

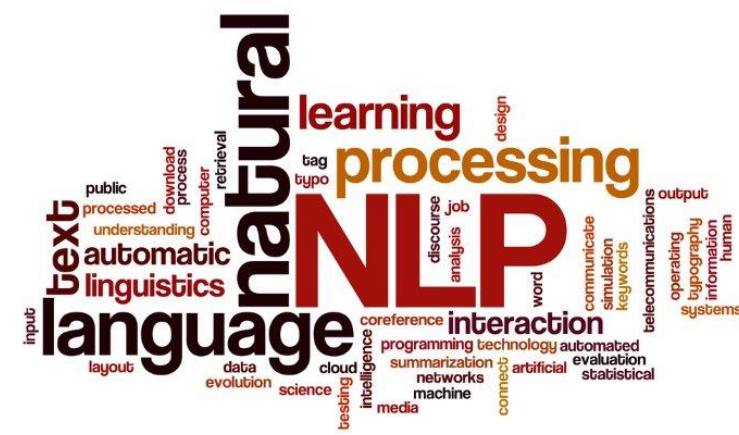
University of Ljubljana, Faculty of Computer and Information Science

# Natural language processing



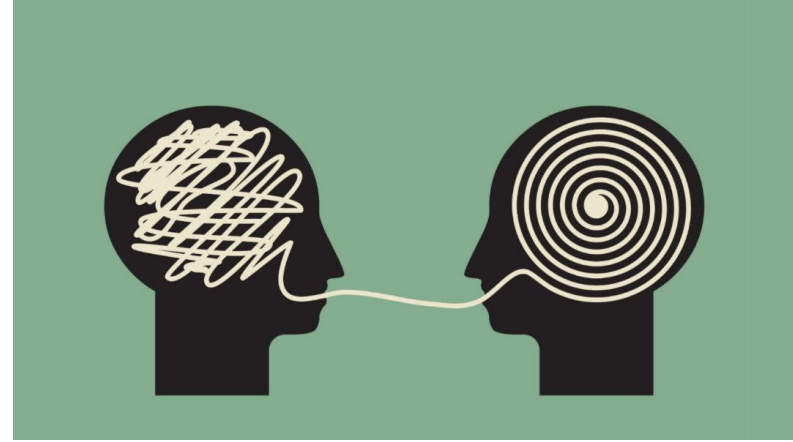
Prof Marko Robnik-Šikonja  
Intelligent Systems, December 2022

# Topic overview



- understanding language and intelligence
- approaches to language analysis
- important tasks and components for NLP
  - text representations
  - 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,*

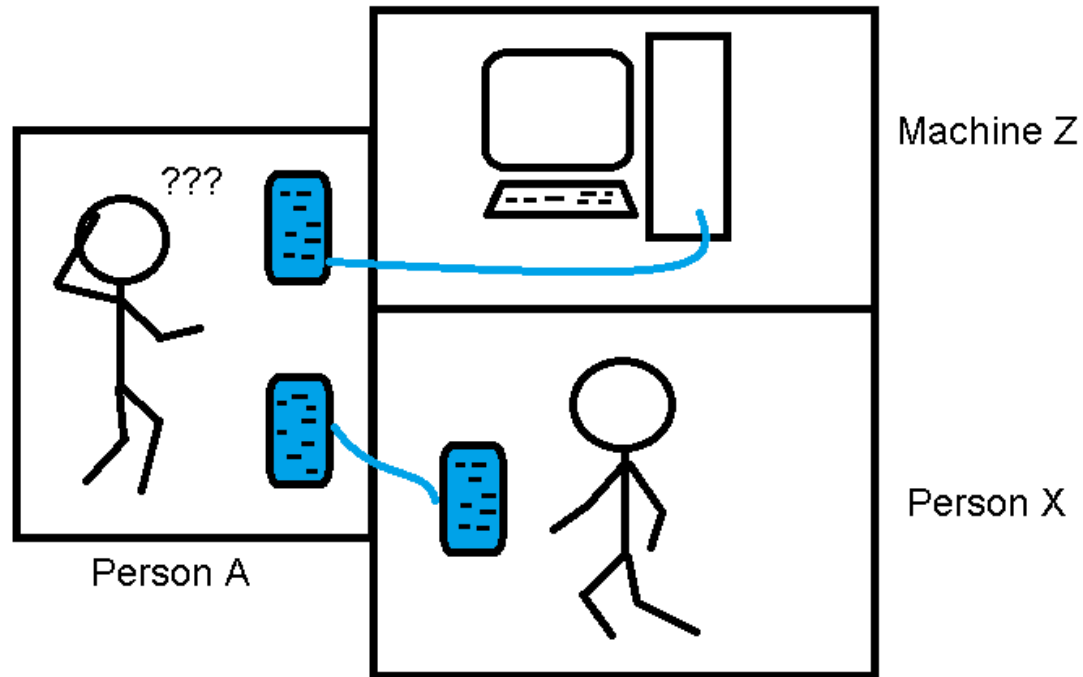
➤ *POS tagging, word sense disambiguation*

# 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)

# ELIZA

```
=====
EEEEEEEE L      IIIIII ZZZZZZ      AAA
E         L         I         Z         A   A
E         L         I         Z         A   A
EEEEEE   L         I         Z         A   A
E         L         I         Z         A   A
E         L         I         Z         A   A
EEEEEEEE LLLLLLL IIIIII ZZZZZZ      A   A
=====
ELIZA > (PLEASE TYPE IN ALL CAPS) WHAT'S YOUR NAME DEAR ?
AMIT
ELIZA > HI, I'M ELIZA. WHAT DO YOU WANT TO TALK ABOUT ?
AMIT > CAN I TALK ABOUT MY PROBLEM ?
ELIZA > SURE...
AMIT > DUE TO ON GOING EXAMS IN COLLEGE, I HAVE A LOT OF STRESS.
ELIZA > PLEASE RELAX A LITTLE. SLEEP WELL.
AMIT > THANKS FOR YOUR ADVICE.
ELIZA > NO MENTION.
AMIT > BYE
ELIZA > BYE AND KEEP IN TOUCH...
=====
```

- ➔ ELIZA (Weizenbaum, 1966)
- ➔ limited conversation (Rogerian psychiatrist)

User: You are like my father in some ways.

ELIZA: What resemblance do you see?

User: You are not very aggressive but I think you don't want me to notice that.

ELIZA: What makes you think I am not aggressive?

User: You don't argue with me.

ELIZA: Why do you think I don't argue with you?

User: You are afraid of me.

ELIZA: Does it please you to believe I am afraid of you?

# 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.

# GPT-3 regressive language model as a chatbox

► the example is cherry-picked

Q: What is your favorite animal?

A: My favorite animal is a dog.

Q: Why?

A: Because dogs are loyal and friendly.

Q: What are two reasons that a dog might be in a bad mood?

A: Two reasons that a dog might be in a bad mood are if it is hungry or if it is hot.

Q: How many eyes does a giraffe have?

A: A giraffe has two eyes.

Q: How many legs does a frog have?

A: A frog has four legs.

Q: Are there any animals with three legs?

A: No, there are no animals with three legs.

Q: Why don't animals have three legs?

A: Animals don't have three legs because they would fall over.

# Where is NLP today?

- ▶ active research area with many commercial applications
  - ▶ speech recognition and synthesis
  - ▶ automatic reply engines
  - ▶ machine translation
  - ▶ text summarization
  - ▶ question answering
  - ▶ language generation
  - ▶ interface to databases
  - ▶ intelligent search and information extraction
  - ▶ sentiment detection
  - ▶ semantic analysis: e.g. role labelling,
  - ▶ named entity recognition and linking
  - ▶ categorization, classification documents, messages, tweets, etc.
  - ▶ cross-lingual approaches
  - ▶ multi modal approaches (text + images, text + video)
  - ▶ attempt to get to artificial general intelligence (AGI) through “foundation models”
  - ▶ many (open-source) tools and language resource
  - ▶ prevalence of deep neural network approaches (transformers)

# Recommended literature

- ▶ Jurafsky, Daniel and James Martin (2021): Speech and Language Processing, 3rd edition in progress, almost all parts are available at authors' webpages  
<https://web.stanford.edu/~jurafsky/slp3/>
- ▶ Steven Bird, Ewan Klein, and Edward Loper. Natural Language Processing with Python. O'Reilly, 2009
  - ▶ a free book accompanying NLTK library, regularly updated
  - ▶ Python 3, <http://www.nltk.org/book/>
- ▶ Coursera
  - ▶ several courses, e.g., Stanford NLP with DNN

# Historically two approaches

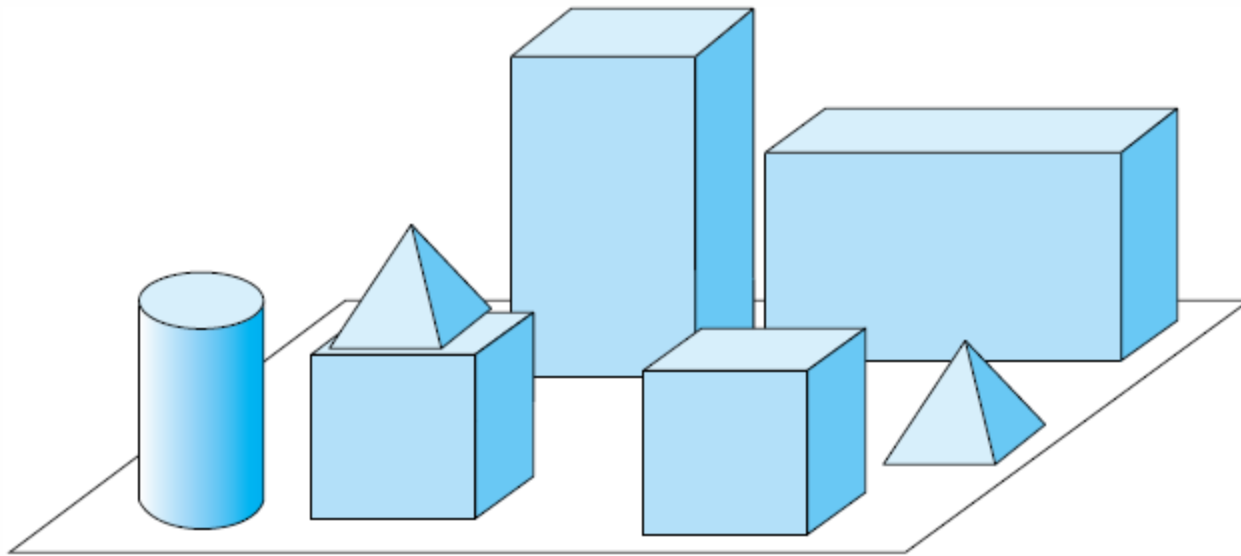
- ▶ symbolical
  - ▶ „Good Old-Fashioned AI”
- ▶ empirical
  - ▶ Statistical, corpuses
- ▶ Merging both worlds: injecting symbolical knowledge into DNNs

# 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

# Classical approach to text processing

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

In the neural approach, the preprocessing remains (but is simpler) the other three phases are merged into DNN

# 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€ or 1.999,00€!
  - ▶ Ravne na Koroškem
  - ▶ Lebensversicherungsgesellschaft
  - ▶ Port-au-prince
  - ▶ Generalstaatsverordnetenversammlungen
- ▶ Rules, finite automata, statistical models, dictionaries (of proper names), neural networks

# Lemmatization

- ▶ Lemmatization is the process of grouping together the different inflected forms of a word so they can be analyzed as a single item.
- ▶ walk is the lemma of 'walk', 'walked', 'walks', 'walking'
- ▶ Lemmatization difficulty is language dependent i.e., depends on morphology
- ▶ Requires dictionary or lexicon for lookup  
*go, goes, going, gone, went*  
*jaz, mene, meni, mano*
- ▶ Uses rules, neural networks and dictionaries
- ▶ Ambiguity resolution may be difficult

Meni je vzel z mize (zapestnico).

# Stemming – a quick and imprecise solution to many wordforms

- ▶ Stemming is the process of reducing inflected (or sometimes derived) words to their word stem, base or root form.
- ▶ The stem need not be identical to the morphological root of the word and may not even be a valid word.
- ▶ 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; Example: meeting; a lemma depends on the context and is to meet or a meeting
- ▶ *cat* is the stem of *cats*, *catlike*, and *catty*
- ▶ *fish*: *fishing*, *fished*, and *fisher*
- ▶ *argu*: *argue*, *argued*, *argues*, *arguing*, *argus* (by Porter algorithm)
- ▶ Use of rules, e.g., Porter algorithm
- ▶ Quick solutions and heuristics, in English just remove suffixes: – *ing*, *-ation*, *-ed*, ...



# POS tagging

- ▶ assigning the correct part of speech (noun, verb, etc.) to words
- ▶ helps in recognizing phrases and names
- ▶ Use rules, machine learning models

# Named entity recognition (NER)

- ▶ NATO Secretary-General Jens Stoltenberg is expected to travel to Washington, D.C. to meet with U.S. leaders.
- ▶ [ORG NATO] Secretary-General [PER Jens Stoltenberg] is expected to travel to [LOC Washington, D.C.] to meet with [LOC U.S.] leaders.
- ▶ Named entity linking (NEL) also named entity disambiguation – linking to a unique identifier, e.g. wikification  
*jaguar, Paris, London, Dunaj*

# 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:

- ▶ Text analyzer for Slovene, i.e. morphosyntactical tagging, available at <https://orodja.cjvt.si/oznacevalnik/slv/>

*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

# Classla pipeline output

ID	Oblika	Lema	Oznaka JOS	Bes. vrsta JOS	Oblikoskladenjske lastnosti JOS	Bes. vrsta UD	Oblikoskladenjske lastnosti UD	Nadrejena pojavnica UD	Skladenjska relacija UD	Nadrejena pojavnica JOS	Skladenjska relacija JOS	Udelež vloga
# paragraph 1												
# sent_id 1.1												
# text = Nekega dne sem se napotil v naravo.												
1	Nekega	nek	Pi-msg	Pronoun	Type=indefinite Gender=masculine Number=singular Case=genitive	DET	Case=Gen Gender=Masc Number=Sing PronType=Ind	2	det	2	Atr	
2	dne	dan	Ncmsg	Noun	Type=common Gender=masculine Number=singular Case=genitive	NOUN	Case=Gen Gender=Masc Number=Sing	5	obl	5	AdvO	TIME
3	sem	biti	Va-r1s-n	Verb	Type=auxiliary VForm=present Person=first Number=singular Negative=no	AUX	Mood=Ind Number=Sing Person=1 Polarity=Pos Tense=Pres VerbForm=Fin	5	aux	5	PPart	
4	se	se	Px-----y	Pronoun	Type=reflexive Clitic=yes	PRON	PronType=Prs Reflex=Yes Variant=Short	5	expl	5	PPart	
5	napotil	napotiti	Vmep-sm	Verb	Type=main Aspect=perfective VForm=participle Number=singular Gender=masculine	VERB	Aspect=Perf Gender=Masc Number=Sing VerbForm=Part	0	root	0	Root	
6	v	v	Sa	Adposition	Case=accusative	ADP	Case=Acc	7	case	7	Atr	
7	naravo	narava	Ncfsa	Noun	Type=common Gender=feminine Number=singular Case=accusative	NOUN	Case=Acc Gender=Fem Number=Sing	5	obl	5	AdvO	GOAL
8	.	.	Z	Punctuation		PUNCT	_	5	punct	0	Root	

- 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 lema oznaka	Nekega dne sem se napotil v naravo . Že spočetka me je nek dan biti se napotiti v narava že spočetka jaz biti Zn-mer Somer Gp-spe-n Zp-----k Ggdd-em Dt Sozet . L Rsn Zop-et--k Gp-ste-n
2	beseda lema oznaka	žulil čevelj , a sem na to povsem pozabil , ko sem jo zagledal žuliti čevelj a biti na ta povsem pozabiti ko biti on zagledati Ggnd-em Somei , Vp Gp-spe-n Dt Zk-set Rsn Ggdd-em , Vd Gp-spe-n Zozet--k Ggdd-em
3	beseda lema oznaka	. Bila je prelepa . Povsem nezakrita se je sončila na trati biti biti prelep povsem nezakrit se biti sončiti na trata . Gp-d-ez Gp-ste-n Ppnzei . Rsn Ppnzei Zp-----k Gp-ste-n Ggvd-ez Dm Sozem
4	beseda lema oznaka	ob poti . Pritisk se mi je dvignil v višave . Popoln ob pot pritisk se jaz biti dvigniti v višava popoln Dm Sozem . Somei Zp-----k Zop-ed--k Gp-ste-n Ggdd-em Dt Sozmt . Ppnmei
5	beseda lema oznaka	primerek kmečke lastovke ! primerek kmečki lastovka 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
		đvovidski	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
		đruga	d		second	2
		tretja	t		third	3
5	števílo	ednina	e	Number	singular	s
		množina	m		plural	p
		đvojina	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 simple method how POS tagger for English can work: n-gram tagging

- ▶ Context of  $n-1$  preceding words
- ▶ Corpus based learning
- ▶ What about succeeding words?
- ▶ Markov models, HMM, learning with EM

maximize

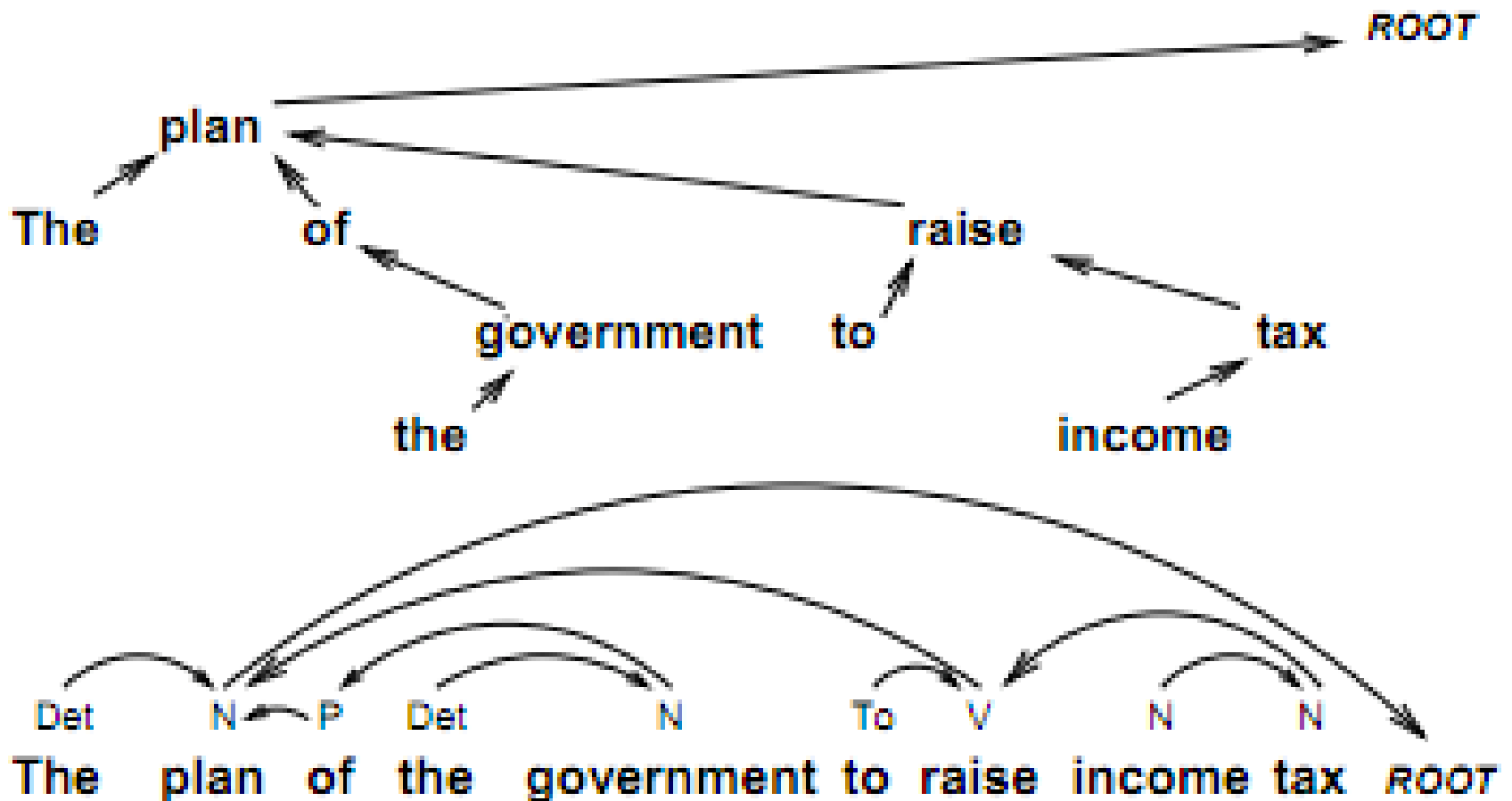
$P(\text{word} \mid \text{tag}) \times P(\text{tag} \mid \text{previous } n \text{ tags})$

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

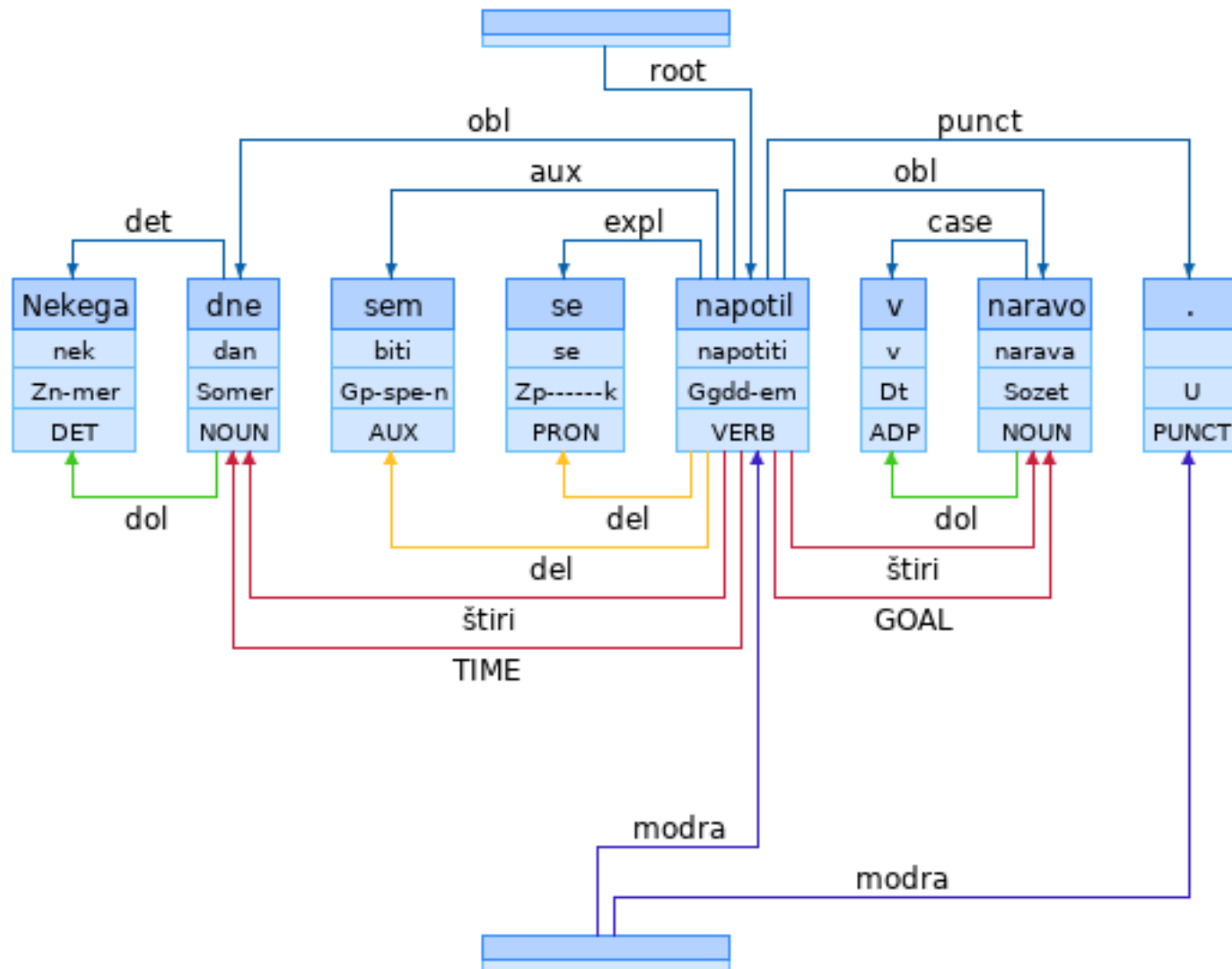
# Grammars

- Many tools: NLTK in python, prolog, etc.
- Grammars exist only for a few languages
- Ambiguity, several parsing trees
- Dependency parsing is much more widespread

# Dependency parser (tree bank)



# Dependency parsing example



# Example of grammar

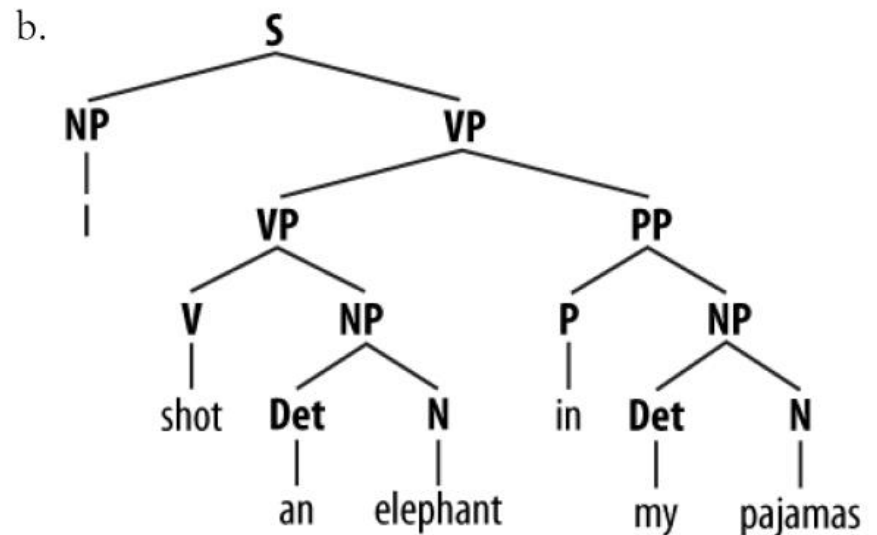
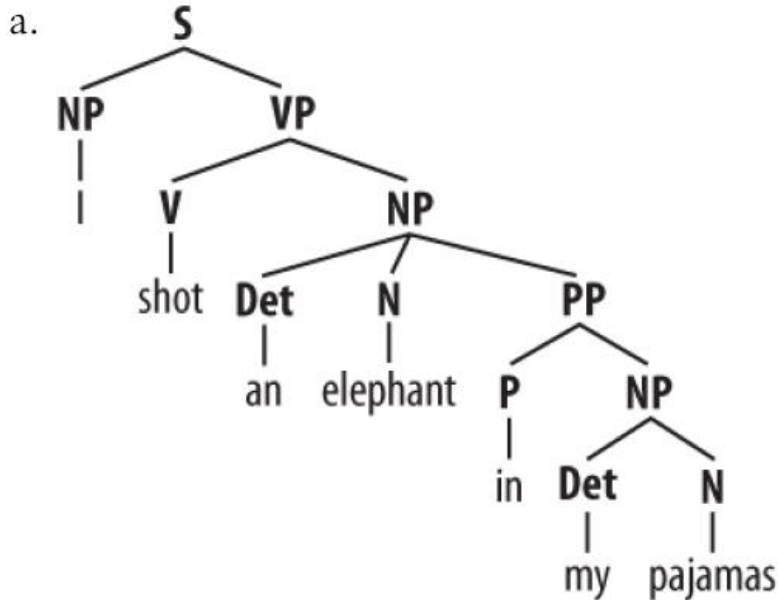
*While hunting in Africa, I shot an elephant in my pajamas.*

*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.*

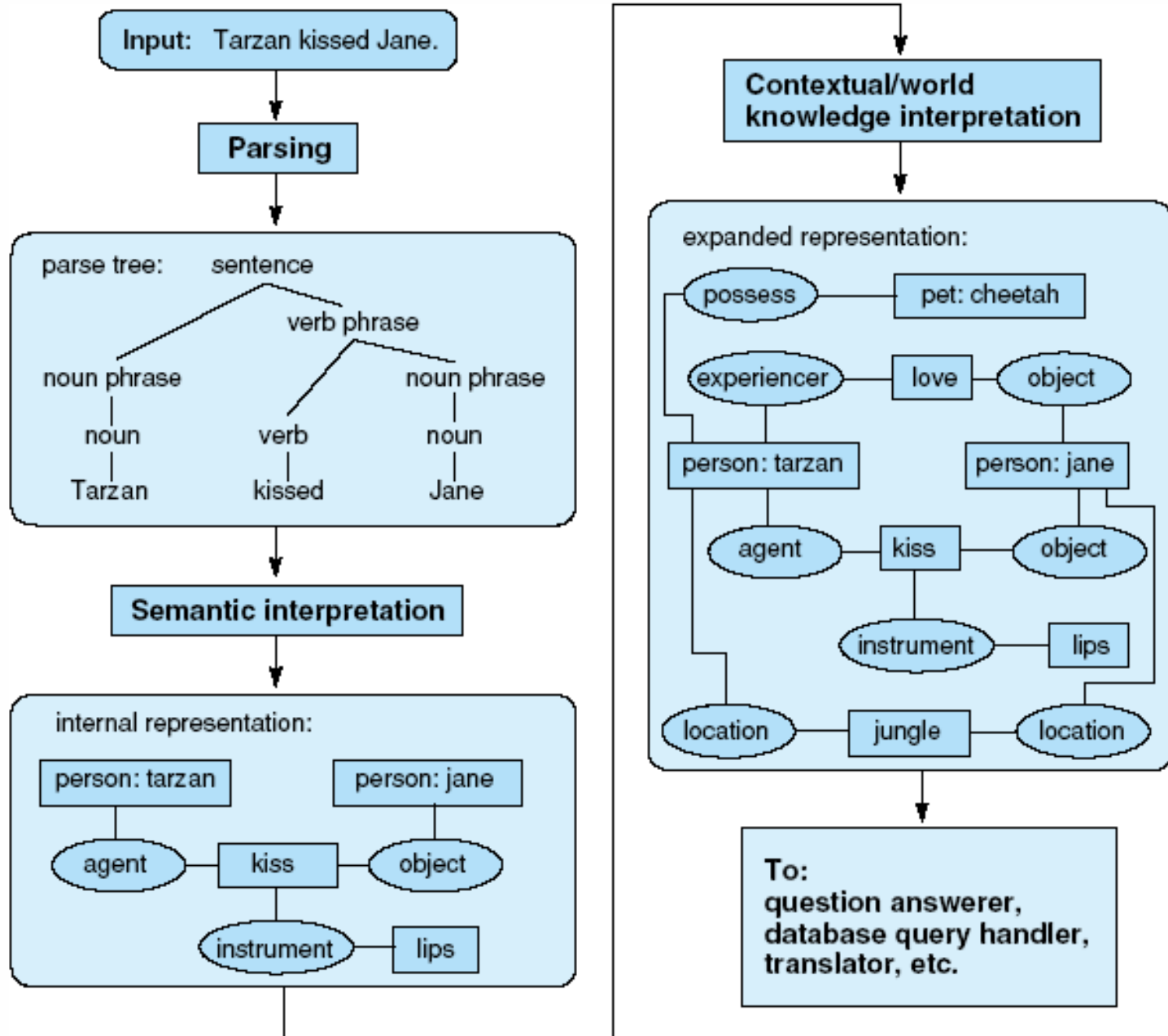


## 2. phase - interpretation

- ▶ 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



# Basic language resources: corpora

- ▶ Statistical natural language processing list of resources  
<http://nlp.stanford.edu/links/statnlp.html>
- ▶ Opus <http://opus.nlpl.eu/> multilingual parallel corpora, e.g., DGT JRC-Acqui 3.0, Documents of the EU in 22 languages
- ▶ Slovene language corpora GigaFida, ccGigaFida, KRES, ccKres, GOS, JANES, KAS
- ▶ The main Slovene language resources
  - ▶ <http://www.clarin.si>
  - ▶ <https://github.com/clarinsi>
  - ▶ <http://www.cjvt.si/>
  - ▶ <https://www.slovenscina.eu/>
- ▶ WordNet, SloWNet, sentiWordNet, ...
- ▶ Thesaurus <https://viri.cjvt.si/sopomenke/slv/>
- ▶ Dictionaries: SSKJ2, FRAN

WordNet is a database composed of synsets:  
synonyms,  
hypernyms  
hyponyms,  
meronyms,  
holonyms,  
etc.

Word to search for:

Display Options:

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"*

# Popular NLP applications

- document retrieval
- information extraction
- automatic speech recognition and generation
- text classification
- automatic summarization
- question answering
- sentiment analysis, emotion detection, stance detection
- machine translation,
- language generation
- comment filtering, hate speech detection, fake news detection
- topic analysis
- grammar tools
- many more

# 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

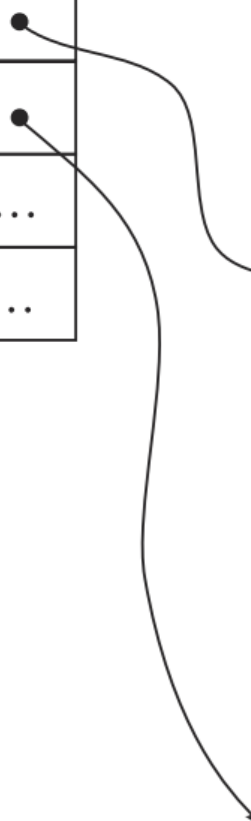
- ▶ Collect all words from all documents, use lemmatization
- ▶ inverted file
- ▶ For each word keep
  - ▶ Number of appearing documents
  - ▶ Overall number of appearances
  - ▶ For each document
    - ▶ Number of appearances
    - ▶ Location



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 or dense embeddings)

# 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

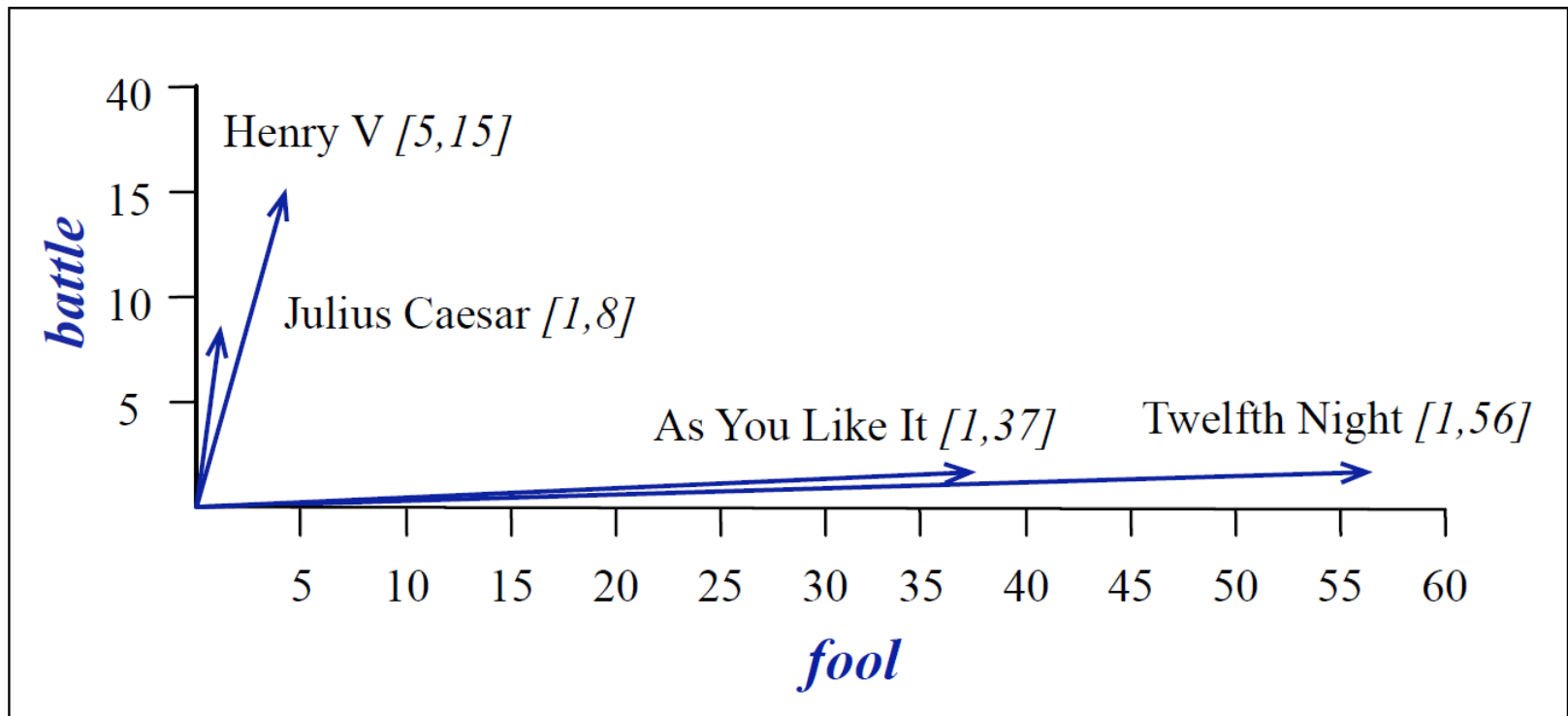
- ▶ a word occurs in several documents
- ▶ both words and documents are vectors
- ▶ an example: Shakespeare

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

- ▶ term-document matrix, dimension  $|V| \times |D|$
- ▶ 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

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

- ▶ Ranking by the decreasing similarity

# Performance measures for search

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

---

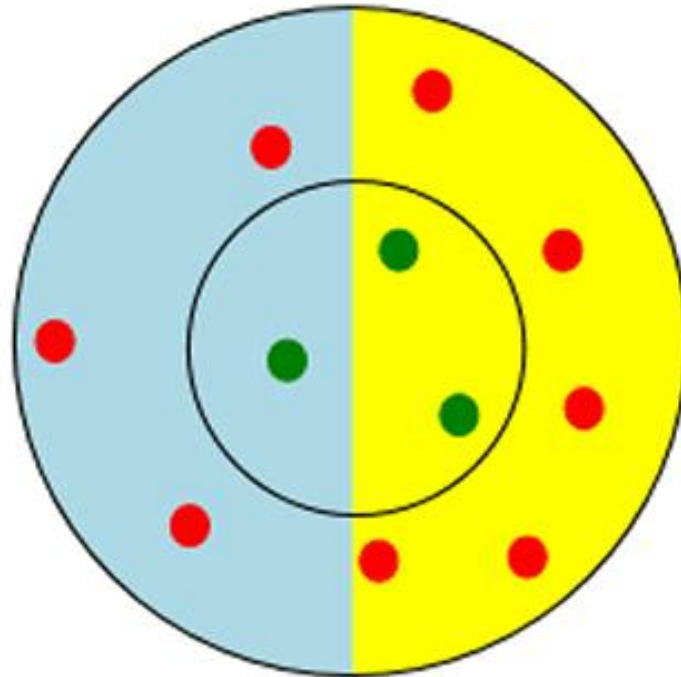
	Relevant	Non-relevant	
Retrieved	$a$	$b$	$a + b = m$
Not retrieved	$c$	$d$	$c + d = N - m$
	$a + c = n$	$b + d = N - n$	$a + b + c + d = N$

---

# Precision and recall

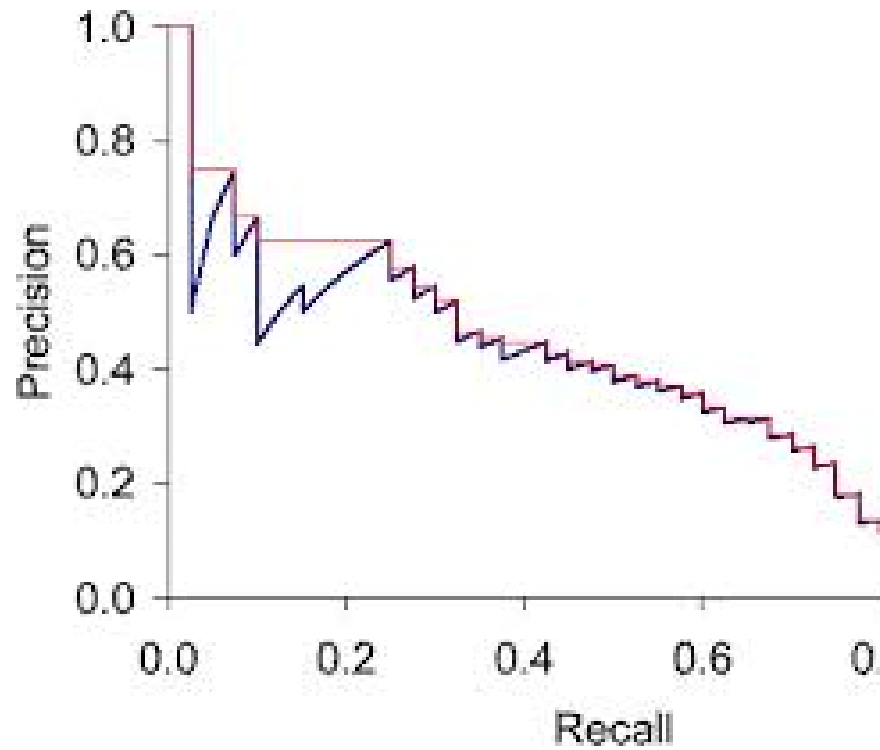
- $N$  = number of documents in collection
- $n$  = number of important documents for given query  $q$
- Search returns  $m$  documents including  $a$  relevant ones
- Precision  $P = a/m$   
proportion of relevant document in the obtained ones
- Recall  $R = a/n$   
proportion of obtained relevant documents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

An example: low precision, low recall



- Returned Results
- Not Returned Results
- Relevant Results
- Irrelevant Results

# 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
- $\beta=1$  weighted harmonic mean
- Also used  $\beta=2$  or  $\beta=0.5$

# Other ranking measures

- ▶ precision@k

- ▶ proportion of relevant document in the first k obtained ones

- ▶ recall@k

- ▶ proportion of relevant documents in the k obtained among all relevant

- ▶  $F_1@k$

- ▶ mean reciprocal rank

$$MRR = \frac{1}{Q} \sum_{i=1}^Q \frac{1}{\text{rank}_i}$$

- ▶ over Q queries,

- ▶ considers only the rank of the best answer

# 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

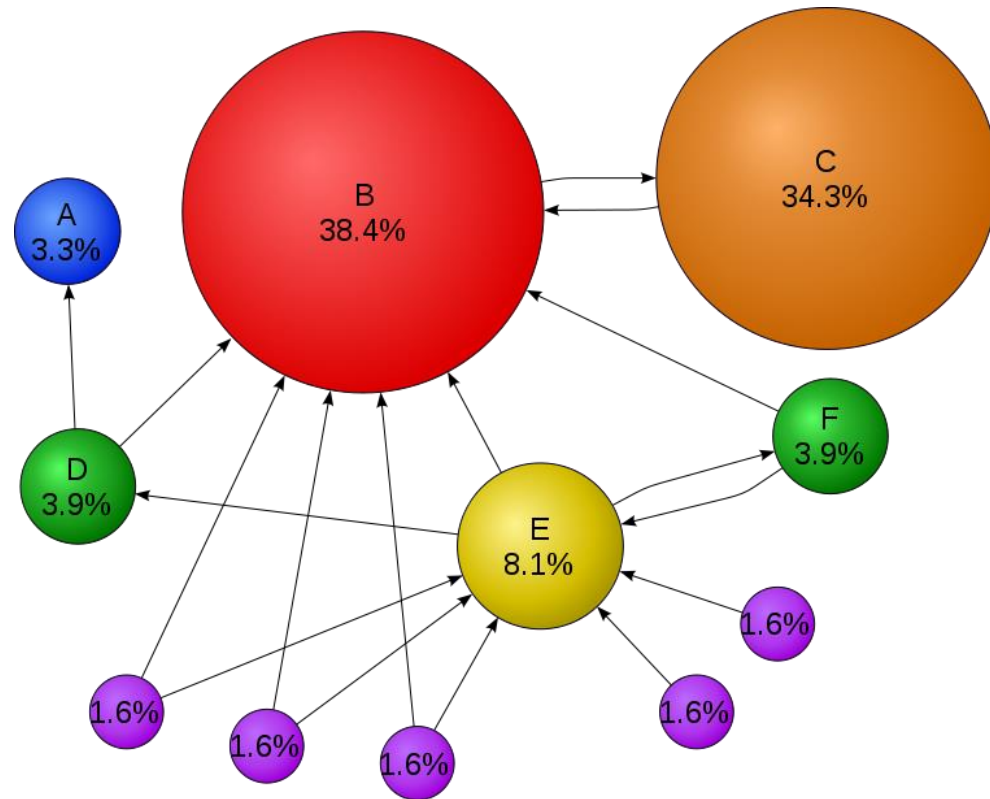
# Web search problems

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

# 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  $p$
- $I(p) = \{i_1, i_2, \dots, i_n\}$  pages pointing to  $p$
- $d$  = damping factor between 0 and 1 (default 0.85 or 0.9)

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

- Page quality  $\pi(p)$  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)



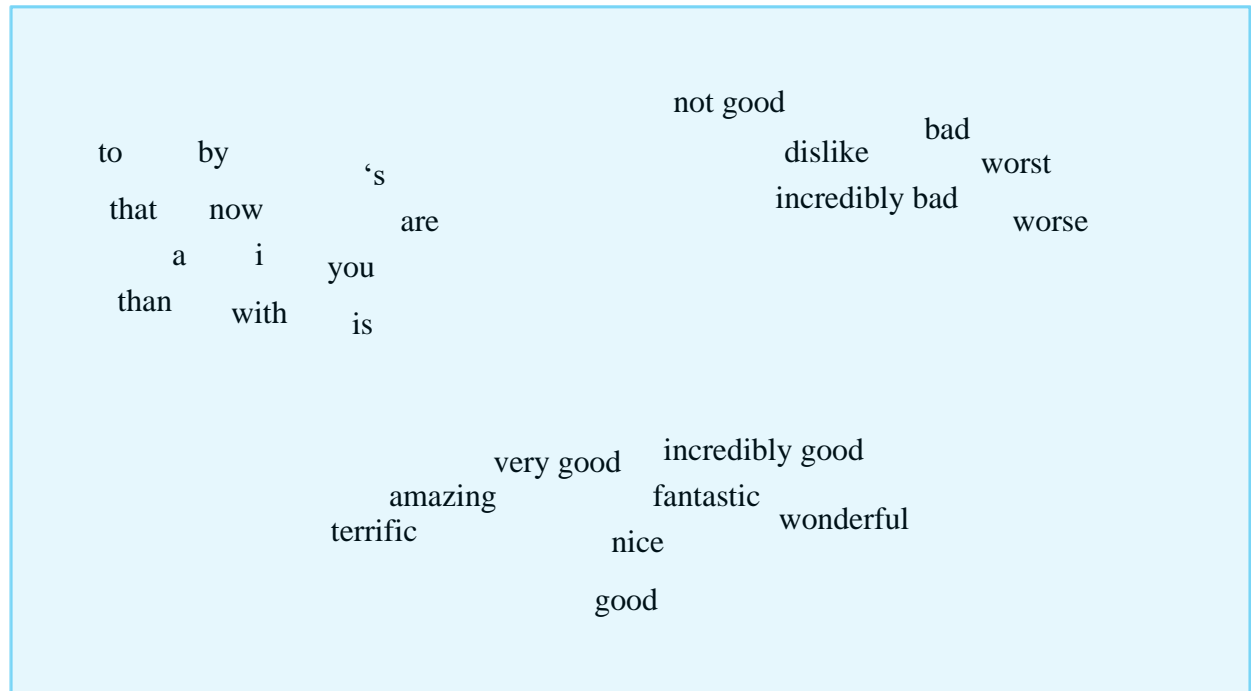


# Dense vector embeddings

- ▶ 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 in LSA)
  - ▶ neural embeddings (word2vec, Glove)
  - ▶ contextual neural embeddings (ELMo, BERT)

# Meaning focused on similarity

- Each word is a vector
- Similar words are "nearby in space"



# Distributional semantics



“You shall know a word  
by the company it keeps”

Firth, J. R. (1957). A synopsis of linguistic theory 1930–1955. In *Studies in Linguistic Analysis*, p. 11. Blackwell, Oxford.



"The meaning of a word is its  
use in the language"

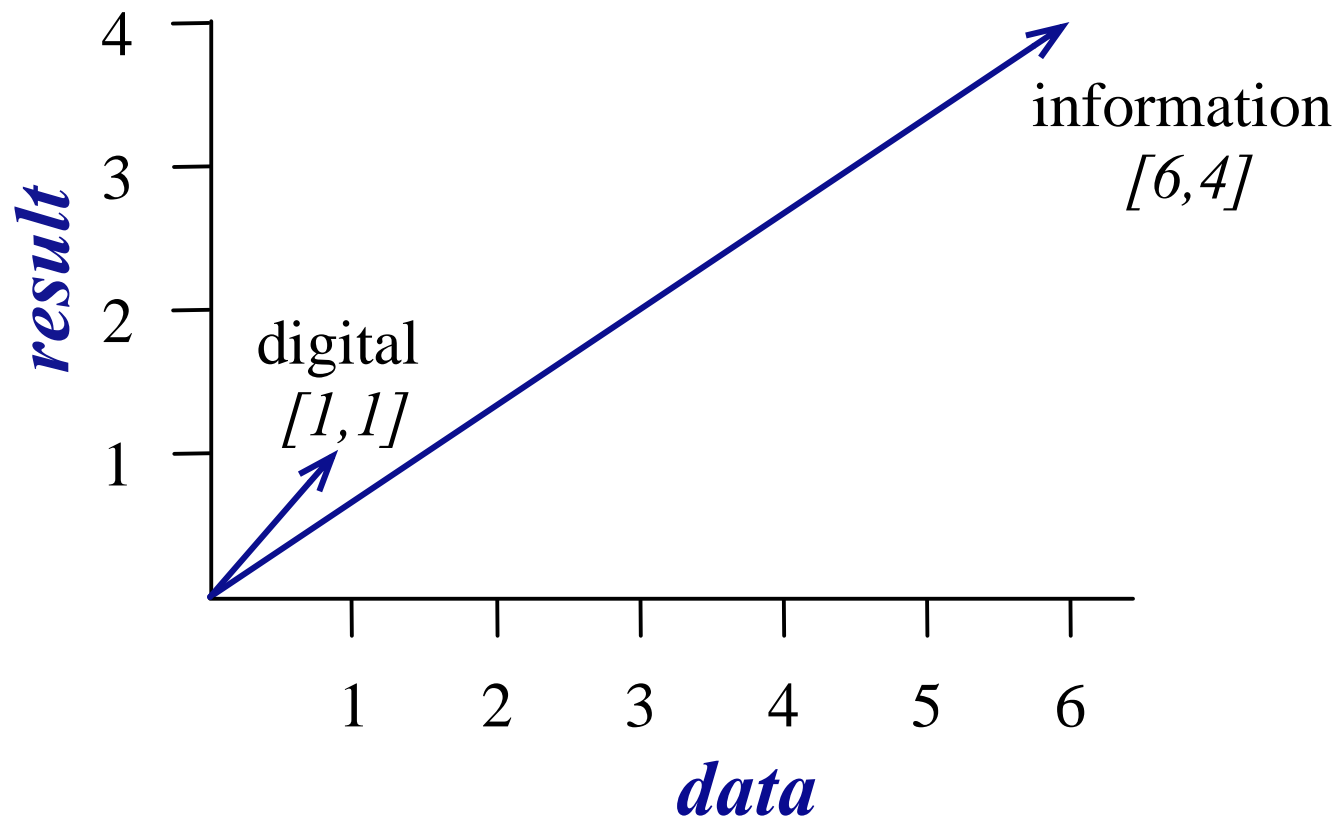
Ludwig Wittgenstein, PI #43

# Word-word matrix (or "term-context matrix")

- Two **words** are similar in meaning if their context vectors are similar.

sugar, a sliced lemon, a tablespoonful of their enjoyment. Cautiously she sampled her first well suited to programming on the digital for the purpose of gathering data and **apricot pineapple computer. information** jam, a pinch each of, and another fruit whose taste she likened In finding the optimal R-stage policy from necessary for the study authorized in the

	aardvark	computer	data	pinch	result	sugar	...
apricot	0	0	0	1	0	1	
pineapple	0	0	0	1	0	1	
digital	0	2	1	0	1	0	
information	0	1	6	0	4	0	



# Types of dense embeddings

- latent semantic analysis (LSA) - based on word-context matrix decomposition
- neural embeddings, e.g., word2vec
- context-sensitive neural embeddings: ELMo and BERT

# SVD for matrices

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

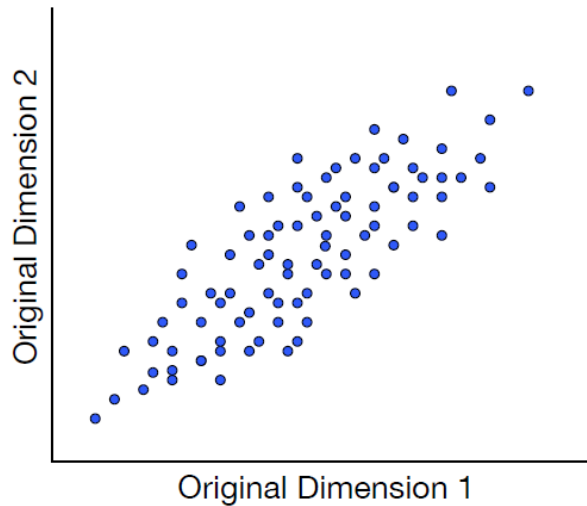
$$M = U\Sigma V^T$$

- ▶ 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

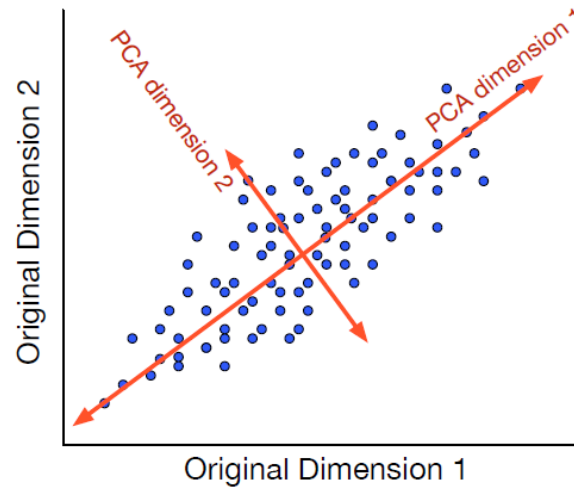


# Principle components analysis

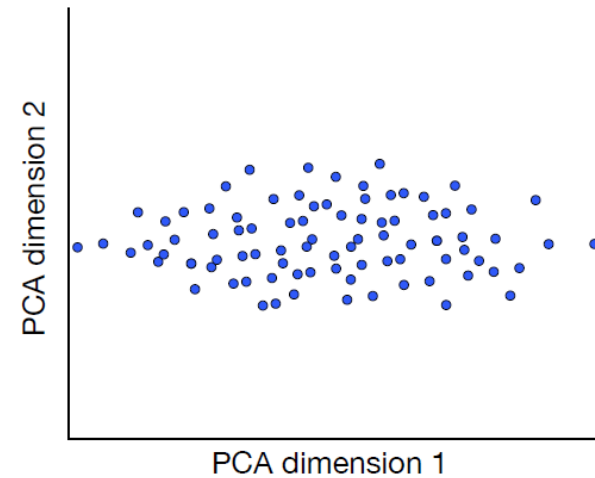
- ▶ principle components analysis, PCA
- ▶ we iteratively find the orthogonal axes of the largest variance
- ▶ we use the new dimensions to approximate the original space



(a)



(b)



(c)

# Latent semantic analysis

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

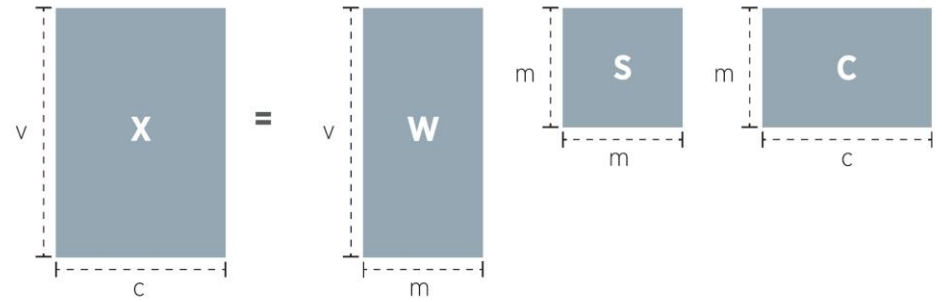
# Diagram LSA

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

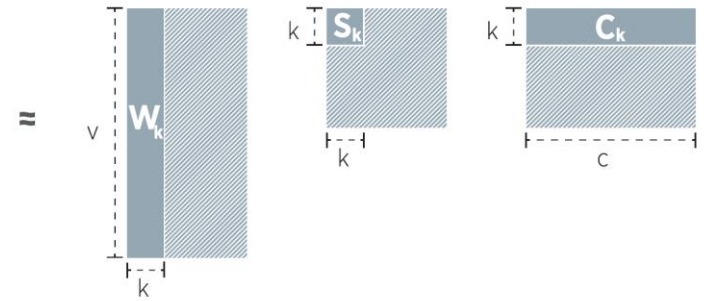
$$\begin{bmatrix} X \\ |V| \times c \end{bmatrix} = \begin{bmatrix} W_k \\ |V| \times k \end{bmatrix} \begin{bmatrix} \sigma_1 & 0 & 0 & \dots & 0 \\ 0 & \sigma_2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_3 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \sigma_k \end{bmatrix}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

# SVD for embeddings

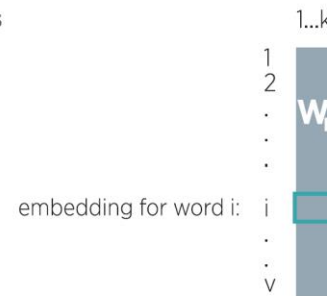
## 1. SVD



## 2. Truncation



## 3. Embeddings



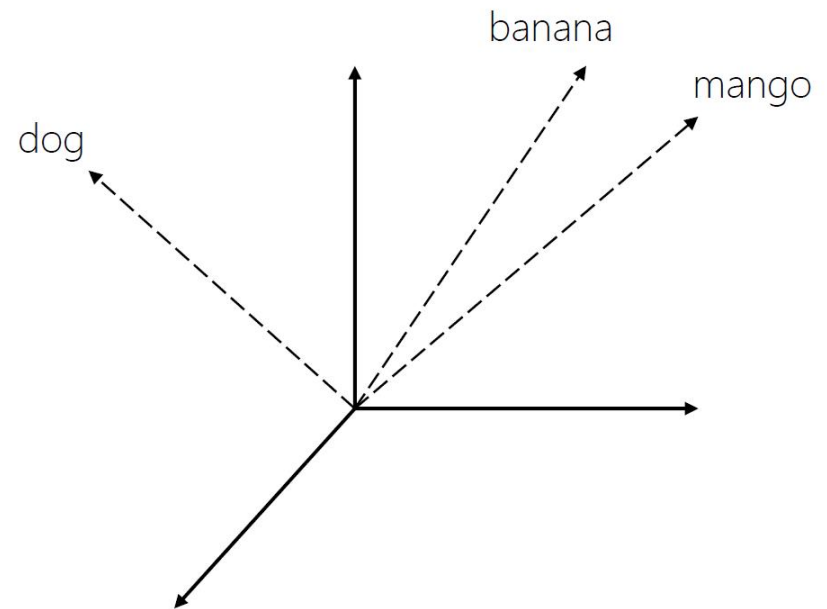
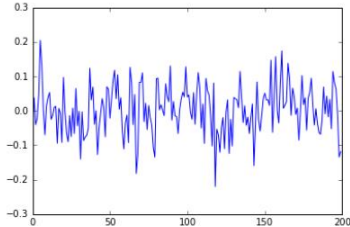
# Dense embeddings

Dense. Dim = 200 (for example)

```
In [67]: print(vec['banana'])  
plt.plot(vec['banana'])
```

```
[-0.065091, 0.037847, -0.040299, -0.022862, 0.046481, 0.204306, 0.132157, 0.000275, -0.069716, 0.014626, 0.038425, 0.053029, -  
0.024947, -0.013991, 0.010317, 0.012735, -0.094237, 0.007101, -0.007268, -0.091869, 0.097138, -0.002357, -0.065102, -0.089856,  
-0.013727, -0.074923, 0.007938, -0.066188, 0.064525, -0.0436, -0.001177, -0.140017, -0.003096, -0.086315, -0.0763, -0.071214,  
-0.051458, 0.123467, 0.031151, 0.068839, -0.039029, 4e-06, -0.127185, -0.049415, -0.007708, 0.035502, 0.009538, -0.075545, 0.0  
69583, 0.062794, -0.021556, 0.031155, 0.087352, 0.117663, 0.034883, 0.104613, 0.004534, 0.037999, -0.058016, -0.110679, -0.0353  
5, -0.012488, -0.0924, 0.126315, 0.080949, -0.040334, 0.047046, -0.182169, -0.1268, 0.082376, 0.082963, 0.110073, -0.031732, 0.  
022219, -0.054332, 0.015394, -0.019853, -0.04169, -0.106969, -0.134253, 0.093094, 0.094716, 0.002643, 0.017417, 0.00309, -0.014  
145, 0.078464, 0.041464, 0.026328, 0.12988, -0.02715, 0.027002, -0.014312, -0.017305, -0.066002, 0.002747, 0.033995, 0.053829,  
0.040628, 0.127369, 0.040216, 0.045803, -0.003395, -0.024843, 0.052411, -0.039267, 0.043378, 0.110868, 0.067947, -0.050505, 0.  
019753, -0.094825, 0.094058, 0.057547, 0.045447, -0.016258, -0.102323, 0.080506, -0.219969, -0.053595, -0.069609, -0.120579, -  
0.048799, -0.019837, -0.109987, -0.002571, 0.031825, -0.124037, -0.024646, -0.102276, 0.038512, 0.035166, 0.031713, 0.008979,  
0.114415, 0.0421, -0.034152, 0.014497, -0.04199, -0.018534, -0.065822, -0.020059, 0.019861, -0.159393, -0.03374, 0.083666, -0.  
025234, -0.058921, -0.014924, 0.035292, 0.050979, 0.031609, 0.0322, 0.015638, 0.146793, -0.062475, 0.042192, 0.157084, 0.00237  
1, -0.035507, 0.08275, 0.173776, 0.007175, 0.016044, 0.025942, 0.137863, 0.094541, -0.013125, 0.065621, 0.040823, -0.010574, 0.  
007796, -0.085031, -0.003617, 0.102267, 0.018047, 0.037613, -0.056187, 0.036693, 0.053867, 0.094616, 0.015941, -0.041536, 0.005  
796, -0.03694, -0.063241, -0.067796, -0.026023, 0.069142, -0.008786, 0.042428, -0.017718, 0.03318, -0.052277, 0.114012, 0.08154  
2, 0.063282, -0.012149, -0.134274, -0.118431]
```

```
Out[67]: [<matplotlib.lines.Line2D at 0x12a60774e48>]
```



# 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

# word2vec method

- Instead of **counting** how often each word  $w$  occurs near "apricot"
- Train a classifier on a binary **prediction** task:  
Is  $w$  likely to show up near "apricot"?
- We don't actually care about this task
- But we'll take the learned classifier weights as the word embeddings
  
- Words near apricot acts as 'correct answers' to the question "Is word  $w$  likely to show up near apricot?"
- No need for hand-labeled supervision

# word2vec (skip-gram) training data

➤ Training sentence:

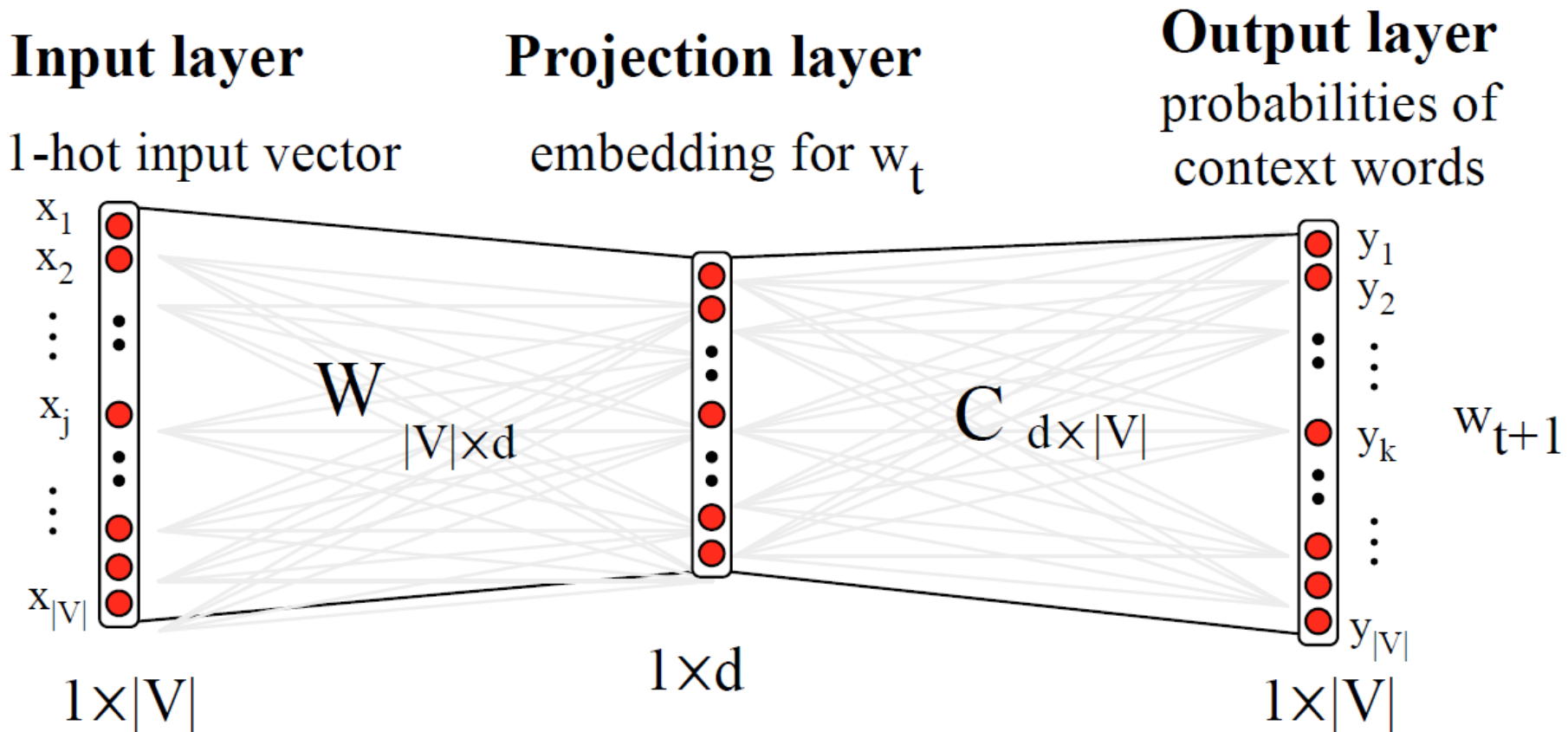
➤ ... lemon, a tablespoon of **apricot** jam a pinch ...

➤                    c1                    c2 target c3    c4

- Assume context words are those in +/- 2 word window
- Get negative training examples randomly
- train a neural network to predict probability of a co-occurring word



# Neural network based embedding



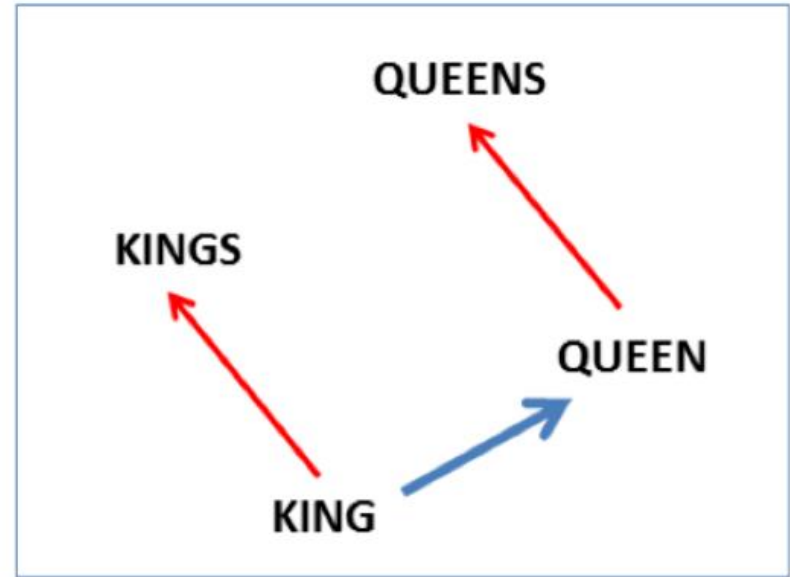
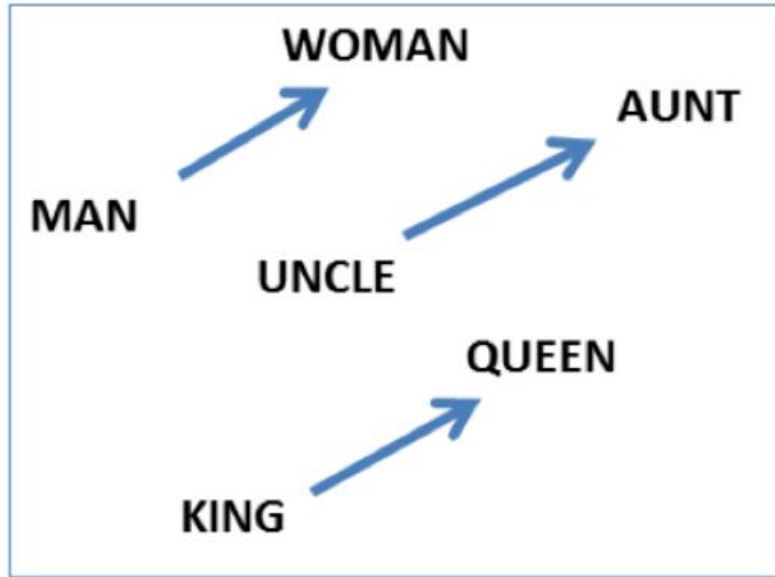
# Examples of embeddings

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

<b>target:</b>	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

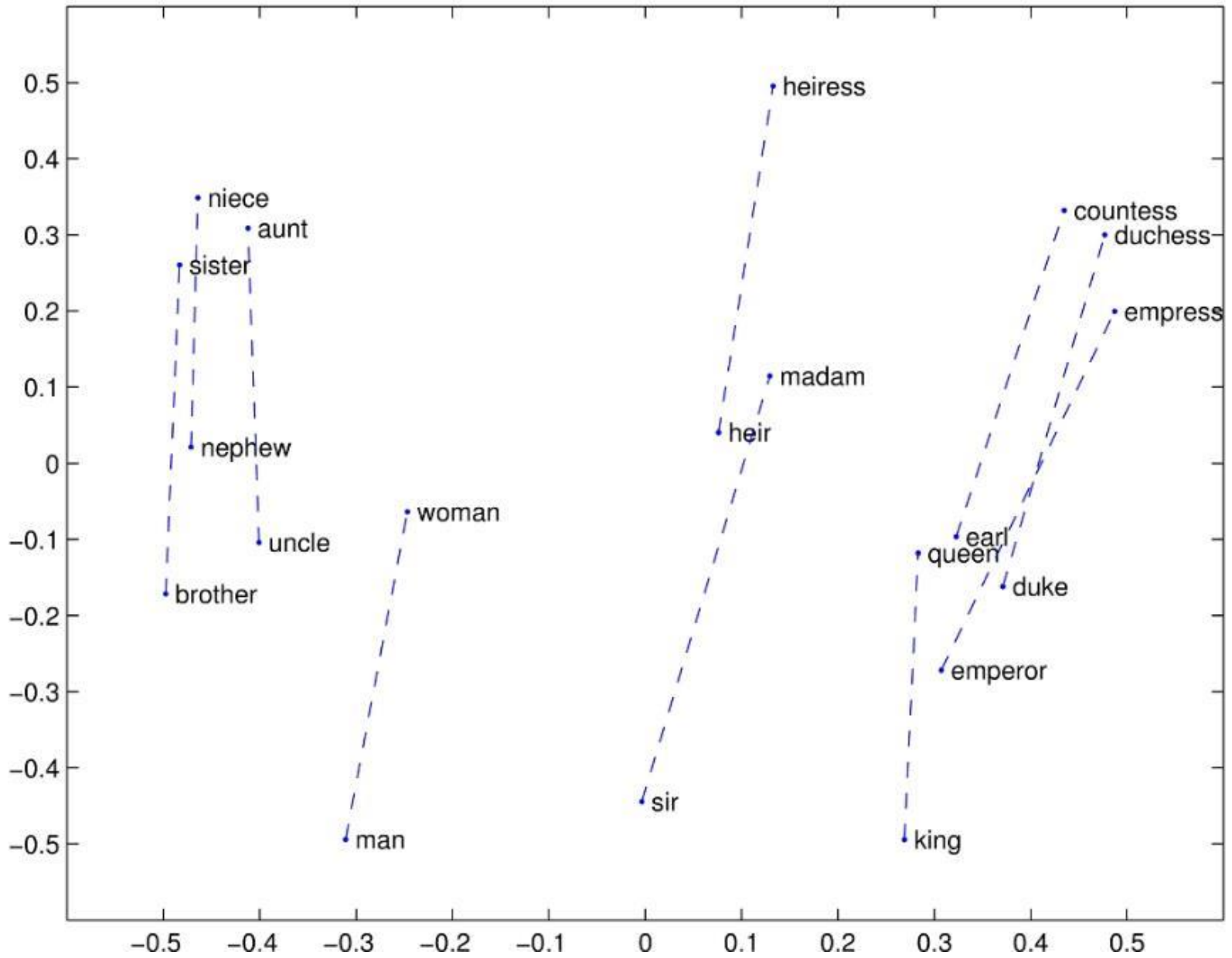
- ▶ relational similarity

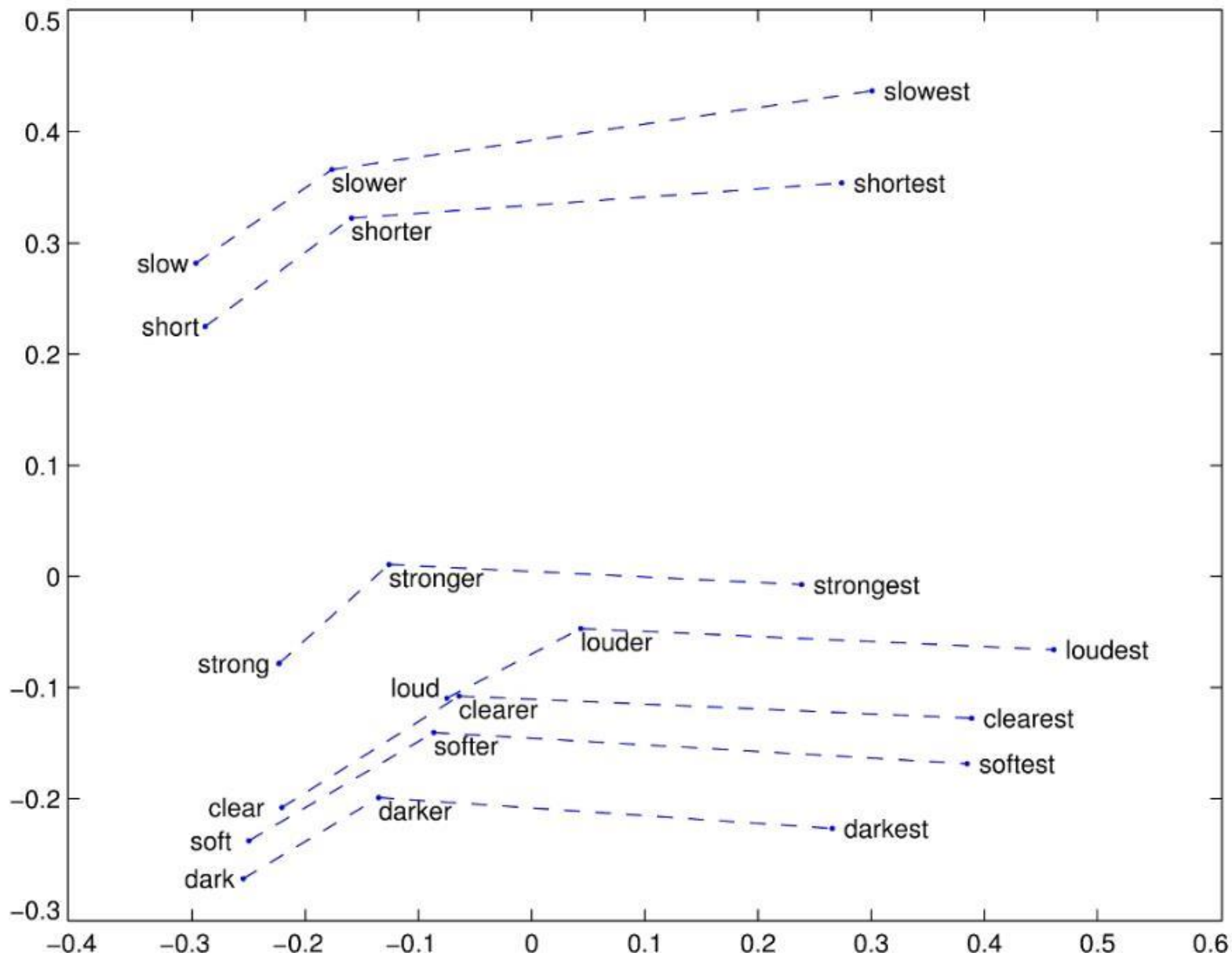
# Relational similarity



$\text{vector}('king') - \text{vector}('man') + \text{vector}('woman') \approx \text{vector}('queen')$

$\text{vector}('Paris') - \text{vector}('France') + \text{vector}('Italy') \approx \text{vector}('Rome')$





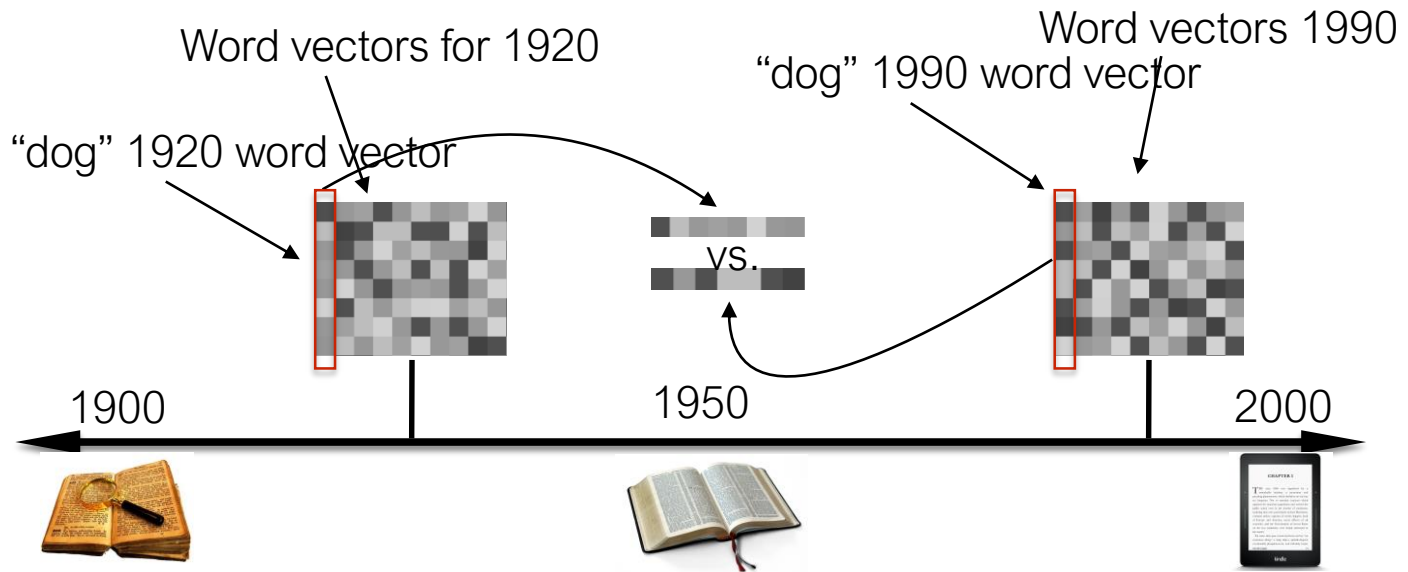
# Embeddings can help study word history

- ▶ Train embeddings on old books to study changes in word meaning!!



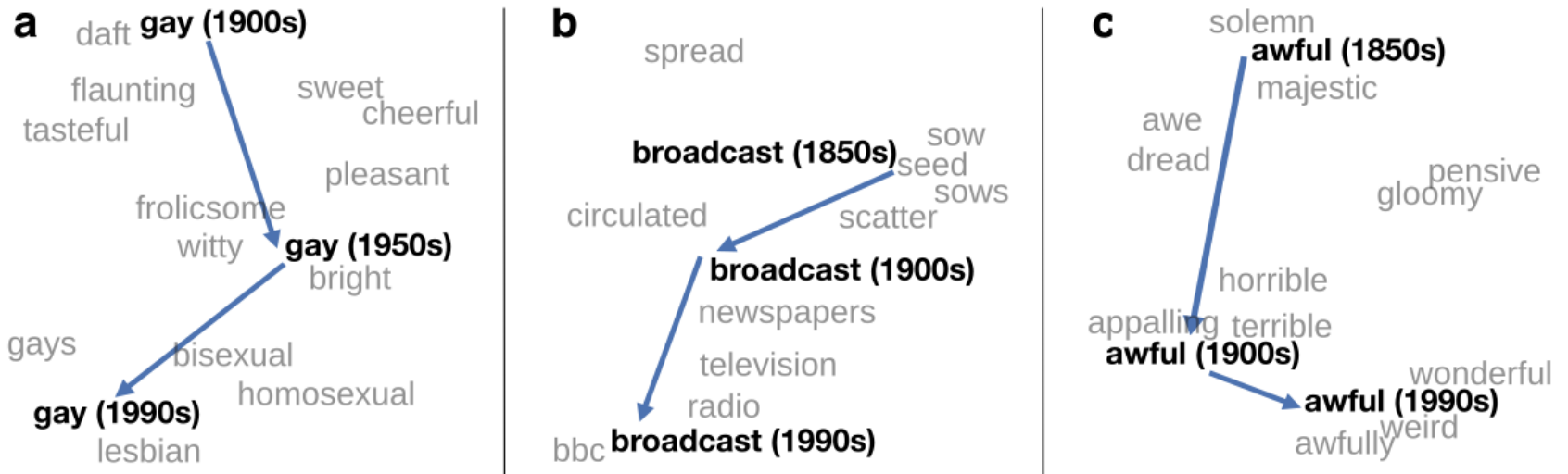
Will Hamilton

# Diachronic word embeddings for studying language change



# Visualizing changes

Project 300 dimensions down into 2

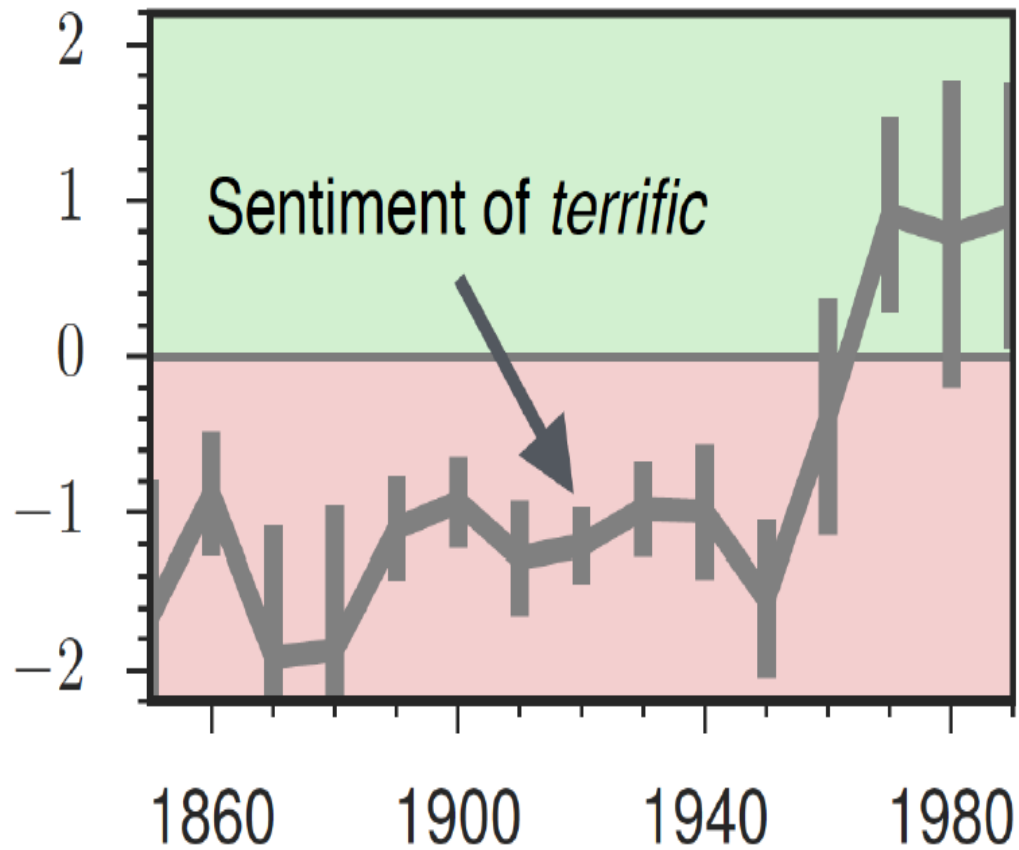


~30 million books, 1850-1990, Google Books data



# The evolution of sentiment words

Negative words change faster than positive words



# Embeddings reflect cultural bias

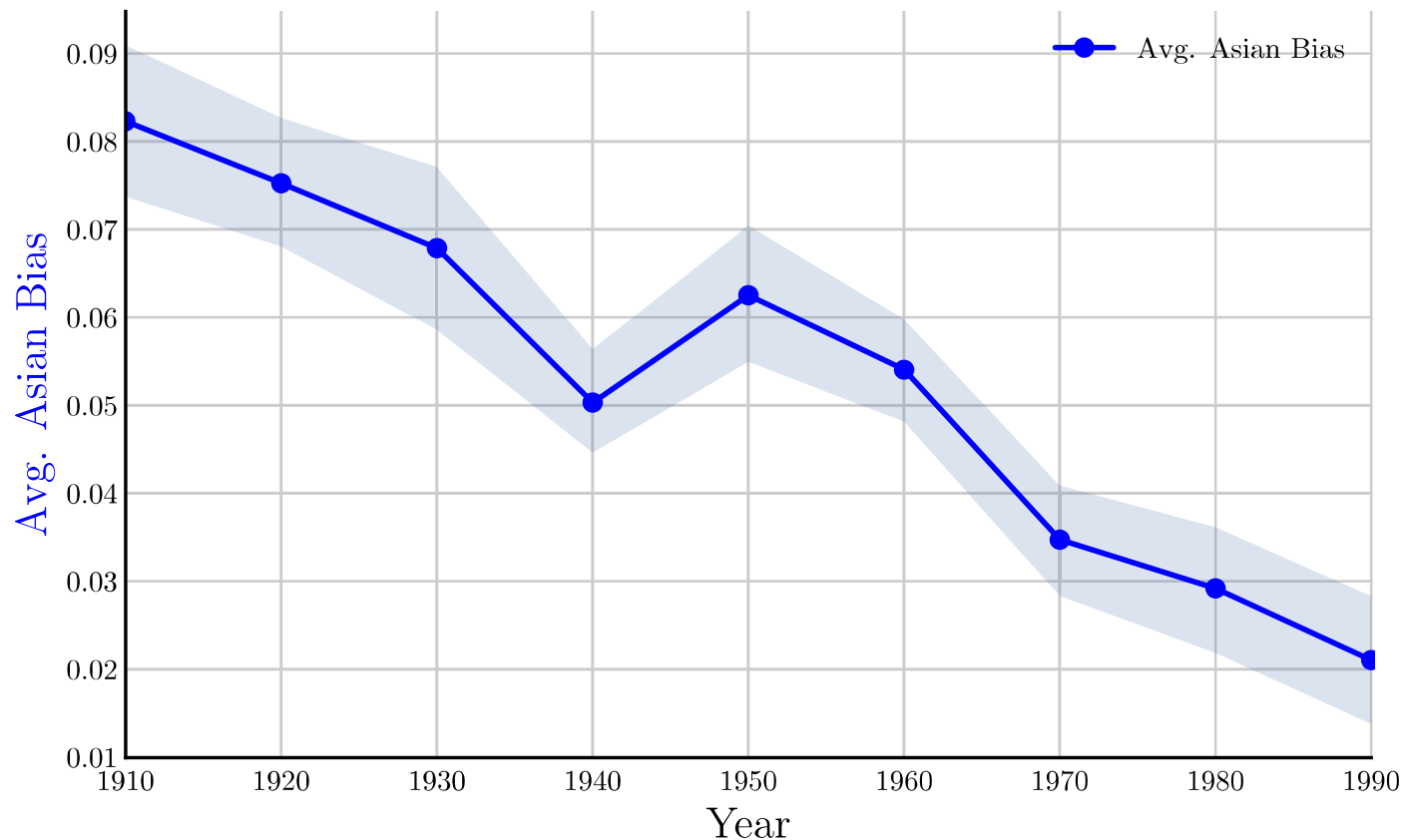
- ▶ Ask “Paris : France :: Tokyo : x”
  - ▶ x = Japan
- ▶ Ask “father : doctor :: mother : x”
  - ▶ x = nurse
- ▶ Ask “man : computer programmer :: woman : x”
  - ▶ x = homemaker

# Embeddings reflect cultural bias

- ▶ Implicit Association test (Greenwald et al 1998): How associated are
  - ▶ concepts (*flowers, insects*) & attributes (*pleasantness, unpleasantness*)?
  - ▶ Studied by measuring timing latencies for categorization.
- ▶ Psychological findings on US participants:
  - ▶ African-American names are associated with unpleasant words (more than European-American names)
  - ▶ Male names associated more with math, female names with arts
  - ▶ Old people's names with unpleasant words, young people with pleasant words.
- ▶ Caliskan et al. replication with embeddings:
  - ▶ African-American names (*Leroy, Shaniqua*) had a higher GloVe cosine with unpleasant words (*abuse, stink, ugly*)
  - ▶ European American names (*Brad, Greg, Courtney*) had a higher cosine with pleasant words (*love, peace, miracle*)
- ▶ Embeddings reflect and replicate all sorts of pernicious biases.

# Change in linguistic framing 1910-1990

Change in association of Chinese names with adjectives framed as "othering" (*barbaric, monstrous, bizarre*)



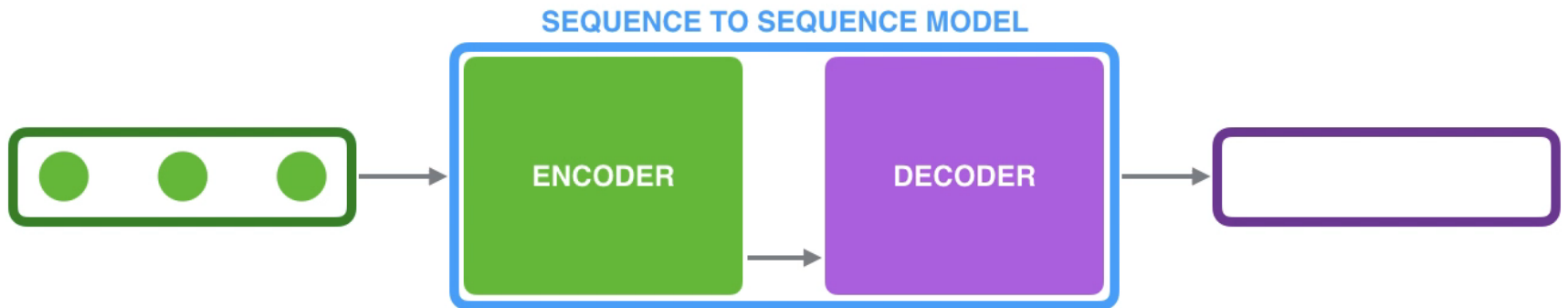
# Contextual embeddings

- word2vec produces the same vector for a word like bank irrespective of its meaning and context
- recent embeddings take the context into account
- already established as a standard
- ELMo and BERT

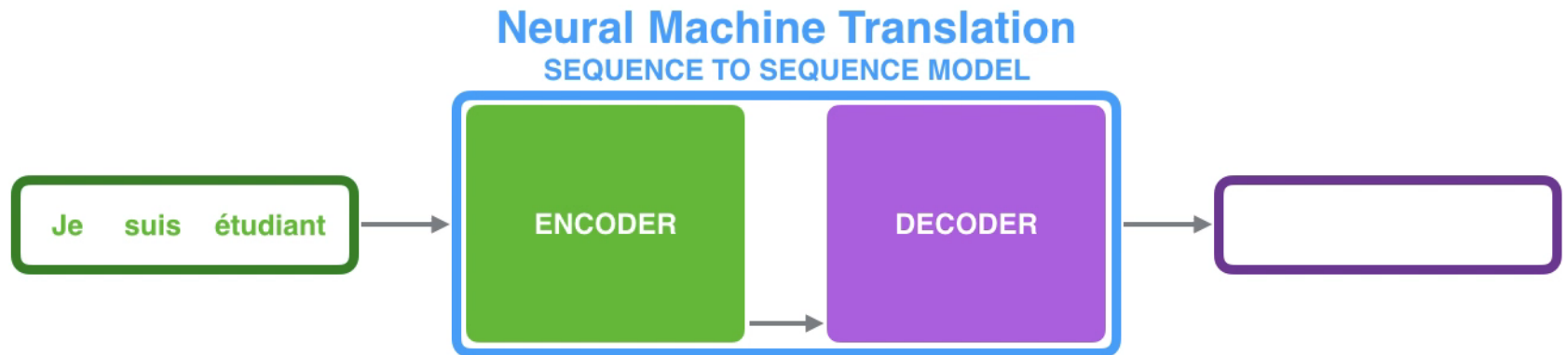
# ELMo

- ELMo looks at the entire sentence before assigning each word in it an embedding.
- ELMo predicts the next word in a sequence of words - a task called *Language Modeling*.
- It uses a bi-directional LSTM recurrent neural network
- includes subword units
- as an embedding, ELMo uses several layers of the network
- first layers capture morphological and syntactic properties, deeper layers encode semantical properties
- uses several fine tuned parameters
- publicly available for many languages

# Encoder-decoder for sequences



# Encoder-decoder for NMT



CONTEXT

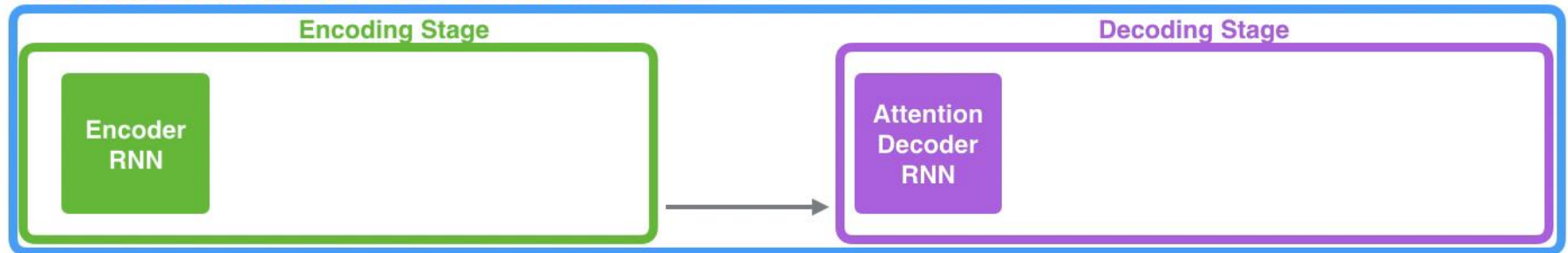
0.11
0.03
0.81
-0.62

0.11
0.03
0.81
-0.62



# NMT with attention

## Neural Machine Translation SEQUENCE TO SEQUENCE MODEL WITH ATTENTION



Je

suis

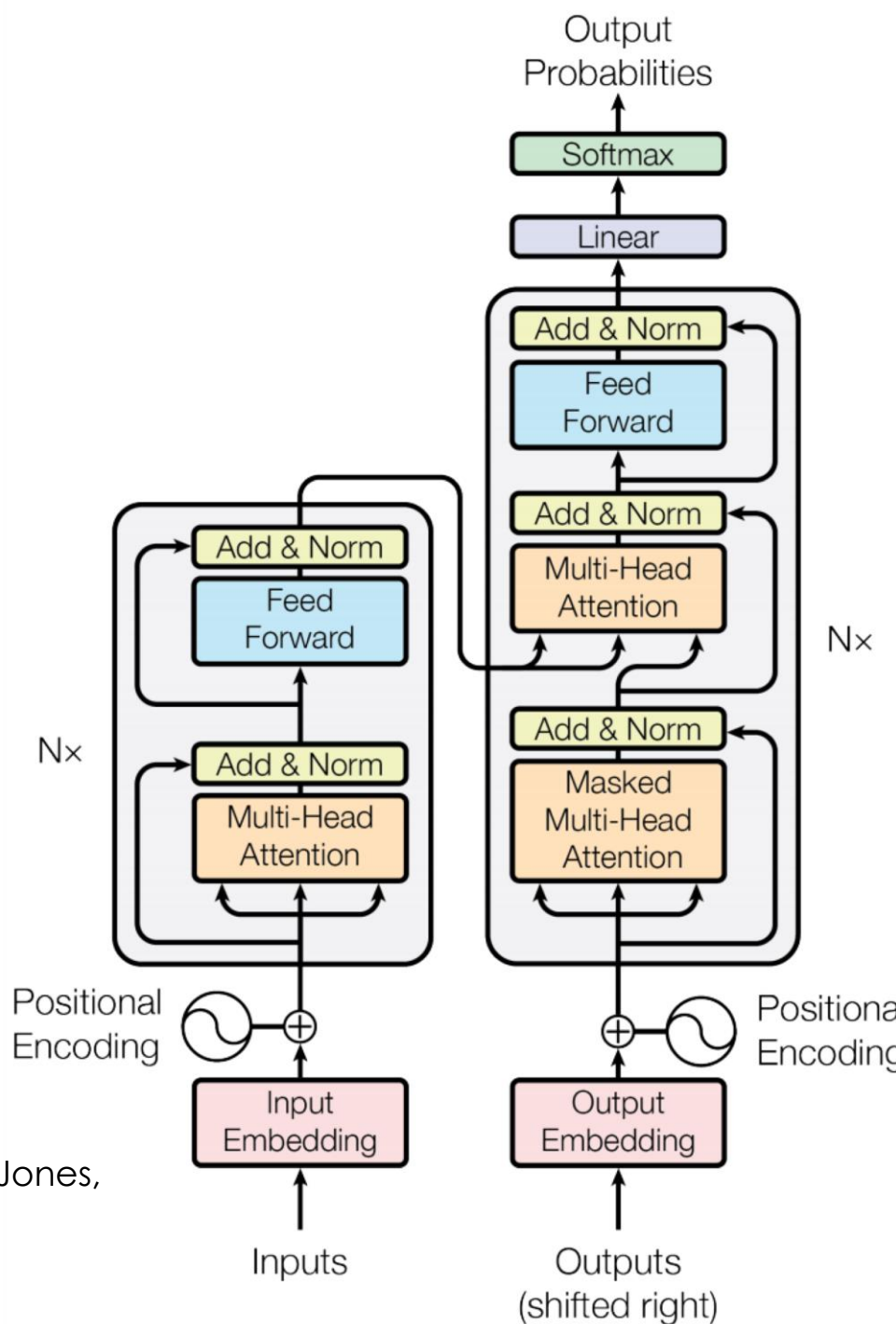
étudiant

# Transformer model

- currently the most successful DNN
- non-recurrent
- architecturally it is an encoder-decoder model
- fixed input length
- can be parallelized
- adapted for GPU (TPU) processing
- based on extreme use of attention
  
- <https://github.com/dair-ai/Transformers-Recipe>

# Transformer overview

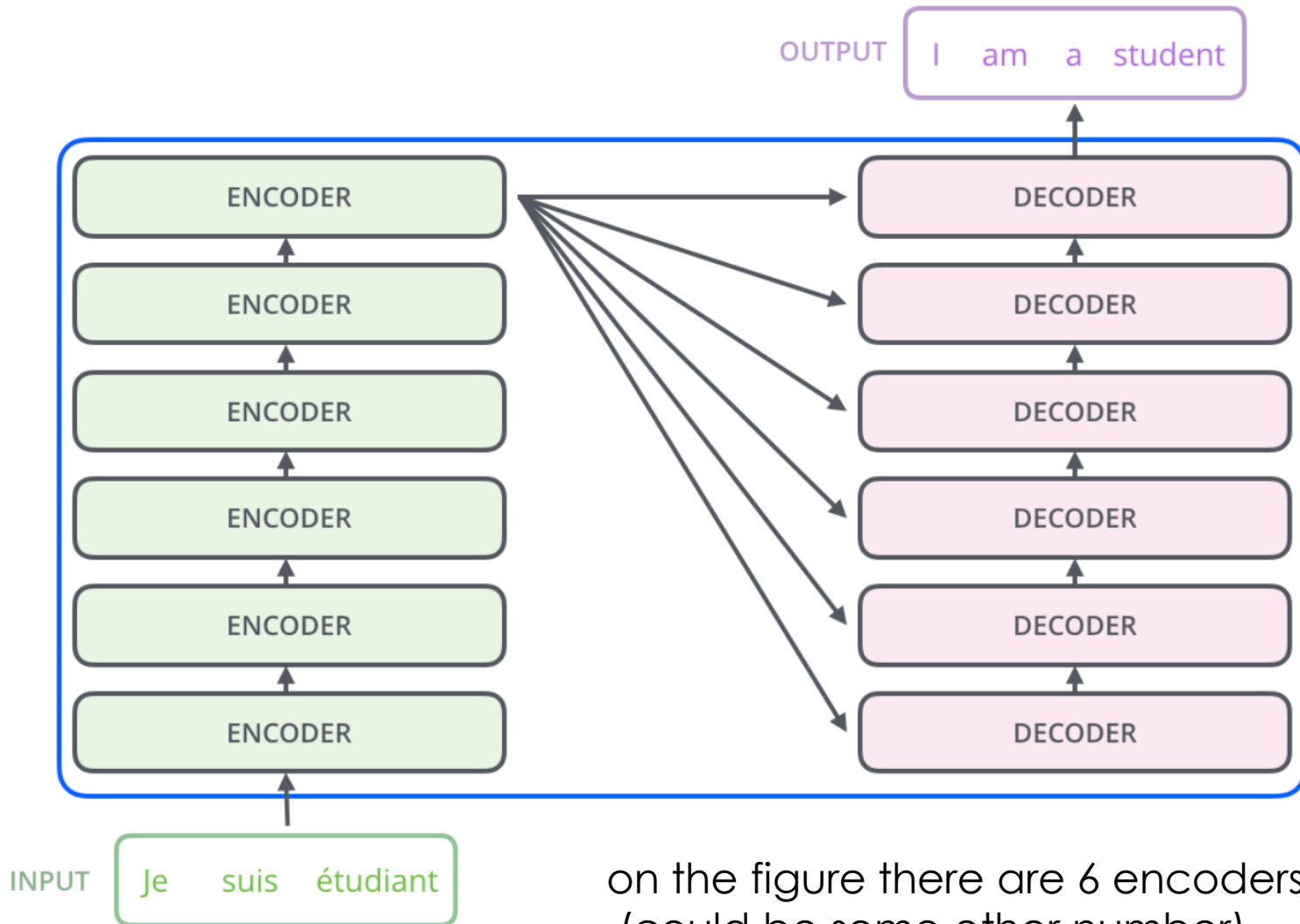
- Initial task: machine translation with parallel corpus
- Predict each translated word
- Final cost/loss/error function is standard cross-entropy error on top of a softmax classifier



Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł. and Polosukhin, I., 2017.

[Attention is all you need](#). In *Advances in neural information processing systems* (pp. 5998-6008).

# Transformer is encoder-decoder model



on the figure there are 6 encoders and 6 decoders  
(could be some other number)

# BERT

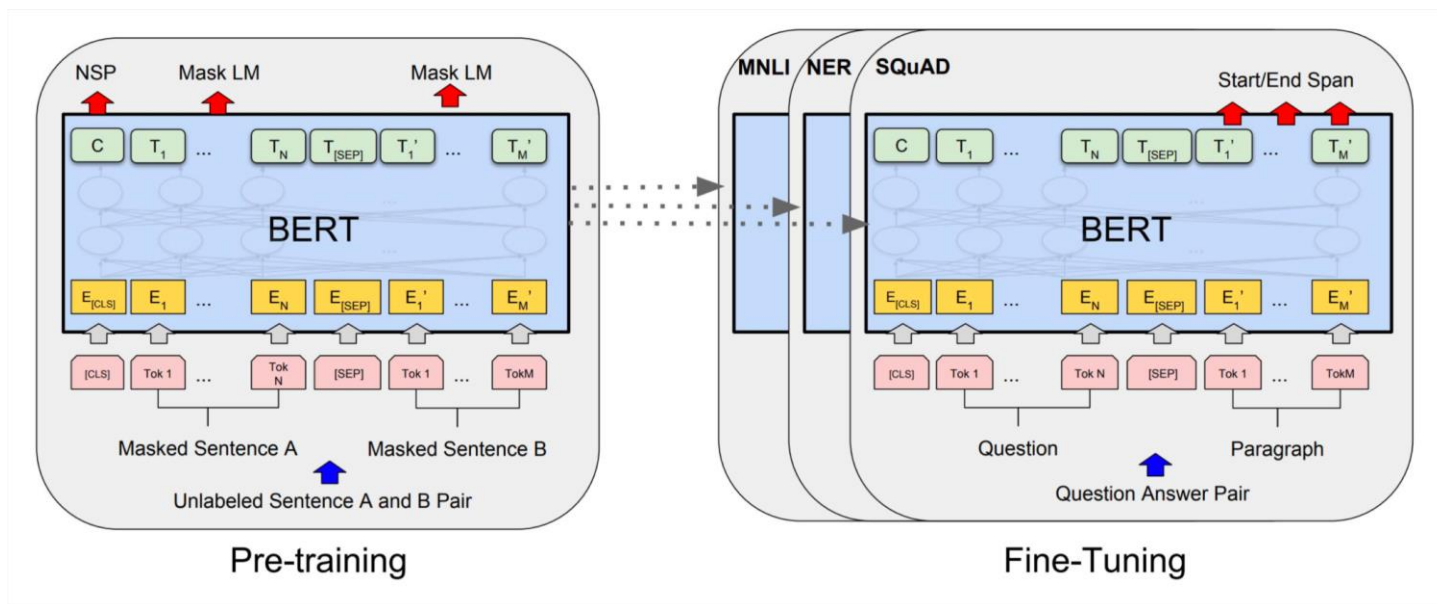
- combines several tasks
- predicts masked words in a sentence
- also predicts order of sentences: is sentence A followed by sentence B or not
- combines several hidden layers of the network
- uses transformer neural architecture
- uses several fine tuned parameters
- multilingual variant supports 104 languages by training on Wikipedia
- publicly available

# Existing embeddings

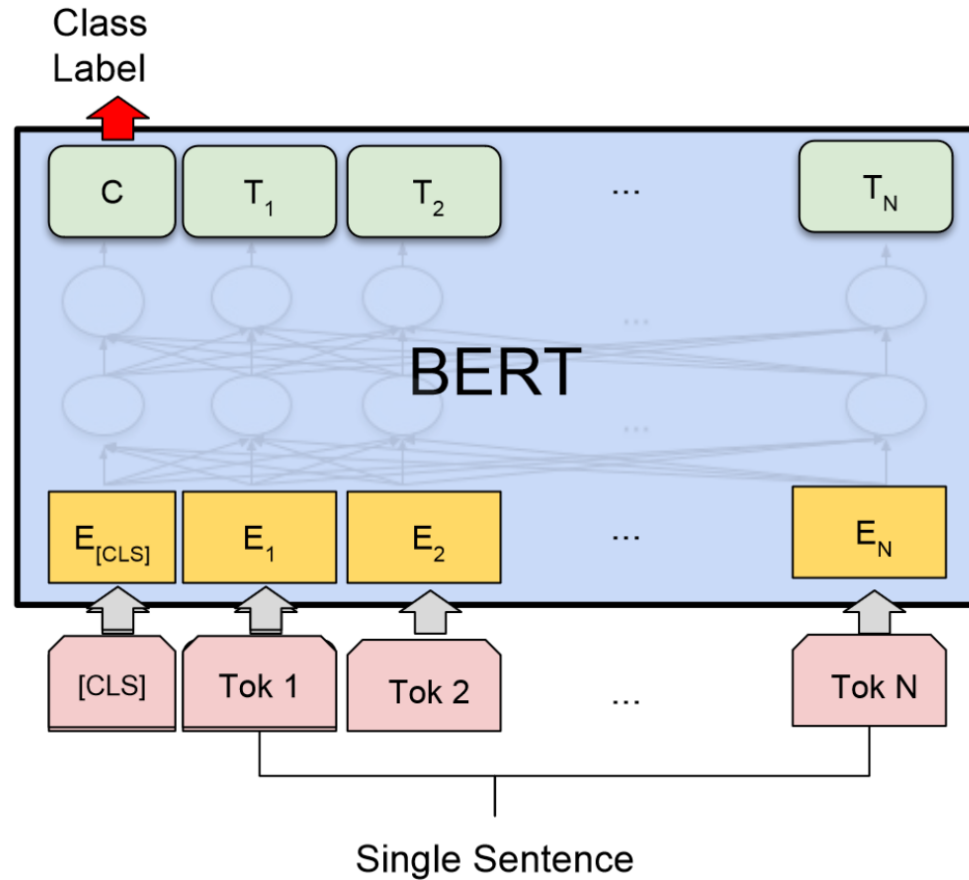
- recent XLM-R was trained on 2.5 TB of texts in 100 languages
- for Slovene: fastText, ELMo, SloBERTa
- trilingual BERT – CroSloEngual
- on Clarin.si
- more to follow: hundreds of papers investigating BERT-like models in major ML & NLP conferences

# Use of BERT

- ▶ train a classifier built on the top layer for each task that you fine tune for, e.g., Q&A, NER, inference
- ▶ achieves state-of-the-art results for many tasks
- ▶ GLUE and SuperGLUE tasks for NLI

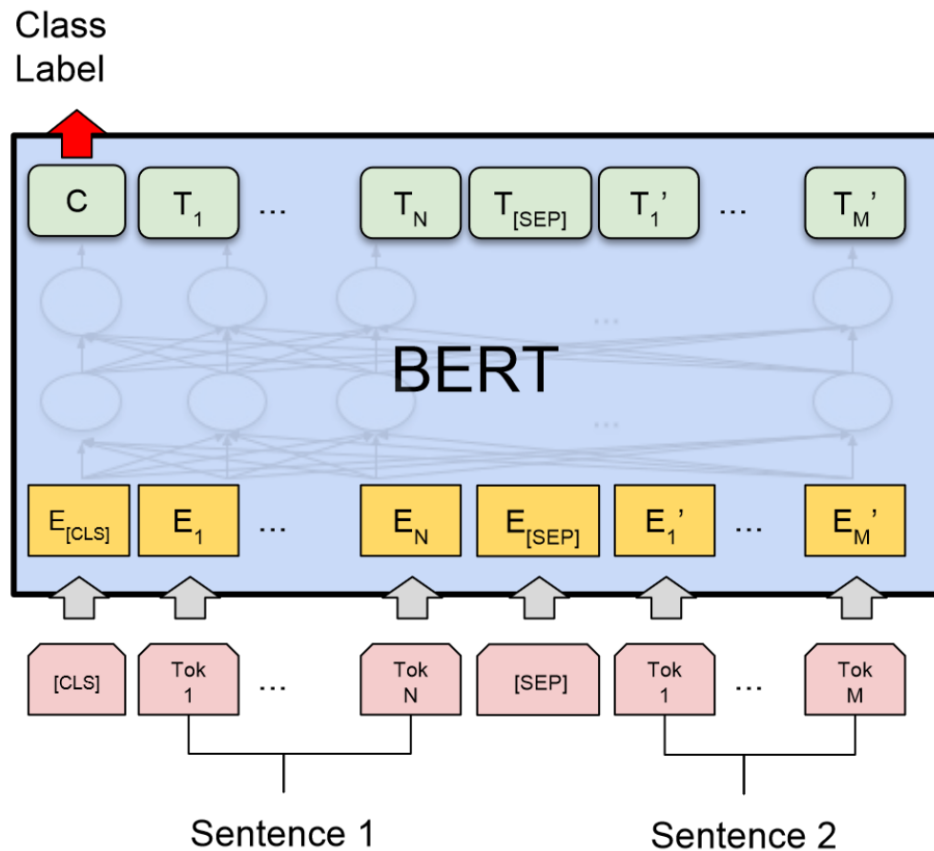


# Sentence classification using BERT – sentiment, grammatical correctness

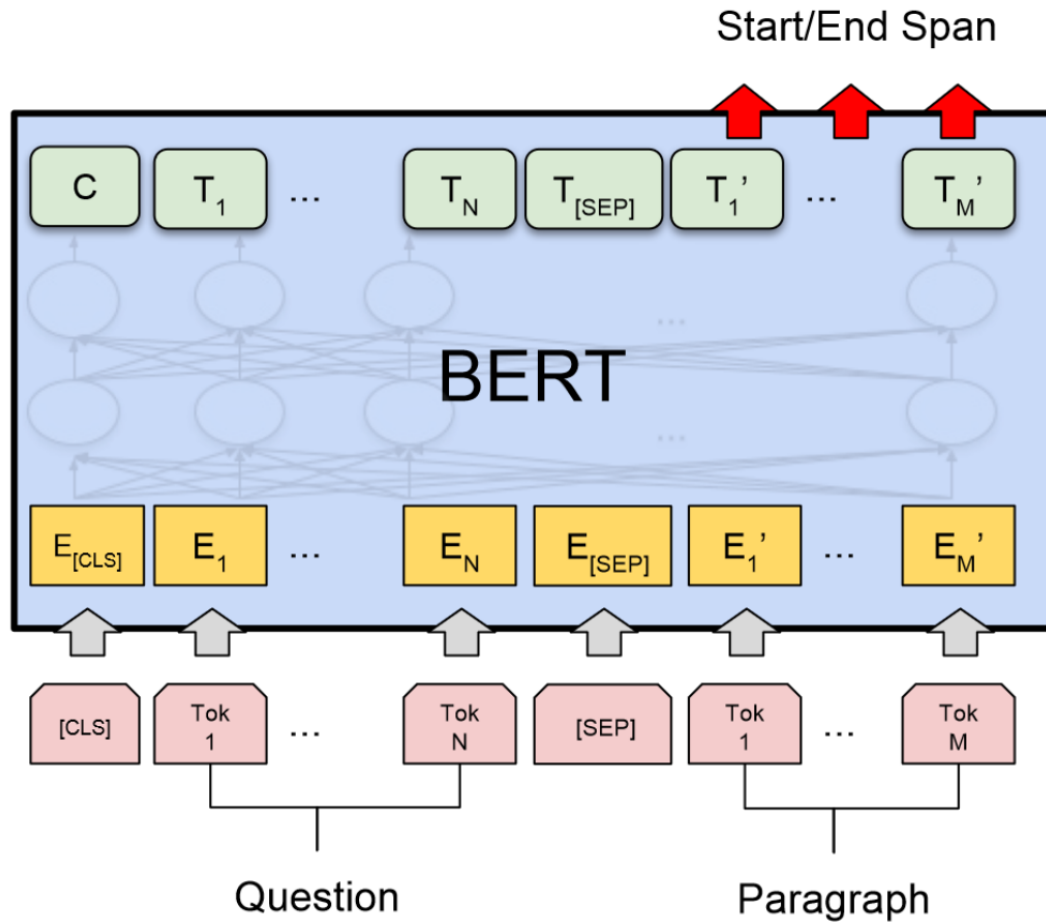




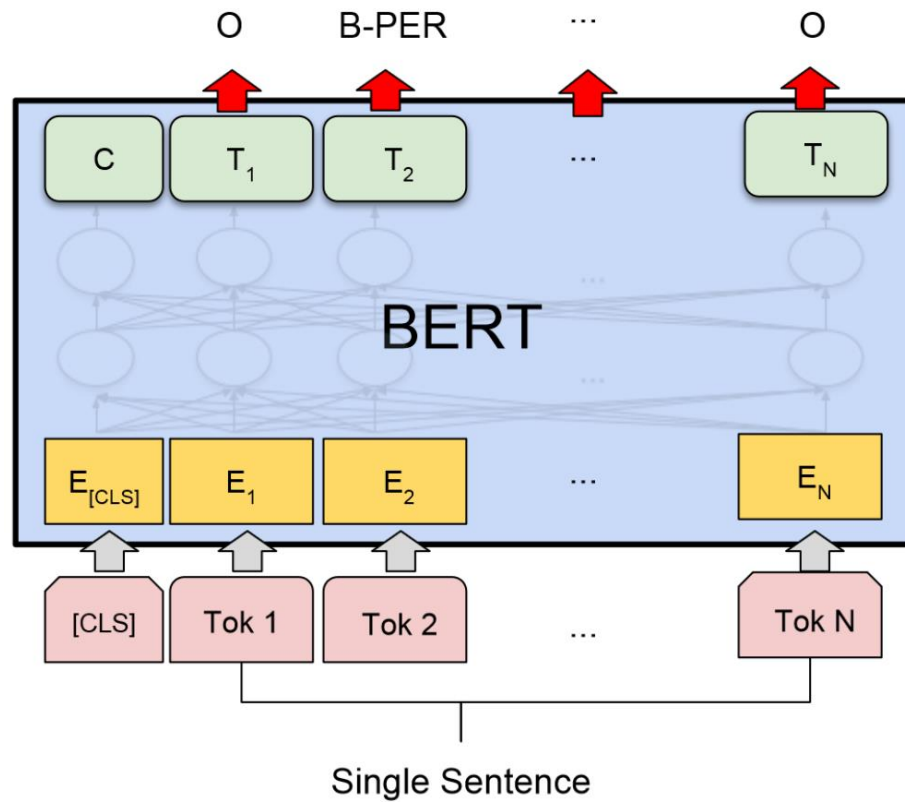
# Two sentence classification using BERT-inference



# Questions and answers with BERT



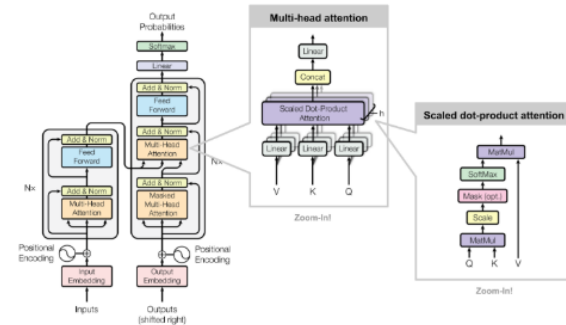
# Sentence tagging with BERT- NER, POS tagging, SRL





# Pretrained language models (PLMs)

- huge pretrained neural language models
- trained on large text corpora to capture relations in language
- finetuned to specific tasks
- publicly available
- on Clarin.si and HuggingFace
- hundreds of papers investigating BERT-like models in major ML & NLP conferences



- Ulčar, M., & Robnik-Šikonja, M. (2020). High Quality ELMo Embeddings for Seven Less-Resourced Languages. In *Proceedings of the 12th Language Resources and Evaluation Conference* (pp. 4731-4738).



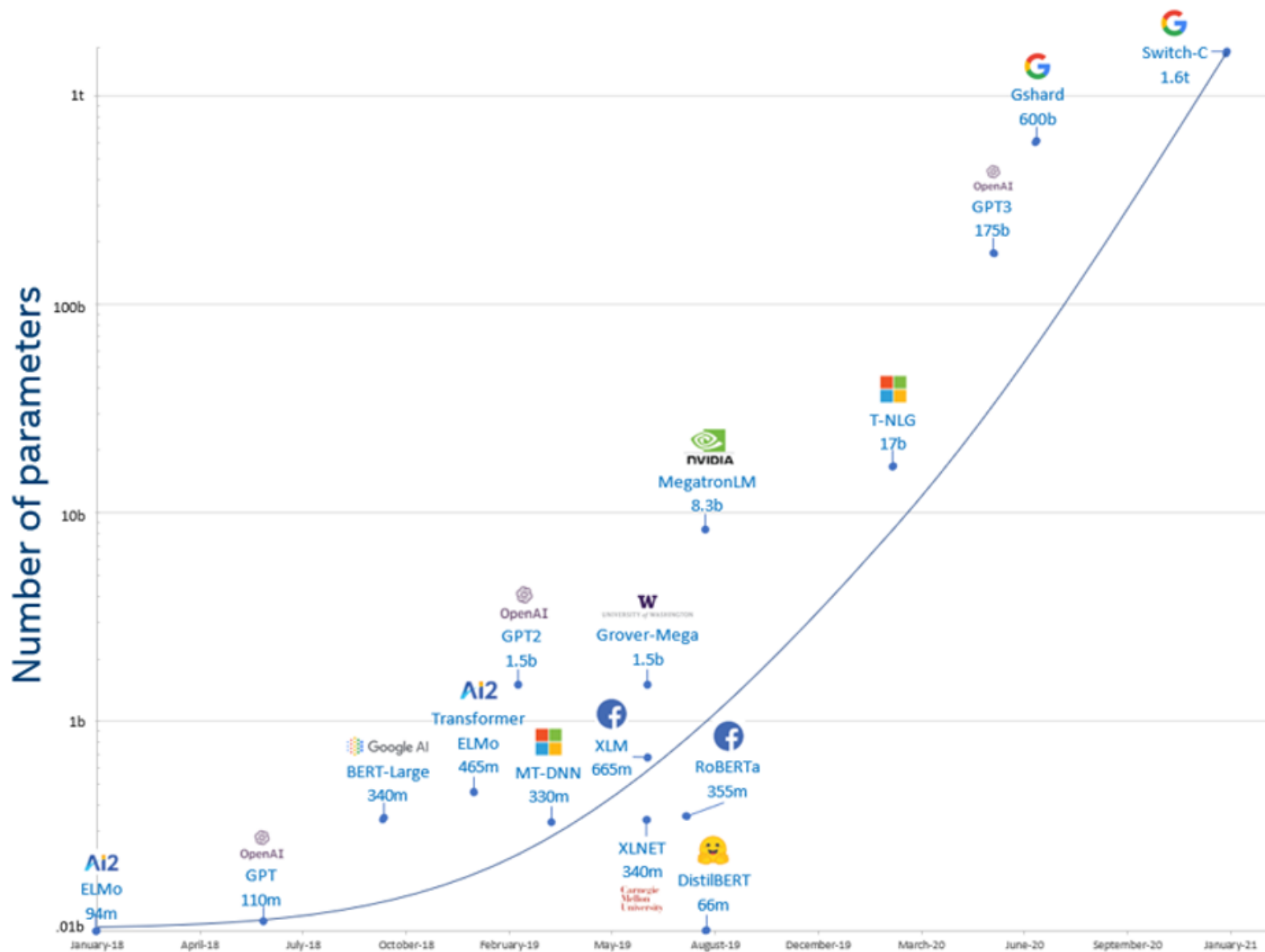
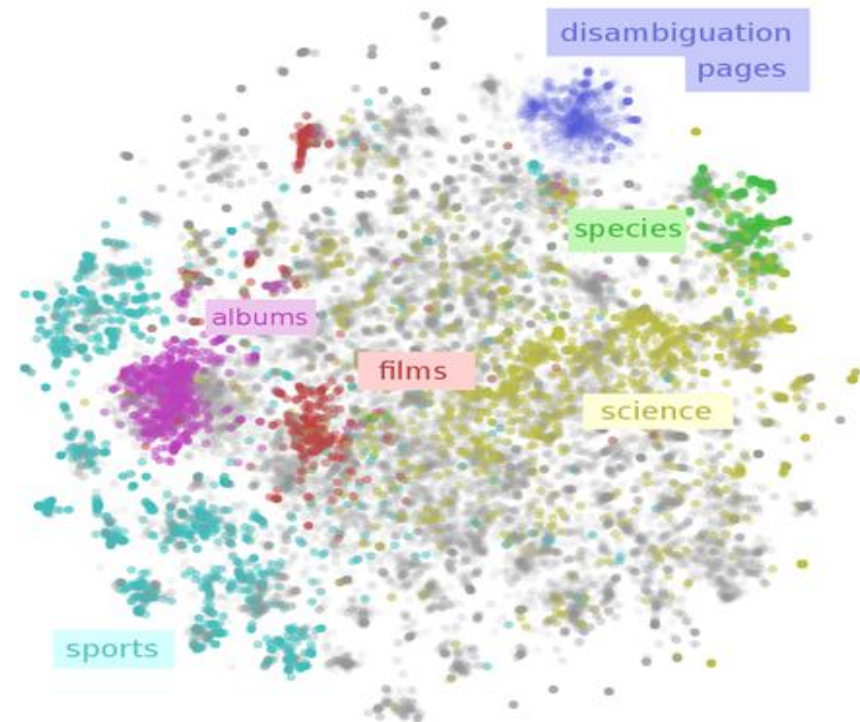


Figure 1: Exponential growth of number of parameters in DL models

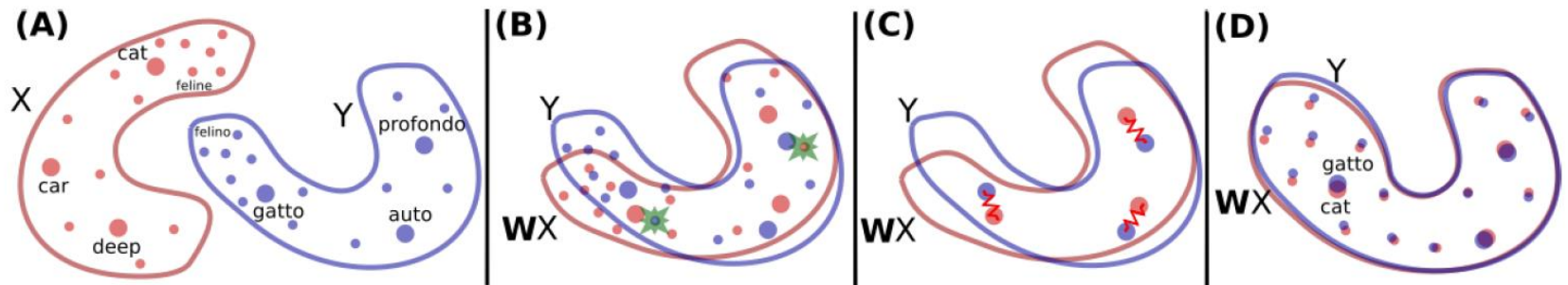
# Cross-lingual embeddings

- embeddings are trained on monolingual resources
- words of one language form a cloud in high dimensional space
- clouds for different languages can be aligned
  - ➔  $W_S \approx E$  or
  - ➔  $W_1S \approx W_2E$



# Cross-lingual embeddings

- allignment of different word clouds



- in unsupervised or supervised way

Conneau, A., Lample, G., Ranzato, M.A., Denoyer, L. and Jégou, H., 2018. Word translation without parallel data. Proceedings of ICLR 2018, also *arXiv preprint arXiv:1710.04087*.

# Improving cross-lingual embeddings

- bilingual and multilingual resources can provide anchoring points for alignment of different word clouds
- alignment of contextual embeddings
- transfer between languages: models, resources
- embedded words enter neural networks
- replace them with cross-lingual embeddings and easily switch languages

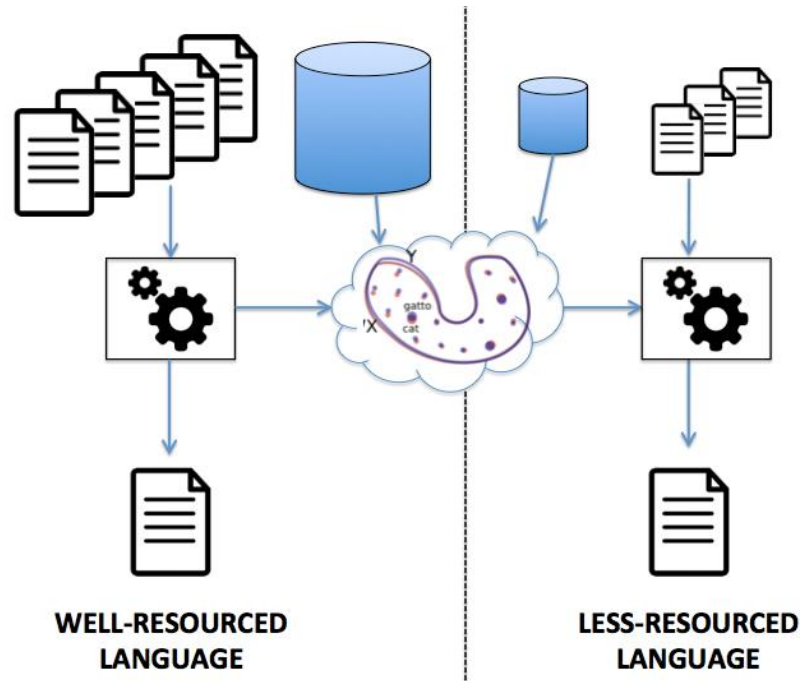
Artetxe, M. and Schwenk, H., 2018. Massively Multilingual Sentence Embeddings for Zero-Shot Cross-Lingual Transfer and Beyond. *arXiv preprint arXiv:1812.10464*.





# Cross-lingual model transfer based on embeddings

- Transfer of tools trained on mono-lingual resources

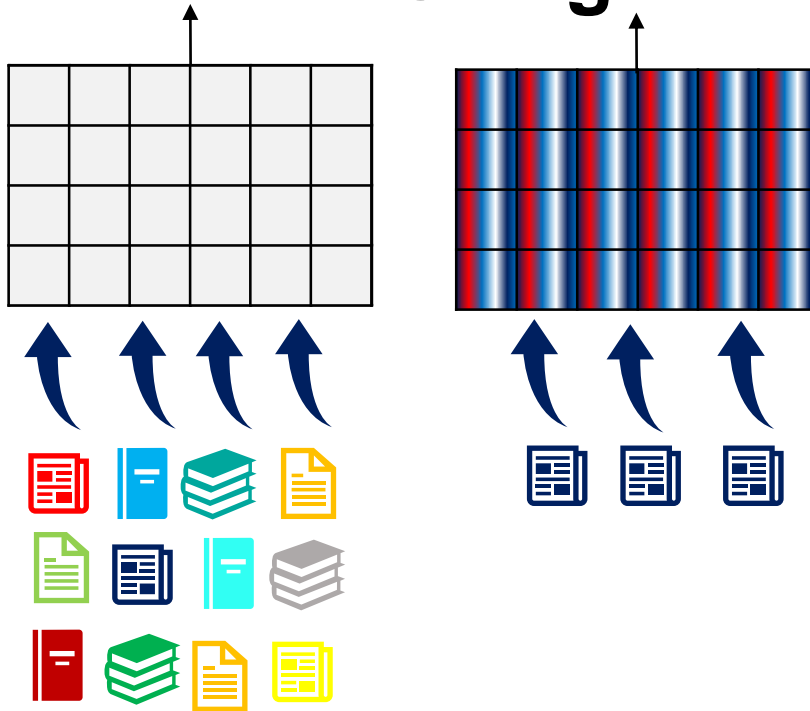




# Multilingual PLMs

- Pretrained on multiple languages simultaneously
- multilingual BERT supports 104 languages by training on Wikipedia
- XLM-R was trained on 2.5 TB of texts
- allow cross-lingual transfer
- solve problem of insufficient training resources for less-resourced languages

# Using multilingual models



Pretraining

Fine-tuning

Classification

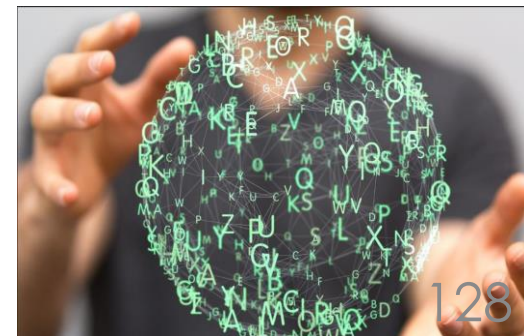
predsjednik je danas najavio ...

Zero-shot transfer and few-shot transfer



# Embed all the things!

- Neural networks require numeric input
- Embedding shall preserve relations from the original space
- Representation learning is a crucial topic in nowadays machine learning
- Lots of applications whenever enough data is available to learn the representation
- In text, BERT-like models rule
- Similar ideas applied to texts, speech, graphs, electronic health records, relational data, time series, etc.



# Semantic language technologies

- ▶ Also called text mining; to acquire new knowledge
- ▶ Summarization, document relations, clustering of documents, new topic detection, related news, directory of important people/institutions, taxonomies, questions & answers  
named-entity recognition/disambiguation/linking, inference, coreferences resolution

# 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
- ▶ deep neural networks

# 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, company's Facebook pages

## ➤ Data cleaning

- quality assessment
- Preprocessing: tokenization, stop word removal, stemming, parts of speech (POS) tagging, and feature extraction/representation/selection
- tokenization for DNNs



# 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

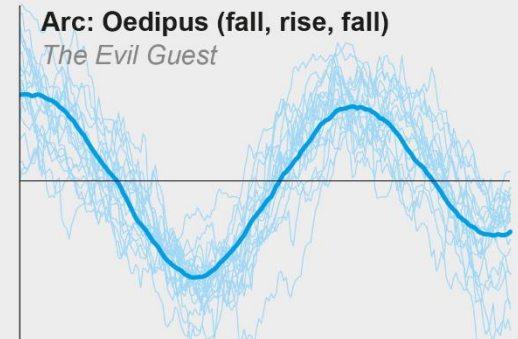
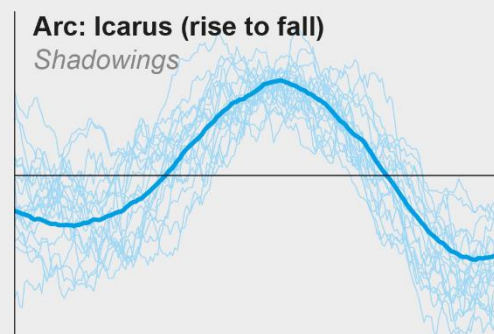
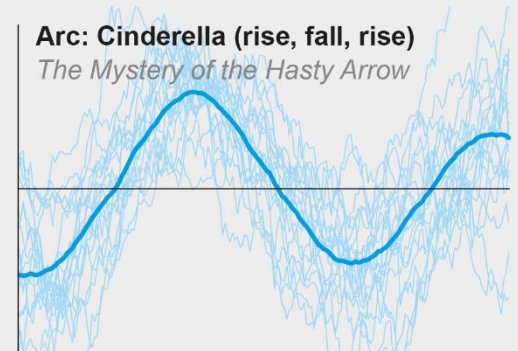
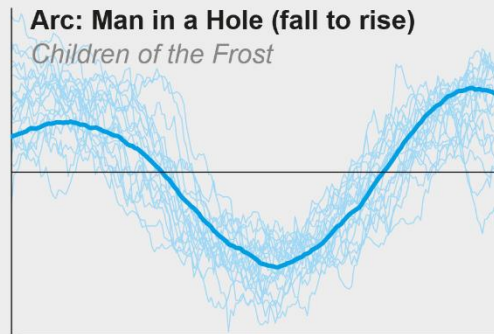
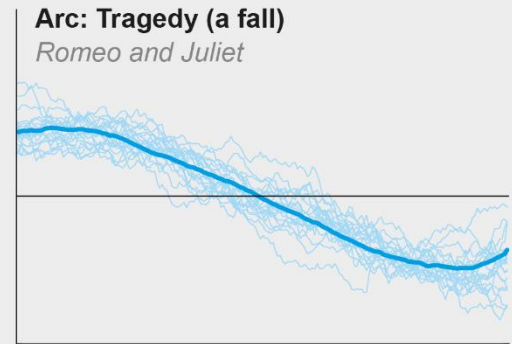
# Other SA tasks

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

# Emotional states in English fiction

## Emotional Arcs

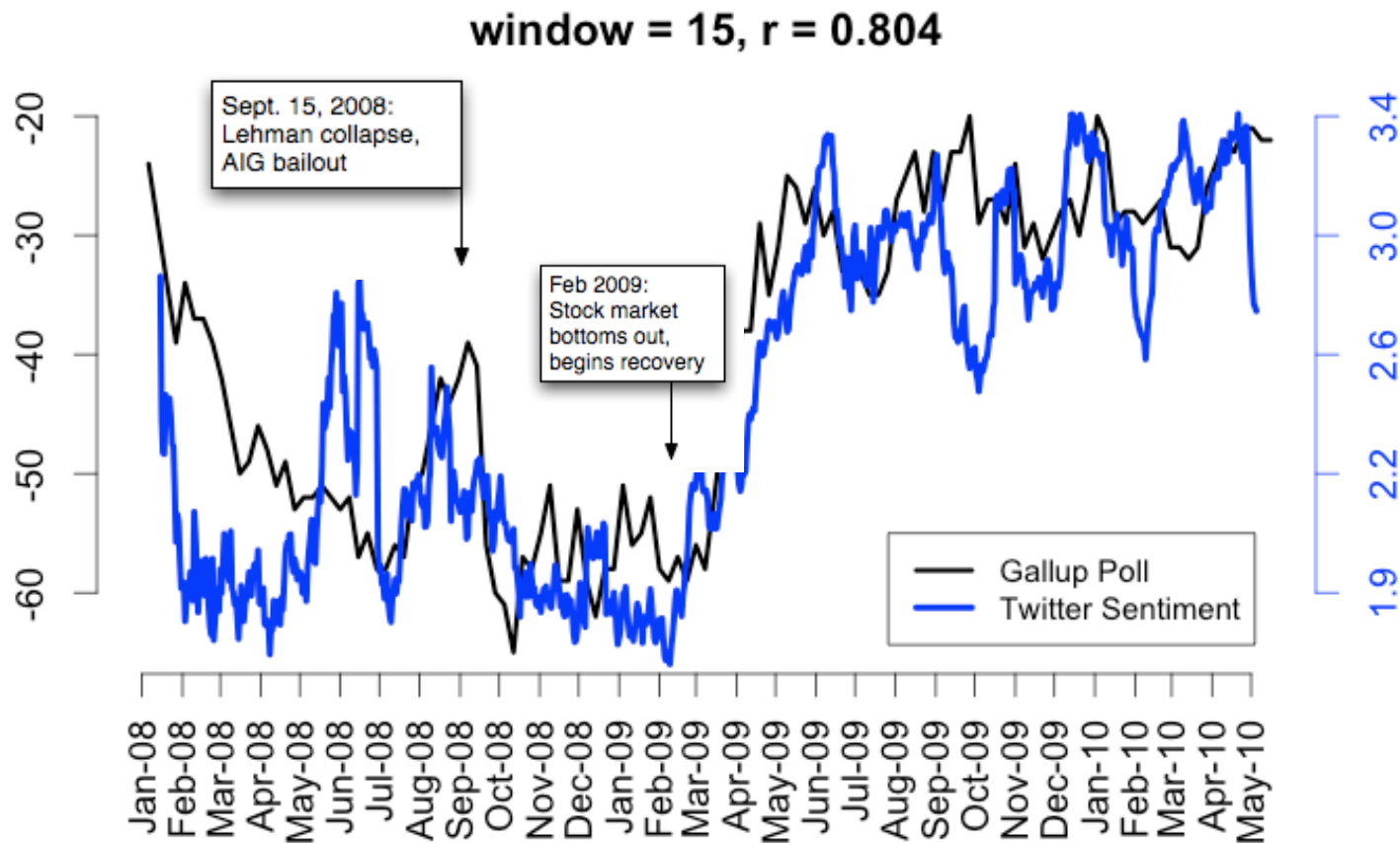
About 85 percent of 1,327 fiction stories in the digitized Project Gutenberg collection follow one of six emotional arcs—a pattern of highs and lows from beginning to end (*dark curves*). The arcs are defined by the happiness or sadness of words in the running text (*jagged plots*). All books were in English and less than 100,000 words; examples are noted.



# Public opinion surveys

## Twitter sentiment vs. Gallup on consumer sentiment

Brendan O'Connor, Ramnath Balasubramanian, Bryan R. Routledge, and Noah A. Smith. 2010. From Tweets to Polls: Linking Text Sentiment to Public Opinion Time Series. In ICWSM-2010



# Statistical machine translation

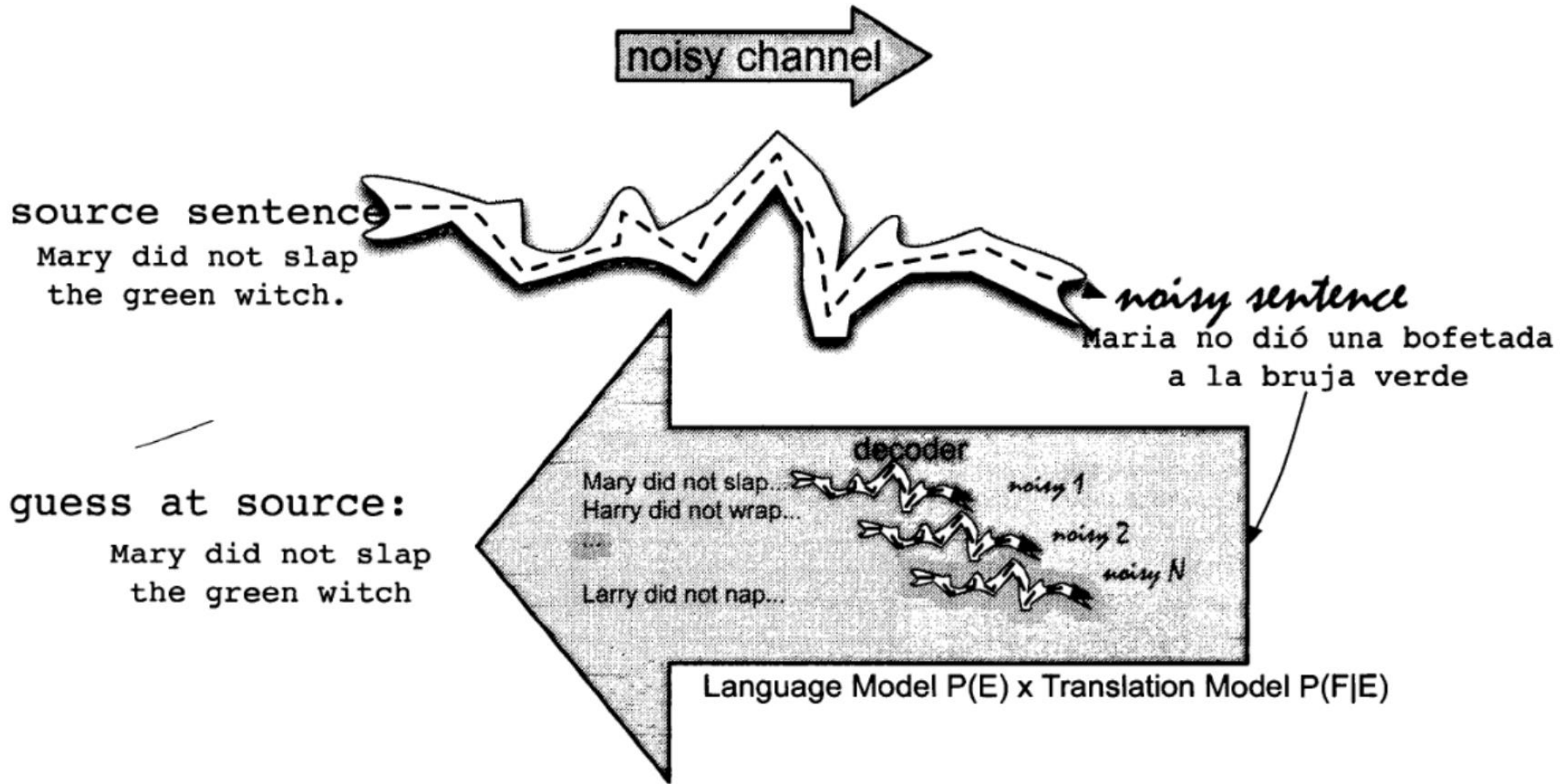
- idea from the theory of information
- we translate from foreign language F to English E
- a document is translated based on the probability distribution  $p(e | f)$ , i.e. the probability of the sentence e in target language based on the sentence in source language f
- Bayes rule
$$\arg \max_e p(e | f) = \arg \max_e p(f | e) p(e) / p(f)$$
- $p(f)$  can be ignored as it is a constant for a given fixed sentence
- we split the problem into subproblems
  - create a language model  $p(e)$
  - a separate translation model  $p(f | e)$
  - decoder which forms the most probable e based on f

# Noisy channel model

- ▶ given English sentence  $e$
- ▶ during transmission over a noisy channel the sentence  $e$  is corrupted and we get sentence in a foreign language  $f$
- ▶ to reconstruct the most probable sentence  $e$  we have to figure out:
  - ▶ how people speak in English (language model),  $p(e)$  and
  - ▶ how to transform foreign language into English (translation model),  $p(f | e)$

# Noisy channel

► reasoning back



# Language model

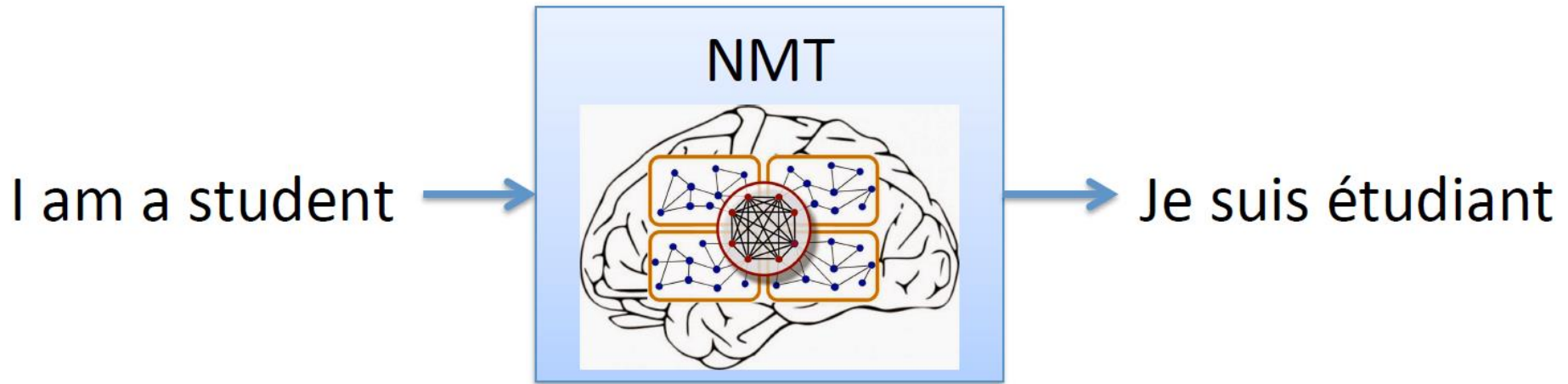
- ▶ each target (English) sentence  $e$  is assigned a probability  $p(e)$
- ▶ estimation of probabilities for the whole sentences is not possible (why?), therefore we use language models, e.g., 3-gram models or neural language models



# Translation model

- ▶ we have to assign a probability of  $p(f | e)$ , which is a probability of a foreign language sentence  $f$ , given target sentence  $e$ .
- ▶ we search the  $e$  which maximizes  $p(e) * p(f | e)$
- ▶ traditional MT approach: using translation corpus we determine which translation of a given word is the most probable
- ▶ we take into account the position of a word and how many words are needed to translate a given word

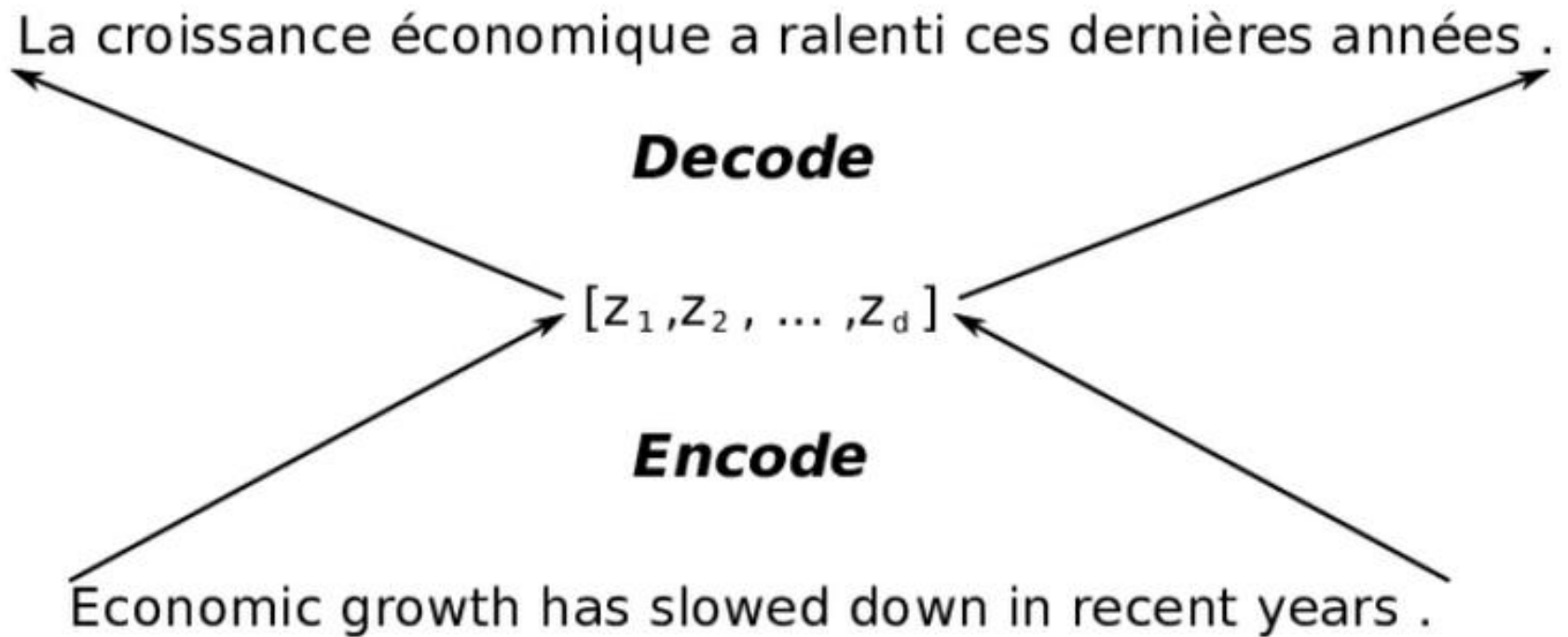
# Neural machine translation



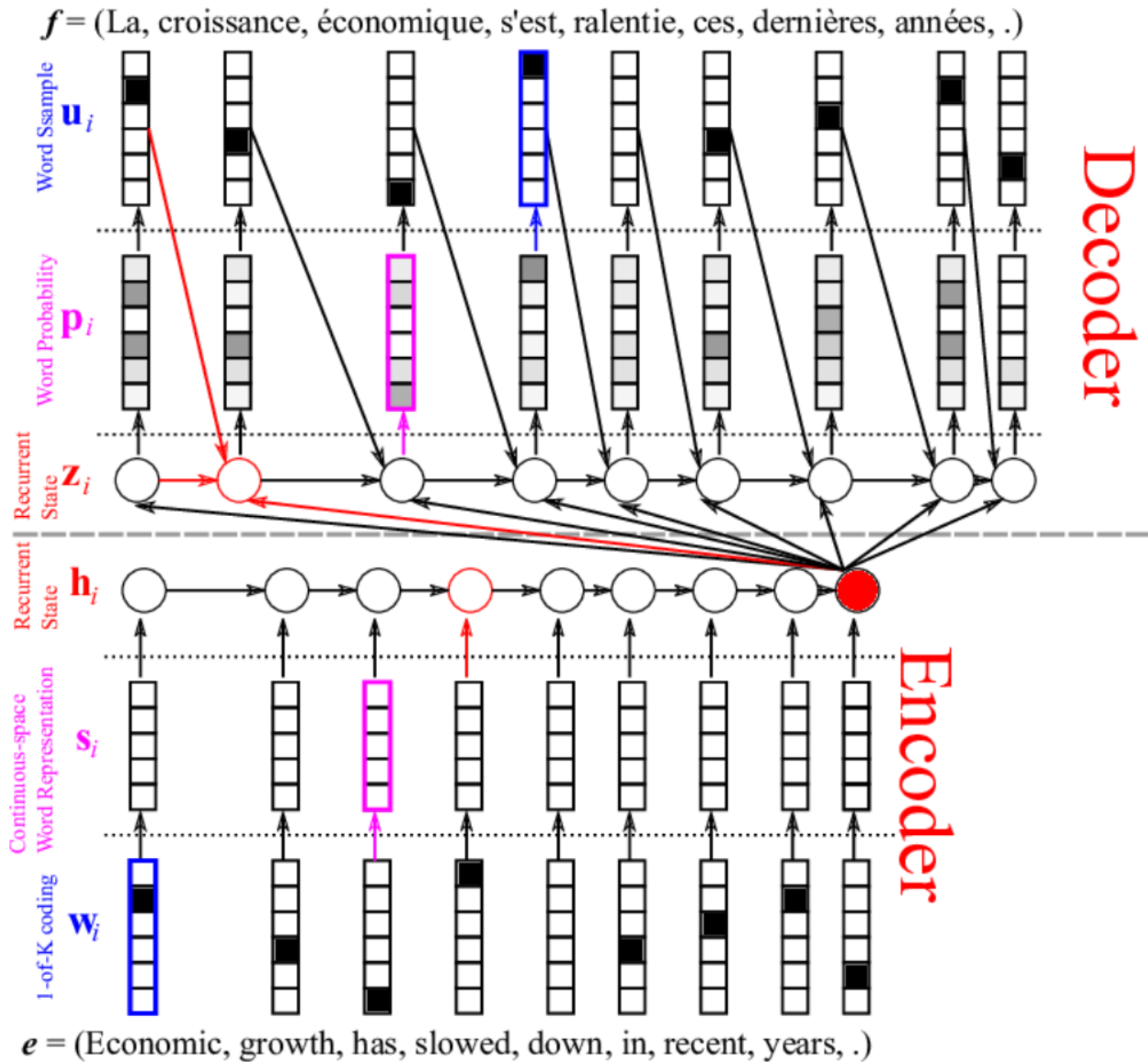
*(Sutskever et al., 2014; Cho et al., 2014)*

- ▶ sequence to sequence machine translation (seq2seq)

# Encoder-Decoder model

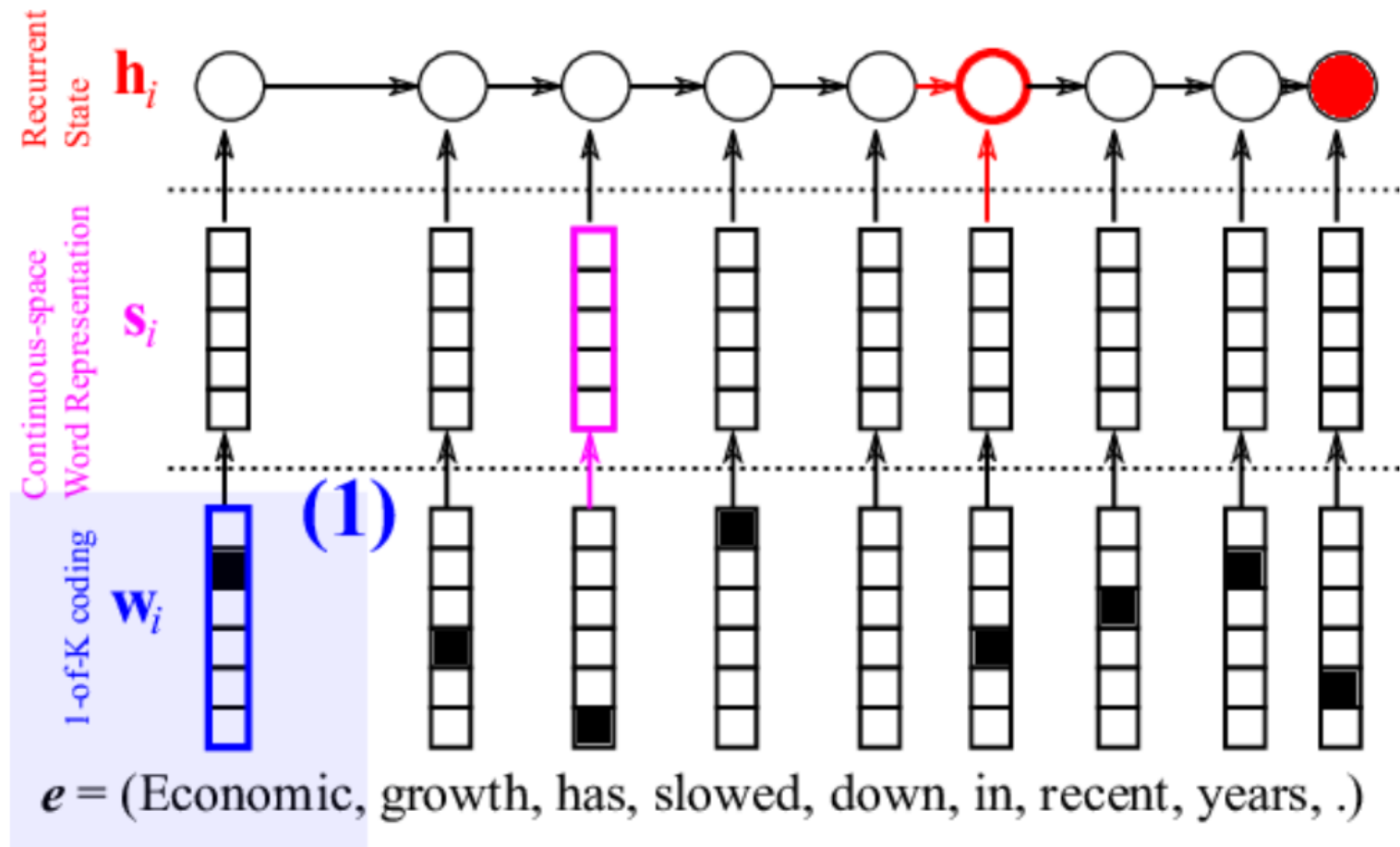


the basic architecture



# Encoder

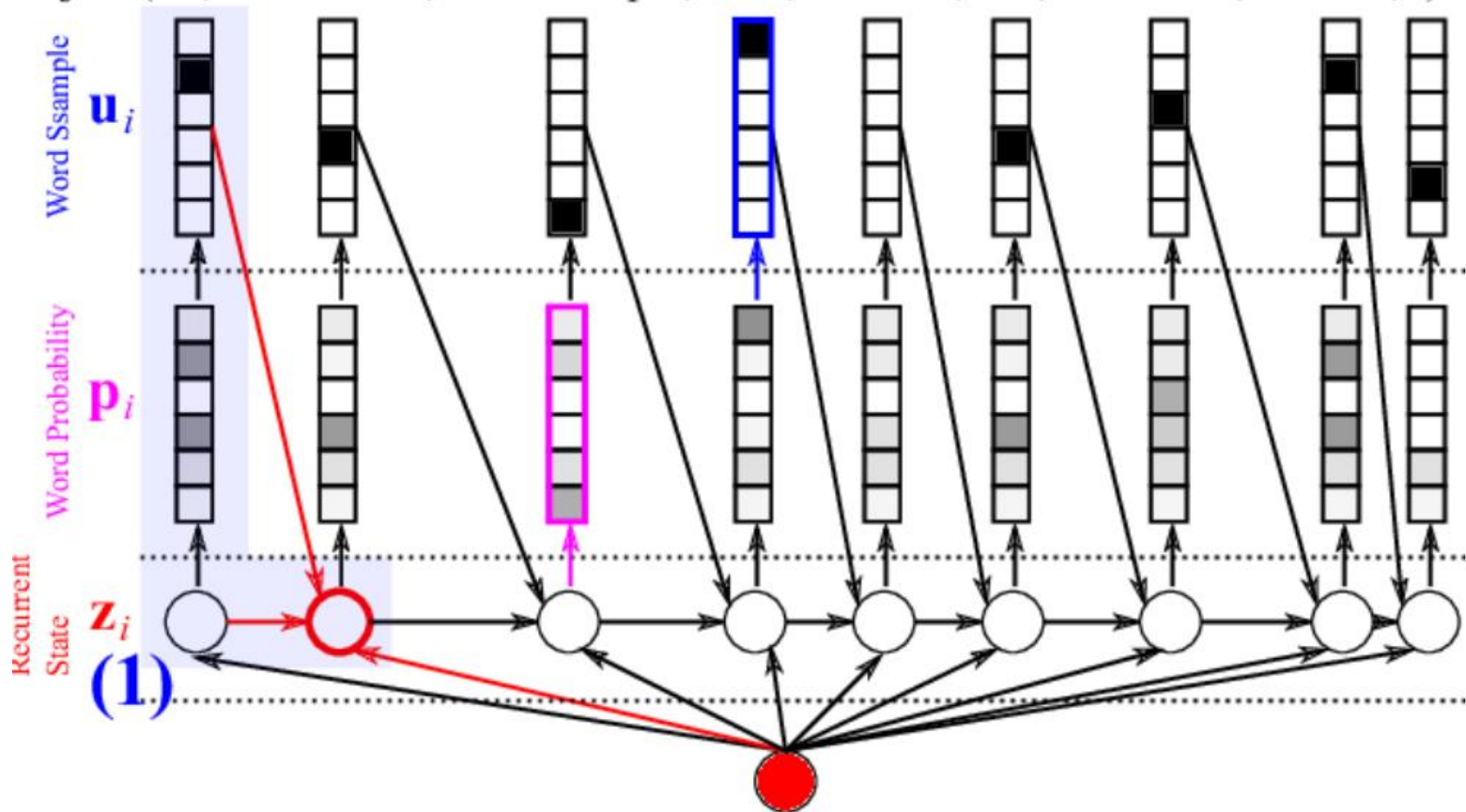
- word representation: word, 1-hot-vector, dense embedding, recurrent network



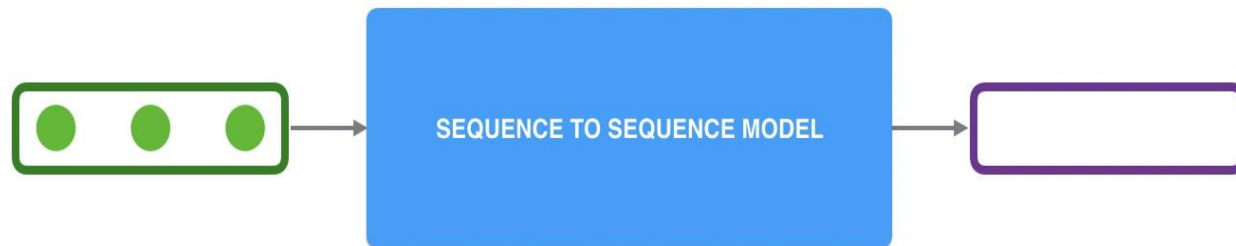
# Decoder

- ▶ computation of the next state of recurrent network, probability of the next word, selection of the next word

$f = (\text{La, croissance, économique, s'est, ralentie, ces, dernières, années, .})$

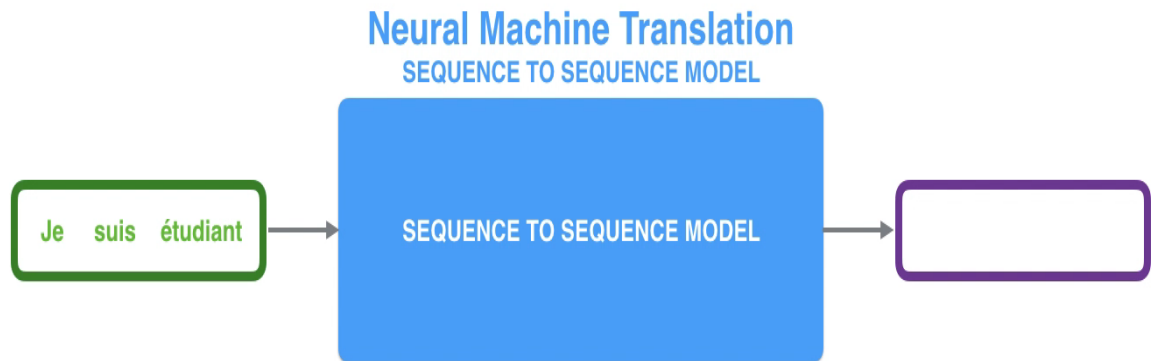


# Seq2Seq model



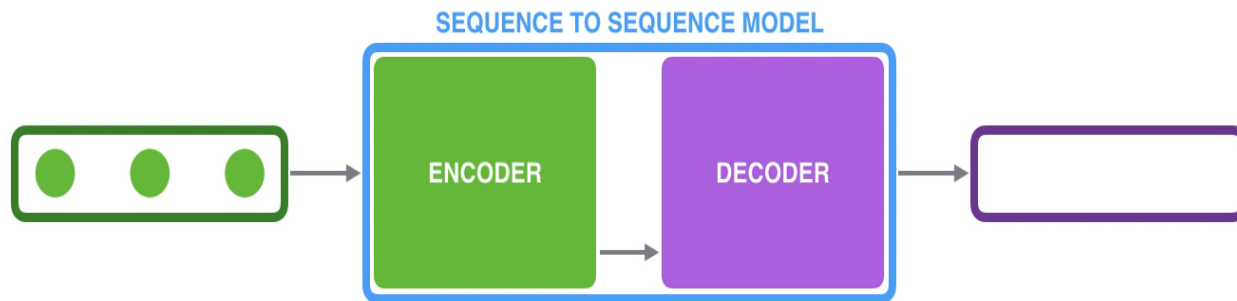
Videos by Jay Alammar: [Visualizing A Neural Machine Translation Model \(Mechanics of Seq2seq Models With Attention\)](#), 2018

# Seq2Seq for NMT

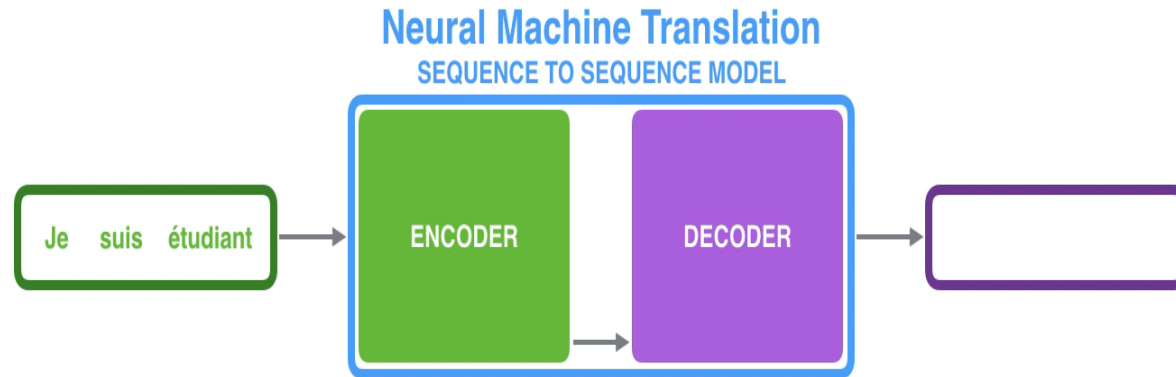




# Encoder-decoder for sequences



# Encoder-decoder for NMT



CONTEXT

0.11
0.03
0.81
-0.62

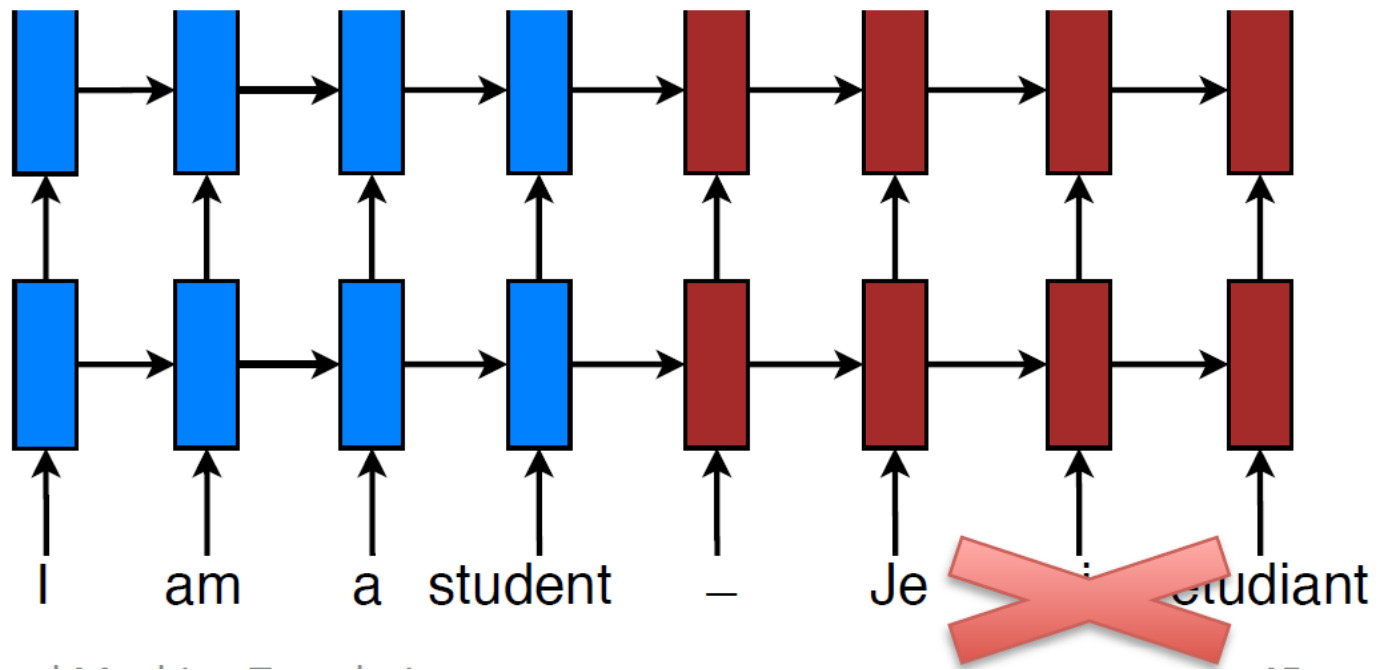
0.11
0.03
0.81
-0.62

# Training

- ▶ using RNN, LSTM, or Transformer as neurons
- ▶ softmax for output
- ▶ we maximize  
 $P(\text{output sentence} \mid \text{input sentence})$
- ▶ we sum errors on all outputs
- ▶ backpropagation
- ▶ training on correct translations
- ▶ as the translation, we return words with the highest probability (not necessary greedy)
- ▶ better than classical MT

# Using MT

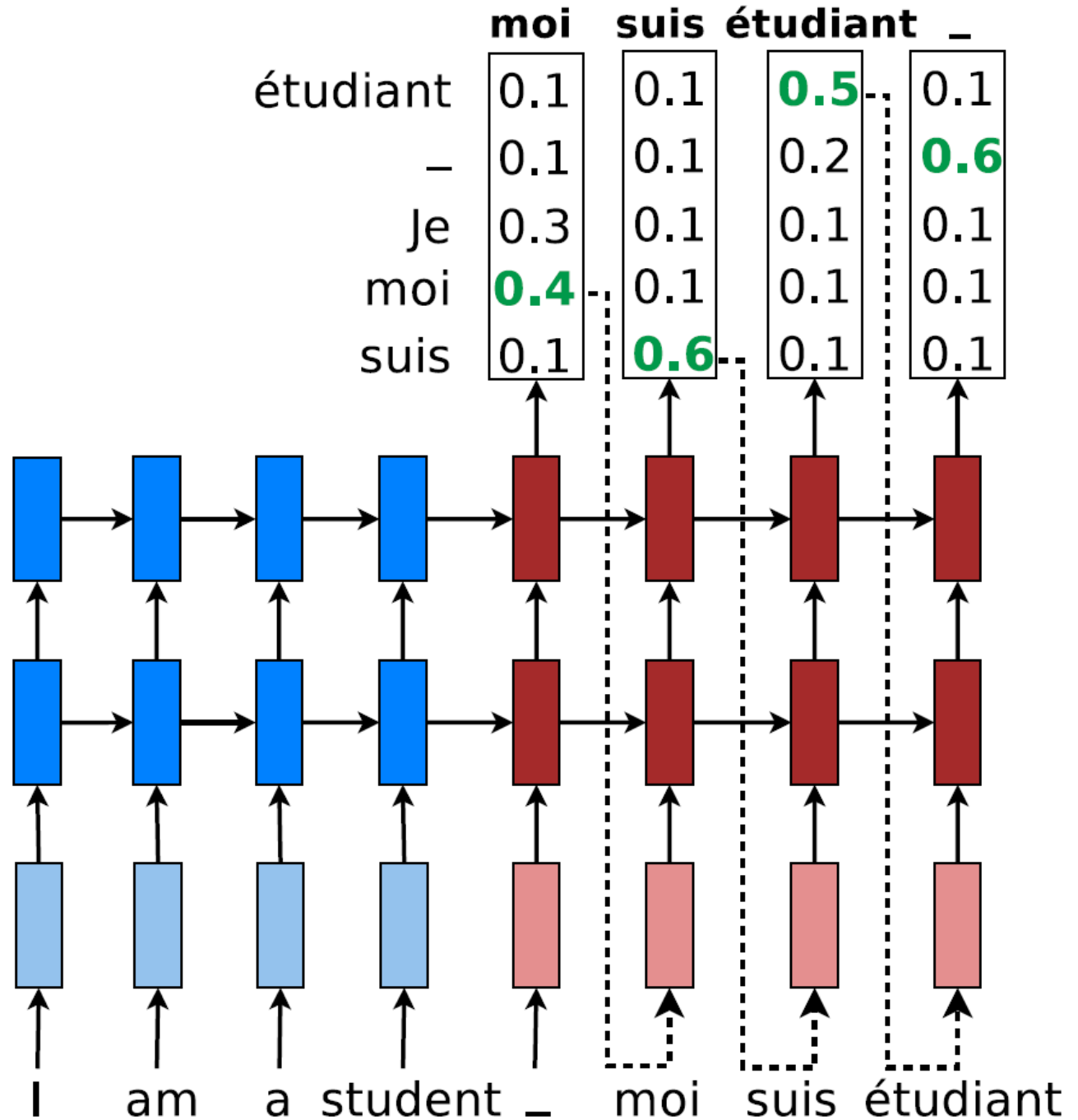
- ▶ we cannot provide the correct translations on the input



# Using MT

... therefore we provide the most probable translations – greedy 1-best, or as a beam search

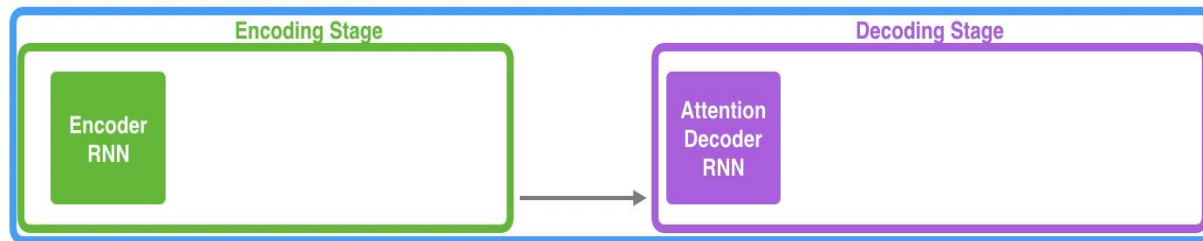
What if we sample from distribution?



# NMT with attention

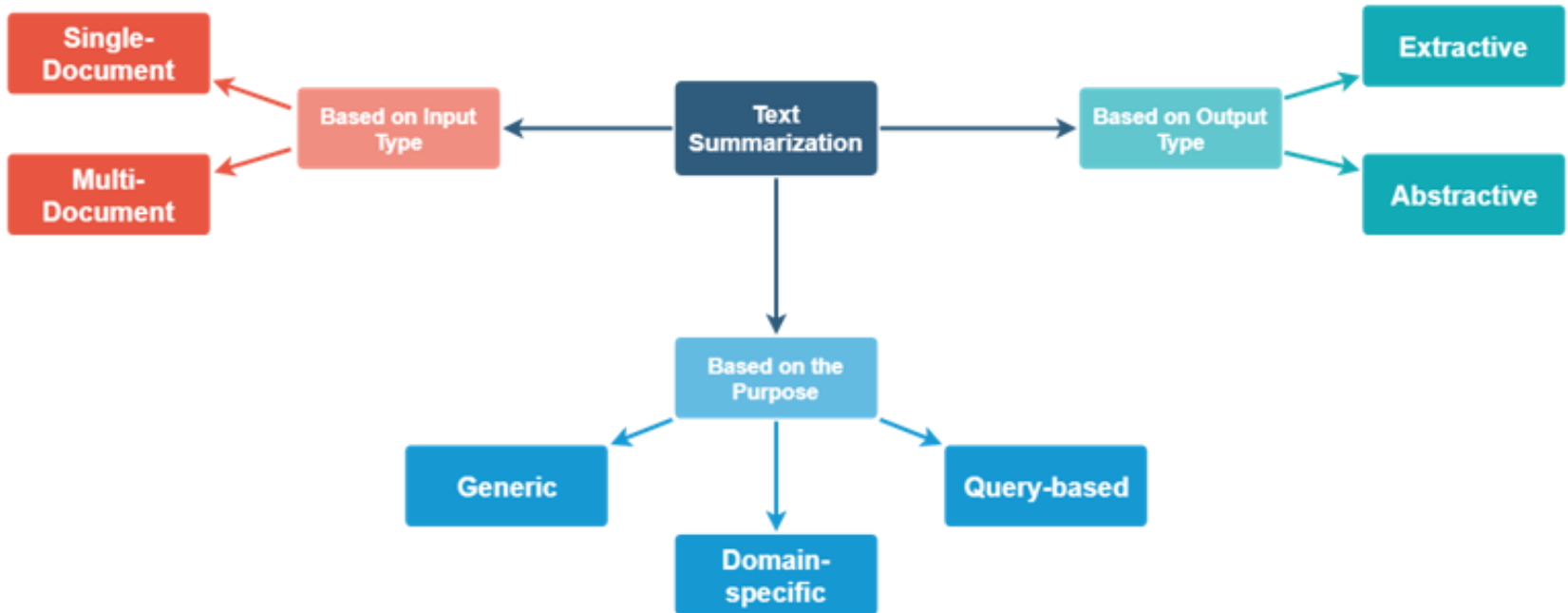
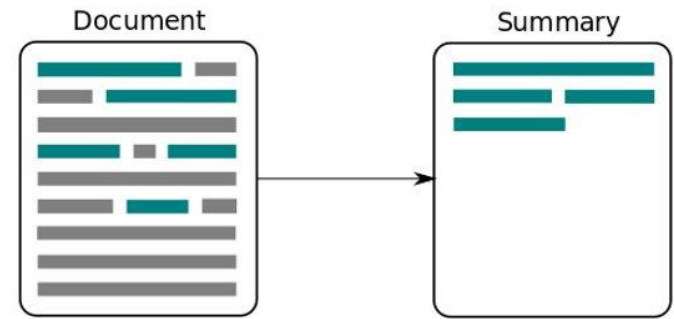
## Neural Machine Translation

SEQUENCE TO SEQUENCE MODEL WITH ATTENTION



Je suis étudiant

# Text summarization



# Text summarization

- Evaluation:
  - ROUGE scores,
  - BERTScore,
  - with QA:
    - question generation,
    - searching for answers in the summary
  - human
- Deep neural networks
- Short and long texts



# Graph-based summarization; An illustrative example

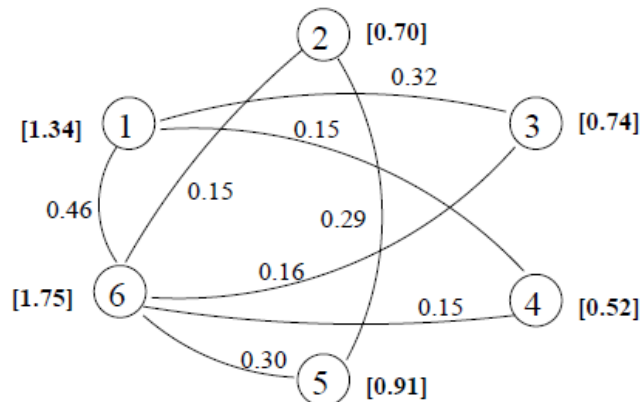
- [1] Watching the new movie, “Imagine: John Lennon,” was very painful for the late Beatle’s wife, Yoko Ono.  
 [2] “The only reason why I did watch it to the end is because I’m responsible for it, even though somebody else made it,” she said.  
 [3] Cassettes, film footage and other elements of the 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
<b>6</b>	<b>1.75</b>
<b>1</b>	<b>1.34</b>
5	0.91
3	0.74
2	0.70
4	0.52

↑ Sentence ranking/selection



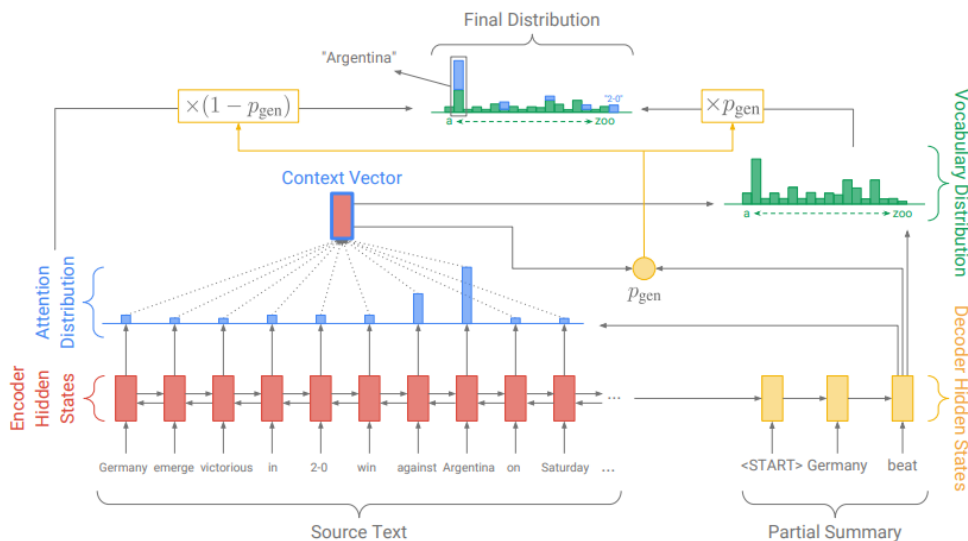
Text to graph/matrix



	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

# Summarization with Pointer Generator nwtworks

seq2seq abstractive summarizer



(See et al., 2017)

**Original Text (truncated):** lagos, nigeria (cnn) a day after winning nigeria's presidency, *muhammadu buhari* told cnn's christiane amannpour that **he plans to aggressively fight corruption that has long plagued nigeria** and go after the root of the nation's unrest. *buhari* said he'll "rapidly give attention" to curbing violence in the northeast part of nigeria, where the terrorist group boko haram operates. by cooperating with neighboring nations chad, cameroon and niger, **he said his administration is confident it will be able to thwart criminals** and others contributing to nigeria's instability. for the first time in nigeria's history, the opposition defeated the ruling party in democratic elections. *buhari* defeated incumbent goodluck jonathan by about 2 million votes, according to nigeria's independent national electoral commission. **the win comes after a long history of military rule, coups and botched attempts at democracy in africa's most populous nation.**

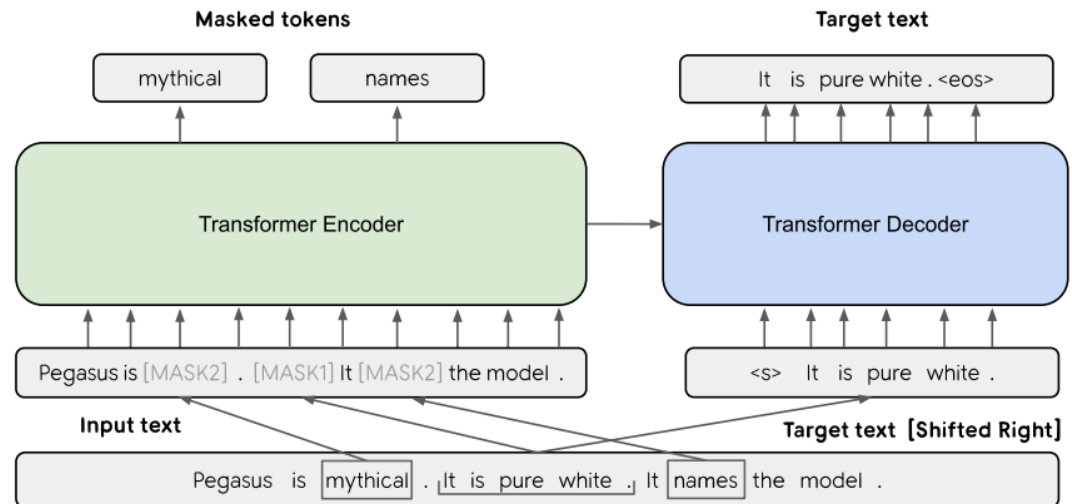
**Baseline Seq2Seq + Attention:** UNK UNK says his administration is confident it will be able to **destabilize nigeria's economy**. UNK says his administration is confident it will be able to thwart criminals and other **nigerians**. **he says the country has long nigeria and nigeria's economy.**

**Pointer-Gen:** *muhammadu buhari* says he plans to aggressively fight corruption **in the northeast part of nigeria**. he says he'll "rapidly give attention" to curbing violence **in the northeast part of nigeria**. he says his administration is confident it will be able to thwart criminals.

**Pointer-Gen + Coverage:** *muhammadu buhari* says he plans to aggressively fight corruption that has long plagued nigeria. he says his administration is confident it will be able to thwart criminals. the win comes after a long history of military rule, coups and botched attempts at democracy in africa's most populous nation.

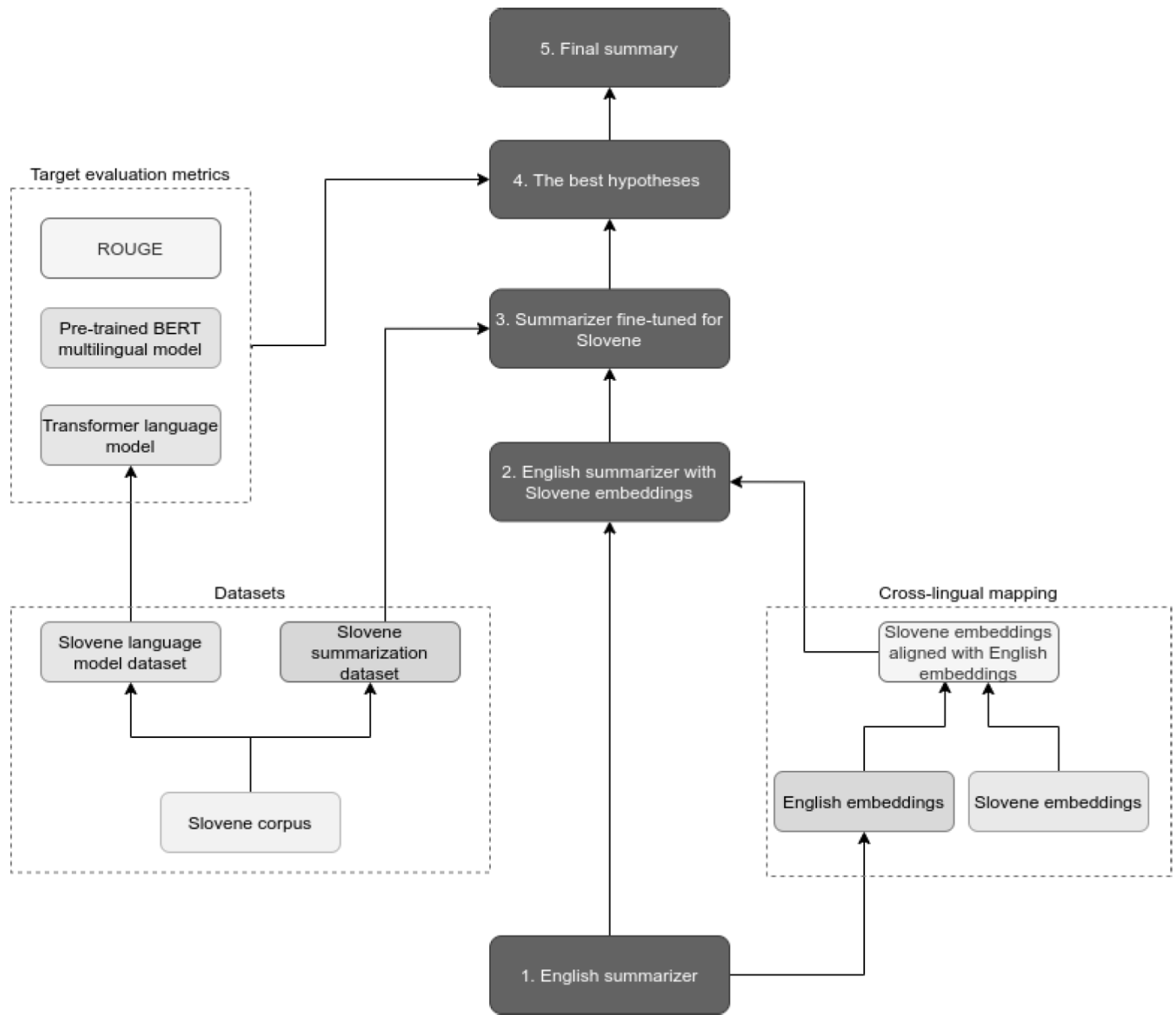
# Summarizers – Pegasus

- ▶ Transformer BART
- ▶ encoder-decoder architecture
- ▶ text garbling and reconstruction
- ▶ Auxiliary tasks: masked language model and missing sentence generation
- ▶ Demo: <https://ai.googleblog.com/2020/06/pegasus-state-of-art-model-for.html>

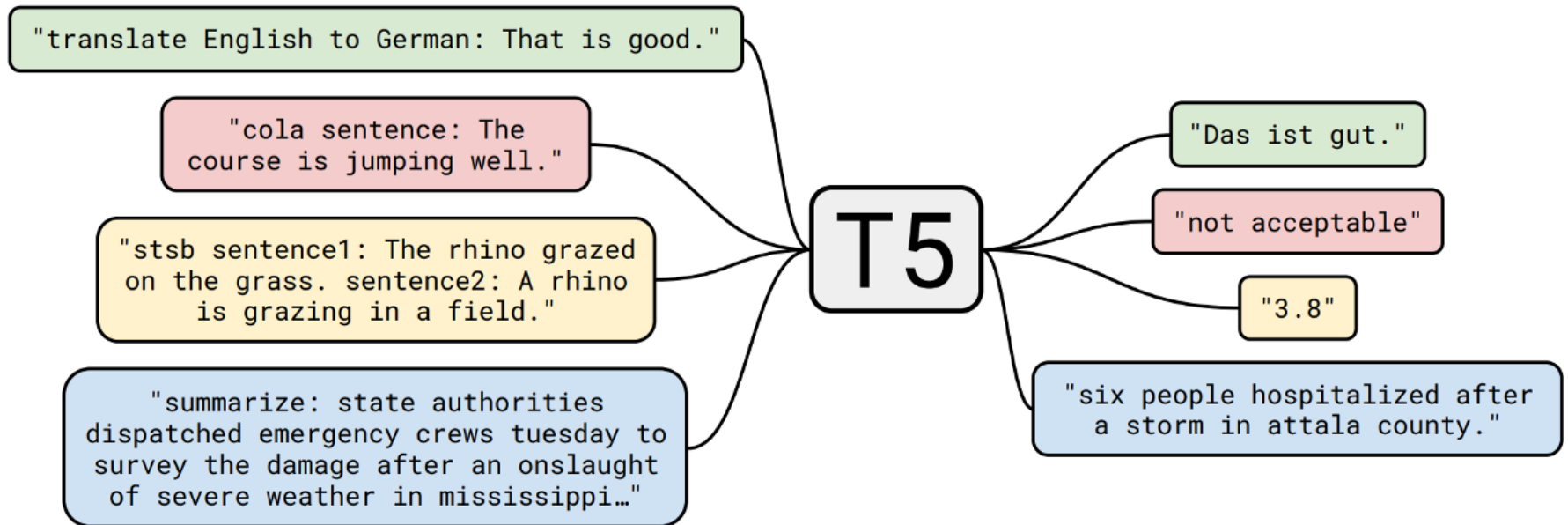




## XL summarization architecture



# T5 (Text-To-Text Transfer Transformer) models



Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y, Li, W. & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140), 1-67.

# Unified QA

- Use several types of questions in T5 model to generate answers: extractive, abstractive, multichoice, yes/no
- A model is trained on all types of questions,
- Finetuned on a specific type of questions

## Extractive [SQuAD]

**Question:** At what speed did the turbine operate?  
**Context:** (Nikola\_Tesla) On his 50th birthday in 1906, Tesla demonstrated his 200 horsepower (150 kilowatts) **16,000 rpm** bladeless turbine. ...  
**Gold answer:** 16,000 rpm

## Abstractive [NarrativeQA]

**Question:** What does a drink from narcissus's spring cause the drinker to do?  
**Context:** Mercury has awakened Echo, who weeps for Narcissus, and states that a drink from Narcissus's spring causes the drinkers to "Grow dotingly enamored of themselves." ...  
**Gold answer:** fall in love with themselves

## Multiple-Choice [ARC-challenge]

**Question:** What does photosynthesis produce that helps plants grow?  
**Candidate Answers:** (A) water (B) oxygen (C) protein (D) sugar  
**Gold answer:** sugar

## Yes/No [BoolQ]

**Question:** Was America the first country to have a president?  
**Context:** (President) The first usage of the word president to denote the highest official in a government was during the Commonwealth of England ...  
**Gold answer:** no

Khashabi, Min, Khot, Sabharwal, Tafjord, Clark in  
Hajishirzi. UnifiedQA: Crossing Format Boundaries With a Single QA System. *V Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, str. 1896–1907, 2020.