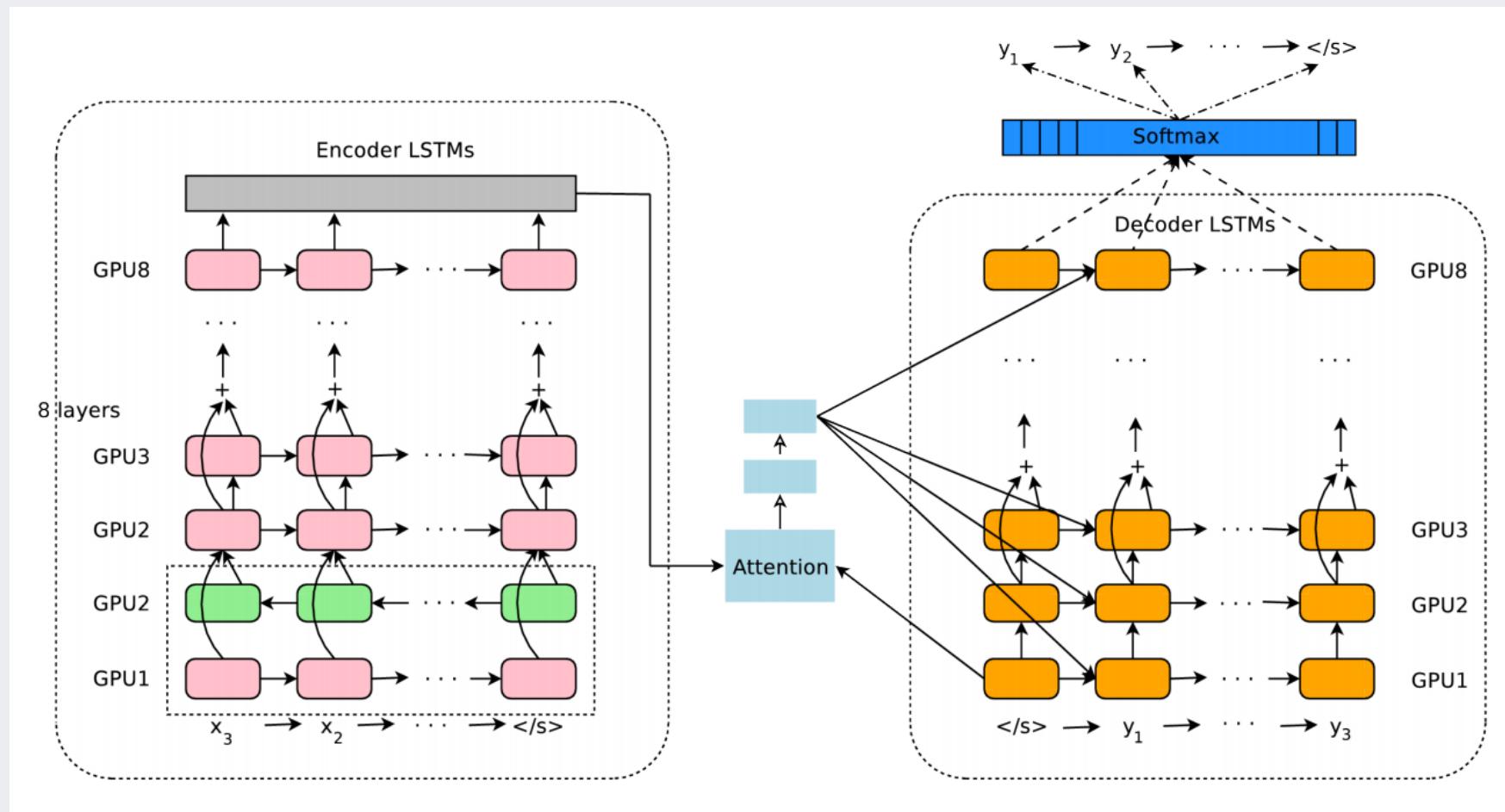


Figure 1: The model architecture of GNMT, Google's Neural Machine Translation system. On the left

Research Trends in NLP

- Statistical translation vs. deep learning-based translation

Translate Text

Input
Enter or paste text from a passage.

Korean

Output
Copy output from this field to clipboard.

English

Text **Rest API**

흔히 기업·가계 등 경제 주체와 금융시장이 가장 싫어하는 건 '불확실성'이라고 한다. 주요국 주식시장은 트럼프 미국 대통령 당선에 따른 경제회복 기대감으로 아직 상승하고 있지만, 정책당국·경제연구기관·학계 모두 우리 경제 안팎에 '불확실성 충격'이 염습하고 있다고 한목소리를 낸다. 트럼프발 정책 불확실성과 혼선, 미·중 통상마찰 심화 등 정치경제적 혼돈과 급변이 '불확실성 시대'의 실체로 지목된다. 전문가들은 "지금의 불확실성은 통제하기 어려운 요인에서 비롯되고 있다"며 "사전에 결과를 예측하기도 어렵고, 시장의 예측과 상반되는 결과가 나타날 가능성이 크므로 다양한 관점에서의 정보 획득이 필요한 때"라고 말한다.

Text **JSON**

Often, companies and households as the economic and financial market hates the most is uncertainty it is called " X-inneciency. Brazilian stocks fell as hopes for an economic recovery with Trump in the US presidential election is still rising, but both government policy-makers and economic research institutes and universities have shelters the 'uncertainty shock' and our economy. With TeuReomPeuBal, the China-US trade policy uncertainty and confusion entity of class variation " era of uncertainty with a deep political and economic chaos. Experts said the current uncertainty is difficult to control factors arising from competing, and

Research Trends in NLP

- Statistical translation vs. deep learning-based translation

The screenshot shows the Google Translate interface. On the left, there is a box containing a block of Korean text. The text discusses the concept of '불확실성' (uncertainty) in the context of the US economy under President Trump, mentioning various entities like corporations, households, and financial markets. It notes that while the stock market is rising, policy makers and experts are calling for more uncertainty. On the right, the same text is displayed in English, providing a detailed explanation of the 'uncertainty age'. The Google Translate UI includes language selection dropdowns at the top, a toolbar with icons for copy, paste, and refresh, and a blue '번역하기' (Translate) button.

번역

즉석 번역 사용 안함

한국어 영어 독일어 한국어 - 감지됨 ▾

영어 한국어 일본어 ▾ 번역하기

흔히 기업·가계 등 경제 주체와 금융시장이 가장 싫어하는 건 '불확실성'이라고 한다. 주요국 주식시장은 트럼프 미국 대통령 당선에 따른 경제회복 기대감으로 아직 상승하고 있지만, 정책당국과 경제연구기관·학계 모두 우리 경제 안팎에 '불확실성 충격'이 엄습하고 있다고 한목소리를 낸다. 트럼프의 정책 불확실성과 혼선, 미-중 통상마찰 심화 등 정치경제적 혼돈과 급변이 '불확실성 시대'의 실체로 지목된다. 전문가들은 "지금의 불확실성은 통제하기 어려운 요인에서 비롯되고 있다"며 "사전에 결과를 예측하기도 어렵고, 시장의 예측과 상반되는 결과가 나타날 가능성이 크므로 다양한 관점에서의 정보 획득이 필요한 때"라고 말한다.

It is often said that economic entities such as corporations and households and financial markets hate the 'uncertainty'. Although the stock market of major countries is still rising due to the expectation of economic recovery following the election of US President George W. Bush, policy makers, economic research institutes and academics both call for a "shock of uncertainty" inside and outside of our economy. Uncertainty, confusion, deepening frictions between the US and China, and political and economic chaos and sudden changes are pointing to the reality of the 'uncertainty age'. Experts say that "uncertainty now comes from factors that are difficult to control", "it is difficult to predict the outcome in advance, and there is a high possibility that the results will be in conflict with the market forecast. It says.

348/5000

수정 제안하기

Research Trends in NLP

- Provide your inputs to improve the machine translator!

The screenshot shows a machine translation interface with two examples. The top example translates "수능 망했다" (Failed the college entrance exam) to "screwed up on SAT". The bottom example translates "수능 만점" (Scored full marks in the college entrance exam) to "Seoul National University". Both examples include input fields, a character counter (6/5000 and 5/5000), and a blue '번역하기' (Translate) button.

한국어 영어 독일어 언어 감지 ▾

수능 망했다 ×
6/5000

영어 한국어 일본어 ▾ 번역하기

screwed up on SAT

수능 만점 ×
5/5000

수정 제안하기

The screenshot shows a machine translation interface with two examples. The top example translates "수능 만점" (Scored full marks in the college entrance exam) to "Seoul National University". The bottom example translates "수능 만점" (Scored full marks in the college entrance exam) to "Seoul National University". Both examples include input fields, a character counter (5/5000), and a blue '번역하기' (Translate) button.

한국어 영어 독일어 언어 감지 ▾

수능 만점 ×
5/5000

영어 한국어 일본어 ▾ 번역하기

Seoul National University

수정 제안하기

AGENDA

01 Introduction to NLP

02 Lexical Analysis

03 Syntax Analysis

04 Other Topics in NLP

Lexical Analysis

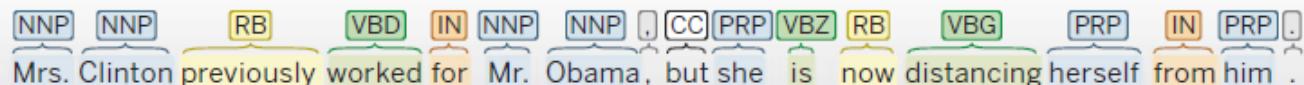
- Goals of lexical analysis
 - ✓ Convert a sequence of characters into a sequence of **tokens**, i.e., meaningful character strings.
 - In natural language processing, **morpheme** is a basic unit
 - In text mining, **word** is commonly used as a basic unit for analysis
- Process of lexical analysis
 - ✓ Tokenizing
 - ✓ Part-of-Speech (POS) tagging
 - ✓ Additional analysis: named entity recognition (NER), noun phrase recognition, sentence split, chunking, etc.

Lexical Analysis

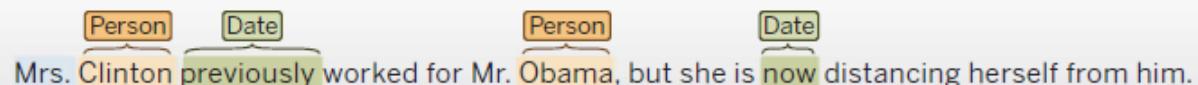
Hirschberg and Manning (2015)

- Examples of Linguistic Structure Analysis

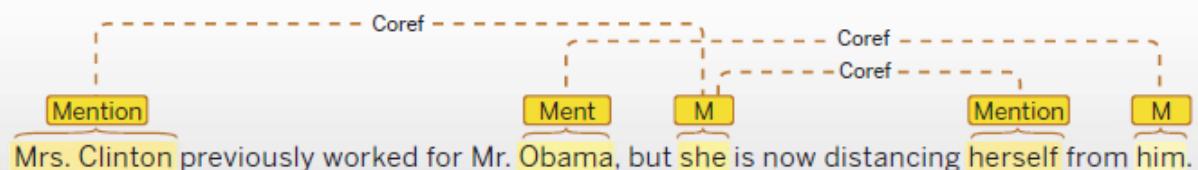
Part of speech:



Named entity recognition:



Co-reference:



Basic dependencies:

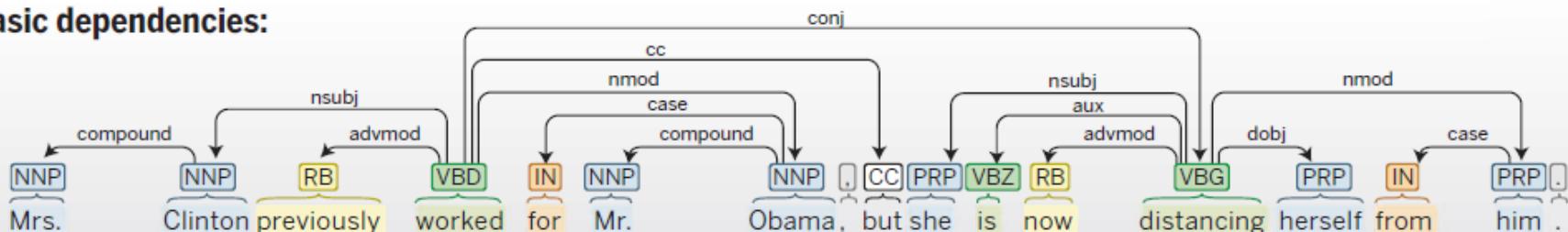


Fig. 1. Many language technology tools start by doing linguistic structure analysis. Here we show output from Stanford CoreNLP. As shown from top to bottom, this tool determines the parts of speech of each word, tags various words or phrases as semantic named entities of various sorts, determines which entity mentions co-refer to the same person or organization, and then works out the syntactic structure of each sentence, using a dependency grammar analysis.

Lexical Analysis I: Sentence Splitting

Witte (2016)

- Sentence is very important in NLP, but it is **not critical** for some Text Mining tasks

Mark Sentence Boundaries

Detects sentence units. Easy case:

- often, sentences end with “.”, “!”, or “?”

Hard (or annoying) cases:

- difficult when a “.” do not indicate an EOS:
“*MR. X*”, “*3.14*”, “*Y Corp.*”, ...
- we can detect common abbreviations (“U.S.”), but what if a sentence ends with one?
“...announced today by the U.S. *The...*
- Sentences can be *nested* (e.g., within quotes)

Correct sentence boundary is important

for many downstream analysis tasks:

- POS-Taggers maximize probabilities of tags within a sentence
- Summarization systems rely on correct detection of sentence

Lexical Analysis 2: Tokenization

- Text is split into basic units called Tokens

- ✓ word tokens, number tokens, space tokens, ...

Diagram illustrating the tokenization process:

```

> crude[[1]]
<<PlainTextDocument (metadata: 15)>>
Diamond Shamrock Corp said that
effective today it had cut its contract prices for crude oil by
1.50 dlr$ a barrel.
The reduction brings its posted price for West Texas
Intermediate to 16.00 dlr$ a barrel, the copany said.
"The price reduction today was made in the light of falling
oil product prices and a weak crude oil market," a company
spokeswoman said.
Diamond is the latest in a line of U.S. oil companies that
have cut its contract, or posted, prices over the last two days
citing weak oil markets.
Reuter
  
```

The text is shown in three panels: the original document, the output of the MC_tokenizer, and the output of the scan_tokenizer.

	MC	Scan
Space	Not removed	Removed
Punctuation	Removed	Not removed
Numbers	Removed	Not removed
Special characters	Removed	Not removed

Lexical Analysis 2: Tokenization

- Even tokenization can be difficult

- ✓ Is John's sick one token or two?

- If one → problems in parsing (where is the verb?)
 - If two → what do we do with John's house?

- ✓ What to do with hyphens?

- database vs. data-base vs. data base

- ✓ What to do with “C++”, “A/C”, “:-)”, “...”, “ㅋㅋㅋㅋㅋㅋㅋㅋ”?

- ✓ Some languages do not use whitespace (e.g., Chinese)

2013年5月，习主席在视察成都战区时，郑重提出在适当时候召开全军政治工作会议，并明确提出到古田召开这次会议，以更好弘扬我党我军的光荣传统和优良作风。6月，总政治部向中央军委提交《关于筹备召开全军政治工作会议的请示》，提出要通过召开会议形成一个指导性文件。习主席随即批示同意，明确要求这个文件要充分体现深厚的历史积淀和政治意蕴，能够管一个时期，起到历史性作用。

- Consistent tokenization is important for all later processing steps.

Lexical Analysis 3: Morphological Analysis

Witte (2016)

- Morphological Variants: Stemming and Lemmatization

Morphological Variants

Words are changed through a morphological process called *inflection*:

- typically indicates changes in case, gender, number, tense, etc.
- example *car* → *cars*, *give* → *gives*, *gave*, *given*

Goal: “normalize” words

Stemming and Lemmatization

Two main approaches to normalization:

Stemming reduce words to a *base form*

Lemmatization reduce words to their *lemma*

Main difference: stemming just finds **any** base form, which doesn't even need to be a word in the language! Lemmatization find the actual *root* of a word, but requires morphological analysis.

Lexical Analysis 3: Morphological Analysis

Witte (2016)

- Stemming

Stemming

Commonly used in Information Retrieval:

- Can be achieved with rule-based algorithms, usually based on suffix-stripping
- Standard algorithm for English: the *Porter* stemmer
- Advantages: simple & fast
- Disadvantages:
 - Rules are language-dependent
 - Can create words that do not exist in the language, e.g., *computers* → *comput*
 - Often reduces different words to the same stem, e.g., *army, arm* → *arm*
 - *stocks, stockings* → *stock*
- Stemming for German: German stemmer in the full-text search engine *Lucene*, *Snowball* stemmer with German rule file

Lexical Analysis 3: Morphological Analysis

Witte (2016)

- Lemmatization

Lemmatization

Lemmatization is the process of deriving the base form, or *lemma*, of a word from one of its inflected forms. This requires a morphological analysis, which in turn typically requires a *lexicon*.

- Advantages:
 - identifies the *lemma* (root form), which is an actual word
 - less errors than in stemming
- Disadvantages:
 - more complex than stemming, slower
 - requires additional language-dependent resources
- While stemming is good enough for Information Retrieval, Text Mining often requires lemmatization
 - Semantics is more important (we need to distinguish an *army* and an *arm!*)
 - Errors in low-level components can multiply when running downstream

Lexical Analysis 3: Morphological Analysis

- Stemming vs. Lemmatization

Word	Stemming	Lemmatization
Love	Lov	Love
Loves	Lov	Love
Loved	Lov	Love
Loving	Lov	Love
Innovation	Innovat	Innovation
Innovations	Innovat	Innovation
Innovate	Innovat	Innovate
Innovates	Innovat	Innovate
Innovative	Innovat	Innovative

Lexical Analysis 3: Morphological Analysis

- Stemming vs. Lemmatization with crude example

```
> crude[[1]]  
<<PlainTextDocument (metadata: 15)>>  
Diamond Shamrock Corp said that  
effective today it had cut its contract prices for crude oil by  
1.50 dlr s a barrel.  
The reduction brings its posted price for West Texas  
Intermediate to 16.00 dlr s a barrel, the company said.  
"The price reduction today was made in the light of falling  
oil product prices and a weak crude oil market," a company  
spokeswoman said.  
Diamond is the latest in a line of U.S. oil companies that  
have cut its contract, or posted, prices over the last two days  
citing weak oil markets.  
Reuter
```

Stemming

```
> stemCorpus[[1]]  
<<PlainTextDocument (metadata: 7)>>  
diamond shamrock corp said that  
effect today it had cut it contract price for crude oil by  
dlrs a barrel  
the reduct bring it post price for west texas  
intermedi to dlr s a barrel the copani said  
the price reduct today was made in the light of falling  
oil product price and a weak crude oil market a company  
spokeswoman said  
diamond is the latest in a line of us oil compani that  
hav cut it contract or post price over the last two days  
cit weak oil markets  
reuter
```

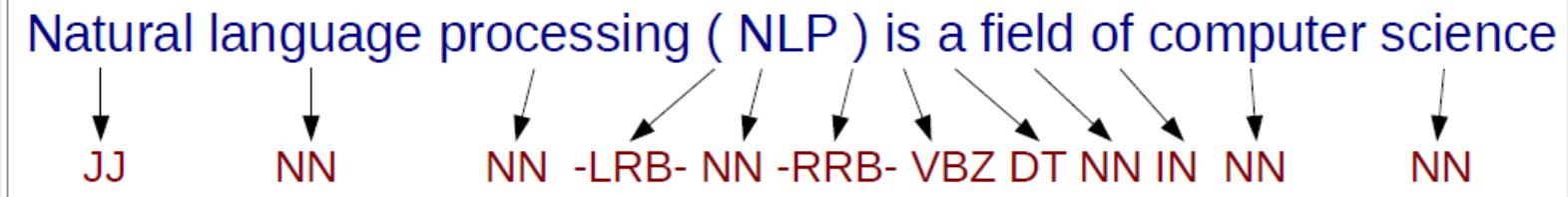
Lemmatization

```
> LemmaCorpus1  
[1] "diamond shamrock corp say that effective today it have  
cut it contract price for crude oil by dlr a barrel the reduc  
tion bring it post price for w texa intermediate to dlr a b  
arrel the copany say the price reduction today be make in th  
e light have fall oil product price and a weak crude oil mar  
ket a company spokeswoman say diamond be the late in a line  
have us oil company that have cut it contract or post price  
ov the last two day cite weak oil market reut"
```

Lexical Analysis 4: Part-of-Speech (POS) Tagging

Witte (2016)

- Part of speech (POS) tagging
 - ✓ Given a sentence X, predict its part of speech sequence Y
 - Input: tokens that sentence may have ambiguity
 - Output: most appropriate tag by considering its definition and contexts (relationship with adjacent and related words in phrases, sentence, or paragraph)
 - ✓ A type of “structured” prediction



- Different POS tags for the same token
 - ✓ I love you. → “love” is a verb
 - ✓ All you need is love. → “love” is noun

Lexical Analysis 4: Part-of-Speech (POS) Tagging

- POS Tagging

POS-Tagging

A statistical POS Tagger scans tokens and assigns **POS Tags**.

A black cat plays... → A/DT black/JJ cat/NN plays/VB...

- relies on different word order probabilities
- needs a manually tagged corpus for machine learning

Note: *this is not parsing!*

Lexical Analysis 4: Part-of-Speech (POS) Tagging

- Tagsets: English

Penn Treebank

TAG	DESCRIPTION	EXAMPLE
CC	conjunction, coordinating	<i>and, or, but</i>
CD	cardinal number	<i>five, three, 13%</i>
DT	determiner	<i>the, a, these</i>
EX	existential there	<i><u>there</u> were six boys</i>
FW	foreign word	<i>mais</i>
IN	conjunction, subordinating or preposition	<i>of, on, before, unless</i>
JJ	adjective	<i>nice, easy</i>
JJR	adjective, comparative	<i>nicer, easier</i>
JJS	adjective, superlative	<i>nicest, easiest</i>
LS	list item marker	
MD	verb, modal auxiliary	<i>may, should</i>
NN	noun, singular or mass	<i>tiger, chair, laughter</i>
NNS	noun, plural	<i>tigers, chairs, insects</i>
NNP	noun, proper singular	<i>Germany, God, Alice</i>
NNPS	noun, proper plural	<i>we met two <u>Christmases</u> ago</i>
PDT	predeterminer	<i><u>both</u> his children</i>
POS	possessive ending	<i>'s</i>
PRP	pronoun, personal	<i>me, you, it</i>
PRP\$	pronoun, possessive	<i>my, your, our</i>
RB	adverb	<i>extremely, loudly, hard</i>
RBR	adverb, comparative	<i>better</i>
RBS	adverb, superlative	<i>best</i>
RP	adverb, particle	<i>about, off, up</i>
SYM	symbol	<i>%</i>
TO	infinitival to	<i>what <u>to</u> do?</i>
UH	interjection	<i>oh, oops, gosh</i>
VB	verb, base form	<i>think</i>
VBD	verb, 3rd person singular present	<i>she <u>thinks</u></i>
VBP	verb, non-3rd person singular present	<i>I <u>think</u></i>
VBN	verb, past tense	<i>they <u>thought</u></i>
VBN	verb, past participle	<i>a <u>sunken</u> ship</i>
VBG	verb, gerund or present participle	<i><u>thinking</u> is fun</i>
WDT	wh-determiner	<i>which, whatever, whichever</i>
WP	wh-pronoun, personal	<i>what, who, whom</i>
WP\$	wh-pronoun, possessive	<i>whose, whomever</i>
WRB	wh-adverb	<i>where, when</i>
.	punctuation mark, sentence closer	<i>.?*</i>
,	punctuation mark, comma	<i>,</i>
:	punctuation mark, colon	<i>:</i>
(contextual separator, left paren	<i>(</i>
)	contextual separator, right paren	<i>)</i>

UCREL CLAWS7 Tagset

APPG	possessive pronoun, pre-nominal (e.g. my, your, our)
AT	article (e.g. the, no)
AT1	singular article (e.g. a, an, every)
BCL	before-clause marker (e.g. in order (that),in order (to))
CC	coordinating conjunction (e.g. and, or)
CCB	adversative coordinating conjunction (but)
CS	subordinating conjunction (e.g. if, because, unless, so, for)
CSA	(as conjunction)
CSN	than (as conjunction)
CST	that (as conjunction)
CSW	whether (as conjunction)
DA	after-determiner or post-determiner capable of pronominal function (e.g. such, former, same)
DA1	singular after-determiner (e.g. little, much)
DA2	plural after-determiner (e.g. few, several, many)
DAR	comparative after-determiner (e.g. more, less, fewer)
DAT	superlative after-determiner (e.g. most, least, fewest)
DB	before determiner or pre-determiner capable of pronominal function (all, half)
DB2	plural before-determiner (both)
DD	determiner (capable of pronominal function) (e.g any, some)
DD1	singular determiner (e.g. this, that, another)
DD2	plural determiner (these, those)
DDQ	wh-determiner (which, what)
DDQE	wh-determiner, genitive (whose)
DDQV	wh-ever determiner, (whichever, whatever)
EX	existential there
FO	formula
FU	unclassified word
FW	foreign word
GE	germanic genitive marker - (' or's)
IF	for (as preposition)
II	general preposition
IO	of (as preposition)
IW	with, without (as prepositions)
JJ	general adjective
JJR	general comparative adjective (e.g. older, better, stronger)
JJT	general superlative adjective (e.g. oldest, best, strongest)
JK	catenative adjective (able in be able to, willing in be willing to)
MC	cardinal number,neutral for number (two, three..)
MC1	singular cardinal number (one)
MC2	plural cardinal number (e.g. sixes, sevens)
MCGE	genitive cardinal number, neutral for number (two's, 100's)
MCNC	hyphenated number (40-50, 1770-1827)
MD	ordinal number (e.g. first, second, next, last)
MF	fraction,neutral for number (e.g. quarters, two-thirds)
NI	singular noun of direction (e.g. north, southeast)
NN	common noun, neutral for number (e.g. sheep, cod, headquarters)
NNI	singular common noun (e.g. book, girl)
NN2	plural common noun (e.g. books, girls)
NNA	following noun of title (e.g. M.A.)
NNB	preceding noun of title (e.g. Mr., Prof.)
NNL1	singular locative noun (e.g. Island, Street)
NNL2	plural locative noun (e.g. Islands, Streets)
NNO	numerical noun, neutral for number (e.g. dozen, hundred)
NNO2	numerical noun, plural (e.g. hundreds, thousands)
NNT1	temporal noun, singular (e.g. day, week, year)
NNT2	temporal noun, plural (e.g. days, weeks, years)
NNU	unit of measurement, neutral for number (e.g. in, cc)
NNU1	singular unit of measurement (e.g. inch, centimetre)
NNU2	plural unit of measurement (e.g. ins., feet)
NP	proper noun, neutral for number (e.g. IBM, Andes)
NP1	singular proper noun (e.g. London, Jane, Frederick)
NP2	plural proper noun (e.g. Browns, Reagans, Koreas)
NPD1	singular weekday noun (e.g. Sunday)
NPD2	plural weekday noun (e.g. Sundays)
NPM1	singular month noun (e.g. October)
NPM2	plural month noun (e.g. Octobers)
PN	indefinite pronoun, neutral for number (none)
PN1	indefinite pronoun, singular (e.g. anyone, everything, nobody, one)
PNQO	objective wh-pronoun (whom)
PNQS	subjective wh-pronoun (who)
PNQV	wh-ever pronoun (whoever)
PNX1	reflexive indefinite pronoun (oneself)
PPQE	nominal possessive personal pronoun (e.g. mine, yours)

Lexical Analysis 4: Part-of-Speech (POS) Tagging

- Tagsets: Korean

한글 형태소 품사 (Part Of Speech, POS) 태그표

한글 형태소의 품사를 체언, 용언, 관형사, 부사, 감탄사, 조사, 어미, 접사, 어근, 부호, 한글 이외'와 같이 나누고 각 세부 품사를 구분한다.

대분류	세종 품사 태그		심광섭 품사 태그		KKMA 단일 태그 V 1.0								
	태그	설명	Class	설명	묶음 1	묶음 2	태그	설명	학점태그	저장사전			
체언	NNG	일반 명사	N	NN	명사	NN	NNG	보통 명사	noun.dic	E	EFN	평서형 종결 어미	
	NNP	고유 명사		NNP	고유 명사		NNP	고유 명사			EFQ	의문형 종결 어미	
	NNB	의존 명사		NX	의존 명사		NNB	일반 의존 명사			EFO	명령형 종결 어미	
	NR	수사		UM	단위 명사		NNM	단위 의존 명사			EFA	형유형 종결 어미	
	NP	대명사		NU	수사		NR	수사			EFI	감탄형 종결 어미	
용언	VV	동사	V	VV	동사	verb.dic	VV	동사	verb.dic	E	EFR	준형형 종결 어미	
	VA	형용사		AJ	형용사		VA	형용사			ECE	대동 연결 어미	
	VX	보조 동언		VX	보조 동사		VX	보조 동사			ECD	의존적 연결 어미	
	VCP	긍정 지정사		CP	서술격 조사 '이다'	VC	VCP	긍정 지정사, 서술격 조사 '이다'	raw.dic		ECS	보조적 연결 어미	
	VCN	부정 지정사					VCN	부정 지정사, 형용사 '아니 다'			ETN	명사형 전성 어미	
관형사	MM	관형사	MD	DT	일반 관형사	MD	MDT	일반 관형사	raw.dic	E	ETD	관형형 전성 어미	
				DN	수 관형사		MDN	수 관형사			XPN	체언 접두사	
	MAG	일반 부사		AD	부사		MAG	일반 부사			XPV	용언 접두사	
부사	MAG	접속 부사	MA			MA	MAC	접속 부사	raw.dic	E	XSN	명사 파생 접미사	
	MAC						MAC				XSV	동사 파생 접미사	
감탄사	IC	감탄사	I	EX	감탄사	I	IC	감탄사	raw.dic	E	XSA	형용사 파생 접미사	
	JKS	주격 조사		JKS	주격 조사		JKS	주격 조사			XSA	형용사 파생 접미사	
조사	JKC	보격 조사	JO	JKC	보격 조사	J	JKC	보격 조사	raw.dic	E	XSM	부사-파생 접미사	
	JKG	관형격 조사		JKG	관형격 조사		JKG	관형격 조사			XSO	카타-접미사	
	JKO	목적격 조사		JKO	목적격 조사		JKO	목적격 조사			XR	어근	
	JKB	부사격 조사		JKM	부사격 조사		JKM	부사격 조사			SF	마침표/물음표, 느낌표	
	JKV	호격 조사		JKI	호격 조사		JKI	호격 조사			SP	쉼표, 가운뎃점, 콜론, 빛금	
	JKQ	인용격 조사		JKQ	인용격 조사		JKQ	인용격 조사			SS	따옴표, 괄호표, 줄표	
	JX	보조사		JX	보조사		JX	보조사			SE	줄임표	
	JC	접속 조사		JC	접속 조사		JC	접속 조사			SO	불임표(물결, 숨김, 빠짐)	
선어말 어미	EP	선어말 어미	EP	EP	선어말 어미	EP	EPH	준형 선어말 어미	raw.dic	E	SW	기타기호 (논리수학기호, 화 폐기호)	
				EPT	시제 선어말 어미						UN	영사추정법주	
				EPP	공손 선어말 어미						UV	용언추정법주	

어말 어미	EF	종결 어미	EM	어말 어미	E	EFN	평서형 종결 어미	EF
						EFQ	의문형 종결 어미	
						EFO	명령형 종결 어미	
						EFA	형유형 종결 어미	
						EFI	감탄형 종결 어미	
	EC	연결 어미	ET	어말 어미	E	EFR	준형형 종결 어미	
						ECE	대동 연결 어미	
						ECD	의존적 연결 어미	
						ECS	보조적 연결 어미	
						ETN	명사형 전성 어미	
접두사	ET	접두사	ETD	어말 어미	E	ETD	관형형 전성 어미	ETD
						XPN	체언 접두사	
						XPV	용언 접두사	
						XSN	명사 파생 접미사	
						XSV	동사 파생 접미사	
	X	접미사	SY	어말 어미	E	XSA	형용사 파생 접미사	ETD
						XSA	형용사 파생 접미사	
						XSM	부사-파생 접미사	
						XSO	카타-접미사	
						XR	어근	
부호	SY	부호 외래어	S	어말 어미	E	SF	마침표/물음표, 느낌표	Symbol class
						SP	쉼표, 가운뎃점, 콜론, 빛금	
						SS	따옴표, 괄호표, 줄표	
						SE	줄임표	
						SO	불임표(물결, 숨김, 빠짐)	
	U	분석 불능	U	어말 어미	E	SW	기타기호 (논리수학기호, 화 폐기호)	
						UN	영사추정법주	
						UV	용언추정법주	
						UE	분석불능법주	
						ON	숫자	
한글 이외	O	한글 이외	O	어말 어미	E	OL	외국어	N/A
						OH	한자	
						ON	숫자	
						NR		

Lexical Analysis 4: Part-of-Speech (POS) Tagging

Witte (2016)

- POS Tagging Algorithms

Fundamentals

POS-Tagging generally requires:

Training phase where a manually annotated corpus is processed by a machine learning algorithm; and a

Tagging algorithm that processes texts using learned parameters.

Performance is generally good (around 96%) when staying in the same domain.

Algorithms used in POS-Tagging

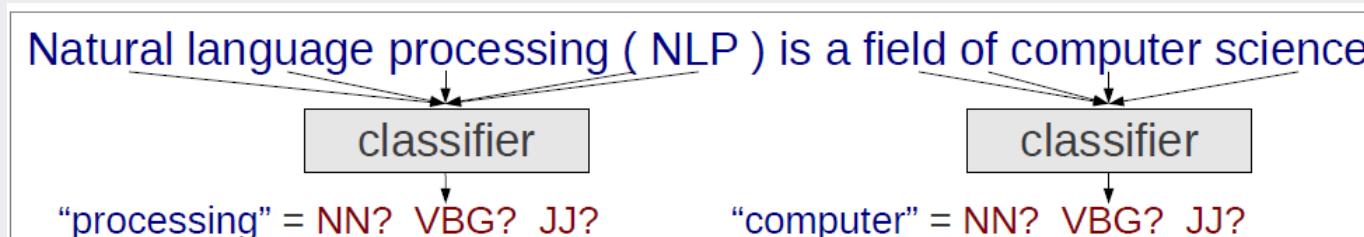
There is a multitude of approaches, commonly used are:

- Decision Trees
- Hidden Markov Models (HMMs)
- Support Vector Machines (SVM)
- Transformation-based Taggers (e.g., the Brill tagger)

Lexical Analysis 4: Part-of-Speech (POS) Tagging

- POS Tagging Algorithms

- ✓ Pointwise prediction: predict each word individually with a classifier (e.g. Maximum Entropy Model, SVM)



- ✓ Probabilistic models

- Generative sequence models: Find the most probable tag sequence given the sentence (Hidden Markov Model; HMM)
- Discriminative sequence models: Predict whole sequence with a classifier (Conditional Random Field; CRF)

- ✓ Neural network-based models

Lexical Analysis 4: Part-of-Speech (POS) Tagging

- Pointwise Prediction: Maximum Entropy Model

- ✓ Encode features for tag prediction

- Information about word/context: suffix, prefix, neighborhood word information
 - eg: $f_i(w_j, t_j) = 1$ if suffix(w_j) = “ing” & t_j = VBG, 0 otherwise

- ✓ Tagging Model

$$p(t|C) = \frac{1}{Z(C)} \exp \left(\sum_{i=1}^n \lambda_i f_i(C, t) \right)$$

$$p(t_1 \dots t_n | w_1 \dots w_n) \approx \prod_{i=1}^n p(t_i | C_i)$$

- f_i is a feature
 - λ_i is a weight (large value implies informative features)
 - $Z(C)$ is a normalization constant ensuring a proper probability distribution
 - Makes no independence assumption about the features

Lexical Analysis 4: Part-of-Speech (POS) Tagging

- Pointwise Prediction: Maximum Entropy Model

- ✓ An example

```
48 # POS Tagging with MaxEnt
49 install.packages("openNLP")
50 library(openNLP)
51
52 s1 <- paste(c("Pierre Vinken, 61 years old, will join the board as a ",
53             "nonexecutive director Nov. 29.\n",
54             "Mr. Vinken is chairman of Elsevier N.V., ",
55             "the Dutch publishing group."),
56             collapse = ""))
57 s1 <- as.string(s1)

> s1
Pierre Vinken, 61 years old, will join the board as a nonexecutive director Nov. 29.
Mr. Vinken is chairman of Elsevier N.V., the Dutch publishing group.
```

Lexical Analysis 4: Part-of-Speech (POS) Tagging

- Pointwise Prediction: Maximum Entropy Model

- ✓ An example

```
59 ## Need sentence and word token annotations.  
60 s2 <- annotate(s1, list(Maxent_Sent_Token_Annotator(), Maxent_Word_Token_Annotator()))  
61  
62 ## POS tag probabilities as (additional) features.  
63 s3 <- annotate(s1, Maxent_POS_Tag_Annotator(probs = TRUE), s2)
```

```
> s3  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>  
#> #> #> #> #>
```

Lexical Analysis 4: Part-of-Speech (POS) Tagging

- Probabilistic Model for POS Tagging
 - ✓ Find the **most probable tag sequence** given the sentence

Natural language processing (NLP) is a field of computer science



$$\underset{Y}{\operatorname{argmax}} P(Y|X)$$

Lexical Analysis 4: Part-of-Speech (POS) Tagging

- Generative Sequence Model

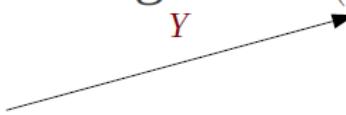
- ✓ Decompose probability using Baye's Rule

$$\operatorname{argmax}_Y P(Y|X) = \operatorname{argmax}_Y \frac{P(X|Y)P(Y)}{P(X)}$$

$$= \operatorname{argmax}_Y P(X|Y)P(Y)$$

Model of word/POS interactions
“natural” is probably a JJ

Model of POS/POS interactions
NN comes after DET



Lexical Analysis 4: Part-of-Speech (POS) Tagging

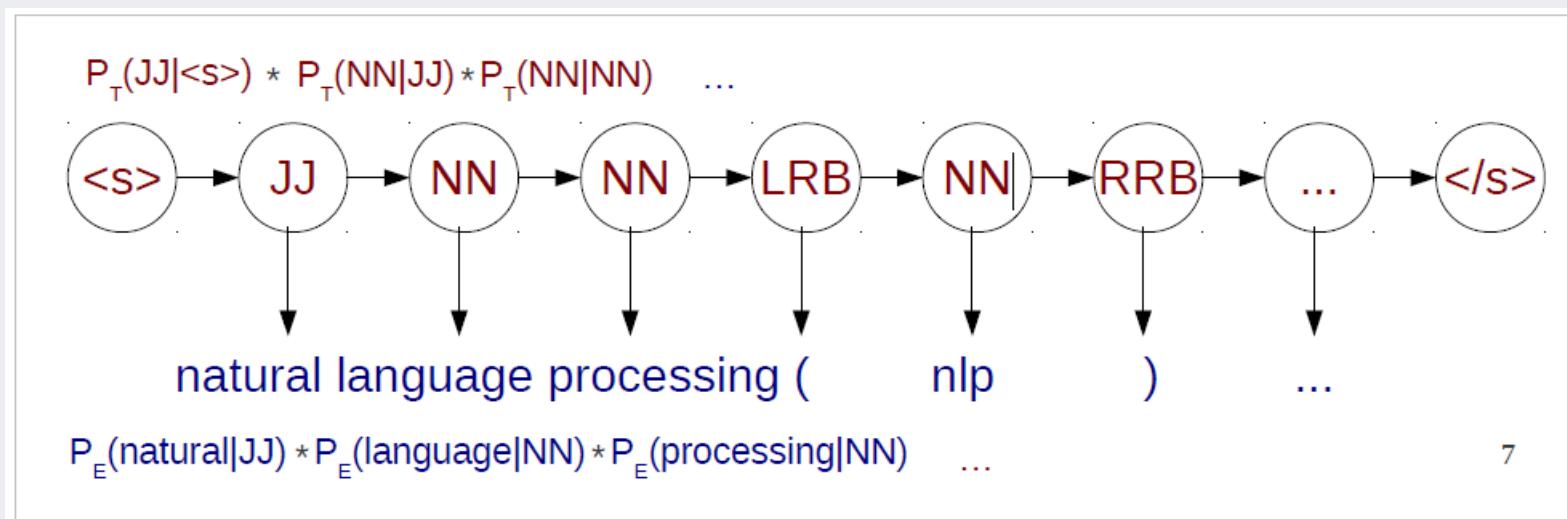
- Generative Sequence Model: Hidden Markov Model

- ✓ POS → POS **transition** probabilities

$$P(Y) \approx \prod_{i=1}^{I+1} P_T(y_i|y_{i-1})$$

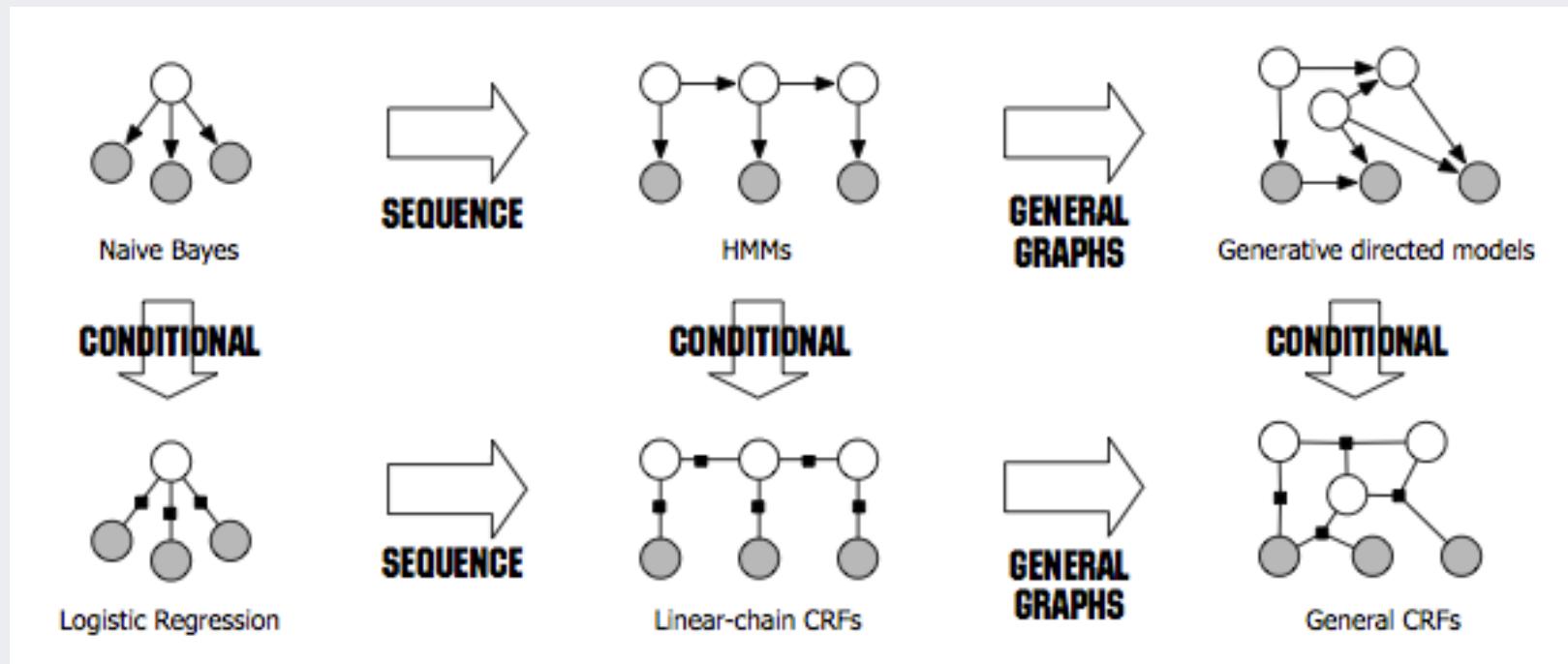
- ✓ POS → Word **emission** probabilities

$$P(X|Y) \approx \prod_1^I P_E(x_i|y_i)$$



Lexical Analysis 4: Part-of-Speech (POS) Tagging

- Discriminative Sequence Model: Conditional Random Field (CRF)
 - ✓ Relieve that constraint that a tag is generated by the previous tag sequence
 - ✓ Predict the whole tag set at the same time, not sequentially

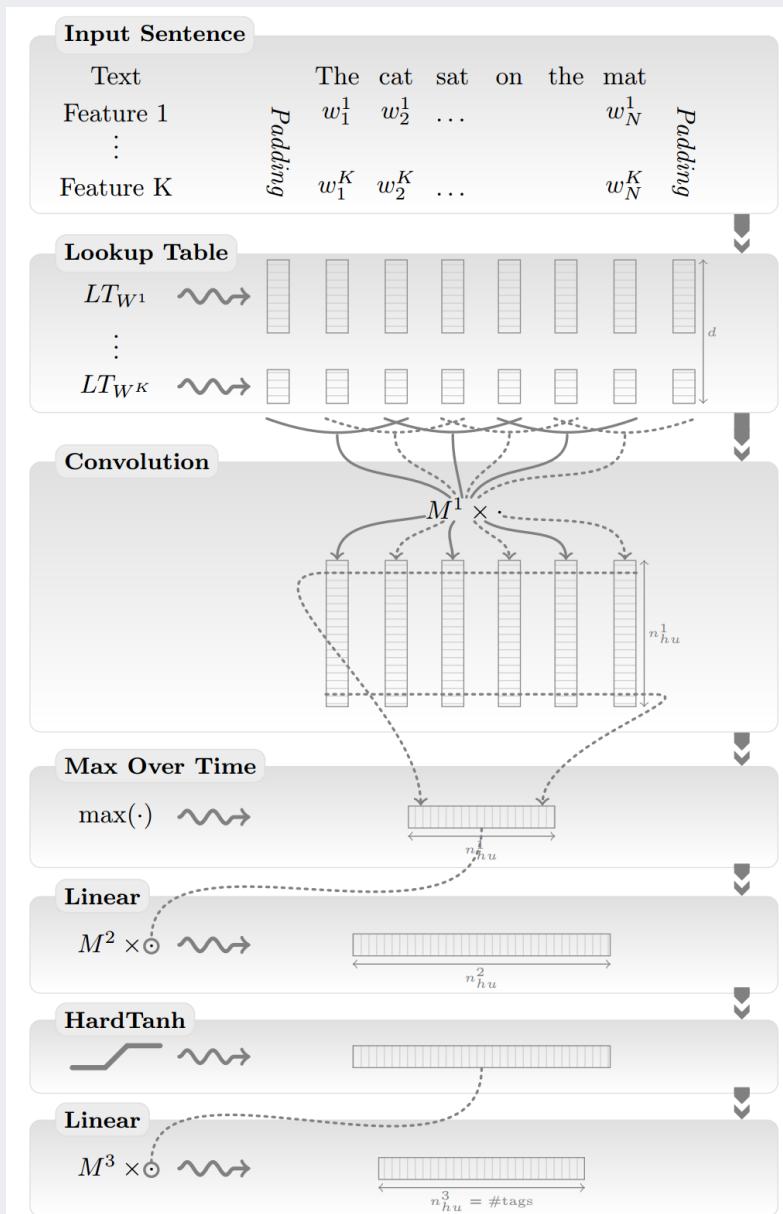
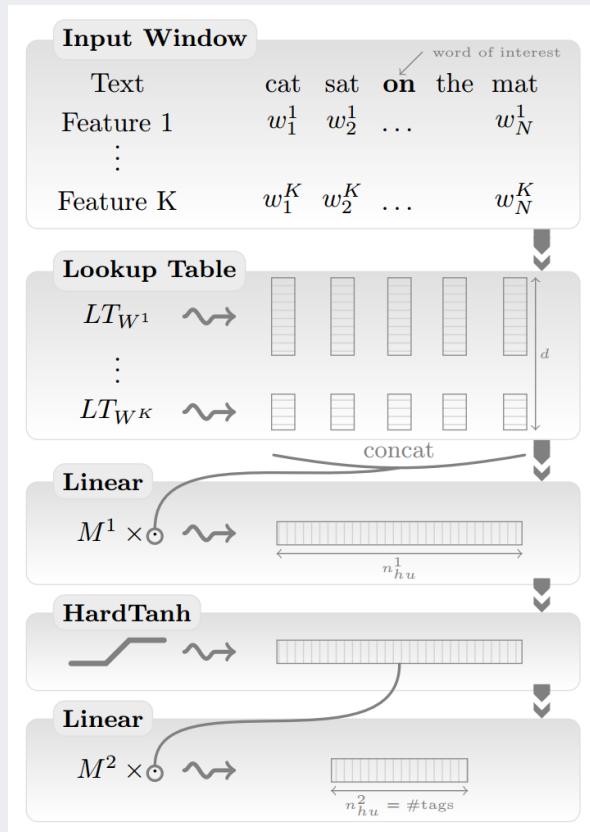


<http://people.cs.umass.edu/~mccallum/papers/crf-tutorial.pdf>

Lexical Analysis 4: Part-of-Speech (POS) Tagging

Collobert et al. (2011)

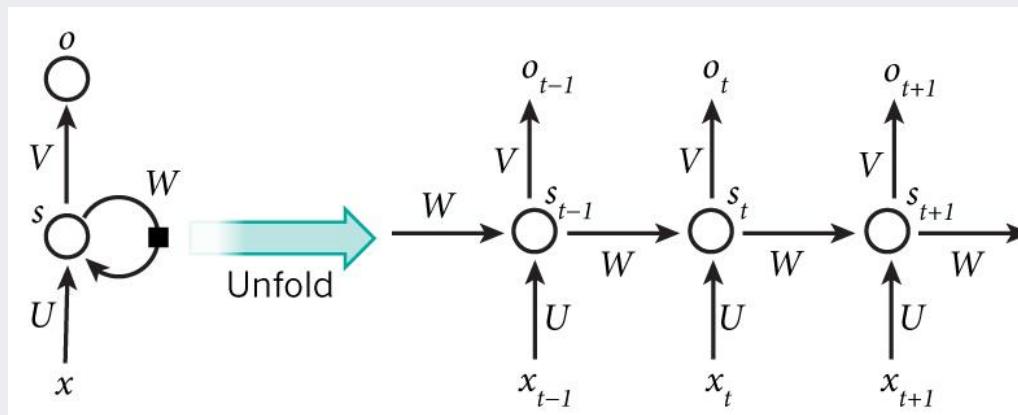
- Neural Network-based Models
- ✓ Window-based vs. sentence-based



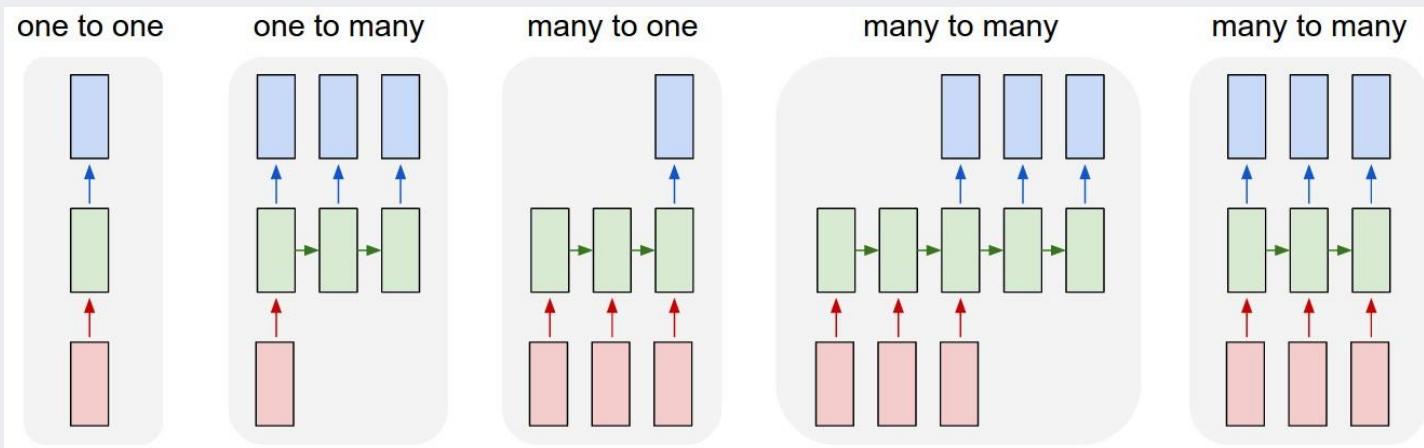
Lexical Analysis 4: Part-of-Speech (POS) Tagging

- Neural network-based models

- ✓ Recurrent neural networks: have a feedback loop within the hidden layer



- ✓ Input-Output mapping of RNNs



Lexical Analysis 4: Part-of-Speech (POS) Tagging

Perez-Ortiz and Forcada (2001)

- Neural network-based models: Recurrent neural networks

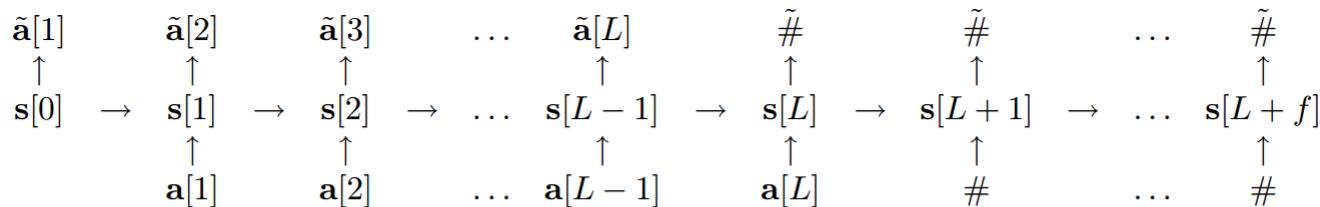


Figure 1: Training, first phase: learning to predict the next ambiguity class. The symbol “~” means *predicted value*; the “#”’s are end-of-sentence markers. Note the trailing sequence of f sentence boundaries, which are needed to tag the f last words of a sentence in second phase.

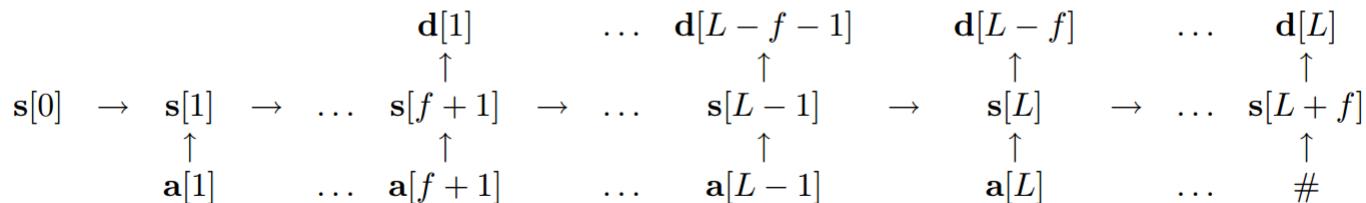
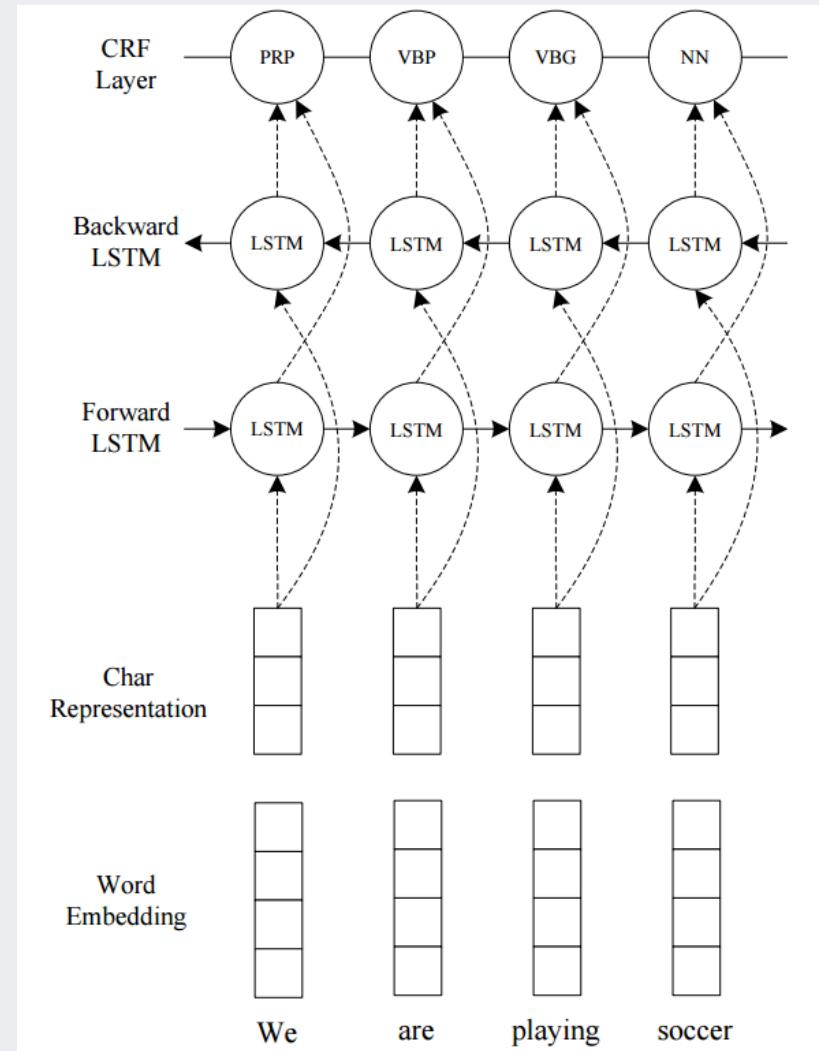
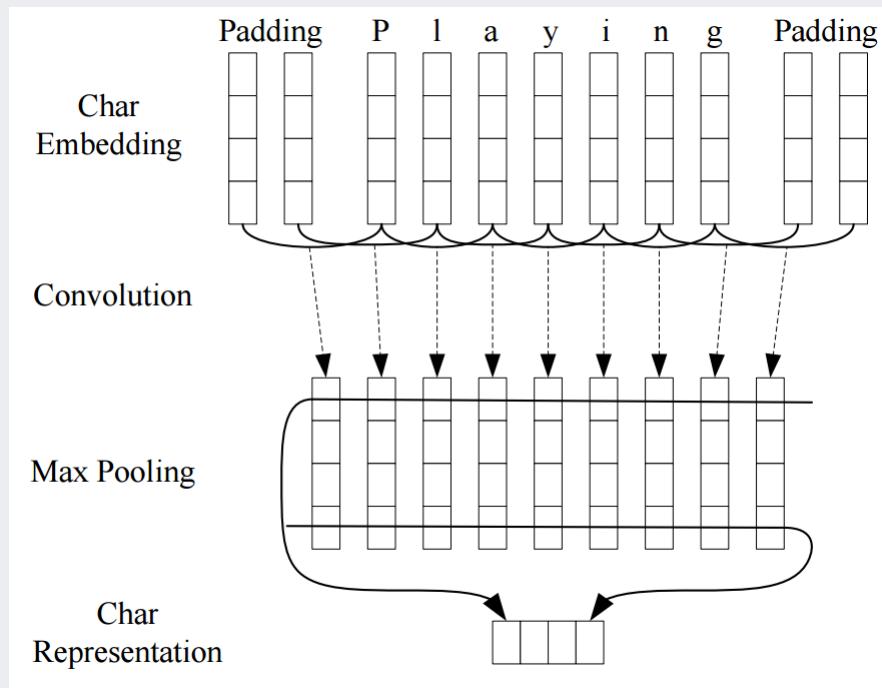


Figure 2: Training, second phase: learning to obtain the correct tag

Lexical Analysis 4: Part-of-Speech (POS) Tagging

Ma and Hovy (2016)

- Hybrid model: LSTM(RNN) + ConvNet + CRF



Lexical Analysis 4: Part-of-Speech (POS) Tagging

Ma and Hovy (2016)

- Hybrid model: LSTM(RNN) + ConvNet + CRF

✓ Experimental results

Dataset		WSJ	CoNLL2003
Train	SENT	38,219	14,987
	TOKEN	912,344	204,567
Dev	SENT	5,527	3,466
	TOKEN	131,768	51,578
Test	SENT	5,462	3,684
	TOKEN	129,654	46,666

Table 2: Corpora statistics. SENT and TOKEN refer to the number of sentences and tokens in each data set.

Model	F1
Chieu and Ng (2002)	88.31
Florian et al. (2003)	88.76
Ando and Zhang (2005)	89.31
Collobert et al. (2011) [‡]	89.59
Huang et al. (2015) [‡]	90.10
Chiu and Nichols (2015) [‡]	90.77
Ratinov and Roth (2009)	90.80
Lin and Wu (2009)	90.90
Passos et al. (2014)	90.90
Luo et al. (2015)	91.20
This paper	91.21

Model	POS		NER					
	Dev	Test	Dev			Test		
			Acc.	Acc.	Prec.	Recall	F1	Prec.
BRNN	96.56	96.76	92.04	89.13	90.56	87.05	83.88	85.44
BLSTM	96.88	96.93	92.31	90.85	91.57	87.77	86.23	87.00
BLSTM-CNN	97.34	97.33	92.18	93.60	92.89	88.77	90.26	89.51
BRNN-CNN-CRF	97.46	97.55	94.85	94.63	94.74	91.35	91.06	91.21

Lexical Analysis 5: Named Entity Recognition

- Named Entity Recognition: NER

- ✓ a subtask of information extraction that seeks to locate and classify elements in text into pre-defined categories such as the names of persons, organizations, locations, expressions of times, quantities, monetary values, percentages, etc.



Lexical Analysis 5: Named Entity Recognition

Approaches for NER: Dictionary/Rule-based

- List lookup: systems that recognizes only entities stored in its lists
 - ✓ Advantages: simple, fast, language independent, easy to retarget.
 - ✓ Disadvantages: collection and maintenance of list cannot deal with name variants and cannot resolve ambiguity
- Shallow Parsing Approach
 - ✓ Internal evidence – names often have internal structure. These components can be either stored or guessed.
 - Location: Cap Word + {Street, Boulevard, Avenue, Crescent, Road}
 - e.g.: Wall Street

Lexical Analysis 5: Named Entity Recognition

Approaches for NER: Model-based

- **MITIE**
 - ✓ An open sourced information extraction tool developed by MIT NLP lab.
 - ✓ Available for English and Spanish
 - ✓ Available for C++, Java, R, and Python
- **CRF++**
 - ✓ NER based on conditional random fields
 - ✓ Supports multi-language models
- **Convolutional neural networks**
 - ✓ I-of-M coding, Word2Vec, N-Grams can be used as encoding methods

AGENDA

01 Introduction to NLP

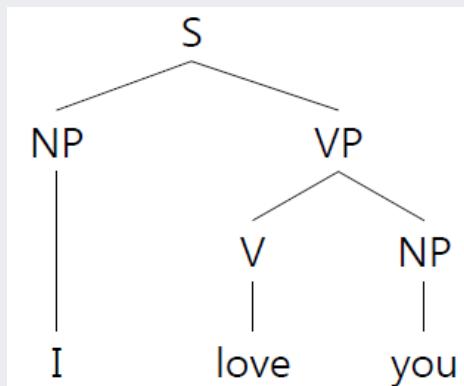
02 Lexical Analysis

03 Syntax Analysis

04 Other Topics in NLP

Syntax Analysis

- Syntax Analysis
 - ✓ Process of analyzing a string of symbols conforming to the rules of a formal grammar

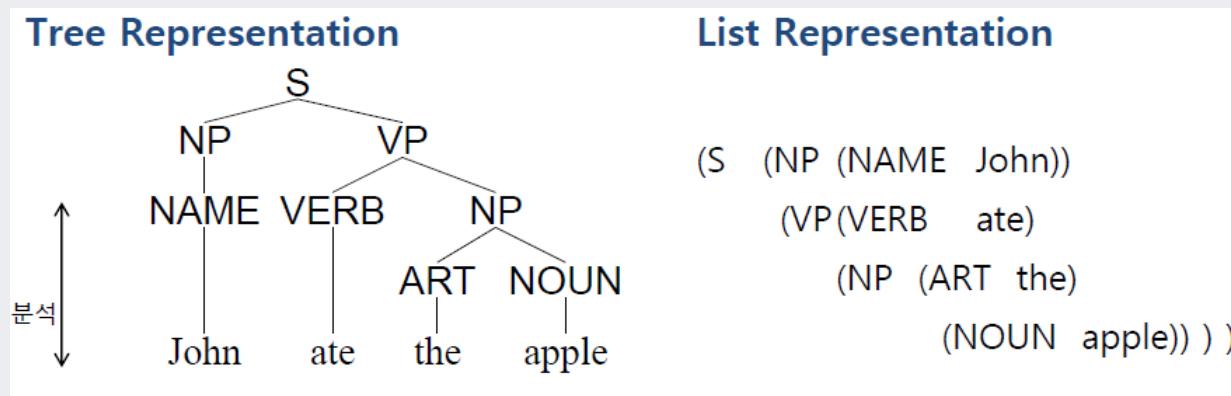


- Parser
 - ✓ An algorithm that computes a structure for an input string given a grammar
 - ✓ All parsers have two fundamental properties
 - **Directionality:** the sequence in which the structures are constructed (e.g., top-down or bottom-up)
 - **Search strategy:** the order in which the search space of possible analysis explored (e.g., depth-first, breadth-first)

Syntax Analysis

- Parsing Representation

- ✓ Tree vs List

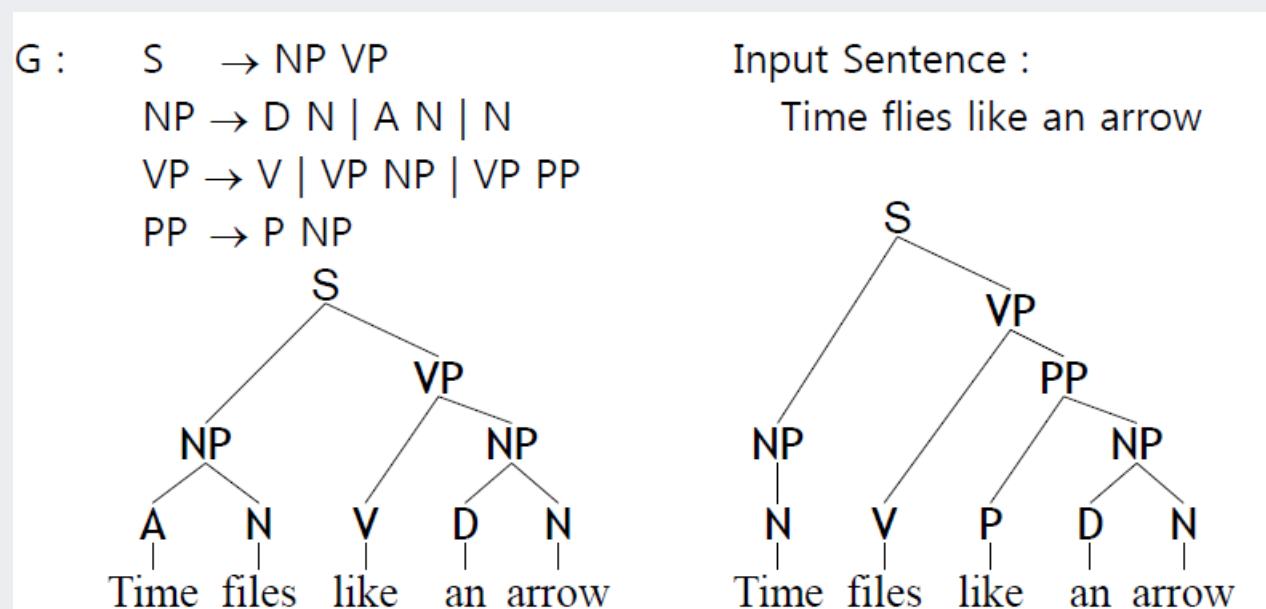


- ✓ Meaning

- S (Sentence) consists of NP (Noun Phrase) and VP (Verb Phrase)
 - NP consists of Name (John)
 - VP consists of VERB (ate) and the other NP
 - NP consists of ART (the) and Noun (apple)

Syntax Analysis

- Not a single parsing tree due to language ambiguity
- Lexical ambiguity
 - ✓ One word can be used for multiple parts of speech
 - ✓ Lexical ambiguity causes structural ambiguity

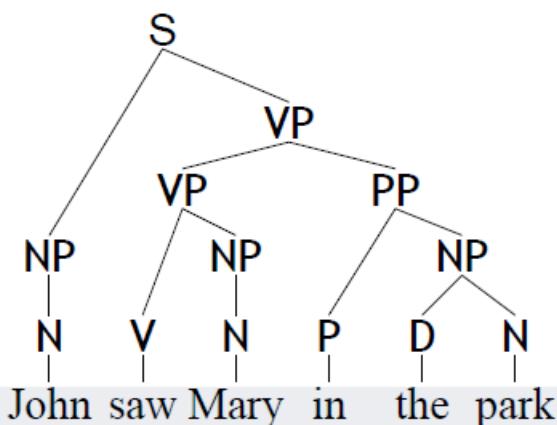


Syntax Analysis

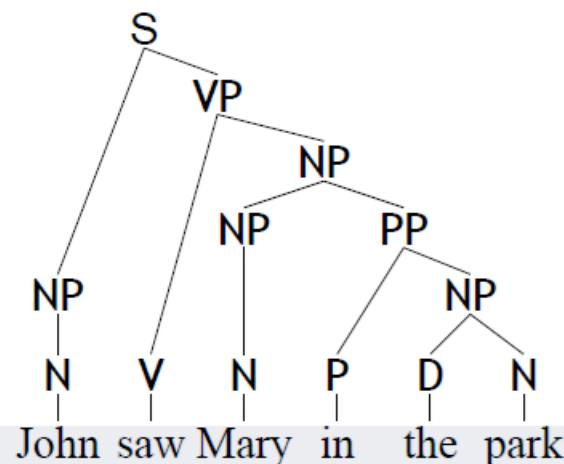
- Structural Ambiguity

- ✓ One sentence can be understood in different ways

G : S \rightarrow NP VP
 NP \rightarrow N | D N | NP PP
 VP \rightarrow V NP | VP PP
 PP \rightarrow P NP



Input Sentence :
John saw Mary in the park.



AGENDA

01 Introduction to NLP

02 Lexical Analysis

03 Syntax Analysis

04 Other Topics in NLP

Language Modeling

Jurafsky, Language Modeling

- Probabilistic Language Model

- ✓ Assign a probability to a sentence (not POS tags, but the sentence itself)

- Applications

- ✓ Machine Translation

- $P(\text{high wind tonight}) > P(\text{large wind tonight})$

- ✓ Spell correction

- The office is about fifteen **minuets** from my house
 - $P(\text{about fifteen minutes from}) > P(\text{about fifteen minuets from})$

- ✓ Speech recognition

- $P(\text{I saw a van}) \gg P(\text{eyes awe of an})$

- ✓ Summarization, question-answering, etc.

Language Modeling

Jurafsky, Language Modeling

- **Probabilistic Language Modeling**

- ✓ Compute the probability of a sentence or sequence of words

$$P(W) = P(w_1, w_2, w_3, w_4, w_5 \dots w_n)$$

- ✓ Related task: probability of an upcoming word

$$P(w_5 | w_1, w_2, w_3, w_4)$$

- ex) I love you more than I can _____? (swim? say?)

- **How to compute $P(W)$**

- ✓ What is $P(\text{its, water, is, so, transparent, that})$?

- ✓ Chain Rules of Probability: $P(x_1, x_2, x_3, \dots, x_n) = P(x_1)P(x_2|x_1)P(x_3|x_1, x_2)\dots P(x_n|x_1, \dots, x_{n-1})$

$P(\text{"its water is so transparent"}) =$

$$\begin{aligned} & P(\text{its}) \times P(\text{water}|\text{its}) \times P(\text{is}|\text{its water}) \\ & \quad \times P(\text{so}|\text{its water is}) \times P(\text{transparent}|\text{its water is so}) \end{aligned}$$

Language Modeling

Jurafsky, Language Modeling

- **Markov Assumption**

- ✓ Consider only k previous words when estimating the conditional probability

$$P(w_1 w_2 \dots w_n) \approx \prod_i P(w_i | w_{i-k} \dots w_{i-1})$$

$$P(w_i | w_1 w_2 \dots w_{i-1}) \approx P(w_i | w_{i-k} \dots w_{i-1})$$

- ✓ Simplest case: Unigram model

$$P(w_1 w_2 \dots w_n) \approx \prod_i P(w_i)$$

- ✓ An example of automatically generated sentences from a unigram model

```
fifth, an, of, futures, the, an, incorporated, a,  
a, the, inflation, most, dollars, quarter, in, is,  
mass
```

```
thrift, did, eighty, said, hard, 'm, july, bullish
```

```
that, or, limited, the
```

Language Modeling

Jurafsky, Language Modeling

- Bigram model

- ✓ Condition on the previous word

$$P(w_i | w_1 w_2 \dots w_{i-1}) \approx P(w_i | w_{i-1})$$

texaco, rose, one, in, this, issue, is, pursuing, growth, in,
a, boiler, house, said, mr., gurria, mexico, 's, motion,
control, proposal, without, permission, from, five, hundred,
fifty, five, yen

outside, new, car, parking, lot, of, the, agreement, reached

this, would, be, a, record, november

- N-gram models

- ✓ Can extend to trigrams, 4-grams, 5-grams

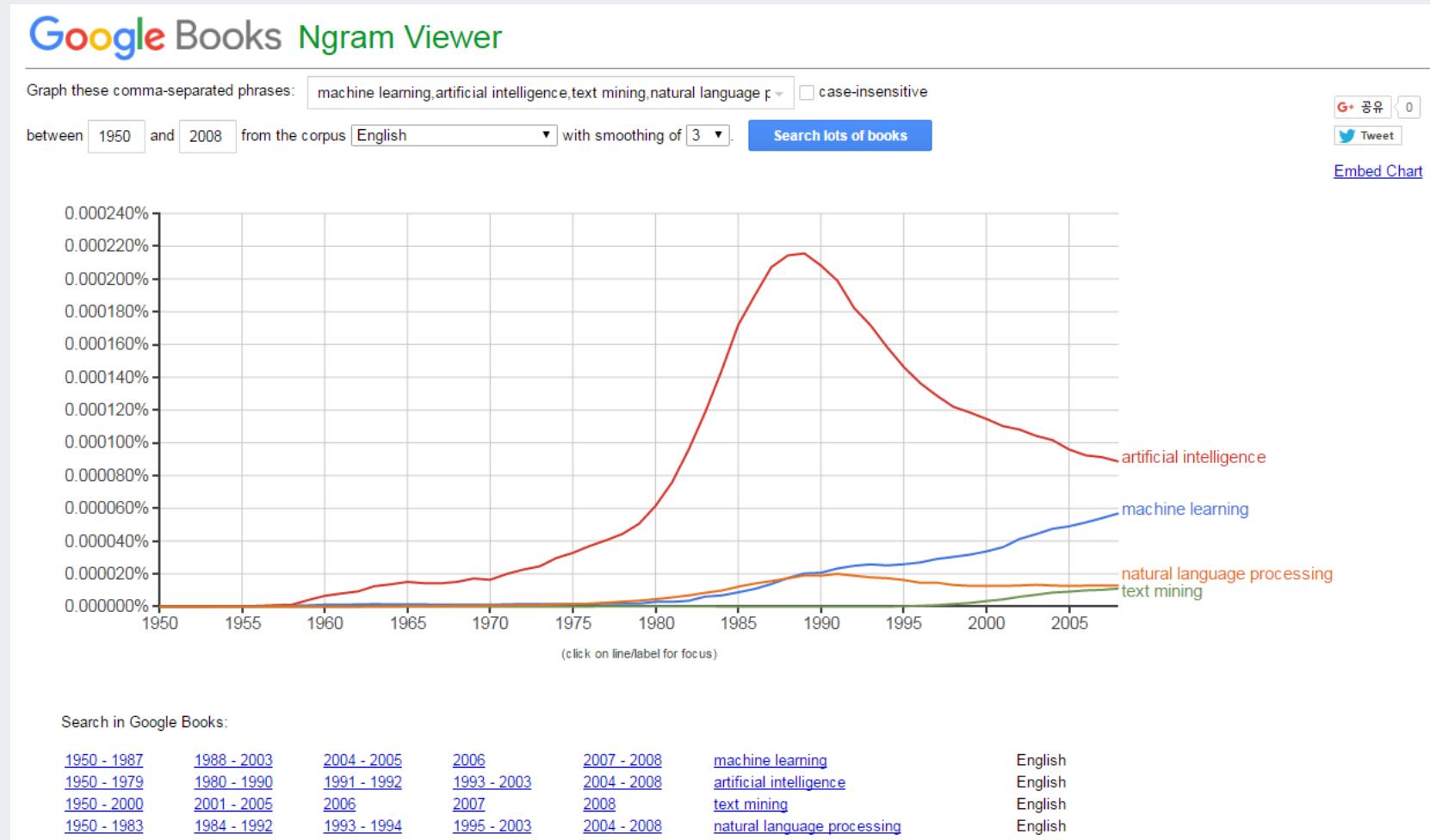
- In sufficient model of language because language has long-distance dependencies
- “**The computer** when I had just put into the machine room on the fifth floor **crashed**.”

- ✓ We can often get away with N-gram models

Language Modeling

- Google Books N-Gram

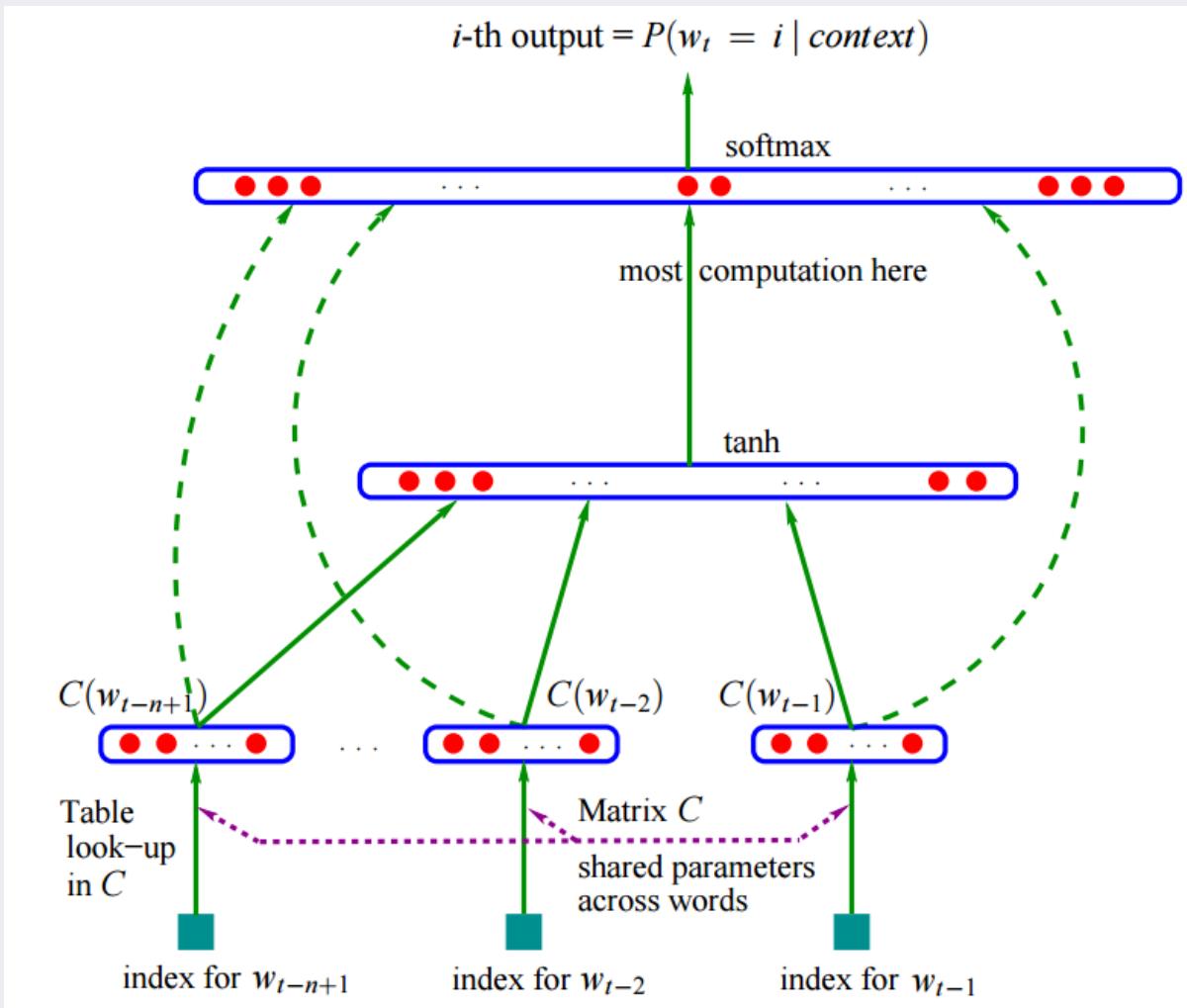
- ✓ 1,024 billion words & 1.1 billion 5-grams that appeared at least 40 times (2006)



Language Modeling

Bengio et al. (2003)

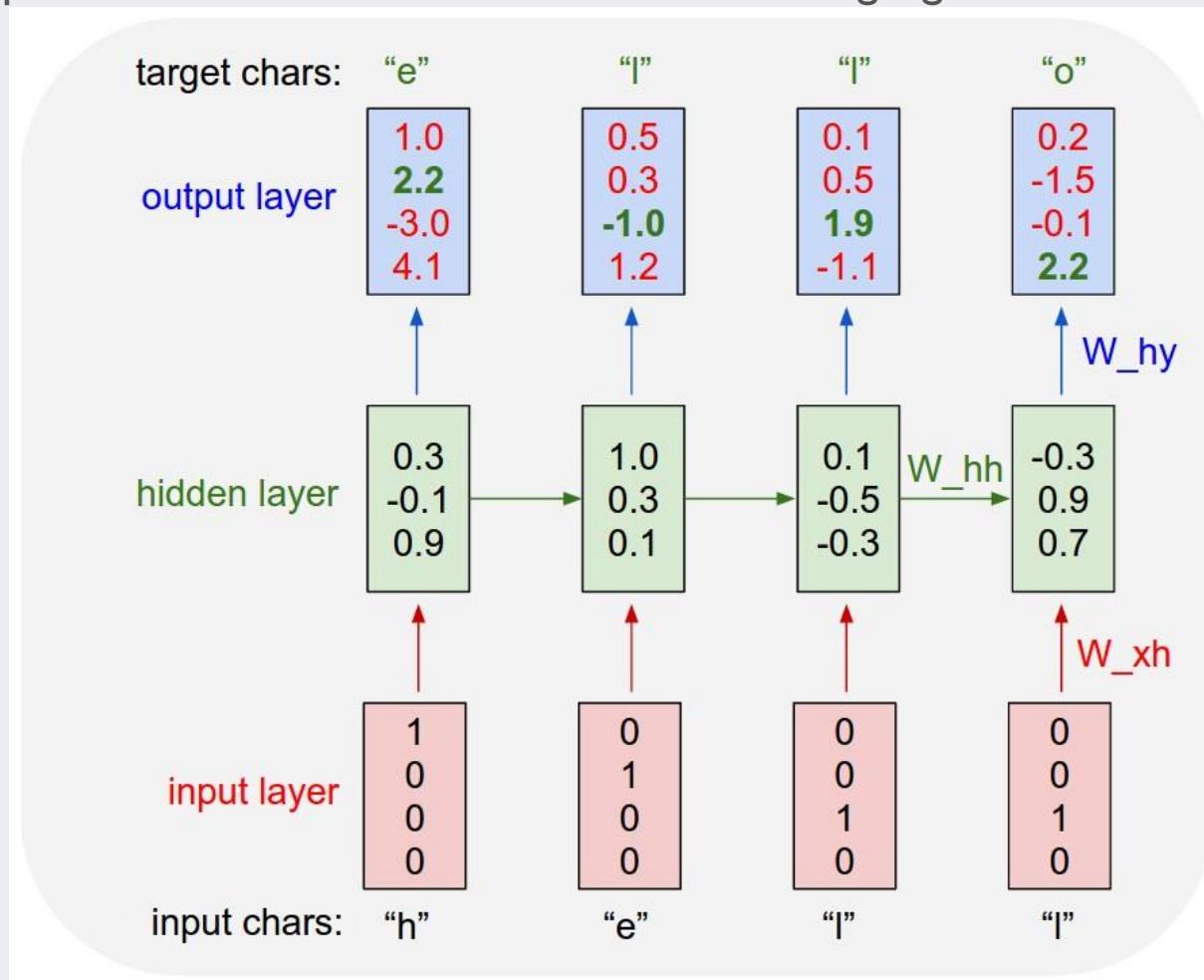
- Neural Network-based Language Model



Language Modeling

Mikolov et al. (2010)

- Recurrent Neural Network (RNN)-based Language Model
 - ✓ A simplified RNN structure for character-level language model



Language Modeling

- Recurrent Neural Network (RNN)-based Language Model
 - ✓ Character-level RNN vs Word-level RNN

Char-RNN

ESCALUS:
What is our honours, such a Richard story
Which you mark with bloody been Thillld we'll adverses:
That thou, Aurtructs a greques' great
Jmander may to save it not shif theseen my news
Clisters it take us?
Say the dulterout apy showd. They hance!

AnBESS OF GUCESTER:
Now, glarding far it prick me with this queen.
And if thou met were with revil, sir?

KATHW:
I must not my naturation disery,
And six nor's mighty wind, I fairs, if?

Messenger:
My lank, nobles arms;

<https://github.com/hunkim/word-rnn-tensorflow>

Word-RNN

LEONTES:
Why, my Irish time?
And argue in the lord; the man mad, must be deserved a spirit as drown the warlike Pray him, how seven in.
KING would be made that, methoughts I may married a Lord dishonour
Than thou that be mine kites and sinew for his honour
In reason prettily the sudden night upon all shalt bid him thus again. times than one from mine unaccustom'd sir.

LARTIUS:
O,'tis aediles, fight!
Farewell, it himself have saw.

SLY:
Now gods have their VINCENTIO:
Whipt fearing but first I know you you, hinder truths.

ANGELO:
This are entitle up my dearest state but deliver'd.

DUKE look dissolved: seemeth brands
That He being and
full of toad, they knew me to joy.

Language Modeling

Lee (2017)

- Recurrent Neural Network (RNN)-based Language Model

- ✓ Character-level RNN (Korean)

- 이광수 장편소설 「무정」(총 323,660 음절, 1,680개 단어)
 - 특징: 1917년 작품이라 한자어가 많이 쓰였음, 큰따옴표와 줄바꿈을 포함한 대화체 문장이 많으며, 중고교생 대상으로 읽히는 작품이라 중간중간 괄호 속에 편집자 주석이 끼어 있음

형식은, 아뿔싸! 내가 어찌하여 이러한 생각을 하는가, 내 마음이 이렇게 약하던가 하면서 두 주먹을 불끈 쥐고 전신에 힘을 주어 이러한 약한 생각을 떼어 버리려 하나, 가슴속에는 이상하게 불길이 확확 일어난다. 이때에,

“미스터 리, 어디로 가는가” 하는 소리에 깜짝 놀라 고개를 들었다. (중략) 형식은 얼마큼 마음에 수치한 생각이 나서 고개를 돌리며, “아직 그런 말에 익숙지를 못해서……” 하고 말끝을 못 맺는다. “대관절 어디로 가는 길인가? 급지 않거든 점심이나 하세그려.”

“점심은 먹었는걸.”

“그러면 맥주나 한잔 먹지.”

“내가 술을 먹는가.”

(중략)

“요— 오메데토오(아— 축하하네). 이이나즈케(약혼한 사람)가 있나 보네그려. 음 나루호도(그려려니). 그러구도 내게는 아무 말도 없단 말이야. 예, 여보게” 하고 손을 후려친다.

Language Modeling

Lee (2017)

Iter 0 :

랫萬게좁뉘뿔름끈玄른작밭裸觀갈나맡文풀조바눔形伍下잇별흘툁혈調記운피悲렙司狼독벗칼둡걷착날完잣老
엇낫業4改'촉수를낮뱅잇쯤죽道년友련친씩았용타雲채發造거크휘亨律與命텐암면형평琵해落유 리벤產이馨텐

Iter 1300 : 를 웃 사가 려만다밤 말어변 대니 심로 려이, 순 과 이을 죄사글를 . 사람을 영채와 이니아베을 니러,
다가 달고 면 를 아잘 하 기 성구을 을 실틈으로 아잠 고 이 그와 매못 더 (띄어쓰기)

Iter 4900 : 를 왔다내 루방덩이종 은 얼에는 집어흔영채는 아무 우선을 에서가며 건들하아버전는 애양을 자에
운 모양이 랐다. 은 한다선과 '마는 .식세식가들어 ,

형식다

"내었다.있이 문 (줄바꿈)

Iter 100000 : 면서 치현분들더 중 한통 선교잤다.

"처럼 우셨다시가..... 것이 말사도? 여자려겠습니다" 하는 마음(裸生)은 이런 적드렸다. 그 말이 얼굴이 딸로
나고 얼굴이 마음불 하고 따라 선

Iter 300000 : 씻었다. 선형은 형식의 형식은 빛이 가슴을 오고 걸현감에는 일이 는 눈과 의고 아이얗어 알으로
자기의 구원을 내어려가 여러 짓을 쾌처게 안아 말고였는 악한 순간에 속으로 두 학교에

Iter 500000 : 본다. 성학과 평양으로 새로도 처음의 타던 공격하였다. '영채의 꽁인의 생각을 하면 때에 기생의
이는 것 보더니 나는 듯이 제인 소세건과 영채의 모양이를 대하였다. 형식을 생각하여

Iter 750000 : 으로 유안하였다. 더할까 하는 세상이 솔이요, 알고 게식도 들어울는 듯하였다. 태에그려 깔깔고
웃는 듯이 흔반다. 우선형은 사람을 어려보낸다.

"그려가?" (간접 인용)

한다. 영채는 손을 기쁘

Iter 1000000 : 에 돌내면서,

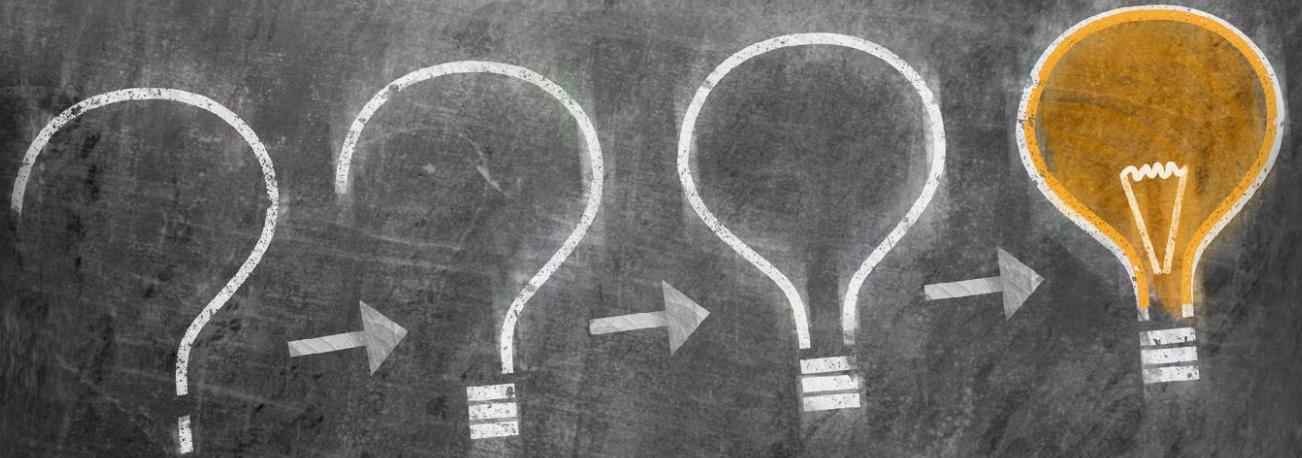
"여러 넣어오습데다. 그 말대 아무도 좀 집림과 시오 백매, 저는 열녀더러, 기린 소년이가 아니라."

"어리지요."

노파도 놀라며,

"저희마다가 말없습니까."

"아니 (대화체)



References

Research Papers

- Bengio, Y., Ducharme, R., Vincent, P., & Jauvin, C. (2003). A neural probabilistic language model. *Journal of machine learning research*, 3(Feb), 1137-1155.
- Bollen, J., Mao, H., Zeng, X. (2011). Twitter Mood Predicts the Stock Market, *Journal of Computational Science*, 2(1): 1-8.
- Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12, 2493-2537.
- Hirschberg, J., & Manning, C. D. (2015). Advances in natural language processing. *Science*, 349(6245), 261-266.
- Johnson, M., Schuster, M., Le, Q.V., Krikun, M., Wu, Y., Chen, Z., ... & Hughes, M. (2016). Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation. *arXiv preprint arXiv:1611.04558*.
- Kim, T., Hong, J., & Kang, P. (2015). Box office forecasting using machine learning algorithms based on SNS data. *International Journal of Forecasting*, 31(2), 364-390.
- Ma, X., & Hovy, E. (2016). End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. *arXiv:1603.01354*.
- Mikolov, T., Karafiat, M., Burget, L., Cernocky, J., & Khudanpur, S. (2010, September). Recurrent neural network based language model. In *Interspeech* (Vol. 2, p. 3).
- Perez-Ortiz, J.A., & Forcada, M. L. (2001). Part-of-speech tagging with recurrent neural networks. Universitat d'Alacant, Spain.

References

Research Papers

- Socher, R., Perelygin, A., Wu, J.Y., Chuang, J., Manning, C. D., Ng, A.Y., & Potts, C. (2013, October). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the conference on empirical methods in natural language processing (EMNLP)* (Vol. 1631, p. 1642).
- Wu, Yonghui, et al. "Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation." *arXiv preprint arXiv:1609.08144* (2016).

References

Other materials

- Jurafsky, D. Language Modeling. Lecture Slides from the Stanford Coursera course. <http://spark-public.s3.amazonaws.com/nlp/slides/languagemodeling.pdf>
- Jurafsky, D. Sentiment Analysis. Lecture Slides from the Stanford Coursera course. <http://spark-public.s3.amazonaws.com/nlp/slides/sentiment.pdf>
- Koehn, P. (2013). Advanced Natural Language Processing Lecture 12: Parsing. School of Informatics. <http://www.inf.ed.ac.uk/teaching/courses/anlp/slides/anlp12.pdf>
- Koehn, P. (2013). Advanced Natural Language Processing Lecture 13: Parsing (Continued). School of Informatics. <http://www.inf.ed.ac.uk/teaching/courses/anlp/slides/anlp13.pdf>
- Lee, G. (2017). Recurrent neural network & long short-term memory models. DSBA Lab seminar (cs231n).
- Neubig, G. Part of Speech Tagging with Hidden Markov Models. <http://www.phontron.com/slides/nlp-programming-en-05-hmm.pdf>
- Witte, R. (2006). Introduction to Text Mining. Tutorial at EDBT'06. <http://rene-witte.net/text-mining-tutorial-edbt2006>
- <http://web.stanford.edu/class/archive/cs/cs143/cs143.1128/lectures/01/Slides01.pdf>