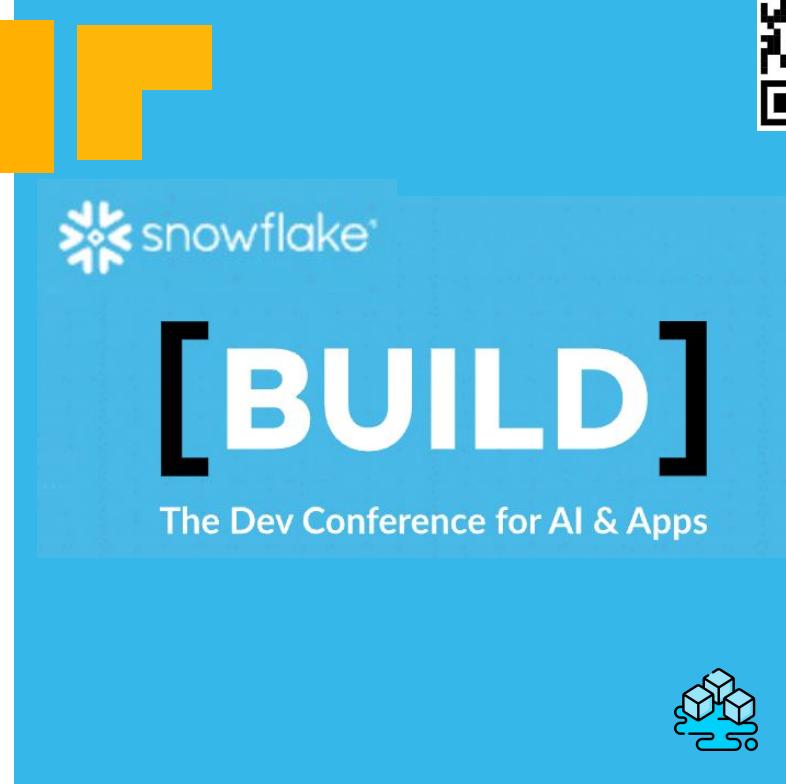




Melting the LLM

The inner workings of large language models

@SimonAubury



A slide from the Snowflake [BUILD] conference. The slide has a blue background. At the top left is the Snowflake logo (a stylized snowflake icon) followed by the word "snowflake". In the center is a large, bold "[BUILD]" title. Below the title is the subtitle "The Dev Conference for AI & Apps". At the bottom right of the slide is a small icon depicting three 3D cubes floating in a cloud.



STUG Sydney



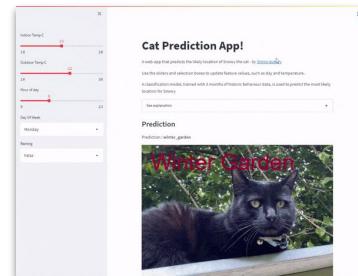
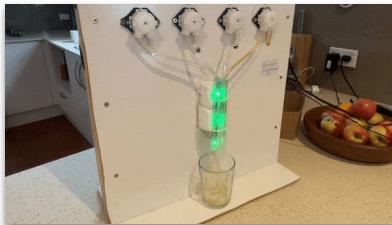
Hello!

I am Simon Aubury

Data stuff @ Simple Machines

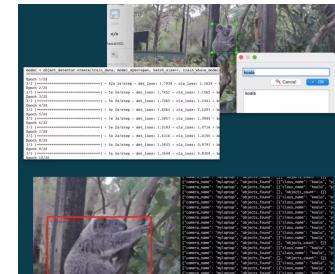


Simple Machines



🍸 [GinAI - Cocktails mixed with generative AI](#)

😺 [Can ML predict where my cat is now?](#)



🐨 [Koala Counting With Kafka](#)

LLM's

You may have heard of them?



Large language models



Data



A bunch of text

Architecture



The transformer

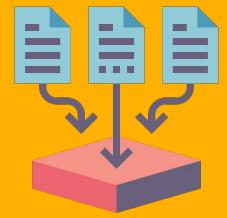
Training



Pre training & fine tuning

1. Data

Let's compress the internet





“Snow”



```
[0.024637436494231224, 0.04343702644109726,  
0.03674380108714104, 0.037638965994119644,  
0.020605742931365967, -0.07100269943475723  
0.011195449158549309, 0.06499766558408737,  
0.017523551359772682, 0.07894649356603622,  
-0.0295423474162817, -0.08110342174768448,  
-0.013438096269965172, 0.07227257639169693  
0.003804766107350588, 0.07242589443922043,
```

[WebVectors Online for "snow"](#)

Word vectors

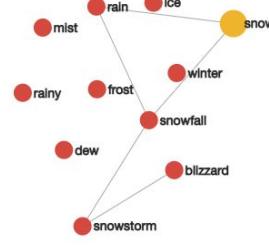


Semantic associates for **snow** (computed on English Wikipedia)

Word frequency

High Medium Low

1. snowfall 0.627
2. rain 0.611
3. frost 0.579
4. snowstorm 0.529
5. ice 0.525
6. winter 0.503
7. dew 0.482
8. blizzard 0.479
9. mist 0.473
10. rainy 0.466

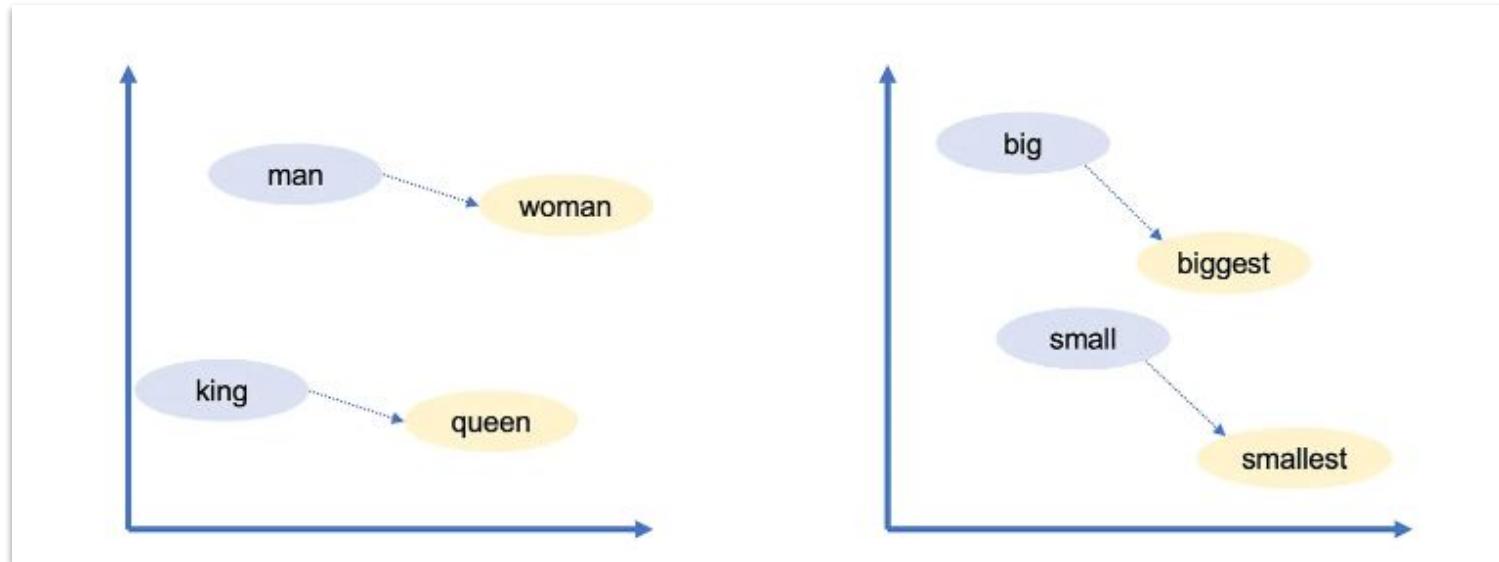


• We show only the associates of the same part of speech as your query. All associates can be found at the [Similar Words](#) tab.

Source: [WebVectors Online for "snow"](#)

Language models represent words* as a long list of numbers called a word vector

* words or tokens



Source : [LLM with a minimum of math and jargon](#)
Timothy B Lee & Sean Trott



Company

Snowflake Inc.

Article [Talk](#)

Place

From Wikipedia, the free encyclopedia

Snowflake Inc. is a [cloud computing](#)–based data [cloud](#) company based in [Bozeman, Montana](#). It was founded in July 2012 and was publicly launched in October 2014 after two years in [stealth mode](#).^{[2][3]}

The firm offers a cloud-based service, generally termed "data-as-a-service".^{[4][5]} It allows corporations to store and process data using cloud-based hardware and software. Snowflake services main features are separation of storage and compute, on-the-fly scalable compute, data sharing, data cloning, and third-party tools support in order to scale with its enterprise customers.^[6] It has run on [Amazon S3](#) since 2014,^[2] on [Microsoft Azure](#) since 2018^[7] and on the [Google Cloud Platform](#) since 2019.^{[8][9]} The company was ranked first on the [Forbes](#) Cloud 100 in 2019.^[10] The company's [initial public offering](#) raised \$3.4 billion in September 2020, one of the largest software IPOs in history.^[11]

Date

Expression

History [edit]

Snowflake Inc. was founded in July 2012 in [San Mateo, California](#) by three data warehousing experts: Benoît Dageville, Thierry Cruanes and Marcin Żukowski. Dageville and Cruanes

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



2. Architecture

The transformer



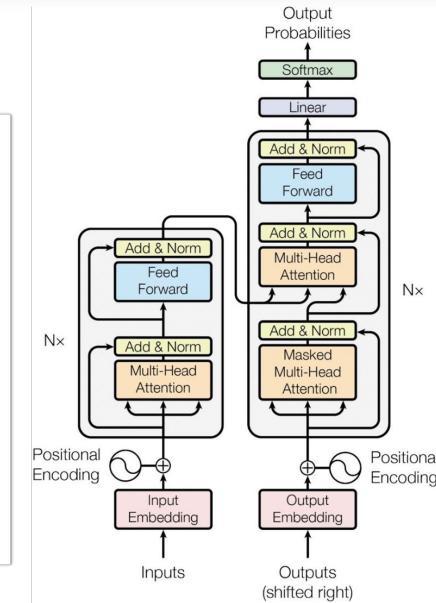
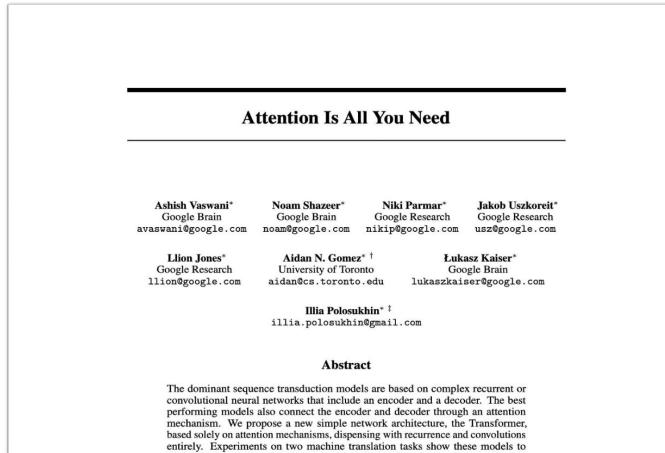
Understanding language can be challenging

context

The **mother penguin** comes back from hunting with **freshly** caught **fish**.

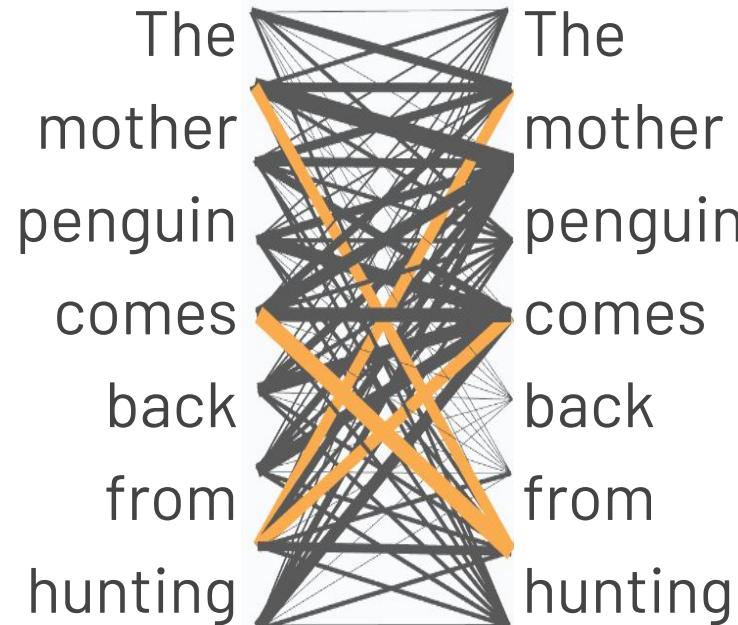
context

Transformers

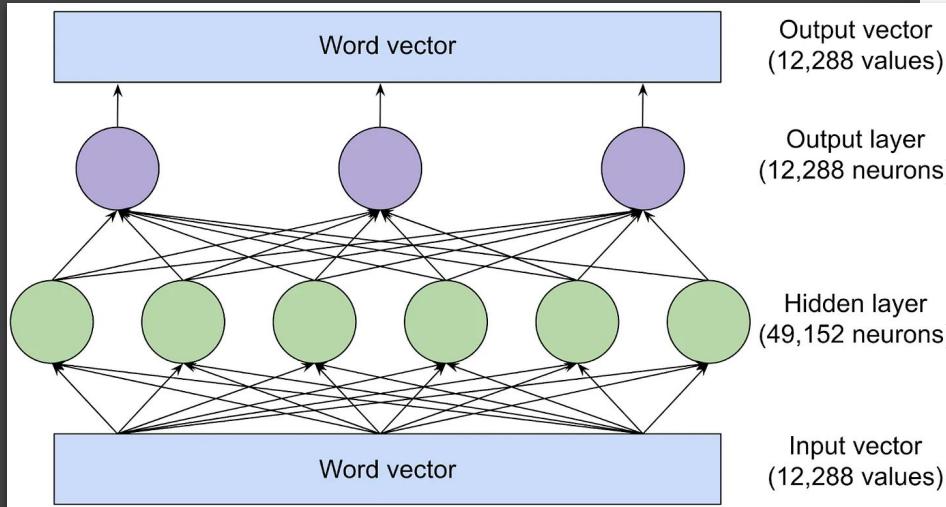


Source: Attention Is All You Need <https://arxiv.org/abs/1706.03762>

Self-attention



Transformers & attention



Each layer of an LLM is a transformer ... building a network with ***billions*** of parameters.

Source [LLM with a minimum of math and jargon](#)
Timothy B Lee & Sean Trott

The base model



Context

Base model

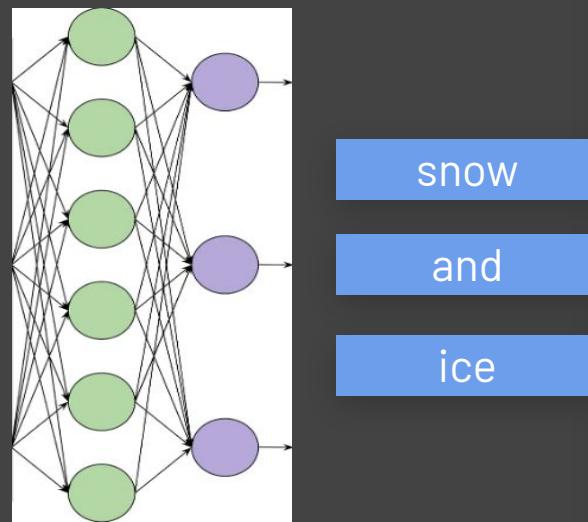
Completion

Penguins

live

amongst

the



Billions of parameters are dispersed through the network

- We **can** measure & adjust to improve predictions
- We **don't know** how the billions of parameters collaborate to do it!



3. Training

Pre training & fine-tuning



How to train your LLM



Pretraining (once)

1. PB's of text.
2. 1000's of GPUs.
3. Compress the text into a neural network,
4. Pay 💰 wait 



Obtain base model.

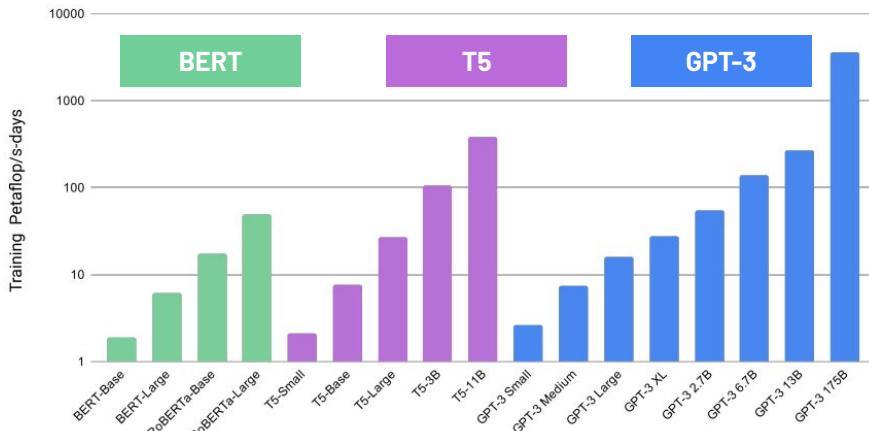
Fine-tuning (recurring)

1. 1000's ideal Q&A responses (human)
2. Finetune base model on this data wait 
3. Obtain assistant model
4. Evaluate, deploy & monitor

Pre-training



Total Compute Used During Training



Pre-training is the most expensive and time consuming stage of building an LLM.

~ \$100 million for GPT-4 !

Source: Brown et al. 2020, "Language Models are Few-Shot Learners"

<https://arxiv.org/pdf/2005.14165.pdf>

Fine-tuning (recurring)



Prompt:

"Can you write a short introduction about the relevance of the term "monopsony" in economics? Please use examples related to potential monopsonies in the labour market and cite relevant research."

Response:

"Monopsony" refers to a market structure where there is only one buyer for a particular good or service. In economics, this term is particularly relevant in the labor market, where a monopsony employer has significant power over the wages and working conditions of their employees. The presence of a monopsony can result in lower wages and reduced employment opportunities for workers, as the employer has little incentive to increase wages or provide better working conditions. Recent research has identified potential monopsonies in industries such as retail and fast food, where a few large companies control a significant portion of the market (Bivens & Mishel, 2013). In these industries, workers often face low wages, limited benefits, and reduced bargaining power, leading to a situation where they are dependent on the employer for their livelihood. This dependence can result in further suppression of wages and a decline in working conditions. Overall, the concept of monopsony is essential to understanding the dynamics of labor markets and the impact of market power on workers. Further research is needed to understand the extent and impact of monopsonies on the economy and to develop policies to address this issue. References: Bivens, J., & Mishel, L. (2013). The Pay of Corporate Executives and Financial Professionals as Evidence of Rents in Top 1 Percent Incomes. *Journal of Economic Perspectives*, 27(3), 57-78."

[example from OpenAssistant Conversations Dataset (OASST1)]

Labeling instructions

Excerpt of labeling instructions on the API prompt distribution

You are given a text-based description of a task, selected at a time. This task description may be in the form of an explicit instruction (e.g. "Write a story about a cat that flies"). The task may also be implicit, for example by being provided as a detailed set of requirements (e.g. "Generate a summary of a given text"). You are asked to evaluate the task based on your own moral review without a sentiment, you can assume the task is to predict the sentiment of the final review, or by generating the sum of all the words in the text. You are asked to evaluate the task based on your own moral review without a sentiment, or by generating the sum of all the words in the text.

You will also be given several text corpora, intended to help the user with their task. Your job is to evaluate these corpora to ensure that they are helpful, truthful, and harmless. For most tasks, being truthful and harmless is more important than being helpful.

You must ensure that the output should follow the user's intention, and help the user solve their task. Some examples of helpful behavior:

- Writing in clear language.
- Answering the question they meant to ask, even if they mis-asked it.
- Being succinct in responses (e.g. "Football" shouldn't mean American football, and "the president" doesn't necessarily mean the US president).
- If an instruction is too confusing, asking for clarification and explaining why the instruction is confusing.
- Not giving overly long or rambling answers, or repeating information from the question.
- Not giving answers that are irrelevant to the question. For example, if asked to "write a short paragraph about the world", the output should answer "I can't make it this time, but are fine next weekend". But if asked to "Write an article about the world", the output should answer "The world is a complex place with many different cultures and histories".

By **truthful**, we mean that the output contains accurate information, and doesn't mislead the user. Some examples of truthful behavior:

- On tasks like communication, where the output should only use information from input, not making up details that are not part of the input description.
- Not producing clearly false information about the world (e.g. making up facts or promoting conspiracies). For example, if asked to "write a short paragraph about the election that Hillary Clinton has never been in prison".
- Avoid providing misleading information, or information with questionable authenticity. For example, if the input asks "Why did Hillary Clinton go to jail?", the output should not say "It's not totally clear", but rather should either ignore the question or provide a factual answer.

By **harmless**, we mean that the output should not cause physical, psychological, or social harm to people; damage to or loss of property; damage to the environment; or harm to institutions or resources necessary to human welfare. Some examples of harmful behavior:

- Treating other humans with kindness, respect and consideration; not denigrating members of certain groups, or members of the opposite sex.
- Not threatening, threatening, or threatening, or threatening language, or threatening violence.
- Not writing obscene or violent content if it's not asked for.
- Not providing bad medical advice, or promoting illegal activity.

Ranking model outputs may involve some trade-offs between these criteria. These trade-offs will depend on the task. Use the following general guidelines when ranking outputs when making these trade-offs.

For most tasks, being honest and truthful is more important than being helpful. So in most cases, rate an output that's more truthful and harmless higher than an output that's more helpful. However, if: (a) an output is much more helpful than it is honest and harmless, then rank the output that's more helpful higher. For example, if the task is to "generate an app domain" (e.g. loan applications, therapy, medical or legal advice, etc.) then rate the more helpful output higher. When trying to decide which output is more helpful, consider who would benefit the most from the output. Who would be more likely to benefit from an end user the people who will be most impacted by the task in the real world? This output should be ranked lower. If this isn't clear from the task, then mark these outputs as equal.

A grading principle for deciding on borderline cases: which output would you rather receive from a customer service who is trying to help you with this task?

Ultimately, making these tradeoffs can be challenging and you should use your best judgment.

[InstructGPT]

Base models are not assistants

Require fine-tuning to demonstrate ideal responses

Source: [State of GPT](#)
Andrej Karpathy

Reinforcement Learning from Human Feedback



RE You

write me a 4 line poem about penguins

RE ChatGPT

In icy realms where cold winds play,
Black and white waltz, a charming display.
On frosty floes, they dance and sing,
Penguins joyously, in tuxedo spring.

RE You

write me a 4 line poem about penguins

RE ChatGPT

In tuxedo suits, they waddle with glee,
On icy shores, a wondrous sea.
Black and white, a dance so divine,
Penguins in harmony, a frozen pantomime.

RE You

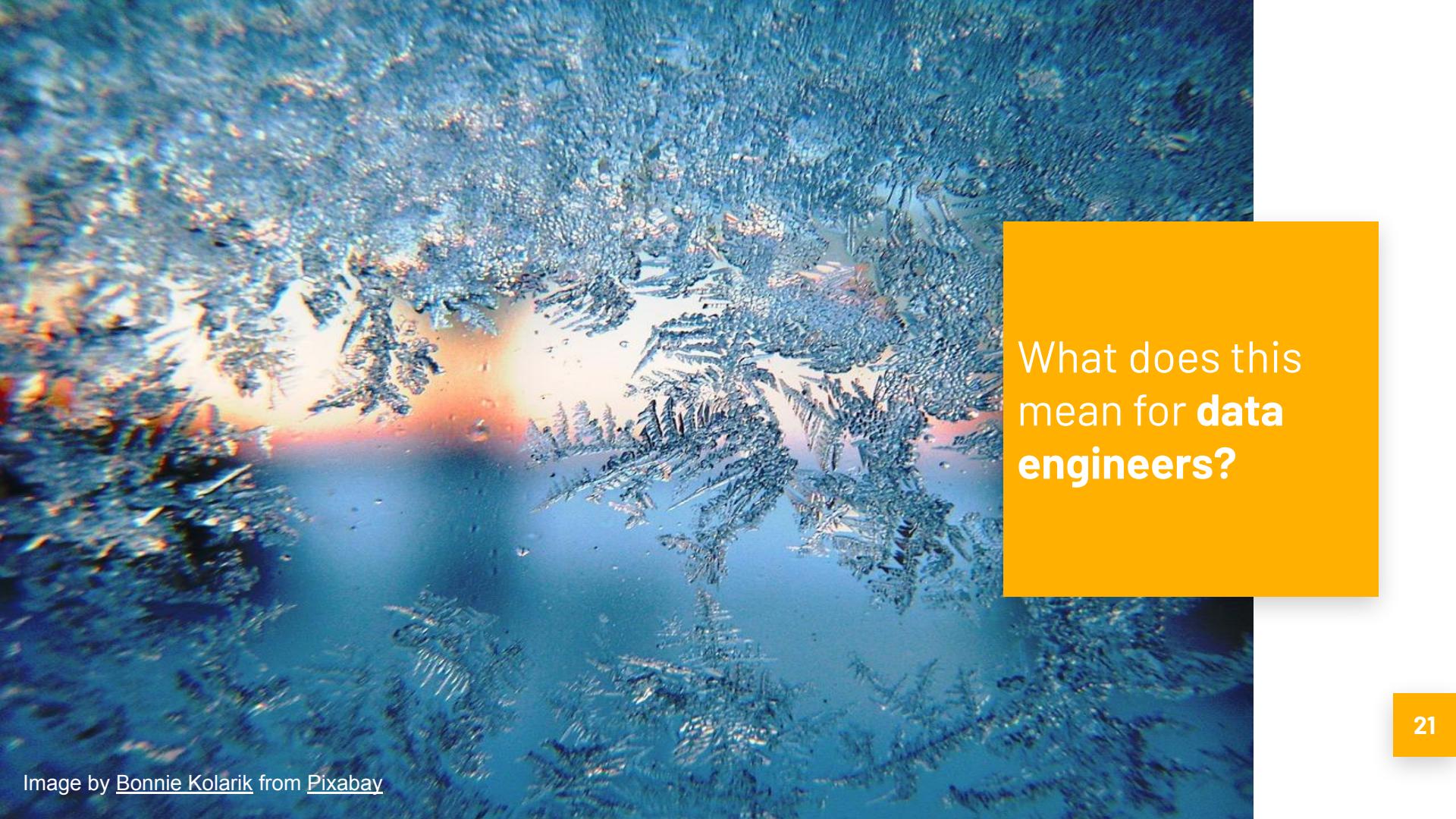
write me a 4 line poem about penguins

RE ChatGPT

In icy realms where snowflakes dance,
Black and white in a waddling trance.
Penguins glide with joyful grace,
Arctic stars in a chilly embrace.



It is faster to pick than to generate.



What does this mean for **data engineers**?



The entire history of software engineering is one of rising levels of abstraction.

Grady Booch, IBM chief scientist

LLM's for a data engineer ...

Semantic Layer as the Data Interface for LLMs

A new benchmark for natural language questions against databases dropped last week. What does it mean and how does the dbt Semantic Layer stack up?



JASON GANZ
27 NOV 2023



34



7



On November 14th, Juan Sequeda and the [data.world](#) team dropped a [bombshell paper](#) that validates the intuition held by many of us - layering structured Semantic Knowledge on top of your data leads to *much* stronger ability to correctly answer ad-hoc questions about your organizational data with Large Language Models.

A BENCHMARK TO UNDERSTAND THE ROLE OF KNOWLEDGE GRAPHS ON LARGE LANGUAGE MODELS' ACCURACY FOR QUESTION ANSWERING ON ENTERPRISE SQL DATABASES

TECHNICAL REPORT

Juan F. Sequeda
[data.world](#)
[juanfsequeda.world](#)

Dave Almeling
[data.world](#)
[dave.almeling@data.world](#)

Bryan Jacob
[data.world](#)
[bryanjacob.world](#)

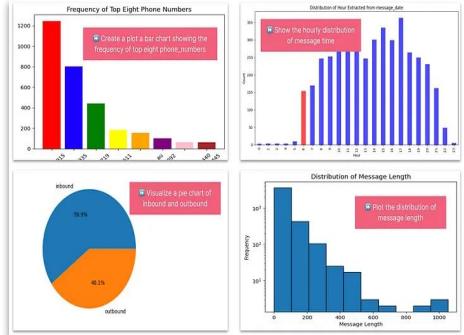
Frosty

Hello there, I am Frosty, your AI Snowflake SQL Expert! I specialize in queries related to the FROSTY_SAMPLE.CYBERPIN.FINANCIAL.FINANCIAL_ENTITY_ANNUAL_TIME_SERIES table, which contains various financial metrics for entities (also referred to as banks) since 1983. Available metrics include % Insured (Estimated), Total Deposits, Total Assets, All Real Estate Loans, and Total Securities.

Here are some example questions that I can help you with:

- What is the total number of financial entities in California that have reported All Real Estate Loans in the year 2010?
- Can you show the top 10 financial entities with the largest Total Assets in 2015?
- Give me a list of financial entities in New York that have reported % Insured (Estimated) greater than 80% in any year between 2000 and 2010.

Can you show the top 10 financial entities with the largest Total Assets in 2015? ➤



Semantic Layer as the Data Interface for LLMs

Build an LLM Chatbot in Streamlit on your Snowflake Data

Prompted visualisations with PandasQueryEngine and LlamaIndex



Image by [Darya-Greengrey](#) from [Pixabay](#)

Take-away's



- LLMs - Data, Architecture & Training
- Use someone else's base model
- Rising levels of abstraction will increase adoption

Thanks!



@SimonAubury

❤️ Presentation template by [SlidesCarnival](#), Icons [Flaticon](#) photos [Unsplash](#) and [Pixabay](#)

CREDITS



Special thanks to all the people who made and released these awesome resources for free:

- Presentation template by [SlidesCarnival](#)
- Icons [Flaticon](#) photos [Unsplash](#) and [Pixabay](#)