Information Retrieval

- Introduction
- Text and Language

Development:
Moshe Friedman

Credits:
Yoav Goldberg, Ido Dagan, Reut Tsarfaty , Moshe Koppel, Wei Song, David Bamman, Ed Grefenstette, Chris Manning, Tsvi Kuflik, Hinrich Schütze, Christina Lioma and more

Information Retrieval - administration

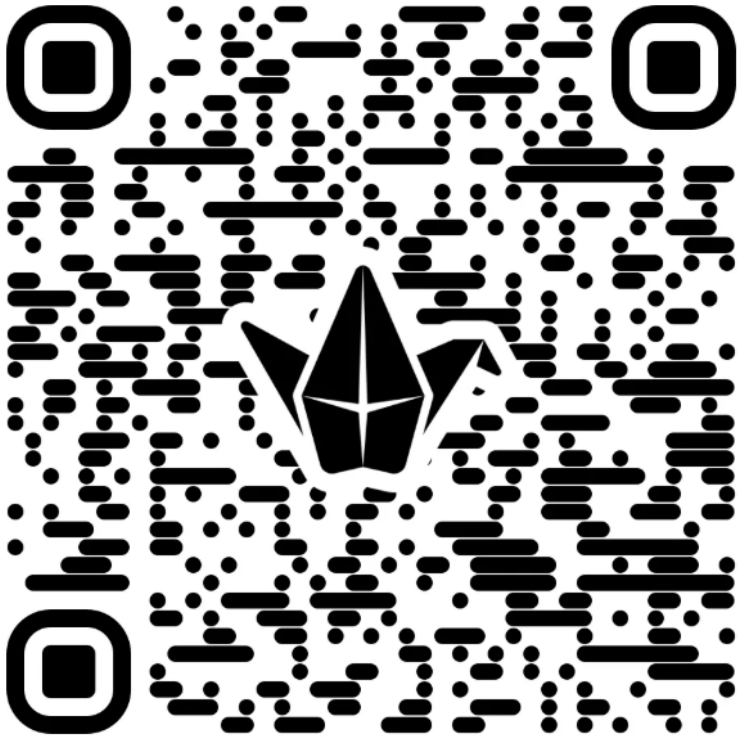
Moshe Friedman

Email: moshefr.teach@gmail.com

Reception time: before/after lesson/zoom with coordination

What do you have in mind re: information retrieval (IR)

https://padlet.com/moshe_cs/my-terrific-sandbox-8fe8d1ocox4uwqe1?frame_id=page%3A-IaBA3DbY1Xf4ptwglVU4

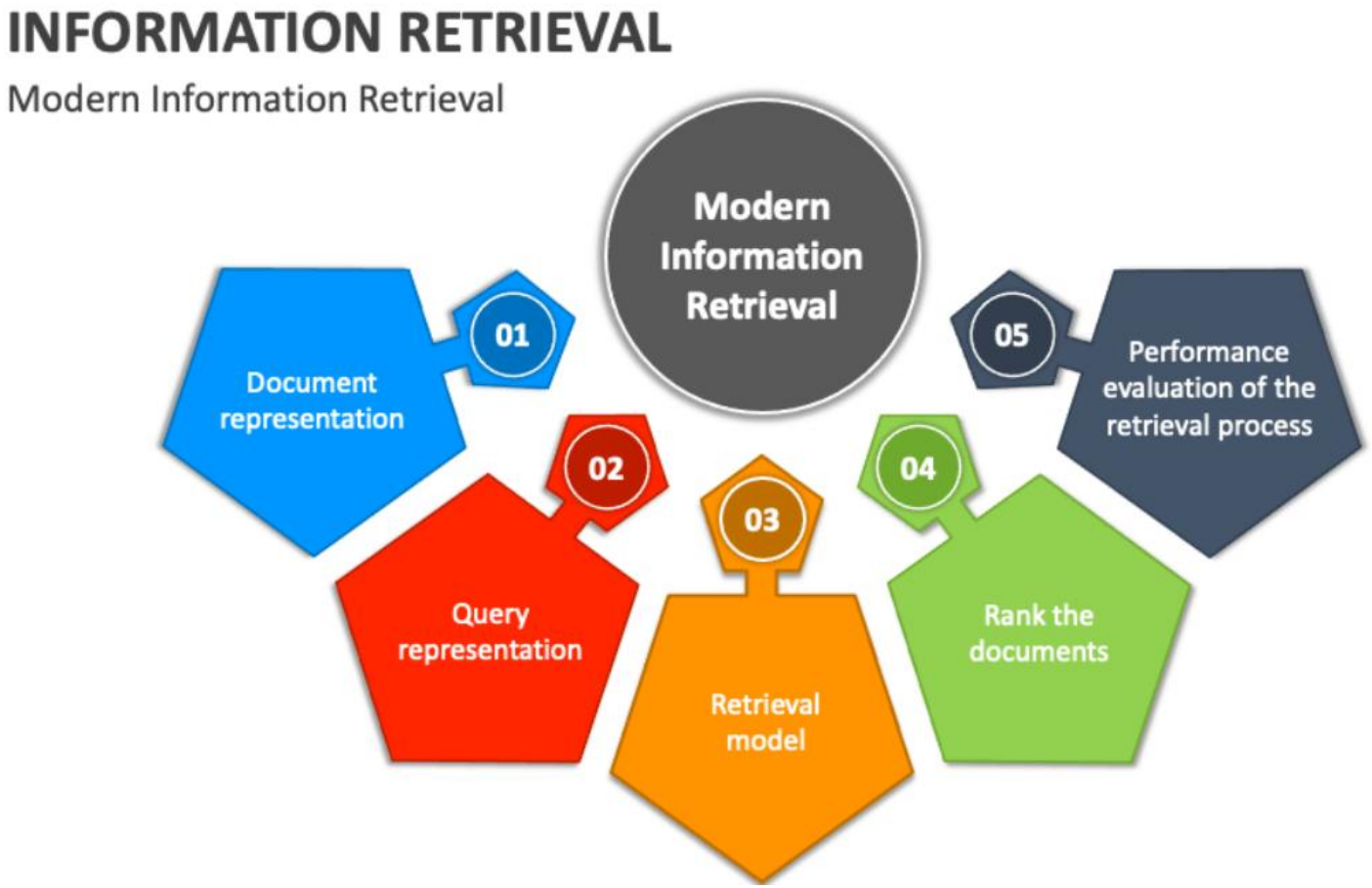


Introduction to

Information Retrieval

Introduction

Classic Information Retrieval Components

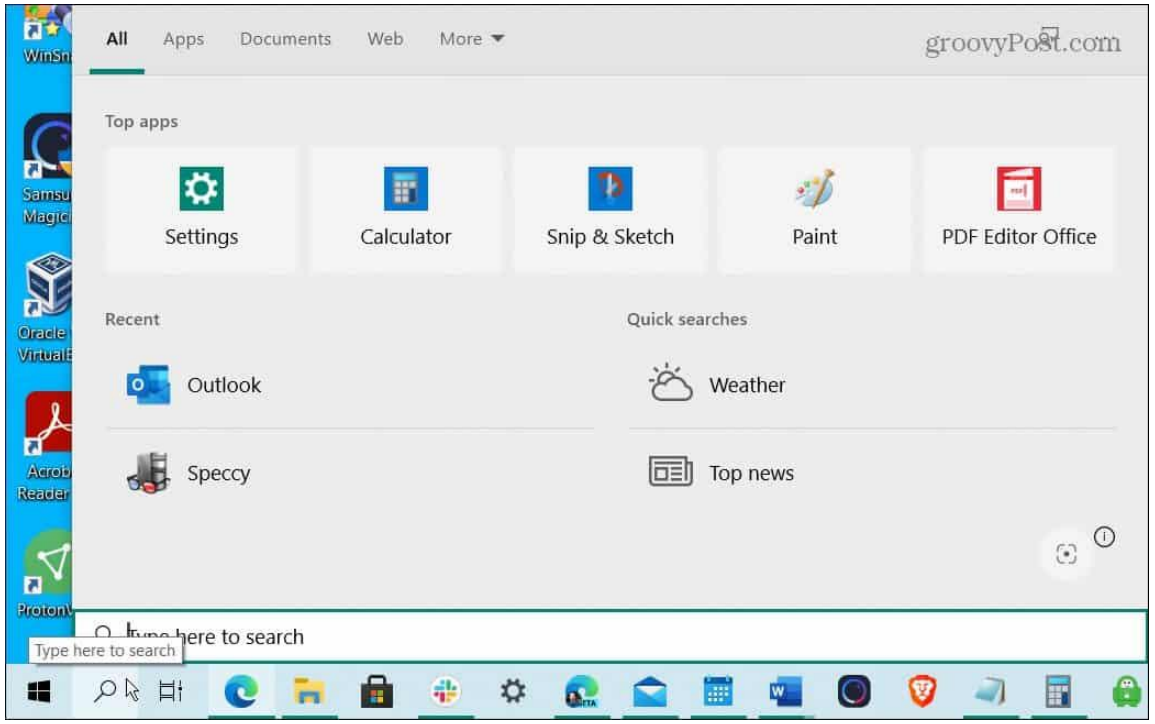


Information Retrieval

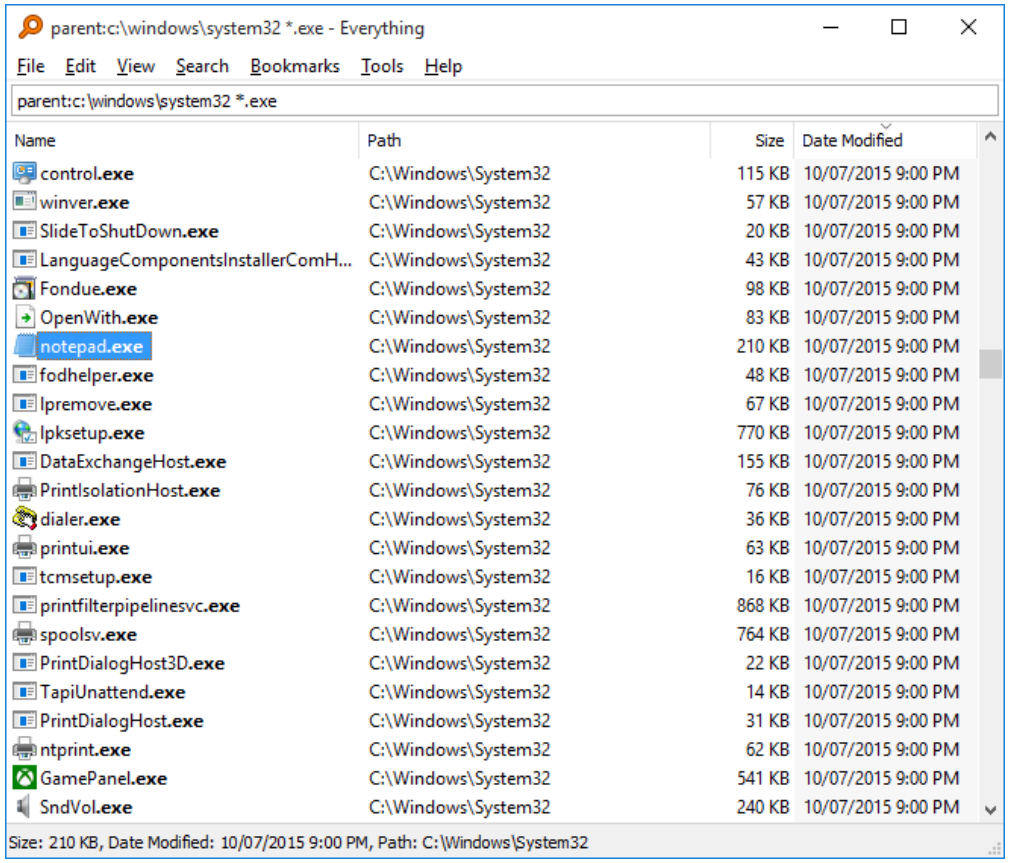
- Information Retrieval (IR) is finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers).
  - These days we frequently think first of web search, but there are many other cases:
    - E-mail search
    - Searching your laptop
    - Corporate knowledge bases
    - Legal information retrieval
    - Autocomplete

Information Retrieval Use Cases: Desktop Search

Window Search

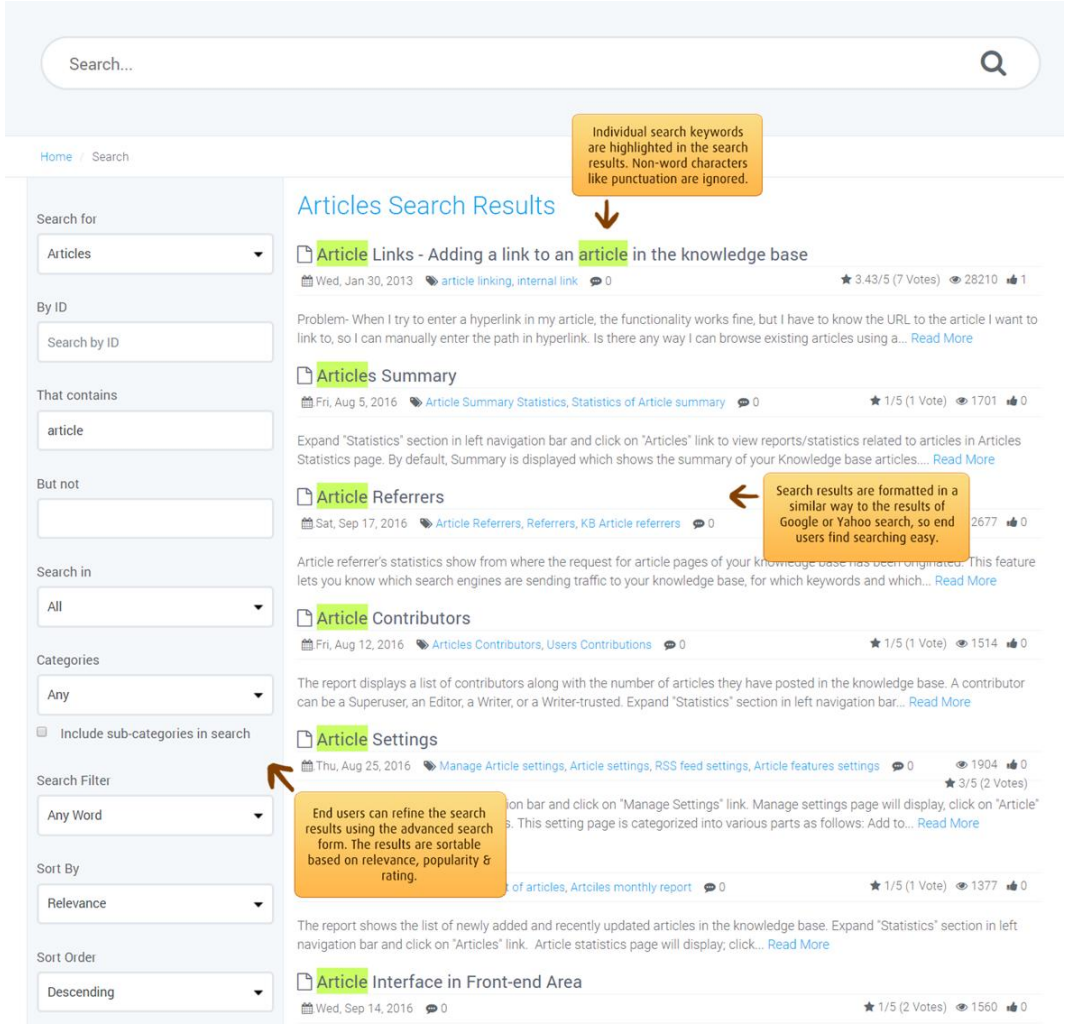


Everything

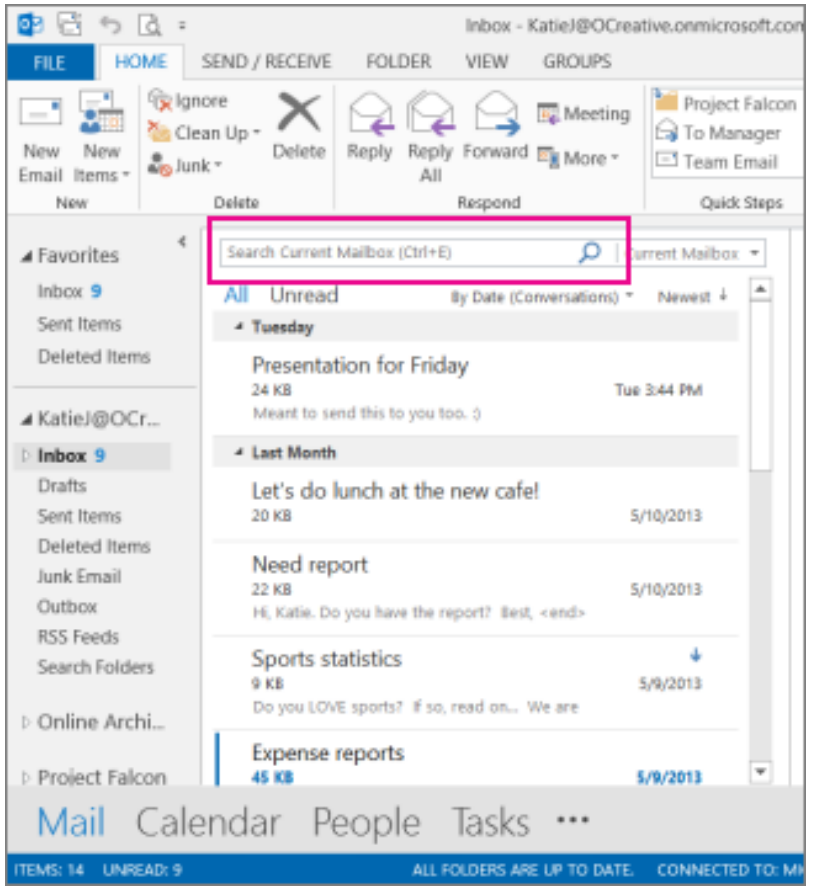


Information Retrieval Use Cases

Corporate KB Search

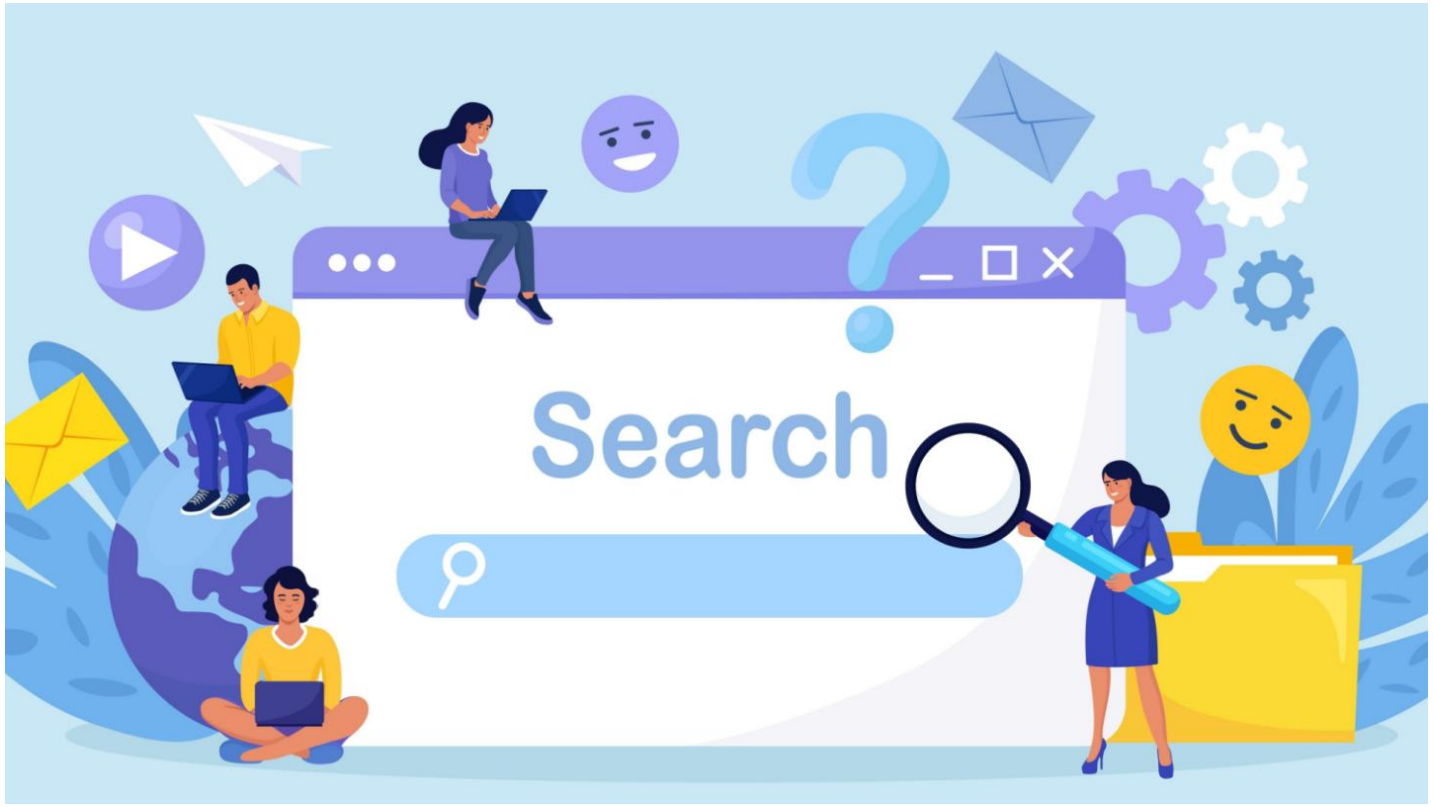


Email Search

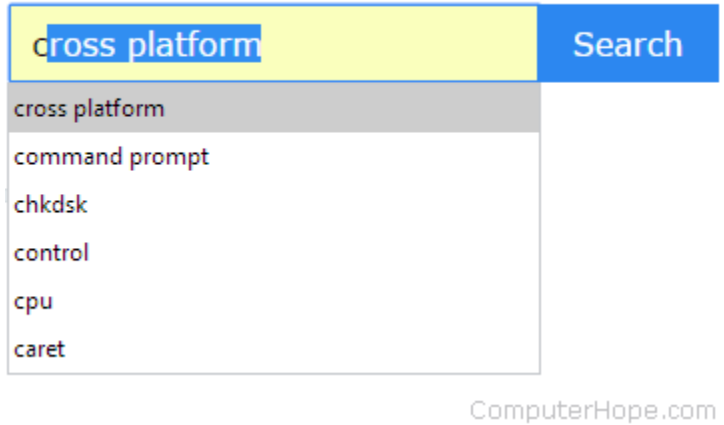


Information Retrieval Use Cases

Web Search



Autocomplete

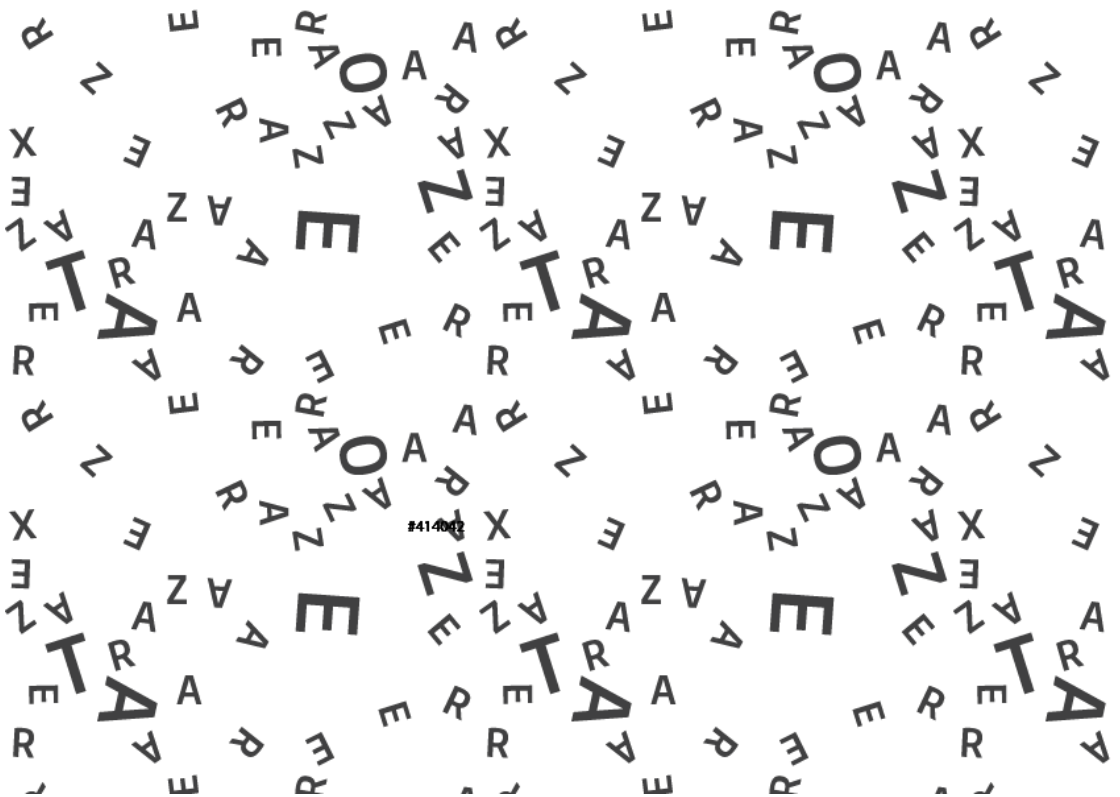


Introduction to

Information Retrieval

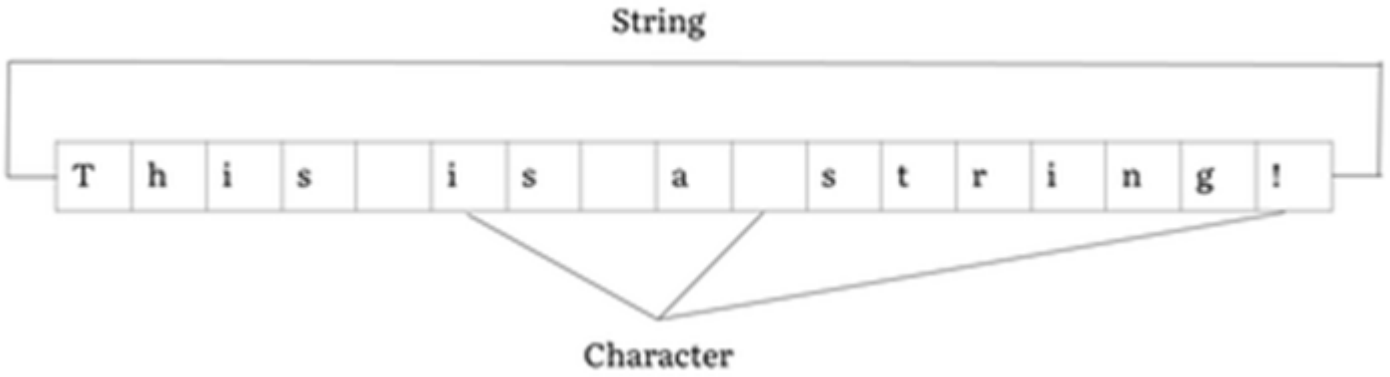
Text and Language

What is a text?
A programmer's perspective



איך נתייחס
לטקסט
בפרספקטיבה
של מתכנת?

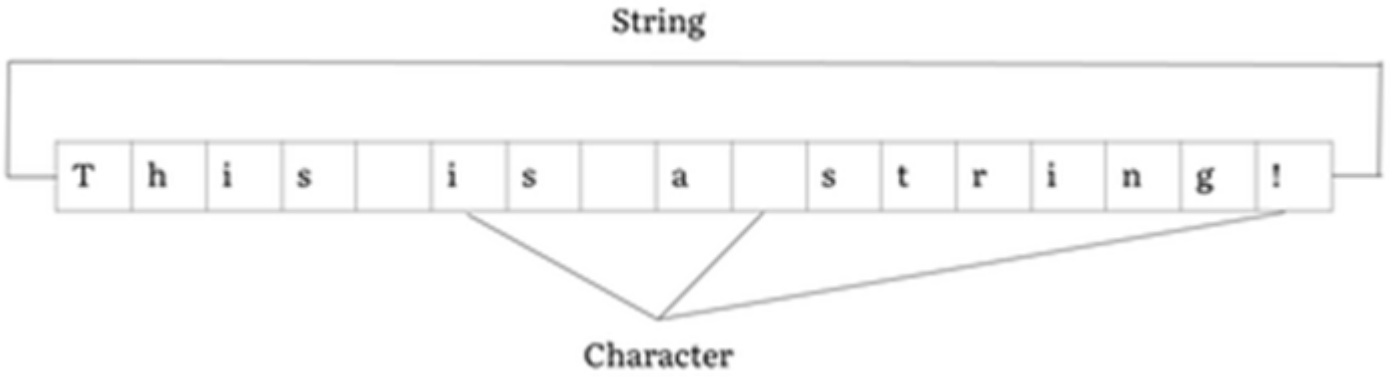
What is a text?
A programmer's perspective



מחרוזת (וויקיפדיה) - בשפות תכנות מחרוזת היא טיפוס נתונים המכיל רצף של תווים.

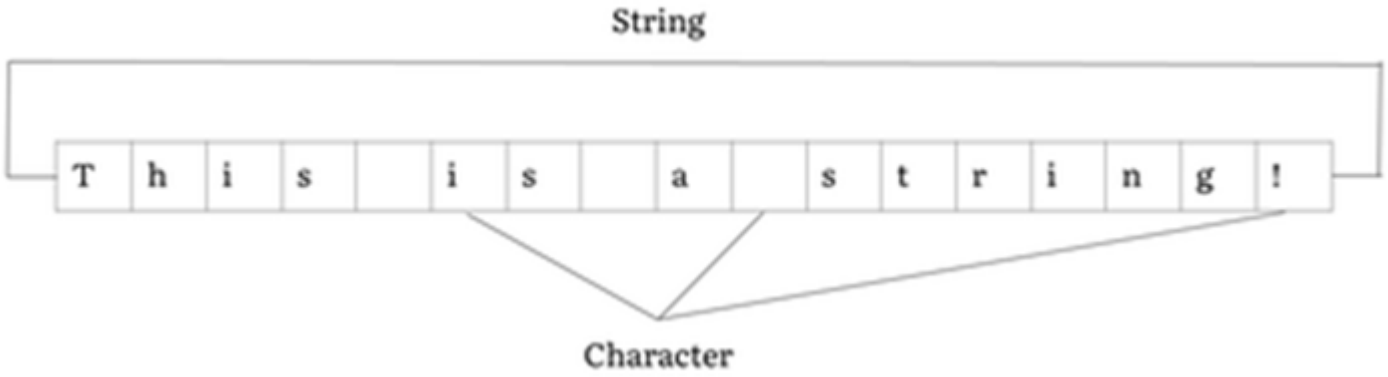
ולכן בראיה של מתכנת – טקסט היא מחרוזת תווים.

What is a text?
A programmer's perspective



אך האם כל מחרוזת תווים משמעותה טקסט?

What is a text?
A programmer's perspective

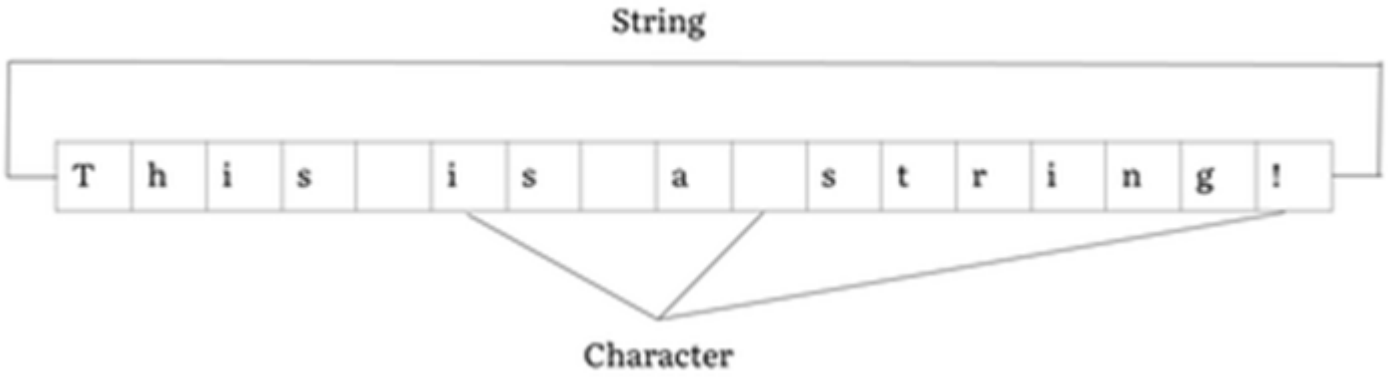


אך האם כל מחרוזת תווים משמעותה טקסט?

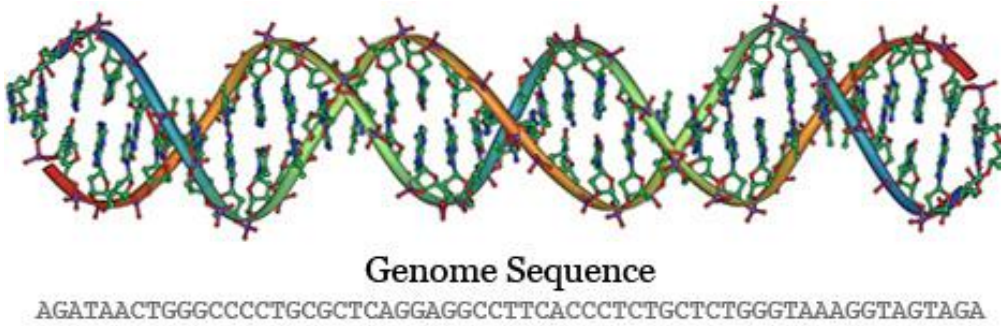
RSA encryption key

```
l1vBCavOBjD22a5RFZQDn+jpr8D/zPS+hhXnXHazZAKkOx1qKbZtIZWDt3Mv
8TFt6YgNZ9RSN19r7G7Vim9q3l35EXbG1Vb/UwLY4WtcNLl9nHuvgclF8ebS
wn3gS+ArRdOa2KsSdg13LU1OsPsRmOlroPrS7jUM3jZS9g3FnukRRCYqJMZZ
nCf/2C33OaeLHCLmNn6pegRqGb6hTy6717g6+xw/bGtLEIrd75AMcW+q72E/
x3UkMOx1wtRROugEfujI8YF/j2PM85YoZgW5HTc9+xrLxJ9Z8N13/TWujgC7
xmFO3VsP6IqfSZCvvatktwkXmvbf9S7e6uNzFs33Rw==
```

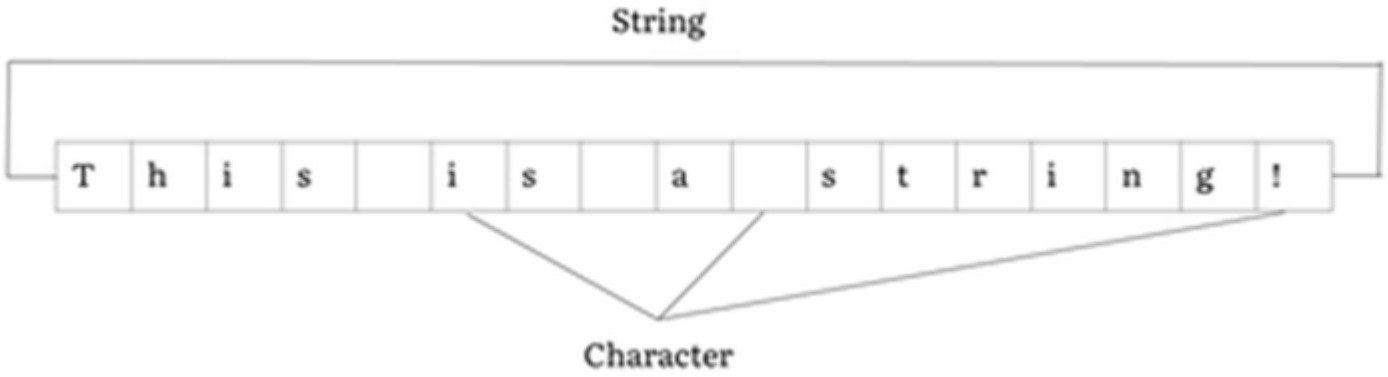
What is a text?
A programmer's perspective

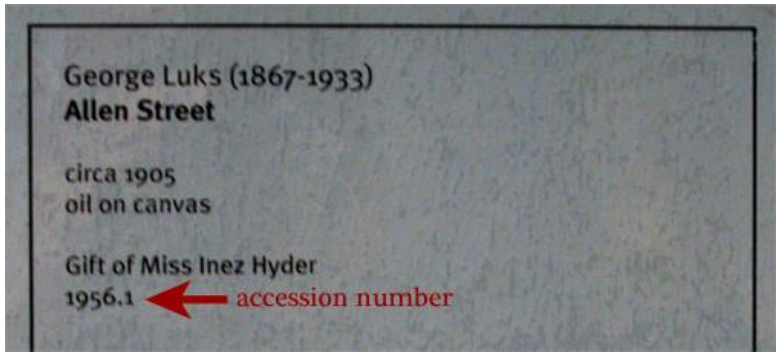


אך האם כל מחרוזת תווים משמעותה טקסט?



What is a text?
A programmer's perspective





אך האם כל מחרוזת תווים משמעותה טקסט?

Accession number (Wikipedia)- In libraries, art galleries, museums and archives, an accession number is a unique identifier assigned to, and achieving initial control of, each acquisition

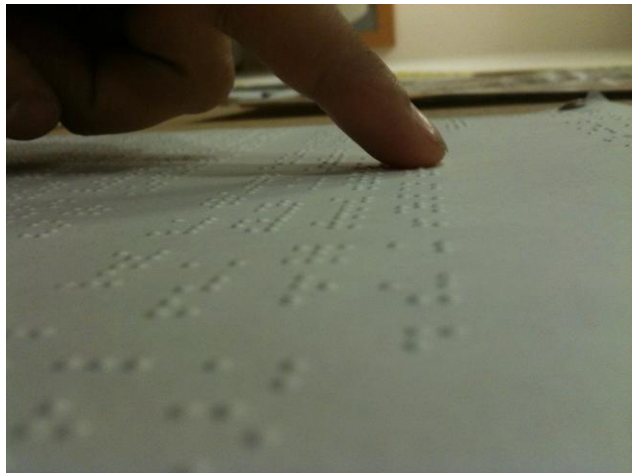
What is a text?

מסקנה – לא כל מחרוזת תווים היא טקסט.

אז מהו טקסט?

Text (Wikipedia) - In literary theory, a text is any object that can be "read“.

נחזור לטקסט עוד מעט



נבחן עוד כמה שאלות ...

מה הוא ידע?

Knowledge (Wikipedia) - Knowledge is a familiarity, awareness, or understanding of someone or something, such as facts (descriptive knowledge).

נבחן עוד כמה שאלות ...

מהי שפה?

Language is the use of a system of communication which consists of a set of sounds or written symbols.

הדרך בה אנחנו צורכים, מעבירים ומבינים ידע הוא דרך שפה

נבחן עוד כמה שאלות ...

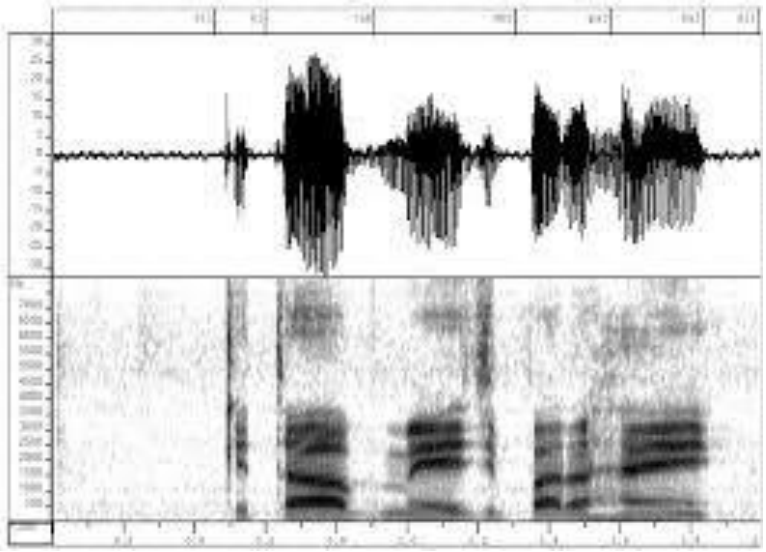
מהי שפה?

Language is the use of a system of communication which consists of a set of sounds or written symbols.

<u>שפה טבעית</u> - להבדיל משפות אחרות (שפות תכנות למשל),
שפה המבטאת תקשורת בין בני אדם נכנה שפה טבעית.

שפה טבעית ואחזור מידע

Voice/Speech recognition /phonology



text

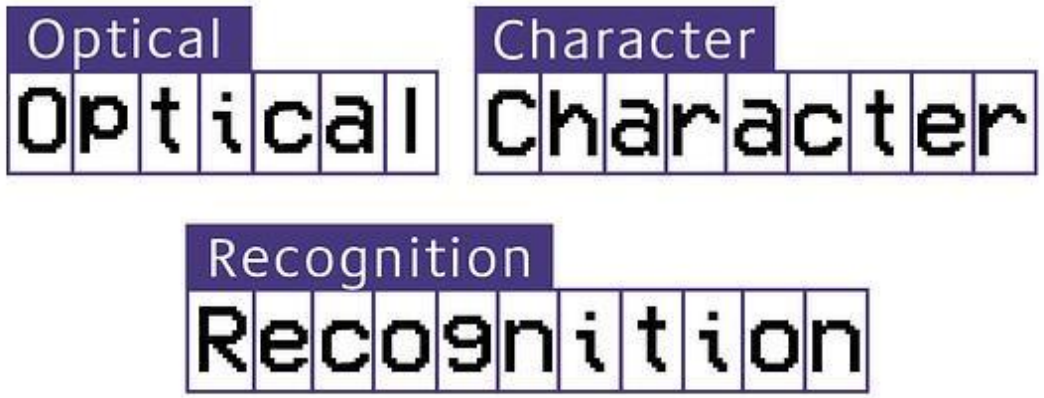
TAVILIETAKADUR

תביא לי את הכדור

שפה טבעית ואחזור מידע

Voice/Speech recognition /phonology

OCR – Optical character recognition



שפה טבעית ואחזור מידע

Voice/Speech recognition /phonology

OCR – Optical character recognition

Nonverbal / Gesture / Micro expression

credit Wei Song, et al

What do we mean by text?

Text (Wikipedia) - In literary theory, a text is any object that can be "read“.

String (Wikipedia) - In computer programming, a string is traditionally a sequence of characters.

Language is the use of a system of communication which consists of a set of sounds or written symbols.

A text string - We refer to text of (verbal) language saved as a string.

We is not included:

- Voice/Speech recognition/Phonology
- OCR – Optical character recognition
- Nonverbal Language / Gesture / Micro expression Recognition

What do we mean by text?

Text (Wikipedia) - In literary theory, a text is any object that can be "read“.

String (Wikipedia) - In computer programming, a string is traditionally a sequence of characters.

Language is the use of a system of communication which consists of a set of sounds or written symbols.

A text string - We refer to text of (verbal) language saved as a string.

<u>We is included – (written) non-spoken language</u>:

- **Emoticons** - :-) :-(
- **Emoji -** 🙂 😢
- **Text messaging** – LOL ("laughing out loud"), "gr8" ("great")

And more

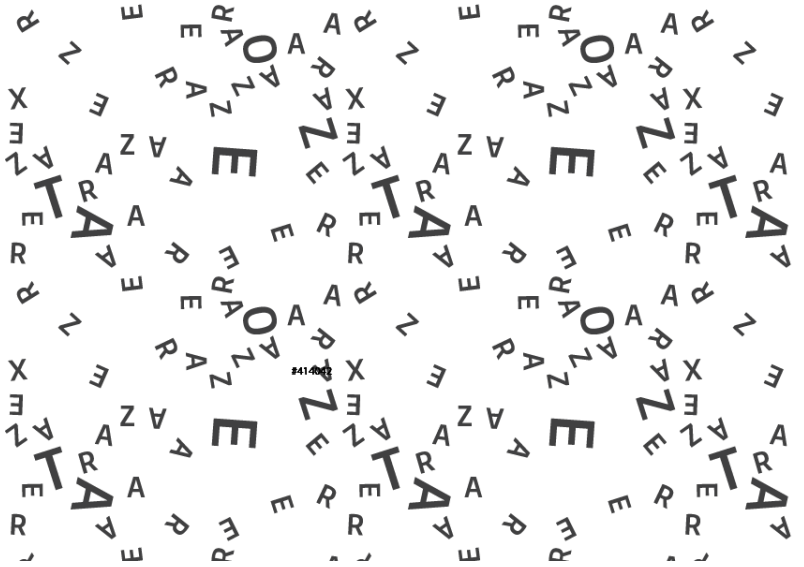
What do we know so far

Terms: Natural language and Text (verbal and non-verbal)

We want to process text, which describes natural language.

What's next?

- Bytes and bits
- Why is language processing hard?

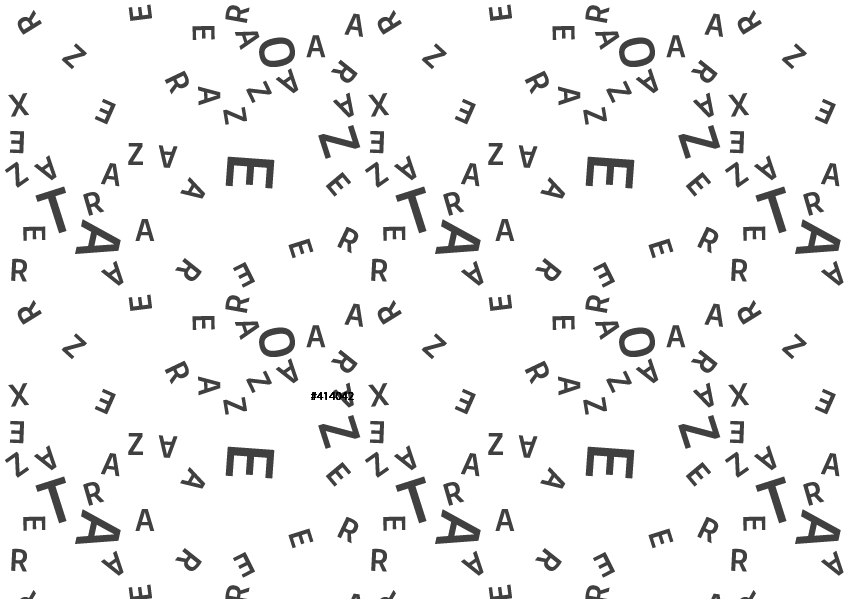


From text to data

- **An introduction**
  - What is a text?
  - Natural Language and Information Retrieval
  - What do we mean by text?
- Some text analysis challenges
- Levels of Language Processing

A lot of Buzz words

- Text analysis
- Text processing
- Text mining
- Natural language processing (aka NLP)
- Computational linguistics



What do these terms mean?

How do they relate to machine learning?

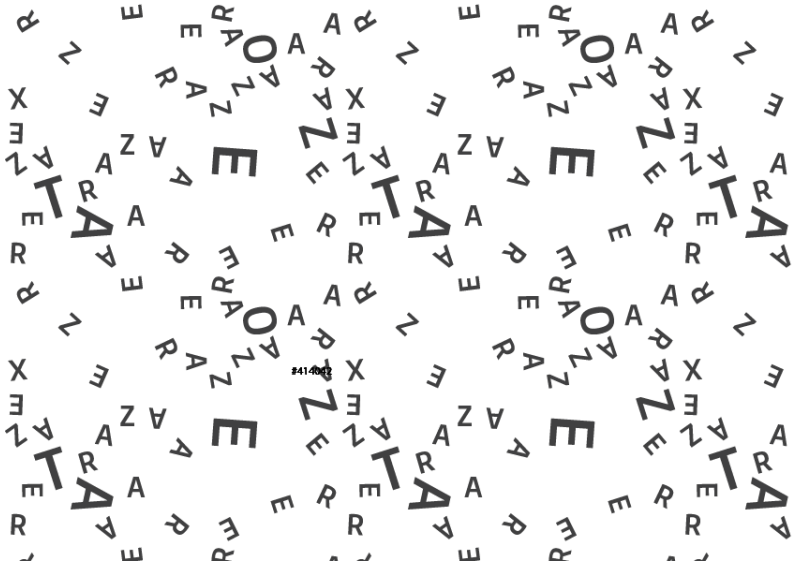
What do we know so far

Terms: Natural language and Text (verbal and non-verbal)

We want to process text, which describes natural language.

What's next?

- Bytes and bits
- Why is language processing hard?



From text to data

- An introduction
- **Some text analysis challenges**
  - Variability - one meaning, many forms
  - Ambiguity - one form, many meanings
  - Zipf law
- Levels of Language Processing

Text data – properties and challenges
Variability - one meaning, many forms

2maro 2marrow 2mor 2mora

2moro 2morow 2morr 2morro 2morrow 2mr

2mro 2mrrw 2mrw 2mw tmmrw tmo tmoro tmorrow

tmoz tmr tmro tmrow tmrrow tmrrw tmrw tmrww tmw

tomaro tomarow tomarro tomarrow tomm

tommarow tommarrow tommoro tommorow

tommorrow tommorw tommrow tomo tomolo tomoro

tomorow tomorro tomorrw tomrw

Text data – properties and challenges
Variability - one meaning, many forms

he acquired it

he purchased it

he bought it

it was bought by him

it was sold to him

she sold it to him

she sold him that

Text data – properties and challenges
Ambiguity - one form, many meanings

Bank

noun

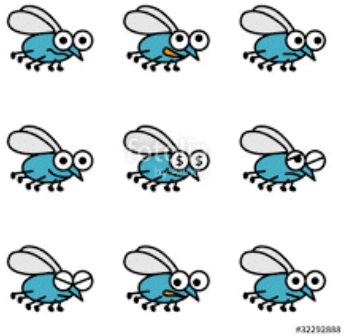
noun

Text data – properties and challenges
Ambiguity - one form, many meanings

Time Flies

If time is a **noun**

If time is a **verb**



Text data – properties and challenges
Ambiguity - one form, many meanings

Ambiguity – (extreme cases)

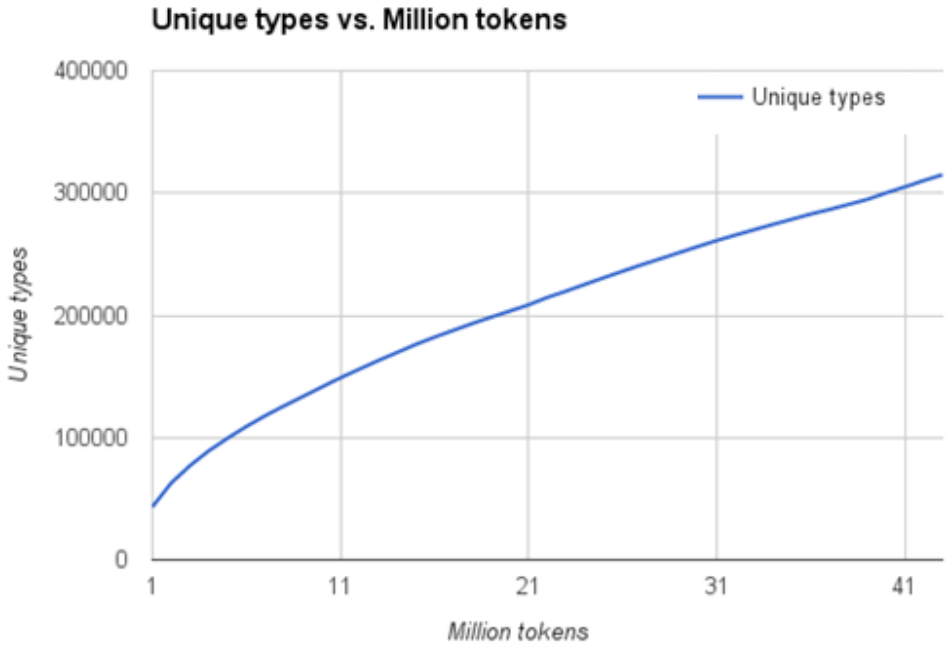
* אם מישהו רכב על סוס ועזב, אז הוא פרש או לא?
* להודות למישהו בפה מלא, זה מנומס או לא?
* אם ארכיאולוג העלה חרס בידו, הוא הצליח או לא?
* באיסטנבול לא סוגרים דלתות. הם טורקים.
* שקלתי הרבה לפני שהחלטתי לעשות דיאטה

Text data – properties and challenges
Zipf law

Word frequencies follow a power-law distribution.
--> Long tail - most words will occur only few times if they occur

There are very likely to be word forms we did not see

In a 43M words text, there are:
- 316K unique words
- 144K words occur once
- 42K words occur twice
...
- 26K words occur >50 times



From text to data

- An introduction
- Some text analysis challenges
- Levels of Language Processing
  - Phonology
  - Morphology
  - Syntax
  - Semantics

Levels of Language Processing

Phonology

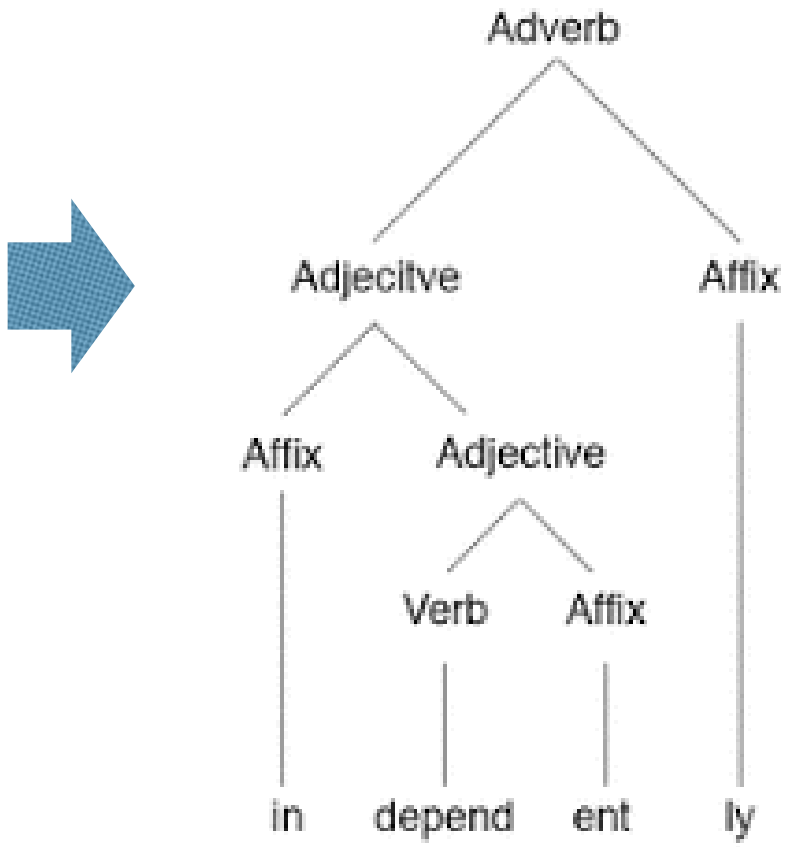
TAVILITAKADUR  תביא לי את הכדור

Levels of Language Processing

Phonology

Morphology

Independently



Levels of Language Processing

Phonology

Morphology

בגרמנית – אפשר להרכיב מילים מורכבות מאוד בעלות תחיליות, אמצעיות וסופיות.

lebensversicherungsgesellschaftsangestellter

Life insurance company employee

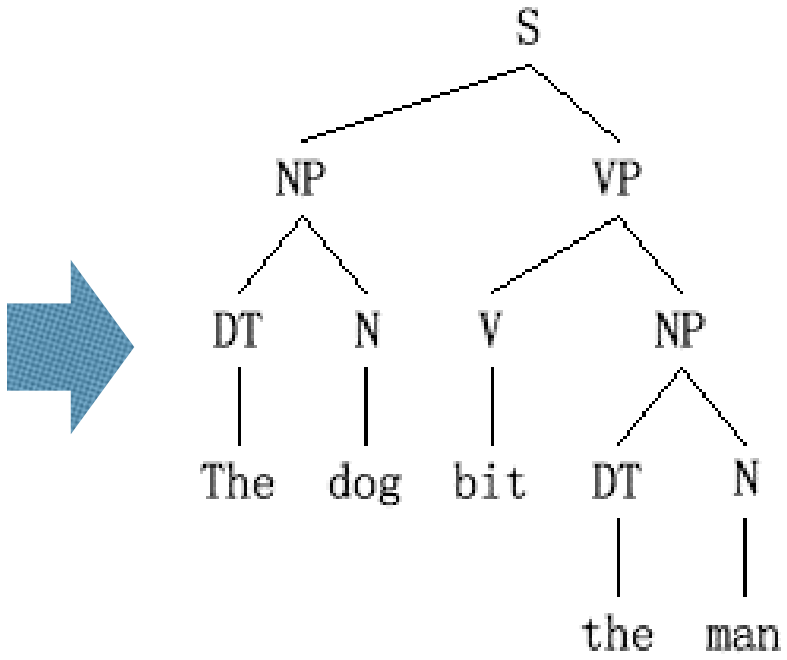
Levels of Language Processing

Phonology

Morphology

Syntax

The dog bit the man



Levels of Language Processing

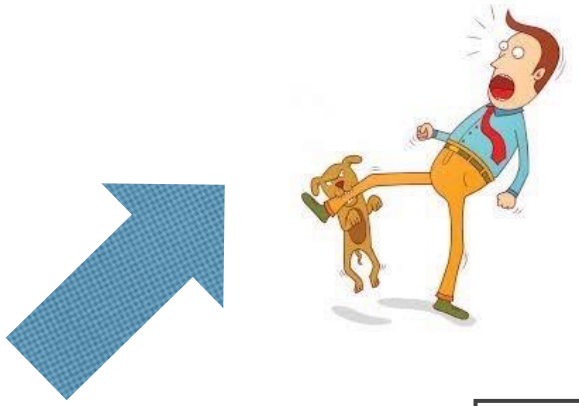
Phonology

Morphology

Syntax

Semantics

The dog bit the man



exists x,y:
    Dog(x)
    Man(y)
BITE(x,y)

Text documents

- Text data units
- Preprocessing text data
- Vectorization
- Text analysis flow and examples
- Text Analysis example

Text data units – different granularity

The basic units of text processing:

- Corpus- document collection
- Document
- Section
- Paragraph
- Sentence
- Phrase
- Word
- Character level

Text data units – different granularity

Document --> label

- Language classification
- Topic classification
- Author classification
- Sentiment classification

Sentence --> label

- Usually the same as in "document --> label"
  shorter text --> harder task

Question: are each of these binary / multi-class / multi-label?

Text data units – different granularity

Word --> label

- Tokens vs. Types

  Type --> label

  - Sad vs happy words
  - Adjectives vs. Nouns

  Token --> label

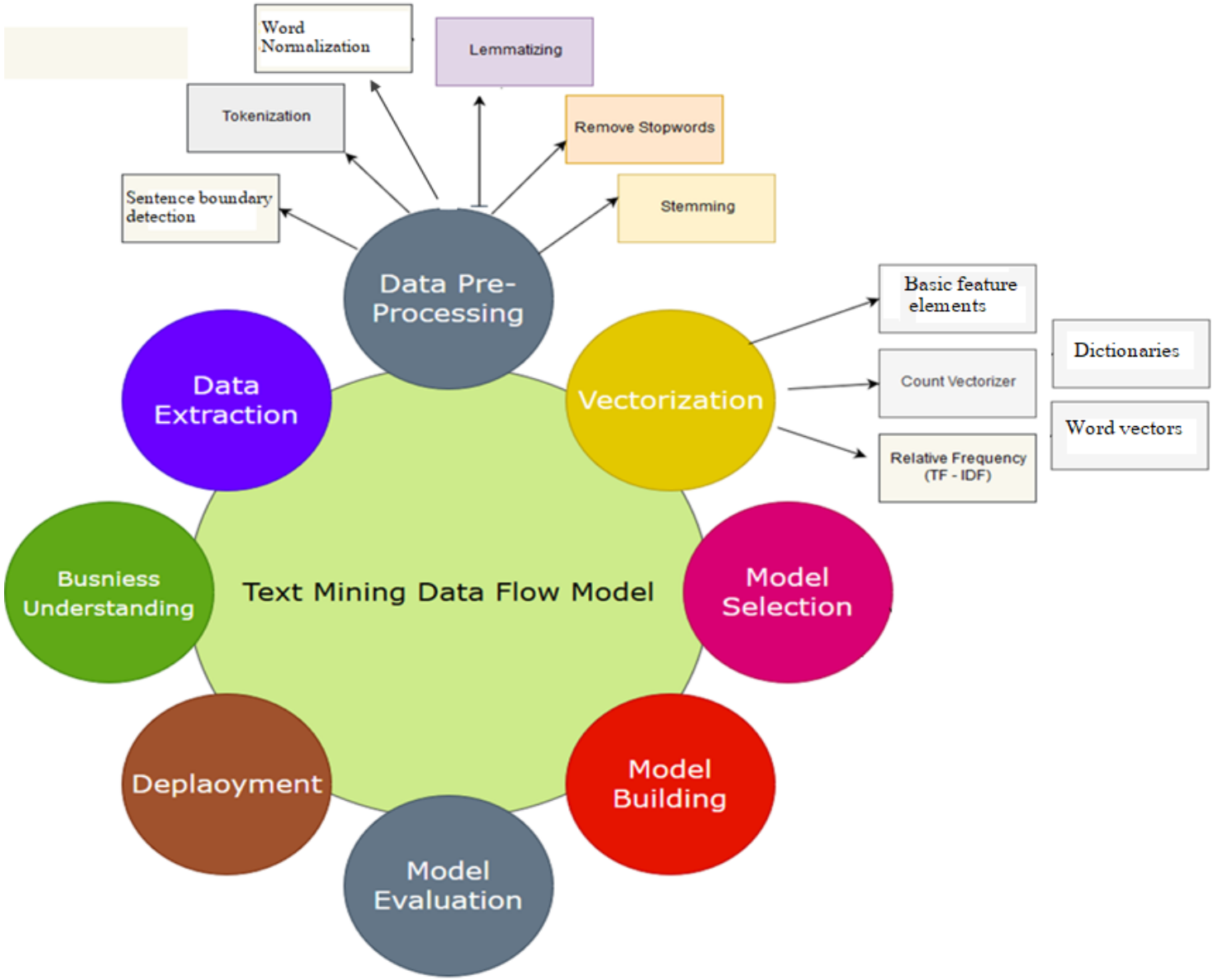
  - Sentence boundary detection
  - Common spelling mistakes ("then" vs "than")

Retrieved Text Documents

What is the basic atom level of text we should relate to?

How do we retrieve documents?

Basic Text Processing & Analysis Flow



What is a word?

sequence of characters (perhaps letters)?

white-space bounded?

Input:

Friends, Romans, Countrymen, lend me your ears;

→ whitespace might not be enough

Tokenization

Given a character sequence and a defined document unit, tokenization is the task of chopping it up into pieces, called *tokens* (perhaps at the same time throwing away certain characters), such as punctuation.

Here is an example of tokenization:

Input:

Friends, Romans, Countrymen, lend me your ears;

Output:

- Friends
- Romans
- Countrymen
- lend
- me
- your
- ears

Tokenization: dividing a text into tokens.

What is a word?
Twitter Text example

sequence of characters (perhaps letters)?

@dbamman have you seen this :) http://popvssoda.com

- @
- dbamman
- have
- you
- seen
- this
- :
-)
- http
- :
- /
- /
- popvssoda
- .
- com

Words, Sentences and Punctuation

We typically don't want to just strip all punctuation, however.

- Punctuation signals boundaries (sentence, clausal boundaries, parentheticals, asides)
  - Some punctuation has illocutionary force, like exclamation points (!) and question marks (?)
  - Emoticons are strong signals of e.g., sentiment
- Most tokenization algorithms (for languages typically delimited by whitespace) use regular expressions to segment a string into discrete tokens.

How much can we rely on white spaces?

Tokenization in languages without spaces

Many languages, such as Mandarin (Chinese), Japanese and Thai don't use spaces to separate words!

How do we decide where the token boundaries should be?

Tokenization in Mandarin

Mandarin words are composed of characters called "**hanzi**" (or sometimes just "**zi**")

Each one represents a meaning unit called a morpheme.

Each word has on average 2.4 of them.

But deciding what counts as a word is complex and not agreed upon.

How to do tokenization in Mandarin?

•姚明进入总决赛 “Yao Ming reaches the finals”

•3 words?
•姚明 进入 总决赛
•YaoMing reaches finals

•5 words?
•姚 明 进入 总 决赛
•Yao Ming reaches overall finals

•7 characters? (don't use words at all):
•姚 明 进 入 总 决 赛
•Yao Ming enter enter overall decision game

Tokenization / segmentation

So, in Chinese it's common to just treat each character (zi) as a token.

- So, the **segmentation** step is very simple

In other languages (like Thai and Japanese), more complex word segmentation is required.

- The standard algorithms are neural sequence models trained by supervised machine learning.

Word / token normalization

Punctuation

- U.S.A. → USA (acronym)
- Ltd. → Ltd

Word / token normalization

Case folding

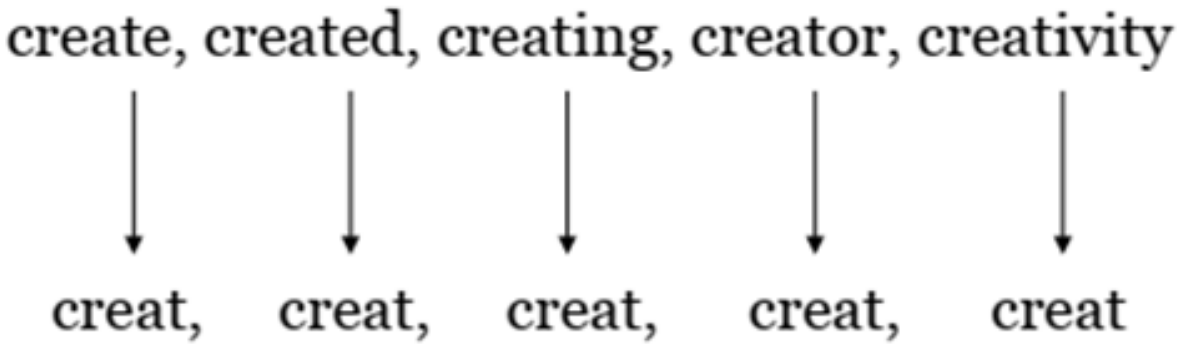
- General Motors → general motors

Risk:

US → us

Stemming

Stem: a "base form", based on heuristics.



Stemming: cutting the inflected words to their root form.

How should we treat Hebrew and Arabic? Approach 1 - Transliteration Scheme

Each character is represented as a Latin character

א	ב	ג	ד	ה	ו	ז	ח	ט	י	כ	ל	מ	נ	ס	ע	פ	צ	ק	ר	ש	ת
a	b	g	d	h	w	z	x	v	i	k	l	m	n	s	y	p	c	q	r	e	t

Latin	Arabic	Latin	Arabic	Latin	Arabic	Latin	Arabic	Latin	Arabic
a	ا	O	ؤ	b	ب	n	ن	N	ڤ
A	ه	U	ۇ	G	ع	t	ت	d	د
e	ى	x	ش	p	پ	q	ق	y	ي
i	ئا	s	س	c	چ	f	ف	w	ۋ
o	و	m	م	H	ح	k	ك	z	ز
u	ۇ	l	ل	j	ج	g	گ	r	ر

For basic difficulty of the usage for these languages:

- Right-to-left
- Non-Unicode tools

Drawback:

- Not supported for most tools.

How should we treat Hebrew & Arabic? Approach 2 – Adapting a general approach

- Tokenization for Hebrew & Arabic
- What is a word/token in Hebrew & Arabic
- Adding Morphology
- Word normalization for Hebrew & Arabic

עכשיו בעברית – Tokenization

Given a character sequence and a defined document unit, tokenization is the task of chopping it up into pieces, called *tokens* (perhaps at the same time throwing away certain characters), such as punctuation.

Here is an example of tokenization:

Input:

אתמול, 8.6.2018, בשעה 17:00, הלכתי עם אמא למכולת.

Output:

- אתמול
- 8.6.2018
- בשעה
- 17:00

- הלכתי
- עם
- אמא
- למכולת

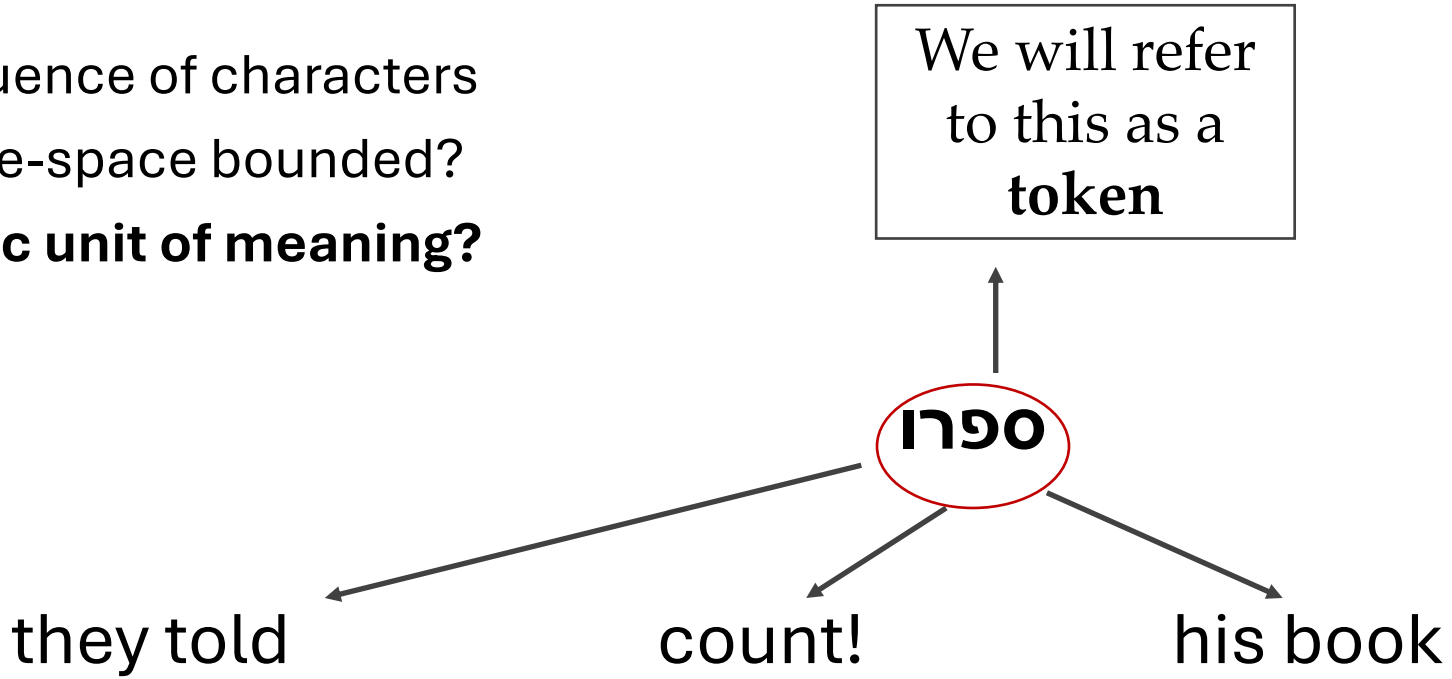
Tokenization: dividing a text into tokens.

עברית מקרה מסובך יותר – What is a word?

sequence of characters

white-space bounded?

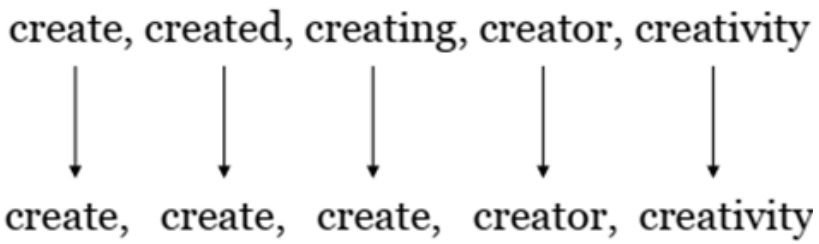
basic unit of meaning?



→ whitespace might not be enough

Morphological analysis

Lemma: the "dictionary entry" of a word.

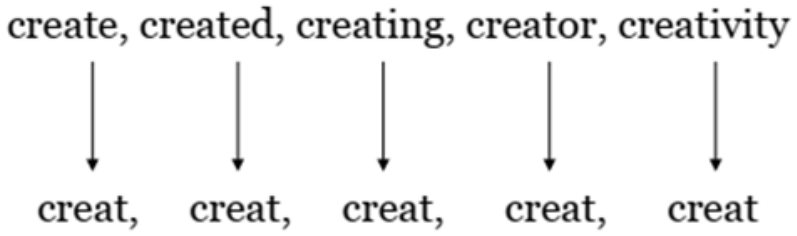


Stemming: cutting the inflected words to their root form.

ספרו - ה-ספר-של-הוא

Lemmatization: reducing the inflected forms of a word into a single form for easy analysis.

Stem: a "base form", based on heuristics.



Word / token normalization

Morphology

Morphemes - The small meaningful units that make up words

- Lemma: The core meaning--bearing units
- Affixes: Bits and pieces that adhere to stems
  - Often with grammatical functions
    - יותר משמעותי בעברית וערבית (שפות בהם יחסית יש הרבה מורפולוגיה)

Word / token normalization - נרמול בעברית (פיסוק)

פיסוק – קיצורים, ראשי תיבות

- ב.ע.מ. --> בעמ
- דוא"ל --> דואל
- וכו'
- די. וי. איי. (לועזית)

סיכון:

- גב' --> גב
- ד"ר --> דר

Word / token normalization - נרמול בעברית (ניקוד)

פיסוק – קיצורים, ראשי תיבות

- פְּרִי --> פרי
- צוּרָה --> צורה

סיכון:

- דֶּרֶךְ --> דרך (איבוד מידע, דרך --> דֶּרֶךְ או דָּרַךְ)

Basic useful preprocessing operation - summary

Tokenization (sometimes segmentation): dividing a text into tokens.

Input: “I love playing soccer with my friends, mostly on the weekends!”
Output: ["I", "love", "playing", "soccer", "with", "my", "friends", “mostly”, "on", "the", "weekends", "!"]

Stemming: cutting the inflected words to their root form.

Input: "The **mice** in the **fields were running** and **jumping** around."
Output: " The **mice** in the **field were run** and **jump** around .”

Lemmatization: reducing the inflected forms of a word into a single form for easy analysis.

Input: "The **mice** in the **fields were running** and **jumping** around.”
Output: "The **mouse** in the **field be run** and **jump** around”

Other Useful operations:

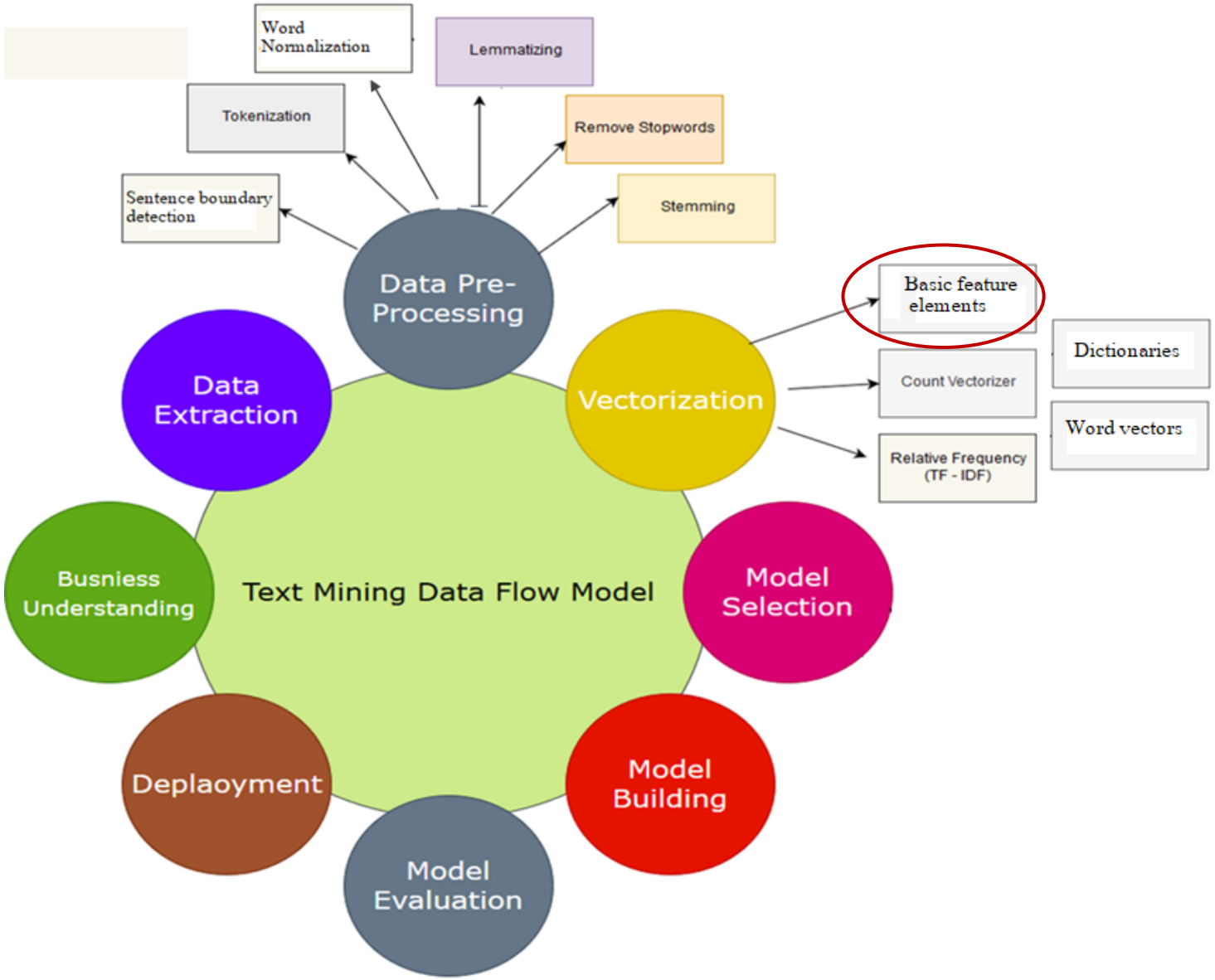
Sentence breaking: placing sentence boundaries on a text.

Word Normalizations and cleaning

Stop word removal: removing words, which are considered as such.

Part-of-speech tagging: identifying the part of speech for every word.

Basic Text Processing & Analysis Flow



Text as data – properties and challenges

Sparse data

- Zipf's law, variability

Symbolic

- abstract symbol to meaning mapping, variability, ambiguity

Many levels of granularity

- document, paragraph, sentence, word, characters

How would these affect a classifier over text data?

Vectorization:
from document to feature vector

The bag of words (BOW) model:

Each **word** is treated as a feature in a unit called **document**.

Each such word will become a feature

How do we measure their strength?

- Word Count
  - What about zipf law?

Vectorization:
extracting basic feature units

The bag of words (BOW) model:

Each **word** is treated as a feature in a unit called **document**.

Each such word will become a feature

- Alternative: **original tokens – no processing**
- Alternative: **normalized words – e.g., lemmas, stems**
- Alternative: **partial word (prefix, suffix)**
- Alternative: **ngrams – unigram, bigram, trigram**
- Alternative: **characters – we will usually use with ngrams**
- More complex alternatives …

Vectorization:
extracting basic feature units

The bag of words (BOW) model:

Each **word** is treated as a feature in a unit called **document**.

Each such word will become a feature

- Alternative: **normalized words – e.g., lemmas, stems**

Lemma: the "dictionary entry" of a word.

create, created, creating, creator, creativity

↓ ↓ ↓ ↓ ↓

create, create, create, creator, creativity

Stem: a "base form", based on heuristics.

create, created, creating, creator, creativity

↓ ↓ ↓ ↓ ↓

creat, creat, creat, creat, creat

Vectorization:
extracting basic feature units

The bag of words (BOW) model:

Each **word** is treated as a feature in a unit called **document**.

Each such word will become a feature

	about	bird	heard	is	the	word	you
About the bird, the bird, bird bird bird	1	5	0	0	2	0	0
You heard about the bird	1	1	1	0	1	0	1
The bird is the word	0	1	0	1	2	1	0

Feature Vectors

(1, 5, 0, 0, 2, 0, 0)

(1, 1, 1, 0, 1, 0, 1)

(0, 1, 0, 1, 2, 1, 0)

Vectorization:
extracting basic feature units

The bag of words (BOW) model:

Each **word** is treated as a feature in a unit called **document**.

Each such word will become a feature

- Alternative: **ngrams – unigram, bigram, trigram**

ngrams: **unigrams** **bigrams** **trigrams**

['the','special', 'onion', 'soup', 'was', 'not', 'very', 'bad',
'the special', 'special onion', 'onion soup', 'soup was',
'was not', 'not very', 'very bad', 'the special onion', 'special onion soup',
'onion soup was', 'soup was not', 'was not very', 'not very bad']

Until the next time ☺