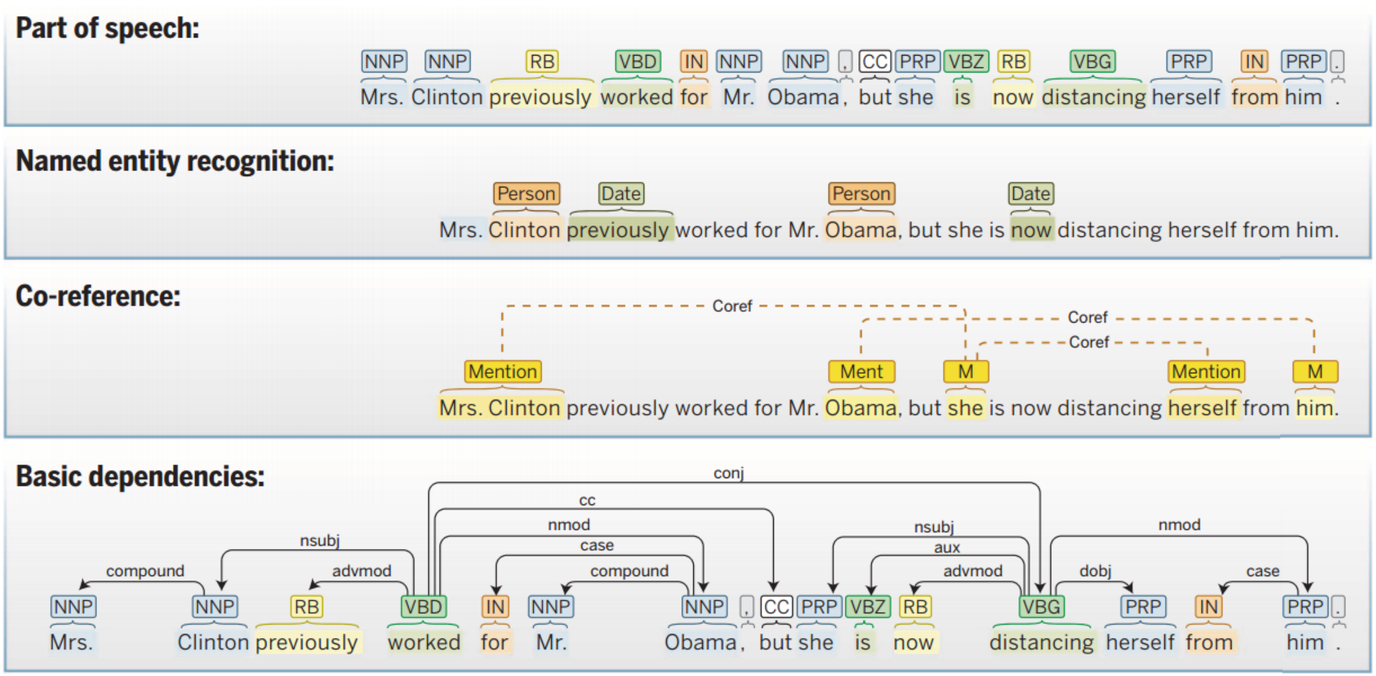


Lecture 2: Text Preprocessing

Pilsung Kang

School of Industrial Management Engineering

Korea University

AGENDA

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

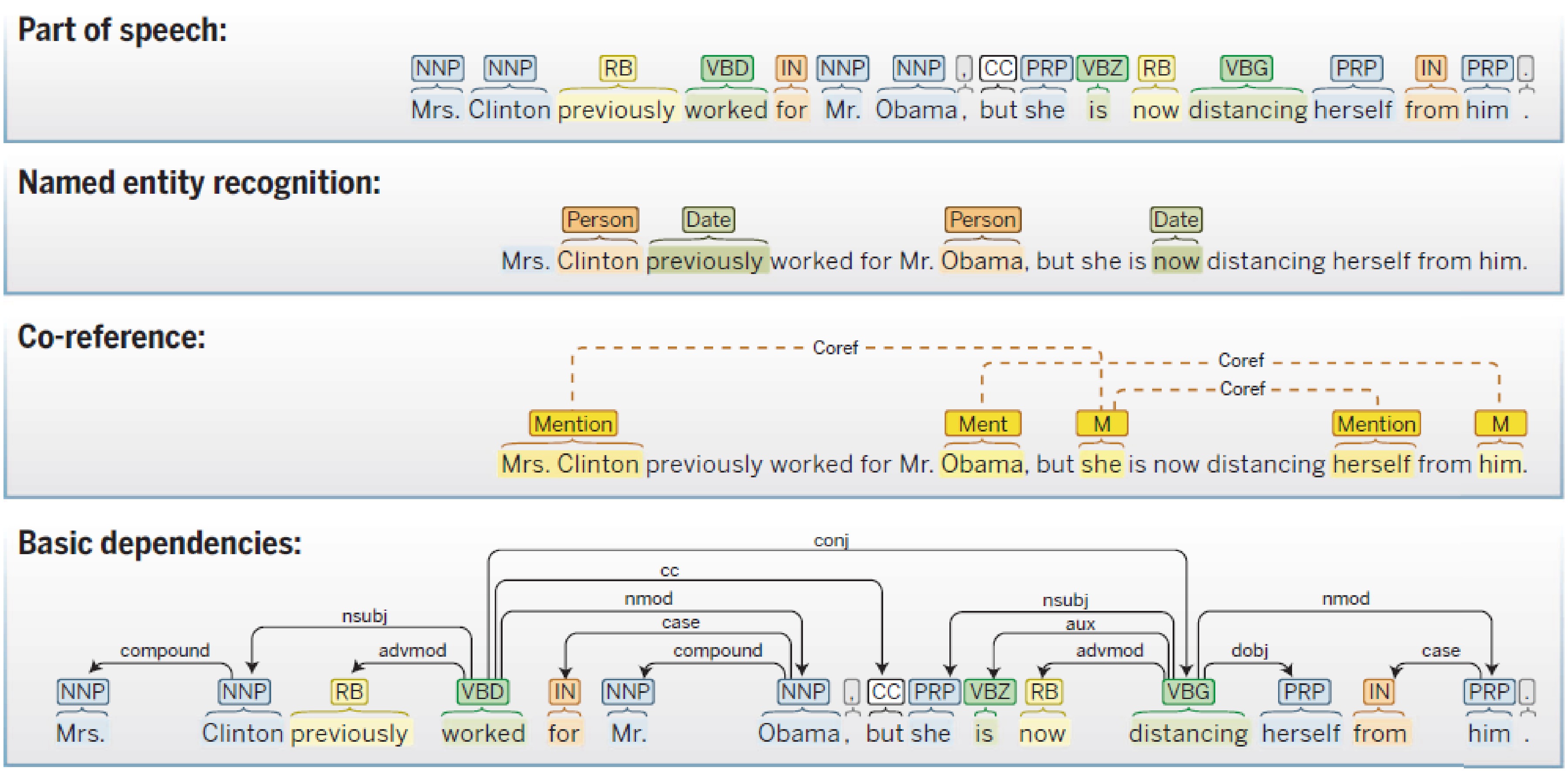


Fig. 1. Many language technology tools start by doing linguistic structure analysis. Here we show output from Stanford CoreNLP. As shown from top to bottom, this tool determines the parts of speech of each word, tags various words or phrases as semantic named entities of various sorts, determines which 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 probabilites 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, …

```
> crude[[1]]
<<PlainTextDocument (metadata: 15)>>
Diamond Shamrock Corp said that
effective today it had cut its contract prices for crude oil by
1.50 dlrs a barrel.
    The reduction brings its posted price for West Texas
Intermediate to 16.00 dlrs 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
```

```
> MC_tokenizer(crude[[1]])
  [1] "Diamond"      "Shamrock"     "Corp"         "said"         "that"
  [6] "effective"    "today"        "it"           "had"          "cut"
 [11] "its"          "contract"     "prices"       "for"          "crude"
 [16] "oil"          "by"           ""             ""             ""
 [21] ""             ""             "dlrs"         "a"            "barrel"
 [26] ""             ""             ""             ""             ""
 [31] "The"          "reduction"    "brings"       "its"          "posted"
 [36] "price"        "for"          "West"         "Texas"        "Intermediate"
 [41] "to"           ""             ""             ""             ""
 [46] ""             ""             "dlrs"         "a"            "barrel"
 [51] ""             "the"          "copany"       "said"         ""
 [56] ""             ""             ""             ""             ""
 [61] "The"          "price"        "reduction"    "today"        "was"
 [66] "made"         "in"           "the"          "light"        "of"
 [71] "falling"      "oil"          "product"      "prices"       "and"
 [76] "a"            "weak"         "crude"        "oil"          "market"
 [81] ""             ""             "a"            "company"      "spokeswoman"
 [86] "said"         ""             ""             ""             ""
 [91] ""             "Diamond"      "is"           "the"          "latest"
 [96] "in"           "a"            "line"         "of"           "U"
[101] "S"            ""             "oil"          "companies"    "that"
[106] "have"         "cut"          "its"          "contract"     ""
[111] "or"           "posted"       ""             "prices"       "over"
[116] "the"          "last"         "two"          "days"         "citing"
[121] "weak"         "oil"          "markets"      ""             ""
[126] "Reuter"
```

```
> scan_tokenizer(crude[[1]])
 [1] "Diamond"      "Shamrock"     "Corp"         "said"         "that"
 [6] "effective"    "today"        "it"           "had"          "cut"
[11] "its"          "contract"     "prices"       "for"          "crude"
[16] "oil"          "by"           "1.50"         "dlrs"         "a"
[21] "barrel."      "The"          "reduction"    "brings"       "its"
[26] "posted"       "price"        "for"          "West"         "Texas"
[31] "Intermediate" "to"           "16.00"        "dlrs"         "a"
[36] "barrel,"      "the"          "copany"       "said."        "\"The"
[41] "price"        "reduction"    "today"        "was"          "made"
[46] "in"           "the"          "light"        "of"           "falling"
[51] "oil"          "product"      "prices"       "and"          "a"
[56] "weak"         "crude"        "oil"          "market,\""    "a"
[61] "company"      "spokeswoman"  "said."        "Diamond"      "is"
[66] "the"          "latest"       "in"           "a"            "line"
[71] "of"           "U.S."         "oil"          "companies"    "that"
[76] "have"         "cut"          "its"          "contract,"    "or"
[81] "posted,"      "prices"       "over"         "the"          "last"
[86] "two"          "days"         "citing"       "weak"         "oil"
[91] "markets."     "Reuter"
```

	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* $\longrightarrow$ cars, *give* $\longrightarrow$ *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 dlrs a barrel.
    The reduction brings its posted price for West Texas
Intermediate to 16.00 dlrs 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
```

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  dlrs 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
```

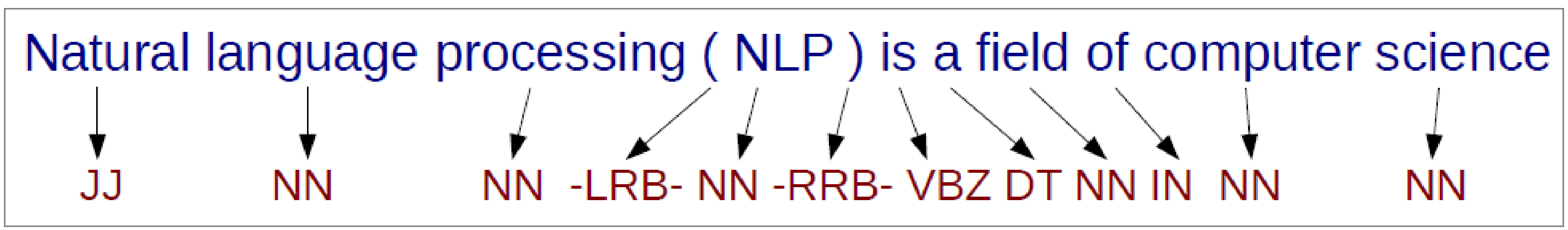
Lemmatization

```
> LemmaCorpus1
[1] "diamond shamrock corp say that effective today it have
cut it contract price for crude oil by dlr a barrel the redu
ction 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... $\rightarrow$ *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	and, or, but
CD	cardinal number	five, three, 13%
DT	determiner	the, a, these
EX	existential there	there were six boys
FW	foreign word	mais
IN	conjunction, subordinating or preposition	of, on, before, unless
JJ	adjective	nice, easy
JJR	adjective, comparative	nicer, easier
JJS	adjective, superlative	nicest, easiest
LS	list item marker	
MD	verb, modal auxillary	may, should
NN	noun, singular or mass	tiger, chair, laughter
NNS	noun, plural	tigers, chairs, insects
NNP	noun, proper singular	Germany, God, Alice
NNPS	noun, proper plural	we met two Christmases ago
PDT	predeterminer	both his children
POS	possessive ending	's
PRP	pronoun, personal	me, you, it
PRP$	pronoun, possessive	my, your, our
RB	adverb	extremely, loudly, hard
RBR	adverb, comparative	better
RBS	adverb, superlative	best
RP	adverb, particle	about, off, up
SYM	symbol	%
TO	infinitival to	what to do?
UH	interjection	oh, oops, gosh
VB	verb, base form	think
VBZ	verb, 3rd person singular present	she thinks
VBP	verb, non-3rd person singular present	I think
VBD	verb, past tense	they thought
VBN	verb, past participle	a sunken ship
VBG	verb, gerund or present participle	thinking is fun
WDT	*wh*-determiner	which, whatever, whichever
WP	*wh*-pronoun, personal	what, who, whom
WP$	*wh*-pronoun, possessive	whose, whosever
WRB	*wh*-adverb	where, when
.	punctuation mark, sentence closer	.;?*
,	punctuation mark, comma	,
:	punctuation mark, colon	:
(	contextual separator, left paren	(
)	contextual separator, right paren	)

UCREL CLAWS7 Tagset

APPGE	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 (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)
DDQGE	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)
MCMC	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)
ND1	singular noun of direction (e.g. north, southeast)
NN	common noun, neutral for number (e.g. sheep, cod, headquarters)
NN1	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	numeral noun, neutral for number (e.g. dozen, hundred)
NNO2	numeral 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)
PPGE	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	확률태그	저장사전
체언	NNG	일반 명사	NN	명사	N	NN	NNG	보통 명사	NNA	noun.dic
	NNP	고유 명사					NNP	고유 명사		
	NNB	의존 명사	NX	의존 명사			NNB	일반 의존 명사	NNB	simple.dic
			UM	단위 명사			NNM	단위 의존 명사		
	NR	수사	NU	수사		NR	NR	수사	NR	
	NP	대명사	NP	대명사		NP	NP	대명사	NP	
용언	VV	동사	VV	동사	V	VV	VV	동사	VV	verb.dic
	VA	형용사	AJ	형용사		VA	VA	형용사	VA	
	VX	보조 용언	VX	보조 동사		VX	VXV	보조 동사	VX	
			AX	보조 형용사			VXA	보조 형용사		
	VCP	긍정 지정사	CP	서술격 조사 '이다'		VC	VCP	긍정 지정사, 서술격 조사 '이다'	VCP	raw.dic
	VCN	부정 지정사					VCN	부정 지정사, 형용사 '아니다'	VCN	
관형사	MM	관형사	DT	일반 관형사	M	MD	MDT	일반 관형사	MD	simple.dic
			DN	수 관형사			MDN	수 관형사		
부사	MAG	일반 부사	AD	부사		MA	MAG	일반 부사	MAG	
	MAJ	접속 부사					MAC	접속 부사	MAC	
감탄사	IC	감탄사	EX	감탄사	I	IC	IC	감탄사	IC	
조사	JKS	주격 조사	JO	조사	J	JK	JKS	주격 조사	JKS	
	JKC	보격 조사					JKC	보격 조사	JKC	
	JKG	관형격 조사					JKG	관형격 조사	JKG	
	JKO	목적격 조사					JKO	목적격 조사	JKO	
	JKB	부사격 조사					JKM	부사격 조사	JKM	
	JKV	호격 조사					JKI	호격 조사	JKI	
	JKQ	인용격 조사					JKQ	인용격 조사	JKQ	
	JX	보조사				JX	JX	보조사	JX	
	JC	접속 조사				JC	JC	접속 조사	JC	
선어말 어미	EP	선어말 어미	EP	선어말 어미		EP	EPH	존칭 선어말 어미	EP	
							EPT	시제 선어말 어미		
							EPP	공손 선어말 어미		raw.dic
어말 어미	EF	종결 어미	EM	어말 어미	E	EF	EFN	평서형 종결 어미	EF	
							EFQ	의문형 종결 어미		
							EFO	명령형 종결 어미		
							EFA	청유형 종결 어미		
							EFI	감탄형 종결 어미		
							EFR	존칭형 종결 어미		
	EC	연결 어미				EC	ECE	대등 연결 어미	EC	
							ECD	의존적 연결 어미		
							ECS	보조적 연결 어미		
	ETN	명사형 전성 어미				ET	ETN	명사형 전성 어미	ETN	
	ETM	관형형 전성 어미					ETD	관형형 전성 어미	ETD	
접두사	XPN	체언 접두사	PF	접두사	X	XP	XPN	체언 접두사	XPN	simple.dic
							XPV	용언 접두사	XPV	
접미사	XSN	명사 파생 접미사	SN	명사화 접미사		XS	XSN	명사 파생 접미사	XSN	
	XSV	동사 파생 접미사	SV	동사화 접미사			XSV	동사 파생 접미사	XSV	
	XSA	형용사 파생 접미사	SJ	형용사화 접미사			XSA	형용사 파생 접미사	XSA	
			SA	부사화 접미사			~~XSM~~	~~부사 파생 접미사~~	~~XSM~~	
			SF	기타 접미사			~~XSO~~	~~기타 접미사~~	~~XSO~~	
어근	XR	어근	XR			XR	XR	어근	XR	
부호	SF	마침표물음표,느낌표	SY	부호 외래어	S	SF	SF	마침표물음표,느낌표	SF	Symbol class
	SP	쉼표,가운뎃점, 콜론,빗금				SP	SP	쉼표,가운뎃점, 콜론,빗금	SP	
	SS	따옴표,괄호표, 줄표				SS	SS	따옴표,괄호표, 줄표	SS	
	SE	줄임표				SE	SE	줄임표	SE	
	SO	붙임표(물결, 숨김,빠짐)				SO	SO	붙임표(물결, 숨김,빠짐)	SO	
	SW	기타기호 (논리수학기호, 화폐기호)				SW	SW	기타기호 (논리수학기호, 화폐기호)	SW	
분석 불능	NF	명사추정범주	NR	미등록어	U	UN	UN	명사추정범주	NNA	N/A
	NV	용언추정범주				UV	~~UV~~	~~용언추정범주~~	~~N/A~~	
	NA	분석불능범주				UE	~~UE~~	~~분석불능범주~~	~~N/A~~	
한글 이외	SL	외국어			O	OL	OL	외국어	NNA	
	SH	한자				OH	OH	한자	NNA	
	SN	숫자				ON	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

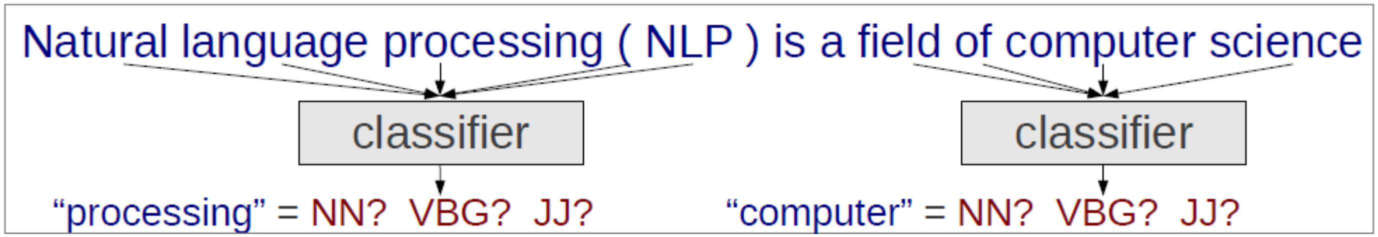
There is a multitude of approaches, commonly used are:

- 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

$$p(t|C) = \frac{1}{Z(C)} \exp\Big(\sum_{i=1}^{n} \lambda_i f_i(C, t) \Big) \qquad p(t_1, \ldots, t_n | w_1, \ldots, w_n) \approx \prod_{i=1}^{n} p(t_i | w_i)$$

    - f_i is a feature
    - λi 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

```
# POS Tagging with MaxEnt
install.packages("openNLP")
library(openNLP)

s1 <- paste(c("Pierre Vinken, 61 years old, will join the board as a ",
              "nonexecutive director Nov. 29.\n",
              "Mr. Vinken is chairman of Elsevier N.V., ",
              "the Dutch publishing group."),
            collapse = "")
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

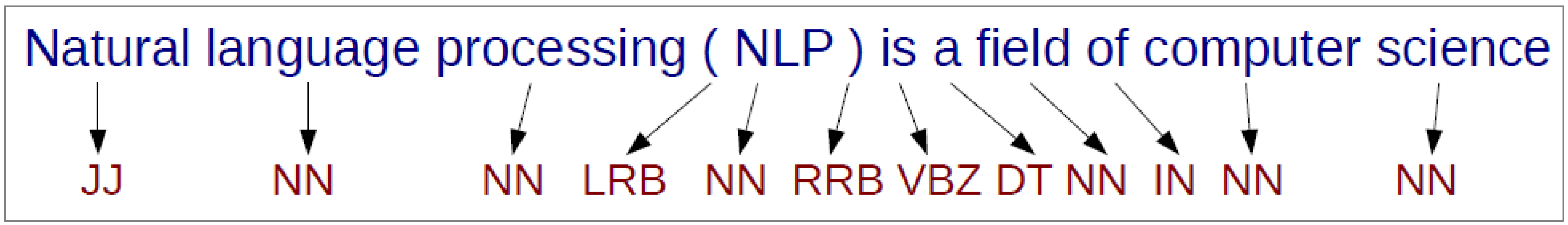
```r
## Need sentence and word token annotations.
s2 <- annotate(s1, list(Maxent_Sent_Token_Annotator(), Maxent_Word_Token_Annotator()))

## POS tag probabilities as (additional) features.
s3 <- annotate(s1, Maxent_POS_Tag_Annotator(probs = TRUE), s2)
```

```
> s3
 id type     start end features
  1 sentence     1  84 constituents=<<integer,18>>
  2 sentence    86 153 constituents=<<integer,13>>
  3 word         1   6 POS=NNP, POS_prob=0.9476405
  4 word         8  13 POS=NNP, POS_prob=0.9692841
  5 word        14  14 POS=,, POS_prob=0.9884445
  6 word        16  17 POS=CD, POS_prob=0.9926943
  7 word        19  23 POS=NNS, POS_prob=0.9893489
  8 word        25  27 POS=JJ, POS_prob=0.9693832
  9 word        28  28 POS=,, POS_prob=0.9873552
 10 word        30  33 POS=MD, POS_prob=0.9460105
 11 word        35  38 POS=VB, POS_prob=0.9865564
 12 word        40  42 POS=DT, POS_prob=0.9692801
 13 word        44  48 POS=NN, POS_prob=0.9928681
 14 word        50  51 POS=IN, POS_prob=0.9592474
 15 word        53  53 POS=DT, POS_prob=0.9890297
 16 word        55  66 POS=JJ, POS_prob=0.7213763
 17 word        68  75 POS=NN, POS_prob=0.987327
 18 word        77  80 POS=NNP, POS_prob=0.9581523
 19 word        82  83 POS=CD, POS_prob=0.9502215
 20 word        84  84 POS=., POS_prob=0.9943433
 21 word        86  88 POS=NNP, POS_prob=0.9762001
 22 word        90  95 POS=NNP, POS_prob=0.9904051
 23 word        97  98 POS=VBZ, POS_prob=0.9820713
 24 word       100 107 POS=NN, POS_prob=0.8300819
 25 word       109 110 POS=IN, POS_prob=0.9838273
 26 word       112 119 POS=NNP, POS_prob=0.9231359
 27 word       121 124 POS=NNP, POS_prob=0.9969889
 28 word       125 125 POS=,, POS_prob=0.9762171
 29 word       127 129 POS=DT, POS_prob=0.9811851
 30 word       131 135 POS=JJ, POS_prob=0.8021723
 31 word       137 146 POS=NN, POS_prob=0.9669352
 32 word       148 152 POS=NN, POS_prob=0.9940887
 33 word       153 153 POS=., POS_prob=0.9898899
```

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

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



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

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

- Generative Sequence Model
  - ✓ Decompose probability using Baye's Rule

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

Model of word/POS interactions
"natural" is probably a JJ

Model of POS/POS interactions
NN comes after DET

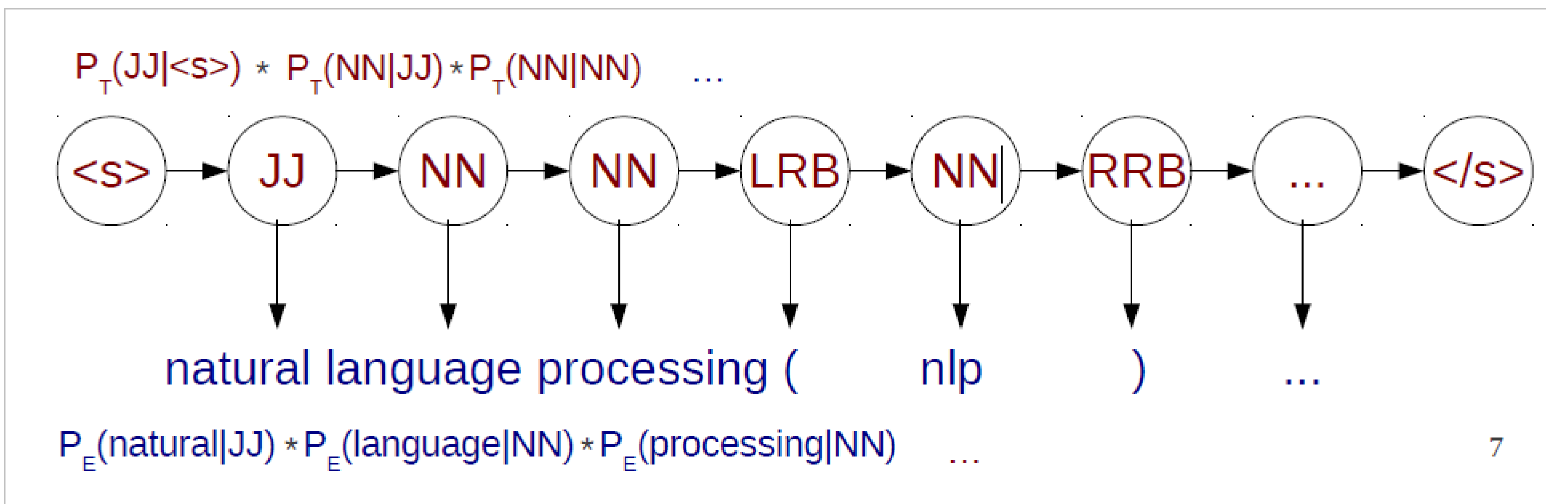
Lexical Analysis 4: Part-of-Speech (POS) Tagging

- Generative Sequence Model: Hidden Markov Model
  - ✓ POS → POS transition probabilities

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

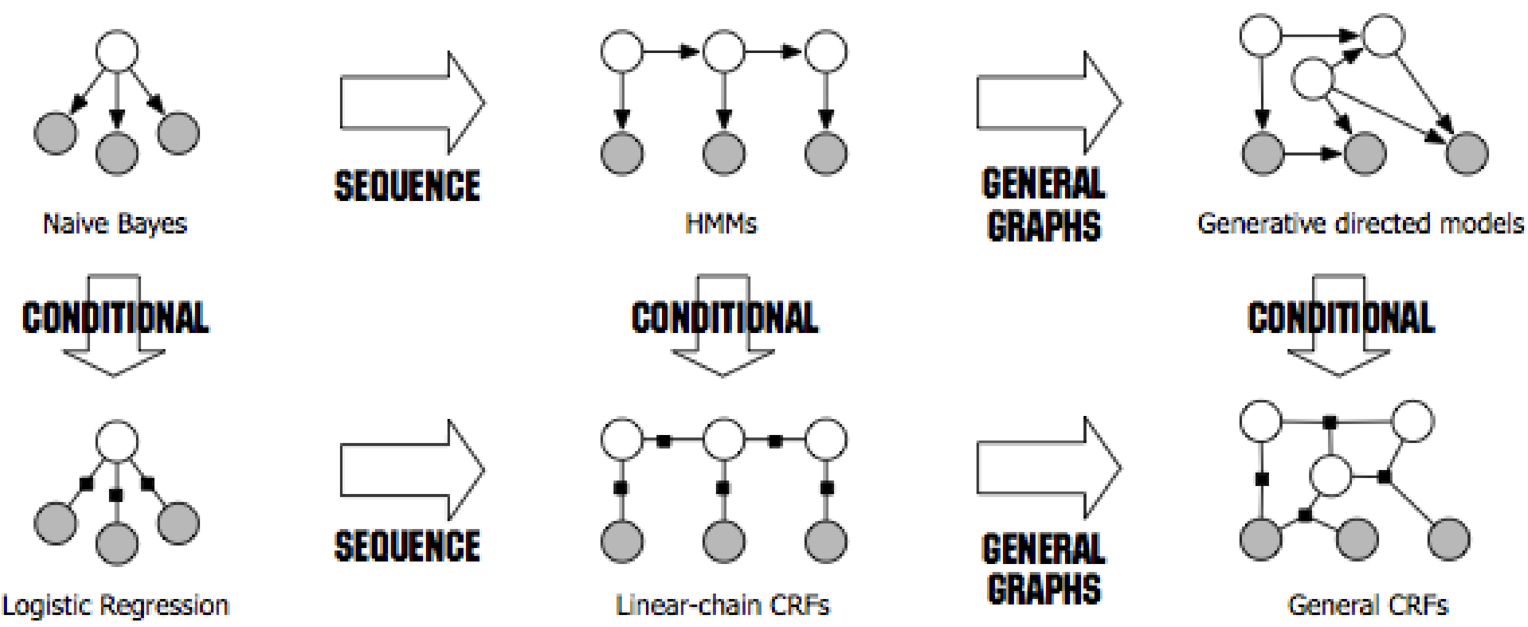
  - ✓ POS → Word emission probabilities

$$P(X|Y) \approx \prod_{1}^{l} P_E(x_i | y_i)$$



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

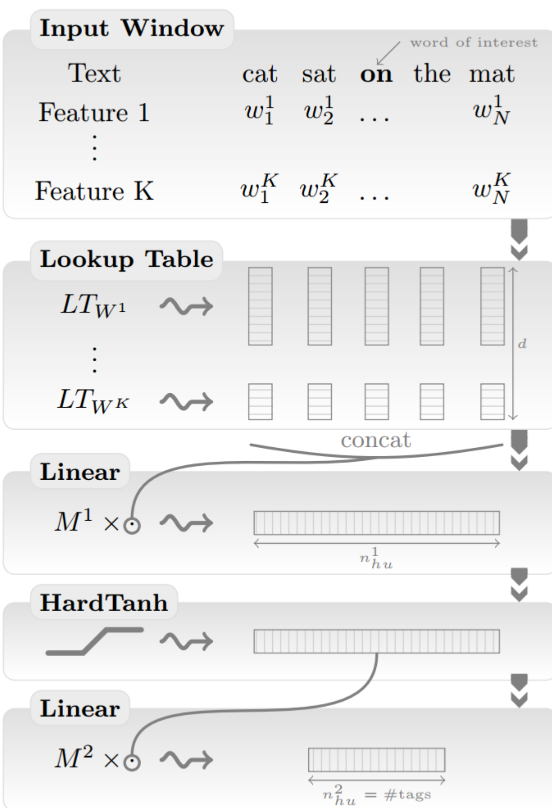


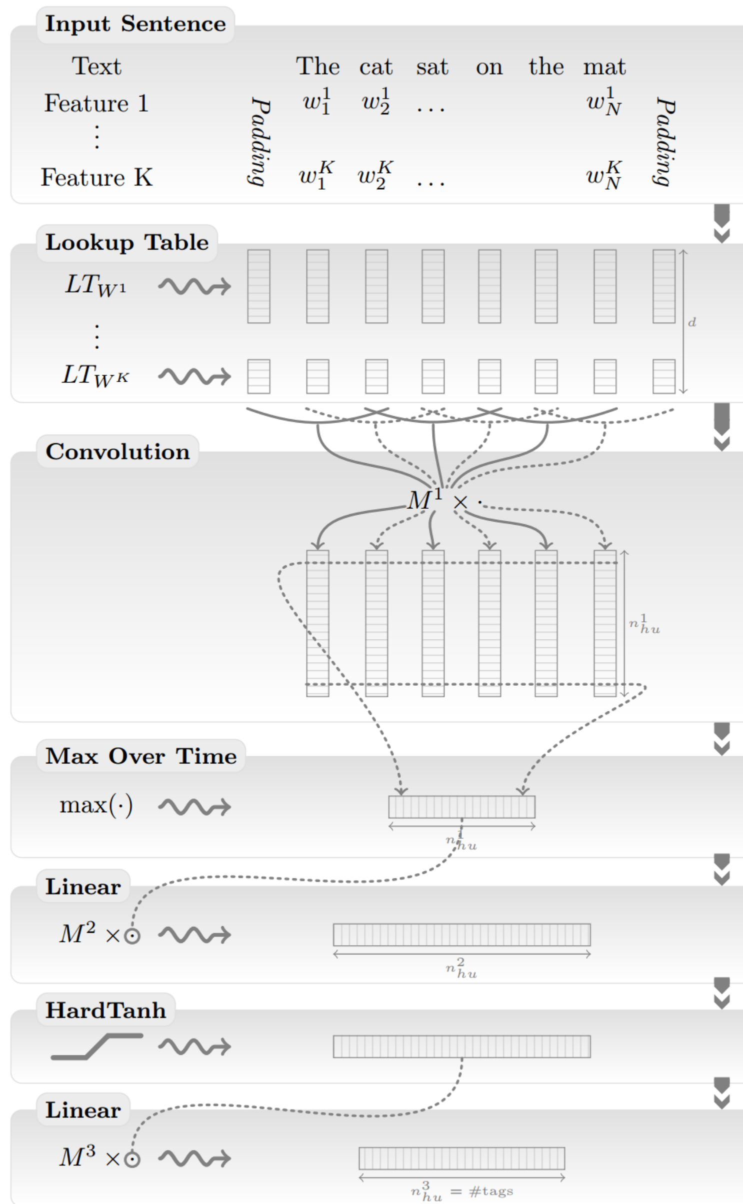
http://people.cs.umass.edu/~mccallum/papers/crf-tutorial.pdf

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

Collobert et al. (2011)

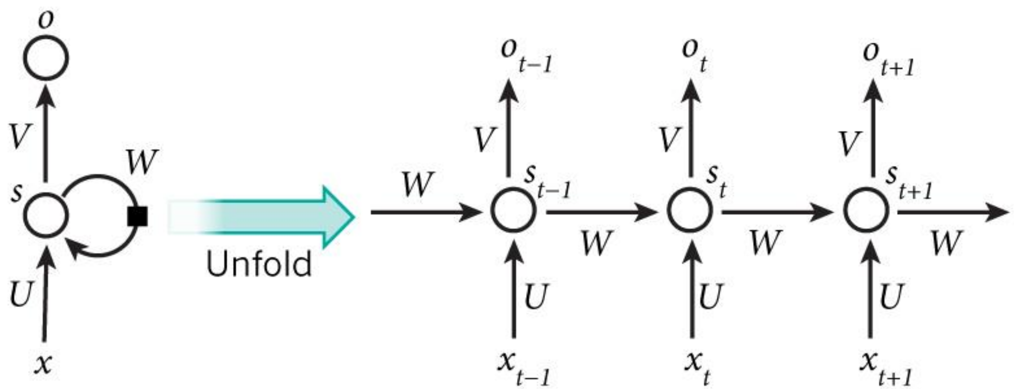
- Neural Network-based Models
  - ✓ Window-based vs. sentence-based



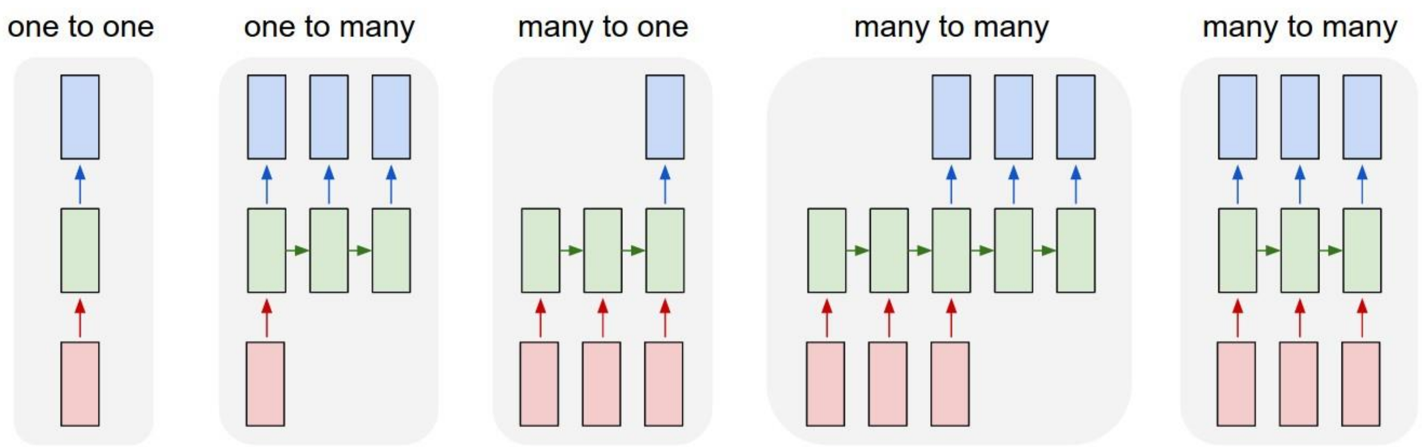


Lexical Analysis 4: Part-of-Speech (POS) Tagging

- Neural network-based models
  - ✓ Recurrent neural networks: have a feedback loop within the hidden layer

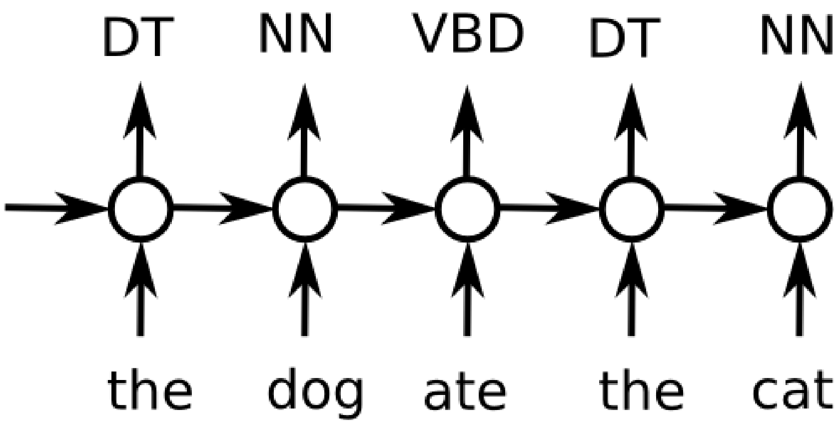


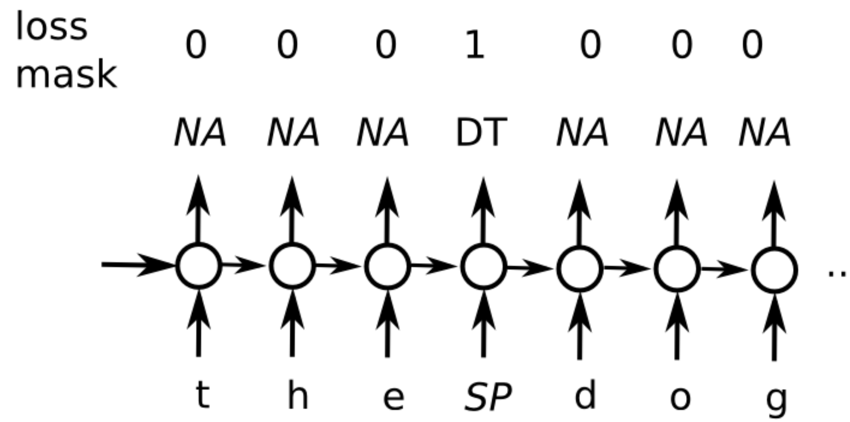
  - ✓ Input-Output mapping of RNNs



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

- Neural network-based models: Recurrent neural networks

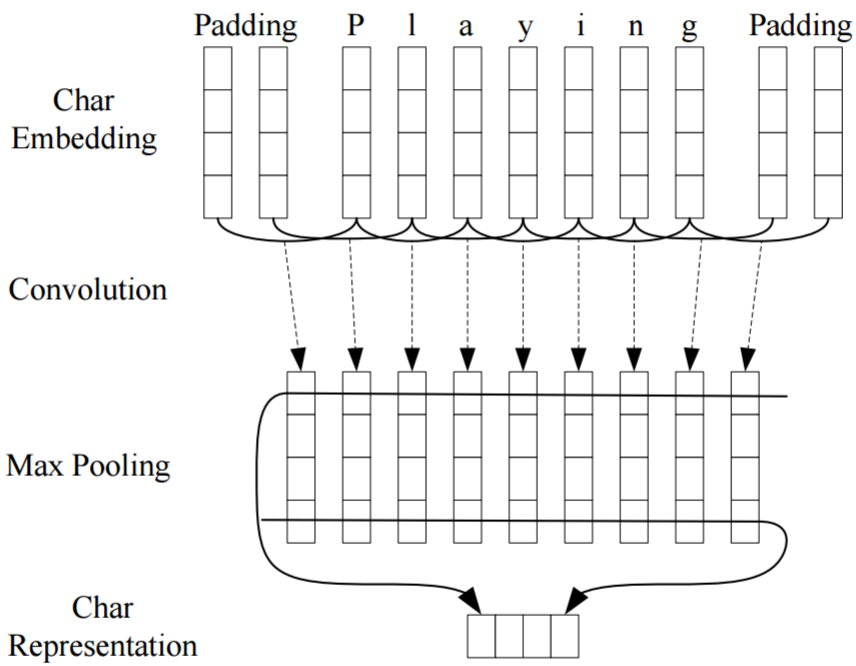


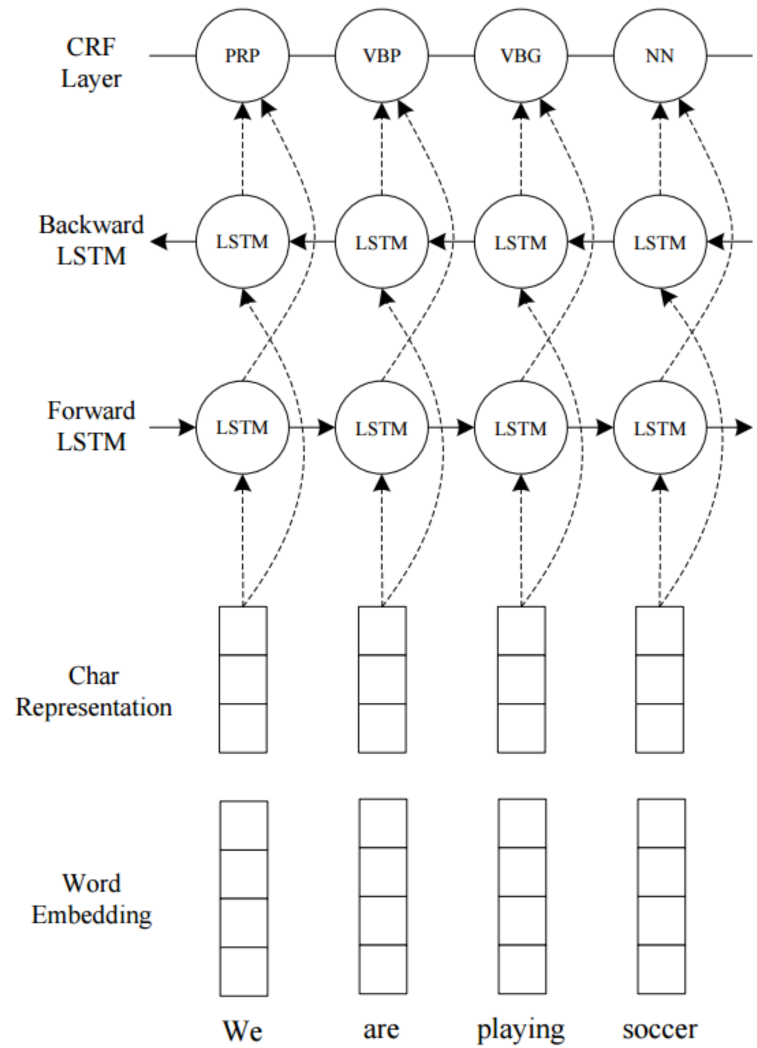


Lexical Analysis 4: Part-of-Speech (POS) Tagging

Ma and Hovy (2016)

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





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.

http://eric-yuan.me/ner_1/

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

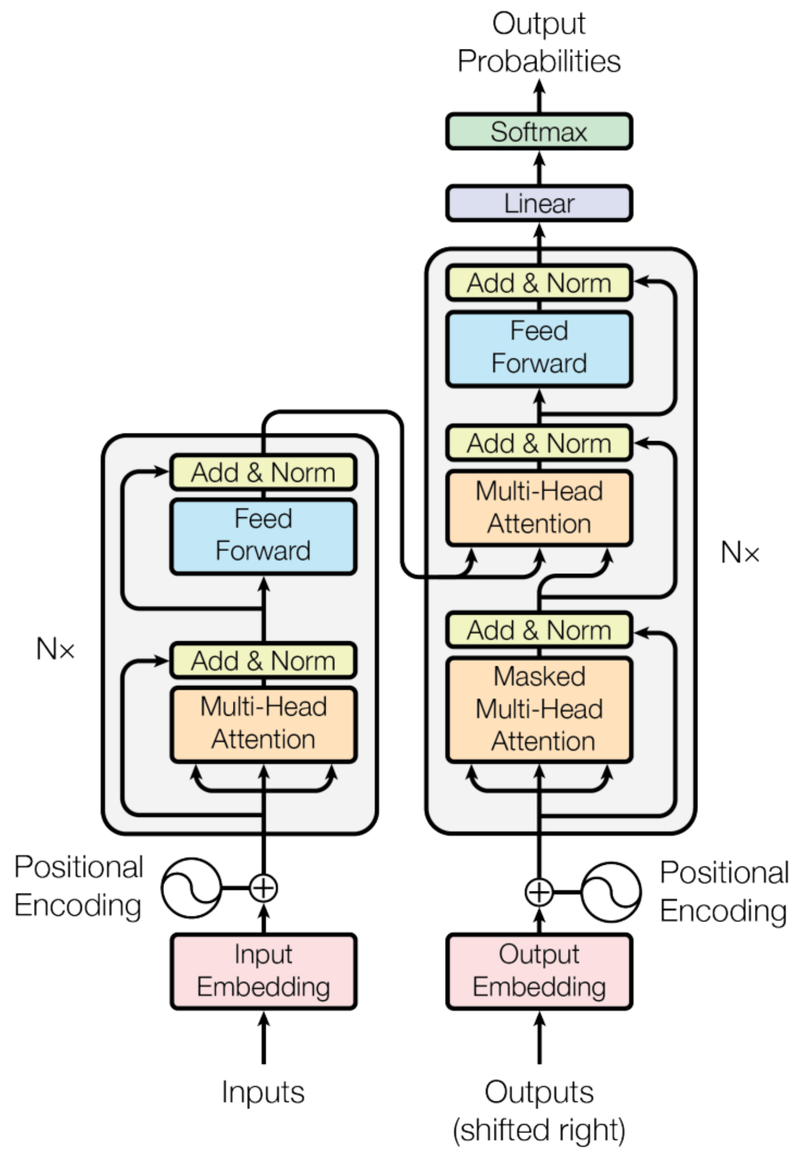
Lexical Analysis 5: Named Entity Recognition

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
  - ✓ 1-of-M coding, Word2Vec, N-Grams can be used as encoding methods

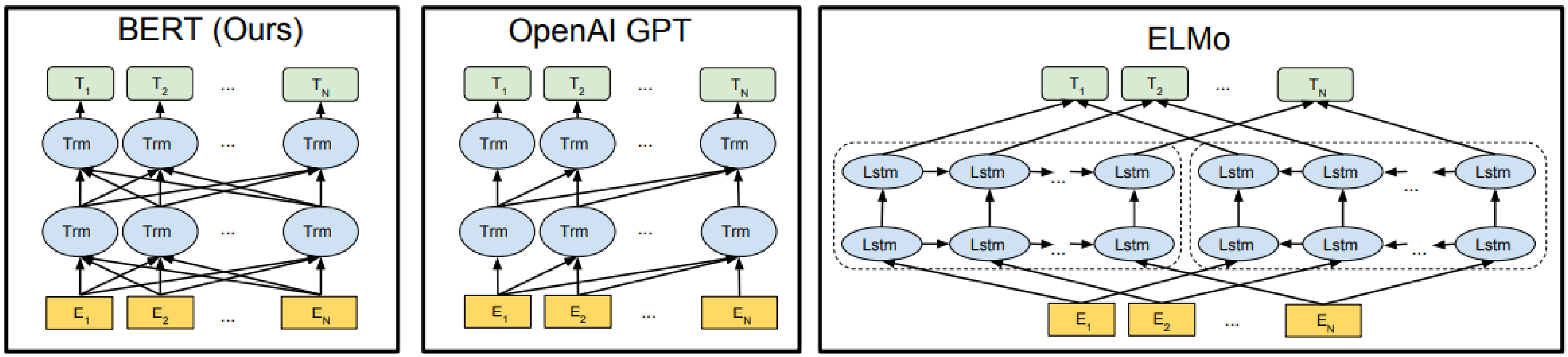
BERT for Multi NLP Tasks

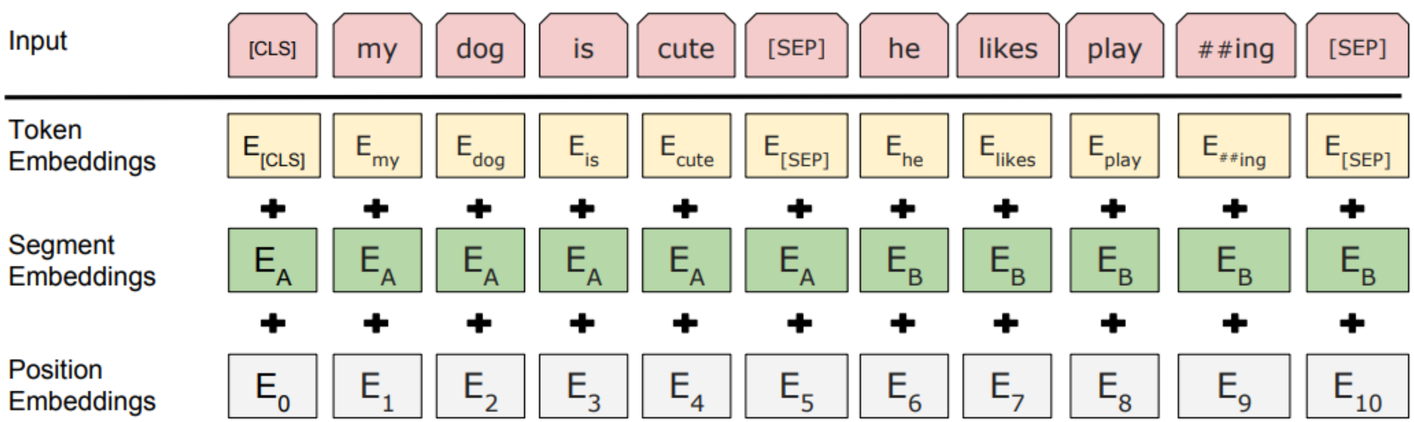
- Google Transformer
  - ✓ Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems*(pp. 5998-6008).
  - ✓ Excellent blog post explaining Transformer
    - http://jalammar.github.io/illustrated-transformer/



BERT for Multi NLP Tasks

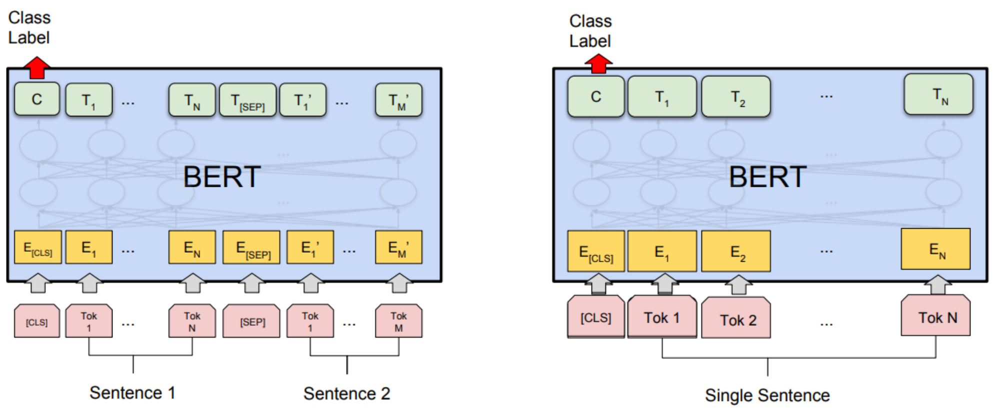
- BERT
  - ✓ Devlin, J., Chang, M.W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.





BERT for Multi NLP Tasks

- BERT
  - ✓ Devlin, J., Chang, M.W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.



(a) Sentence Pair Classification Tasks: MNLI, QQP, QNLI, STS-B, MRPC, RTE, SWAG

(b) Single Sentence Classification Tasks: SST-2, CoLA

(c) Question Answering Tasks: SQuAD v1.1

(d) Single Sentence Tagging Tasks: CoNLL-2003 NER


ANY
questions?