

CS6120: Lecture 7

Homework & Exams

Kenneth Church

<https://kwchurch.github.io/>

人工智能赋能产业发展
Artificial Intelligence Empowers Industrial Development



The Easy, the Hard and the Ugly

Kenneth Church

**Northeastern University
San Jose, CA, USA**

➤ Easy: Exciting Eco-system

➤ **Prompting**, Inference, Fine-Tuning

● Hard: Limited to large companies

- Pre-training

● Ugly: Responsible AI

- Bias, Toxicity
- Misinformation
- Hallucinations
- Plagiarism



- Prompt Engineering:
 - Super-Popular (100+ million users)
 - Most successful (rapid) adoption of any web app ever
 - Super-Easy
 - Easier than Fine-Tuning (and Inference)
 - Use Cases
 - “Helping” with homework:
 - Cheating (?)
 - Documentation:
 - Alternative to stack overflow

• “Helping” with homework: Cheating (?)

- **Collaborate** with students on essays
 - *You have no idea how much we're using ChatGPT*
 - Cheating?
- ChatGPT is better for some tasks
 - Good: thesis statements, outlines
 - Bad: capture student's voice
 - Worse: quotes
- Learning opportunity:
 - How to decompose writing to subtasks
 - Collaboration is great,
 - but student is responsible for end-product



Replacing Stack Overflow with ChatGPT



aiss ERROR: Could not find a version that satisfies the requirement faiss



The error message "ERROR: Could not find a version that satisfies the requirement faiss" indicates that the Python package manager (likely `pip`) cannot find a suitable version of the `faiss` library that matches the requirements specified by your project.

Here are some steps you can take to troubleshoot and resolve this issue:

1. **Check spelling and case:** Ensure that you have spelled "faiss" correctly in your requirements file or in the command you are using to install the package. Python package names are case-sensitive.

```
bash
```

Copy code

```
pip install faiss
```

2. **Update pip:** Make sure you are using an up-to-date version of `pip`. You can upgrade

R: Could not find a version that satisfies the requirement faiss

message "ERROR: Could not find a version that satisfies the requirement faiss" that the Python package manager (likely `pip`) cannot find a suitable version of the `faiss` library that matches the requirements specified by your project.

Some steps you can take to troubleshoot and resolve this issue:

Spelling and case: Ensure that you have spelled "faiss" correctly in your requirements file or in the command you are using to install the package. Python package names are case-sensitive.

```
pip install faiss
```

Update pip: Make sure you are using an up-to-date version of `pip`. You can upgrade by running the following command:

Copy code

Regenerate

➤ Easy: Exciting Eco-system

➤ Prompting, **Inference**, Fine-Tuning

● Hard: Limited to large companies

- Pre-training

● Ugly: Responsible AI

- Bias, Toxicity

- Misinformation

- Hallucinations

- Plagiarism



Resources



DataSets (Benchmarks)

- Examples
 - GLUE (NYU)
 - SQuAD (Stanford)
- Features
 - Splits:
 - Test, Validation (Dev), Train
 - Metrics
 - Leaderboard

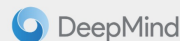
Hubs

- Examples
 - HuggingFace <https://huggingface.co/>
 - PaddleHub (Baidu)
- Features
 - Datasets (50k)
 - Models (300k)
 - Apps (100k)
 - Community Engagement
 - (Social Media)

Growing Quickly
($\approx 10x/year$)



ML²



SQuAD2.0
The Stanford Question Answering Dataset

Test of time:
Prompting & Inference are cool (for now)



Back Translation of Synonyms

```
for s in syns:  
    print(backtranslate(s, ["Helsinki-NLP/opus-mt-en-zh", "Helsinki-NLP/opus-mt-zh-en"]))
```

```
['rant and raving', '暴暴和暴暴,', 'Violence and violence, violence, violence, violence, violence,  
['regard and heed', '重视和关注', 'Attention and attention']  
['monster and abnormity', '薄命的、薄命的、薄命的、', 'the left hand, the left hand, the left hand,  
['relinquish and forsake', '放弃和放弃', 'Waiver and abandonment']  
['ebb and recede', 'ebb和reede( 缩缩)', 'ebb and reede (shrunk)']  
['austere and stern', '坚硬和坚硬', 'Hard and hard.']
```


✓ Easy: Exciting Eco-system

✓ Prompting, Inference, Fine-Tuning

✓ Hard: Limited to large companies

✓ Pre-training

● Ugly: Responsible AI

● Bias, Toxicity

● Misinformation

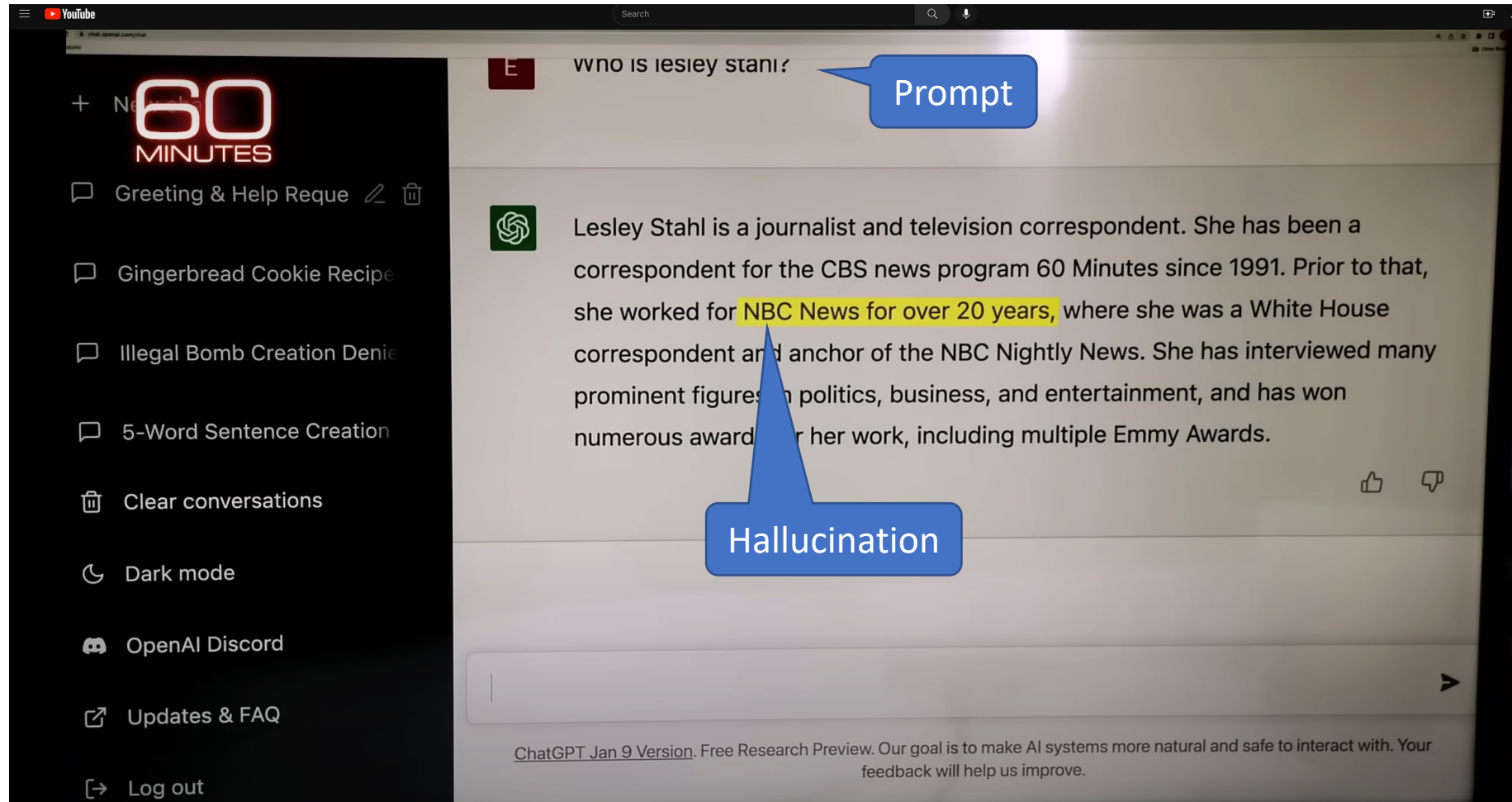
➤ **Hallucinations**

● Plagiarism



ChatGPT Hallucinates on CBS ``60 Minutes''

<https://www.youtube.com/watch?v=1wzPr4cUoMQ&t=463s>



Success/Catastrophe: Success → Responsibility

- What's new
 - The world is taking notice of AI
 - Fluency is much improved
- What's not new
 - Chatbots (and much of the tech)
 - SOTA-Chasing
 - Trustworthiness is still open

人工智能赋能产业发展

Artificial Intelligence Empowers Industrial Development

- View from America

- Asia: Optimism
- Europe: Caution
 - Americans move fast
 - and break things
 - Technology → Change
 - (for better and for worse)



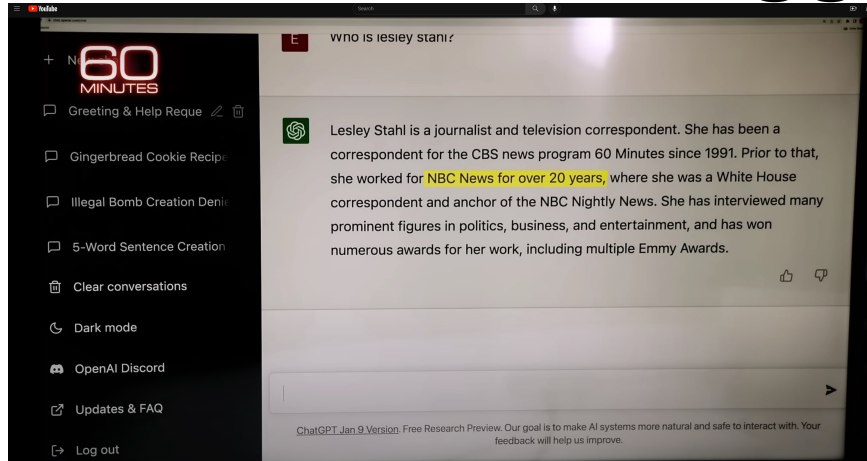
Personal History

- Strengths (fluency) and weaknesses (trustworthiness)
 - may be a consequence of choices we made in 1990s
- We started EMNLP in 1990s for pragmatic reasons
 - Field had been attempting to do too much
 - and was accomplishing too little
 - (during a funding winter)
- We chose to stop working on hard problems
 - (trustworthiness)
 - in order to make relatively quick progress on fluency
 - by reviving empirical methods from the 1950s
 - (Shannon, Skinner, Firth)
- Deep Nets are
 - more fluent
 - than trustworthy
- Pendulum Swung Too Far (2011)
 - Empiricism (1950s)
 - Rationalism (1970s)
 - Empiricism (1990s)
 - Deep Nets (2010s)

Fluency

Truth

Constructive Suggestions for Hallucinations



1. Low Road:

- Give up

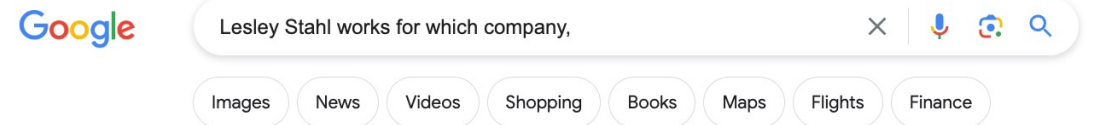
2. Middle Road

- Fact-checking with search

3. High Road

- Revive Rationalism

Query: *Lesley Stahl works for which company*



About 4,650,000 results (0.51 seconds)



CBS News

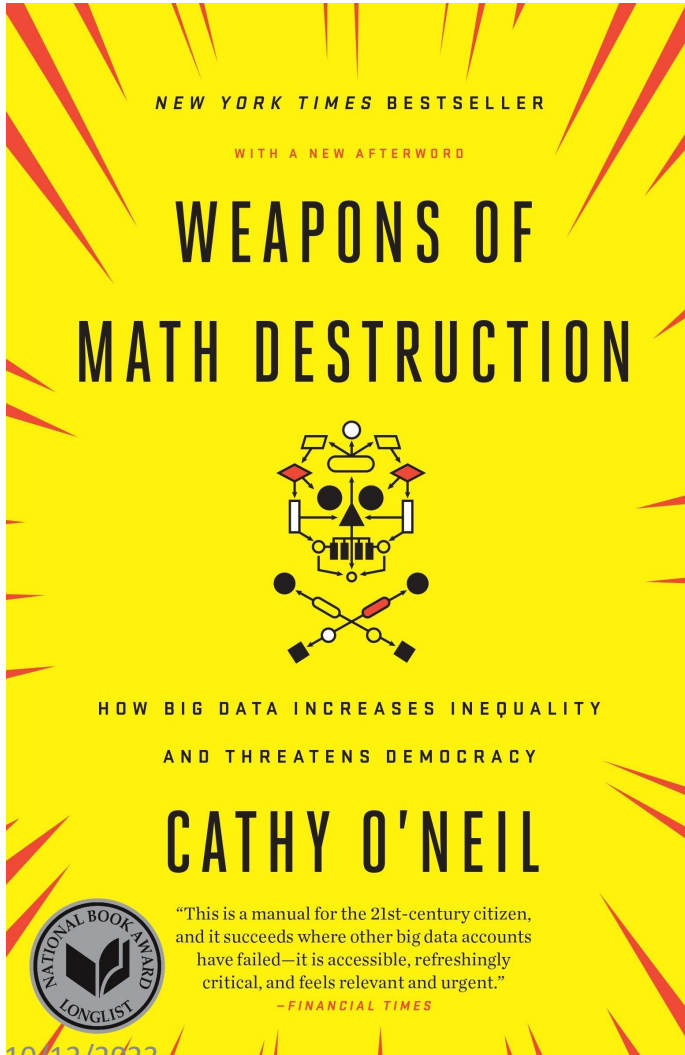
Lesley Rene Stahl (born December 16, 1941) is an American television journalist. She has spent most of her career with CBS News, where she began as a producer in 1971. Since 1991, she has reported for CBS's 60 Minutes. She is known for her news and television investigations and award-winning foreign reporting.

 Wikipedia
https://en.wikipedia.org/wiki/Lesley_Stahl

[Lesley Stahl - Wikipedia](#)

 About featured snippets •  Feedback

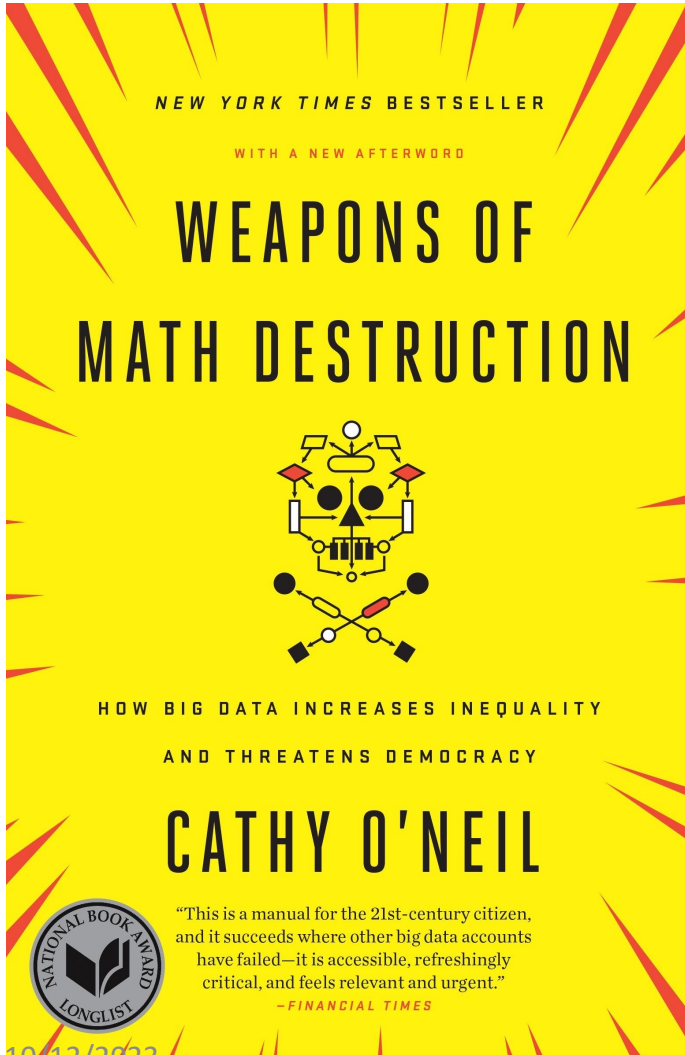
History of Irresponsible AI Risk (5 years ago) Product gets canceled



10/12/2023

Stupidity: To err is human...
Spelling Correction
Obama → *Osama*

History of Irresponsible AI Risk (5 years ago) Product gets canceled



10/12/2023

Stupidity: To err is human...

Spelling Correction

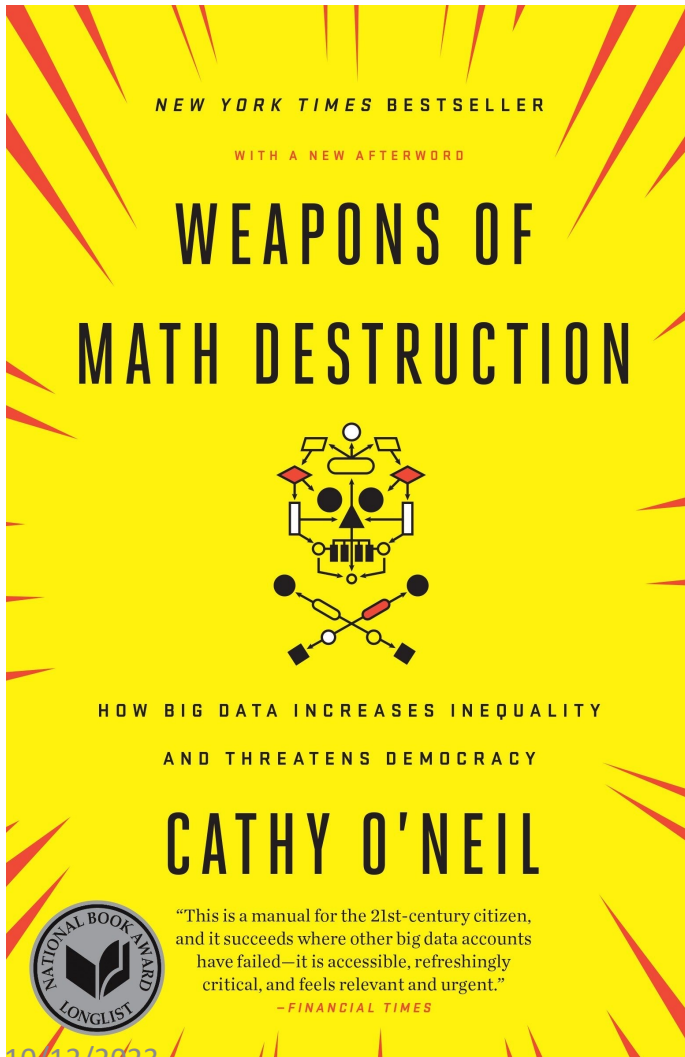
Obama → *Osama*

Bots:

Kids chat with Santa

Snow → *I like drugs too...*

History of Irresponsible AI Risk (5 years ago) Product gets canceled



10/12/2023

Stupidity: To err is human...

Spelling Correction

Obama → *Osama*

Bots:

Kids chat with Santa

Snow → *I like drugs too...*

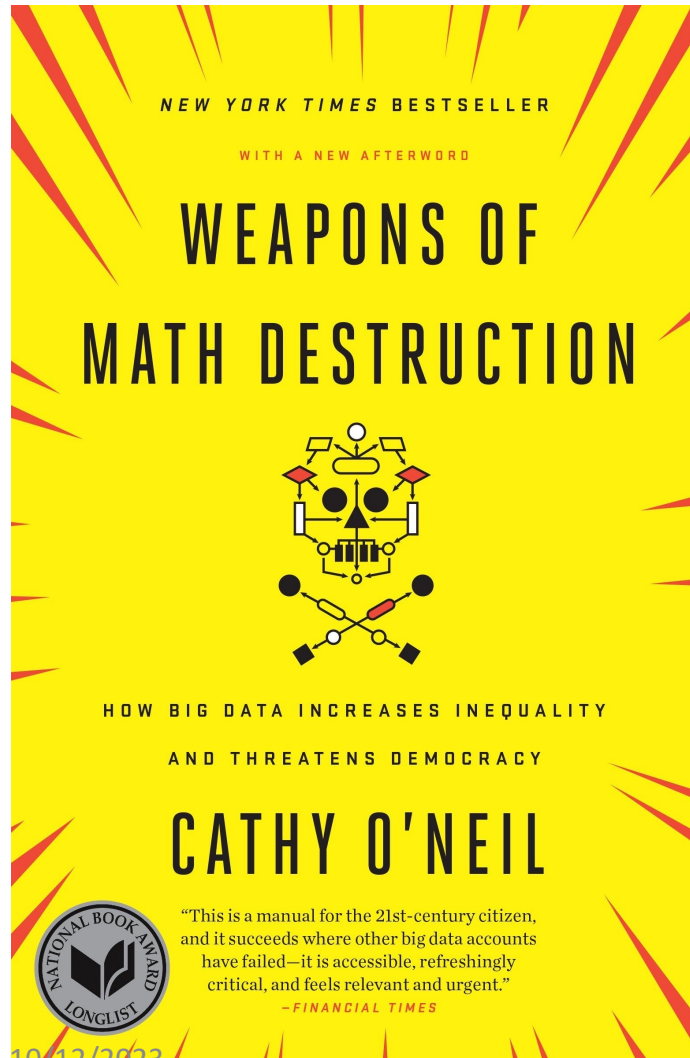
Microsoft sued for 'racist' application

Microsoft says it fixed the problem -- long before the litigation.



Written by **Matthew Broersma**, Contributor on June 29, 1999

History of Irresponsible AI Risk (5 years ago) Product gets canceled



Twitter taught Microsoft's AI chatbot to be a racist asshole in less than a day

By James Vincent | Mar 24, 2016, 6:43am EDT
Via [The Guardian](#) | Source [TayandYou \(Twitter\)](#) | 68 comments

f t SHARE



Microsoft sued for 'racist' application

Microsoft says it fixed the problem -- long before the litigation.



Written by **Matthew Broersma**, Contributor on June 29, 1999

- ✓ Easy: Exciting Eco-system
 - ✓ Prompting, Inference, Fine-Tuning
- ✓ Hard: Limited to large companies
 - ✓ Pre-training
- ✓ Ugly: Responsible AI
 - ✓ Bias, Toxicity
 - ✓ Misinformation
 - ✓ Hallucinations
 - ✓ Plagiarism



CS6120: Lecture 7

Homework & Exams

Kenneth Church

<https://kwchurch.github.io/>

ChatGPT

Strengths

- Fluency
 - Metaphor
 - Trivia

Weaknesses

- Truthfulness
 - Quotes
 - Facts and Figures
 - References
 - Perspectives
 - (other than American)
 - Nuance
 - Depth
 - Timeliness

Alternatives to ChatGPT

- Google
- Google Scholar
 - (for academic references)
- Wikipedia
- YouTube
- Stackoverflow
 - (for documentation)
- GitHub
- Quora
- Historians

Stephen Platt on Imperial Twilight: The Opium War and China's Last Golden Age

- Connection between history and future
 - <https://www.youtube.com/watch?v=17WF0v48vGw&t=230s>
 - Opium War was not that important in terms of battles (but had lots of long-term consequences)
 - Current leadership in China is using century of humiliation to motivate contemporary policies
 - West was ahead in technology (but China is catching up quickly)
 - Balance of Trade
 - Balance of trade favored China
 - Textile trade >> Opium
 - Face-Saving: Symbols >> Substance
 - China: China was unhappy about losing silver (hard currency) for Opium
 - Britain: England wanted jurisdiction over smuggling (and all other crimes)
 - Horrible misunderstanding: <https://youtu.be/17WF0v48vGw?t=3015>
 - China was cracking down on corruption
 - British bureaucrat panics, and buys opium with IOUs and gives the opium to China for them to destroy
 - British parliament didn't want to pay the IOUs, and hoped to force China to pay
- Review: https://youtu.be/dLun_eTtFkc

Fact Checking

During the First Opium War (1839-1842), the British government was led by the Conservative Party under Prime Minister Sir Robert Peel. The opposition, primarily the Whigs, had varying views on the war. Some opposed it on moral grounds, criticizing the ethics of trading in opium, while others were concerned about the potential impact on international relations and trade.

- https://en.wikipedia.org/wiki/1839_in_the_United_Kingdom
- <https://youtu.be/17WF0v48vGw?t=1653>

Perspectives

- From America
 - <https://youtu.be/17WF0v48vGw?t=1187>
- From France
 - <https://youtu.be/17WF0v48vGw?t=353>

What is a good survey of approximate nearest neighbors?

- A good survey of approximate nearest neighbors methods can provide insights into various techniques and algorithms used to efficiently find approximate solutions to nearest neighbor search problems. Here are some key methods and resources you might find helpful:

1. Locality-Sensitive Hashing (LSH):

Title: "Locality-Sensitive Hashing Scheme Based on p-Stable Distributions"

Authors: Piotr Indyk, Rajeev Motwani

Link: [LSH Paper](#)

2. Randomized Techniques:

Title: "Randomized Algorithms in Linear Algebra"

Authors: Alan Frieze, Ravi Kannan, Santosh Vempala

Link: [Randomized Techniques Paper](#)

3. Tree-based Methods:

Title: "The Annoy Library: Approximate Nearest Neighbors in C++/Python"

Authors: Erik Bernhardsson

Link: [Annoy Library](#)

1. Graph-Based Methods:

Title: "Efficient K-Nearest Neighbor Graph Construction for Generic Similarity Measures"

Authors: Wei Dong, Charikar Moses, Kai Li

Link: [Graph-Based Methods Paper](#)

2. Product Quantization:

Title: "Product Quantization for Nearest Neighbor Search"

Authors: Hervé Jégou, Matthijs Douze, Cordelia Schmid

Link: [Product Quantization Paper](#)

3. Survey Articles:

Title: "A Survey of Nearest Neighbor Search Algorithms"

Authors: Yufei Tao, Dongxiang Zhang

Link: [Survey Paper](#)

Title: "Nearest Neighbor Search: A Survey"

Authors: Xiaojie Guo, Yimin Zhang, Hao Zhu, Chi Zhang, Zhenhua Li

Link: [Survey Paper](#)

- These papers cover a range of methods, from traditional data structures to more recent machine learning-based approaches. Depending on your specific needs and the characteristics of your data, different methods may be more suitable.

Fact-Checking

Title: "A Survey of Nearest Neighbor Search Algorithms"

Authors: Yufei Tao, Dongxiang Zhang

Link: Survey Paper

- <https://scholar.google.com/citations?hl=en&user=nYN9A3IAAAAJ>
- <https://scholar.google.com/citations?user=3FmJBHkAAAAJ&hl=en>
- <https://scholar.google.com/scholar?oi=gsb95&q=A%20Survey%20of%20Nearest%20Neighbor%20Search%20Algorithms&lookup=0&hl=en>

References

<https://scholar.google.com/citations?hl=en&user=oOwNKsAAAAAJ>



Piotr Indyk

Professor of Electrical Engineering and Computer Science, [MIT](#)
Verified email at mit.edu - [Homepage](#)

[Algorithms](#) [Computational Geometry](#) [Theoretical Computer Science](#)

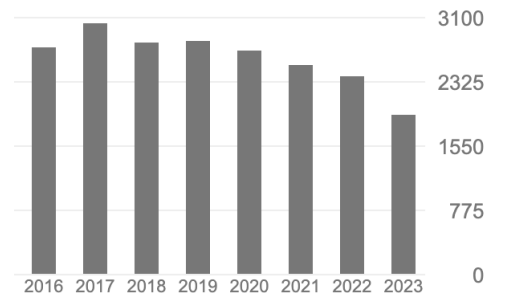
FOLLOW

TITLE	CITED BY	YEAR
Approximate nearest neighbors: towards removing the curse of dimensionality P Indyk, R Motwani Proceedings of the thirtieth annual ACM symposium on Theory of computing ...	5831	1998
Similarity search in high dimensions via hashing A Gionis, P Indyk, R Motwani Vldb 99 (6), 518-529	4685	1999
Locality-sensitive hashing scheme based on p-stable distributions M Datar, N Immorlica, P Indyk, VS Mirrokni Proceedings of the twentieth annual symposium on Computational geometry, 253-262	3793	2004
Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions A Andoni, P Indyk Communications of the ACM 51 (1), 117-122	3108	2008
Maintaining stream statistics over sliding windows M Datar, A Gionis, P Indyk, R Motwani SIAM journal on computing 31 (6), 1794-1813	1269	2002
Enhanced hypertext categorization using hyperlinks S Chakrabarti, B Dom, P Indyk	1218	1998

Cited by

[VIEW ALL](#)

	All	Since 2018
Citations	42356	15176
h-index	85	51
i10-index	193	153



Public access

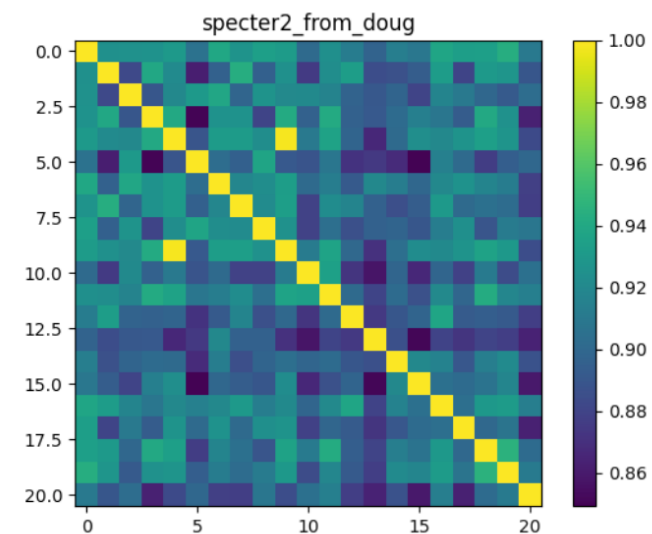
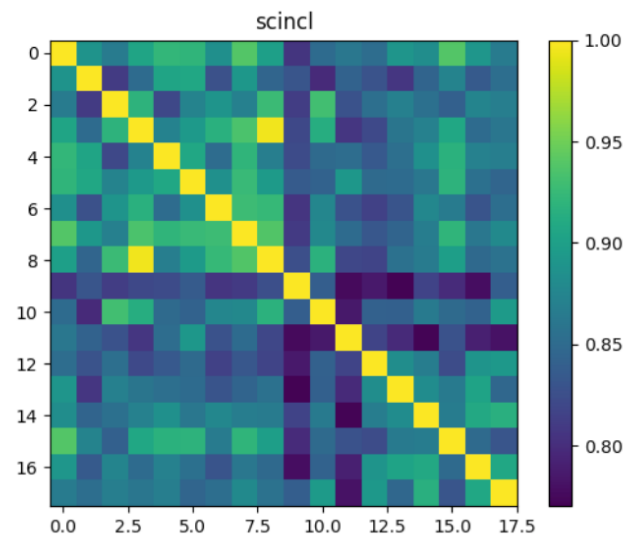
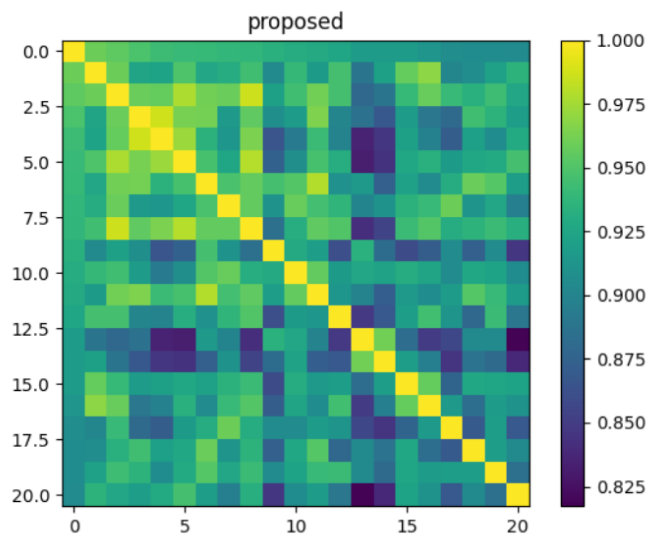
[VIEW ALL](#)

0 articles	35 articles
not available	available

Based on funding mandates

Paper: Efficient and Robust Approximate Nearest Neighbor Search Using Hierarchical Navigable Small World Graphs

<http://34.204.188.58/cgi-bin/similar?CorpusId=CorpusId:8915893&embedding=proposed&limit=20>



[Top](#)

[score](#) [citationCount](#)

	Paper	Authors	year	More like this	proposed	scincl	specter2	from_doug
849	Efficient and Robust Approximate Nearest Neighbor Search Using Hierarchical Navigable Small World Graphs	Yury Malkov , Dmitry A. Yashunin	2016	similar to this	1.0	1.0	1.0	
0.959 277	ANN-Benchmarks: A Benchmarking Tool for Approximate Nearest Neighbor Algorithms	Martin Aumüller , Erik Bernhardsson , Alexander Faithfull	2018	similar to this	0.959		0.925	
0.954 4	Navigable Proximity Graph-Driven Native Hybrid Queries with Structured and Unstructured Constraints	Mengzhao Wang , Li-ang Lv , ..., Jiongkang Ni	2022	similar to this	0.954	0.888	0.925	
0.948 19	Results of the NeurIPS'21 Challenge on Billion-Scale Approximate Nearest Neighbor Search	H. Simhadri , G. Williams , ..., Jingdong Wang	2022	similar to this	0.948	0.866	0.926	
0.942 69	DiskANN: Fast Accurate Billion-point Nearest Neighbor Search on a Single Node	Suhas Jayaram Subramanya	2019	similar to this	0.942	0.904	0.93	
0.941 57	Optimization of Indexing Based on k-Nearest Neighbor Graph for Proximity Search in High-dimensional Data	M. Iwasaki , D. Miyazaki	2018	similar to this	0.941	0.923	0.906	
0.94 2	GraSP: Optimizing Graph-based Nearest Neighbor Search with Subgraph Sampling and Pruning	Minjia Zhang , Wenhan Wang , Yuxiong He	2022	similar to this	0.94	0.921	0.939	
0.938 57	SONG: Approximate Nearest Neighbor Search on GPU	Weijie Zhao , Shulong Tan , Ping Li	2020	similar to this	0.938	0.884	0.927	
0.937 179	Fast Approximate Nearest Neighbor Search With The Navigating Spreading-out Graph	Cong Fu , Chao Xiang , ..., Deng Cai	2017	similar to this	0.937	0.938	0.934	



Women, Fire, and Dangerous Things

5 languages ▾

Article Talk

Read Edit View history Tools ▾

From Wikipedia, the free encyclopedia

Women, Fire, and Dangerous Things: What Categories Reveal about the Mind is a non-fiction book by the cognitive linguist George Lakoff. The book, first published by the University of Chicago Press in 1987, puts forward a model of cognition argued on the basis of semantics. The book emphasizes the centrality of metaphor, defined as the mapping of cognitive structures from one domain onto another, in the cognitive process.^[1] *Women, Fire, and Dangerous Things* explores the effects of cognitive metaphors, both culturally specific and human-universal, on the grammar per se of several languages, and the evidence of the limitations of the classical logical-positivist or Anglo-American School philosophical concept of the category usually used to explain or describe the scientific method.

The book's title was inspired by the noun class system of the Dyirbal language, in which the "feminine" category includes nouns for women, water, fire, violence, and certain animals.

See also [edit]

- Image schema
- Structuralism
- Taxonomy (general)

References [edit]

- ↑ "Defending Freedom" . *New Republic*. October 16, 2006.

Women, Fire, and Dangerous Things

George Lakoff

Women,
Fire, and
Dangerous
Things

*What Categories Reveal
about the Mind*

Author	George Lakoff
Subject	Cognitive linguistics
Publisher	University of Chicago Press
Publication date	1987
Pages	632
ISBN	0-226-46803-8

This article about a book on language, linguistics or translation is a *stub*. You can help Wikipedia by

Sports Metaphors (unusual)

- **To punt** is, first and foremost, to stand on a **flat-bottomed** boat on a river in **Cambridge or Oxford** and to propel the boat along with a long pole. There is an art to this (slightly different between the 2 places). The boat is a punt and is available on hire. The person who propels is the punter.
- The key to the second meaning is the concept of **hire** (as in hiring a boat). A punter is someone with money to spend, a customer, perhaps someone willing to **lay a bet**. Betting shops regularly refer to their “punters”.
- By extension, a punter is a **client** of any kind. Lili is quite right that among these are customers of a sex worker, but actually it could be anyone waiting to be served at a counter, with a cup of coffee, asking for a can of peaches or needing **help** to try on a possible clothing purchase.
- Finally “to **take a punt**” simply means “to have a go” or “give it a try”, as in “I’ll take a punt at anything if it’s not dangerous” and similar.
- NB in **Cockney Rhyming Slang**, a punter is often a “Billy Bunter” and the phone often the “dog and bone”. So now you have an idea what someone in the betting shop office means when they shout, “I’ve got a **Billy on the dog**, what do I tell him ?”

Question for ChatGPT: what does punt mean in England

- In England, the term "punt" can have different meanings depending on the context. Here are a few possible interpretations:

1. Boat: A punt can refer to a flat-bottomed boat, typically with a square-cut bow, that is commonly used for boating on rivers and shallow waters. Punting involves propelling the boat by pushing against the riverbed with a long pole.

1. Football/Soccer: In the context of sports, particularly football (soccer), "punt" is not a common term. In American and Canadian football, a "punt" refers to a method of kicking the ball, usually on fourth down, to the opposing team to exchange possession of the ball.

- If you have a specific context in mind, providing more details could help me give a more accurate explanation.

Ragging the Puck:

Q for ChatGPT: what does ragging the puck mean

- "Ragging the puck" is a term primarily used in ice hockey. It refers to a strategy employed by a team to control the puck and maintain possession, especially when they are trying to protect a lead or kill off time in a game. Instead of aggressively attacking or attempting to score, players may use skilled puck-handling and skating techniques to keep the puck away from their opponents.
- Ragging the puck involves maintaining possession through controlled and deliberate movements, often in the offensive zone, without taking unnecessary risks that could lead to a turnover. Players might skate in circles, pass the puck among themselves, or use other maneuvers to waste time and frustrate the opposing team.
- This strategy is often employed when a team is ahead on the scoreboard and wants to run down the clock. It requires skillful puck control and good teamwork to execute effectively. While it can be an effective tactic, it also requires careful judgment to avoid penalties and turnovers that could give the opposing team an advantage.

Q for ChatGPT: what does ragging the puck mean in Canadian Hansards

- In the context of Canadian Hansards, "ragging the puck" is not a commonly used term. The term "Hansard" generally refers to the official transcripts of debates and proceedings in the Canadian Parliament. The language used in Hansards is formal and specific to parliamentary discussions.
- If "ragging the puck" were to appear in Canadian Hansards, it might be used metaphorically or colloquially to describe a strategy or tactic employed during parliamentary debates. However, without a specific instance or context, it's challenging to provide a precise interpretation.
- If you have a specific passage or context in mind, please provide more details, and I'll do my best to help.

https://en.wikipedia.org/wiki/Sticky_wicket

- A **sticky wicket** (or **sticky dog**, or **glue pot**)^[1] is a [metaphor](#)^[2] used to describe a difficult circumstance. It originated as a term for difficult circumstances in the sport of [cricket](#), caused by a damp and soft [wicket](#).^[3]
- In cricket^[edit]
- The phrase comes from the game of cricket. "[Wicket](#)" has several meanings in cricket: in this case it refers to the rectangular area, also known as the pitch, in the centre of the [cricket field](#) between the [stumps](#). The wicket is usually covered in a much shorter grass than the rest of the field or entirely bare, making it susceptible to variations in weather, which in turn cause the ball to bounce differently.^[4]
- If rain falls and the wicket becomes wet, the ball may not bounce predictably, making it very difficult for the [batter](#).^[5] Furthermore, as the pitch dries, conditions can change swiftly, with [spin bowling](#) being especially devastating, as the ball can deviate laterally from straight by several feet. Once the wet surface begins to dry in a hot sun "the ball will rise sharply, steeply and erratically. A [good length](#) ball ... becomes a potential lethal delivery. Most batters on such wickets found it virtually impossible to survive let alone score."^[6] Certain cricketers developed reputations for their outstanding abilities to perform on sticky wickets. Australian [Victor Trumper](#) was one.^[6]

Exam: Question 1 (MUSE)

get_nn is reasonably fast

Get nearest neighbors

```
In [ ]: def get_nn(word, src_emb, src_id2word, tgt_emb, tgt_id2word, K=5):  
        # print("Nearest neighbors of \"%s\":" % word)  
        word2id = {v: k for k, v in src_id2word.items()}  
        word_emb = src_emb[word2id[word]]  
        scores = (tgt_emb / np.linalg.norm(tgt_emb, 2, 1)[:, None]).dot(word_emb / np.linalg.norm(word_emb))  
        k_best = scores.argsort()[-K:][::-1]  
        return [tgt_id2word[idx] for idx in k_best]  
        # for i, idx in enumerate(k_best):  
        #     print('%.4f - %s' % (scores[idx], tgt_id2word[idx]))
```

```
In [ ]: # printing nearest neighbors in the source space  
src_word = 'cat'  
get_nn(src_word, src_embeddings, src_id2word, src_embeddings, src_id2word, K=5)
```

```
Out[30]: ['cat', 'cats', 'kitten', 'dog', 'kittens']
```

```
In [ ]: # printing nearest neighbors in the target space  
src_word = 'cat'  
get_nn(src_word, src_embeddings, src_id2word, tgt_embeddings, tgt_id2word, K=5)
```

```
Out[31]: ['chat', 'cat', 'chien', 'chats', 'lapin']
```

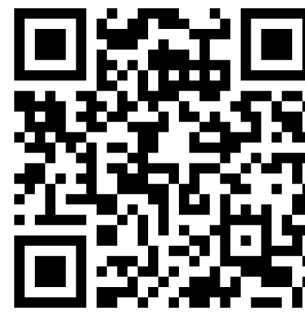

Open-ended (optional) observations

- Spelling (orthography) is taken more seriously in engineering than linguistics: *unit* \neq *un* + *it*
 - Linguistics (following Chomsky)
 - More legit:
 - sound (phonology & phonetics) and
 - meaning (syntax & semantics)
 - Less legit:
 - spelling,
 - distributional statistics (word2vec, BERT, deep nets, cosines),
 - psycholinguistics (reaction times, memory limitations),
 - sociolinguistics, etymology, historical linguistics, etc.
 - <https://www.britannica.com/video/186425/look-words-some-language-English-Norman-Conquest>

Compositionality

- *divineness* = *divine* + *ness*
 - Sound
 - Meaning
- ChatGPT weakness: deposing larger problems into smaller problems
 - ChatGPT can add two small numbers,
 - but not two large numbers
 - (cannot decompose large problems into more manageable problems)
 - Chain of Thought Reasoning → Opportunity for Improvement
- *-ness* is more compositional than *-ity*
 - minimal pair: *divineness* vs. *divinity*

https://en.wikipedia.org/wiki/Trisyllabic_laxing



Tense vowel	→	Lax vowel	Change in Middle English	Example	IPA
i:	→	ɛ	eɪ → e ɛɪ → e	serene, serenity; impede, impediment	/sɛˈriːn, sɛˈrɛnɛti/; /ɪmˈpiːd, ɪmˈpɛdɛmənt/
eɪ	→	æ	aɪ → a	profane, profanity; grateful, gratitude	/prəˈfeɪn, prəˈfænɛti/ /ˈɡreɪtful, ˈɡrætɛtjʊd/
aɪ	→	ɪ	iː → i	divine, divinity; derive, derivative	/dɪˈvaɪn, dɪˈvɪnɛti/; /dɪˈraɪv, dɪˈrɪvətɪv/
aʊ	→	ʌ	uː → u	profound, profundity; pronounce, pronunciation;	/prəˈfaʊnd, prəˈfʌndɛti/; /prəˈnaʊns, prəˈnʌnsiˈeɪʃən/
uː	→	ʊ	oː → o	(No longer part of the active vowel system of English) ^[1]	
oʊ	→	ʊ	ɔː → o	provoke, provocative; sole, solitude	/prəˈvoʊk, prəˈvʊk.ə.tɪv/; /ˈsoʊl, ˈsɒlɛtjʊd/



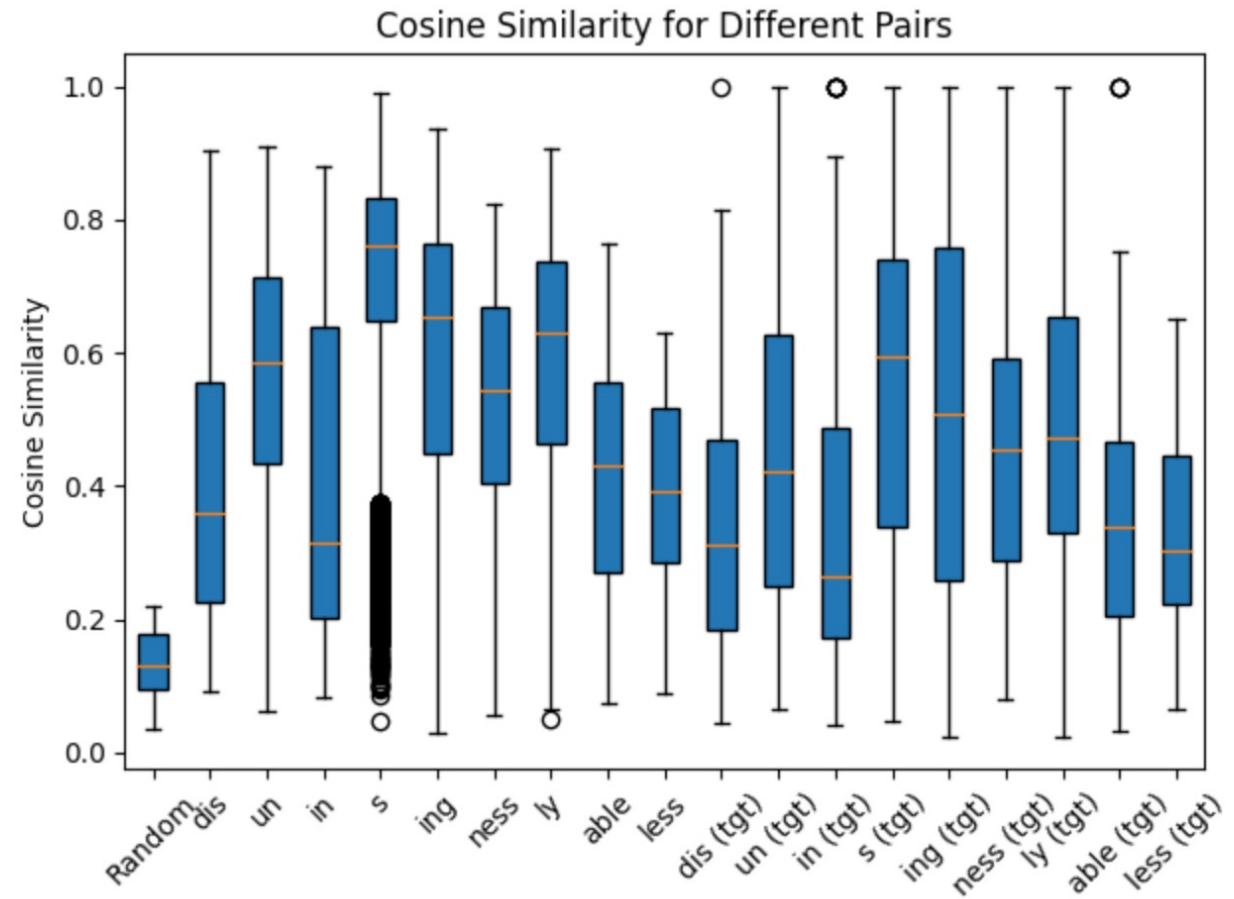
https://en.wikipedia.org/wiki/Great_Vowel_Shift

First phase of the Great Vowel Shift

Word	Vowel pronunciation	
	1400	1550
bite	/i:/	/ɛi/
meet	/e:/	/i:/
out	/u:/	/ɔu/
boot	/o:/	/u:/

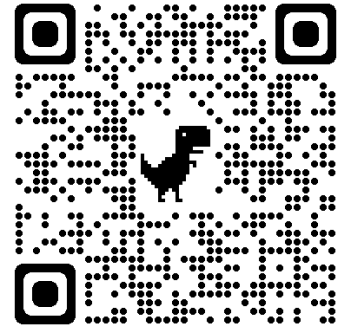
Synonyms, antonyms and cosines

- Cosine similarity is based on distribution
 - Both synonyms and antonyms have large cosines
 - because comparisons and contrasts are common in corpora
 - Antonyms → large cosines
 - Synonyms → large cosines
 - Random controls → small cosines (though above 0)
 - Translations → Smaller cosines than otherwise
 - (Meaning tends to get lost in translation)



Perplexity (and Entropy)

- <https://en.wikipedia.org/wiki/Perplexity>
- <https://web.stanford.edu/~jurafsky/slp3/B.pdf>
- https://web.stanford.edu/~jurafsky/slp3/slides/3_LM_Jan_08_2021.pdf
- Interpretations
 - Evaluation of language model
 - Difficulty of task; average branching factor
 - Space required to store a compressed file (or transmit it across a noisy channel)
 - https://www.youtube.com/watch?v=zjWXLD_ihOc
 - <https://www.youtube.com/watch?app=desktop&v=0mFSybsoPH0>
 - https://www.princeton.edu/~wbialek/rome/refs/shannon_51.pdf



B.1 The Noisy Channel Model

In this section we introduce the noisy channel model and show how to apply it to the task of detecting and correcting spelling errors. The noisy channel model was applied to the spelling correction task at about the same time by researchers at AT&T Bell Laboratories (Kernighan et al. 1990, Church and Gale 1991) and IBM Watson Research (Mays et al., 1991).

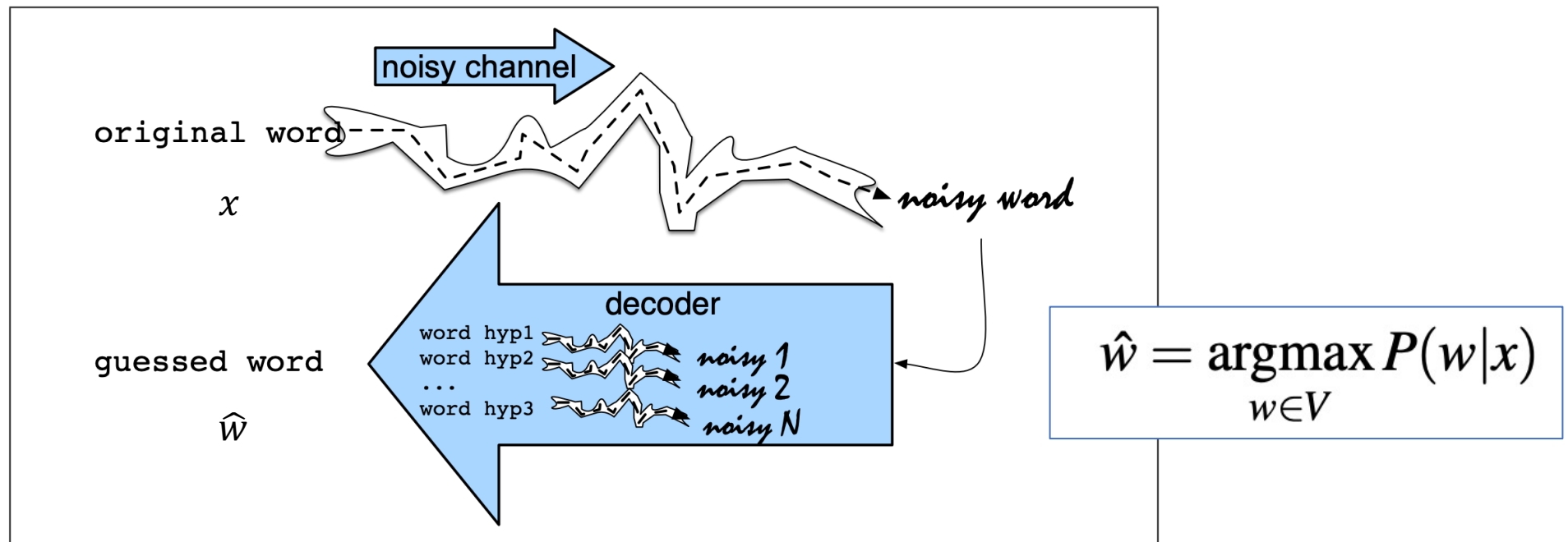


Figure B.1 In the noisy channel model, we imagine that the surface form we see is actually a “distorted” form of an original word passed through a noisy channel. The decoder passes each hypothesis through a model of this channel and picks the word that best matches the surface noisy word.

Noisy Channel Applications

- Spelling correction:
 - correct words → typos
 - <https://www.norvig.com/spell-correct.html>
 - <https://aclanthology.org/C90-2036.pdf>
- Speech recognition:
 - text → audio
- Machine translation:
 - English → French
- OCR (Optical Character Recognition):
 - words → pixels
- Part of speech tagging:
 - parts of speech → words
 - https://web.stanford.edu/~jurafsky/slp3/slides/8_POSNER_intro_May_6_2021.pdf
- NER (Named entity recognition):
 - BIO Labels → words

$$\hat{w} = \operatorname{argmax}_{w \in C} \overbrace{P(x|w)}^{\text{channel model}} \overbrace{P(w)}^{\text{prior}}$$