Q&A

Luciano Barbosa

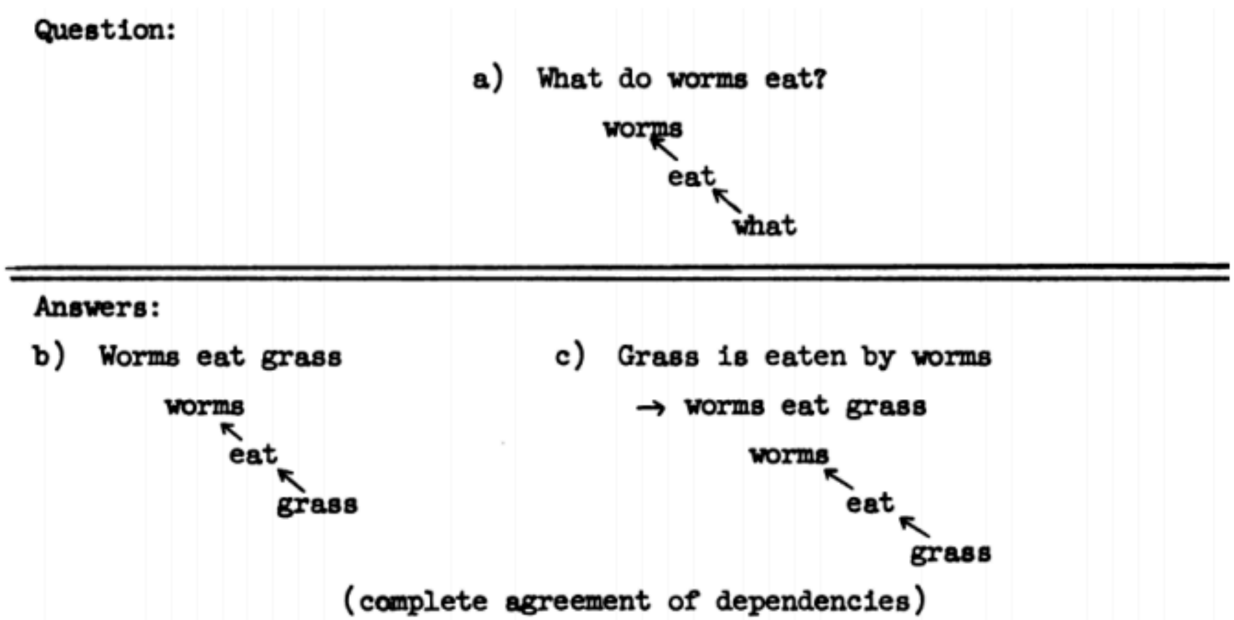
(baseado nos slides do curso de PLN de Stanford e livro Speech and Language Processing)

Centro de Informática UFPE

Universidade Federal de Pernambuco

Objetivo

- Responder perguntar feitas por humanos em linguagem natural
- Primeiros sistemas criados nos anos 60 (Simmons et al., 1964)



Taxonomia

- Corpus de onde extrair as respostas
  - Sentenças, documentos Web, bases de conhecimento, tabelas, imagens etc
- Tipo de pergunta
  - Factoide vs não factoide, domínio aberto
- Tipo de resposta
  - Segmento de texto curto, parágrafo, uma lista, sim/não

Aplicações

Google

Where is the deepest lake in the world?

All Maps Images News Videos More Settings Tools

About 21,100,000 results (0.71 seconds)

Siberia

Lake **Baikal**, in Siberia, holds the distinction of being both the deepest lake in the world and the largest freshwater lake, holding more than 20% of the unfrozen fresh water on the surface of Earth.

Aplicações

Google

How can I protect myself from COVID-19?

All | Images | News | Shopping | Videos | More | Settings | Tools

The best way to prevent illness is to avoid being exposed to this virus. Learn how COVID-19 spreads and practice these actions to help prevent the spread of this illness.

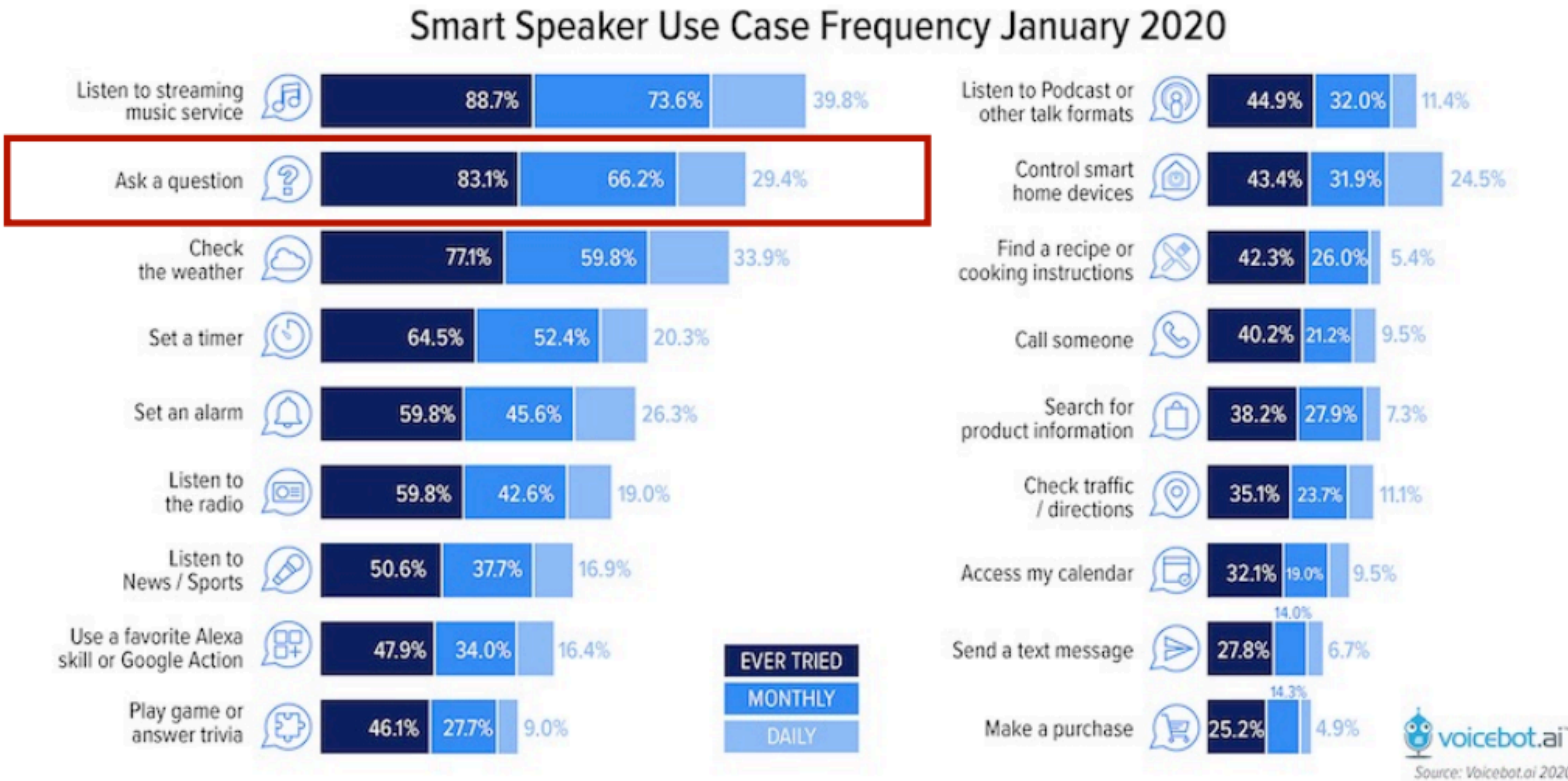
To help prevent the spread of COVID-19:

- Cover your mouth and nose with a mask when around people who don't live with you. Masks work best when everyone wears one.
- Stay at least 6 feet (about 2 arm lengths) from others.
- Avoid crowds. The more people you are in contact with, the more likely you are to be exposed to COVID-19.
- Avoid unventilated indoor spaces. If indoors, bring in fresh air by opening windows and doors.
- Clean your hands often, either with soap and water for 20 seconds or a hand sanitizer that contains at least 60% alcohol.
- Get vaccinated against COVID-19 when it's your turn.
- Avoid close contact with people who are sick.
- Cover your cough or sneeze with a tissue, then throw the tissue in the trash.
- Clean and disinfect frequently touched objects and surfaces daily.

Learn more on cdc.gov

For informational purposes only. Consult your local medical authority for advice.

Aplicações

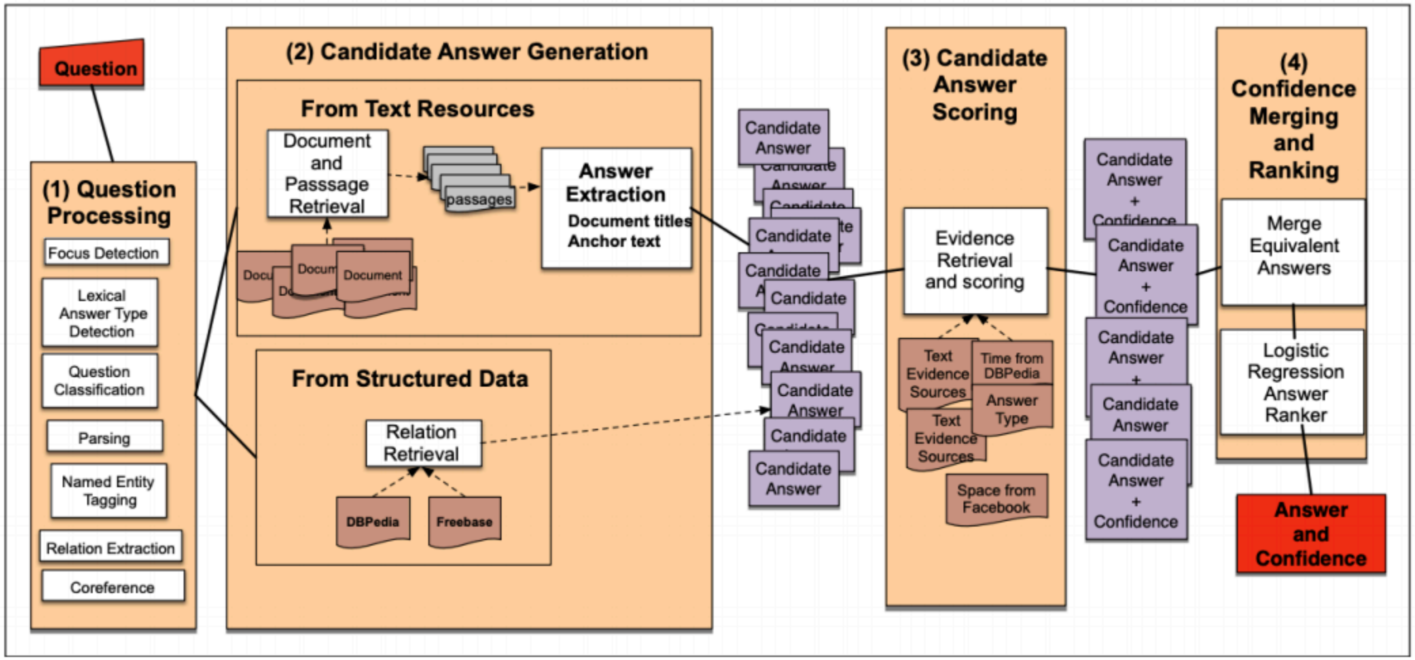


Smart Speaker Use Case Frequency January 2020
Listen to streaming music service
88.7%
73.6%
39.8%
Ask a question
83.1%
66.2%
29.4%
Check the weather
77.1%
59.8%
33.9%
Set a timer
64.5%
52.4%
20.3%
Set an alarm
59.8%
45.6%
26.3%
Listen to the radio
59.8%
42.6%
19.0%
Listen to News / Sports
50.6%
37.7%
16.9%
Use a favorite Alexa skill or Google Action
47.9%
34.0%
16.4%
Play game or answer trivia
46.1%
27.7%
9.0%
Listen to Podcast or other talk formats
44.9%
32.0%
11.4%
Control smart home devices
43.4%
31.9%
24.5%
Find a recipe or cooking instructions
42.3%
26.0%
5.4%
Call someone
40.2%
21.2%
9.5%
Search for product information
38.2%
27.9%
7.3%
Check traffic / directions
35.1%
23.7%
11.1%
Access my calendar
32.1%
19.0%
9.5%
Send a text message
27.8%
14.0%
6.7%
Make a purchase
25.2%
14.3%
4.9%
EVER TRIED
MONTHLY
DAILY
voicebot.ai
Source: Voicebot.ai 2020

IBM Watson

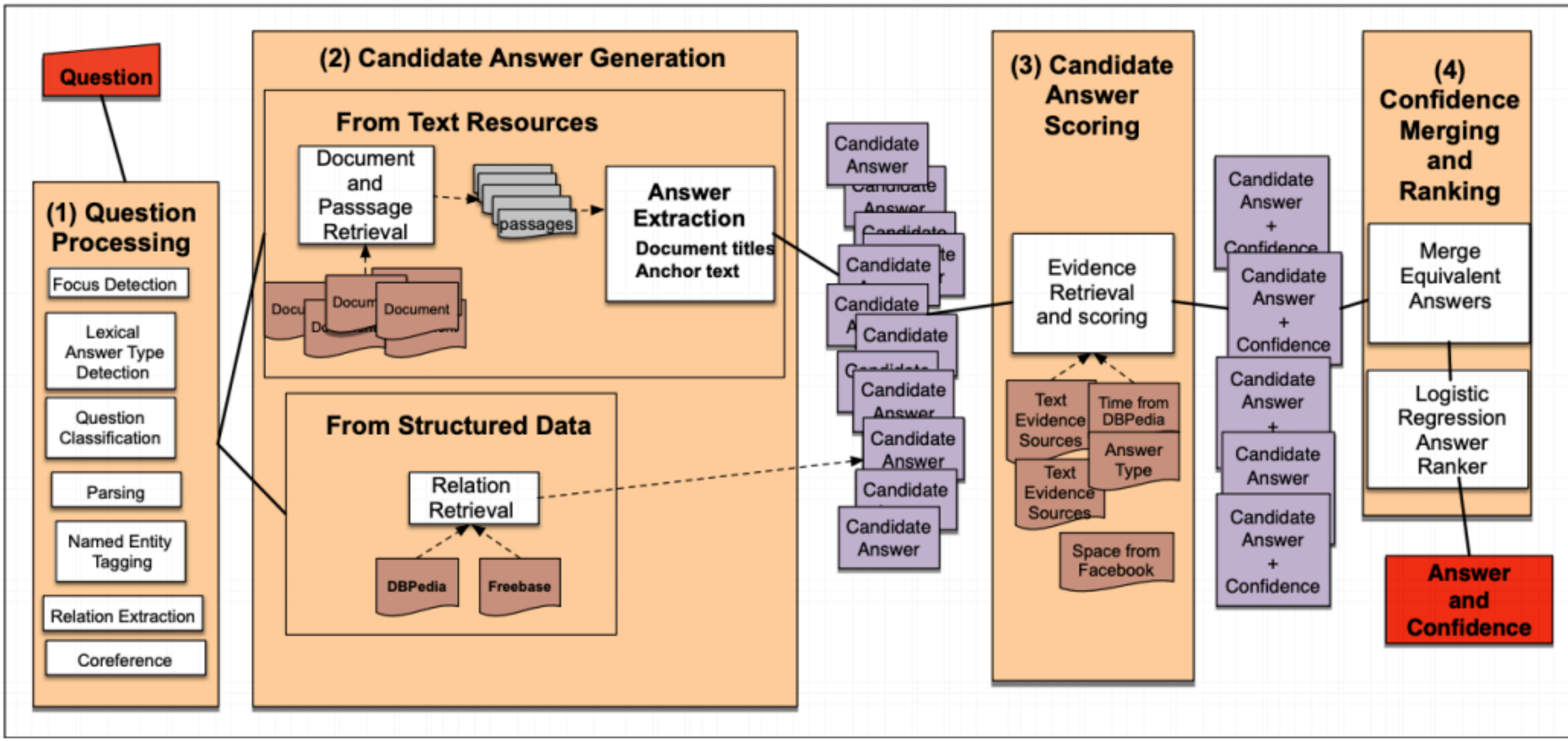
IBM Watson defeated two of Jeopardy's greatest champions in 2011

Arquitetura do IBM Watson



- Question processing
  - Detecta tipo da resposta
    - Resposta sobre pessoas, animais, data etc
    - Ex: Quem fundou Brasília?
  - Detecta tipo das perguntas
    - Definição de algo, pergunta matemática, lista etc

Arquitetura do IBM Watson



Q&A em Tabelas da Web

Table

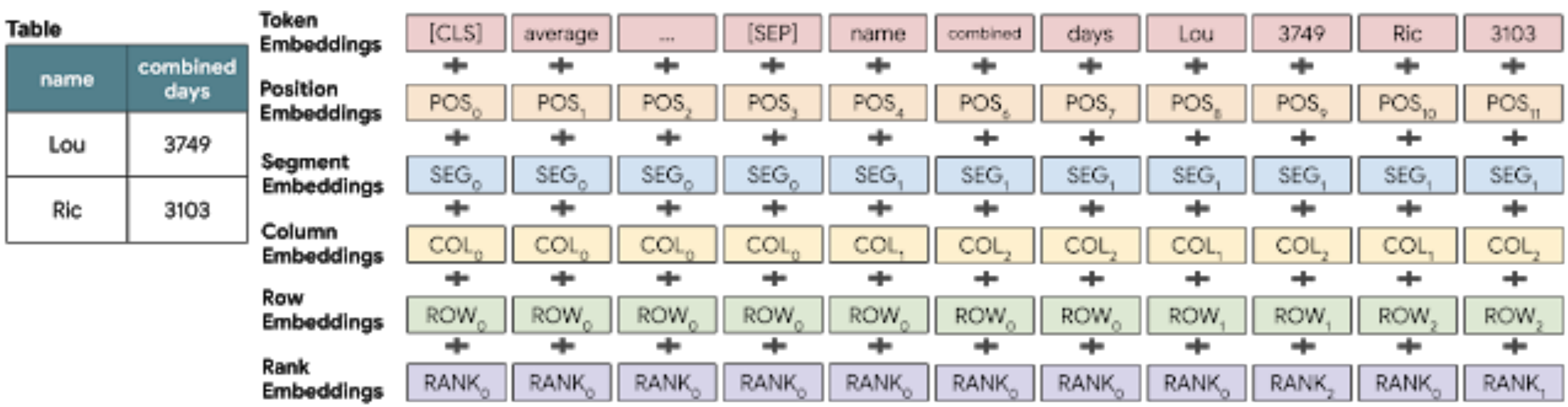
Rank	Name	No. of reigns	Combined days
1	Lou Thesz	3	3,749
2	Ric Flair	8	3,103
3	Harley Race	7	1,799
4	Dory Funk Jr.	1	1,563
5	Dan Severn	2	1,559
6	Gene Kiniski	1	1,131

Example questions

#	Question	Answer
1	*Which wrestler had the most number of reigns?*	Ric Flair
2	*Average time as champion for top 2 wrestlers?*	AVG(3749,3103)=3426
3	*How many world champions are there with only one reign?*	COUNT(Dory Funk Jr., Gene Kiniski)=2
4	*What is the number of reigns for* Harley Race?	7

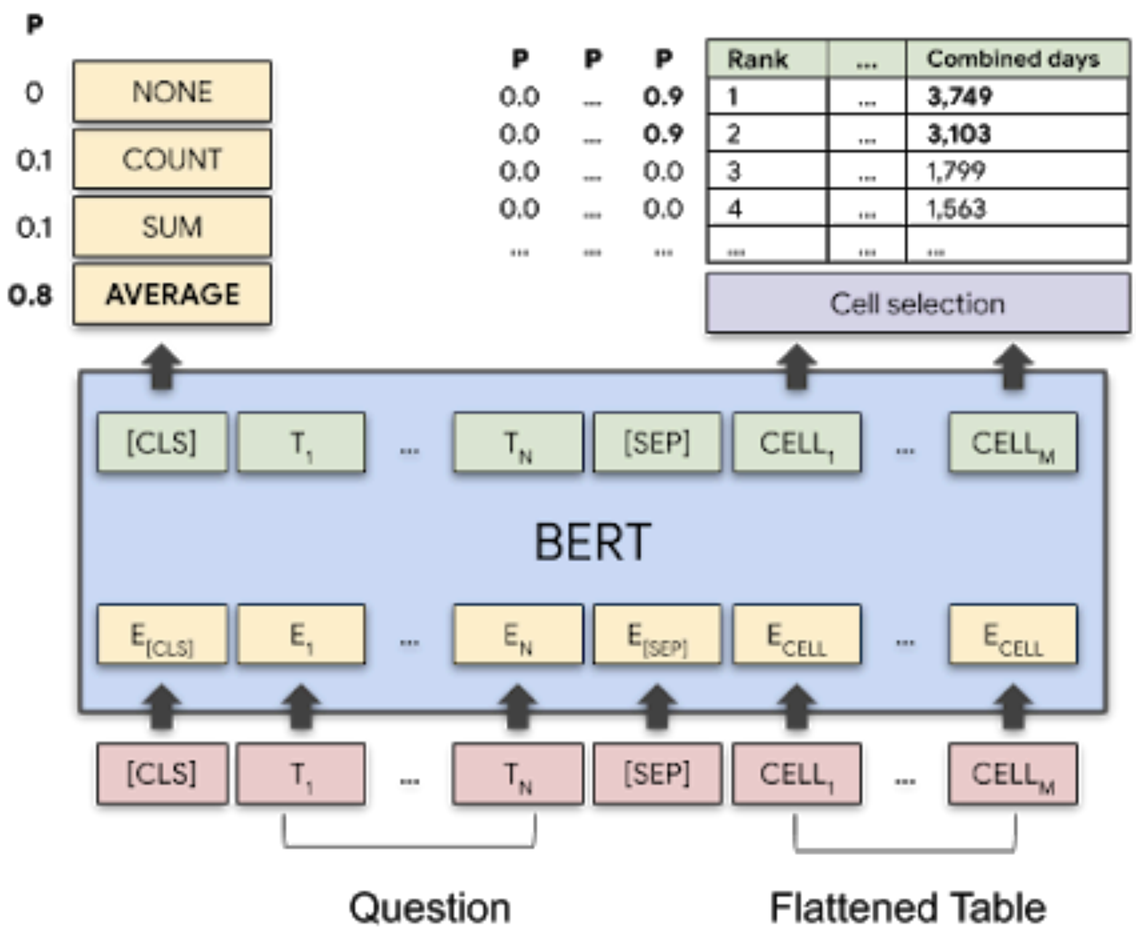
https://ai.googleblog.com/2020/04/using-neural-networks-to-find-answers.html

Q&A em Tabelas da Web



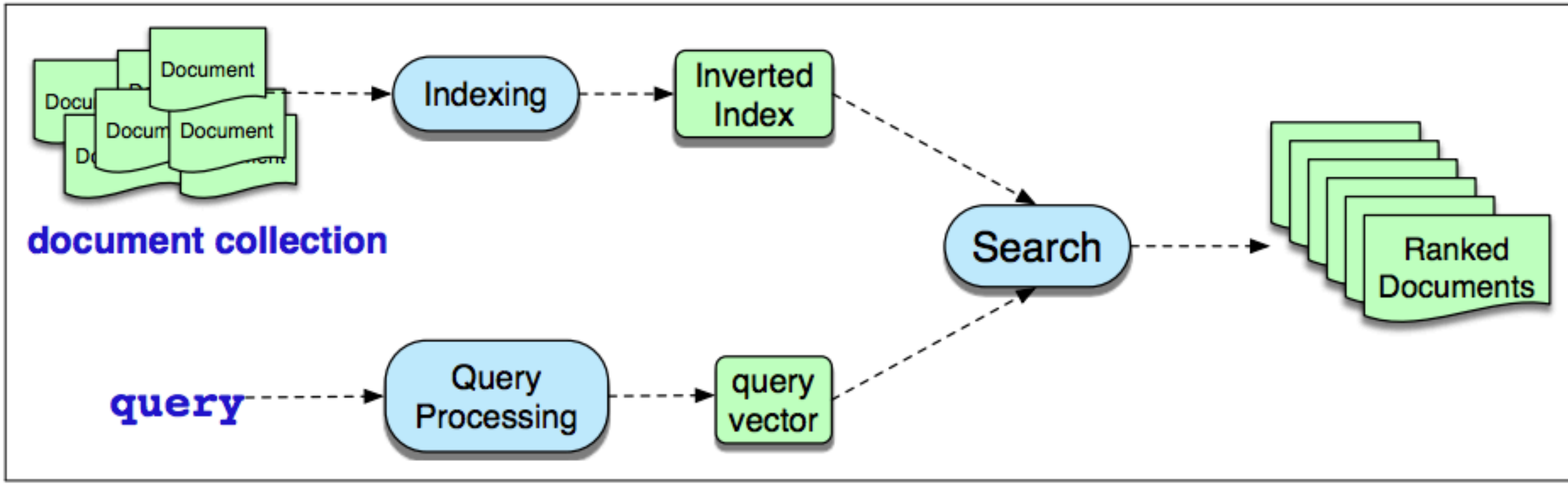
https://ai.googleblog.com/2020/04/using-neural-networks-to-find-answers.html

Q&A em Tabelas da Web



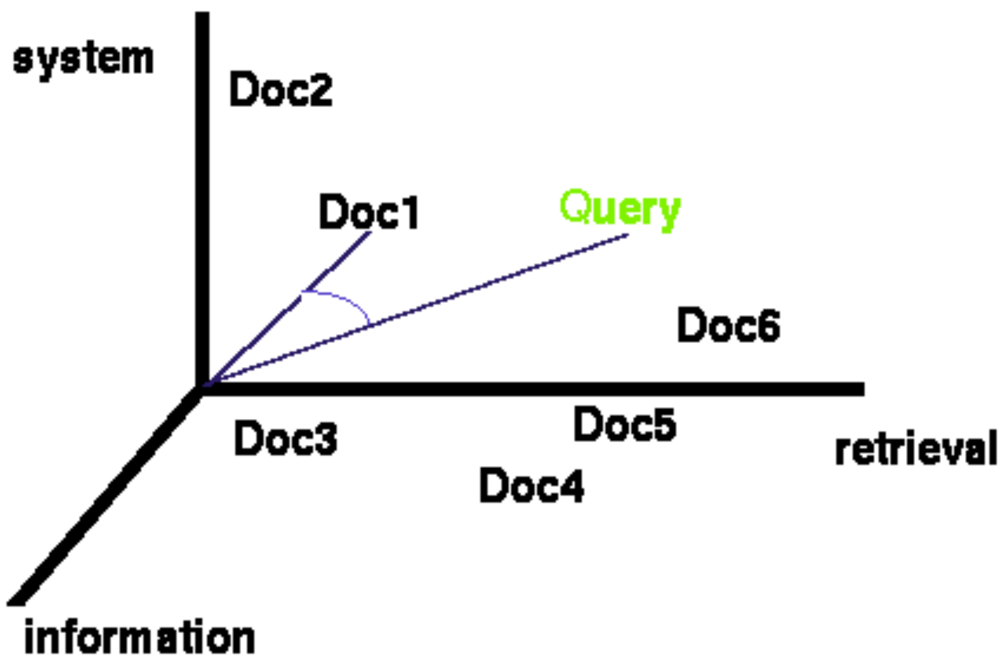
https://ai.googleblog.com/2020/04/using-neural-networks-to-find-answers.html

Q&A baseado em Recuperação de Informação



Modelo de Espaço de Vetores

- Documento e consulta representados por um vetor de palavras
- Cada palavra é uma dimensão do vetor



Modelo de Espaço de Vetores

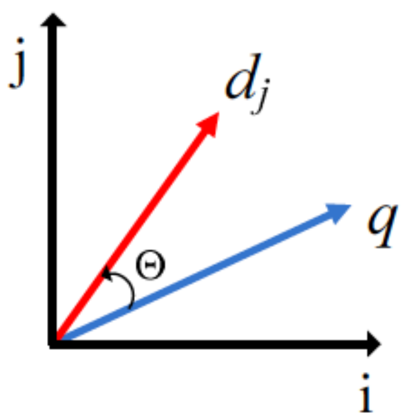
- Espaço é do tamanho do vocabulário (alta dimensão)
- Documentos são vetores esparsos
- Similaridade entre os vetores da consulta e dos documentos
  - Tamanho da intersecção
  - Jaccard:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}.$$

  - Cosseno

Similaridade de Cosseno

- Documentos ranqueados pela proximidade de pontos representando a consulta e os documentos



$$\vec{d_j} = (w_{1j}, w_{2j}, \dots, w_{tj})$$
$$\vec{q} = (w_{1q}, w_{2q}, \dots, w_{tq})$$

$$cos(\theta) = \frac{\vec{d_j} \bullet \vec{q}}{|\vec{d_j}| \times |\vec{q}|}$$

$$\boxed{sim(d_j, q) = \frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2} \times \sqrt{\sum_{j=1}^{t} w_{i,q}^2}}}$$

Cálculo da Similaridade

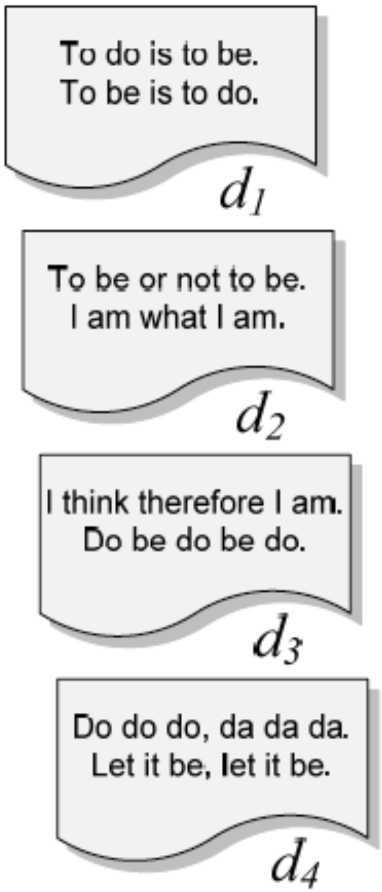
- Considere dois documentos D_1 e D_2 e uma consulta Q
  - $D_1 = (0.5, 0.8, 0.3)$, $D_2 = (0.9, 0.4, 0.2)$, $Q = (1.5, 1.0, 0)$

$$
\begin{aligned}
Cosine(D_1, Q) &= \frac{(0.5 \times 1.5) + (0.8 \times 1.0)}{\sqrt{(0.5^2 + 0.8^2 + 0.3^2)(1.5^2 + 1.0^2)}} \\
&= \frac{1.55}{\sqrt{(0.98 \times 3.25)}} = 0.87
\end{aligned}
$$

$$
\begin{aligned}
Cosine(D_2, Q) &= \frac{(0.9 \times 1.5) + (0.4 \times 1.0)}{\sqrt{(0.9^2 + 0.4^2 + 0.2^2)(1.5^2 + 1.0^2)}} \\
&= \frac{1.75}{\sqrt{(1.01 \times 3.25)}} = 0.97
\end{aligned}
$$

Exemplo: Cálculo usando TF-IDF

- Consulta: to do



doc	rank computation	rank
d_1	$\frac{1*3+0.415*0.830}{5.068}$	0.660
d_2	$\frac{1*2+0.415*0}{4.899}$	0.408
d_3	$\frac{1*0+0.415*1.073}{3.762}$	0.118
d_4	$\frac{1*0+0.415*1.073}{7.738}$	0.058

Modelo de Espaço de Vetores

- Vantagens:
  - Eficiente
  - Permite casamento parcial
  - Fácil de implementar
  - Funciona bem na prática
- Cons:
  - Assume independência dos termos
  - Sem informação semântica e sintática

Peso dos Termos

- Termos em um documento não são igualmente úteis para descrever seu conteúdo
  - Ex: palavras frequentes no documento -> importantes
  - Ex: palavras que aparecem em todos documentos da coleção -> não importantes
- Peso usado para caracterizar a importância do termo
- Útil para computar ranqueamento de documentos dada uma consulta
  - Documentos com termos da consulta com alto peso são melhores ranqueados

Frequência do Termo no Documento - TF

- Intuição: a importância do termo em um documento é proporcional à sua frequência nele

	tf weight
binary	{0,1}
raw frequency	$f_{i,j}$
log normalization	$1 + \log f_{i,j}$
double normalization 0.5	$0.5 + 0.5 \frac{f_{i,j}}{max_i f_{i,j}}$
double normalization K	$K + (1 - K) \frac{f_{i,j}}{max_i f_{i,j}}$

Frequência do Termo

- Usando a variação de tf com log

Vocabulary		$tf_{i,1}$	$tf_{i,2}$	$tf_{i,3}$	$tf_{i,4}$
1	to	3	2	-	-
2	do	2	-	2.585	2.585
3	is	2	-	-	-
4	be	2	2	2	2
5	or	-	1	-	-
6	not	-	1	-	-
7	I	-	2	2	-
8	am	-	2	1	-
9	what	-	1	-	-
10	think	-	-	1	-
11	therefore	-	-	1	-
12	da	-	-	-	2.585
13	let	-	-	-	2
14	it	-	-	-	2

Inverse Document Frequency (IDF)

- Medir a espcifidade de um termo
- Não mede a especificidade semântica de um termo
  - Depende do seu significado
  - Pode ser usado um theasurus: wordnet
  - Ex: o termo bebida é mais genérico que café ou chá
- Em RI, especifidade estatística ao invés da semântica
  - O inverso do número de documentos nos quais o termo ocorre

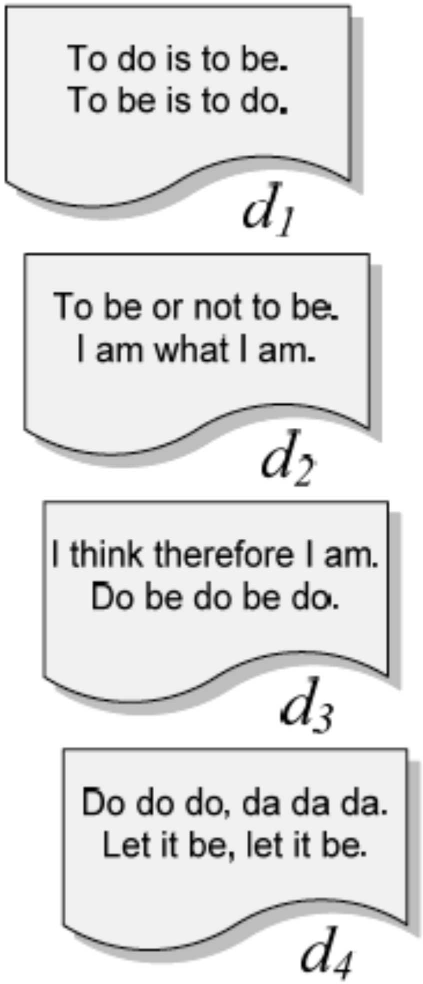
$$idf_i = \log \frac{N}{n_i}$$

- Usado amplamente em algoritmos de ranqueamento

Inverse Document Frequency (IDF): Variações

	idf weight
unary	1
inverse frequency	$\log \frac{N}{n_i}$
inv frequency smooth	$\log(1 + \frac{N}{n_i})$
inv frequeny max	$\log(1 + \frac{max_i n_i}{n_i})$
probabilistic inv frequency	$\log \frac{N - n_i}{n_i}$

Inverse Document Frequency (IDF)



	term	n_i	$idf_i = \log(N/n_i)$
1	to	2	1
2	do	3	0.415
3	is	1	2
4	be	4	0
5	or	1	2
6	not	1	2
7	I	2	1
8	am	2	1
9	what	1	2
10	think	1	2
11	therefore	1	2
12	da	1	2
13	let	1	2
14	it	1	2

TF-IDF

- Combinação do tf com o idf

$$tf_{ij} \times idf_i$$

- Variações:

weighting scheme	document term weight	query term weight
1	$f_{i,j} * \log \frac{N}{n_i}$	$(0.5 + 0.5 \frac{f_{i,q}}{max_i \ f_{i,q}}) * \log \frac{N}{n_i}$
2	$1 + \log f_{i,j}$	$\log(1 + \frac{N}{n_i})$
3	$(1 + \log f_{i,j}) * \log \frac{N}{n_i}$	$(1 + \log f_{i,q}) * \log \frac{N}{n_i}$

To do is to be.
To be is to do.

d_1

To be or not to be.
I am what I am.

d_2

I think therefore I am.
Do be do be do.

d_3

Do do do, da da da.
Let it be, let it be.

d_4

TF-IDF

		d_1	d_2	d_3	d_4
1	to	3	2	-	-
2	do	0.830	-	1.073	1.073
3	is	4	-	-	-
4	be	-	-	-	-
5	or	-	2	-	-
6	not	-	2	-	-
7	I	-	2	2	-
8	am	-	2	1	-
9	what	-	2	-	-
10	think	-	-	2	-
11	therefore	-	-	2	-
12	da	-	-	-	5.170
13	let	-	-	-	4
14	it	-	-	-	4

IDF

	term	n_i	$idf_i = \log(N/n_i)$
1	to	2	1
2	do	3	0.415
3	is	1	2
4	be	4	0
5	or	1	2
6	not	1	2
7	I	2	1
8	am	2	1
9	what	1	2
10	think	1	2
11	therefore	1	2
12	da	1	2
13	let	1	2
14	it	1	2

TF

Vocabulary		$tf_{i,1}$	$tf_{i,2}$	$tf_{i,3}$	$tf_{i,4}$
1	to	3	2	-	-
2	do	2	-	2.585	2.585
3	is	2	-	-	-
4	be	2	2	2	2
5	or	-	1	-	-
6	not	-	1	-	-
7	I	-	2	2	-
8	am	-	2	1	-
9	what	-	1	-	-
10	think	-	-	1	-
11	therefore	-	-	1	-
12	da	-	-	-	2.585
13	let	-	-	-	2
14	it	-	-	-	2

Centro de Informática UFPE

Universidade Federal de Pernambuco

To do is to be.
To be is to do.

d_1

To be or not to be.
I am what I am.

d_2

I think therefore I am.
Do be do be do.

d_3

Do do do, da da da.
Let it be, let it be.

d_4

TF-IDF

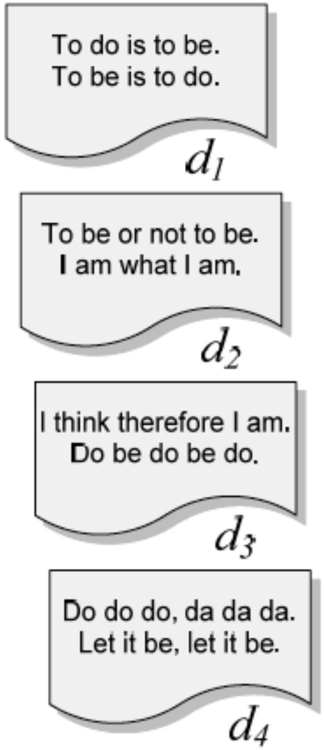
		d_1	d_2	d_3	d_4
1	to	3	2	-	-
2	do	0.830	-	1.073	1.073
3	is	4	-	-	-
4	be	-	-	-	-
5	or	-	2	-	-
6	not	-	2	-	-
7	I	-	2	2	-
8	am	-	2	1	-
9	what	-	2	-	-
10	think	-	-	2	-
11	therefore	-	-	2	-
12	da	-	-	-	5.170
13	let	-	-	-	4
14	it	-	-	-	4

IDF

	term	n_i	$idf_i = \log(N/n_i)$
1	to	2	1
2	do	3	0.415
3	is	1	2
4	be	4	0
5	or	1	2
6	not	1	2
7	I	2	1
8	am	2	1
9	what	1	2
10	think	1	2
11	therefore	1	2
12	da	1	2
13	let	1	2
14	it	1	2

TF

Vocabulary		$tf_{i,1}$	$tf_{i,2}$	$tf_{i,3}$	$tf_{i,4}$
1	to	3	2	-	-
2	do	2	-	2.585	2.585
3	is	2	-	-	-
4	be	2	2	2	2
5	or	-	1	-	-
6	not	-	1	-	-
7	I	-	2	2	-
8	am	-	2	1	-
9	what	-	1	-	-
10	think	-	-	1	-
11	therefore	-	-	1	-
12	da	-	-	-	2.585
13	let	-	-	-	2
14	it	-	-	-	2



TF-IDF

		d_1	d_2	d_3	d_4
1	to	3	2	-	-
2	do	0.830	-	1.073	1.073
3	is	4	-	-	-
4	be	-	-	-	-
5	or	-	2	-	-
6	not	-	2	-	-
7	I	-	2	2	-
8	am	-	2	1	-
9	what	-	2	-	-
10	think	-	-	2	-
11	therefore	-	-	2	-
12	da	-	-	-	5.170
13	let	-	-	-	4
14	it	-	-	-	4

IDF

	term	n_i	$idf_i = \log(N/n_i)$
1	to	2	1
2	do	3	0.415
3	is	1	2
4	be	4	0
5	or	1	2
6	not	1	2
7	I	2	1
8	am	2	1
9	what	1	2
10	think	1	2
11	therefore	1	2
12	da	1	2
13	let	1	2
14	it	1	2

TF

Vocabulary		$tf_{i,1}$	$tf_{i,2}$	$tf_{i,3}$	$tf_{i,4}$
1	to	3	2	-	-
2	do	2	-	2.585	2.585
3	is	2	-	-	-
4	be	2	2	2	2
5	or	-	1	-	-
6	not	-	1	-	-
7	I	-	2	2	-
8	am	-	2	1	-
9	what	-	1	-	-
10	think	-	-	1	-
11	therefore	-	-	1	-
12	da	-	-	-	2.585
13	let	-	-	-	2
14	it	-	-	-	2

BM25

- Um dos mais populares e efetivos algoritmos de ranqueamento
- Baseado no BIM
- 3 princípios básicos: tf, idf e normalização pelo tamanho do documento
- Criado como resultado de experimentos em variações de modelos probabilísticos
- Usado como baseline em experimentos de RI

BM25

$$\sum_{i \in Q} \underbrace{\log \frac{(r_i+0.5)/(R-r_i+0.5)}{(n_i-r_i+0.5)/(N-n_i-R+r_i+0.5)}}_{\text{BIM}} \cdot \frac{(k_1+1)f_i}{K+f_i} \cdot \frac{(k_2+1)qf_i}{k_2+qf_i}$$

Normalização pelo tamanho do documento

TF

- k1, k2 e b são parâmetros definidos empiricamente (depende da coleção)

$$K = k_1((1-b) + b \cdot \frac{dl}{avdl})$$

- dl: tamanho do documento

BM25: Exemplo

- Query with two terms, “president lincoln”, ($qf = 1$)
- No relevance information (*r and R are* zero)
- N = 500,000 documents
- *“president”* occurs in 40,000 documents (n_1 = 40, 000)
- *“lincoln”* occurs in 300 documents (n_2 = 300)
- “president” occurs 15 times in doc (f_1 = 15)
- *“lincoln”* occurs 25 times (f_2 = 25)
- document length is 90% of the average length ($dl/avdl$ = .9)
- $k_1 = 1.2$, $b = 0.75$, and $k_2 = 100$
- $K = 1.2 \cdot (0.25 + 0.75 \cdot 0.9) = 1.11$

BM25: Exemplo

$$\sum_{i \in Q} \log \frac{(r_i+0.5)/(R-r_i+0.5)}{(n_i-r_i+0.5)/(N-n_i-R+r_i+0.5)} \cdot \frac{(k_1+1)f_i}{K+f_i} \cdot \frac{(k_2+1)qf_i}{k_2+qf_i}$$

$$\begin{aligned}
BM25(Q,D) &= \\
&\log \frac{(0+0.5)/(0-0+0.5)}{(40000-0+0.5)/(500000-40000-0+0+0.5)} \\
&\times \frac{(1.2+1)15}{1.11+15} \times \frac{(100+1)1}{100+1} \\
&+ \log \frac{(0+0.5)/(0-0+0.5)}{(300-0+0.5)/(500000-300-0+0+0.5)} \\
&\times \frac{(1.2+1)25}{1.11+25} \times \frac{(100+1)1}{100+1} \\
&= \log 460000.5/40000.5 \cdot 33/16.11 \cdot 101/101 \\
&\quad + \log 499700.5/300.5 \cdot 55/26.11 \cdot 101/101 \\
&= 2.44 \cdot 2.05 \cdot 1 + 7.42 \cdot 2.11 \cdot 1 \\
&= 5.00 + 15.66 = 20.66
\end{aligned}$$

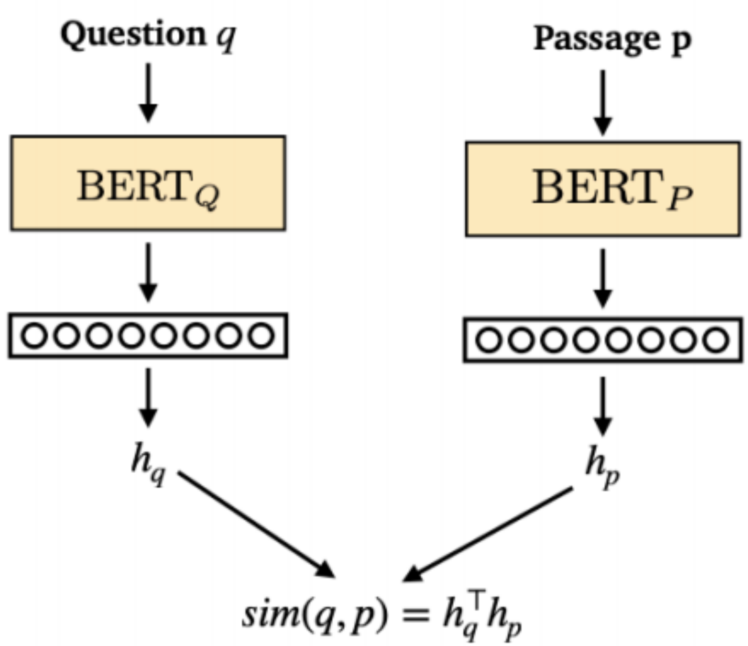
BM25: Exemplo

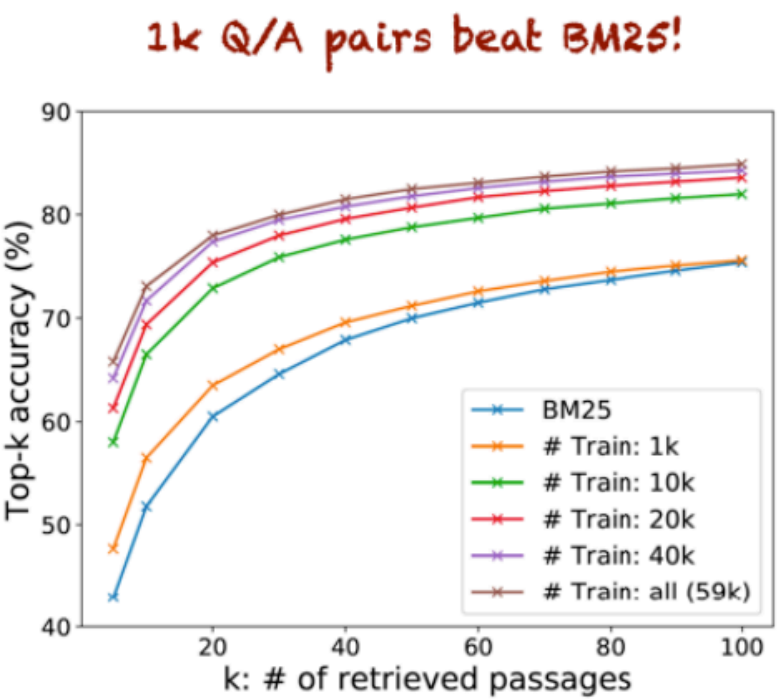
- Efeito da frequência dos termos

Frequency of "president"	Frequency of "lincoln"	BM25 score
15	25	20.66
15	1	12.74
15	0	5.00
1	25	18.2
0	25	15.66

Q&A com Vetores Densos

- Limitação de abordagens tradicionais:
  - Assume interseção entre vocabulário da pergunta e resposta
- Solução usando BERT





Q&A Extrativa

Tesla was the fourth of five children. He had an older brother named Dane and three sisters, Milka, Angelina and Marica. Dane was killed in a horse-riding accident when Nikola was five. In 1861, Tesla attended the "Lower" or "Primary" School in Smiljan where he studied German, arithmetic, and religion. In 1862, the Tesla family moved to Gospić, Austrian Empire, where Tesla's father worked as a pastor. Nikola completed "Lower" or "Primary" School, followed by the "Lower Real Gymnasium" or "Normal School."

Q: What language did Tesla study while in school?

A: German

Q&A Extrativa

Kannada language is the official language of Karnataka and spoken as a native language by about 66.54% of the people as of 2011. Other linguistic minorities in the state were Urdu (10.83%), Telugu language (5.84%), Tamil language (3.45%), Marathi language (3.38%), Hindi (3.3%), Tulu language (2.61%), Konkani language (1.29%), Malayalam (1.27%) and Kodava Takk (0.18%). In 2007 the state had a birth rate of 2.2%, a death rate of 0.7%, an infant mortality rate of 5.5% and a maternal mortality rate of 0.2%. The total fertility rate was 2.2.

Q: Which linguistic minority is larger, Hindi or Malayalam?

A: Hindi

Q&A Extrativa: Soluções

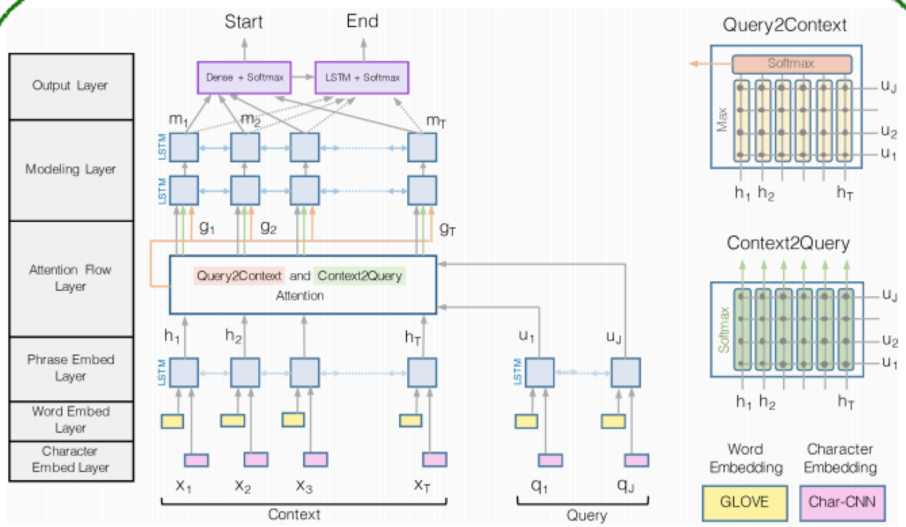


Image credit: (Seo et al, 2017)

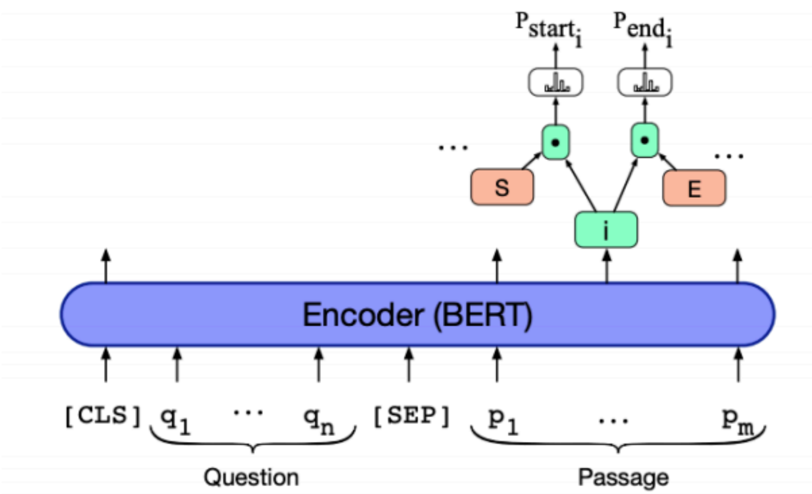


Image credit: J & M, edition 3

Q&A Extrativa: BERT

Question = Segment A

Passage = Segment B

Answer = predicting two endpoints in segment B

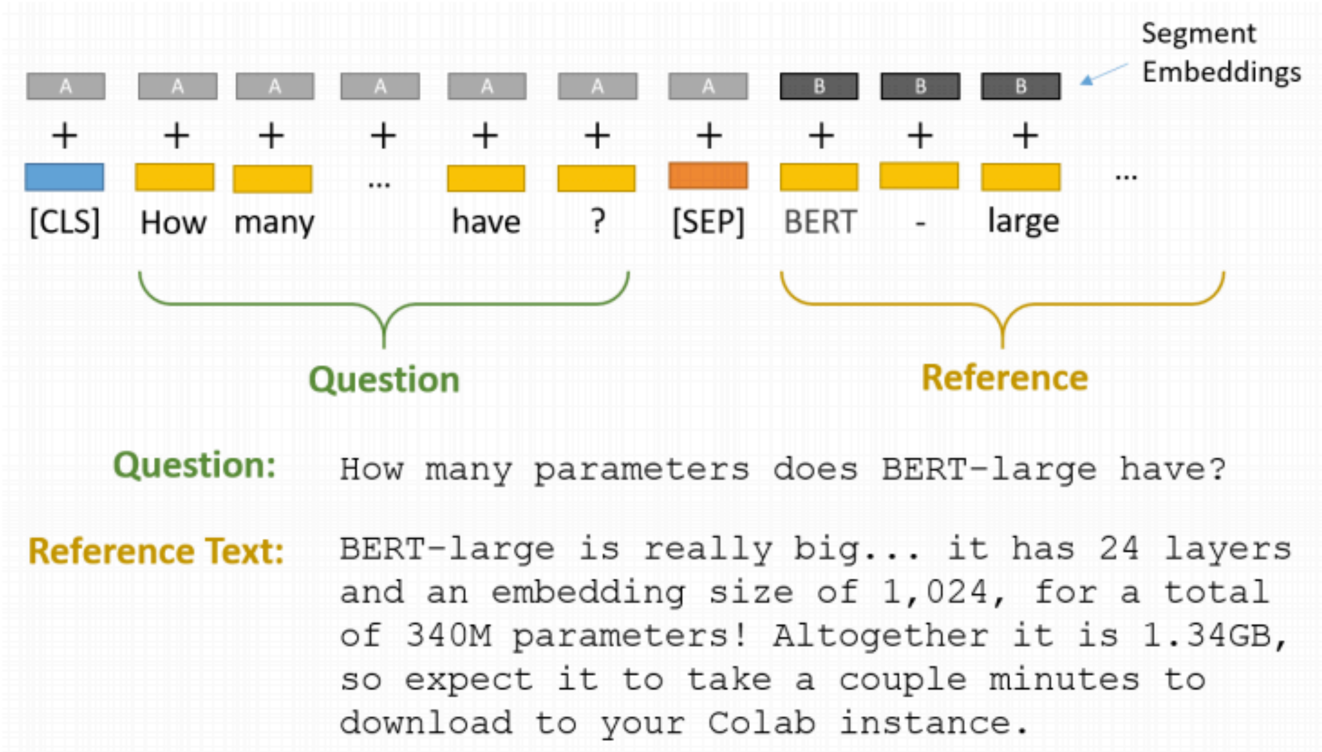
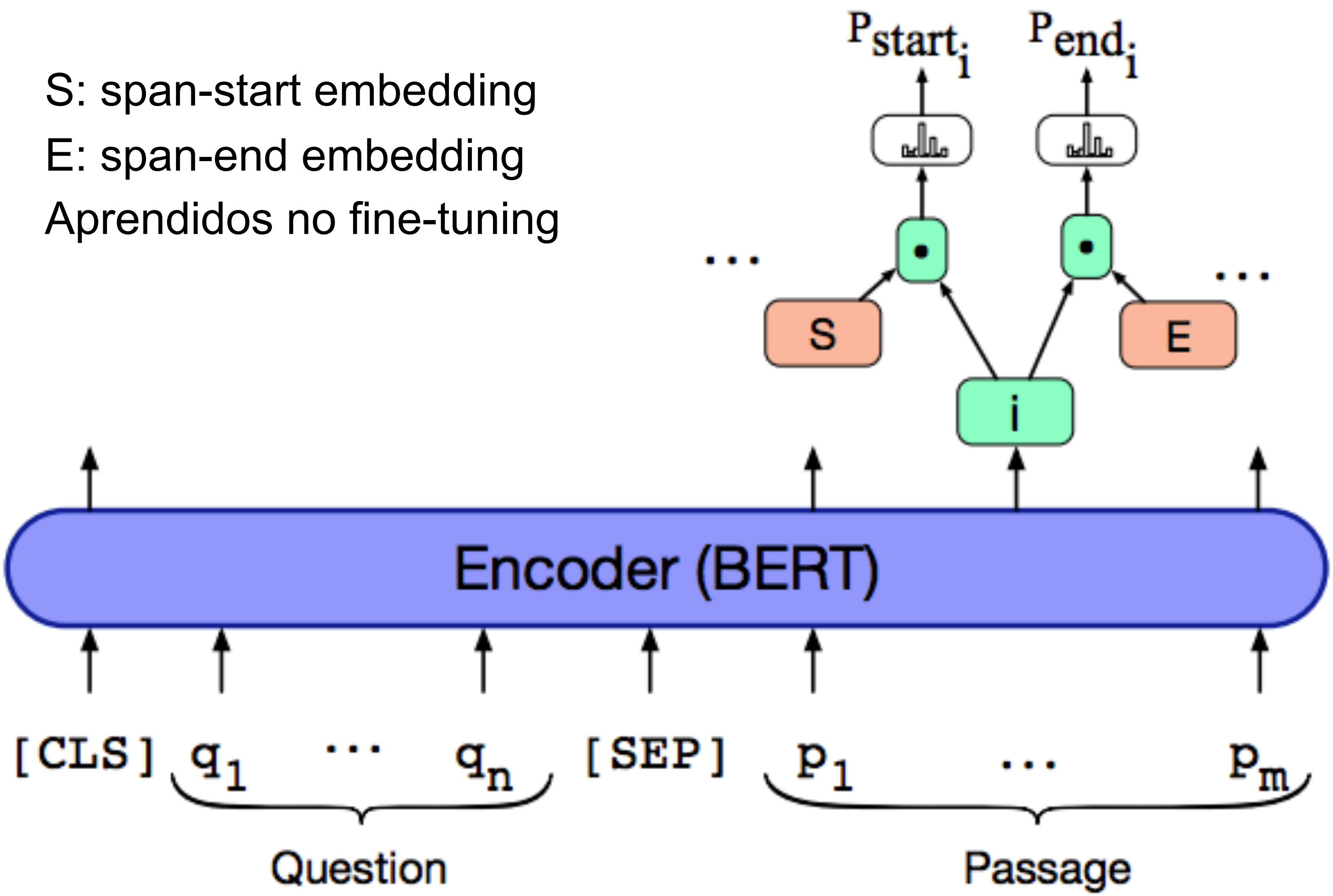


Image credit: https://mccormickml.com/

Q&A Extrativa: BERT

- S: span-start embedding
- E: span-end embedding
- Aprendidos no fine-tuning



Q&A Extrativa: BERT

$$P_{\text{start}_i} = \frac{\exp(S \cdot p_i')}{\sum_j \exp(S \cdot p_j')}$$

$$P_{\text{end}_i} = \frac{\exp(E \cdot p_i')}{\sum_j \exp(E \cdot p_j')}$$

$$L = -\log P_{\text{start}_i} - \log P_{\text{end}_i}$$