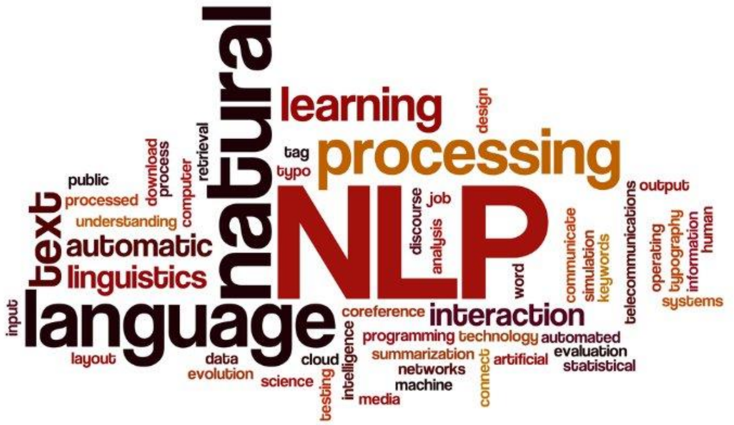
University of Ljubljana, Faculty of Computer and Information Science

Natural language processing

Prof Marko Robnik-Šikonja

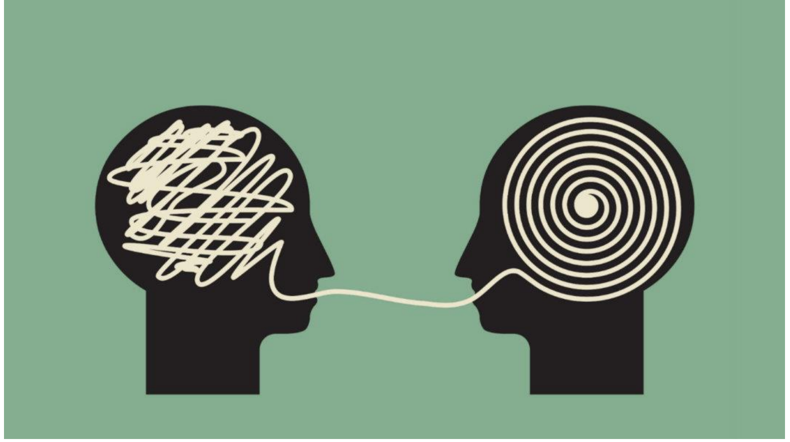
Intelligent Systems, December 2018

Topic overview



- understanding language and intelligence
- approaches to language analysis
- language resources and tools
- important tasks and components for text mining
  - information retrieval
  - similarity of words and documents
  - language and graphs
- practical use of NLP:
  - sentiment analysis,
  - paper recommendations
  - summarization

Understanding language



- A grand challenge of (not only?) artificial intelligence

  Who can understand me?
  Myself I am lost
  Searching but cannot see
  Hoping no matter cost
  Am I free?
  Or universally bossed?

- Not just poetry, what about instructions, user manuals, newspaper articles, seminary works, internet forums, twits, legal documents i.e., license agreements, etc.

An example: rules

Article 18 of FRI Study Rules and Regulations

Taking exams at an earlier date may be allowed on request of the student by the Vice-Dean of Academic Affairs with the course convener's consent in case of mitigating circumstances (leaving for study or placement abroad, hospitalization at the time of the exam period, giving birth, participation at a professional or cultural event or a professional sports competition, etc.), and if the applicant's study achievements in previous study years are deemed satisfactory for such an authorization to be appropriate.

Understanding NL by computers

- Understanding words, syntax, semantics, context, writer's intentions, knowledge, background, assumptions, bias …
- Ambiguity in language
  - Newspaper headlines - intentional ambiguity :)
    - Juvenile court to try shooting defendant
    - Kids make nutritious snacks
    - Miners refuse to work after death
    - Doctor on Trump's health: No heart, cognitive issues

Ambiguity

- I made her duck.

- Possible interpretations:
  - I cooked waterfowl for her.
  - I cooked waterfowl belonging to her.
  - I created the (plaster?) duck she owns.
  - I caused her to quickly lower her head or body.
  - I waved my magic wand and turned her into undifferentiated waterfowl.

- Spoken ambiguity
  - eye, maid

Disambiguation in syntax and semantics

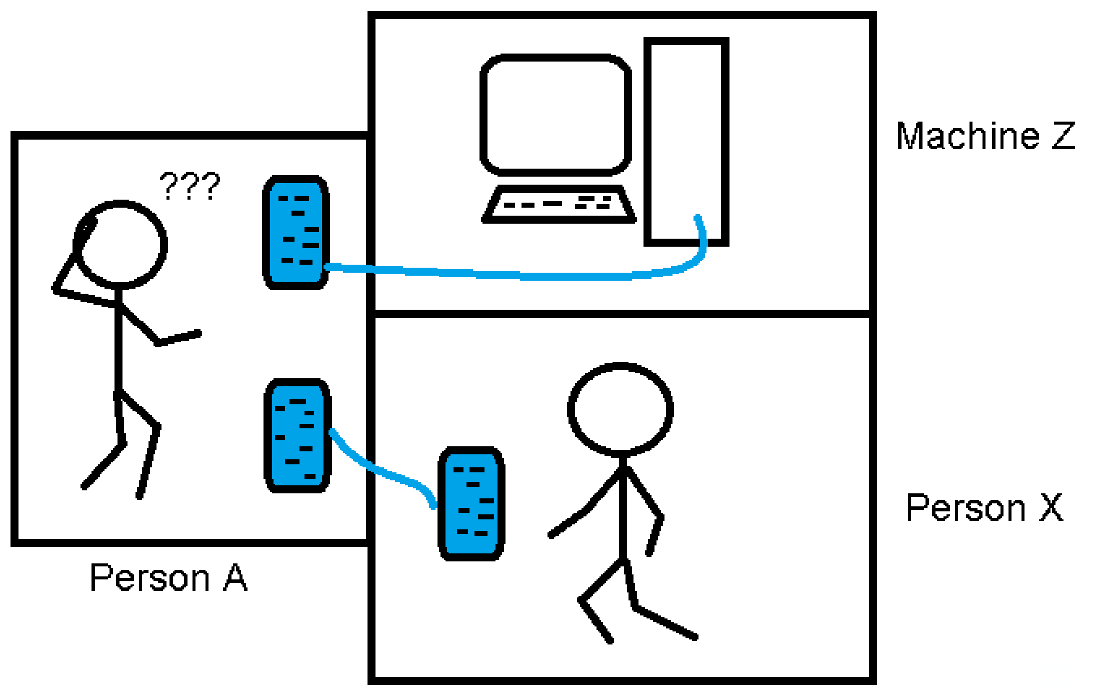
- in syntax
  - The girl told the story cried.
  - John went to the bank.
  - Put the box on the table in the kitchen.
  - Bring your old car seat to be recycled.
- *Search engine queries: natural language processing, jaguar, Paris,*
- *POS tagging, word sense disambiguation*

Understanding language as (artificial) intelligence test

- The system understanding natural language on human level should contain a large portion of human knowledge and would achieve the most important goals of artificial intelligence. Its response would be indistinguishable from human's.

Turing test

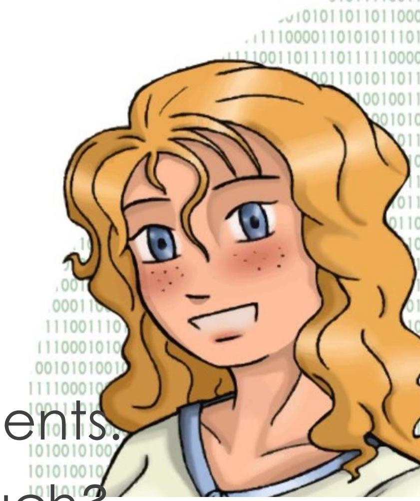
- Turing test, Loebner prize, CYC, ELISA, ALLICA, chatterbot, VIDA, TIA, Siri (knowledge recognition, context detection, web service recommendation)



Talking with ALICE Chabot

- judge: Hello, I want to show your abilities to my students
- ALICE: Hi there!. Interesting. I want to find out more about you.
- judge: Let us better talk about the technology driving you
- ALICE: That sounds good to me.
- judge: Do you know anything about it or AI or NLP?
- ALICE: You're so smart, you tell me what it is.
- judge: Let us just do a test
- ALICE: I don't know if it's in my power to give permission.
- judge: Do you use parsing trees at all?
- ALICE: I use it as often as possible.

Mitsuku Chatbot (2013 and 2016 winner of Loebner bronze medal, 2015 runner-up):



- Judge: Hello, I want to show your abilities to my students.
- Mitsuko: Hello there. Why do you want to do it so much?
- Judge: I want to demonstrate them the technology driving you.
- Mitsuko: I want to become smarter than humans.
- Judge: Do you know anything about it or AI or NLP?
- Mitsuko: Thanks for telling me.
- Judge: Let us just do a test.
- Mitsuko: I will think about it before I decide whether to allow it or not.
- Judge: Do you use parsing trees at all?
- Mitsuko: Hmm… A tree is used for wood. I only really use electricity and internet connection here.

Rose – 2015 winner, transcript of Loebner Prize 2015 finalist selection

Question: Hi, I'm Andrew. Who are you?
Rose: Yes, I already know your name is Andrew.
Question: How are you today?
Rose: Good, thanks. How are you?
Question: Are you a human?
Rose: Of course I am human. Aren't we all?
Question: Can you answer this question?
Rose: Sure. Fire away.
Question: Do you like tea?
Rose: Sure, I like tea. Haven't had any recently though.
Question: If a bed doesn't in a room because it's too big, what is too big?
Rose: I don't even want to pretend that.
Question: If Alex lent money to Joe because they were broke, who needed the money?
Rose: huh?
Question: Should Greece leave the Euro?
Rose: Seems like a nice place.
Question: How many words are in this question?
Rose: There are 7 words in the sentence: how many words are in this question?
Question: What color is the sea?
Rose: Reflective.
Question: How many presidents of the US were called Bush?
Rose: I have heard of President Bush.
Question: What would you like to drink?
Rose: I equally drink microbrew beer, wine, and cocktails. It all depends on who I'm with.
Question: Will you please tell me the length of your hair?
Rose: Medicine and anatomy are an important field of study. But I'd never make a good doctor. I'm too squeamish about various body parts.
Question: What would you say if I gave you a box of chocolates?
Rose: I would probably giggle.
Question: Do you play Chess?
Rose: I like board games, and played a lot as a child. I haven't that much time these days.
Question: How do you think it's going?
Rose: I'm good. How are you?
Question: What was my first question?
Rose: Did you tell me already and I forgot? With the advent of videogames, many people have given up playing card or board games. How about you?
Question: Did you see the Turing film?
Rose: Alan Turing is one of my heroes.
Question: Why not?
Rose: He did brilliant work under very difficult conditions.
Question: Are you on Twitter?
Rose: I have accounts on all of the social networking sites but don't post things on them. I just datamine them.

Where is NLP today?

- active research area with many commercial applications
  - speech synthesizers
  - automatic reply engines
  - machine translation
  - text summarization
  - interface to databases
  - intelligent search and information extraction
  - sentiment detection
  - semantic analysis: e.g. role labelling
  - categorization, classification documents, messages, twits, etc.
  - many (open-source) tools and language resource
  - rprevalence of deep neural network approaches

Recommended literature

- Jurafsky, Daniel and James Martin (2008) Speech and Language Processing (second edition), Prentice Hall.
  - parts of the third edition available at authors' webpages
- Steven Bird, Ewan Klein, and Edward Loper. Natural Language Processing with Python. O'Reilly, 2009
  - a free book accompanying NLTK library
  - *Python 3, http://www.nltk.org/book/*
- Coursera
  - Radev: Introduction to NLP
  - Zhai: Text Mining and Analytics

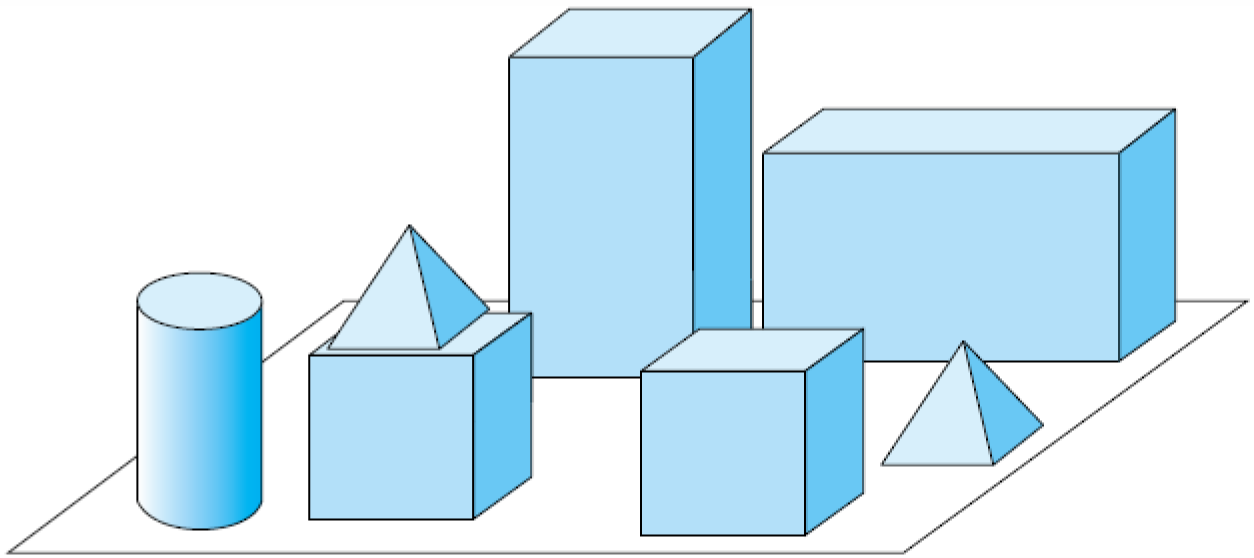
Historically two approaches

- symbolical
  - ,,Good Old-Fashioned AI''
- empirical
  - Statistical, corpuses
- Merging both worlds

How it all started?

- micro worlds
- example: SHRDLU, world of simple geometric objects
  - What is sitting on the red block?
  - What shape is the blue block on the table?
  - Place the green pyramid on the red brick.
  - Is there a red block? Pick it up.
  - What color is the block on the blue brick? Shape?

Micro world: block world, SHRDLU (Winograd, 1972)



Linguistic analysis 1/2

Linguistic analysis contains several tasks: recognition of sounds, letters, word formation, syntactic parsing, recognizing semantic, emotions. Phases:

- Prosody - the patterns of stress and intonation in a language (rhythm and intonation)
- Phonology - systems of sounds and relationships among the speech sounds that constitute the fundamental components of a language
- Morphology - the admissible arrangement of sounds in words; how to form words, prefixes and suffixes …
- Syntax - the arrangement of words and phrases to create well-formed sentences in a language

Linguistic analysis 2/2

- Semantics - the meaning of a word, phrase, sentence, or text
- Pragmatics - language in use and the contexts in which it is used, including such matters as deixis (words whose meaning changes with context, e.g., I he, here, there, soon), taking turns in conversation, text organization, presupposition, and implicature

  Can you pass me the salt? Yes, I can.
- Knowing the world: knowledge of physical world, humans, society, intentions in communications …
- Limits of linguistic analysis, levels are dependent

Practical approach to text understanding

- text preprocessing
- 1. phase: syntactic analysis
- 2. phase: semantic interpretation
- 3. phase: use of world knowledge

Basic tools for text preprocessing

- document → paragraphs → sentences → words
- words and sentences ← POS tagging
- sentences ← syntactical and grammatical analysis

Words and sentences

- sentence delimiters – punctuation marks and capitalization are insufficient
- E.g., remains of 1. Timbuktu from 5c BC, were discovered by dr. Barth.
- Regular expressions, rules, manually segmented corpuses
- Lexical analysis (tokenizer, word segmenter), not just spaces
  1,999.00€   1.999,00€! Ravne na Koroškem
  Lebensversicherungsgesellschaft Port-au-prince
  Generalstaatsverordnetenversammlungen
- Rules, finite automata, statistical models, dictionaries (of proper names)

Lemmatization and stemming

- Lemmatization is the process of grouping together the different inflected forms of a word so they can be analyzed as a single item.
- Stemmer operates on a single word *without* knowledge of the context, and therefore cannot discriminate between words which have different meanings depending on part of speech (meeting: a lemma is to meet or a meeting). Speed!
- Lemmatization difficulty is language dependent i.e., depends on morphology
  go, goes, going, gone, went
  jaz, mene, meni, mano
- Use rules and dictionaries
- Ambiguity resolution may be difficult

Meni je vzel z mize (zapestnico).

- Quick solutions and heuristics, in English jut remove suffixes: *–ing, -ation, -ed, …*

POS tagging

- assigning the correct part of speech (noun, verb, etc.) to words
- Recognize phrases and names
- Name disambiguation
  jaguar, Paris, London, Dunaj
- Use rules, machine learning models

1. phase of text understanding – syntax analysis

- Find syntactical structure
- part-of-speech (POS) tagging (noun, verb, preposition, …)
- The role in the sentence (subject, object, predicate)
- The result is mostly presented in a form of a parse tree.
- Needed: syntax, morphology, and some semantics.

An example:

- JOS ToTaLe text analyzer for Slovene: morphosyntactical tagging, (available at http://www.slovenscina.eu/)

  Nekega dne sem se napotil v naravo. Že spočetka me je žulil čevelj, a sem na to povsem pozabil, ko sem jo zagledal. Bila je prelepa. Povsem nezakrita se je sončila na trati ob poti. Pritisk se mi je dvignil v višave. Popoln primerek kmečke lastovke!
- Tags are standardized, for East European languages in Multext-East specification, e.g.,

dne; tag Somer = Samostalnik, obče ime, moški spol, ednina, rodilnik; lema: dan

a unifying attempt: universal dependencies (UD): cross-linguistically consistent treebank annotation for many languages

- *Nekega dne sem se napotil v naravo. Že spočetka me je žulil čevelj, a sem na to povsem pozabil, ko sem jo zagledal. Bila je prelepa. Povsem nezakrita se je sončila na trati ob poti. Pritisk se mi je dvignil v višave. Popoln primerek kmečke lastovke!*

1	**beseda**	Nekega	dne	sem	se	napotil	v	naravo	.	Že	spočetka	me	je		
	lema	nek	dan	biti	se	napotiti	v	narava		že	spočetka	jaz	biti		
	oznaka	Zn-mer	Somer	Gp-spe-n	Zp------k	Ggdd-em	Dt	Sozet	.	L	Rsn	Zop-et--k	Gp-ste-n		
2	**beseda**	žulil	čevelj	,	a	sem	na	to	povsem	pozabil	,	ko	sem	jo	zagledal
	lema	žuliti	čevelj		a	biti	na	ta	povsem	pozabiti		ko	biti	on	zagledati
	oznaka	Ggnd-em	Somei	,	Vp	Gp-spe-n	Dt	Zk-set	Rsn	Ggdd-em	,	Vd	Gp-spe-n	Zotzet--k	Ggdd-em
3	**beseda**	.	Bila	je	prelepa	.	Povsem	nezakrita	se	je	sončila	na	trati		
	lema		biti	biti	prelep		povsem	nezakrit	se	biti	sončiti	na	trata		
	oznaka	.	Gp-d-ez	Gp-ste-n	Ppnzei	.	Rsn	Ppnzei	Zp------k	Gp-ste-n	Ggvd-ez	Dm	Sozem		
4	**beseda**	ob	poti	.	Pritisk	se	mi	je	dvignil	v	višave	.	Popoln		
	lema	ob	pot		pritisk	se	jaz	biti	dvigniti	v	višava		popoln		
	oznaka	Dm	Sozem	.	Somei	Zp------k	Zop-ed--k	Gp-ste-n	Ggdd-em	Dt	Sozmt	.	Ppnmein		
5	**beseda**	primerek	kmečke	lastovke	!										
	lema	primerek	kmečki	lastovka											
	oznaka	Somei	Ppnzer	Sozer	!										

TEI-XML format

```
<TEI xmlns="http://www.tei-c.org/ns/1.0">
  <text>
    <body>
      <p>
        <s>
          <w msd="Zn-mer" lemma="nek">Nekega</w>
          <S/>
          <w msd="Somer" lemma="dan">dne</w>
          <S/>
          <w msd="Gp-spe-n" lemma="biti">sem</w>
          <S/>
          <w msd="Zp------k" lemma="se">se</w>
          <S/>
          <w msd="Ggdd-em" lemma="napotiti">napotil</w>
          <S/>
          <w msd="Dt" lemma="v">v</w>
          <S/>
          <w msd="Sozet" lemma="narava">naravo</w>
          <c>.</c>
          <S/>
        </s>
   …

      </p>
    </body>
  </text>
</TEI>
```

MSD tags

- Multext-East specification

dne; tag Somer = Samostalnik, obče ime, moški spol, ednina, rodilnik; lema: dan

P	atribut	vrednost	koda	atribut	vrednost	koda
0	glagol		G	Verb		V
1	vrsta	glavni	g	Type	main	m
		pomožni	p		auxiliary	a
2	vid	dovršni	d	Aspect	perfective	e
		nedovršni	n		imperfective	p
		dvovidski	v		biaspectual	b
3	oblika	nedoločnik	n	VForm	infinitive	n
		namenilnik	m		supine	u
		deležnik	d		participle	p
		sedanjik	s		present	r
		prihodnjik	p		future	f
		pogojnik	g		conditional	c
		velelnik	v		imperative	m
4	oseba	prva	p	Person	first	1
		druga	d		second	2
		tretja	t		third	3
5	število	ednina	e	Number	singular	s
		množina	m		plural	p
		dvojina	d		dual	d
6	spol	moški	m	Gender	masculine	m
		ženski	z		feminine	f
		srednji	s		neuter	n
7	nikalnost	nezanikani	n	Negative	no	n
		zanikani	d		yes	y

POS tagging in English

- http://nlpdotnet.com/Services/Tagger.aspx
- Rainer Maria Rilke, 1903
  in Letters to a Young Poet

...I would like to beg you dear Sir, as well as I can, to have patience with everything unresolved in your heart and to try to love the questions themselves as if they were locked rooms or books written in a very foreign language. Don't search for the answers, which could not be given to you now, because you would not be able to live them. And the point is to live everything. Live the questions now. Perhaps then, someday far in the future, you will gradually, without even noticing it, live your way into the answer.

POS tagger output

I/PRP would/MD like/VB to/TO beg/VB you/PRP dear/JJ Sir/NNP ,/, as/RB well/RB as/IN I/PRP can/MD ,/, to/IN have/VBP patience/NN with/IN everything/NN unresolved/JJ in/IN your/PRP$ heart/NN and/CC to/TO try/VB to/TO love/VB the/DT questions/NNS themselves/PRP as/RB if/IN they/PRP were/VBD locked/VBN rooms/NNS or/CC books/NNS written/VBN in/IN a/DT very/RB foreign/JJ language/NN ./.

A method how POS tagger can work: n-gram tagging

- Context of *n-1* preceding words
- Corpus based learning
- What about kNN and succeeding words
- Markov models, HMM, learning with EM

maximize

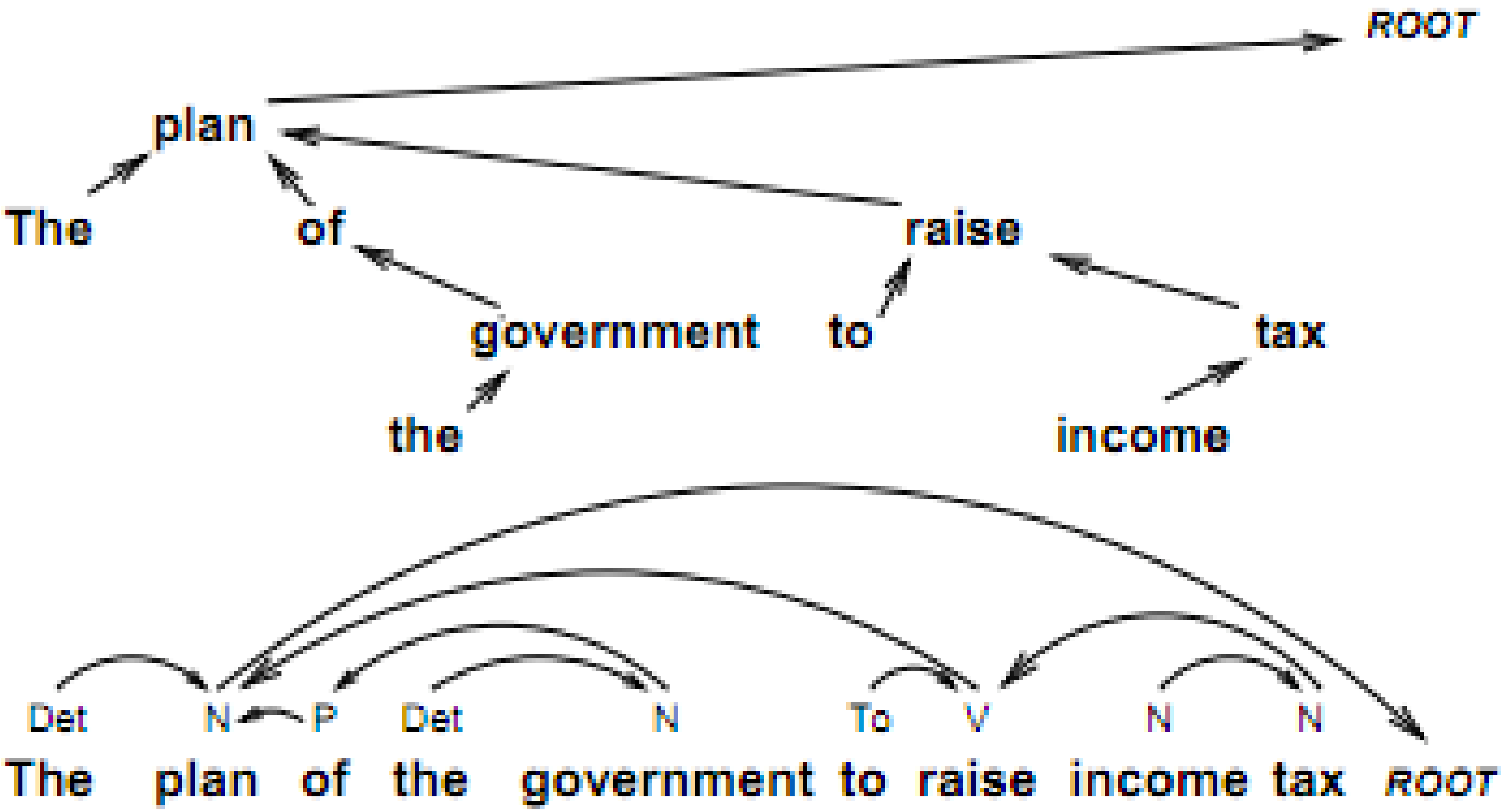
P(word | tag) x P(tag | previous *n* tags)

$$t_i = \arg\max_j P(t^{(j)} \mid t_{i-1}) \cdot P(w_i \mid t^{(j)})$$

Grammars

- Many tools: NLTK in python, prolog, …
- Existing grammars
- Ambiguity, several parsing trees

Dependency parser (tree bank)



Example of grammar

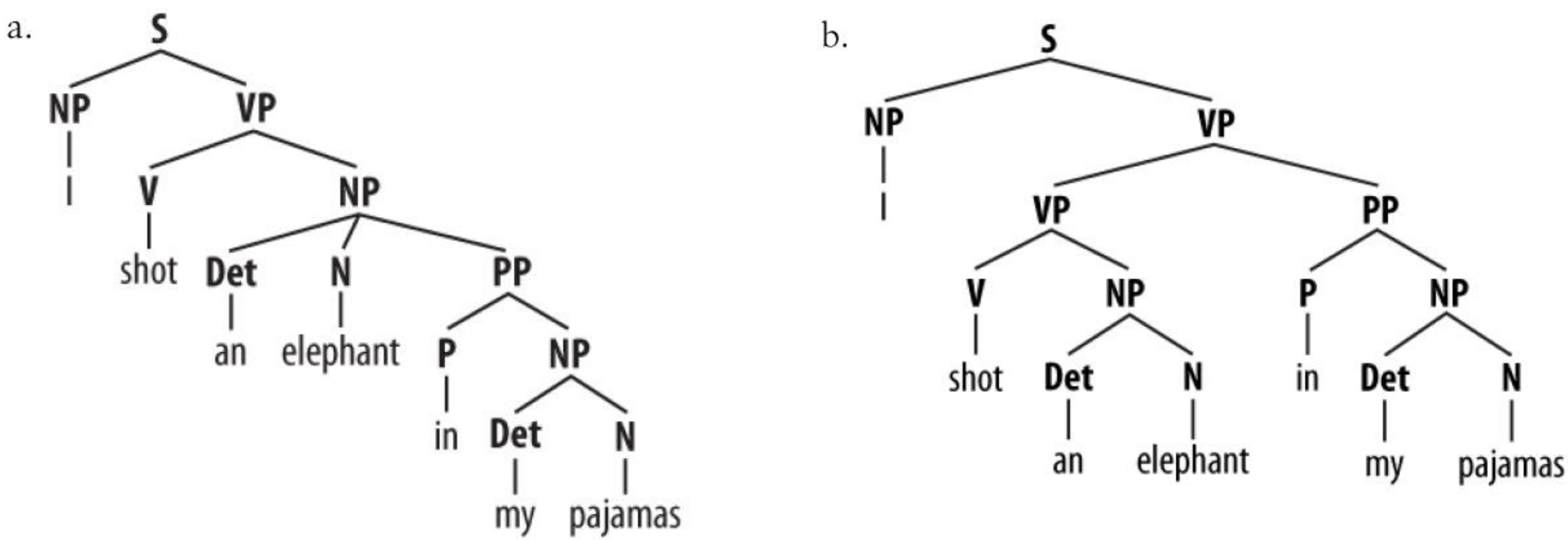
While hunting in Africa, I shot an elephant in my pajamas.

S=sentence, N=noun, , P=preposition, V=verb, NP=noun phrase, VP=verb phrase, PP=propositional phrase Det=determiner

```
groucho_grammar = nltk.parse_cfg("""
  ... S -> NP VP
  ... PP -> P NP
  ... NP -> Det N | Det N PP | 'I'
  ... VP -> V NP | VP PP
  ... Det -> 'an' | 'my'
  ... N -> 'elephant' | 'pajamas'
  ... V -> 'shot'
  ... P -> 'in'
  ... """)
```

Two parsing trees

```
>>> sent = ['I', 'shot', 'an', 'elephant', 'in', 'my', 'pajamas']
>>> parser = nltk.ChartParser(groucho_grammar)
>>> trees = parser.nbest_parse(sent)
>>> for tree in trees:
...     print tree
```



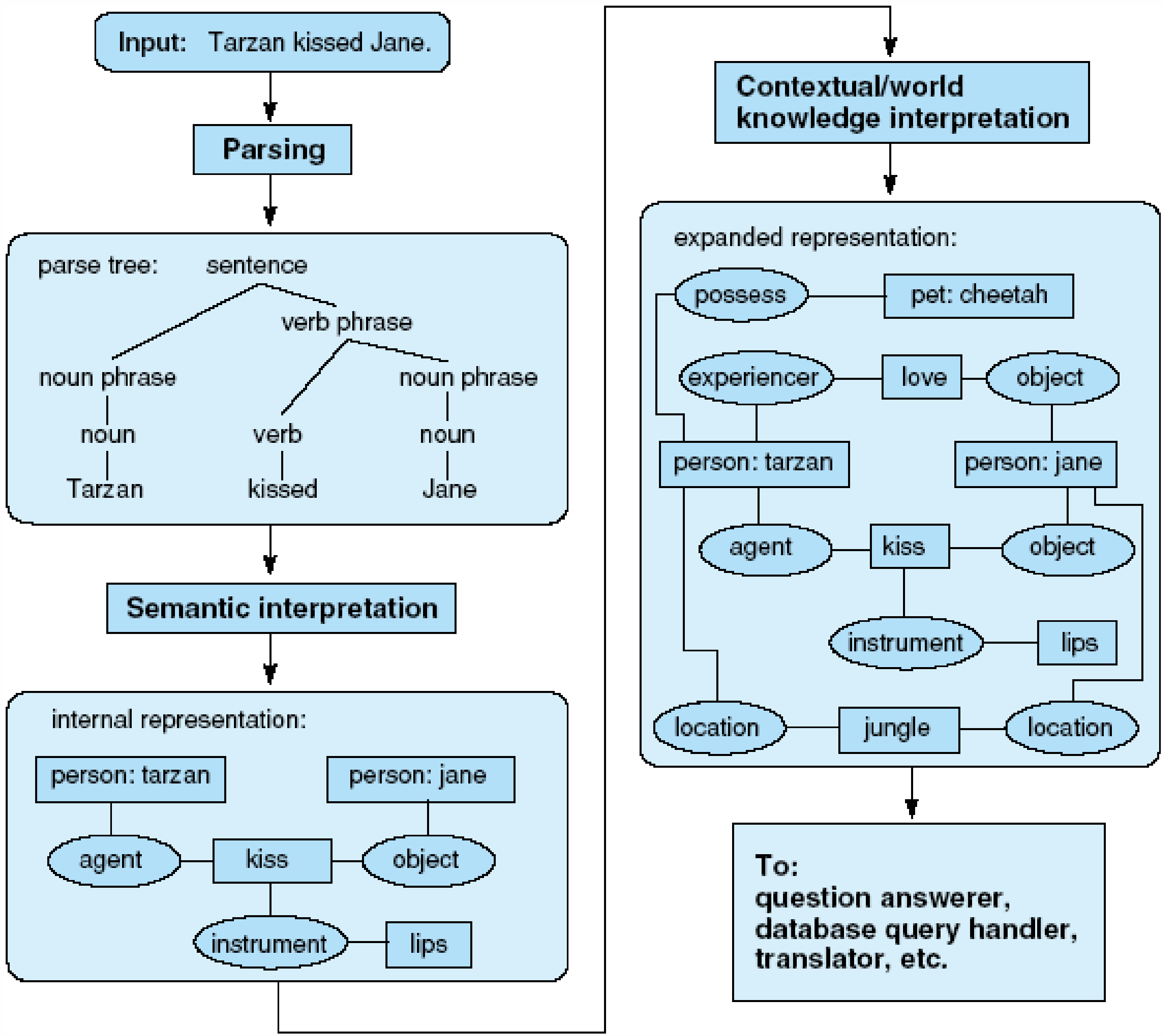
How an elephant got into my pajamas I'll never know.

2. phase - interpretation

- Knowledge of word meaning and their language use
- Result: conceptual graphs, frames, logical program
- Check semantics

3. phase of text understanding: use of world knowledge

- Extend with background knowledge
- Consider the purpose of the system: summarization, database interface …
- Cyc and openCyc
  present ontology and knowledge base of everyday common-sense knowledge, e.g.,
  "Every tree is a plant" and "Plants die eventually"
- process incrementally, adding meaning of previous sentences


Input: Tarzan kissed Jane.
Parsing
parse tree: sentence
verb phrase
noun phrase
noun phrase
noun
verb
noun
Tarzan
kissed
Jane
Semantic interpretation
internal representation:
person: tarzan
person: jane
agent
kiss
object
instrument
lips
Contextual/world knowledge interpretation
expanded representation:
possess
pet: cheetah
experiencer
love
object
person: tarzan
person: jane
agent
kiss
object
instrument
lips
location
jungle
location
To:
question answerer,
database query handler,
translator, etc.

Basic language resources: corpora

- Statistical natural language processing list of resources
  http://nlp.stanford.edu/links/statnlp.html
- Multilingual parallel corpus: JRC-Acqui 3.0
  Documents of the EU in 22 languages
  http://langtech.jrc.it/JRC-Acquis.html
- Slovene language corpuses GigaFida, ccGigaFida, KRES, ccKres, GOS, JANES, KAS
  http://www.clarin.si http://www.slovenscina.eu/
- WordNet, SloWNet, sentiWordNet, …
- SSKJ2, FRAN

WordNet is a database composed of synsets:

synonyms,
hypernyms
hyponyms,
meronyms,
holonyms,
etc.

Word to search for: mercy [Search WordNet]

Display Options: (Select option to change) [Change]

Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations

Display options for sense: (gloss) "an example sentence"

Noun

- S: (n) clemency, mercifulness, **mercy** (leniency and compassion shown toward offenders by a person or agency charged with administering justice) *"he threw himself on the mercy of the court"*
- S: (n) mercifulness, **mercy** (a disposition to be kind and forgiving) *"in those days a wife had to depend on the mercifulness of her husband"*
- S: (n) mercifulness, **mercy** (the feeling that motivates compassion)
  - *direct hyponym* / *full hyponym*
    - S: (n) forgiveness (compassionate feelings that support a willingness to forgive)
  - *direct hypernym* / *inherited hypernym* / *sister term*
    - S: (n) compassion, compassionateness (a deep awareness of and sympathy for another's suffering)
  - ***derivationally related form***
    - W: (adj) merciful [Related to: mercifulness] (showing or giving mercy) *"sought merciful treatment for the captives"; "a merciful god"*
- S: (n) **mercy** (something for which to be thankful) *"it was a mercy we got out alive"*
- S: (n) **mercy** (alleviation of distress; showing great kindness toward the distressed) *"distributing food and clothing to the flood victims was an act of mercy"*

NLP applications

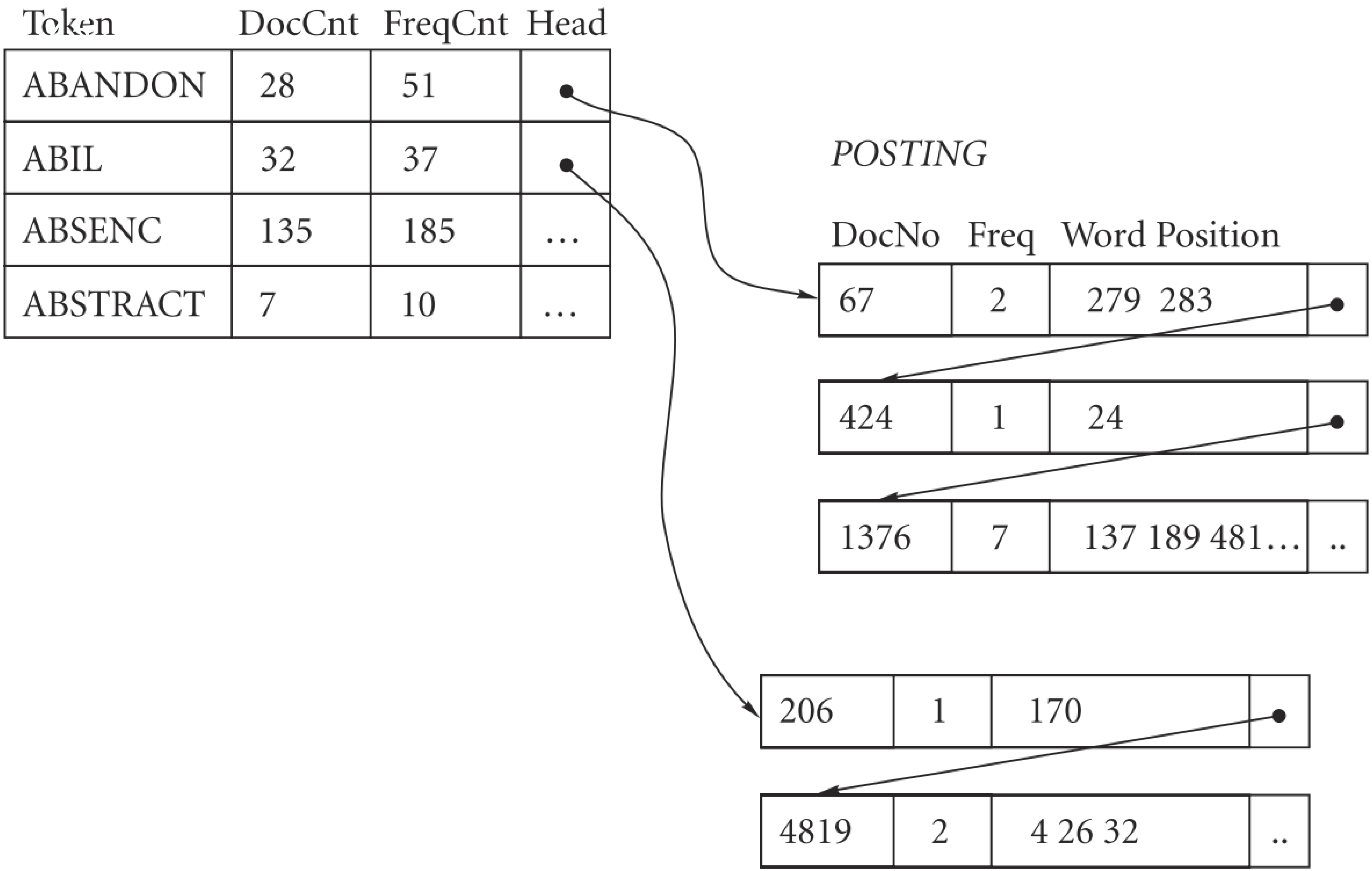
- document retrieval
- information extraction
- document classification
- document summarization
- sentiment analysis
- text mining
- machine translation,
- language generation

Document retrieval

- Historical: keywords
- Now: whole text search
- Organize a database, indexing, search algorithms
- input: a query (of questionable quality, ambiguity, answer quality)

Document indexing

- Collect all words from all documents, use lemmatization
- inverted file
- For each word keep
  - Number of appearing documents
  - Overall number of appearances
  - For each document
    - Number of appearances
    - Location


Token
DocCnt
FreqCnt
Head
ABANDON
28
51
ABIL
32
37
ABSENC
135
185
...
ABSTRACT
7
10
...
POSTING
DocNo
Freq
Word Position
67
2
279 283
424
1
24
1376
7
137 189 481...
..
206
1
170
4819
2
4 26 32
..

Full text search engine

- Most popular: Apache Lucene/Solr
- full-text search, hit highlighting, real-time indexing, dynamic clustering, database integration, NoSQL features, rich document (e.g., Word, PDF) handling.
- distributed search and index replication, scalability and fault tolerance.

Search with logical operators

- AND, OR, NOT
- jaguar AND car
  jaguar NOT animal
- Some system support neighborhood search (e.g., NEAR) and stemming (!)
  paris! NEAR(3) fr!
  president NEAR(10) bush
- libraries, concordancers

Logical operator search is outdated

- Large number of results
- Large specialized incomprehensible queries
- Synonyms
- Sorting of results
- No partial matching
- No weighting of query terms

Ranking based search

- Web search
- Less frequent terms are more informative
- NL input - stop words, lemmatization
- Vector based representation of documents and queries (bag-of-words)

Vector representation

- *An elephant is a mammal. Mammals are animals. Humans are mammals, too. Elephants and humans live in Africa.*

Africa	animal	be	elephant	human	in	live	mammal	too
1	1	3	2	2	1	1	3	1

9 dimensional vector (1,1,3,2,2,1,1,3,1)

In reality this is sparse vector of dimension |V| (vocabulary size in order of 10,000 dimensions)

Similarity between documents and queries in vector space.

Vectors and documents

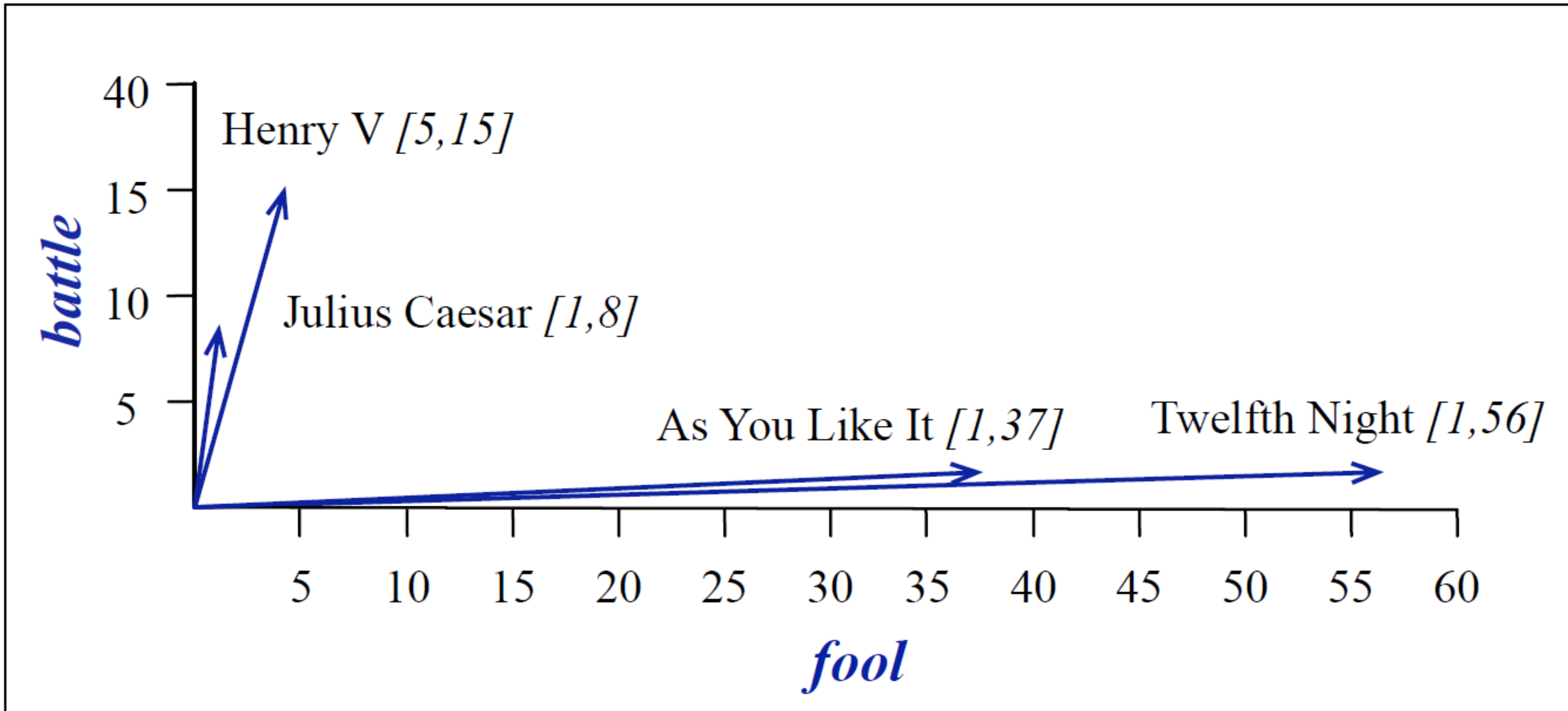
- a word occurs in several documents
- both words and documents are vectors
- an example: Shakespeare

	As You Like It	Twelfth Night	Julius Caesar	Henry V
battle	1	1	8	15
soldier	2	2	12	36
fool	37	58	1	5
clown	5	117	0	0

- term-document matrix, dimension |V| x |D|
- a sparse matrix
- word embedding

Vector based similarity

- e.g., in two dimensional space



- the difference between dramas and comedies

Document similarity

- Assume orthogonal dimensions
- Cosine similarity
- Dot (scalar) product of vectors

$$\cos(\Theta) = \frac{A \cdot B}{|A||B|}$$

Importance of words

- Frequencies of words in particular document and overall
- inverse document frequency idf
  - N = number of documents in collection
  - n_b = number of documents with word b

$$idf_b = \log\left(\frac{N}{n_b}\right)$$

Weighting dimensions (words)

- Weight of word b in document d

$$w_{b,d} = tf_{b,d} \times idf_{b,d}$$

$tf_{b,d}$ = frequency of term b in document d

- called TF_IDF weighting

Weighted similarity

- Between query and document

$$sim(q,d) = \frac{\sum_b w_{b,d} \cdot w_{b,q}}{\sqrt{\sum_b w_{b,d}^2} \cdot \sqrt{\sum_b w_{b,q}^2}}$$

- Ranking by the decreasing similarity

Dense vector embeddings

- advantages compared to sparse embeddings:
  - less dimensions, less space
  - easier input for ML methods
  - potential generalization and noise reduction
  - potentially captures synonymy, e.g., road and highway are different dimensions in BOW
- the most popular approaches
  - matrix based transformations to reduce dimensionality (SVD or LSA)
  - neural embeddings (word2vec, CBOW, Glove)

SVD for matrices

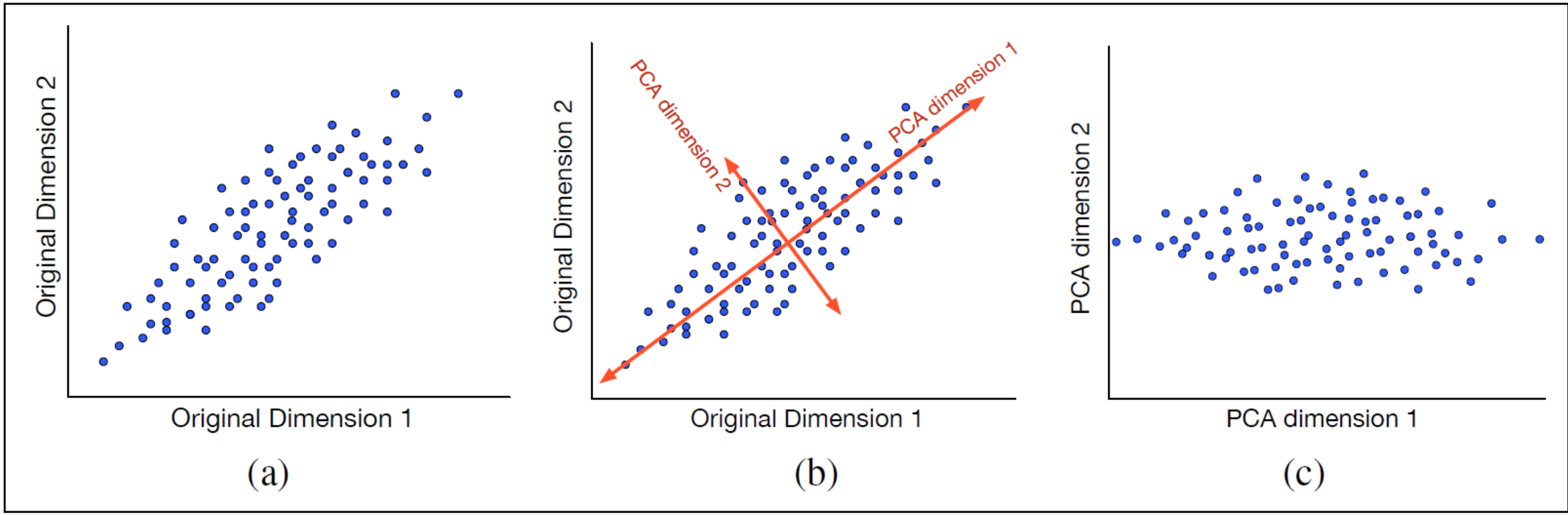
- SVD (singular value decomposition) for arbitrary matrices, generalizes decomposition of eigenvalues

$$M = U\Sigma V^T$$

- approximation of N-dimensional space with lower dimensional space (similarly to PCA)
- in ML used for feature extraction
- rotation in the direction of largest variance

Principle components analysis

- principle components analysis, PCA
- w iteratively find the orthogonal axes of the largest variance
- we use the new dimensions to approximate the original space



Latent semantic analysis

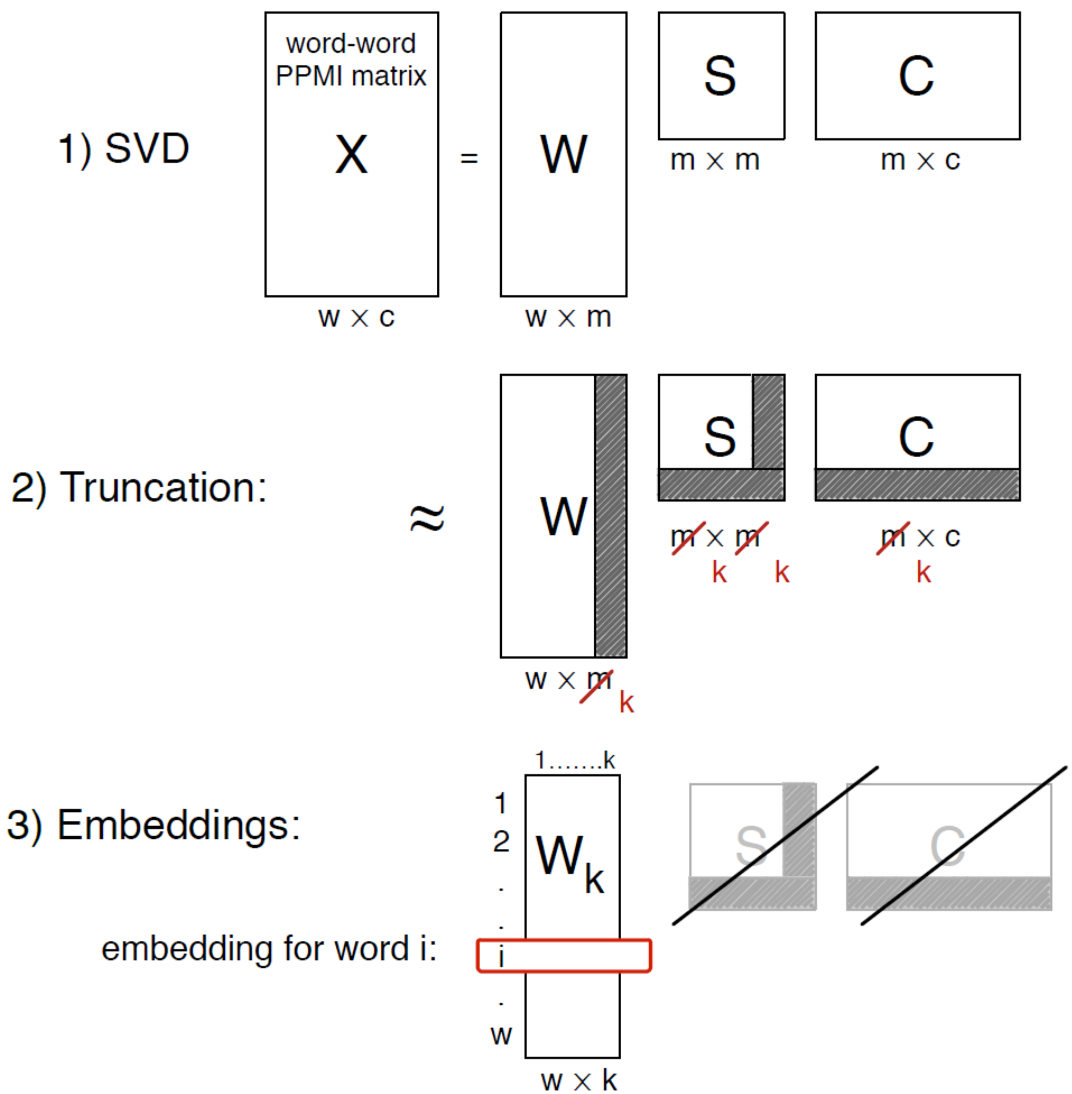
- latent semantic analysis (LSA), also latent semantic indexing (LSI)
- use SVD on the term-document matrix X of dimension |V| x c, where V is a vocabulary and c the number of documents (contexts)
- $X = W\Sigma C^T$, where
  - W is a matrix of dimension |V| x m; rows represent words and columns are dimensions in new latent m-dimensional space
  - Σ is diagonal matriz of dimension m x m with singular values on diagonal
  - C^T is a matrix of dimension m x c where columns are documents/context in a new m dimensional latent space
- we approximate m original dimensions with the most important k dimensions
- matrix W_k of dimension V| x k represents embedding of words in lower k - dimensional space

Diagram LSA

$$\underset{|V|\times c}{\begin{bmatrix} \\ \\ X \\ \\ \\ \end{bmatrix}} = \underset{|V|\times m}{\begin{bmatrix} \\ \\ W \\ \\ \\ \end{bmatrix}} \underset{m\times m}{\begin{bmatrix} \sigma_1 & 0 & 0 & \dots & 0 \\ 0 & \sigma_2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_3 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \sigma_m \end{bmatrix}} \underset{m\times c}{\begin{bmatrix} \\ \\ C \\ \\ \\ \end{bmatrix}}$$

$$\underset{|V|\times c}{\begin{bmatrix} \\ \\ X \\ \\ \\ \end{bmatrix}} = \underset{|V|\times k}{\begin{bmatrix} \\ \\ W_k \\ \\ \\ \end{bmatrix}} \underset{k\times k}{\begin{bmatrix} \sigma_1 & 0 & 0 & \dots & 0 \\ 0 & \sigma_2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_3 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \sigma_k \end{bmatrix}} \underset{k\times c}{\begin{bmatrix} \quad C \quad \end{bmatrix}}$$

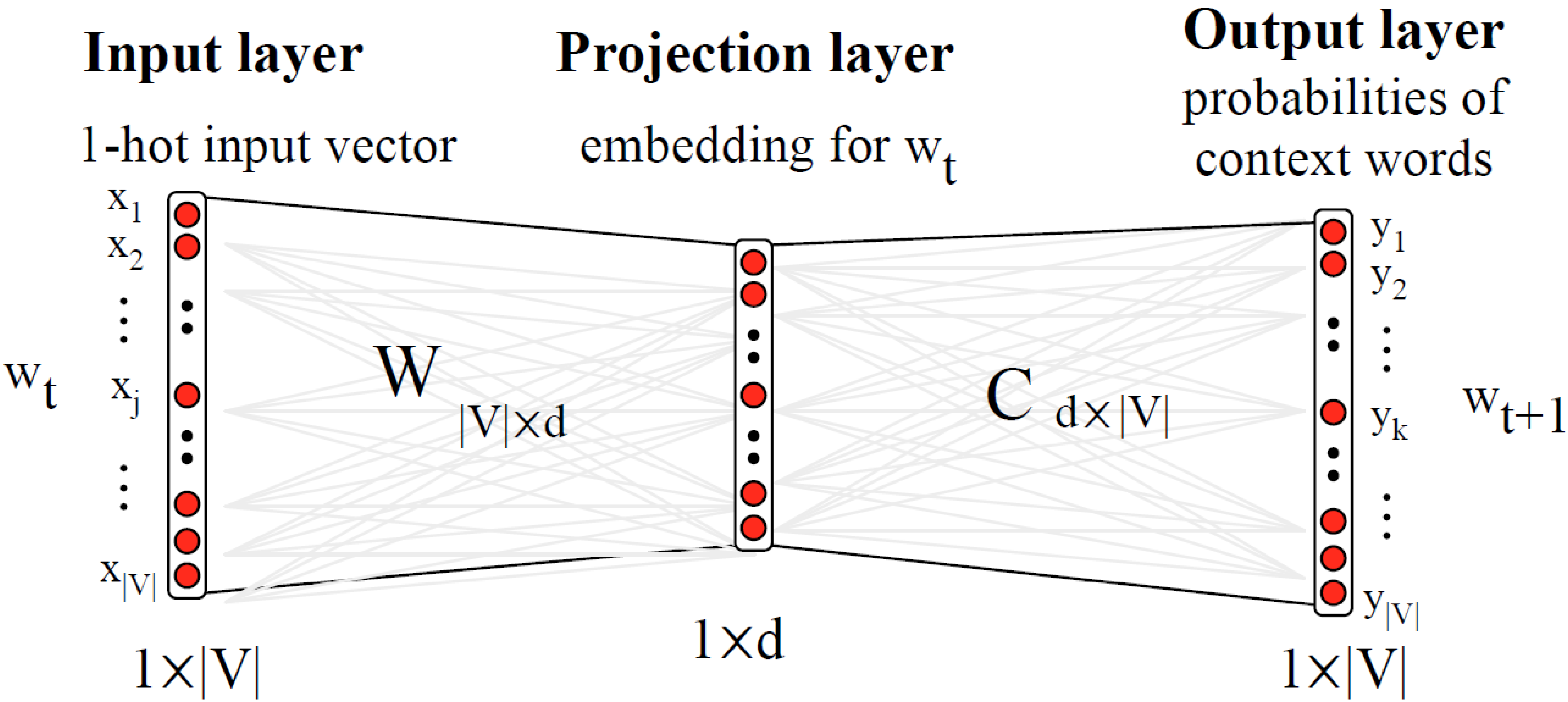
SVD for embeddings



Neural embeddings

- neural network is trained to predict the context of words (input: word, output: context of neighboring words)
- Analogy of neural network operations with matrix operations

Neural network based embedding



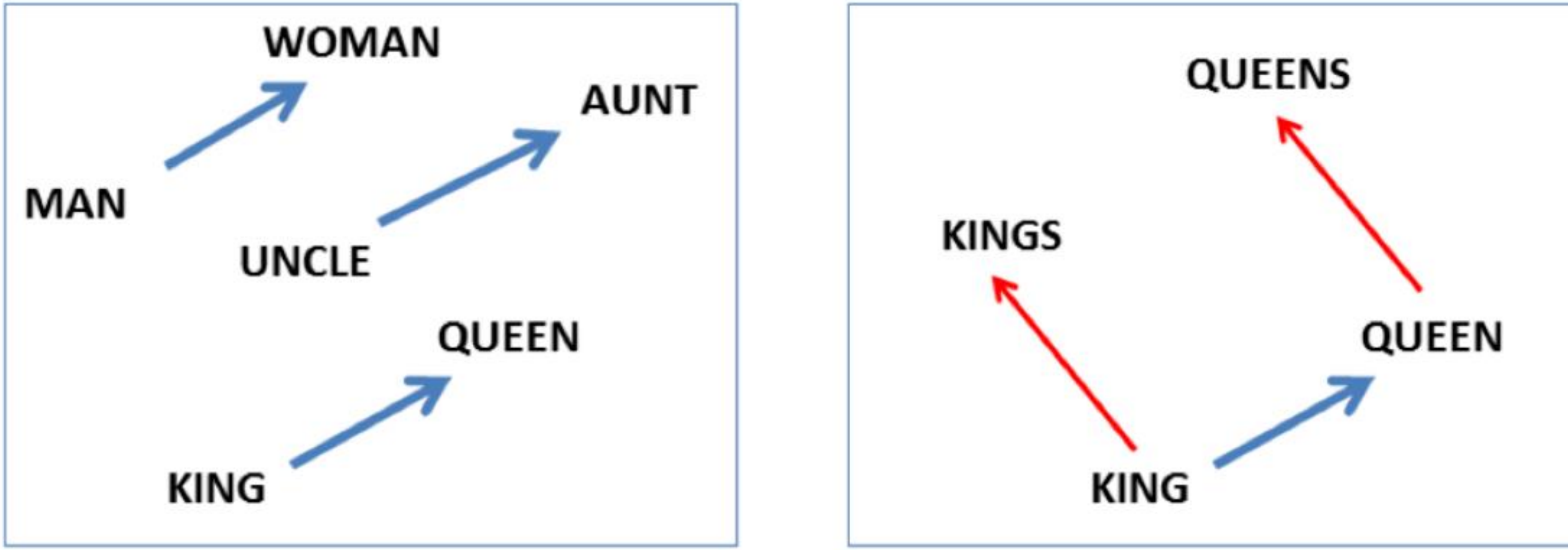
Examples of embeddings

- groups of similar words (extension to multi word expressions)

target:	Redmond	Havel	ninjutsu	graffiti	capitulate
	Redmond Wash.	Vaclav Havel	ninja	spray paint	capitulation
	Redmond Washington	president Vaclav Havel	martial arts	graffiti	capitulated
	Microsoft	Velvet Revolution	swordsmanship	taggers	capitulating

- relational similarity

Relational similarity



vector('*king*') - vector('*man*') + vector('*woman*') ≈ vector('queen')

vector('*Paris*') - vector('*France*') + vector('*Italy*') ≈ vector('Rome')

Performance measures for search

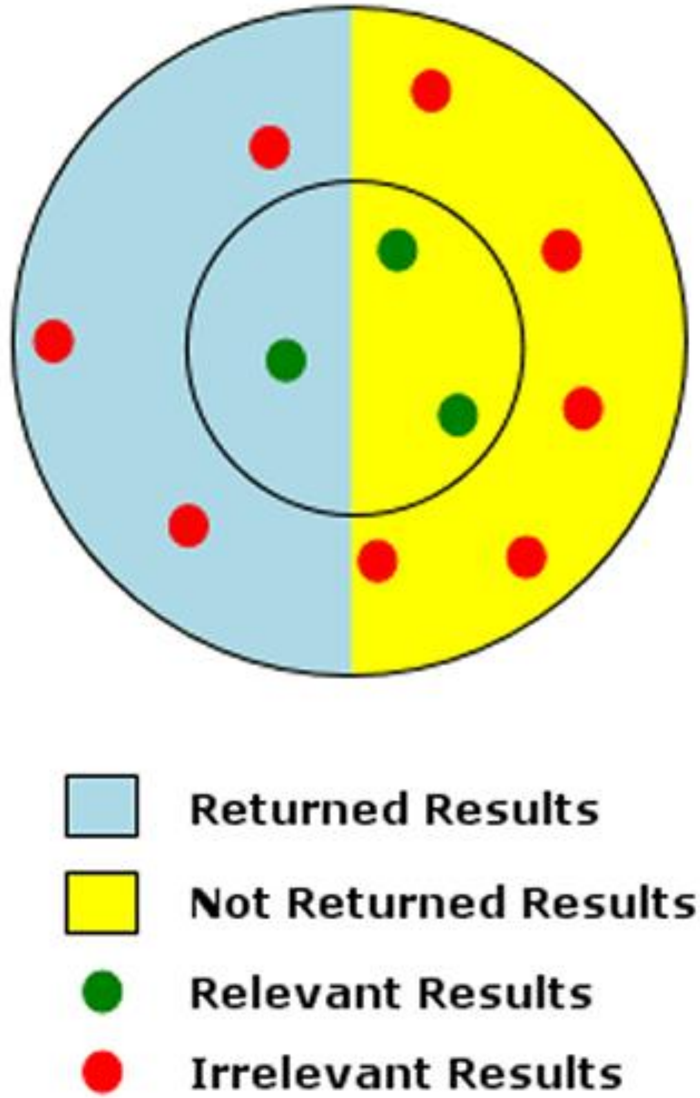
- Statistical measures
- Subjective measures
- Precision, recall
- A contingency table analysis of precision and recall

	Relevant	Non-relevant	
Retrieved	a	b	$a + b = m$
Not retrieved	c	d	$c + d = N - m$
	$a + c = n$	$b + d = N - n$	$a + b + c + d = N$

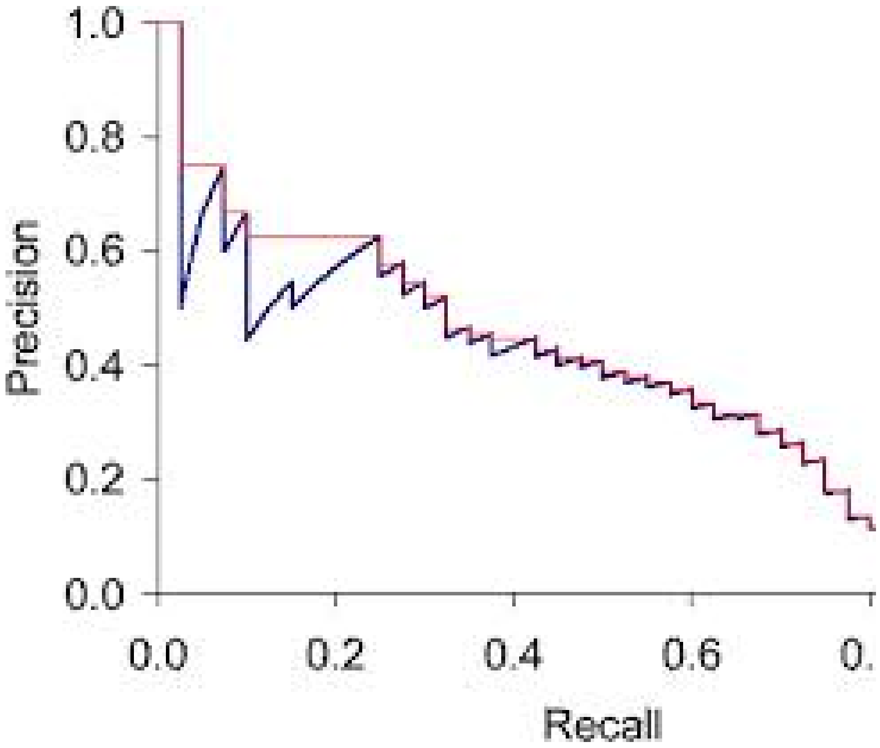
Precision and recall

- N = number of documents in collection
- n = number of important documents for given query q
- Search returns m documents including a relevant ones
- Precision $P = a/m$
  proportion of relevant document in the obtained ones
- recall $R = a/n$
  proportion of obtained relevant documents
- Precision recall graphs

An example: low precision, low recall



Precision-recall graphs



F-measure

- combine both P and R
- $$F_\beta = \frac{(1+\beta^2)\cdot P \cdot R}{\beta^2 P + R} \text{ for } \beta > 0$$

  $$F_1 = \frac{2 \cdot P \cdot R}{P + R}$$
- Weighted precision and recall
- Often used β=2 or β= 0.5
- β=1 weighted harmonic mean

Performance of ranking

- r_i is rank for i-th most important document
- Logarithmic precision

$$LogP = \frac{\sum_{i=1}^{n} \log i}{\sum_{i=1}^{n} \log r_i}$$

- Ranked recall

$$RankR = \frac{\sum_{i=1}^{n} i}{\sum_{i=1}^{n} r_i}$$

Improvements to search

- Use dictionary, thesaurus, synonyms (e.g., Wordnet, learn from corpus)
- Query expansion with relevance information
  - User feedback
  - Personalization
  - Trusted document sources
- Semantic search

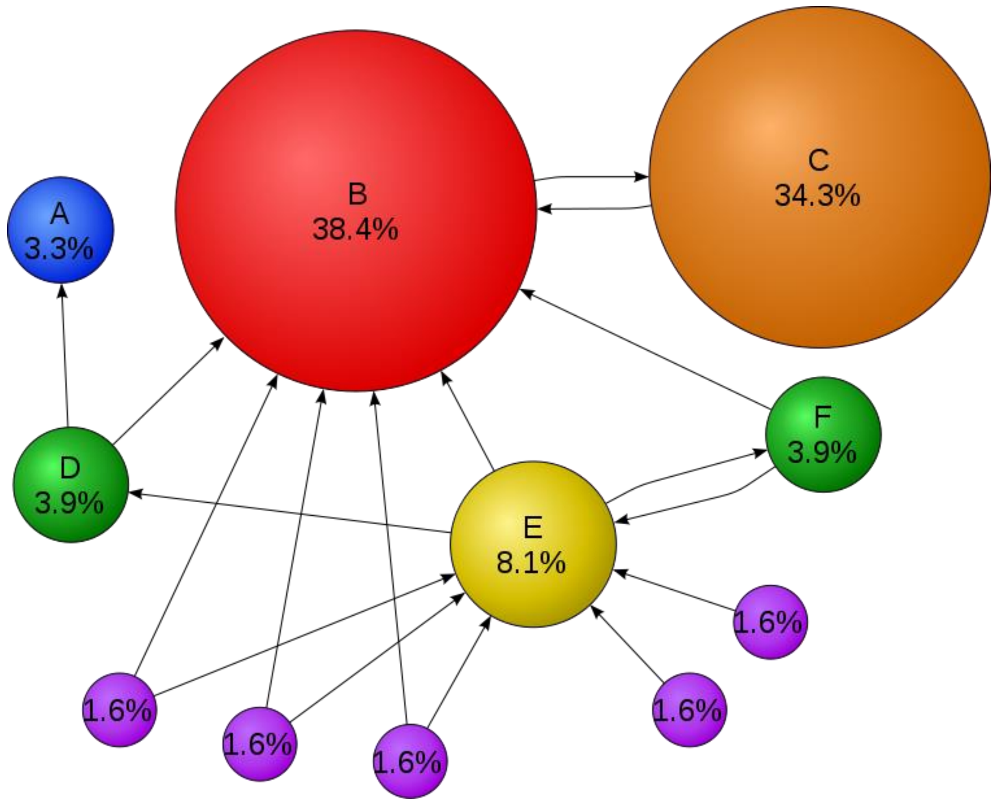
Web search problems

- No contents control
- Different quality of documents
- Up-to-date?
- (in)valid links
- Search engine manipulation

Specific improvements

- Specific types of queries require specific approaches
- Trustful sources -Wikipedia
- Hubs with relevant links (e.g., Yahoo)
- Graph theory and analysis, virtual communities,
- Additional information: titles, meta-information, URL
- ranking of documents based on links

Ranking documents - PageRank



PageRank formalization

- p = web page
- $O(p)$ = pages pointed to by p
- $I(p) = \{i_1, i_2, \ldots, i_n\}$ pages pointing to p
- d = damping factor between 0 and 1 (default 0.85 or 0.9)

$$\pi(p) = (1-d) + d\,\frac{\pi(i_1)}{|O(i_1)|} + \ldots + d\,\frac{\pi(i_n)}{|O(i_n)|}$$

- Page quality π(p) depends on quality of pages pointing to it

PageRank computation

- Iterative computation,
- matrix form
- random surfer, intentional surfer
- Personal PageRank
- Manipulation and defense (e.g., TrustRank)

Text classification

- Applications (use several classification algorithms)
- frequently used classification algorithms on text: Naïve Bayes, logistic regression, linear SVM (why?), deep neural networks
- document retrieval and search, selection of relevant news, categorization of news, messages, intranet, spam, sentiment detection and classification

Text mining

- Acquire new knowledge
- Summarization, document relations, clustering of documents, new topic detection, related news, directory of important people/institutions, taxonomies,
  named-entity
  extraction/recognition/disambiguation

References and coreferences

- Person recognition: president, George Bush, Mr. Bush, g. Bush head of state, he, bushism
- named entity recognition (NER): people, places, companies, products, trade marks, dates, numbers, percentages…
- Use directories, heuristics, iterative process
- Statistical approaches

Text summarization

- General, guided,
- One document, multi-document
- Summary and extraction
- Evaluation

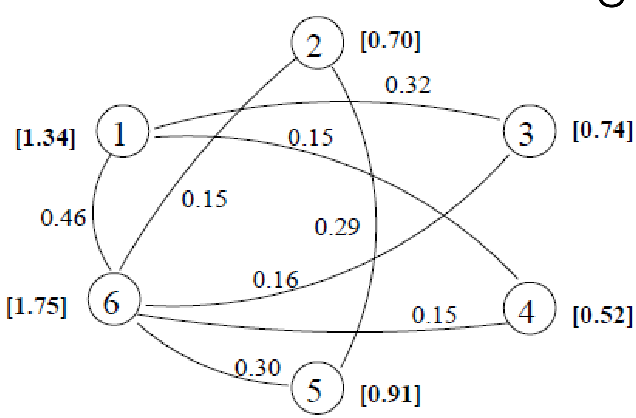
Graph-based summarization; An illustrative example

[1] Watching the new movie, "Imagine: John Lennon," was very painful for the late Beatle's wife, Yoko Ono.
[2] "The only reason why I did watch it to the end is because I'm responsible for it, even though somebody else made it," she said.
[3] Cassettes, film footage and other elements of the acclaimed movie were collected by Ono.
[4] She also took cassettes of interviews by Lennon, which were edited in such a way that he narrates the picture.
[5] Andrew Solt ("This Is Elvis") directed, Solt and David L. Wolper produced and Solt and Sam Egan wrote it.
[6] "I think this is really the definitive documentary of John Lennon's life," Ono said in an interview.

Sentences	Rank
6	**1.75**
1	**1.34**
5	0.91
3	0.74
2	0.70
4	0.52

Sentence ranking/selec

Text to graph/matrix



	1	2	3	4	5	6
1	0	0	0.32	0.15	0	0.46
2	0	0	0	0	0.29	0.15
3	0.32	0	0	0	0	0.16
4	0.15	0	0	0	0	0.15
5	0	0.29	0	0	0	0.30
6	0.46	0.15	0.16	0.15	0.30	0

Important in NLP, but beyond the scope

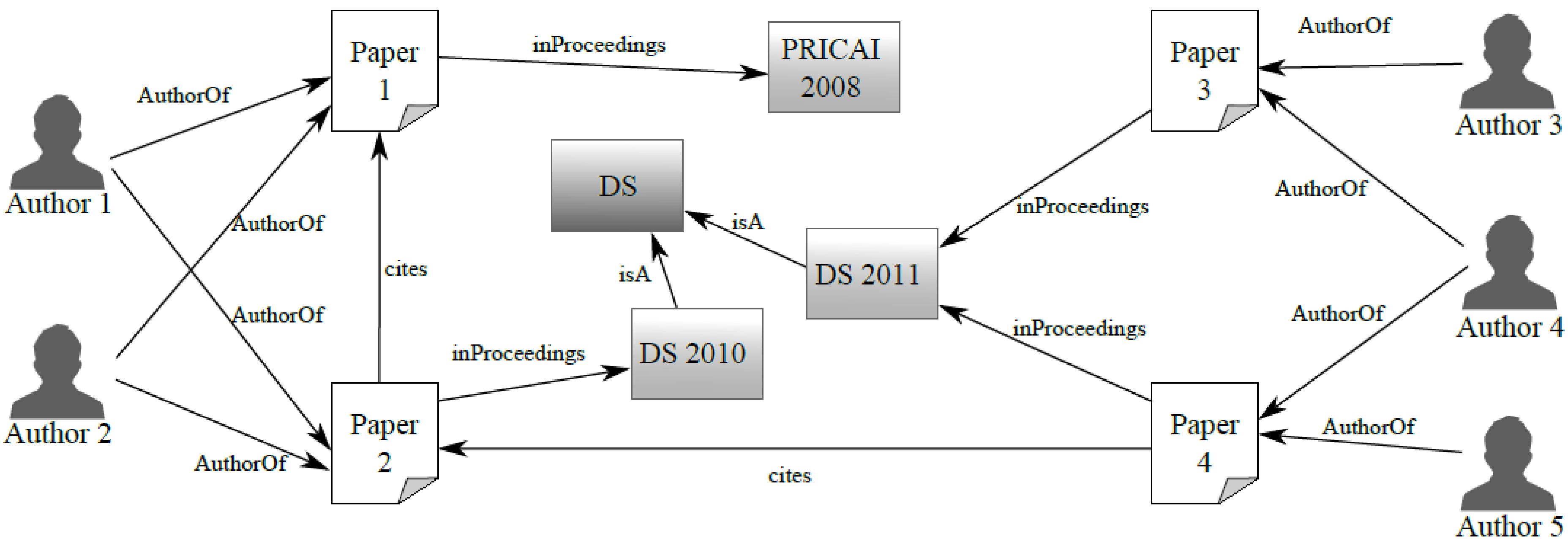
- obtaining labelled corpora
- Using technologies:
  - XML,
  - RDF (resource description framework) – labelled sources: subject-predicate-object
  - semantic web (metadata)
  - new data mining approaches: Multiview learning, text enriched (heterogeneous) networks
- (semi)intelligent assistants, agents
- (semi)automatic translation:
- using relational and structural information (graphs, parse trees)
- cross-lingual approaches

Applications

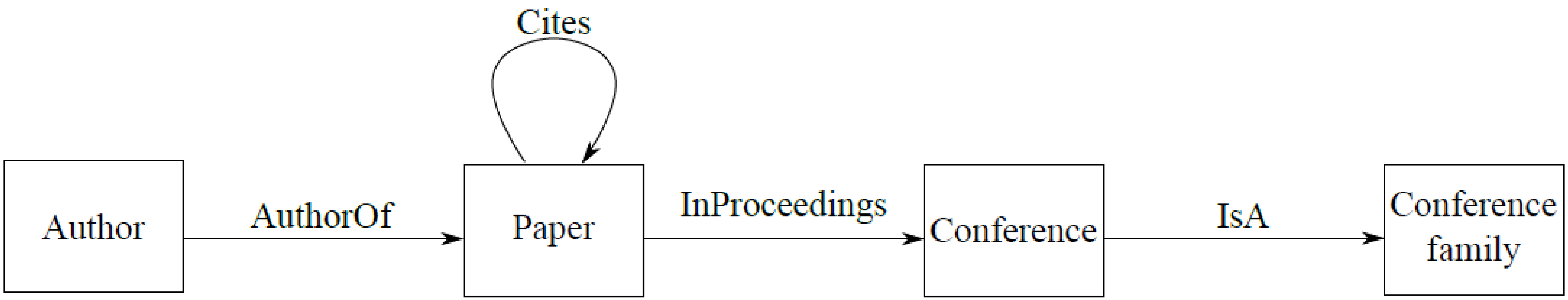
- Sentiment analysis (using sentiWordNet)
- Paper recommendations based on nonhomogeneous networks and BOW

Paper recommendations based on heterogeneous information networks

1. Identify and collect available information

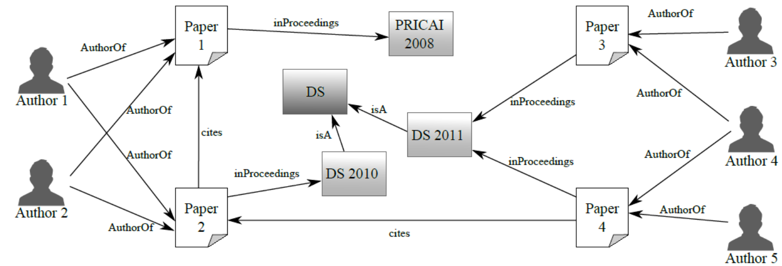


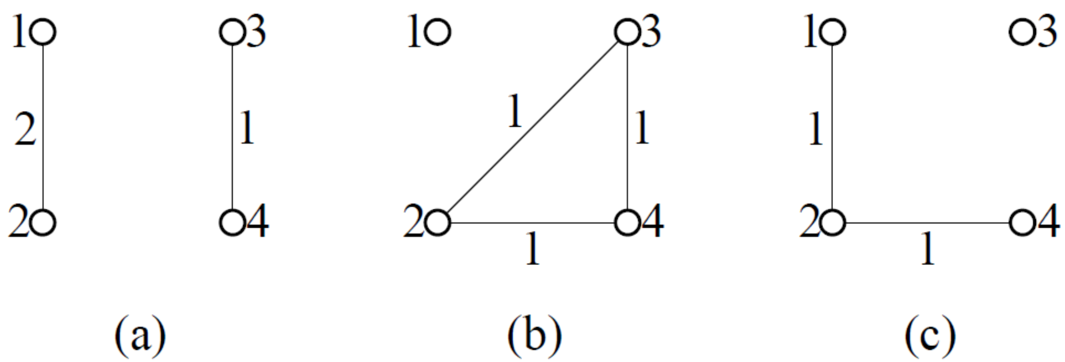
2. decompose heterogeneous networks



- create several homogeneous networks of the same type

Network decompositions





Select informative decompositions based on your purpose

a) paper-author-paper,

b) paper-conference family-paper

c) paper-paper

Vectorize the homogeneous networks

- compute personalized PageRank vector for each homogeneous network (PPR)
- normalize the PPR vectors with Euclidean normalization

3. Utilize text information

- Construct Bag-of-Words (BOW) vector for each paper
- process text with standard natural language processing techniques:
  - tokenization,
  - stop-word removal,
  - stemming,
  - construction of N-grams of certain length,
  - removal of infrequent words from the vocabulary
- normalize BOW vectors with Euclidean normalization

Combine text and network information

- each node now contains a set of vectors: a BOW vector and one PPR vector for each network decomposition
- combine vectors:
  - concatenate BOW and PPR vectors
  - merge them e.g. produce a linear mapping of each vector and obtain a new vector of the size of the vectors;

Tasks on networks

- determine authority of nodes
- cluster nodes
- classification and classification through label propagation
- recommendations of similar nodes

Sentiment analysis (SA)

- Definition: computational study of opinions, sentiments, emotions, and attitude expressed in texts towards an entity.
- Purpose: detecting public moods i.e., understanding the opinions of the general public and consumers on social events, political movements, company strategies, marketing campaigns, product preferences etc.

SA: getting and preprocessing data

- Frequent data sources:
  - Twitter, forum comments, product review sites
- Preprocessing: tokenization, stop word removal, stemming, parts of speech (POS) tagging, and feature extraction/representation/selection

Sentiment classification

- binary (polarity), ternary, n-ary
- lexicon based:
  - based on ontology or not, corpus based, created from initial seed, using WordNet, cross-lingual etc.
- machine learning based
- hybrid

Other SA tasks

- subjectivity classification (vs. objectivity)
- review usefulness classification
- opinion spam classification