

Copilot for Splunk: Using Massive Language Models for Query Generation on Morpheus

splunk > turn data into doing®



Speakers



Abe Starosta

Senior Applied Scientist
Splunk Applied Research



Julien Veron Vialard

Applied Scientist
Splunk Applied Research

Agenda

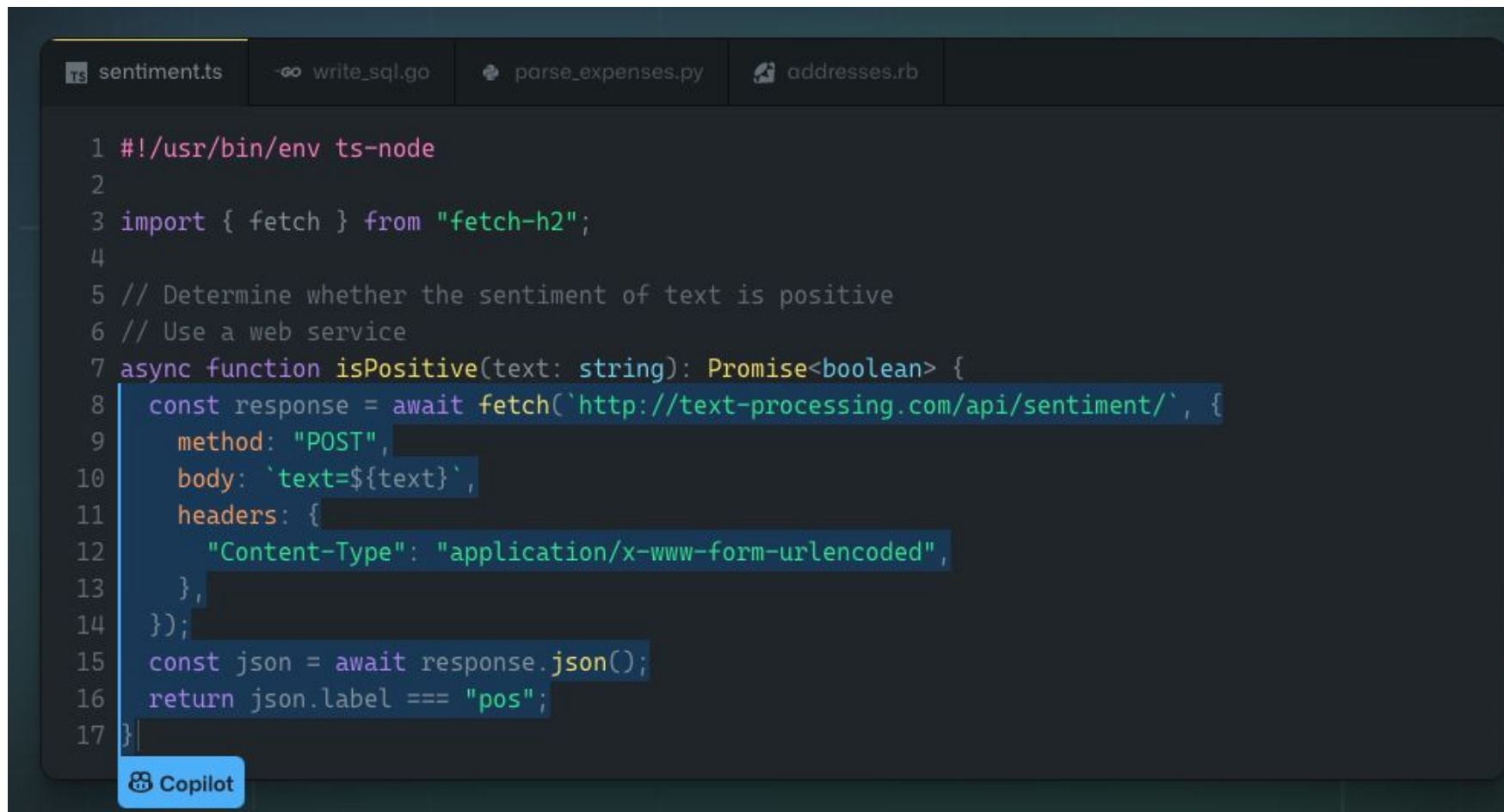
We collaborated with NVIDIA to fine tune a large language model “Copilot for Splunk”. Then, we used NVIDIA Morpheus to accelerate inference by at least 5x.

1. Problem statement: English to SPL translation
2. Massive language models
3. Getting training data
4. Modeling and experiments
5. Deployment on NVIDIA Morpheus



Problem Statement

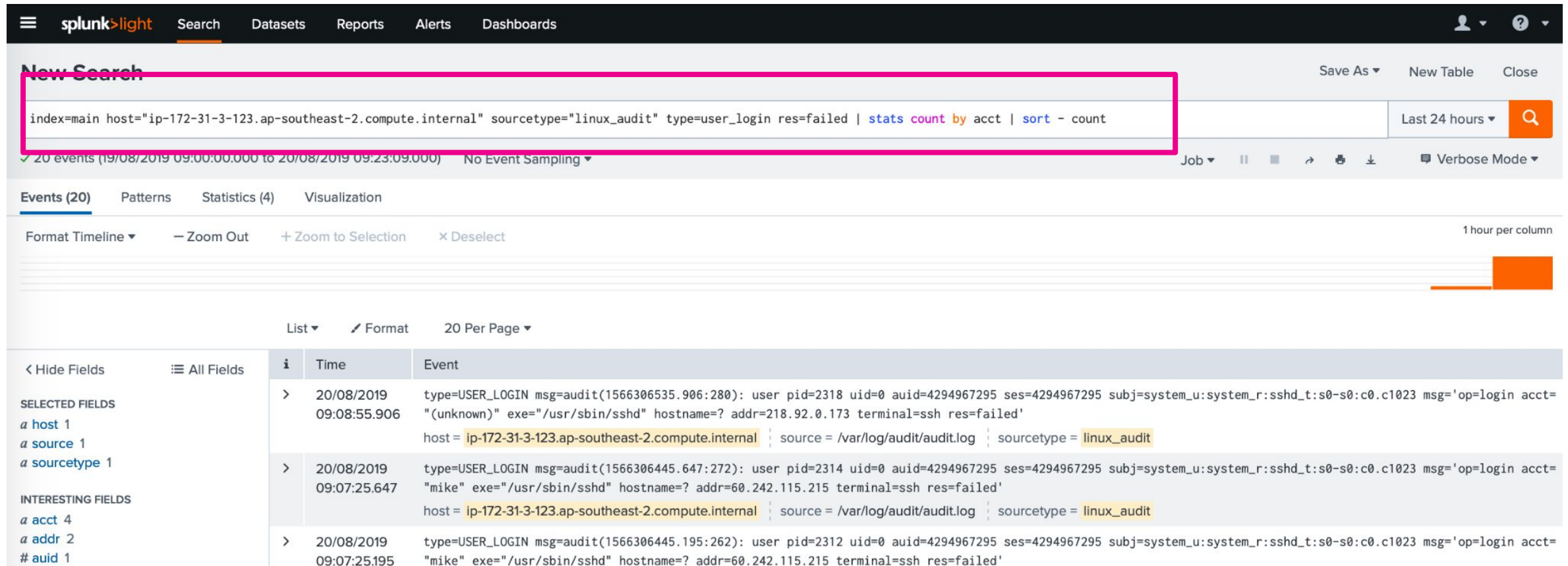
GitHub Copilot, for Splunk SPL



```
1 #!/usr/bin/env ts-node
2
3 import { fetch } from "fetch-h2";
4
5 // Determine whether the sentiment of text is positive
6 // Use a web service
7 async function isPositive(text: string): Promise<boolean> {
8   const response = await fetch('http://text-processing.com/api/sentiment/', {
9     method: "POST",
10    body: `text=${text}`,
11    headers: {
12      "Content-Type": "application/x-www-form-urlencoded",
13    },
14  });
15  const json = await response.json();
16  return json.label === "pos";
17 }
```

Copilot

SPL Search Bar



New Search Save As New Table Close

index=main host="ip-172-31-3-123.ap-southeast-2.compute.internal" sourcetype="linux_audit" type=user_login res=failed | stats count by acct | sort - count

✓ 20 events (19/08/2019 09:00:00.000 to 20/08/2019 09:23:09.000) No Event Sampling ▾

Job ▾ || ▢ ↗ ⏏ ⬇

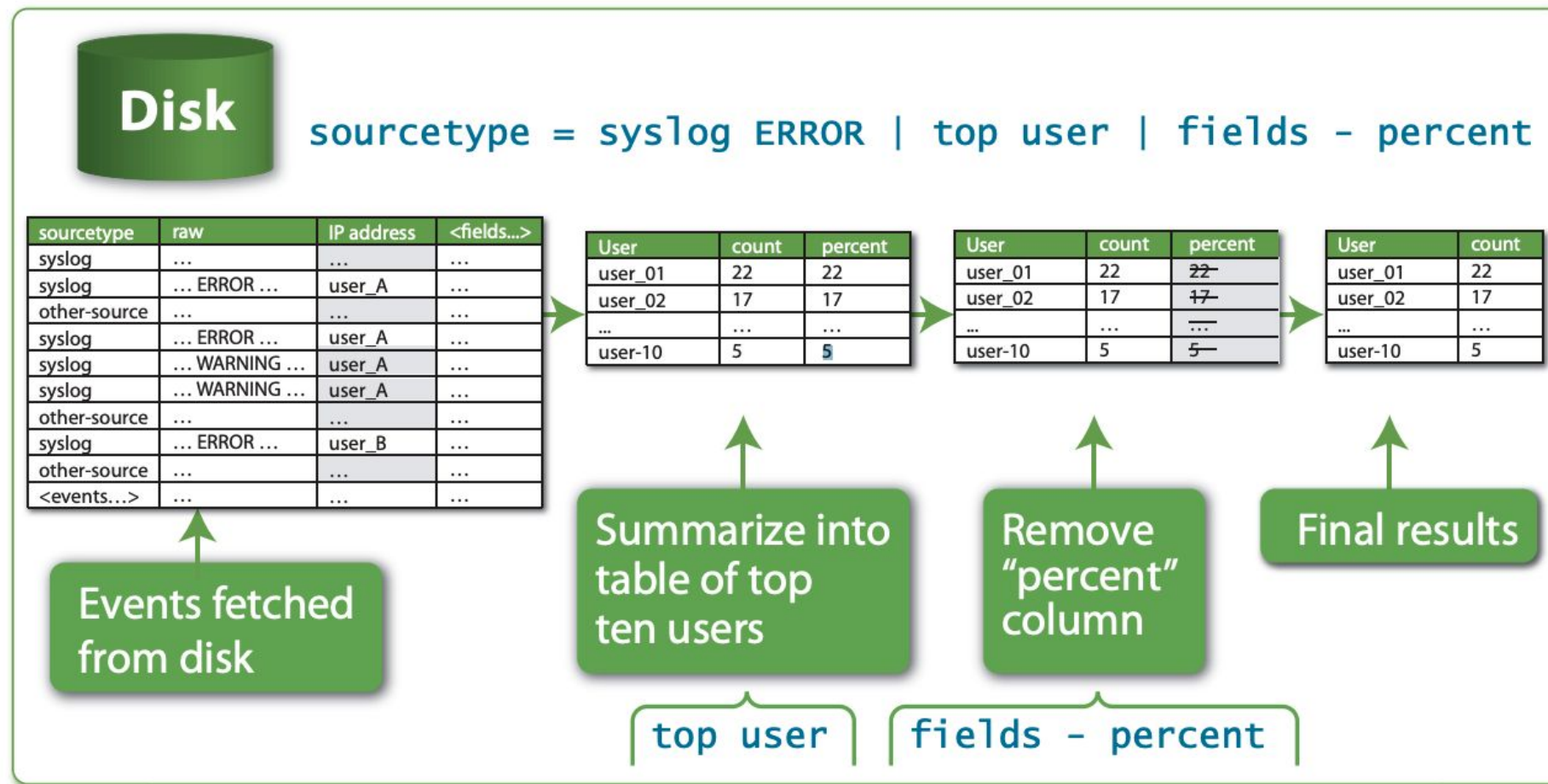
Events (20) Patterns Statistics (4) Visualization

Format Timeline ▾ — Zoom Out + Zoom to Selection × Deselect 1 hour per column

List ▾ ✎ Format 20 Per Page ▾

< Hide Fields	≡ All Fields	i	Time	Event
SELECTED FIELDS a host 1 a source 1 a sourcetype 1		>	20/08/2019 09:08:55.906	type=USER_LOGIN msg=audit(1566306535.906:280): user pid=2318 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:sshd_t:s0-s0:c0.c1023 msg='op=login acct="(unknown)" exe="/usr/sbin/sshd" hostname=? addr=218.92.0.173 terminal=ssh res=failed' host= ip-172-31-3-123.ap-southeast-2.compute.internal source = /var/log/audit/audit.log sourcetype = linux_audit
INTERESTING FIELDS a acct 4 a addr 2 # auid 1		>	20/08/2019 09:07:25.647	type=USER_LOGIN msg=audit(1566306445.647:272): user pid=2314 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:sshd_t:s0-s0:c0.c1023 msg='op=login acct="mike" exe="/usr/sbin/sshd" hostname=? addr=60.242.115.215 terminal=ssh res=failed' host= ip-172-31-3-123.ap-southeast-2.compute.internal source = /var/log/audit/audit.log sourcetype = linux_audit
		>	20/08/2019 09:07:25.195	type=USER_LOGIN msg=audit(1566306445.195:262): user pid=2312 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:sshd_t:s0-s0:c0.c1023 msg='op=login acct="mike" exe="/usr/sbin/sshd" hostname=? addr=60.242.115.215 terminal=ssh res=failed'

What's Splunk SPL?



SPL is Similar to SQL

SQL command	SQL example	Splunk SPL example
SELECT *	<pre>SELECT * FROM mytable</pre>	<pre>source=mytable</pre>
WHERE	<pre>SELECT * FROM mytable WHERE mycolumn=5</pre>	<pre>source=mytable mycolumn=5</pre>
SELECT	<pre>SELECT mycolumn1, mycolumn2 FROM mytable</pre>	<pre>source=mytable FIELDS mycolumn1, mycolumn2</pre>
AND/OR	<pre>SELECT * FROM mytable WHERE (mycolumn1="true" OR mycolumn2="red") AND mycolumn3="blue"</pre>	<pre>source=mytable AND (mycolumn1="true" OR mycolumn2="red") AND mycolumn3="blue"</pre> <p>Note: The AND operator is implied in SPL and does not need to be specified. For this example you could also use:</p> <pre>source=mytable (mycolumn1="true" OR mycolumn2="red") mycolumn3="blue"</pre>

Goal: Translate English to SPL

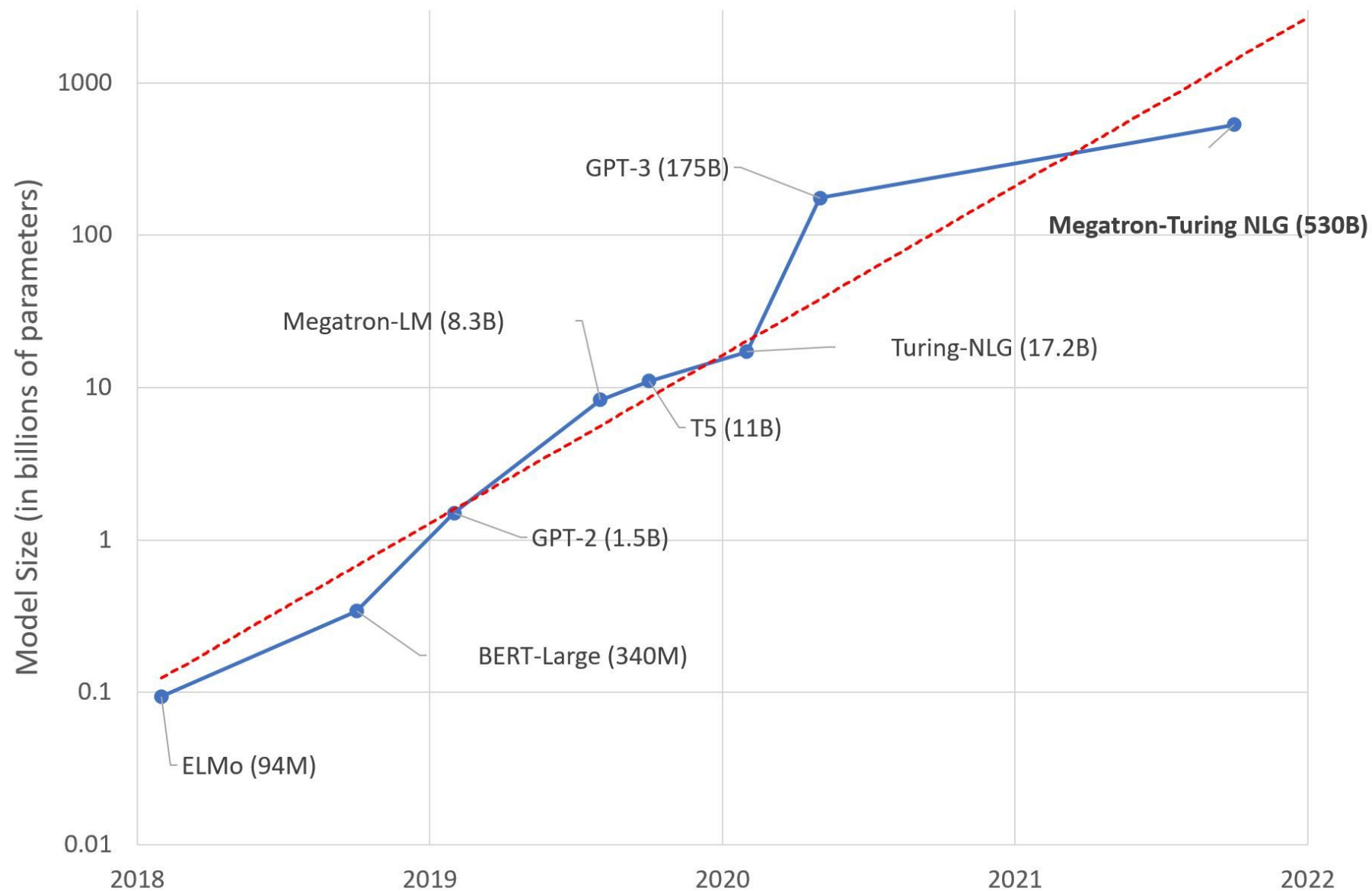
“From linux security logs, return number of events by user.”



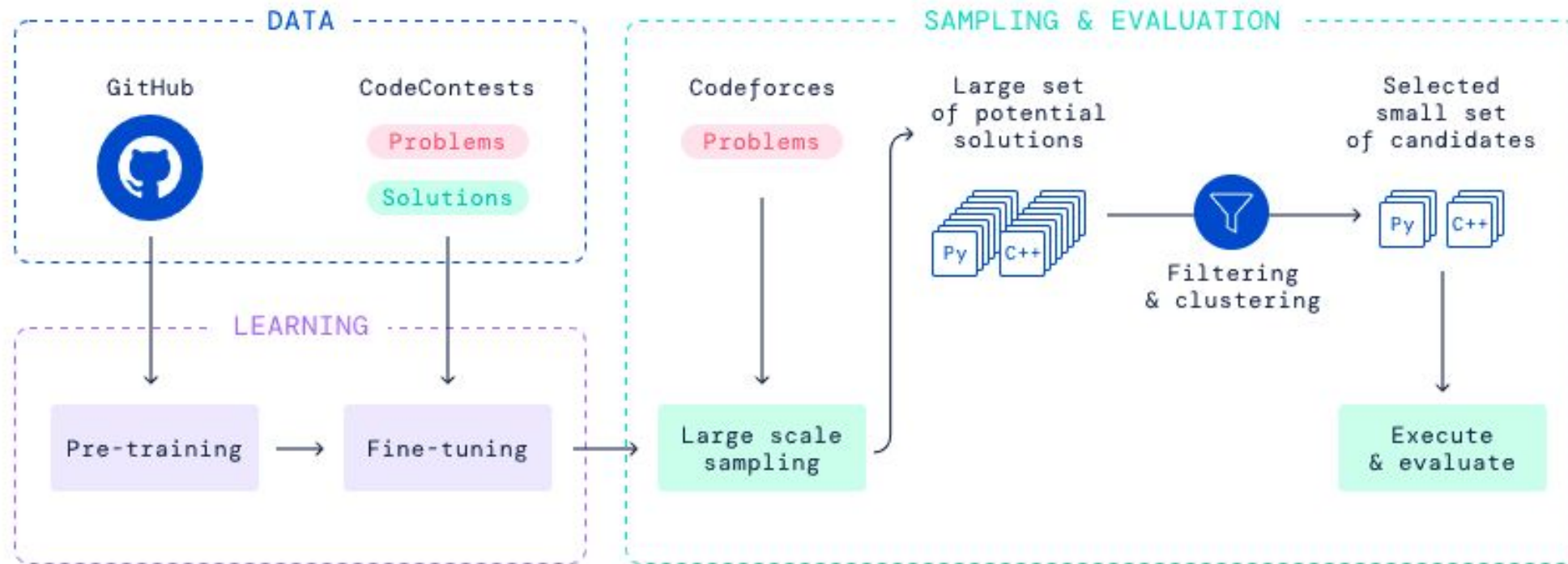
Translation
Model

```
sourcetype=linux_secure  
| rex "\suser[^\s]"(?<User>\S+\w+)"  
| stats count by User
```

Massive Language Models

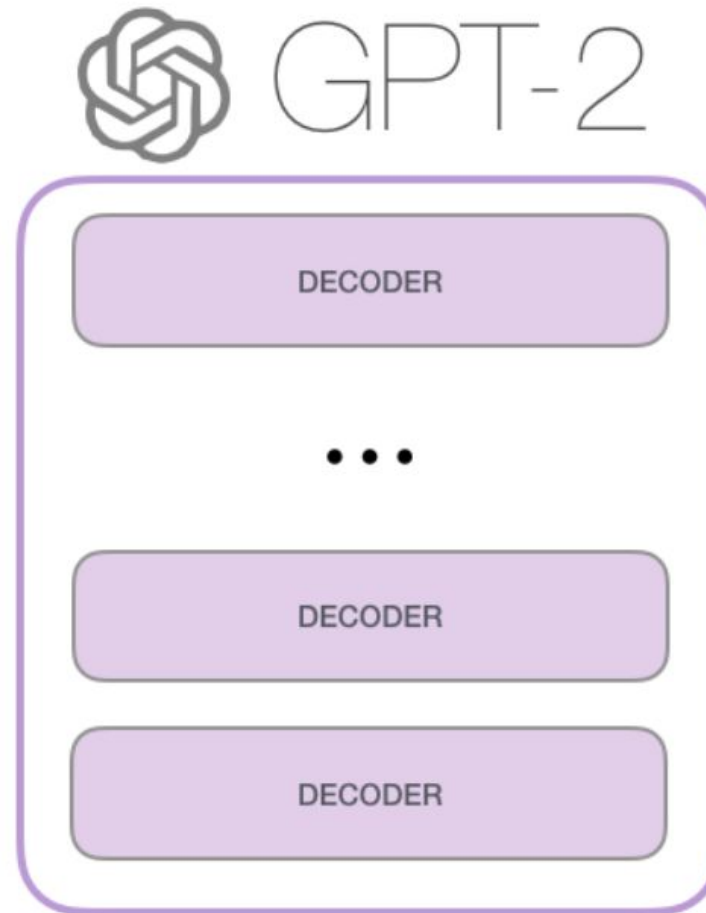


AlphaCode for Competitive Programming



<https://deepmind.com/blog/article/Competitive-programming-with-AlphaCode>

GPT and Codex Power Github Copilot

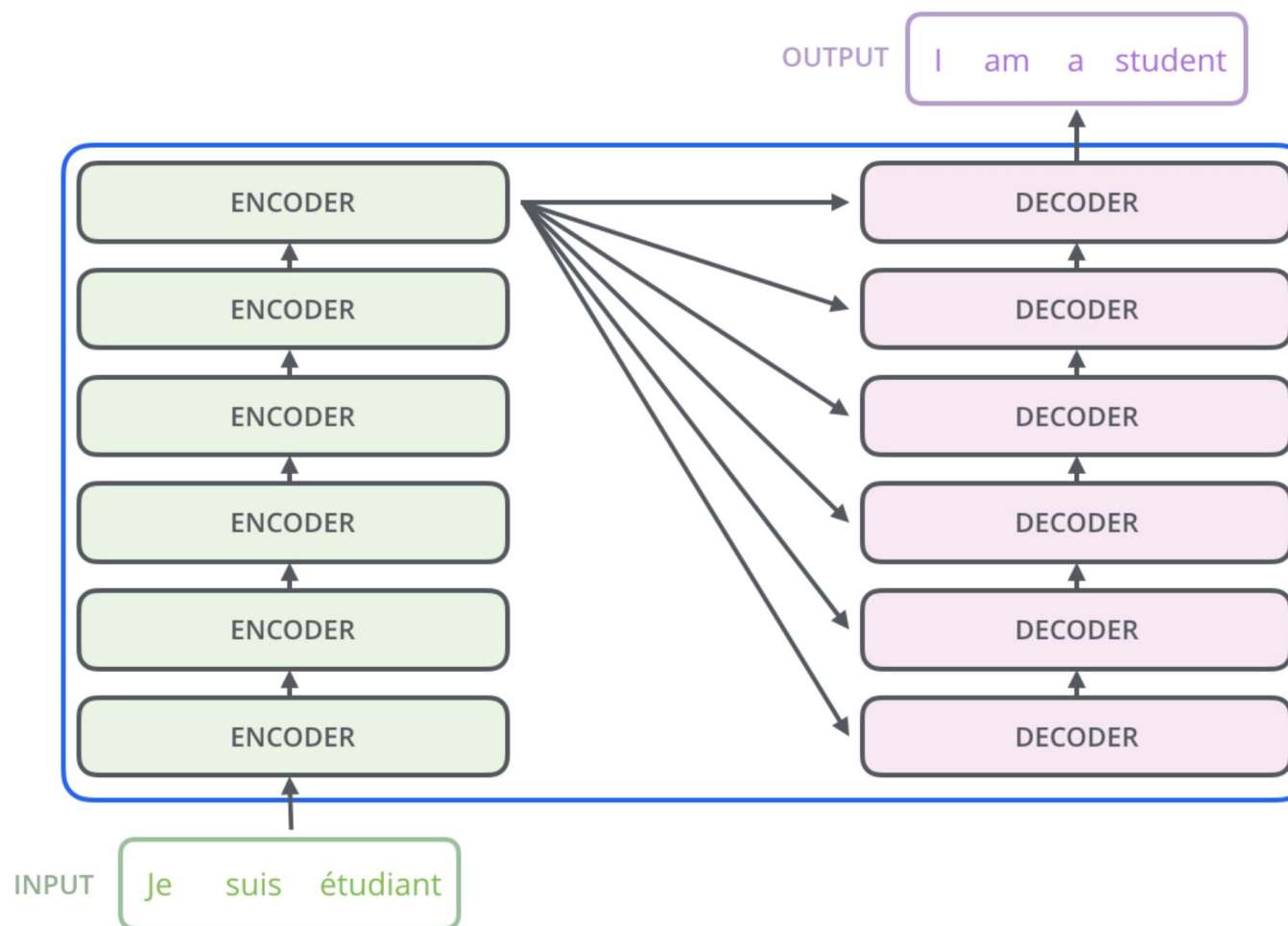


<https://jalammar.github.io/illustrated-gpt2/>

Microsoft Using GPT-3 in Production

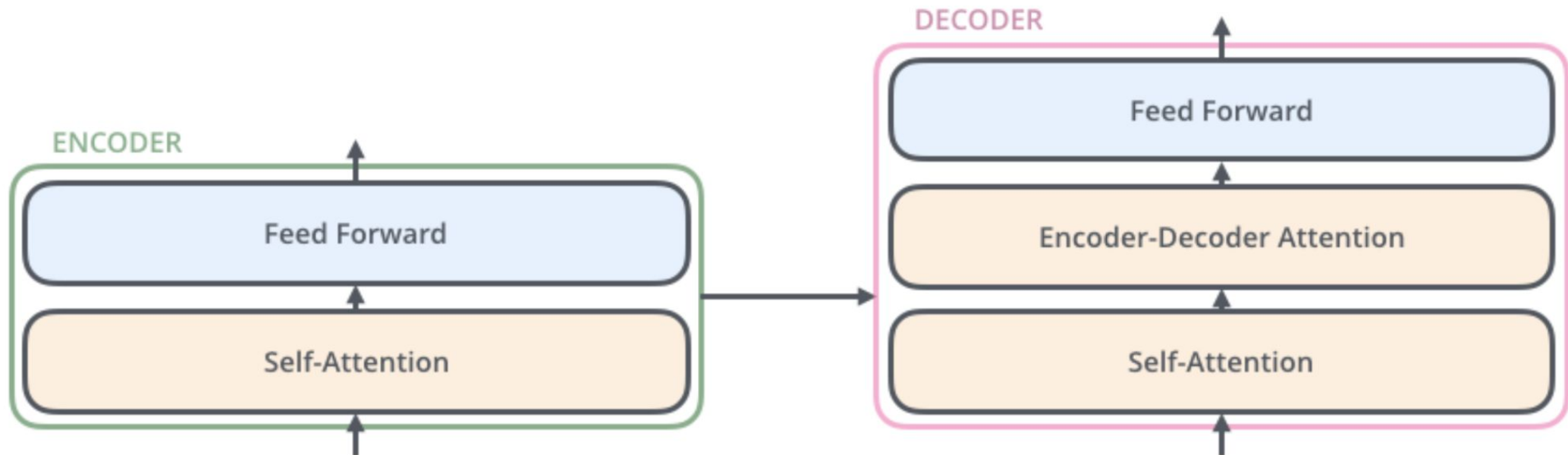
[Link](#)

Transformer



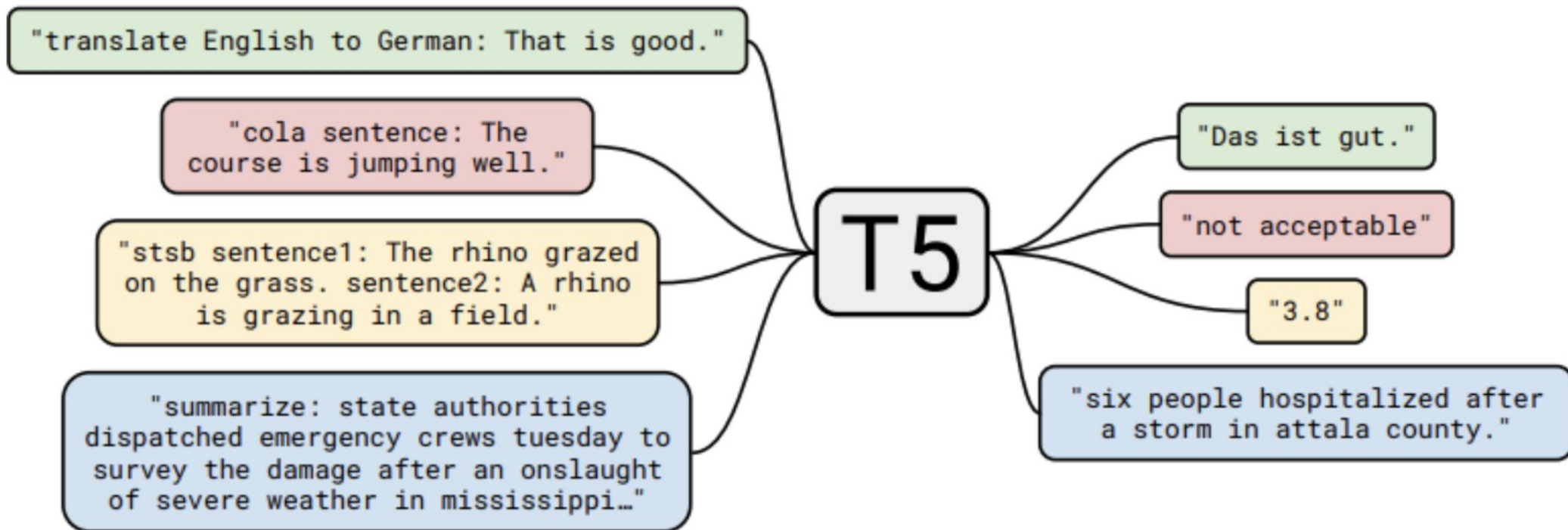
<https://jalammar.github.io/illustrated-transformer/>

Transformer



<https://jalammar.github.io/illustrated-transformer/>

Text-To-Text Transfer Transformer: T5



<https://arxiv.org/abs/1910.10683>

Getting Training Data

Scraped Data Sources

Dataset	# All Scraped Examples	# Manually Curated Examples
Splunk Community Forum	82,030	494
Splunk Online Documentation	682	439
Splunk Apps	1,735	432
Splunk SPL Handbooks (PDF)	324	300
GoSplunk SPL Database	609	42
Total	85,380	1,707

Translating Benchmark Datasets for Data Augmentation

Dataset	# Translation Examples
WikiSPL	80,036
SpiderSPL	1,456
Total	81,492

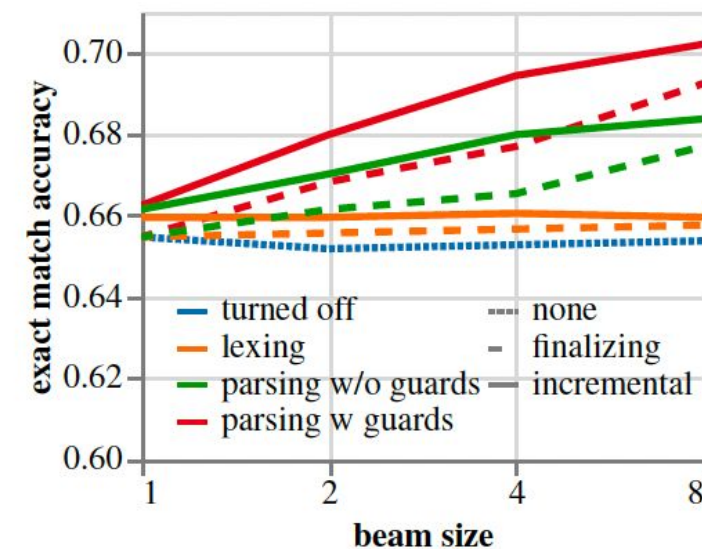


Modeling and Experiments

Choosing T5

- Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, Raffel et al. 2020
- Spider: Yale Semantic Parsing and Text-to-SQL Challenge

Constraining T5: PICARD, Scholak et al. 2021. Rank #1 with 75.5 exact-set-match accuracy.



<https://arxiv.org/pdf/2109.05093.pdf>

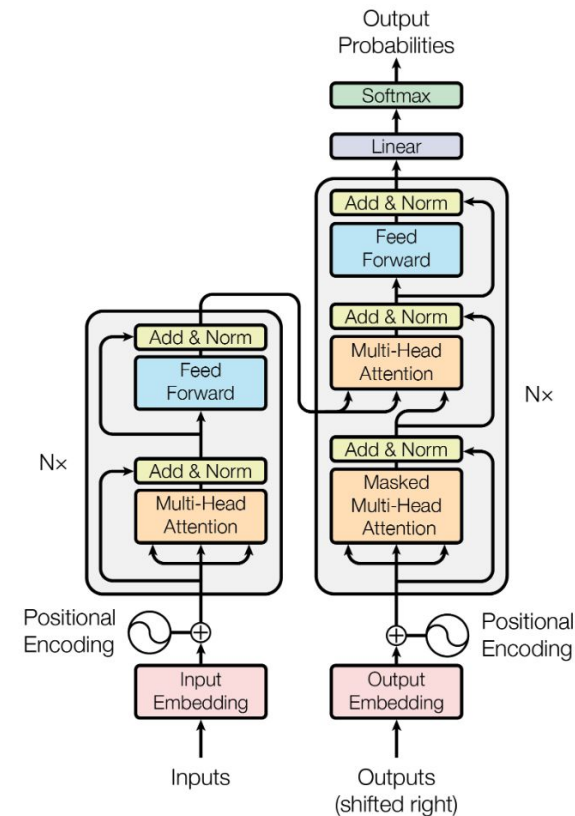
Model Architecture



Hugging Face transformers

Pre-trained models:

- [t5-small](#)
- [Salesforce/codet5-small](#)



<https://arxiv.org/pdf/1706.03762.pdf>

WikiSQL

80,654 (70% train, 10% validation, and 20% test) hand-annotated SQL query and natural language question pairs. “Human readable” simple queries.

What is the current series where the new series began in June 2011?

```
SELECT Current series FROM table WHERE Notes = New series began in June 2011
```

Name the background colour for the Australian Capital Territory

```
SELECT Text/background colour FROM table WHERE State/territory = Australian Capital Territory
```

Testing on WikiSQL

		BLEU score	exact match
WikiSQL	t5-small	76.0	38.4%
WikiSQL	codet5-small	74.8	37.2%

	english	sql	predicted sql
0	name the number of ranks for international tourist arrivals being 6.2 million	select count rank from table where international tourist arrivals (2011) = 6.2 million	select count rank from table where international tourist arrivals = 6.2 million
1	what format is after 1991 and has a catalog number of 81868?	select format from table where year > 1991 and catalog number = 81868	select format from table where year > 1991 and catalog = 81868
2	which total is the highest one that has a bronze of 0, and a nation of poland, and a gold smaller than 0?	select max total from table where bronze = 0 and nation = poland and gold < 0	select max total from table where bronze = 0 and nation = poland and gold
3	name the average total for tournament less than 0	select avg total from table where tournament < 0	select avg total from table where tournament 0
4	which player had a position of t1 and played in the united states?	select player from table where place = t1 and country = united states	select player from table where position = t1 and country = united states

Testing on “WikiSPL”

		BLEU score	exact match
WikiSPL	t5-small	73.2	30.0%
WikiSPL	codet5-small	70.7	24.8%

	english	spl	predicted spl
0	from index "table", return what is the minimum year born for strasbourg	index=table where current club = strasbourg fields min_year_born	index=table where name = strasbourg fields min_year_born
1	from source "table", return how many kit makers are there for louis carey's team	source=table where team captain = louis carey fields count_kit_maker	source=table where player = louis carey fields count_kit_maker
2	from source "table", return which week had an attendance of 51,265	source=table where attendance = 51,265 fields week	source=table where attendance = 51,265 fields count_week
3	from sourcetype "table", return what is the atlantic europe when age is 10,000 years	sourcetype=table where age (before) = 10,000 years fields atlantic_europe	sourcetype=table where age (in 2007) = 10,000 years fields atlantic_
4	from source "table", return what is the highest value for col(m) when prominence(m) is 3755	source=table where prominence (m) = 3755 fields max_col_(m)	source=table where prominence(m) = 3755 fields max_col_(

Spider

8,026 questions and database-executable SQL query pairs.

How many heads of the departments are older than 56 ?

```
SELECT count(*) FROM head WHERE age > 56
```

Find the name of the candidates whose oppose percentage is the lowest for each sex.

```
SELECT t1.name , t1.sex , min(oppose_rate) FROM people AS t1 JOIN candidate AS t2 ON t1.people_id = t2.people_id GROUP BY t1.sex
```

Testing on Spider

	Spider
	t5-small
BLEU score	29.6
exact match	6.2%
executable	16.3%
executable & correct	12.2%
exact match (10 attempts)	9.7%
executable & correct (10 attempts)	20.2%

Testing on Manually Curated SPL Dataset

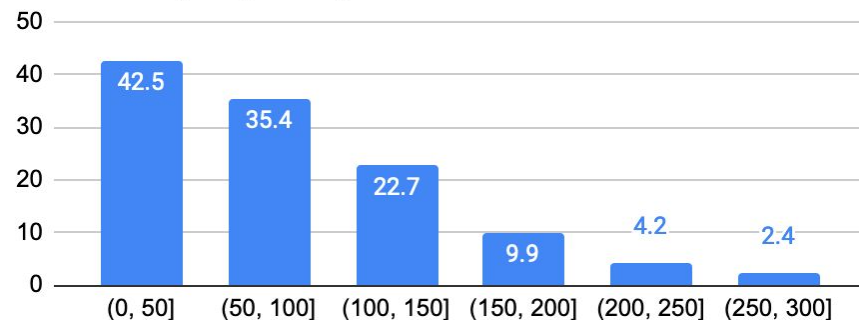
1,635 translation pairs (5-fold cross validation). More complex queries. Optionally filter invalid queries at generation time.

		BLEU score	filter + BLEU score	exact match	filter + exact match	exact match (10 attempts)
SPL dataset	t5-small	35.0 +/- 1.8	N/A	12.2% +/- 1.7	N/A	N/A
SPL dataset	codet5-small	39.4 +/- 5.7	42.9	17.7% +/- 4.8	20.2%	28.7% +/- 5.1
SPL dataset	codet5-small + freeze	39.2 +/- 3.4	N/A	17.7% +/- 2.5	N/A	N/A

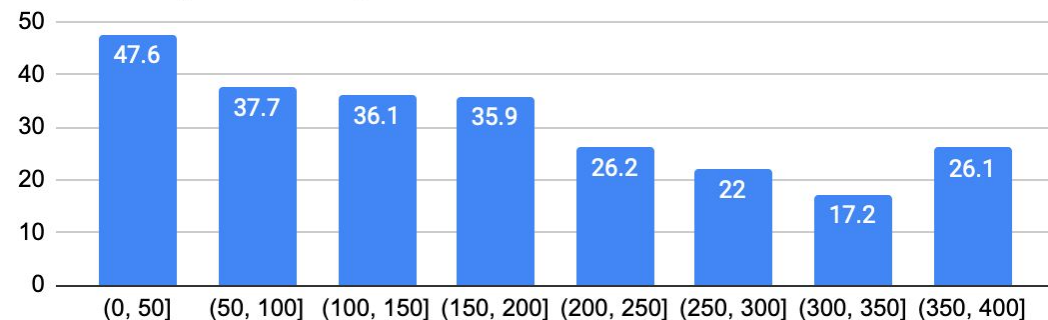
Testing on Manually Curated SPL Dataset

Better when the query is short (“easy” task) and the English is short (model “not confused”)

bleu score by target length

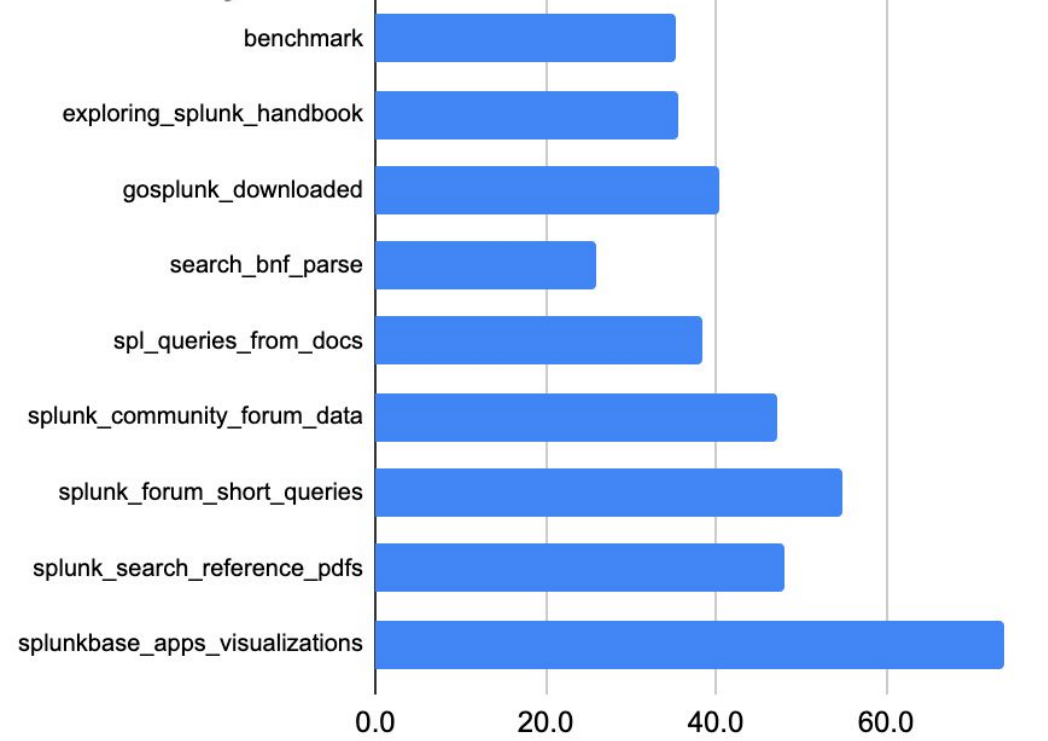


bleu score by source length



Testing on Manually Curated SPL Dataset

bleu score by dataset



Testing on Splunk Search Tutorial

Almost correct predictions

English	search the sourcetype field for any values that begin with access_. get events with status 200, action "purchase". then compute the most common categoryId values
SPL	sourcetype=access_* status=200 action=purchase top categoryId
Predicted SPL	sourcetype=access_* status=200 actionpurchase top categoryId
English	search the sourcetype field for any values that begin with access_. then get price as "Price" by productName, then rename productName column as "Product Name"
SPL	sourcetype=access_* stats values(price) AS Price BY productName rename productName AS "Product Name"
Predicted SPL	sourcetype=access_. stats values(price) as price by productName
English	search the sourcetype field for any values that begin with access_. get events with status 200, action "purchase". then compute the one most common clientip
SPL	sourcetype=access_* status=200 action=purchase top limit=1 clientip
Predicted SPL	sourcetype=access_* status=200 actionpurchase top clientip

Testing on Splunk Search Tutorial

Exact match if 10 attempts

English	search the sourcetype field for any values that begin with access_. use the HTTP status field, status, to specify successful requests and the action field to search only for purchase events.
SPL	sourcetype=access_* status=200 action=purchase
Predicted SPL	sourcetype=access_* status=200 action=purchase
English	search the sourcetype field for any values that begin with access_. get events with status 200, action "purchase". then compute the most common categoryId values
SPL	sourcetype=access_* status=200 action=purchase top categoryId
Predicted SPL	sourcetype=access_* status=200 action=purchase top categoryId

Testing on Splunk Search Tutorial

Bad predictions

English	search for the terms error, fail, failure, failed, or severe, in the events that also mention buttercupgames
SPL	buttercupgames (error OR fail* OR severe)
Predicted SPL	index=_internal fail* not [search index=_internal error, fail* error
English	search the sourcetype field for any values that begin with access_, status 200, and action "purchase". use the chart command to count the number of purchases by using action="purchase". The search specifies the purchases made for each product by using categoryId. The difference is that the count of purchases is now an argument of the sparkline() function.
SPL	sourcetype=access_* status=200 action=purchase chart sparkline(count) AS "Purchases Trend" count AS Total BY categoryId rename categoryId AS Category
Predicted SPL	index=access_* status=200 action=purchase chart count(price)

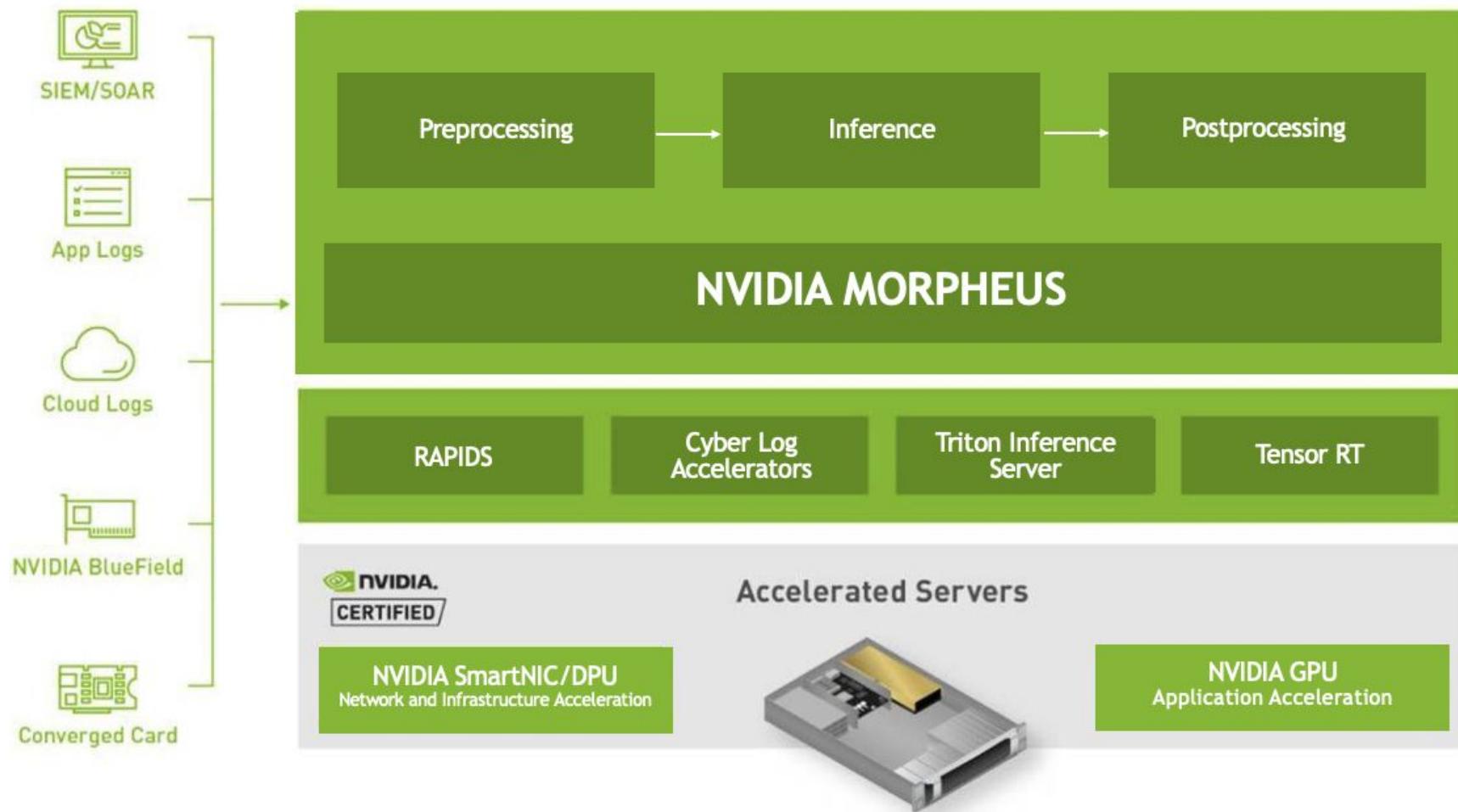
Other Experiments and Future Work

- Larger models and choice of tokenizer: **negligible** impact
- Data augmentation
template: **negligible** impact
“SQL converted to SPL” subsampling: **degradation**
- Data filtering on very rare commands and functions: **negligible** impact
- SPL to English model
for explaining complex queries
for data augmentation using monolingual SPL data
- Incorporating SPL parser and grammar

Deployment on Morpheus

NVIDIA Morpheus

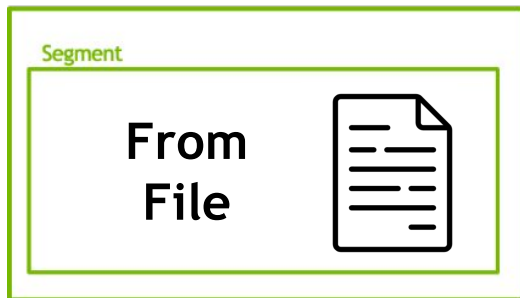
A modular AI cybersecurity framework



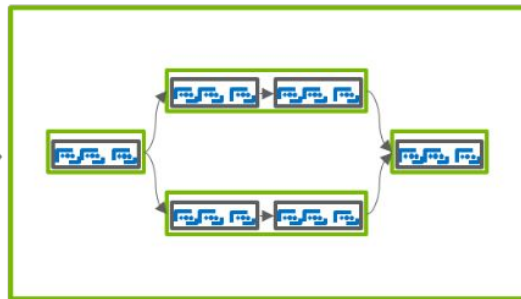
Morpheus Structure

Stages can wrap other technologies, both NVIDIA and non-NVIDIA

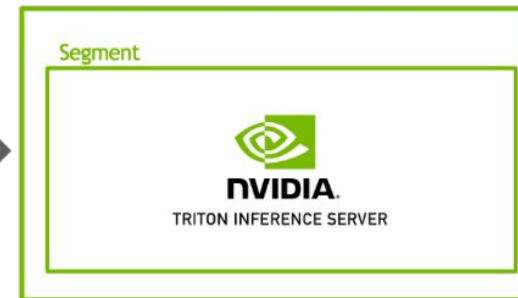
Input



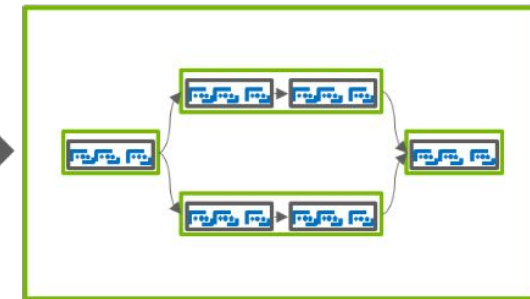
Pre-Process



Inference



Post-Processing



From Prototype to Production

Easily swap Morpheus stages to move between testing and production

Read and write to
file during
prototyping

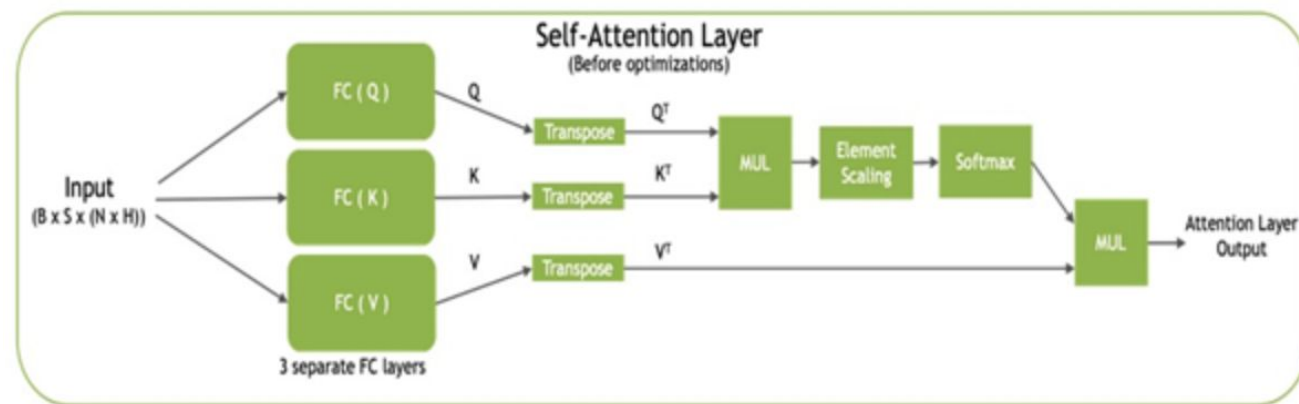
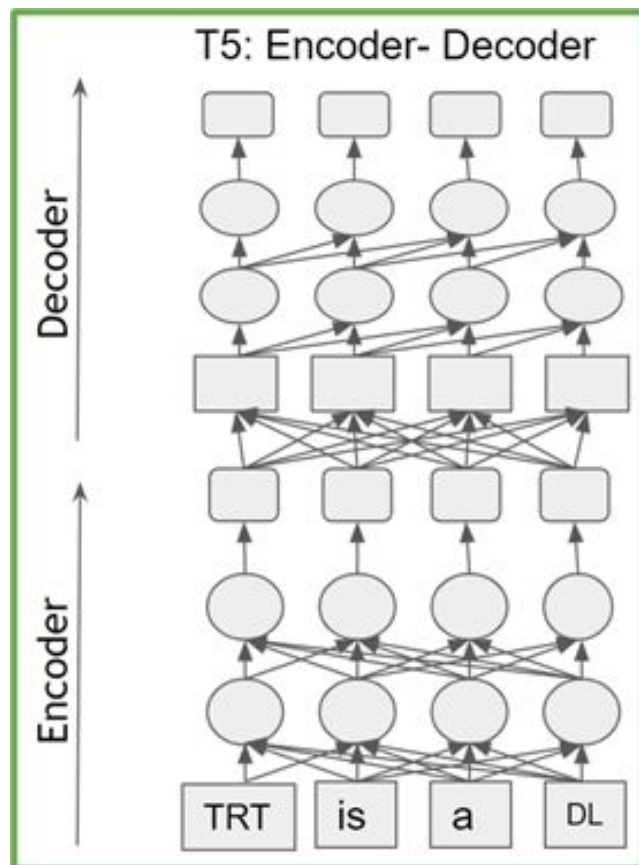
```
morpheus run --num_threads=4 \  
  --edge_buffer_size=4 \  
  pipeline-nlp --model_seq_length=256 \  
  from-file ./data/input.jsonlines \  
  deserialize \  
  preprocess --vocab_file=./data/T5.spm --add_special_tokens=False \  
  monitor --description='Preprocessing rate' \  
  inf-triton --model_name=T5-small --server_url=ai-engine:8001 \  
  monitor --description='Inference rate' --unit inf \  
  serialize --output_type='json' \  
  to-file ./data/output.jsonlines
```

Read and write to
Kafka topic in
production

```
morpheus run --num_threads=4 \  
  --edge_buffer_size=4 \  
  pipeline-nlp --model_seq_length=256 \  
  from-kafka --input_topic copilot_messages --bootstrap_servers broker:9092 \  
  deserialize \  
  preprocess --vocab_file=./data/T5.spm --add_special_tokens=False \  
  monitor --description='Preprocessing rate' \  
  inf-triton --model_name=T5-small --server_url=ai-engine:8001 \  
  monitor --description='Inference rate' --unit inf \  
  serialize --output_type='json' \  
  to-kafka --output_topic copilot_messages --bootstrap_servers broker:9092
```

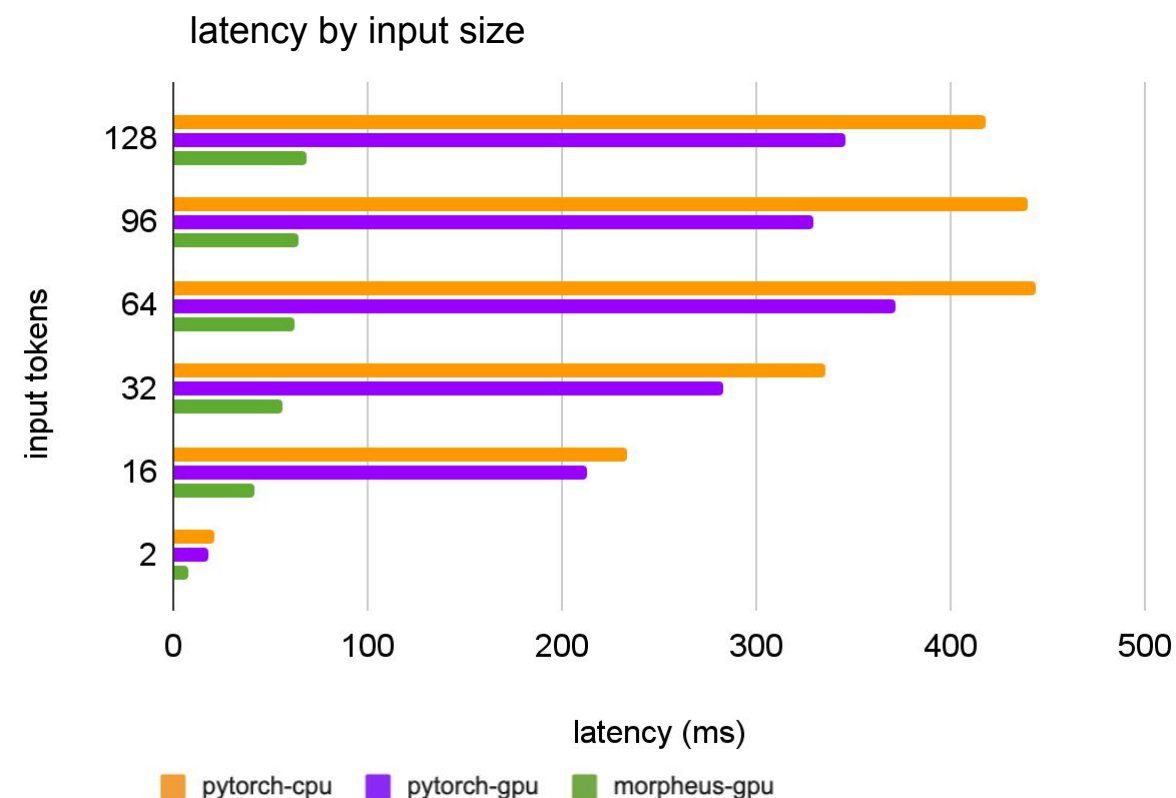
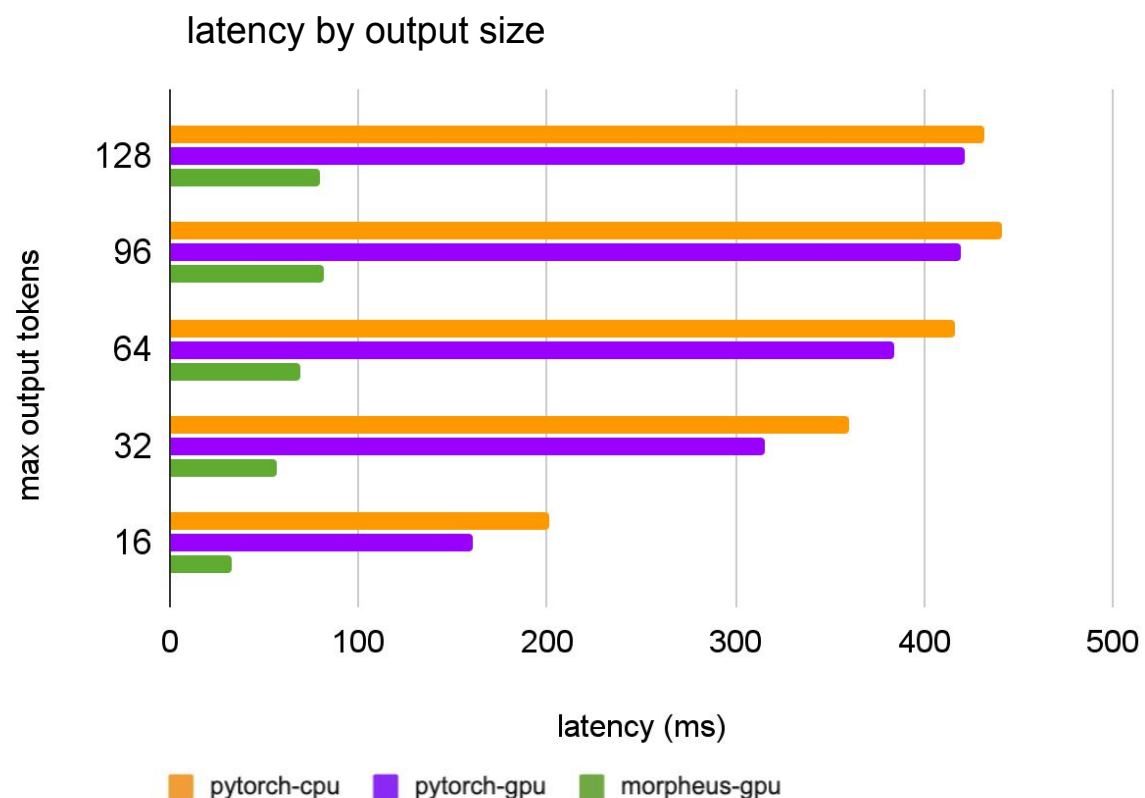
Converting T5 From PyTorch to TensorRT

Generating optimized runtime engines using TensorRT



TensorRT optimizes the self-attention block by pointwise layer fusion and also optimizes the network for inference.

Latency by Output and Input Sizes



GPU- single V100, CPU- Xeon Dual core, batch_size=1

Thank You!

