Deep Learning with Applications in Natural Language Processing

Mihaela Breabăn
Mădălina Răschip
Diana Trandabăț

Academic year 2024-2025

Course structure

- Basic text processing (tokens, lemmas, edit distance, POS tagging)
- Syntactic structure and dependency parsing
- Language modelling: statistical approaches
- Machine translation - traditional approaches
- Question answering. Sentiment analysis
- Topic modelling
- Neural network architectures - a review with a focus on NLP (feed forward, CNNs explained for training and applications in NLP)
- Recurrent neural networks. Attention mechanisms
- Text vectorization: word2vec, glove
- Text classification (multinomial Naive Bayes, maximum entropy classifier, multinomial logistic regression)
- Transformers. BERT

Evaluation criteria

Course	Written test	40%
Seminary/ Laboratory	Weekly lab work evaluation during the first half of the semester	30%
Project	Evaluating the progress of the project during the second half of the semester and final presentation	30%

Introduction

Basic text processing steps

- tokens
- lemmas
- stems
- POS tagging
- edit distance
- corpora

How do people communicate?

Different ways:

- speaking and listening
- making gestures
- specialized hand signals (such as when driving or directing traffic)
- sign languages for the deaf
- various forms of text

Communication breakdown

- S (speaker) wants to convey P (proposition) to H (hearer) using W (words in a formal or natural language)

1. Speaker

- **Intention:** S wants H to believe P
- **Generation:** S chooses words W
- **Synthesis:** S utters words W

2. Hearer

- **Perception:** H perceives words W” (ideally W” = W)
- **Analysis:** H infers possible meanings P1,P2,…,Pn for W”
- **Disambiguation:** H infers that S intended to convey Pi (ideally Pi=P)
- **Incorporation:** H decides to believe or disbelieve Pi

What is NLP?

NLP - subdomain of Artificial Intelligence

≈ computational linguistics

≈ human language technologies

Goal: communication human-machine in natural language

Two major NLP directions

1. Natural Language Understanding and Analyzing
   - Input: spoken/written sentence
   - Output: some representation of the meaning of the sentence
2. Natural Language Generation
   - Input: some formal representation of what you intend to communicate
   - Output: expression of what we want to convey in a natural (human) language, i.e. a text or speech

Examples of NLP applications

- Machine Translation
- Database Access
- Information Retrieval
- Text Categorization
- Extracting data from text
- Spoken language control systems
- Spelling and grammar checkers
- Document similarity (plagiarism, fake news)
- Discourse
- etc. etc. etc.

Understanding a text

Goliat, the first Romanian nanosatellite, was successfully launched on the orbit Monday, 13th February 2012, from a base in the Frech Guyana, during the inaugural flight of the VEGA rocket. The satellite has been developed by a research team headed by the Romanian Space Agency between 2005-2009.

Source: http://science.hotnews.ro/stiri-spatiul-11496155-live-video-ora-12-00-goliat-primul-satelit-romanesc-lansat-orbita.htm

Understanding a text - morphology

By morphological analysis we can identify part of speeches:

- Nouns;
- Verbs
- etc.

Goliat, the first Romanian nanosatellite, was successfully launched on the orbit Monday, 13th February 2012, from a base in the Frech Guyana, during the inaugural flight of the VEGA rocket. The satellite has been developed by a research team headed by the Romanian Space Agency between 2005-2009.

Source: http://science.hotnews.ro/stiri-spatiul-11496155-live-video-ora-12-00-goliat-primul-satelit-romanesc-lansat-orbita.htm

Understanding a text - syntax

By syntactical analysis we can identify grammatical constituents:

- Subject;
- Predicate;
- etc.

Goliat, the first Romanian nanosatellite, was successfully launched on the orbit Monday, 13th February 2012, from a base in the Frech Guyana, during the inaugural flight of the VEGA rocket. The satellite has been developed by a research team headed by the Romanian Space Agency between 2005-2009.

Source : http://science.hotnews.ro/stiri-spatiul-11496155-live-video-ora-12-00-goliat-primul-satelit-romanesc-lansat-orbita.htm

Understanding a text - semantics

By semantical analysis we can understand a text:

- Who, what, where, when, how, why etc. peforms an action
- The meaning of words
- References/Anaphora

Goliat, the first Romanian nanosatellite, was successfully launched on the orbit Monday, 13th February 2012, from a base in the Frech Guyana, during the inaugural flight of the VEGA rocket. The satellite has been developed by a research team headed by the Romanian Space Agency between 2005-2009.

Source : http://science.hotnews.ro/stiri-spatiul-11496155-live-video-ora-12-00-goliat-primul-satelit-romanesc-lansat-orbita.htm

Understanding a text

01000111011011110110110001101001011000010111010000101100001
00000011100000011100100110100101101101011101010110110000 1000
000110111001100001011011100110111101110011011000010111010000
11001010110110001101001011101000010000001110010011011110110
110111000011101000100110111001100101011100110110 0011......

Goliat, the first Romanian nanosatellite, was successfully launched on the orbit Monday, 13th February 2012, from a base in the Frech Guyana, during the inaugural flight of the VEGA rocket. The satellite has been developed by a research team headed by the Romanian Space Agency between 2005-2009.

Source : http://science.hotnews.ro/stiri-spatiul-11496155-live-video-ora-12-00-goliat-primul-satelit-romanesc-lansat-orbita.htm

Tokenization

Breaking up a stream of characters into **tokens**: words, punctuation marks, numbers and other discrete items

Goliat, the first Romanian nanosatellite, was successfully launched on the orbit Monday, 13th February 2012, from a base in the Frech Guyana, during the inaugural flight of the VEGA rocket. The satellite has been developed by a research team headed by the Romanian Space Agency between 2005-2009.

Source : http://science.hotnews.ro/stiri-spatiul-11496155-live-video-ora-12-00-goliat-primul-satelit-romanesc-lansat-orbita.htm

How many tokens?

今天天气晴朗

it's sunny today

c'est ensoleillé aujourd’hui

Heute ist ein Sonnentag

What can we learn just through tokenization?

- Text statistics: no. of words, multi-word expressions, length of words/sentences, freq. of vowels/consonants > Language Identification

Goliat, the first Romanian nanosatellite, was successfully launched on the orbit Monday, 13th February 2012, from a base in the Frech Guyana, during the inaugural flight of the VEGA rocket. The satellite has been developed by a research team headed by the Romanian Space Agency between 2005-2009.

Source : http://science.hotnews.ro/stiri-spatiul-11496155-live-video-ora-12-00-goliat-primul-satelit-romanesc-lansat-orbita.htm

What can we learn just through tokenization?

- Text statistics: no. of words, multi-word expressions, length of words/sentences, freq. of vowels/consonants > Language Identification
- Named Entity Recognition

Goliat, the first Romanian nanosatellite, was successfully launched on the orbit Monday, 13th February 2012, from a base in the Frech Guyana, during the inaugural flight of the VEGA rocket. The satellite has been developed by a research team headed by the Romanian Space Agency between 2005-2009.

Source : http://science.hotnews.ro/stiri-spatiul-11496155-live-video-ora-12-00-goliat-primul-satelit-romanesc-lansat-orbita.htm

Named Entity Recognition

Entity = "something that exists by itself; something that is separate from other things; the existence of a thing as contrasted with its attributes" (Merriam-Webster dictionary)

Named Entities = *objects ("entities") from the real world that have a name*

-> persons, locations, organizations

-> time expressions, diseases, chemicals, laws, legal references, emails, bank accounts, currency, …

Why Named Entity Recognition?

Information extraction

- Find information relevant to a set of entities
- Extract text related to a particular product / brand / political figure etc.

Content recommendation

- Recommend content with the same NEs

Customer support

- Automatically show relevant information from different systems about identified NEs

Anonymization

- Anonymize persons, organizations, etc.

Computational morphology

- Computational morphology deals with
  - developing theories and techniques for computational analysis and synthesis of **word forms**.

- Analysis: Separate and identify the constituent morphemes and mark the information they encode
- Synthesis (Generation): Given a set constituent morphemes or information be encoded, produce the corresponding word(s)

- Morphemes can be
  - suffixes (at the end of the word): plann*ing*
  - prefixes (at the beginning): *re*do
  - or both: *un*believ*able*

Computational Morphology -Analysis

- Extract any information encoded in a word and bring it out so that later layers of processing can make use of it

stopping	⇒ stop+Verb+Cont
happiest	⇒ happy+Adj+Superlative
went	⇒ go+Verb+Past
books	⇒ book+Noun+Plural
	⇒ book+Verb+Pres+3SG.

Computational Morphology -Generation

- In a machine translation applications, one may have to generate the word corresponding to a set of features
  - stop+Past ⇒ stopped
  - cânta+Past+1Pl ⇒ cântaserăm/cântasem

    +2Pl ⇒ cântaserăți/cântasei

Stemming

- Suffixes are often added to words for inflection.
- However, they are not imperatives to understand the meaning of a word.
- Stemming gets rid of the unneeded parts of a word, while keeping its *root*, also called the "**stem**“.
- Based on reduction rules (sses > ss; ies > i/y; etc.)

Word	Stem
connected	connect
connections	connect
connects	connect

Stemming

- However, sometimes it deletes too much do differentiate between meanings.

Word	Stem
university	univers
universe	univers

Lemmatization

- An algorithm that replaces a word by its most basic form, also called "**lemma**".
- A lemma can be an infinitive form, a noun, an adjective, etc. (usually a *dictionary form*), while a stem often has no meaning.
- Lemmatization requires a morphological analysis of the word and the existence of a detailed dictionary for the algorithm to work on, making it more complex to implement than stemming.

Form	Morphological info	Lemma
studies	Study (n) + pl.	study
studying	Study (v) + ing	study
are	Be + 3rd pl.	be
is	Be + 1st. sg	be

Natural Language Processing

... The first Romanian nanosatellite was ... launched ...

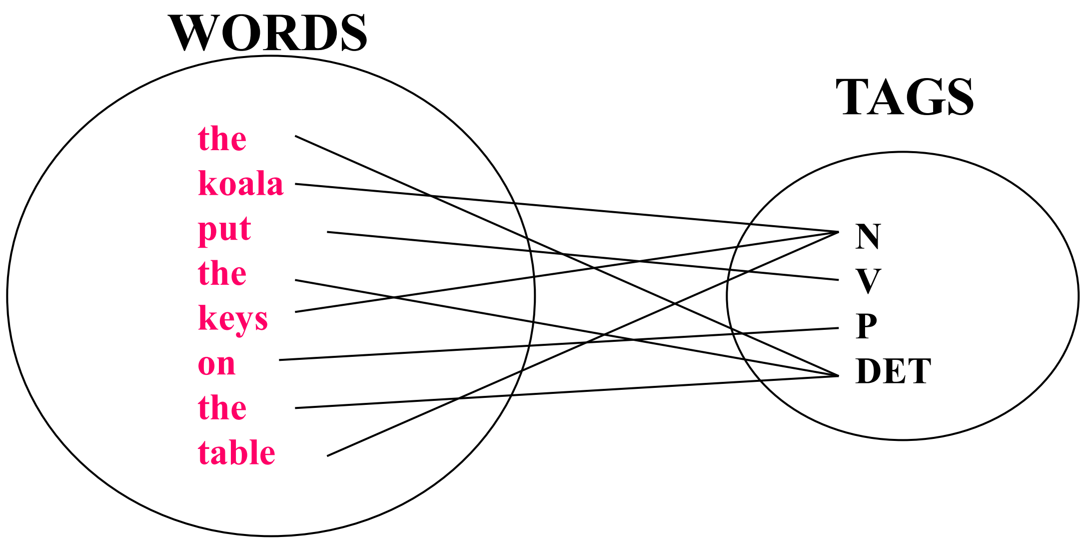
- Syntax:
  - Use a dictionary to identify part of speeches
    - First = numeral;
    - nanosatellite = noun;
    - launched = verb; ...
  - Difficulty: ambiguity
    - I like research
    - I research natural language processing

Penn Treebank Tagset

Tag	Description	Example	Tag	Description	Example
CC	Coordin. Conjunction	*and, but, or*	SYM	Symbol	+,%, &
CD	Cardinal number	*one, two, three*	TO	"to"	*to*
DT	Determiner	*a, the*	UH	Interjection	*ah, oops*
EX	Existential 'there'	*there*	VB	Verb, base form	*eat*
FW	Foreign word	*mea culpa*	VBD	Verb, past tense	*ate*
IN	Preposition/sub-conj	*of, in, by*	VBG	Verb, gerund	*eating*
JJ	Adjective	*yellow*	VBN	Verb, past participle	*eaten*
JJR	Adj., comparative	*bigger*	VBP	Verb, non-3sg pres	*eat*
JJS	Adj., superlative	*wildest*	VBZ	Verb, 3sg pres	*eats*
LS	List item marker	*1, 2, One*	WDT	Wh-determiner	*which, that*
MD	Modal	*can, should*	WP	Wh-pronoun	*what, who*
NN	Noun, sing. or mass	*llama*	WP$	Possessive wh-	*whose*
NNS	Noun, plural	*llamas*	WRB	Wh-adverb	*how, where*
NNP	Proper noun, singular	*IBM*	$	Dollar sign	*$*
NNPS	Proper noun, plural	*Carolinas*	#	Pound sign	#
PDT	Predeterminer	*all, both*	"	Left quote	(' or ")
POS	Possessive ending	*'s*	"	Right quote	(' or ")
PRP	Personal pronoun	*I, you, he*	(	Left parenthesis	([, (, {, <)
PRP$	Possessive pronoun	*your, one's*	)	Right parenthesis	( ],), }, >)
RB	Adverb	*quickly, never*	,	Comma	,
RBR	Adverb, comparative	*faster*	.	Sentence-final punc	(. ! ?)
RBS	Adverb, superlative	*fastest*	:	Mid-sentence punc	(: ; ... – -)
RP	Particle	*up, off*			

Defining POS Tagging

- The process of assigning a part-of-speech or lexical class marker to each word in a corpus:



Applications for POS Tagging

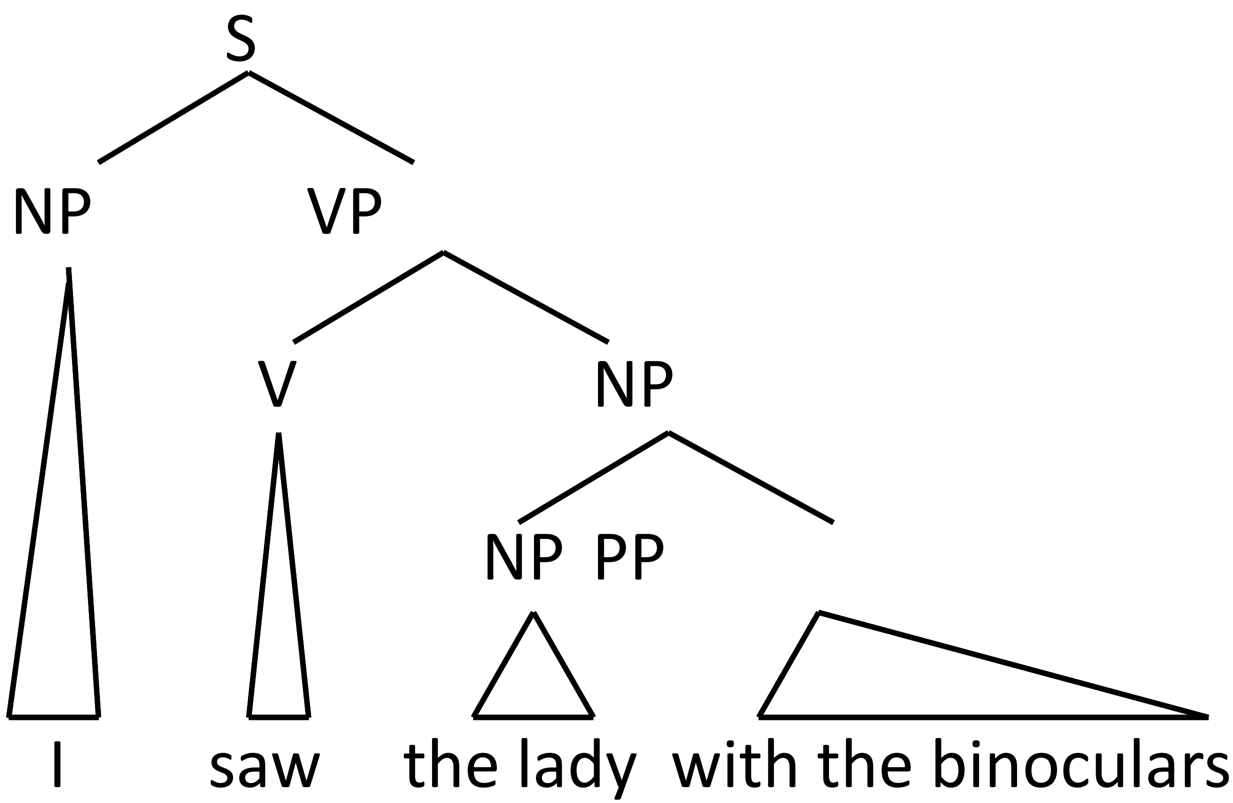
- Speech synthesis pronunciation
  - *Lead* *Lead*
  - *INsult* *inSULT*
  - *OBject* *objECT*
  - *OVERflow* *overFLOW*
  - *DIScount* *disCOUNT*
  - *CONtent* *conTENT*
- Word Sense Disambiguation: e.g. *Time flies like an arrow*
  - Is *flies* an N or V?
- Word prediction in speech recognition
  - Possessive pronouns (*my, your, her*) are likely to be followed by nouns
  - Personal pronouns (*I, you, he*) are likely to be followed by verbs
- Machine Translation

Natural Language Processing

... The first Romanian nanosatellite was ... launched ...

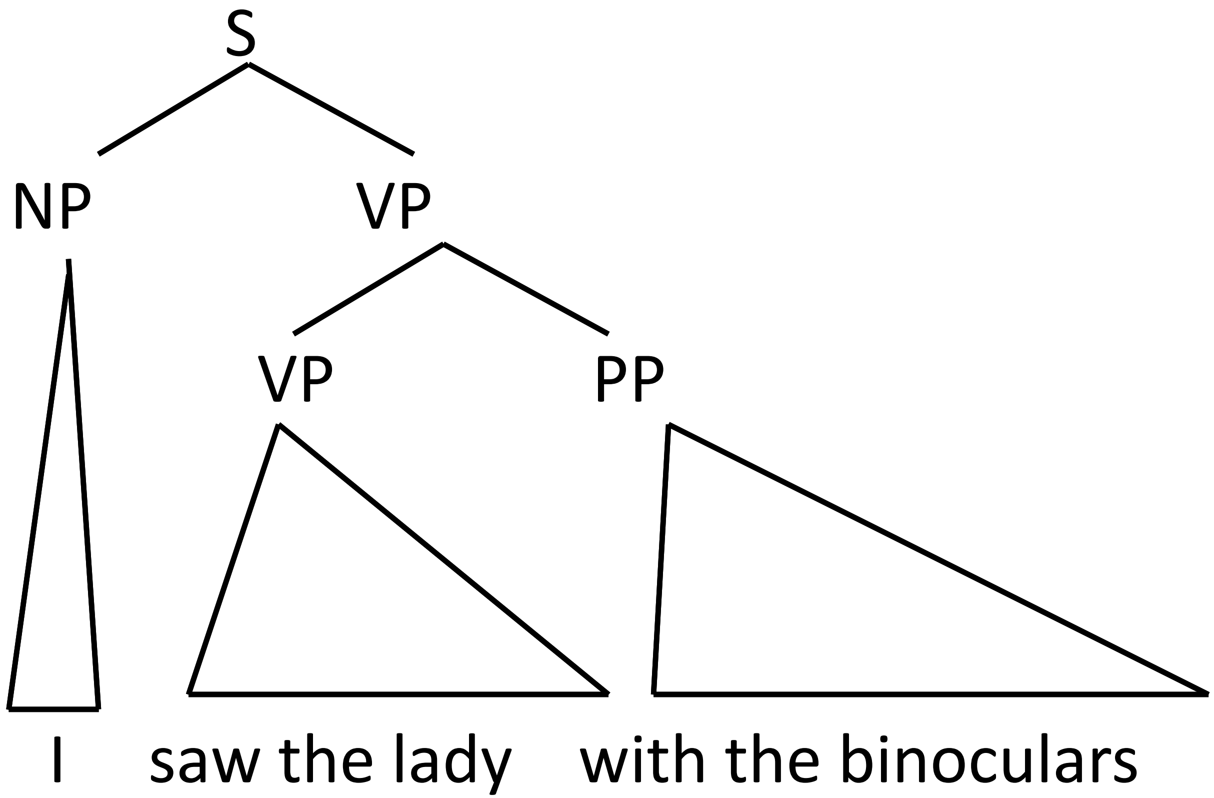
- Syntax:
  - coded in formal grammmars
    - determiner + numeral + noun+ adjectiv = noun group;
    - auxiliary + verb= verb group;
    - noun group + verb group = sentence; ...
  - extracted from corpora
  - Difficulty: Ambiguity
    - I saw the lady with the binoculars.

I saw the lady with the binoculars



I saw [the lady with the binoculars]

I saw the lady with the binoculars



I [saw the lady] [with the binoculars]

Natural Language Processing

... The first Romanian nanosatellite was ... launched ...

RSA lauched the first Romanian nanosatellite

- Sematics:
  - Identify semantic roles around a predicate
    - X (subject) is launched by Y (object);
    - Y (subject) launches X (object).
  - => concepts

Similarity Models

- Similarity models measure how alike are two objects (products, patients, molecules, words, sentences, . . .).
- Objects (words, sentences, documents...) are represented as feature-vectors, feature-sets, distribution-vectors, etc.
- Similarity may also be interpreted as proximity or affinity
- Similarity may also be seen as the opposite of distance, difference, or divergence.
- Different uses and applications in AI.

Similarity in NLP

- **Text similarity tasks**: Plagiarism detection, news items tracking, related readings recommendation, question answering, FAQ management, ...
- **Text analysis tasks**: Tasks such as PoS Tagging, parsing, NERC, etc can be approached using EBL.
- **Text Classification tasks**: (EBL, again). E.g.: news items routing, sentiment analysis, spam detection, ...
- **Evaluation of NL generation tasks**: Evaluate machine translation, automatic summarization, or report generation comparing the system output with reference texts.
- **Alias detection**: (Useful for coreference detection) find different mentions of the same entity (e.g. *Stanford President John Hennessy*, *Stanford University President Hennessy*, *President John Hennessy*, *Stanford Provost John Hindirck*).

Distance, Similarity, & Relatedness

- We talk about *distance* when metric properties hold:
  - $d(x, x) = 0$
  - $d(x, y) > 0$ when $x \neq y$
  - $d(x, y) = d(y, x)$ (simmetry)
  - $d(x, z) \leqslant d(x, y) + d(y, z)$ (triangular inequation)
- We use *similarity* in the general case
  - Function: $sim : A \times B \to S$ (where S is often $[0, 1]$)
  - Homogeneous: $sim : A \times A \to S$ (e.g. word-to-word)
  - Heterogeneous: $sim : A \times B \to S$ (e.g. word-to-document)
  - Not necessarily symmetric, or holding triangular inequation.
- We can compute one from the other:

$$sim(A, B) = \frac{1}{1 + d(A, B)}; \quad d(A, B) = \frac{1}{sim(A, B)} - 1$$

- *Similarity* is often interpreted as a measure of *relatedness*.

Information used for similarity

The utility/meaning of a similarity/distance measure depends on how compared objects are represented.

- Information internal to compared units
  - Words: char n-grams, word form, lemma, morphology, PoS, sense, domain, ...
  - Sentences/Documents: bag of words, parse tree, syntactic roles, collocations, word n-grams, Named Entities, ...
- Information external to compared units (context)
  - Words: bag-of-words in context, parse tree, collocations, word n-grams, Named Entities, ...
  - Sentences/Documents: Words in nearby sentences, document meta-information, ...

Approaches to Similarity Computation

- **String/Sequence edit-distance approaches.**
  Can only be applied to sequences of elements (characters, words, proteins...)
- **Vector/Set based approaches.**
  General approach, can be applied to any kind of object once we represent it as a [feature] vector or set.
  - Vector similarities/distances
  - Set similarities/distances
- **Knowledge-based approaches.**
  Require some (graph-like) knowledge representation.
  - WordNet distances
- **Corpus-based approaches (distributional semantics).**
  Describe meaning based on occurrence contexts.
  - Sparse representations (term-term/term-document matrix)
  - Dense representations (LSI, Word Embeddigns)

I. Edit distance

- A manner of quantifying how dissimilar two strings (words) are.
- Counts the minimum number of operations required to transform one string into the other.
- **Applications**
  – Automatic spelling correction to determine candidate corrections by selecting from a dictionary words that have a low distance to the target word;
  – Evaluation of machine translation;
  – Speech recognition.

String/Sequence edit-distance approaches

Sequences of any kind

- word : sequence of characters
- sentence : sequence of words (or characters too)
- DNA: sequence of bases A,T,C,G
- Health Record : sequence of clinical events
- ...

Some Edit Distances

- LCS (Longest Common Subsequence): ED allowing deletion and insertion.
- Levenhstein: ED allowing deletion, insertion and substitution.
- Damerau-Levenhstein: ED allowing insertion, deletion, substitution, and transposition of two adjacent elements.

Edit distances can be efficiently computed using dynammic programming.

Edit distance

- Editing operations:
  - Insertion
  - Deletion
  - Substitution

```
I N T E * N T I O N
| | | | | | | | | |
* E X E C U T I O N
d s s   i s
```

kitten → **s**itten (substitution of “s” for “k”)

sitt**e**n → sitt**i**n (substitution of “i” for “e”)

sittin → sittin**g** (insertion of “g” at the end).

- Different operations can have different weights

Example: Levenhstein

```
def Levenshtein(s, t):

    n = len(s)
    m = len(t)
    d = [ [ 0 for j in range(0,m+1) ] for i in range(0,n+1) ]

    # source prefixes can be transformed into empty string by
    # dropping all characters
    for i in range(1,n+1): d[i][0] = i

    # target prefixes can be reached from empty source prefix
    # by inserting every character
    for j in range(1,m+1): d[0][j] = j

    for i in range(1,n+1):
        for j in range(1,m+1):

            subst = 0 if s[i-1] == t[j-1] else 1   # substitution cost

            d[i][j] = min(d[i-1][j] + 1,           # deletion
                          d[i][j-1] + 1,           # insertion
                          d[i-1][j-1] + subst)     # substitution

    return d[n][m]
```

Levenstein

	λ	S	A	T	U	R	D	A	Y
λ	0	1	2	3	4	5	6	7	8
S	1								
U	2								
N	3								
D	4								
A	5								
Y	6								

Levenstein

	λ	S	A	T	U	R	D	A	Y
λ	0	1	2	3	4	5	6	7	8
S	1	0	1	2	3	4	5	6	7
U	2	1	1	2	2	3	4	5	6
N	3	2	2	2	3	3	4	5	6
D	4	3	3	3	3	4	3	4	5
A	5	4	3	4	4	4	4	3	4
Y	6	5	4	4	5	5	5	4	**3**

Still Levenstein

	λ	*The*	*spokesman*	*said*	*the*	*senior*	*advisor*	*was*	*shot*	*dead*
λ	0	1	2	3	4	5	6	7	8	9
Spokesman	1	1	2	3	4	5	6	7	8	9
confirms	2	2	2	3	4	5	6	7	8	9
senior	3	3	3	3	4	4	5	6	7	8
government	4	4	4	4	4	5	5	6	7	8
advisor	5	5	5	5	5	5	5	6	7	8
was	6	6	6	6	6	6	6	5	6	7
shot	7	7	7	7	7	7	7	6	5	**6**

II. Vector similarities/distances

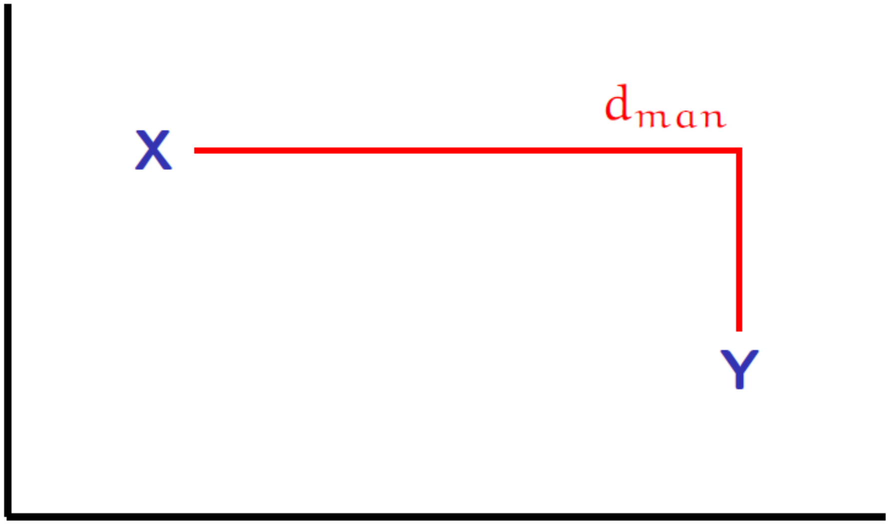
When objects are represented as [feature] vectors, we can use vector-space distances.

- Manhattan distance
- Euclidean distance
- Chebychev distance
- Camberra distance
- Cosine *similarity*
- Dot Product *similarity*
- ...

Vector similarities/distances

- L_1 norm, a.k.a. Manhattan distance, taxi-cab distance, city-block distance:

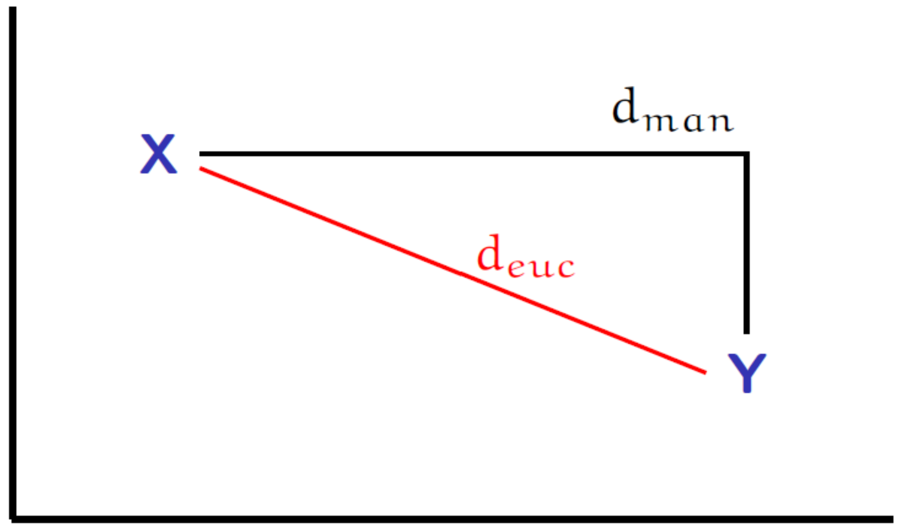
$$d_{man}(\vec{x}, \vec{y}) = L_1(\vec{x}, \vec{y}) = \sum_{i=1}^{N} |x_i - y_i|$$



Vector similarities/distances

- L_2 norm, a.k.a. Euclidean distance:

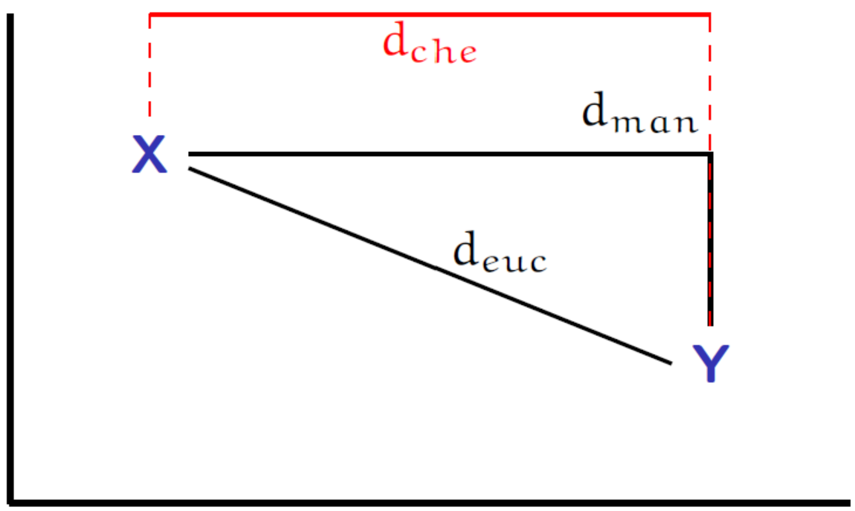
$$d_{euc}(\vec{x}, \vec{y}) = L_2(\vec{x}, \vec{y}) = |\vec{x} - \vec{y}| = \sqrt{\sum_{i=1}^{N} (x_i - y_i)^2}$$



Vector similarities/distances

- The limit of Minkowsky distance is Chebychev distance:

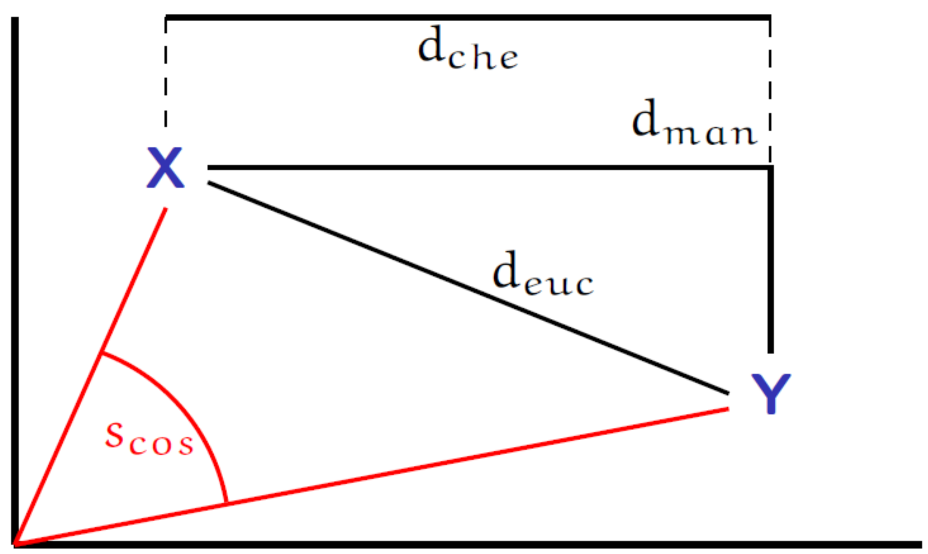
$$d_{che}(\vec{x}, \vec{y}) = L_\infty = \lim_{r \to \infty} L_r(\vec{x}, \vec{y}) = \max_i |x_i - y_i|$$



Vector similarities/distances

- Cosine is a similarity, not a distance:

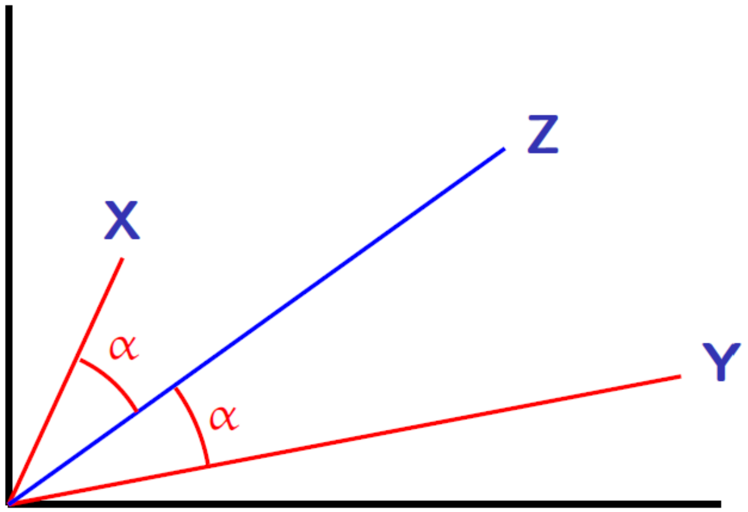
$$\mathrm{sim}_{\cos}(\vec{x},\vec{y}) = \cos(\vec{x},\vec{y}) = \frac{\vec{x}\cdot\vec{y}}{|\vec{x}|\cdot|\vec{y}|} = \frac{\sum_i x_i y_i}{\sqrt{\sum_i x_i^2}\cdot\sqrt{\sum_i y_i^2}}$$



Vector similarities/distances

- Dot product (or scalar product) is also similarity, that takes into account not only the angle but also the norm of the vectors:

$$\mathrm{sim}_{\mathrm{dot}}(\vec{x}, \vec{y}) = \vec{x} \cdot \vec{y} = \sum_i x_i y_i$$



$$\begin{aligned}\mathrm{sim}_{\cos}(X, Z) &= \mathrm{sim}_{\cos}(Y, Z) \\ &= \cos\alpha \approx 0.84\end{aligned}$$

$$\mathrm{sim}_{\mathrm{dot}}(X, Z) = X \cdot Z \approx 8.2$$

$$\mathrm{sim}_{\mathrm{dot}}(Y, Z) = Y \cdot Z \approx 21.3$$

Vector similarities/distances

$s_1 =$ *Spokesman confirms senior government advisor was shot*
$s_2 =$ *The spokesman said the senior advisor was shot dead*
$s_3 =$ *Spokesman said the shot government advisor was dead*

	Spokesman	*confirms*	*said*	*the*	*senior*	*government*	*advisor*	*was*	*shot*	*dead*
s_1	1	1	0	0	1	1	1	1	1	0
s_2	1	0	1	2	1	0	1	1	1	1
s_3	1	0	1	1	0	1	1	1	1	1

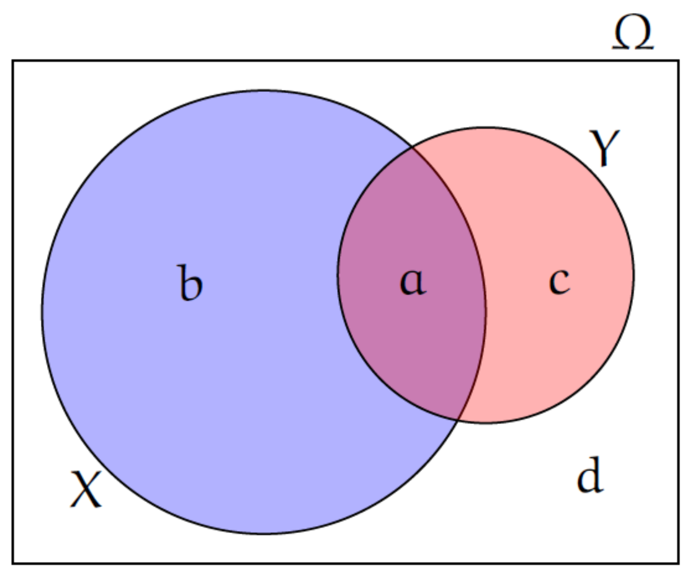
	d_{man}	d_{euc}	d_{che}
$s_1 \leftrightarrow s_2$	6	$\sqrt{8} = 2.83$	2
$s_1 \leftrightarrow s_3$	5	$\sqrt{5} = 2.24$	1
$s_2 \leftrightarrow s_3$	3	$\sqrt{3} = 1.73$	1

sim_{dot}	sim_{cos}
5	$\frac{5}{\sqrt{7}\sqrt{11}} = 0.57$
5	$\frac{5}{\sqrt{7}\sqrt{8}} = 0.67$
8	$\frac{8}{\sqrt{8}\sqrt{11}} = 0.85$

Set similarities/distances

- When objects are represented as [feature] sets (or binary-valued vectors) we can use set similarity measures
- These similarities are in $[0, 1]$ and can be converted to distances simply substracting: $d(X, Y) = 1 - sim(X, Y)$
- Easily computable using a contingency table:

		Y: 1	Y: 0	
X	1	a	b	a + b
	0	c	d	c + d
		a + c	b + d	



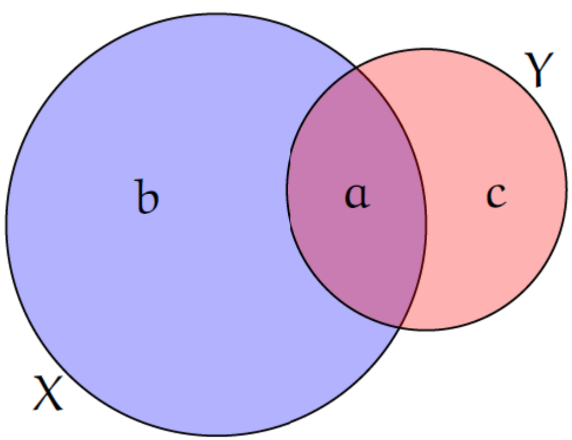
Set similarities/distances

- Dice.

$$\mathrm{sim}_{\mathrm{dic}}(X, Y) = \frac{2 \cdot |X \cap Y|}{|X| + |Y|} = \frac{2a}{2a + b + c}$$

- Jaccard.

$$\mathrm{sim}_{\mathrm{jac}}(X, Y) = \frac{|X \cap Y|}{|X \cup Y|} = \frac{a}{a + b + c}$$



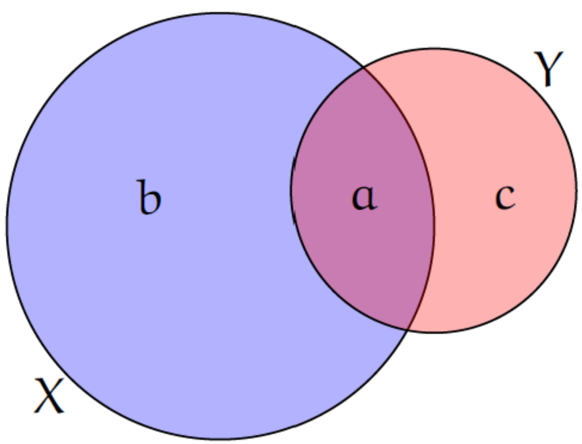
Set similarities/distances

- Overlap.

$$\mathrm{sim}_{ovl}(X, Y) = \frac{|X \cap Y|}{\min(|X|, |Y|)} = \frac{a}{\min(a + b, a + c)}$$

- Cosine.

$$\mathrm{sim}_{cos}(X, Y) = \frac{|X \cap Y|}{\sqrt{|X|} \cdot \sqrt{|Y|}} = \frac{a}{\sqrt{(a + b)}\sqrt{(a + c)}}$$



Set similarities/distances

$s_1 =$ *Spokesman confirms senior government advisor was shot*
$s_2 =$ *The spokesman said the senior advisor was shot dead*
$s_3 =$ *Spokesman said the shot government advisor was dead*

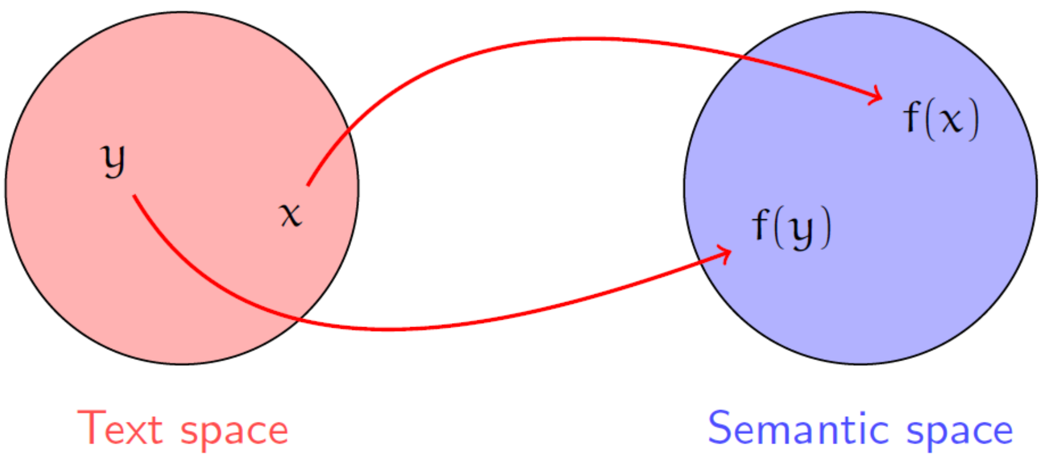
	Spokesman	confirms	said	the	senior	government	advisor	was	shot	dead
s_1	1	1	0	0	1	1	1	1	1	0
s_2	1	0	1	**1**	1	0	1	1	1	1
s_3	1	0	1	1	0	1	1	1	1	1

	$\mathrm{sim}_{\mathrm{dic}}$	$\mathrm{sim}_{\mathrm{jac}}$	$\mathrm{sim}_{\mathrm{ovl}}$	$\mathrm{sim}_{\mathrm{cos}}$
$s_1 \leftrightarrow s_2$	0.33	0.50	0.71	0.67
$s_1 \leftrightarrow s_3$	0.33	0.50	0.71	0.67
$s_2 \leftrightarrow s_3$	0.87	0.78	0.87	0.87

IV. Knowledge-based Approaches

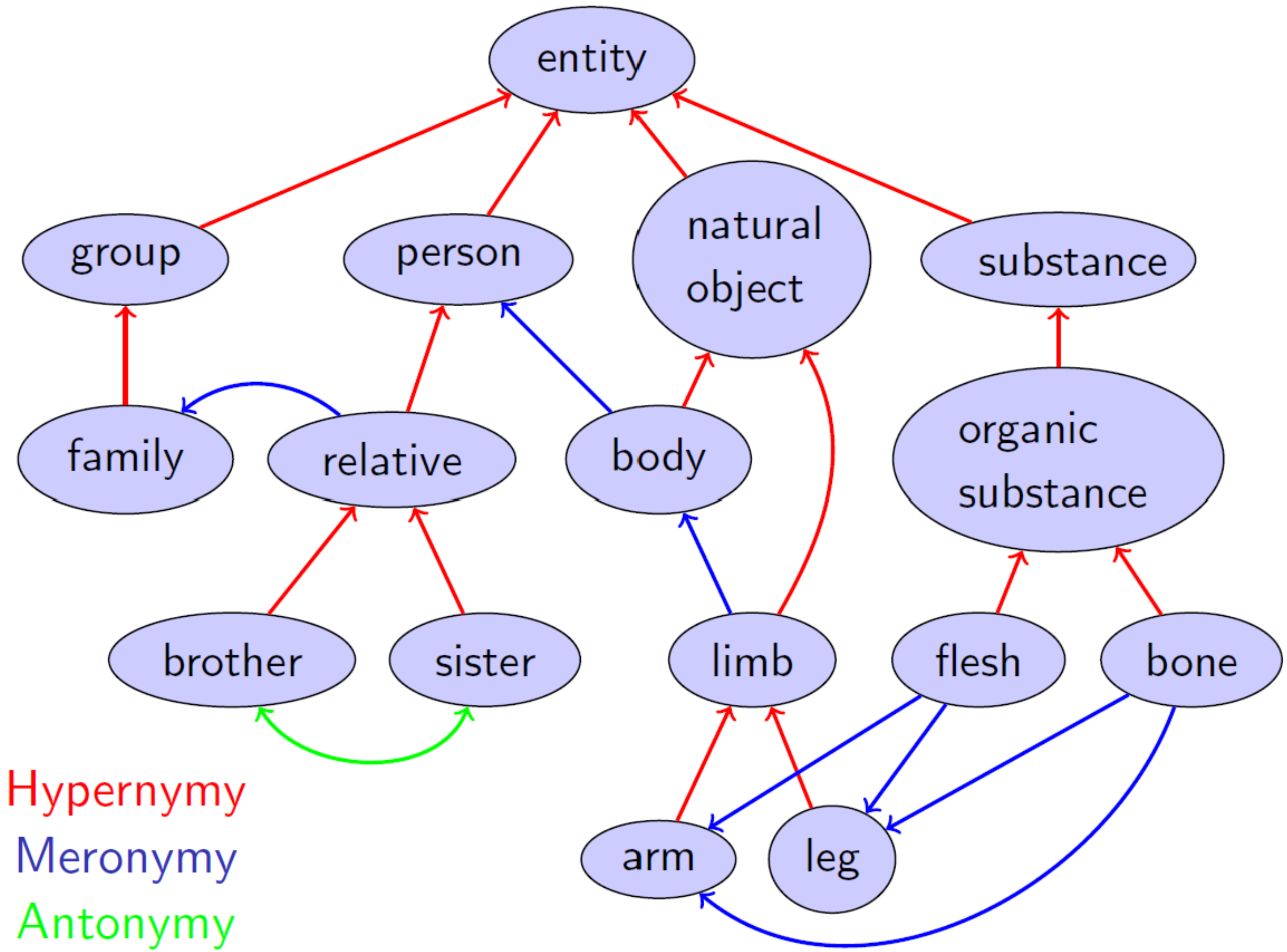
Project objects onto a knowledge-based semantic space:

$$d(x, y) = d_{sem}(f(x), f(y))$$



- Semantic spaces may be ontologies (e.g. WordNet, CYC, SUMO, ...) or graph-shaped knowledge bases (e.g. Wikipedia, DBPedia, ...).
- Projection function $f(x)$ is not trivial, since each word may map to more than one concept in semantic space.

WordNet


entity
group
person
natural object
substance
family
relative
body
organic substance
brother
sister
limb
flesh
bone
arm
leg
Hypernymy
Meronymy
Antonymy

WordNet

- What is missing in traditional dictionaries
  - It does not say, for example, that trees have roots, or that they consist of cells having cellulose walls, or even that they are living organisms
  - “Sense” of the super ordinate term aka hypernym (living plant or industrial plant)
  - Coordinate terms (bushes, shrubs, …)
  - Hyponyms - types of trees (pine, tropical, deciduous..)
  - Information assumed to be known to everyone (trees have barks and leaves, they grow from seeds, they make their own food by photosynthesis- probably information for encyclopedia!)

What is WordNet?

- WordNet is a lexical database
- WordNet 3.0 had:
  - 117,097 nouns (average noun has 1.23 senses)
  - 11,488 verbs (average verb has 2.16 sense)
  - 22,141 adjectives
  - 4,601 adverbs
- Created and maintained at Princeton University
- Accessible online @

  http://wordnetweb.princeton.edu/perl/webwn

  (Also Downloadable)
- Interfaces available in C, .Net , Java, Perl, Php, Python, Sql etc.

What is a synset?

- Basic unit of WordNet
- A group of synonymous words which refer to a common semantic concept
- Words may belong to more than one synset – first sense is the most frequent sense
- Words also include collocations (“eye contact", “mix up”)

Synset examples

- "car" in
  - {car, auto, automobile, machine, motorcar}
  - {car, railcar, railway car, railroad car}.
- "Chocolate" in

Noun

- S: (n) cocoa, **chocolate**, hot chocolate, drinking chocolate (a beverage made from cocoa powder and milk and sugar; usually drunk hot)
- S: (n) **chocolate** (a food made from roasted ground cacao beans)
- S: (n) **chocolate**, coffee, deep brown, umber, burnt umber (a medium brown to dark-brown color)

Beyond WordNet

- eXtended WordNet
- SentiWordNet
  - Each term in WordNet database is assigned a score of 0 to 1 in SentiWordNet which indicates its polarity
- WordNet for languages other than English
- FrameNet
- SentiFrameNet

Distances in WordNet

Based on graph structure:

- Shortest Path Length:

$$d(s_1, s_2) = \mathrm{SLP}(s_1, s_2)$$

- Leacock & Chodorow (similarity, $[0, \infty)$):

$$s(s_1, s_2) = -\log \frac{\mathrm{SLP}(s_1, s_2)}{2 \cdot \mathrm{MaxDepth}}$$

Based on sense information (not relations/structure)

- Gloss overlap: Any vector/set similary measure applied to words in sense glosses.

Distances in Wikipedia

- Graph-based distances (e.g Shortest Path Length, Page Rank, ...)
- Link-based similarities (some set similarity measure applied to the set of links of each page)
- Category-based similarities (some set similarity measure applied to the set of categories of each page)
- Text-based similarities (some text similarity measure applied to the texts of the pages)
- Heterogenous measures (combining several of the above in a weighted average)

V. Corpus based representations

Vectors to represent linguistic objects may be build using the distributional behaviour of the contexts they appear in.

E.g.:

- Represent words depending on the distribution of words frequently appearing nearby.
- Represent documents depending on the [general] distribution of words they contain.

Large corpus are required to pre-compute this distributions.

Corpus based representations

Vectors representing words or document contexts can be obtained in a variety of ways.

- Sparse vector representations
  - PMI
  - TF-IDF
- Dense vector representations
  - LSI
  - LDA
  - Word Embeddings

What is a corpus?

- The word *corpus* comes from Latin ("body") and the plural is *corpora*
- A corpus is a body of naturally occurring language
  - …but rarely a random collection of text
  - Corpora "are generally assembled with particular purposes in mind, and are often assembled to be (informally speaking) *representative* of some language or text type." (Leech 1992)

- "A corpus is a collection of (1) *machine-readable* (2) *authentic* texts (including transcripts of spoken data) which is (3) *sampled* to be (4) *representative* of a particular language or language variety." (MXT 2006: 5)

What is a corpus for?

- A corpus is made for the study of language in a broad sense
  - To test existing linguistic theory and hypotheses
  - To generate and verify new linguistic hypotheses
  - Beyond linguistics, to provide textual evidence in text-based humanities and social sciences subjects
- The purpose is reflected in a well-designed corpus

What corpora cannot do

- Corpora do not provide negative evidence
  - Cannot tell us what is possible or not possible
  - Can show what is central and typical in language

- Corpora can yield findings but rarely provide explanations for what is observed
  - Interfacing other methodologies

- The findings based on a particular corpus only tell us what is true in that corpus
  - Generalisation vs. representativeness

Corpus classification

- Textual vs. Speech Corpus
- Public vs. Private Corpus
- Particular vs. Reference Corpus
    - Particular:
        - literature corpus classified by year/domain/author etc.
        - Corpus with the language of children, etc.
    - Reference:
        - Very large, covers all relevant language varieties and the common vocabulary of a language.
        - Is usually hierarchically structured in sub-corpora
        - Usually built by specialized linguistic institutions

Corpus classification

- Diachronic corpus (language in its evolution)
- Monolingual vs. Multilingual Corpus
- Paralell vs. Comparable corpus

Corpus magic

- Most deep learning techniques need a corpus.
- Yet, a corpus is only useful if it **fits** the problem.

- https://www.youtube.com/watch?v=aboZctrHfK8

Thank you for your attention!

diana.trandabat@info.uaic.ro
diana.trandabat@gmail.com