Statistical Natural Language Processing

Tokenization, normalization, segmentation

Çağrı Çöltekin

University of Tübingen
Seminar für Sprachwissenschaft

Summer Semester 2018

Tokenization – a solved problem?

- Typically, we (in NLP/CL/IR/…) process text as a sequence of tokens
- Tokens are word-like units
- A related task is *sentence segmentation*
- Tokenization is a language dependent task, where it becomes more challenging in some languages
- Tokenization is often regarded as trivial, and a mostly solved task

Classical NLP pipeline

- *Tokenization*
  Sentences, (normalized) words, stems / lemmas
- *Lexical / morphological processing*
  POS tags, morphological features, stems / lemmas, named entities
- *Parsing*
  Constituency / dependency trees
- *Semantic processing*
  word-senses, logical forms
- *Discourse*
  Co-reference resolution, discourse representation

We do not always use a pipeline, not all steps are necessary for all applications

Tokenization in the classical NLP pipeline

Tokenization → Morphology → Syntax → Semantics → Discourse

- Tokenization is the first in the pipeline
- Even for end-to-end approaches, tokenization is often considered given (needs to be done in advance)
- Errors propagate!

But, can't we just tokenize based on spaces?

...and get rid of the punctuation

Some examples from English:

- $10 billion
- rock 'n' roll
- he's
- can't
- O'Reilly
- 5-year-old
- B-52
- C++
- C4.5
- 29.05.2017
- 134.2.129.121
- sfs.uni-tuebingen.de
- New York-based
- wake him up

Gets more interesting in other languages

- Chinese: 猫占领了婴儿床
  ‘The cat occupied the crib’

Gets more interesting in other languages

- Chinese: 猫占领了婴儿床
  ‘The cat occupied the crib’
- German: Lebensversicherungsgesellschaftsangestellter
  ‘life insurance company employee’

Gets more interesting in other languages

- Chinese: 猫占领了婴儿床
  'The cat occupied the crib'
- German: Lebensversicherungsgesellschaftsangestellter
  'life insurance company employee'
- Turkish: İstanbullulaştıramayabileceklerimizdenmişsiniz
  'You were (evidentially) one of those who we may not be able to convert to an Istanbulite'

Gets more interesting in other languages

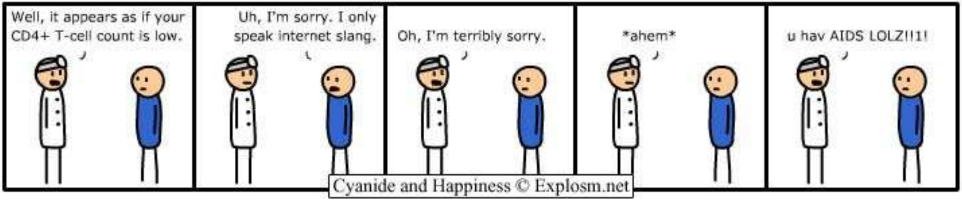
- Chinese: 猫占领了婴儿床
  ‘The cat occupied the crib’
- German: Lebensversicherungsgesellschaftsangestellter
  ‘life insurance company employee’
- Turkish: İstanbullulaştıramayabileceklerimizdenmişsiniz
  ‘You were (evidentially) one of those who we may not be able to convert to an Istanbulite’
- Even more interesting when we need to process ‘mixed’ text with *code-switching*

Specialized and non-standard text

- Much more difficult for non-standard text
  - Many specialized terms use a mixture of letters, numbers, punctuation
  - Frequent misspelling, omitting space (e.g., after sentence final punctuation)

Specialized and non-standard text

- Much more difficult for non-standard text
  - Many specialized terms use a mixture of letters, numbers, punctuation
  - Frequent misspelling, omitting space (e.g., after sentence final punctuation)
- Non-standard text can be
  - Spoken language
  - Old(er) samples of text (e.g., historical records)
  - Specialized domains, e.g., bio-medical texts
  - Informal communication, e.g., social media



Normalization

Normalization is a related task that often interacts with tokenization.

- For most applications (e.g., IR) we want to treat the following the same
  - Linguistics – linguistics
  - color – colour
  - lower case – lowercase – lower-case
  - Tübingen – Tuebingen – Tubingen
  - seee – see
  - flm – film
  - Different date/time formats, phone numbers
- Most downstream tasks require the 'normalized' forms of the words

So, what is a token?

- One token or multiple?
  - `John's`
  - `New York`
  - German: `im` (*in* + *dem*)
  - Turkish: `İstanbullulaştıramayabileceklerimizdenmişsiniz`

So, what is a token?

- One token or multiple?
  - `John's`
  - `New York`
  - German: `im` (*in* + *dem*)
  - Turkish: `İstanbullulaştıramayabileceklerimizdenmişsiniz`
- Answer is language and application dependent

So, what is a token?

- One token or multiple?
  - `John's`
  - `New York`
  - German: `im` (*in* + *dem*)
  - Turkish: `İstanbullulaştıramayabileceklerimizdenmişsiniz`
- Answer is language and application dependent
- Tokenization decisions are often arbitrary

So, what is a token?

- One token or multiple?
  - `John's`
  - `New York`
  - German: `im` (*in* + *dem*)
  - Turkish: `İstanbullulaştıramayabileceklerimizdenmişsiniz`
- Answer is language and application dependent
- Tokenization decisions are often arbitrary
- Consistency is important

Rule based tokenization

Regular expressions and finite-state automata

- The 'easy' solution to the tokenization is rule-based
- Using regular expressions,
  - we can define regular expressions for allowed tokens
  - split after match, disregard/discard the remaining parts
- For example,
  - All alphabetic characters, *word*, `[a-z]+`
  - Capitalization, *John*, `[A-Z]?[a-z]+`
  - Abbreviations, *Prof.*, `[A-Z]?[a-z]+[.]?`
  - Numbers too, *123*, `[A-Z]?[a-z]+[.]?|[0-9]+`
  - Numbers with decimal parts `[A-Z]?[a-z]+[.]?|[0-9.]+`
  - …
- Result is typically imprecise, difficult to maintain

Splitting sentences

- Another relevant task is *sentence tokenization*
- For most applications, we need sentence boundaries
- Sentence-final markers, `[.!?]` are useful
- But the dot '`.`' is ambiguous: can either be end-of- sentence or abbreviation marker, or both
  - `The U.N. is the largest intergovernmental organisation.`
  - `I had the impression he'll be ambassador to U.N.`
- Again, heuristics along with a list of abbreviations is possible

Problems with rule-based approaches

- Rule-based approaches are (still) common in practice, however
  - it is difficult to build a rule set that works well in practice
  - it is difficult to maintain
  - it is not domain or language general: needs re-implementation, re-adjustment for every case

Machine learning for word / sentence tokenization

- Another approach is to use machine learning
- Label each character in the text with
  - I inside a token
  - O outside tokens
  - B beginning of a token,
    alternatively to combine word/sentence tokenization
    - T beginning of a token
    - S beginning of a sentence
- How do we create the training data?
- What are the features for the ML?

I/O/B tokenization: an example

```
The U.N. is the largest intergovernmental
BIIOBIIIOBIOBIIOBIIIIIIOBIIIIIIIIIIIIIIIIO
organisation. I had the impression he'll be
BIIIIIIIIIIIOOBOBIIOBIIOBIIIIIIIIIOBIBIIOBIO
ambassador to U.N.
BIIIIIIIIIOBIOBIIIO
```

I/O/B tokenization example

with sentence boundary markers

```
The U.N. is the largest intergovernmental
SIIOTIIIOTIOTIIOTIIIIIIOTIIIIIIIIIIIIIIIIO
organisation. I had the impression he'll be
TIIIIIIIIIIIOOSOTIIOTIIOTIIIIIIIIIOTITIIOTIO
ambassador to U.N.
TIIIIIIIIIOTIOTIIIO
```

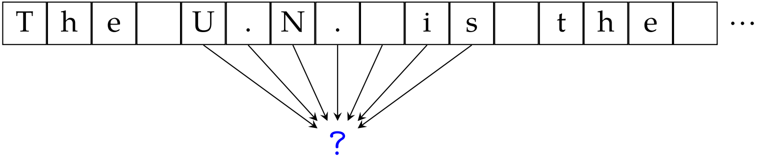
Features for tokenization

T	h	e		U	.	N	.		i	s		t	h	e		…

?

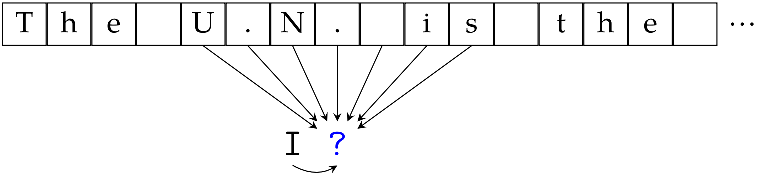
- We predict label of each character

Features for tokenization



- We predict label of each character
- Typical features are the other characters around the target
- Choice of features and the machine learning method vary

Features for tokenization



- We predict label of each character
- Typical features are the other characters around the target
- Choice of features and the machine learning method vary
- Using the previous prediction is also useful

Segmentation

- Segmentation is a related problem in many areas of computational linguistics
  - In some languages, the word boundaries are not marked
    猫占领了婴儿床 → 猫 占领 了 婴儿床
  - We often want to split words into their morphemes
    Lebensversicherungsgesellschaftsangestellter →
    Leben+s+versicherung+s+gesellschaft+s+angestellter
  - In spoken language there are no reliable word boundaries

Supervised segmentation

- I/O/B tokenization is applicable to segmentation as well
- Often produces good accuracy
- The main drawback is the need for labeled data
- Some unsupervised with reasonable accuracy also exist
- In some cases, unsupervised methods are useful and favorable

A simple 'unsupervised' approach

- Using a lexicon, segment at maximum matching lexical item
  - Serves as a good baseline, but fails in examples like

    `theman`

    where maximum match suggests segmentation 'them an'
  - The out-of-vocabulary words are problematic
- One can use already known boundaries as signal for supervision
  - Known to work especially well for sentence segmentation, (e.g., using cues `.?!`)

Unsupervised segmentation

- Two main approaches
  - Learn a compact lexicon that maximizes the likelihood of the data

$$P(s) = \prod_{i=1}^{n} P(w_i)$$

$$P(w) = \begin{cases} (1-\alpha)f(w) & \text{if } w \text{ is known} \\ \alpha \prod_{i=1}^{m} P(a_i) & \text{if } w \text{ is unknown} \end{cases}$$

  - Segment at points where predictability (entropy) is low

    The general idea: the predictability within words is high, predictability between words is low

Summary

- Tokenization is an important part of an NLP application
- Tokens are word-like units that are
  - linguistically meaningful
  - useful in NLP applications
- Tokenization is often treated as trivial, has many difficulties of its own
- White spaces help, but does not solve the tokenization problem completely
- Segmentation is tokenization of input where there are no boundary markers
- Solutions include rule-based (regex) or machine learning approaches

Next

Wed Work on assignments

Fri N-gram language models

Some extra: modeling segmentation by children

NLP can be 'sciency', too

- An interesting application of unsupervised segmentation methods is modeling child language acquisition
- How children learn languages has been one of the central topics in linguistics and cognitive science
- Computational models allow us to
  - test hypotheses
  - create explicit models
  - make predictions

The puzzle to solve

ljuuzuibutsjhiuljuuz
ljuuztbzjubhbjompwfljuuz
xibutuibu
ljuuz
epzpvxbounpsfnjmlipofz
ljuuzljuuzephhjf
opnjxibuepftbljuuztbz
xibuepftbljuuztbz
ephhjfeph
ephhjf
opnjxibuepftuifephhjftbz
xibuepftuifephhjftbz
mjuumfcbczcjsejf
cbczcjsejf
zpvepoumjlfuibupof
plbznpnnzublfuijtpvu
dpx
uifdpxtbztnppnpp
xibuepftuifdpxtbzopnj

The puzzle to solve

```
ljuuzuibutsjhiuljuuz
ljuuztbzjubhbjompwfljuuz
xibutuibu
ljuuz
epzpvxbounpsfnjmlipofz
ljuuzljuuzephhjf
opnjxibuepftbljuuztbz
xibuepftbljuuztbz
ephhjfeph
ephhjf
opnjxibuepftuifephhjftbz
xibuepftuifephhjftbz
mjuumfcbczcjsejf
cbczcjsejf
zpvepoumjlfuibupof
plbznpnnzublfuijtpvu
dpx
uifdpxtbztnppnpp
xibuepftuifdpxtbzopnj
```

- No clear boundary markers
- No lexical knowledge

How do children segment? – a bit of psycholinguistics

Children very early in life (8-months) seem to be sensitive to statistical regularities between syllables (Saffran, Aslin, and Newport 1996)

How do children segment? – a bit of psycholinguistics

Children very early in life (8-months) seem to be sensitive to statistical regularities between syllables (Saffran, Aslin, and Newport 1996)

Training: `bidakupadotigolabubidakugolabupadoti`…

How do children segment? – a bit of psycholinguistics

Children very early in life (8-months) seem to be sensitive to statistical regularities between syllables (Saffran, Aslin, and Newport 1996)

Training: bidakupadotigolabubidakugolabupadoti...

$P(\text{da} \mid \text{bi}) = 1$ $\qquad$ $P(\text{pa} \mid \text{bu}) = \frac{1}{3}$

How do children segment? – a bit of psycholinguistics

Children very early in life (8-months) seem to be sensitive to statistical regularities between syllables (Saffran, Aslin, and Newport 1996)

Training: `bidakupadotigolabubidakugolabupadoti`…

$\mathsf{P}(\mathtt{da} \mid \mathtt{bi}) = 1$

$\mathsf{P}(\mathtt{pa} \mid \mathtt{bu}) = \frac{1}{3}$

test G1: words

`padotibidakugolabupadoti`…

test G2: non-words

`pagolabidotikugobdalaubu`…

How do children segment? – a bit of psycholinguistics

Children very early in life (8-months) seem to be sensitive to statistical regularities between syllables (Saffran, Aslin, and Newport 1996)

Training: bidakupadotigolabubidakugolabupadoti…

$P(\text{da} \mid \text{bi}) = 1$

$P(\text{pa} \mid \text{bu}) = \frac{1}{3}$

test G1: words

padotibidakugolabupadoti…

test G2: non-words

pagolabidotikugobdalaubu…

Children showed preference towards the 'words' that are used in the training phase.

Predictability

Predictability within units is high, predictability between units is low.

Predictability

Predictability within units is high, predictability between units is low.

Given a sequence `lr`, where `l` and `r` are sequences of phonemes:

- If `l` help us predict `r`, `lr` is likely to be part of a word
- If observing `r` after `l` is surprising it is likely that there is a boundary between `l` and `r`

Predictability

Predictability within units is high, predictability between units is low.

Given a sequence `lr`, where `l` and `r` are sequences of phonemes:

- If `l` help us predict `r`, `lr` is likely to be part of a word
- If observing `r` after `l` is surprising it is likely that there is a boundary between `l` and `r`

The strategy dates back to 1950s (**haris1955**), where he used a measure called *successor variety* (SV):

The morpheme boundaries are at the locations where there is a high variety of possible phonemes that follow the initial segment.

How to calculate the measures

`#   I   z   D   &   t   6   k   I   t   i   #`

How to calculate the measures

$\boxed{\texttt{\# I} \mid \texttt{z}}$ D & t 6 k I t i #

P: 0.40

$$P(\texttt{z}|\texttt{\#I}) = 0.40$$

How to calculate the measures

#	I	z	D	&	t	6	k	I	t	i	#
P:		0.40	0.22								

$$P(\mathtt{D}|\mathtt{Iz}) = 0.22$$

How to calculate the measures

`#`	`I`	`z`	`D`	`&`	`t`	`6`	`k`	`I`	`t`	`i`	`#`

P: 0.40 0.22 0.46

$$\mathtt{P}(\mathtt{\&}|\mathtt{zD}) = 0.46$$

How to calculate the measures

`#`	`I`	`z`	`D`	`&`	`t`	`6`	`k`	`I`	`t`	`i`	`#`

P: 0.40 0.22 0.46 0.99

$$\mathtt{P}(\mathtt{t}|\mathtt{D\&}) = 0.99$$

How to calculate the measures

`#`	`I`	`z`	`D`	`&`	`t`	`6`	`k`	`I`	`t`	`i`	`#`
P:		0.40	0.22	0.46	0.99	0.03					

$$P(\texttt{6}|\texttt{\&t}) = 0.03$$

How to calculate the measures

#	I	z	D	&	t	6	k	I	t	i	#
P:	0.40	0.22	0.46	0.99	0.03	0.04					

$$P(\mathtt{k}|\mathtt{t6}) = 0.04$$

How to calculate the measures

#	I	z	D	&	t	6	k	I	t	i	#
P:	0.40	0.22	0.46	0.99	0.03	0.04	0.30				

$$P(\mathtt{I}|\mathtt{6k}) = 0.30$$

How to calculate the measures

#	I	z	D	&	t	6	k	I	t	i	#
P:		0.40	0.22	0.46	0.99	0.03	0.04	0.30	0.48		

$$P(\texttt{t}|\texttt{kI}) = 0.48$$

How to calculate the measures

`#`	`I`	`z`	`D`	`&`	`t`	`6`	`k`	$\boxed{\texttt{I}}$	$\boxed{\texttt{t}}$	$\boxed{\texttt{i}}$	`#`
P:	0.40	0.22	0.46	0.99	0.03	0.04	0.30	0.48	0.10		

$$P(\texttt{i}|\texttt{It}) = 0.10$$

Calculations are done on a corpus of child-directed English

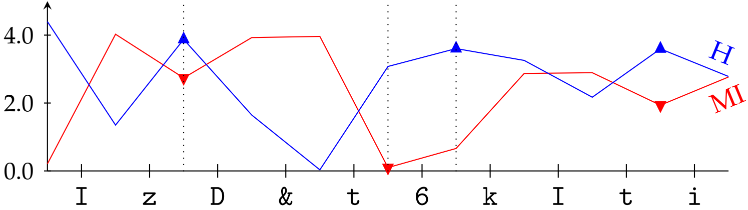
An unsupervised method

- An obvious way to segment the sequence is using a threshold value. However, the choice of threshold is difficult in an unsupervised system.

An unsupervised method

- An obvious way to segment the sequence is using a threshold value. However, the choice of threshold is difficult in an unsupervised system.

A simple unsupervised method: segment at peaks/valleys.



Segmentation puzzle: a solution

```
ljuuz uibut sjhiu ljuuz
ljuuz tbz ju bhbjo mpwf ljuuz
xibut uibu
ljuuz
ep zpv xbou npsf njml ipofz
ljuuz ljuuz ephhjf
opnj xibu epft b ljuuz tbz
xibu epft b ljuuz tbz
ephhjf eph
ephhjf
opnj xibu epft uif ephhjf tbz
xibu epft uif ephhjf tbz
mjuumf cbcz cjsejf
cbcz cjsejf
zpv epou mjlf uibu pof
plbz npnnz ublf uijt pvu
dpx
uif dpx tbzt npp npp
xibu epft uif dpx tbz opnj
```

Segmentation puzzle: a solution

ljuuz uibut sjhiu ljuuz	ljuuz uibu tsjhiuljuuz
ljuuz tbz ju bhbjo mpwf ljuuz	ljuuz tbz jubhbjompwfljuuz
xibut uibu	xibu tuibu
ljuuz	ljuuz
ep zpv xbou npsf njml ipofz	ep zpvxbounpsfnjmli pof z
ljuuz ljuuz ephhjf	ljuuz ljuuz ephhjf
opnj xibu epft b ljuuz tbz	opnj xibu ep ftb ljuuz tbz
xibu epft b ljuuz tbz	xibu ep ftb ljuuz tbz
ephhjf eph	ephhjf eph
ephhjf	ephhjf
opnj xibu epft uif ephhjf tbz	opnj xibu epft uif ephhjf tbz
xibu epft uif ephhjf tbz	xibu ep ft uif ephhjf tbz
mjuumf cbcz cjsejf	mjuumfcbczcjsejf
cbcz cjsejf	cbczcjsejf
zpv epou mjlf uibu pof	zpv epoumj lf uibu pof
plbz npnnz ublf uijt pvu	plbznpnnzublfui jtpvu
dpx	dpx
uif dpx tbzt npp npp	uif dpx tbz tnppnpp
xibu epft uif dpx tbz opnj	xibu epft uif dpx tbz opnj

Segmentation puzzle: a solution

`kitty thats right kitty`	`kitty that srightkitty`
`kitty say it again love kitty`	`kitty say itagainlovekitty`
`whats that`	`what sthat`
`kitty`	`kitty`
`do you want more milk honey`	`do youwantmoremilkh one y`
`kitty kitty doggie`	`kitty kitty doggie`
`nomi what does a kitty say`	`nomi what do esa kitty say`
`what does a kitty say`	`what do esa kitty say`
`doggie dog`	`doggie dog`
`doggie`	`doggie`
`nomi what does the doggie say`	`nomi what does the doggie say`
`what does the doggie say`	`what do es the doggie say`
`little baby birdie`	`littlebabybirdie`
`baby birdie`	`babybirdie`
`you dont like that one`	`you dontli ke that one`
`okay mommy take this out`	`okaymommytaketh isout`
`cow`	`cow`
`the cow says moo moo`	`the cow say smoomoo`
`what does the cow say nomi`	`what does the cow say nomi`

Additional reading, references, credits

- Textbook reference: Jurafsky and Martin (2009, chapter 2 of the 3rd edition draft) sections 2.1–2.3 (inclusive)
- The Chinese word segmentation example is from Ma and Hinrichs (2015)
- Other segmentation examples are from Çöltekin (2011), where there is also a good amount of introductory information on segmentation

Additional reading, references, credits (cont.)

Çöltekin, Çağrı (2011). "Catching Words in a Stream of Speech: Computational simulations of segmenting transcribed child-directed speech". PhD thesis. University of Groningen. URL: http://irs.ub.rug.nl/ppn/33913190X.

Jurafsky, Daniel and James H. Martin (2009). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. second. Pearson Prentice Hall. ISBN: 978-0-13-504196-3.

Ma, Jianqiang and Erhard Hinrichs (2015). "Accurate Linear-Time Chinese Word Segmentation via Embedding Matching". In: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Beijing, China: Association for Computational Linguistics, pp. 1733–1743. URL: http://www.aclweb.org/anthology/P15-1167.

Saffran, Jenny R., Richard N. Aslin, and Elissa L. Newport (1996). "Statistical learning by 8-month old infants". In: *Science* 274.5294, pp. 1926–1928. DOI: 10.1126/science.274.5294.1926.