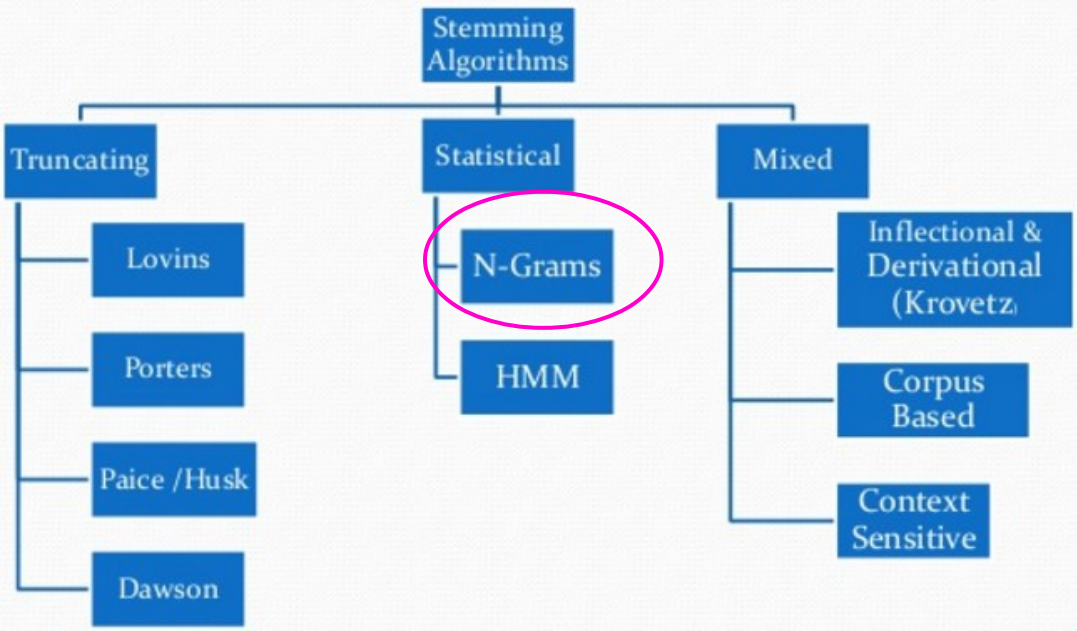
MCZA017-13
Processamento de Linguagem Natural

Modelando a linguagem com N-gramas

Prof. Jesús P. Mena-Chalco
jesus.mena@ufabc.edu.br

2Q-2019

Outras abordagens para stemming?



Bibliografia – Capítulo 3

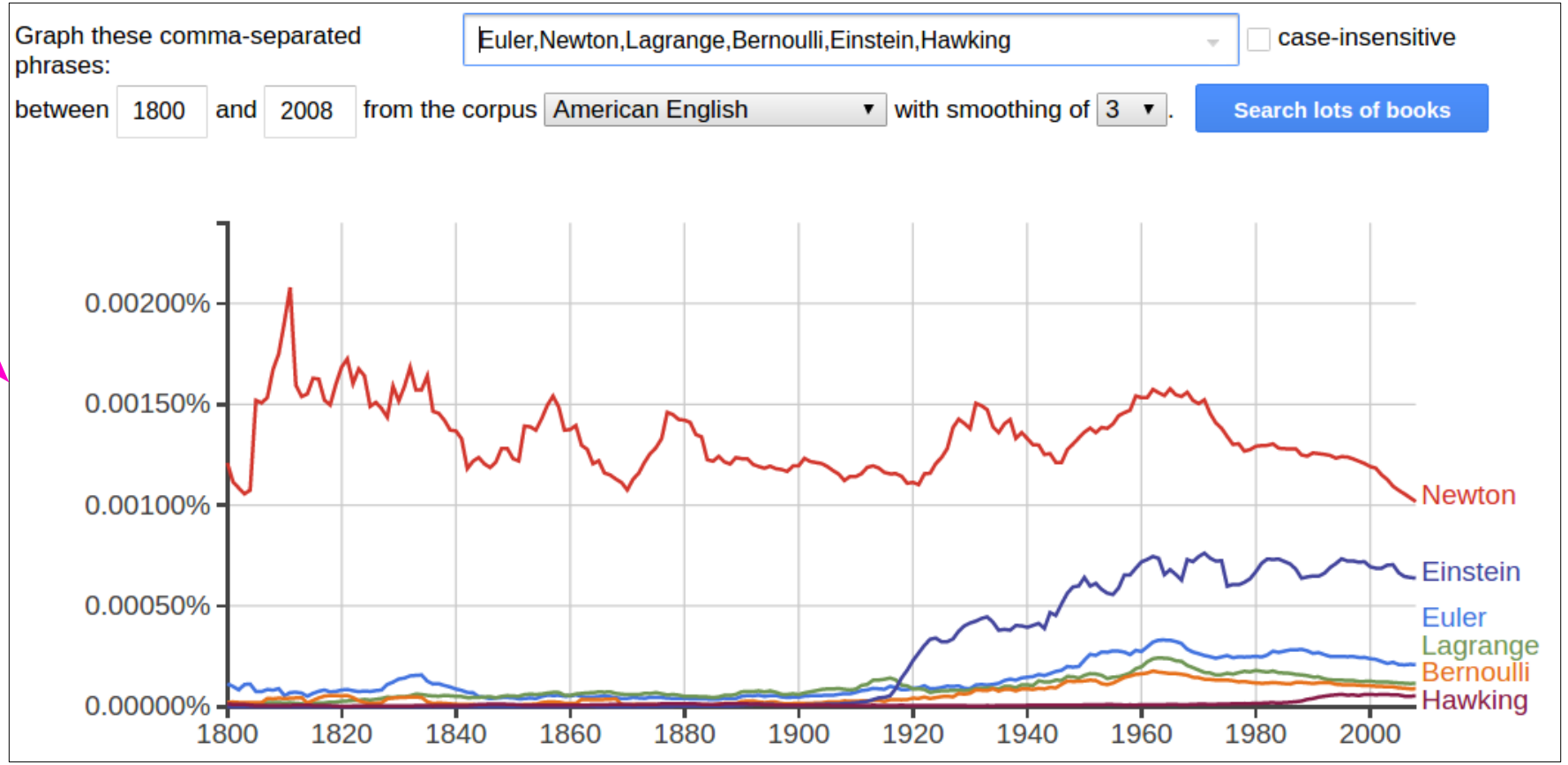
Speech and Language Processing (3rd ed. draft)

Dan Jurafsky and James H. Martin

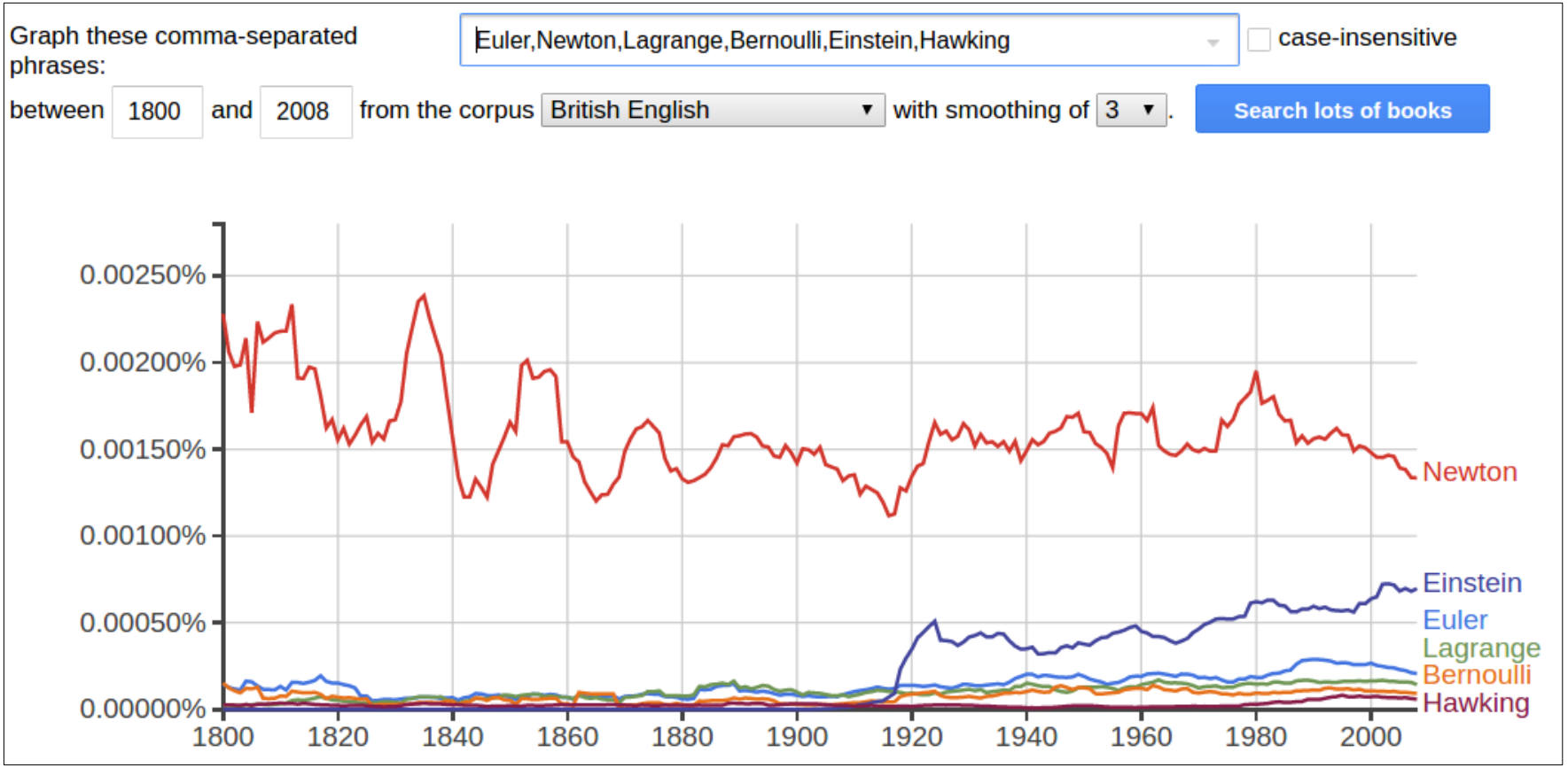
	Chapter	Slides	Relation to 2nd ed.
1:	Introduction		[Ch. 1 in 2nd ed.]
2:	Regular Expressions, Text Normalization, and Edit Distance	Text [pptx] [pdf] Edit Distance [pptx] [pdf]	[Ch. 2 and parts of Ch. 3 in 2nd ed.]
3:	Language Modeling with N-Grams	LM [pptx] [pdf]	[Ch. 4 in 2nd ed.]
4:	Naive Bayes Classification and Sentiment	NB [pptx] [pdf] Sentiment [pptx] [pdf]	[new in this edition]
5:	Logistic Regression		
6:	Vector Semantics	Vector1 [pptx] [pdf] Vector2 [pptx] [pdf]	
7:	Neural Nets and Neural Language Models		[new in this edition]
8:	Part-of-Speech Tagging		[Ch. 5 in 2nd ed.]
9:	Sequence Processing with Recurrent Networks		[new in this edition]
X:	Encoder-Decoder Models and Attention		[new in this edition]
10:	Formal Grammars of English		[Ch. 12 in 2nd ed.]
11:	Syntactic Parsing		[Ch. 13 in 2nd ed.]
12:	Statistical Parsing		[Ch. 14 in 2nd ed.]
13:	Dependency Parsing		[new in this edition]
14:	The Representation of Sentence Meaning		
15:	Computational Semantics		
16:	Semantic Parsing		
17:	Information Extraction		[Ch. 22 in 2nd ed.]

https://web.stanford.edu/~jurafsky/slp3/

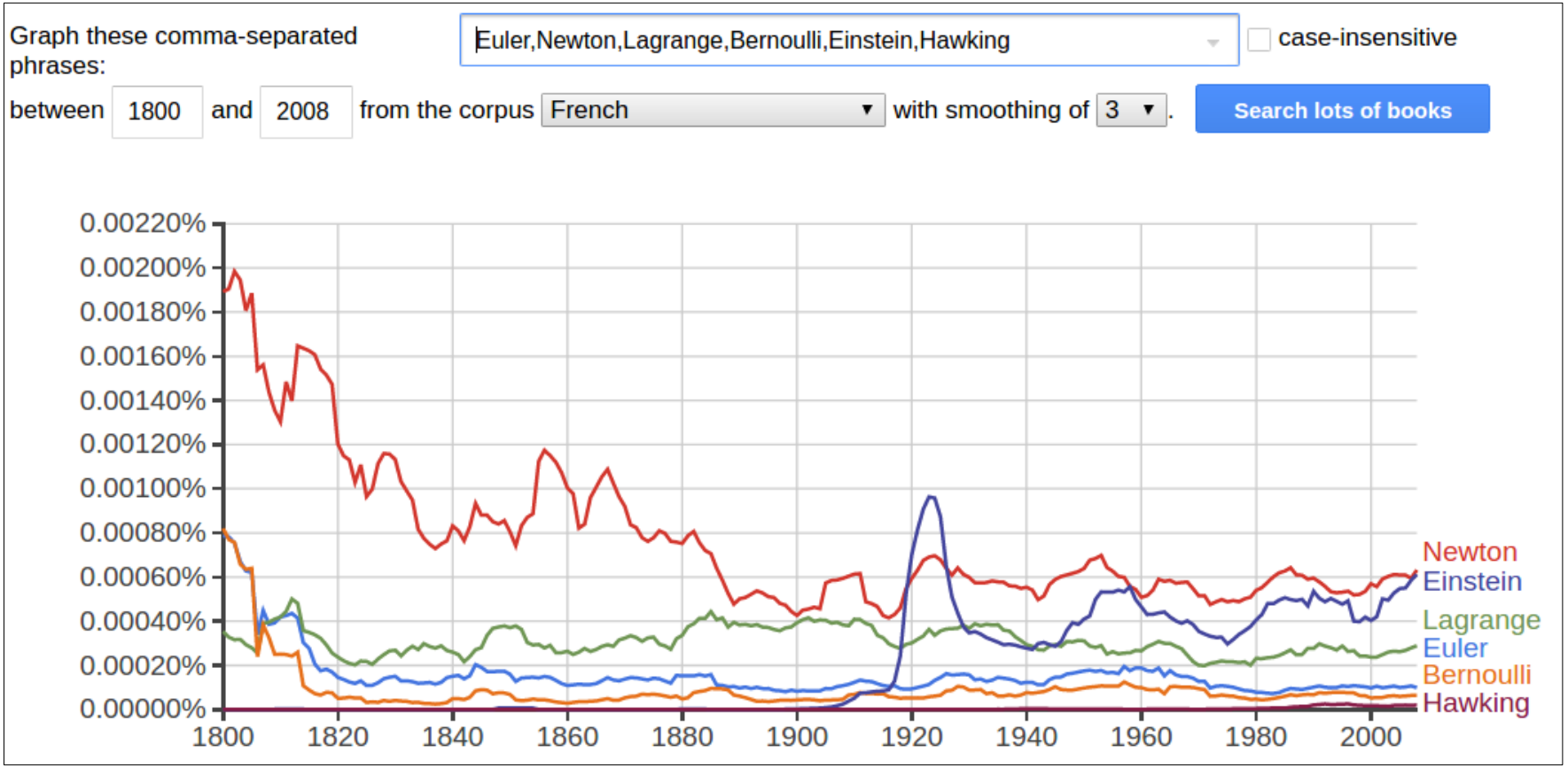
Porcentagem da presença do n-grama no corpus



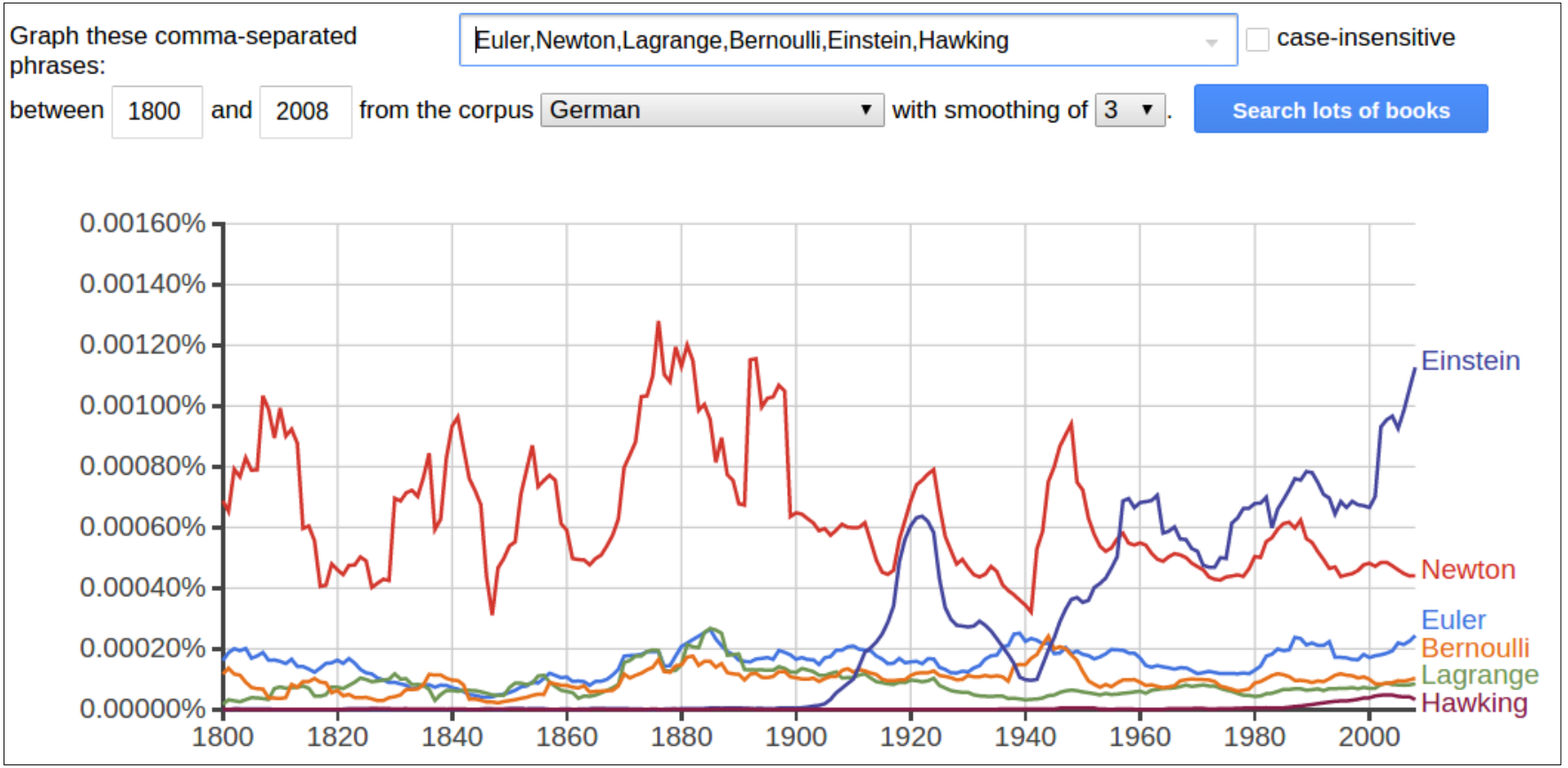
- *Para ser contabilizado o n-grama precisa estar presente em pelo menos 40 livros .*
- *Dados normalizados por ano.*
- *Crítica: Os dados podem estar incompletos (devido a erro nos OCRs).*



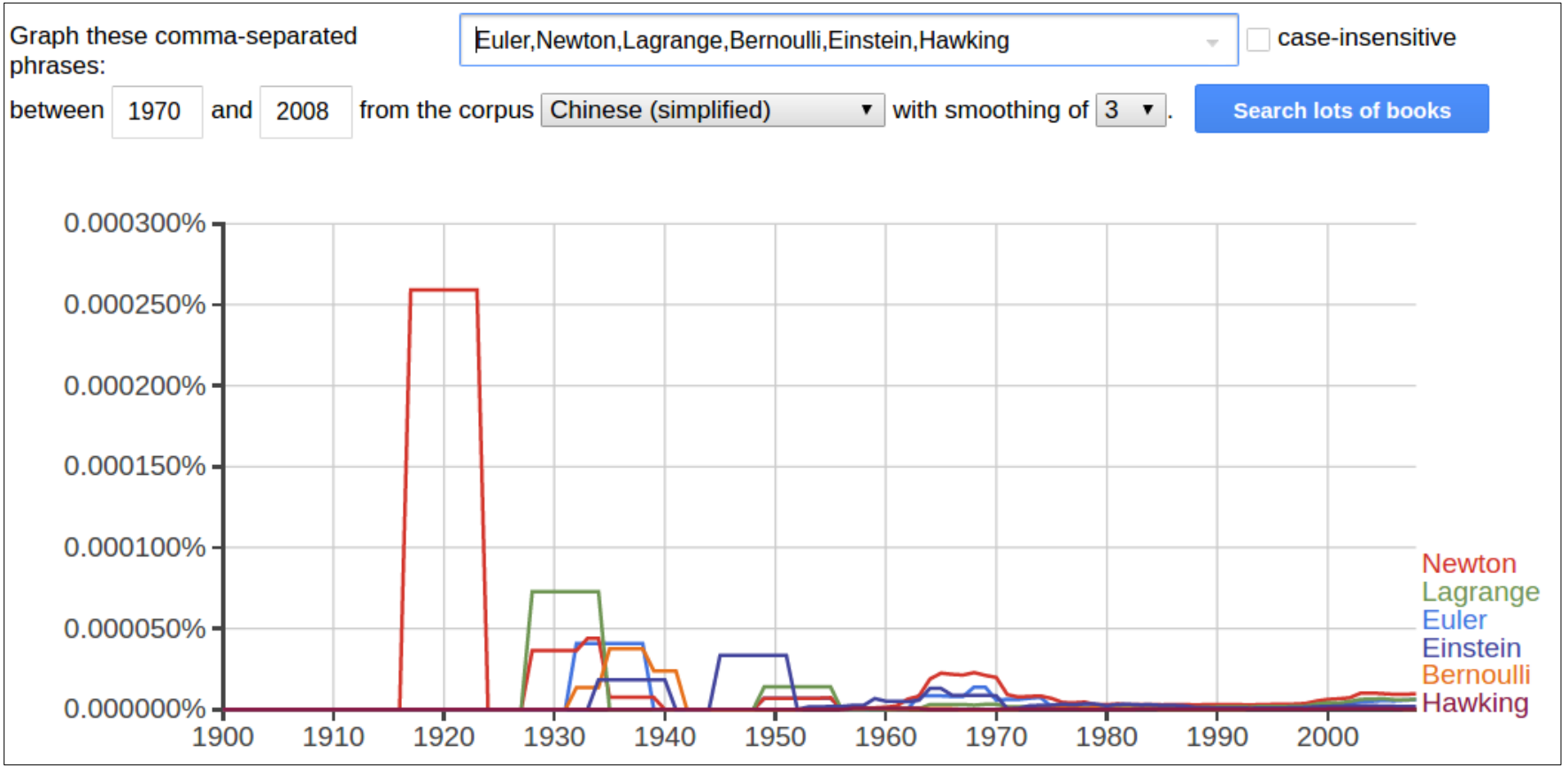
https://books.google.com/ngrams



https://books.google.com/ngrams



https://books.google.com/ngrams



https://books.google.com/ngrams

Google Books N-gram Viewer

- Fonte de dados: livros indexados pelo Google.
- Muitos dos livros foram digitalizados das coleções das bibliotecas (acadêmicas ou públicas)

https://books.google.com

N-gramas

- Um N-grama é uma sequência contigua de **N** elementos (e.g., caracteres, palavras, sílabas, fonemas, pares-base).
- São comumente obtidas (analisadas) a partir de um *corpus*.

Número de elementos	Nome
1	Unigrama
2	Bigrama , Digrama
3	Trigrama
4	4-grama
5	5-grama

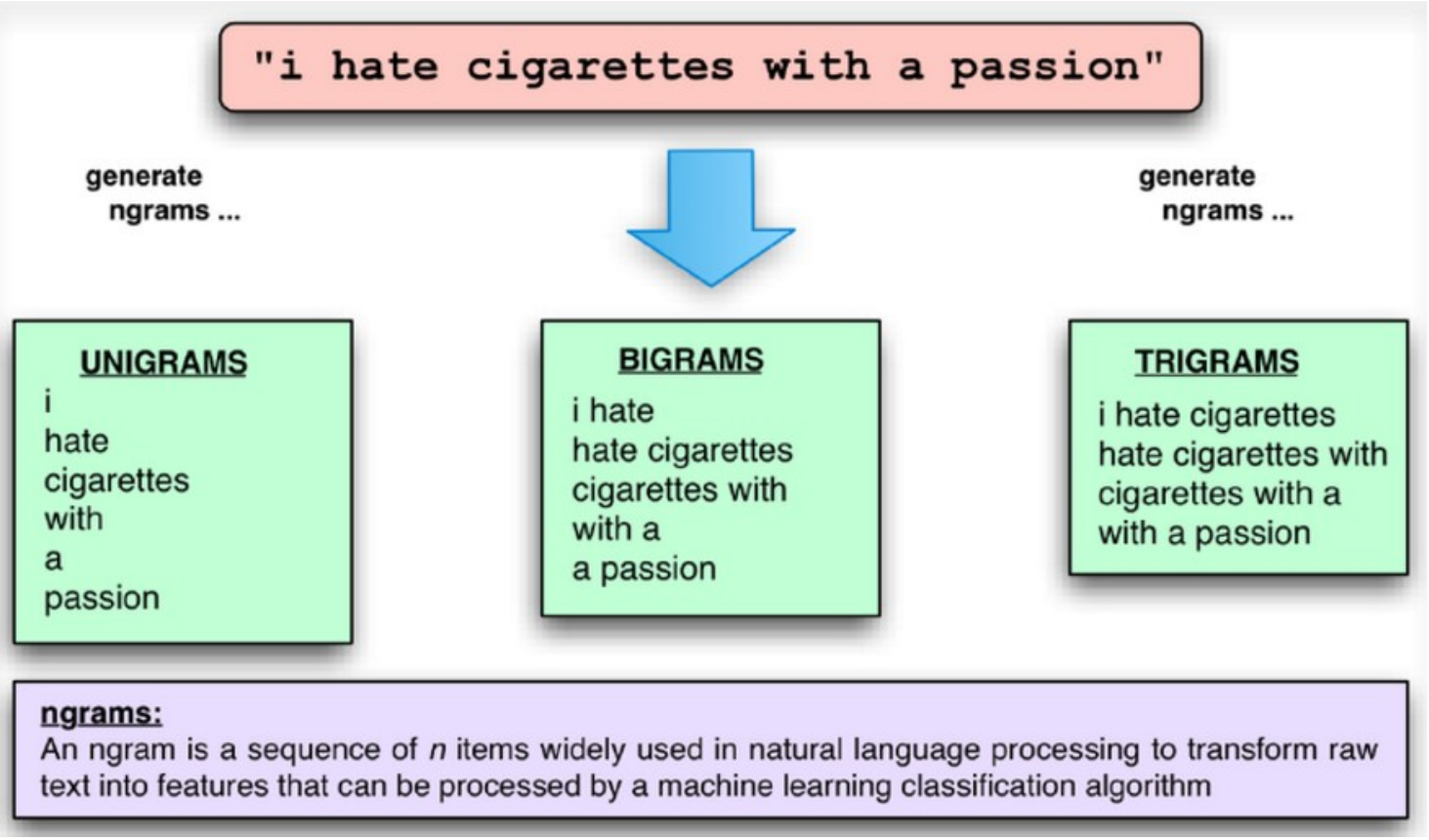
N-gramas

Exemplo:

- Um dois tres quatro — *4 Unigramas*
- Um dois tres quatro — *3 Bigramas*
- Um dois tres quatro — *2 Trigramas*

Field	Unit	Sample sequence	1-gram sequence	2-gram sequence	3-gram sequence
Vernacular name			**unigram**	**bigram**	**trigram**
Order of resulting Markov model			**0**	**1**	**2**
Protein sequencing	amino acid	… Cys-Gly-Leu-Ser-Trp …	…, Cys, Gly, Leu, Ser, Trp, …	…, Cys-Gly, Gly-Leu, Leu-Ser, Ser-Trp, …	…, Cys-Gly-Leu, Gly-Leu-Ser, Leu-Ser-Trp, …
DNA sequencing	base pair	…AGCTTCGA…	…, A, G, C, T, T, C, G, A, …	…, AG, GC, CT, TT, TC, CG, GA, …	…, AGC, GCT, CTT, TTC, TCG, CGA, …
Computational linguistics	character	… to_be_or_not_to_be…	…, t, o, _, b, e, _, o, r, _, n, o, t, _, t, o, _, b, e, …	…, to, o_, _b, be, e_, _o, or, r_, _n, no, ot, t_, _t, to, o_, _b, be, …	…, to_, o_b, _be, be_, e_o, _or, or_, r_n, _no, not, ot_, t_t, _to, to_, o_b, _be, …
Computational linguistics	word	… to be or not to be …	…, to, be, or, not, to, be, …	…, to be, be or, or not, not to, to be, …	…, to be or, be or not, or not to, not to be, …

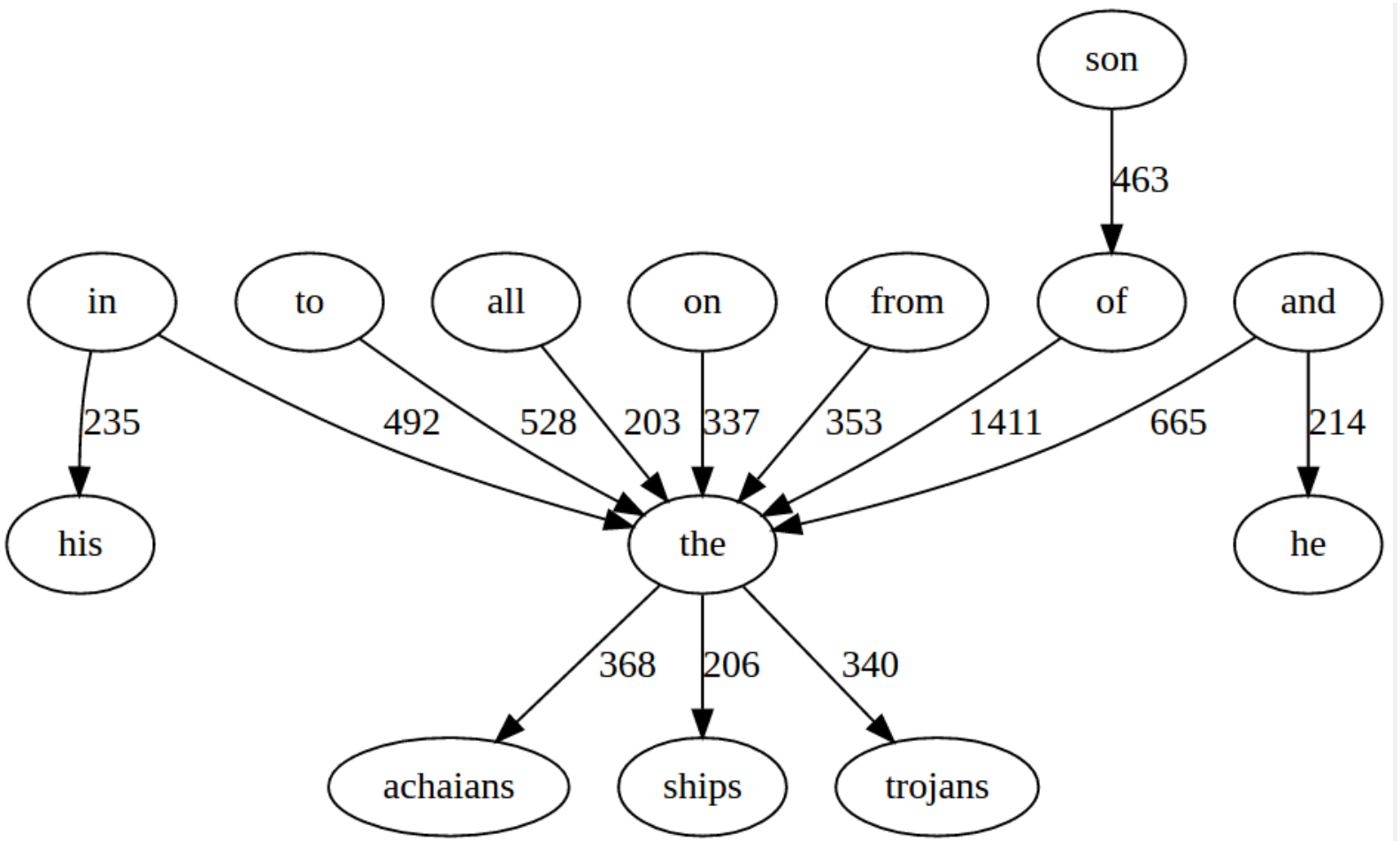
N-gramas



Using Twitter to Examine Smoking Behavior and Perceptions of Emerging Tobacco Products

https://www.researchgate.net/publication/256290162_Using_Twitter_to_Examine_Smoking_Behavior_and_Perceptions_of_Emerging_Tobacco_Products

N-gramas

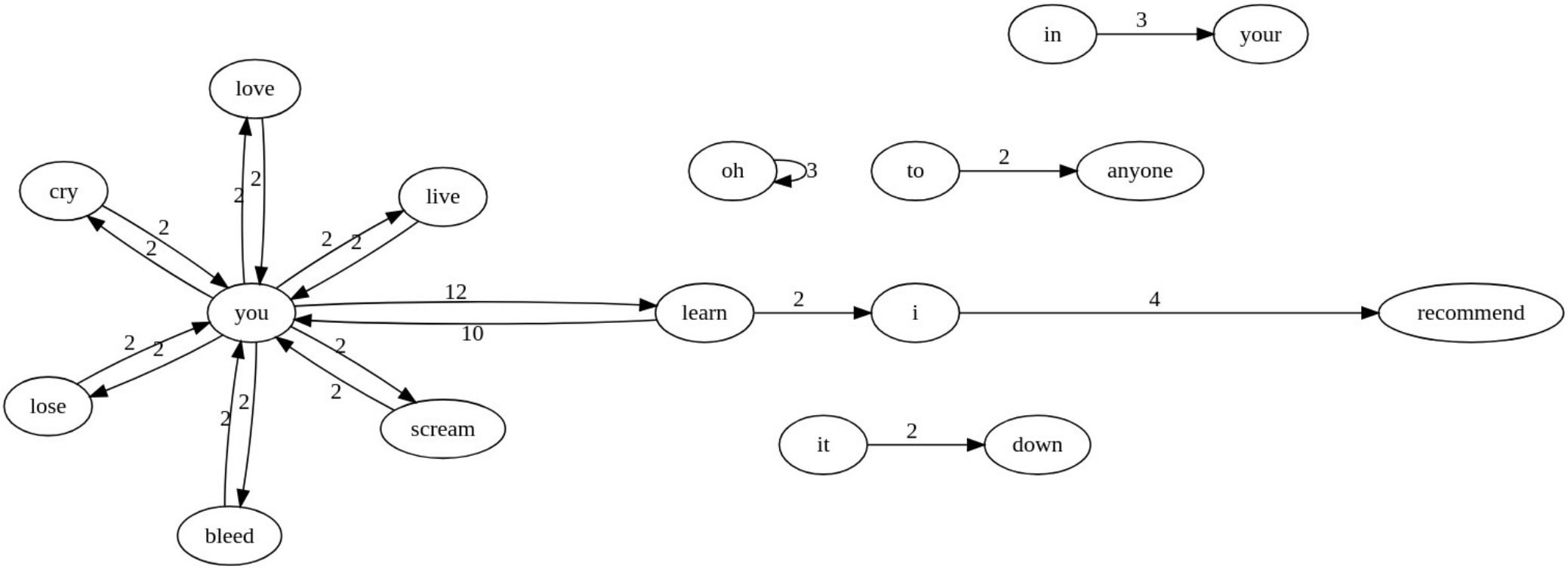


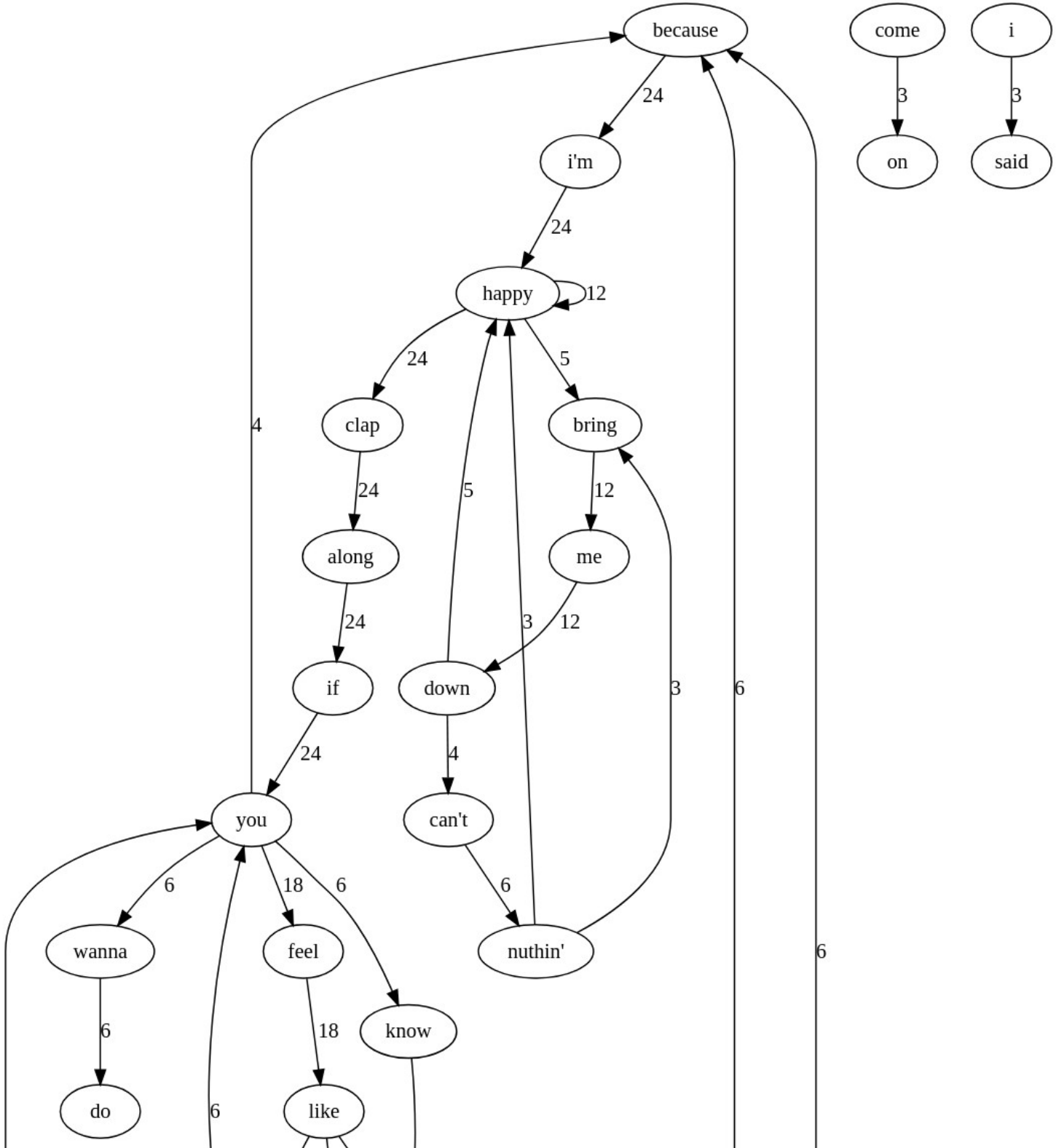
Bigramas: da aula anterior (The Iliad of Homer)

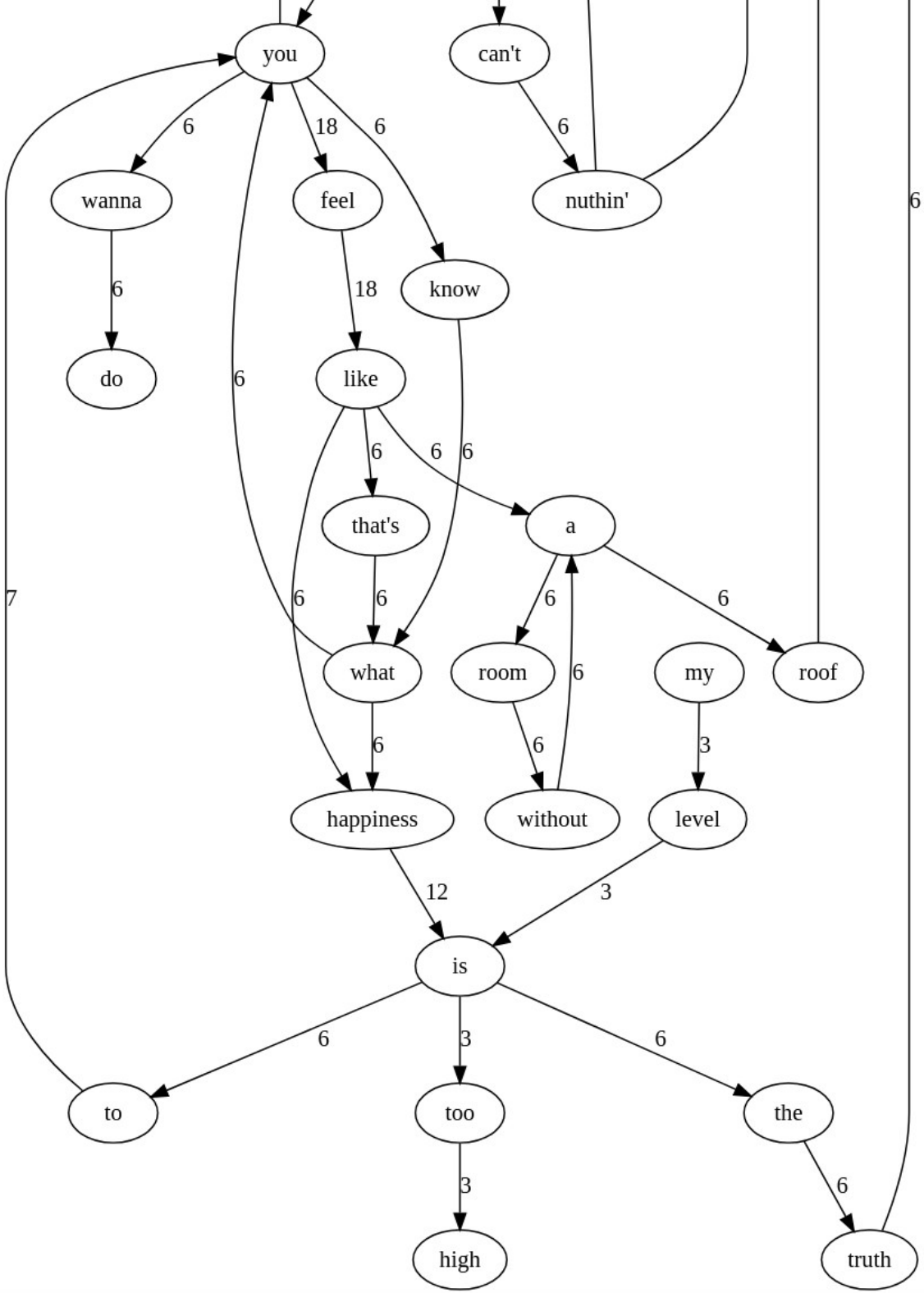
You learn ~Alanis Morissette

Oh, oh, oh
I, recommend getting your heart trampled on to anyone, yeah
I, recommend walking around naked in your living room, yeah
Swallow it down (what a jagged little pill)
It feels so good (swimming in your stomach)
Wait until the dust settles
You live you learn, you love you learn
You cry you learn, you lose you learn
You bleed you learn, you scream you learn
I, recommend biting off more than you can chew to anyone
I certainly do
I, recommend sticking your foot in your mouth at any time
Feel free
Throw it down (the caution blocks you from the wind)
Hold it up (to the rays)
You wait and see when the smoke clears
You live you learn, you love you learn
You cry you learn, you lose you learn
You bleed you learn, you scream you learn
I, I, oh, oh
Wear it…

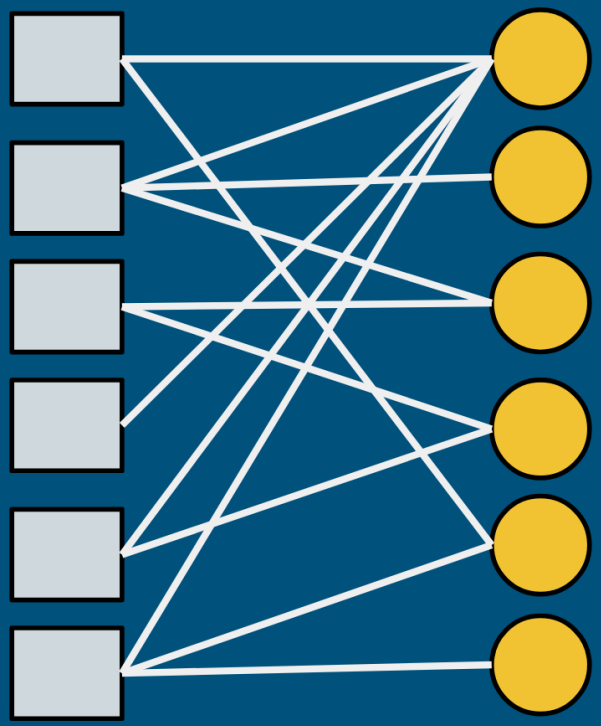
You learn ~Alanis Morissette



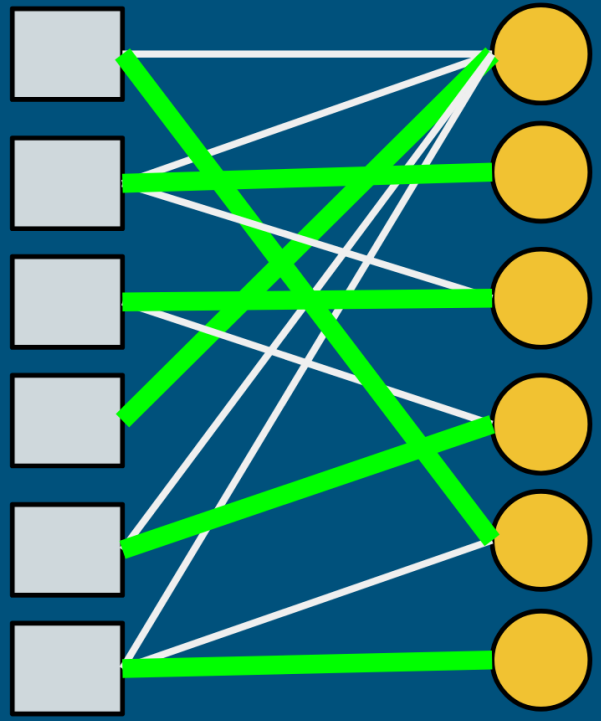

because
come
i
24
3
3
i'm
on
said
24
happy
12
24
5
clap
bring
4
5
24
12
along
me
24
3
12
if
down
3
6
24
4
you
can't
6
6
18
6
wanna
feel
nuthin'
6
6
18
know
do
6
like


you
can't
wanna
feel
nuthin'
know
do
like
that's
a
what
room
my
roof
happiness
without
level
is
to
too
the
high
truth
6
18
6
6
6
6
18
6
6
6
6
6
6
6
6
6
6
6
6
6
7
6
3
12
3
6
3
6
3
6
6

Aplicação:
Atribuição de avaliadores para projetos

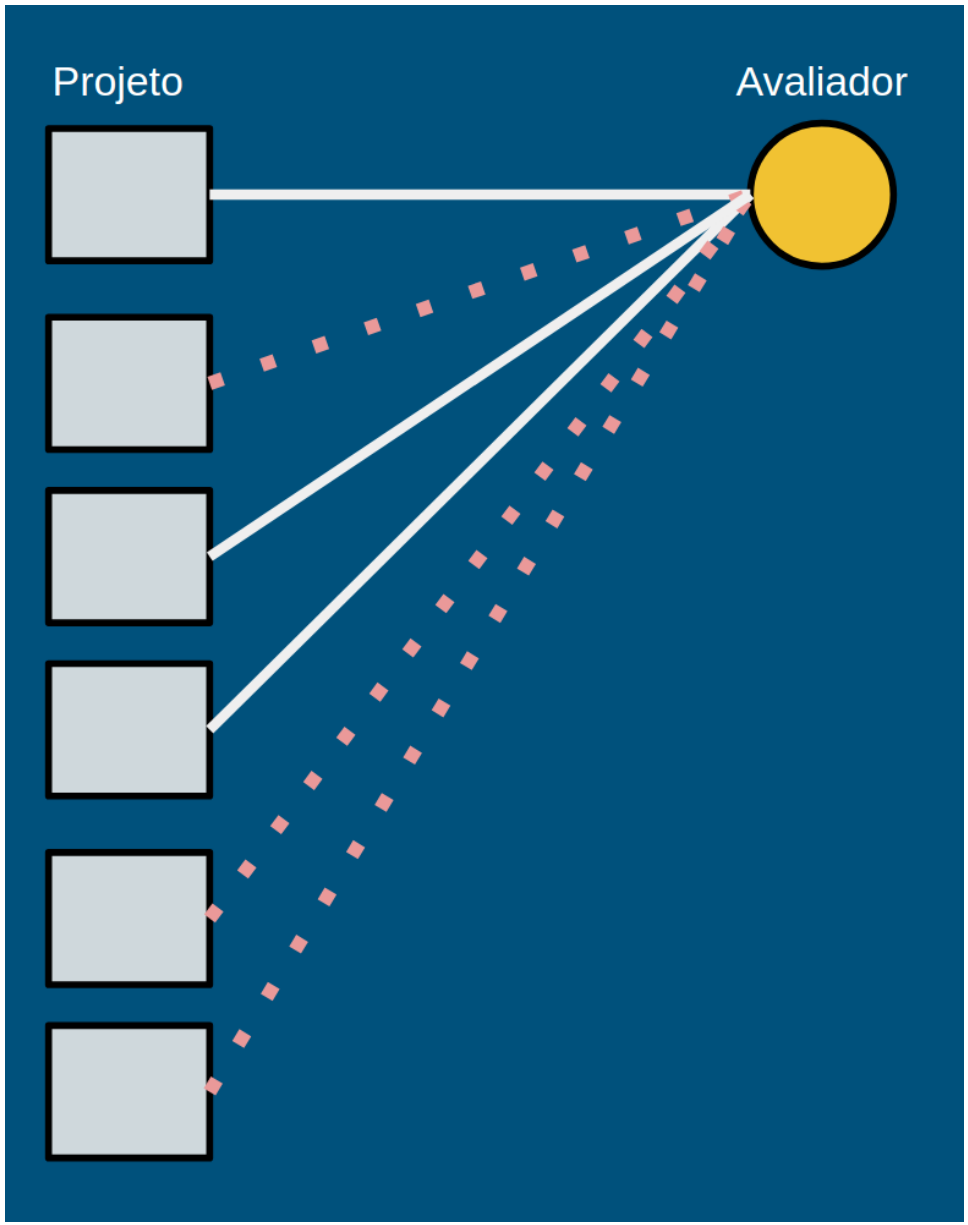


Projetos Avaliadores



Problema:
Atribuir avaliador para cada projeto, respeitando as restrições

Aplicação:
Atribuição de avaliadores para projetos



Aplicação:
Atribuição de avaliadores para projetos

O processo considerou:

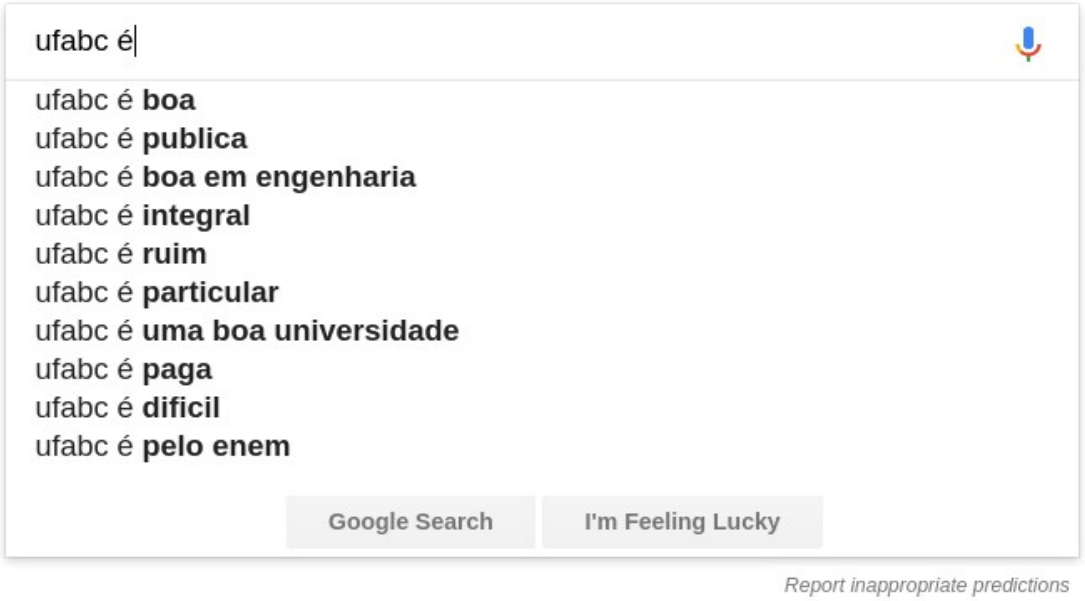
- Padronizar para minúsculas
- Remover as stop-words (EN, PT, SP)
- Radicalizar as palavras (Porter Stemming)
- Agrupamento dos trabalhos por bi-gramas

```
('facial', 'anim')
('pca', 'space')
('3d', 'human')
('princip', 'compon')
('basi', 'function')
('radial', 'basi')
('3d', 'facial')
('3d', 'wavelet')
('express', '2d')
('face', 'high-dimens')
('2d', '3d')
('compon', 'space')
('face', 'reconstruct')
('high-dimens', 'invers')
('3d', 'face')
('face', 'comput')
('photographi', 'pca')
('invers', 'project')
('anim', 'real')
('project', 'radial')
('comput', 'photographi')
('3d', 'linear')
('facial', 'express')
('human', 'face')
('reconstruct', 'princip')
('linear', 'facial')
```

N-gramas

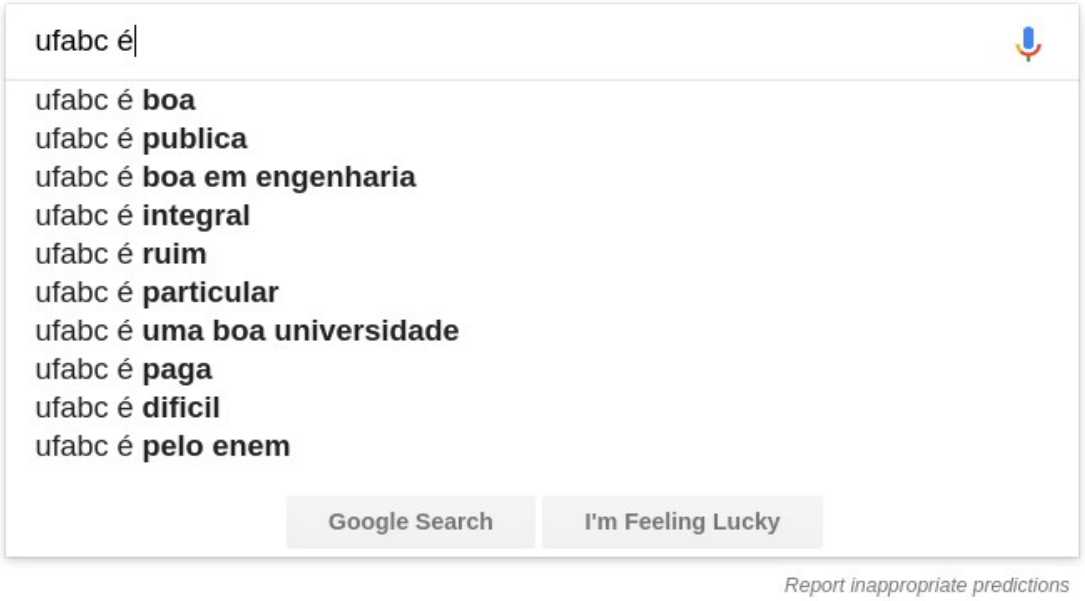
- Modelo de Markov de ordem **0**.

  Geralmente, é utilizado para **predizer** o seguinte item em um n-grama.



N-gramas

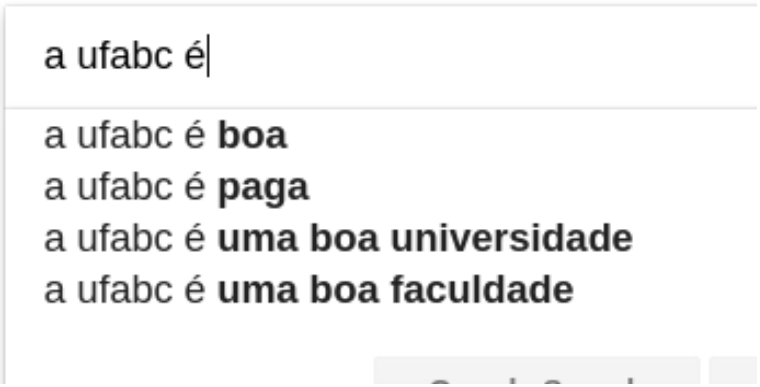
- Um modelo probabilístico pode ser utilizado para:
  - Correção ortográfica.
  - Tradução automática de textos.
  - Reconhecimento de fala.



Modelo probabílistico (estatístico)

Modelo de linguagem (*language model*)

- Um modelo que **atribue probabilidades** a uma **sequência** de palavras é denominado **Modelo de linguagem**.
- Um modelo de linguagem **por N-grama** permite predizer o seguinte item de uma sequência usando um **Modelo de Markov de ordem (N-1).**



- São modelos simples e escaláveis.

Modelo de linguagem (*language model*)

- Probabilidade de uma palavra ***w*** dado um histório ***h***:

$$P(w|h)$$

- Exemplo:

$$P(\text{uma}|\text{a ufabc é})$$

Modelo de linguagem (*language model*)

- Probabilidade de uma palavra ***w*** dado um histório ***h***:

$$P(w|h)$$

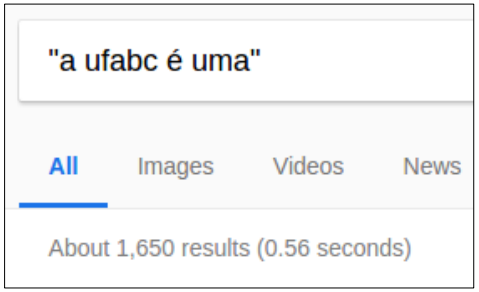
- Exemplo:

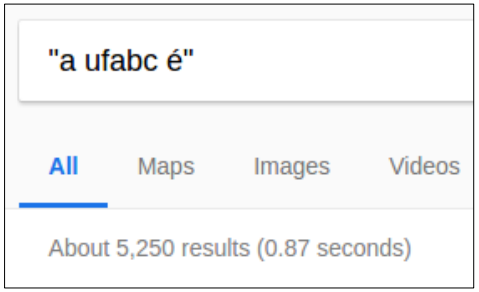
$$P(\text{uma}|\text{a ufabc é})$$

$$P(\text{uma}|\text{a ufabc é}) = \frac{C(\text{a ufabc é uma})}{C(\text{a ufabc é})}$$

Pode ser utilizado um Corpus para contar o número de vezes que w e h aparecem

Modelo de linguagem (*language model*)



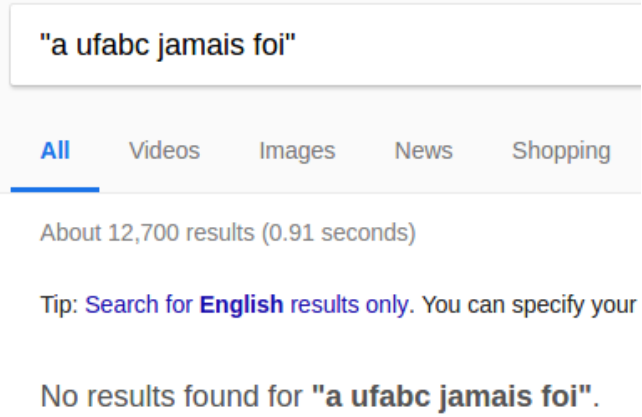


$$P(\text{uma}|\text{a ufabc é}) = \frac{C(\text{a ufabc é uma})}{C(\text{a ufabc é})}$$

$$P(\text{uma}|\text{a ufabc é}) = \frac{1650}{5250} = 0.314285714$$

Modelo de linguagem (*language model*)

- A linguagem humana é criativa e, independente do tamanho do corpus, muitas sequências podem não ser identificadas.



- C("a ufabc jamais foi") = 0

Modelo de linguagem (*language model*)

- Um modelo que calcule $P(W)$ ou $P(w_n|w_1, w_2, \ldots, w_{n-1})$

  é denominado modelo de linguagem.

$$P(W) = P(w_1, w_2, w_3, \ldots, w_n)$$

W é uma frase composta pelas palavras w_{is}

$$P(w_4|w_1, w_2, w_3)$$

Um pouco de formalização

Probabilidade condicional

$$P(B|A) = \frac{P(A,B)}{P(B)}$$

$$P(A,B) = P(A)P(B|A)$$

- Considerando mais variáveis

$$P(A,B,C) = P(A)P(B|A)P(C|A,B)$$

$$P(A,B,C,D) = P(A)P(B|A)P(C|A,B)P(D|A,B,C)$$

$$P(x_1,x_2,x_3,\ldots,x_n) = P(x_1)P(x_2|x_1)P(x_3|x_1,x_2)\ldots P(x_n|x_1,\ldots,x_{n-1})$$

Probabilidade de palavras em uma frase

$$P(x_1, x_2, x_3, \ldots, x_n) = P(x_1)P(x_2|x_1)P(x_3|x_1, x_2)\ldots P(x_n|x_1, \ldots, x_{n-1})$$

$$P(w_1, w_2, \ldots, w_n) = \prod_{i=1}^{n} P(w_i|w_1, \ldots, w_{i-1})$$

$$P(\text{a ufabc é uma}) = P(\text{a}) \times P(\text{ufabc}|\text{a}) \times P(\text{é}|\text{a ufabc}) \times P(\text{uma}|\text{a ufabc é})$$

Probabilidade de palavras em uma frase

$$P(\text{a ufabc é uma}) = P(\text{a}) \times P(\text{ufabc}|\text{a}) \times P(\text{é}|\text{a ufabc}) \times P(\text{uma}|\text{a ufabc é})$$

$$P(\text{uma}|\text{a ufabc é}) = \frac{C(\text{a ufabc é uma})}{C(\text{a ufabc é})}$$

- Muitas possibilidades de arranjos de palavras.
- Não é recomendável pois no corpus não teremos dados suficientes para a contagem das vezes em que a sequência aparece.

Aproximando ou simplificando...

- Podemos usar “apenas” a(s) última(s) palavra(s) para **aproximar** a medida

$$P(\text{uma}|\text{a ufabc é}) \approx P(\text{uma}|\text{é})$$

Uso de bigramas

$$P(\text{uma}|\text{a ufabc é}) \approx P(\text{uma}|\text{ufabc é})$$

Uso de trigramas

Cadeias de Andrei Markov

- **Pressuposto de Markov:**

  é a suposição que a probabilidade de uma palavra **depende apenas** da probabilidade de uma(s) palavra(s) anterior(es).

1856-1922

$$P(w_1, w_2, \ldots, w_n) = \prod_{i=1}^{n} P(w_i | w_1, \ldots, w_{i-1})$$

$$P(w_1, w_2, \ldots, w_n) \approx \prod_{i=1}^{n} P(w_i | w_{i-k}, \ldots, w_{i-1})$$

Modelos por unigrama, bigrama

$$P(w_1, w_2, \ldots, w_n) = \prod_{i=1}^{n} P(w_i)$$

$$P(w_1, w_2, \ldots, w_n) = \prod_{i=1}^{n} P(w_i|w_{i-1})$$

Modelos por unigrama, bigrama e trigrama

$$P(w_1, w_2, \ldots, w_n) = \prod_{i=1}^{n} P(w_i)$$

$$P(w_1, w_2, \ldots, w_n) = \prod_{i=1}^{n} P(w_i|w_{i-1})$$

$$P(w_1, w_2, \ldots, w_n) = \prod_{i=1}^{n} P(w_i|w_{i-2}, w_{i-1})$$

Estimando as probabilidades usando bigramas

$$P(w_1, w_2, \ldots, w_n) = \prod_{i=1}^{n} P(w_i|w_{i-1})$$

- Estimativa por **Máxima verossimilhança** (ou seja, baseados em um corpus podemos determinar as probabilidades)

$$P(w_i|w_{i-1}) = \frac{C(w_{i-1}, w_i)}{\sum_x C(w_{i-1}x)}$$

Estimando as probabilidades usando bigramas

$$P(w_1, w_2, \ldots, w_n) = \prod_{i=1}^{n} P(w_i|w_{i-1})$$

- Estimativa por **Máxima verossimilhança** (ou seja, baseados em um corpus podemos determinar as probabilidades)

$$P(w_i|w_{i-1}) = \frac{C(w_{i-1}, w_i)}{\sum_x C(w_{i-1}x)}$$

$$P(w_i|w_{i-1}) = \frac{C(w_{i-1}, w_i)}{C(w_{i-1})}$$

Todo bigrama que inicia com w_{i-1} é igual ao número de vezes que aparece w_{i-1} (unigrama)!

Exemplo com um corpus de 3 frases

<s> I am Sam </s>
<s> Sam I am </s>
<s> I do not like green eggs and ham </s>

$$P(\text{I}\,|\,\text{<s>}) = \frac{2}{3} = .67$$

$$P(\text{Sam}\,|\,\text{<s>}) = \frac{1}{3} = .33$$

Exemplo com um corpus de 3 frases

<s> I am Sam </s>
<s> Sam I am </s>
<s> I do not like green eggs and ham </s>

$P(\text{I}|\text{<s>}) = \frac{2}{3} = .67$

$P(\text{Sam}|\text{<s>}) = \frac{1}{3} = .33$

$P(\text{am}|\text{I}) = \frac{2}{3} = .67$

$P(\text{do}|\text{I}) = \frac{1}{3} = .33$

$P(\text{</s>}|\text{Sam}) = \frac{1}{2} = 0.5$

$P(\text{Sam}|\text{am}) = \frac{1}{2} = .5$

N-gramas do texto ‘Capivara’

unigrama

Ngrams Ranked by Frequency

Total number of tokens: 199 Types: 139

ngram	count	frequency
de	11	5.5276381909548
e	8	4.0201005025126
a	5	2.5125628140704
até	5	2.5125628140704
do	5	2.5125628140704
no	4	2.0100502512563
uma	4	2.0100502512563
ser	3	1.5075376884422
em	3	1.5075376884422
por	3	1.5075376884422
sexual	2	1.0050251256281
m	2	1.0050251256281
pode	2	1.0050251256281
o	2	1.0050251256281
à	2	1.0050251256281
É	2	1.0050251256281
kg	2	1.0050251256281
as	2	1.0050251256281

bigrama

Ngrams Ranked by Frequency

Total number of tokens: 199 Types: 195

ngram	count	frequency
kg e	2	1.0050251256281
m de	2	1.0050251256281
pesando até	2	1.0050251256281
de idade	2	1.0050251256281
proeminente no	1	0.50251256281407
por conta	1	0.50251256281407
conta da	1	0.50251256281407
da presença	1	0.50251256281407
presença de	1	0.50251256281407
de uma	1	0.50251256281407
uma glândula	1	0.50251256281407
glândula proeminente	1	0.50251256281407
A	1	0.50251256281407
no focinho	1	0.50251256281407
machos por	1	0.50251256281407
apesar do	1	0.50251256281407
do dimorfismo	1	0.50251256281407
dimorfismo sexual	1	0.50251256281407

http://guidetodatamining.com/ngramAnalyzer/

N-gramas do texto ‘Capivara’

trigrama

Ngrams Ranked by Frequency

Total number of tokens: 199 Types: 199

ngram	count	frequency
A	1	0.50251256281407
focinho apesar do	1	0.50251256281407
por conta da	1	0.50251256281407
conta da presença	1	0.50251256281407
da presença de	1	0.50251256281407
presença de uma	1	0.50251256281407
de uma glândula	1	0.50251256281407
uma glândula proeminente	1	0.50251256281407
glândula proeminente no	1	0.50251256281407
proeminente no focinho	1	0.50251256281407
no focinho apesar	1	0.50251256281407
apesar do dimorfismo	1	0.50251256281407
os machos por	1	0.50251256281407
do dimorfismo sexual	1	0.50251256281407
dimorfismo sexual não	1	0.50251256281407
sexual não ser	1	0.50251256281407
não ser aparente	1	0.50251256281407
ser aparente Existe	1	0.50251256281407

4-grama

Ngrams Ranked by Frequency

Total number of tokens: 199 Types: 199

ngram	count	frequency
A	1	0.50251256281407
no focinho apesar do	1	0.50251256281407
machos por conta da	1	0.50251256281407
por conta da presença	1	0.50251256281407
conta da presença de	1	0.50251256281407
da presença de uma	1	0.50251256281407
presença de uma glândula	1	0.50251256281407
de uma glândula proeminente	1	0.50251256281407
uma glândula proeminente no	1	0.50251256281407
glândula proeminente no focinho	1	0.50251256281407
proeminente no focinho apesar	1	0.50251256281407
focinho apesar do dimorfismo	1	0.50251256281407
distinguir os machos por	1	0.50251256281407
apesar do dimorfismo sexual	1	0.50251256281407
do dimorfismo sexual não	1	0.50251256281407
dimorfismo sexual não ser	1	0.50251256281407
sexual não ser aparente	1	0.50251256281407
não ser aparente Existe	1	0.50251256281407

http://guidetodatamining.com/ngramAnalyzer/

Outro exemplo: Berkeley Restaurant Project

THE BERKELEY RESTAURANT PROJECT

Daniel Jurafsky, Chuck Wooters,* Gary Tajchman,
Jonathan Segal, Andreas Stolcke, Eric Fosler, and Nelson Morgan

International Computer Science Institute
1947 Center Street, Suite 600
Berkeley, CA 94704, USA
& University of California at Berkeley†

ABSTRACT

This paper describes the architecture and performance of the Berkeley Restaurant Project (BeRP), a medium-vocabulary, speaker-independent, spontaneous continuous speech understanding system currently under development at ICSI. BeRP serves as a testbed for a number of our speech-related research projects, including robust feature extraction, connectionist phonetic likelihood estimation, automatic induction of multiple-pronunciation lexicons, foreign accent detection and modeling, advanced language models, and lip-reading. In addition, it has proved quite usable in its function as a database frontend, even though many of our subjects are non-native speakers of English.

1 OVERVIEW

The BeRP system functions as a knowledge consultant whose domain is restaurants in the city of Berkeley, California. As a knowledge consultant, it draws inspiration from earlier consultants like VOYAGER [15]. Users ask spoken language questions of BeRP, which directs questions to the user and then queries a database of restaurants and gives advice to the user, based on such use criteria as cost, type of food, and location.

The BeRP recognizer consists of six components: the RASTA-PLP *feature extractor*, a multilayer perceptron (MLP) *phonetic likelihood estimator*, a *Viterbi decoder* called Y_0, an HMM pronunciation *lexicon*, a bigram or SCFG *Language Model (LM)* and the *natural language backend*, including a database of restaurants. The whole system runs on a SPARCstation, although for speed we usually offload the phonetic likelihood estimation (the MLP forward pass) to special purpose hardware. Figure 1 gives an overview of the architecture.

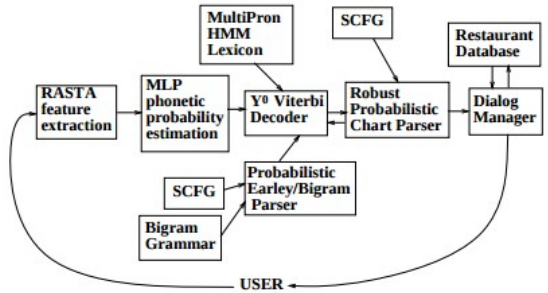


Figure 1: The BeRP Architecture

Training Corpus	4786 sentences + TIMIT	
Test Corpus	563 sentences	
Vocabulary	1274 words	
Data Base	1 database table, 150 restaurants	
Bigram	Perplexity 10.7 with 77% coverage	
Grammar	1389 handwritten SCFG rules	
Implementation	18,000 lines of C++	
Performance	Recognition	32.1% error
	Parsing	63% training 61% test
	Understanding	34% error

Figure 2: BeRP Status in June 1994

Sistema de consulta sobre os restaurantes da Universidade de Berkeley

International Conference on Spoken Language Processing - 1994

Outro exemplo: Berkeley Restaurant Project

can you tell me about any good cantonese restaurants close by
mid priced thai food is what i'm looking for
tell me about chez panisse
can you give me a listing of the kinds of food that are available
i'm looking for a good place to eat breakfast
when is caffe venezia open during the day

Ao todo, 9332 frases.

Alguns bigramas extraídos de 9332 frases

	i	want	to	eat	chinese	food	lunch	spend
i	5	827	0	9	0	0	0	2
want	2	0	608	1	6	6	5	1
to	2	0	4	686	2	0	6	211
eat	0	0	2	0	16	2	42	0
chinese	1	0	0	0	0	82	1	0
food	15	0	15	0	1	4	0	0
lunch	2	0	0	0	0	1	0	0
spend	1	0	1	0	0	0	0	0

Alguns bigramas extraídos de 9332 frases

	i	want	to	eat	chinese	food	lunch	spend
i	5	827	0	9	0	0	0	2
want	2	0	608	1	6	6	5	1
to	2	0	4	686	2	0	6	211
eat	0	0	2	0	16	2	42	0
chinese	1	0	0	0	0	82	1	0
food	15	0	15	0	1	4	0	0
lunch	2	0	0	0	0	1	0	0
spend	1	0	1	0	0	0	0	0

Unigramas

i	want	to	eat	chinese	food	lunch	spend
2533	927	2417	746	158	1093	341	278

Probabilidades calculadas para o corpus

	i	want	to	eat	chinese	food	lunch	spend
i	0.002	0.33	0	0.0036	0	0	0	0.00079
want	0.0022	0	0.66	0.0011	0.0065	0.0065	0.0054	0.0011
to	0.00083	0	0.0017	0.28	0.00083	0	0.0025	0.087
eat	0	0	0.0027	0	0.021	0.0027	0.056	0
chinese	0.0063	0	0	0	0	0.52	0.0063	0
food	0.014	0	0.014	0	0.00092	0.0037	0	0
lunch	0.0059	0	0	0	0	0.0029	0	0
spend	0.0036	0	0.0036	0	0	0	0	0

Probabilidades calculadas para o corpus

	i	want	to	eat	chinese	food	lunch	spend
i	0.002	0.33	0	0.0036	0	0	0	0.00079
want	0.0022	0	0.66	0.0011	0.0065	0.0065	0.0054	0.0011
to	0.00083	0	0.0017	0.28	0.00083	0	0.0025	0.087
eat	0	0	0.0027	0	0.021	0.0027	0.056	0
chinese	0.0063	0	0	0	0	0.52	0.0063	0
food	0.014	0	0.014	0	0.00092	0.0037	0	0
lunch	0.0059	0	0	0	0	0.0029	0	0
spend	0.0036	0	0.0036	0	0	0	0	0

P(<s> I want english food </s>) =

P(I|<s>)	0.25
× P(want|I)	0.33
× P(english|want)	0.0011
× P(food|english)	0.5
× P(</s>|food)	0.68
= 0.000031	

Probabilidades calculadas para o corpus

	i	want	to	eat	chinese	food	lunch	spend
i	0.002	0.33	0	0.0036	0	0	0	0.00079
want	0.0022	0	0.66	0.0011	0.0065	0.0065	0.0054	0.0011
to	0.00083	0	0.0017	0.28	0.00083	0	0.0025	0.087
eat	0	0	0.0027	0	0.021	0.0027	0.056	0
chinese	0.0063	0	0	0	0	0.52	0.0063	0
food	0.014	0	0.014	0	0.00092	0.0037	0	0
lunch	0.0059	0	0	0	0	0.0029	0	0
spend	0.0036	0	0.0036	0	0	0	0	0

```
P(I|<s>)
× P(want|I)
× P(english|want)
× P(food|english)
× P(</s>|food)
= 0.000031
```

Atividade em aula

Atividade 1

9 unigramas

Atividade 1

10 bigramas

	ngram
<s>	1
	1 2
	2 3
	3 4
	4 5
	5 6
	6 7
	7 8
	8 9
	9 </s>

Atividade 1

11 trigramas

	ngram
<s> <s>	1
<s>	1 2
	1 2 3
	2 3 4
	3 4 5
	4 5 6
	5 6 7
	6 7 8
	7 8 9
	8 9 </s>
	9 </s> </s>

4-gramas do texto ‘Capivara’

A capivara (nome científico: Hydrochoerus hydrochaeris) é uma espécie de mamífero roedor da família Caviidae e subfamília Hydrochoerinae. Alguns autores consideram que deva ser classificada em uma família própria. Está incluída no mesmo grupo de roedores ao qual se classificam as pacas, cutias, os preás e o porquinho-da-índia. Ocorre por toda a América do Sul ao leste dos Andes em habitats associados a rios, lagos e pântanos, do nível do mar até 1 300 m de altitude. Extremamente adaptável, pode ocorrer em ambientes altamente alterados pelo ser humano.

<s><s><s>	A	1	0.50251256281407
<s><s>	A capivara	1	0.50251256281407
<s>	A capivara nome	1	0.50251256281407
	A capivara nome científico	1	0.50251256281407

http://guidetodatamining.com/ngramAnalyzer/

Modelos por unigrama, bigrama e trigrama

$$P(w_1, w_2, \ldots, w_n) = \prod_{i=1}^{n} P(w_i)$$

$$P(w_1, w_2, \ldots, w_n) = \prod_{i=1}^{n} P(w_i|w_{i-1})$$

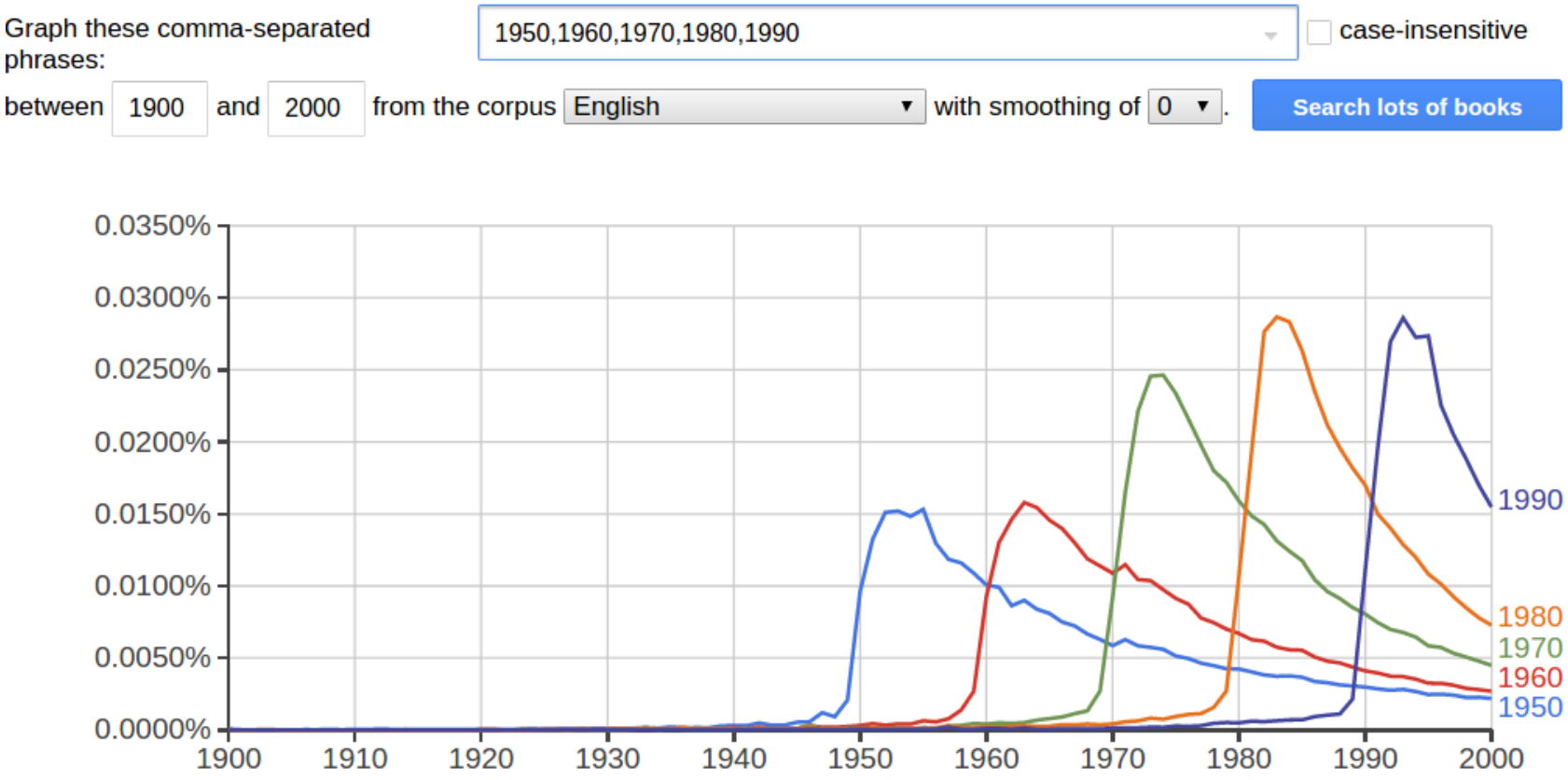
$$P(w_i|w_{i-1}) = \frac{C(w_{i-1}, w_i)}{C(w_{i-1})}$$

$$P(w_1, w_2, \ldots, w_n) = \prod_{i=1}^{n} P(w_i|w_{i-2}, w_{i-1})$$

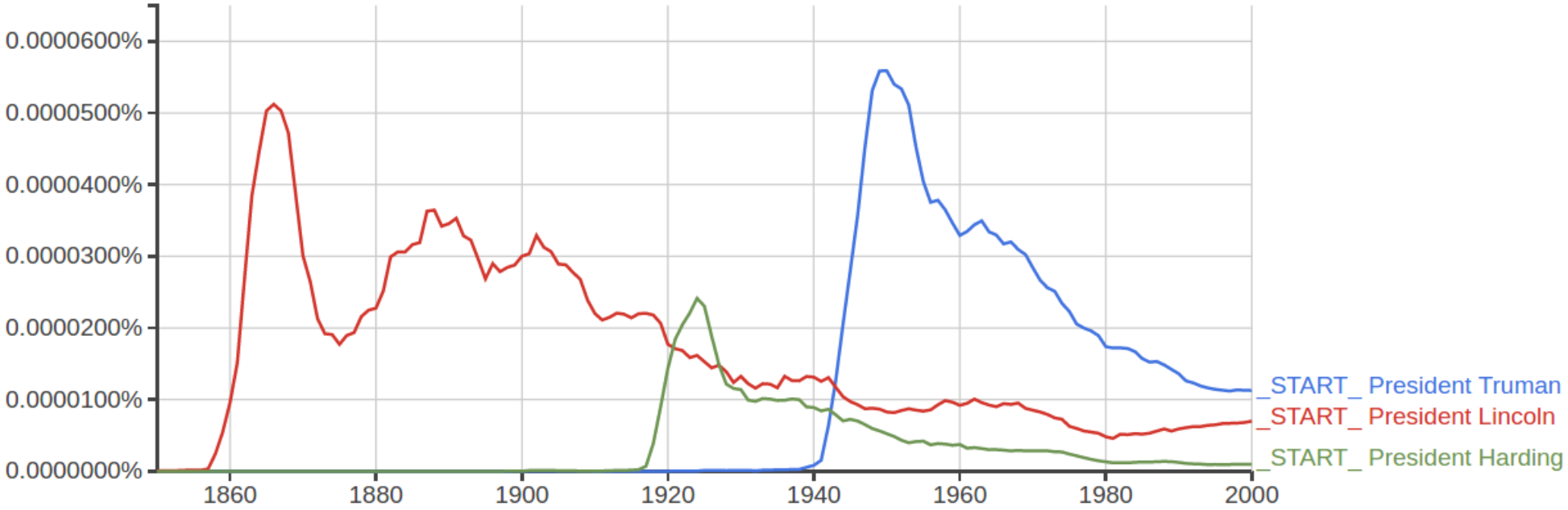
$$P(w_n|w_{n-1}, w_{n-2}) = \frac{C(w_{n-2}w_{n-1}w_n)}{C(w_{n-2}w_{n-1})}$$

Considerações finais

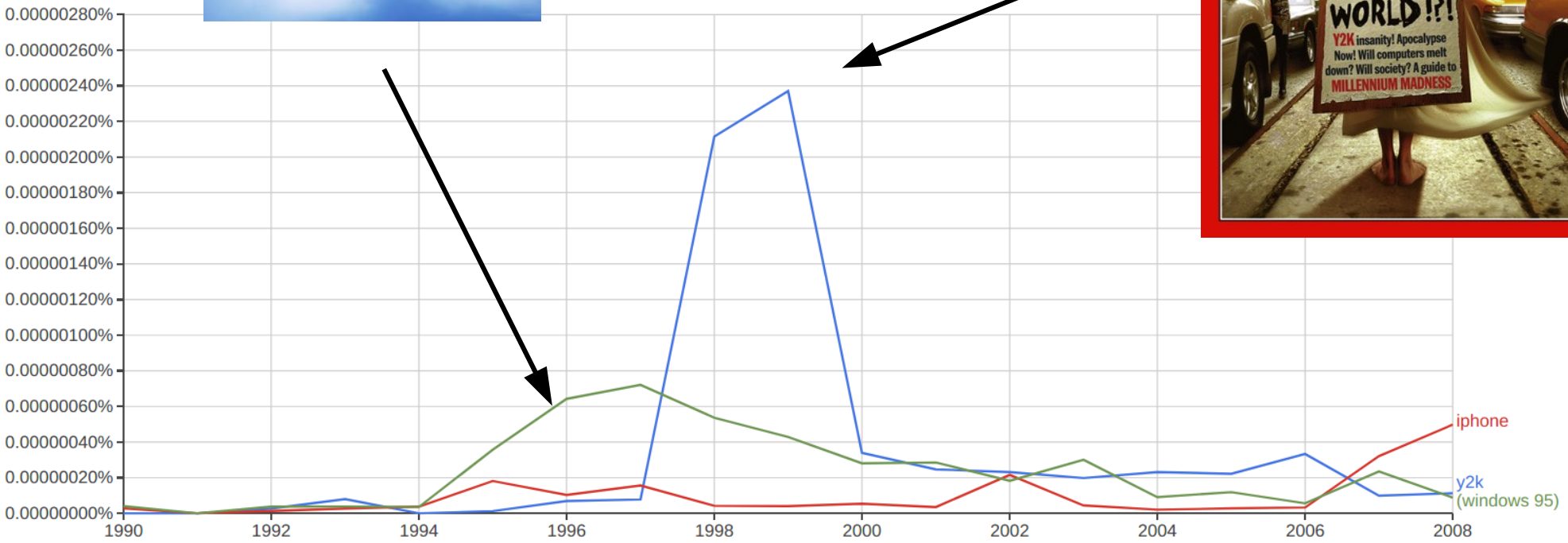
1) Podemos identificar padrões?



1) Podemos identificar padrões?



1) Podemos identificar padrões?



2) N-gramas e sinais de pontuação

- Os sinais de pontuação geralmente não são considerados na análise por N-gramas.

A capivara (nome científico: Hydrochoerus hydrochaeris) é uma espécie de mamífero roedor da família Caviidae e subfamília Hydrochoerinae. Alguns autores consideram que deva ser classificada em uma família própria. Está incluída no mesmo grupo de roedores ao qual se classificam as pacas, cutias, os preás e o porquinho-da-índia. Ocorre por toda a América do Sul ao leste dos Andes em habitats associados a rios, lagos e pântanos, do nível do mar até 1 300 m de altitude. Extremamente adaptável, pode ocorrer em ambientes altamente alterados pelo ser humano www

Total number of tokens: 90 Types: 77

ngram	count	frequency
e	3	3.3333333333333
em	3	3.3333333333333
do	3	3.3333333333333
de	3	3.3333333333333
a	2	2.2222222222222
uma	2	2.2222222222222
ao	2	2.2222222222222

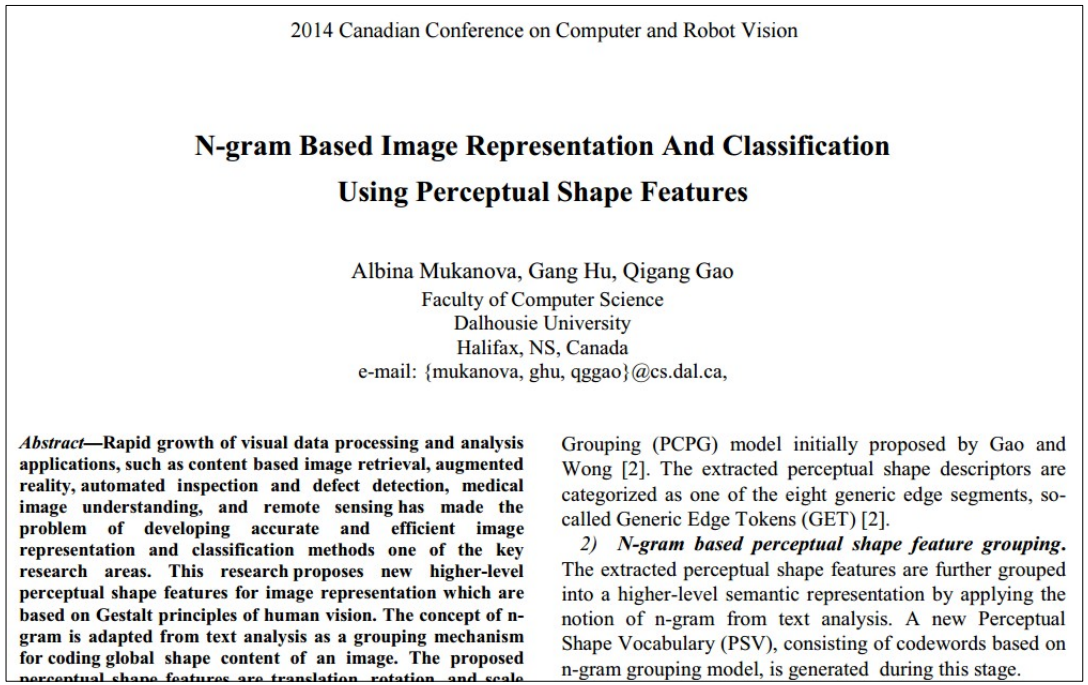
Total number of tokens: 90 Types: 90

ngram	count	frequency
A	1	1.1111111111111
do Sul ao leste dos	1	1.1111111111111
associados a rios lagos e	1	1.1111111111111
habitats associados a rios lagos	1	1.1111111111111
em habitats associados a rios	1	1.1111111111111
Andes em habitats associados a	1	1.1111111111111
dos Andes em habitats associados	1	1.1111111111111

3) N-gramas e imagens?

- N-gramas podem ser utilizadas para diferentes contextos: Por exemplo imagens satelitais:

  Utilizada para determinar a que parte da terra pertence uma determinada imagem.



2014 Canadian Conference on Computer and Robot Vision

N-gram Based Image Representation And Classification Using Perceptual Shape Features

Albina Mukanova, Gang Hu, Qigang Gao
Faculty of Computer Science
Dalhousie University
Halifax, NS, Canada
e-mail: {mukanova, ghu, qggao}@cs.dal.ca,

***Abstract*—Rapid growth of visual data processing and analysis applications, such as content based image retrieval, augmented reality, automated inspection and defect detection, medical image understanding, and remote sensing has made the problem of developing accurate and efficient image representation and classification methods one of the key research areas. This research proposes new higher-level perceptual shape features for image representation which are based on Gestalt principles of human vision. The concept of n-gram is adapted from text analysis as a grouping mechanism for coding global shape content of an image. The proposed**

Grouping (PCPG) model initially proposed by Gao and Wong [2]. The extracted perceptual shape descriptors are categorized as one of the eight generic edge segments, so-called Generic Edge Tokens (GET) [2].

*2) **N-gram based perceptual shape feature grouping.*** The extracted perceptual shape features are further grouped into a higher-level semantic representation by applying the notion of n-gram from text analysis. A new Perceptual Shape Vocabulary (PSV), consisting of codewords based on n-gram grouping model, is generated during this stage.

4) Uma questão prática

P(<s> I want english food </s>) =

P(I|<s>)	0.25
× P(want|I)	0.33
× P(english|want)	0.0011
× P(food|english)	0.5
× P(</s>|food)	0.68
= **0.000031**	

- Podemos ter números muito pequenos para representar.
- Sugestão: Utilizar Log!

$$\log(p_1 \times p_2 \times p_3 \times p_4) = \log p_1 + \log p_2 + \log p_3 + \log p_4$$

$$p_1 \times p_2 \times p_3 \times p_4 = \exp(\log p_1 + \log p_2 + \log p_3 + \log p_4)$$