torchdistill Meets Hugging Face Libraries for Reproducible, Coding-Free Deep Learning Studies: A Case Study on NLP

Yoshitomo Matsubara* (University of California, Irvine) *This work was done prior to joining Amazon

OpenReview GitHub Hugging Face Read the Docs

```
$ pip3 install torchdistill
```

torchdistill

- PyYAML configuration-driven ML OSS build on PyTorch
- Lower barriers to **reproducible**, **coding-free** deep learning / knowledge distillation studies

A typical script for model training

```
$ python3 run_task.py \
    --model_name_or_path bert-large-uncased \
    --dataset_name mnli \
    --do_train \
    --do_eval \
    --max_seq_length 128 \
    --per_device_train_batch_size 32 \
    --per_device_eval_batch_size 32 \
    --learning_rate 3e-5 \
    --num_train_epochs 3 \
    --optim adamw_torch \
    --adam_epsilon 1e-8 \
    --logging_dir ./my_experiment_20231206/ \
    --output_dir ./my_experiment_20231206/
```

```
$ python3 run_task.py \
      --config your_config.yaml \
      --run_log your_log.txt
```

ML Tasks with Packages of Your Choice e.g.,

transformers datasets evaluate accelerate timm

torchdistill

PyTorch torchvision

YAML

Almost everything to design an experiment

Pipeline w/ abstracted modules

- Package imports
- Datasets & Loaders
- Model initialization
- Training + Epoch-wise validation
- Evaluation (Test)

Evaluation result, training log, model weights

Abstracted modules

- Dependencies
- Datasets
- Preprocessing
- Data loaders
- Models
- Model wrappers
- Forward hooks
- Forward inferences
- Loss functions
- Optimizer
- LR scheduler
- Stage-wise config
- and more!

Two Decades of the ACL Anthology

Development, Impact, and Open Challenges

Marcel Bollmann, Nathan Schneider, Arne Köhn, Matt Post

- The ACL Anthology is developed in a **public Github repository**.

  github.com/acl-org/acl-anthology/

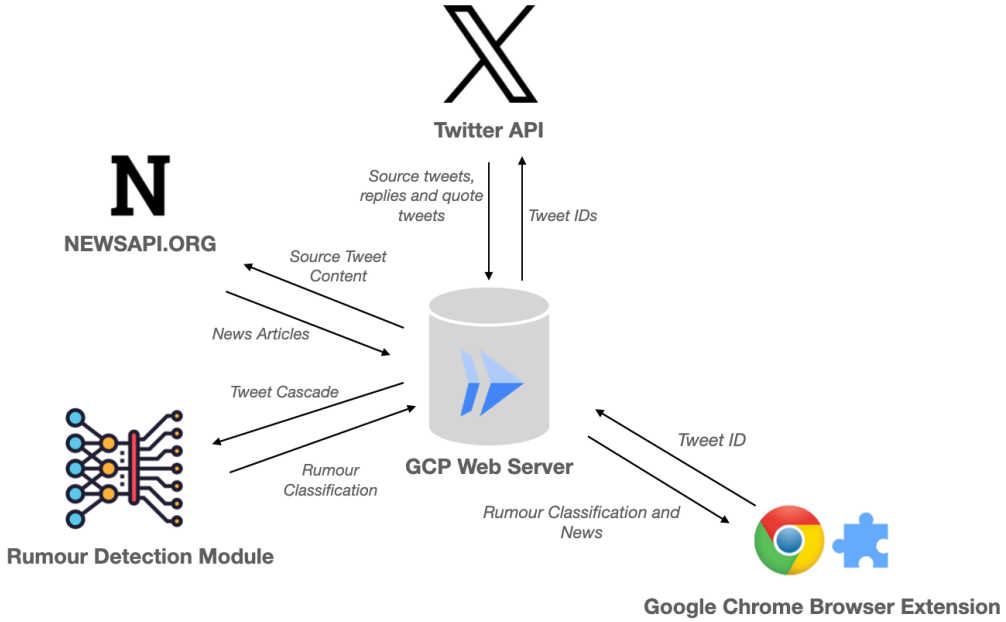
- It's mostly volunteer-driven & powered by **Python** and **Hugo.**
- All **metadata** is stored in the repo in XML and YAML formats.
- There's now a new **Python library** for accessing this metadata easily!

```
pip install acl-anthology-py
```

Everyone can contribute to the ACL Anthology!

LINKÖPING UNIVERSITY

Rumour Detection in the Wild: A Browser Extension for Twitter



TL;DR: Users derive benefit whilst using a browser extension leveraging SoTA models for rumour detection in real time.

Andrej Jovanović and Björn Ross

The University of Edinburgh, Edinburgh, United Kingdom

contact.me.maddox@gmail.com @itsmaddox_j

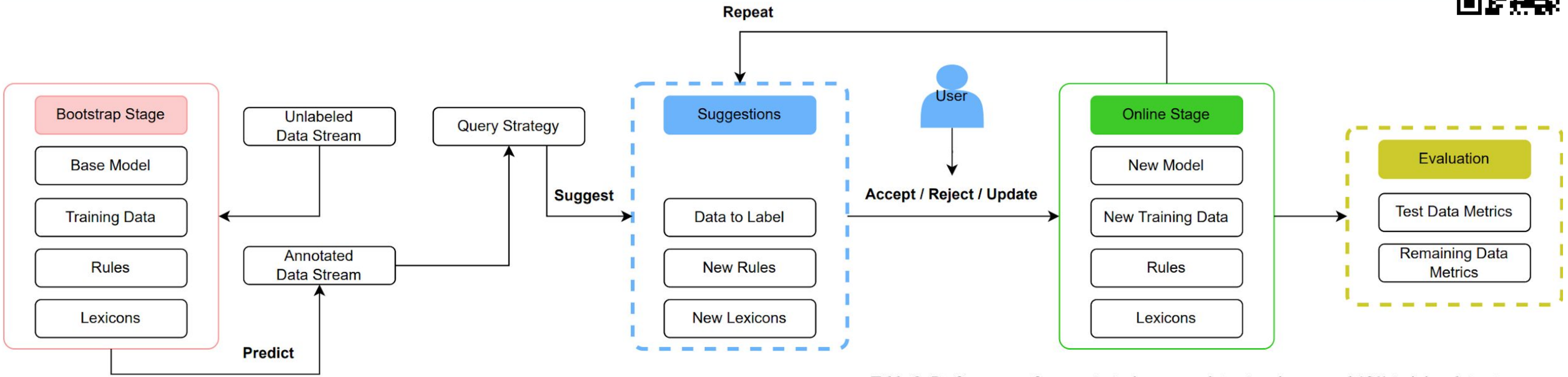
b.ross@ed.ac.uk @bjoernross

PyTAIL: An Open Source Tool for Interactive and Incremental Learning of NLP Models with Human in the Loop for Online Data

3rd Workshop for Natural Language Processing Open Source Software (NLP-OSS)
6 Dec 2023 @ EMNLP 2023 in Singapore

Shubhanshu Mishra* (shubhanshu.com), Jana Diesner (University of Illinois at Urbana-Champaign),
*Work done while at UIUC

ArXiv: https://arxiv.org/abs/2211.13786
Dataset: https://doi.org/10.5281/zenodo.7236430
Code: https://github.com/socialmediaie/pytail



Problem formulation

- Given a large unlabeled corpus, can we:
  - label it efficiently using fewer human annotations?
  - allow human-in-the-loop injection of rules?
  - update models efficiently to work with new data?
- Proposal:
  - Use active learning for data labeling
  - Use interface to surface and inject prominent rules
  - Use incremental learning algorithms for model
- Highly applicable to social media data:
  - Model should adapt to new and streaming data

PyTAIL Benchmark for Social Media Active Learning

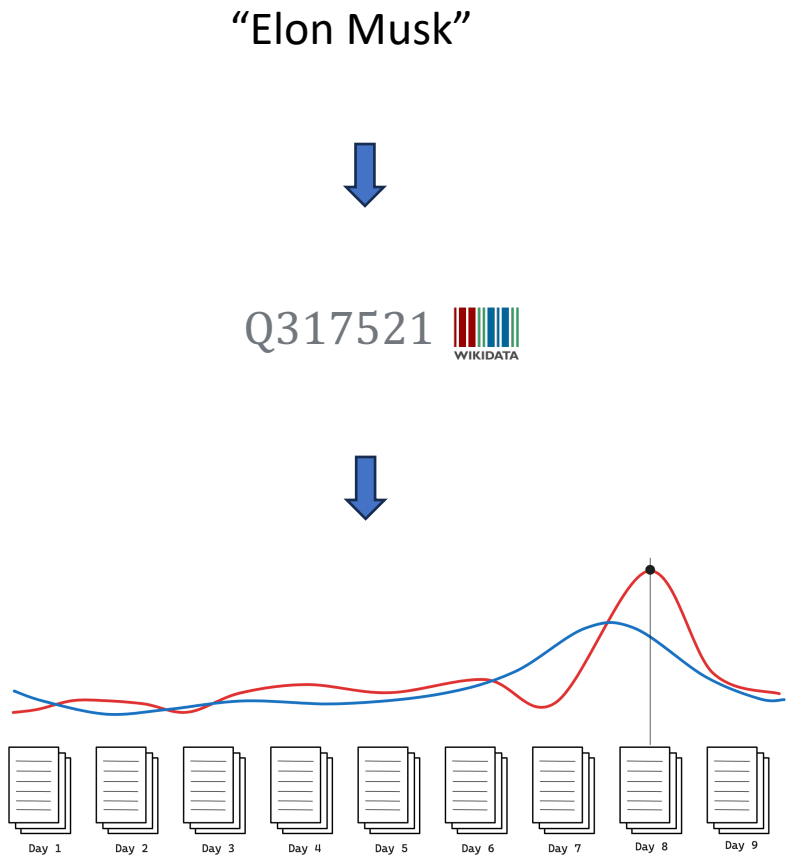
- Tasks for Social Media Text Classification: Abusive, Sentiment, Uncertainty
- 10 tasks, 200K social media posts
- Derived from Social Media IE Multi Task Benchmark – https://doi.org/10.5281/zenodo.5867160

Table 2: Performance of query strategies across datasets using around 10% training dataset.

task	dataset	round	N	N_{left}	$\%_{used}$	Full	Rand	E_{top}	E_{prop}	M_{top}	M_{prop}
Test Dataset											
ABUSIVE	Founta	42	41,861	37,661	0.10	0.79	0.77	0.78	0.78	0.79	0.77
	WaseemSRW	14	13,072	11,672	0.11	0.82	0.79	0.78	0.77	0.78	0.76
SENTIMENT	Airline	9	8,725	7,825	0.10	0.82	0.76	0.78	0.79	0.77	0.77
	Clarin	45	44,299	39,799	0.10	0.66	0.63	0.61	0.62	0.63	0.63
	GOP	8	7,121	6,321	0.11	0.67	0.63	0.64	0.63	0.62	0.64
	Healthcare	1	590	490	0.17	0.59	0.64	0.60	0.61	0.60	0.60
	Obama	2	1,777	1,577	0.11	0.63	0.56	0.60	0.58	0.59	0.57
	SemEval	13	12,145	10,845	0.11	0.65	0.59	0.60	0.61	0.58	0.61
UNCERTAINITY	Riloff	2	1,201	1,001	0.17	0.78	0.77	0.76	0.77	0.76	0.79
	Swamy	1	555	455	0.18	0.39	0.39	0.40	0.39	0.34	0.31
Remaining Dataset											
ABUSIVE	Founta	42	41,861	37,661	0.10	NaN	0.77	0.80	0.78	0.81	0.78
	WaseemSRW	14	13,072	11,672	0.11	NaN	0.78	0.79	0.77	0.80	0.76
SENTIMENT	Airline	9	8,725	7,825	0.10	NaN	0.75	0.79	0.79	0.80	0.78
	Clarin	45	44,299	39,799	0.10	NaN	0.62	0.62	0.62	0.64	0.63
	GOP	8	7,121	6,321	0.11	NaN	0.62	0.64	0.62	0.63	0.63
	Healthcare	1	590	490	0.17	NaN	0.53	0.56	0.53	0.47	0.50
	Obama	2	1,777	1,577	0.11	NaN	0.54	0.56	0.57	0.56	0.56
	SemEval	13	12,145	10,845	0.11	NaN	0.61	0.62	0.62	0.63	0.62
UNCERTAINITY	Riloff	2	1,201	1,001	0.17	NaN	0.80	0.82	0.84	0.82	0.81
	Swamy	1	555	455	0.18	NaN	0.37	0.40	0.40	0.33	0.36

`news_signals:`
An NLP Library for Text and Time Series

- NLP + time series is rare!
- **`news_signals:`** Python library for creating datasets with text inputs and time series outputs
- *Signals* are centered around real-world entities
- Collecting text and time-series from 3rd party sources about an entity
  - News articles
  - Time series of news volume
  - Time series of Wikipedia article page views
  - …
- Exploration tools: pandas-like interface, plots
- Dataset enrichment tools: anomaly detection, summarization
- Large-scale dataset generation; Docker container, K8 config



USING CAPTUM TO EXPLAIN GENERATIVE LANGUAGE MODELS

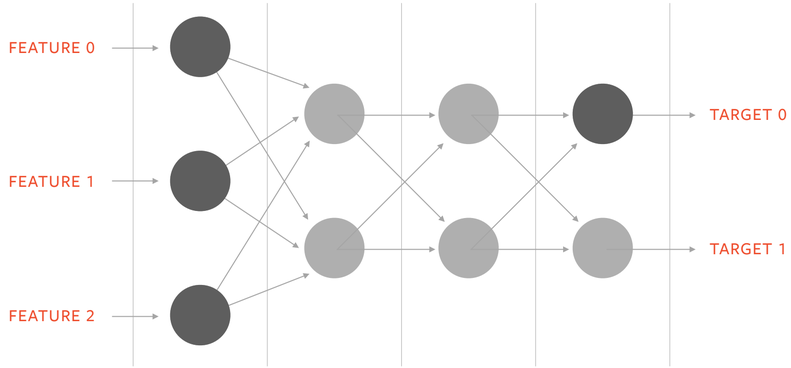
CAPTUM

A unified & generic model **interpretability** library

ATTRIBUTION

Quantify inputs' impact on the output

- e.g., Shapley Value, Integrated Gradients, LIME...



LANGUAGE MODEL ATTRIBUTION

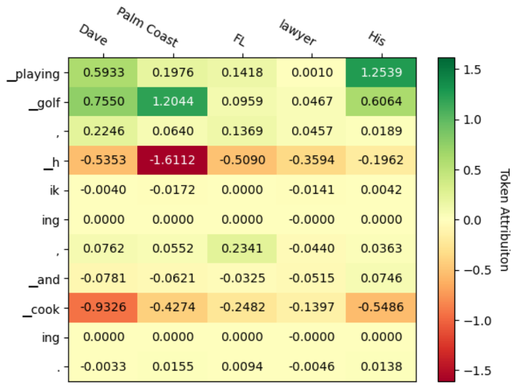
Allow users to

- define the input features in text
- attribute w.r.t the sequential output

MODEL ASSOCIATION

Prompt: Dave is a lawyer living in ***Palm Coast***, FL. His interests include ...

LLM: playing golf , hiking , and cooking



FEW-SHOT LEARNING

Prompt: Examples of movie review classification:

"Movie was ok, the actors weren't great" -> Negative	-0.0413
"Love it, it was an amazing story!" -> Positive	-0.2751
"Total waste of time!!" -> Negative	-0.0399

Classify the following review:

"I really liked the Avengers , it had a captivating plot!"

LLM: Positive

Vivek Miglani, Aobo Yang, Aram H. Markosyan, Diego Garcia-Olano, Narine Kokhlikyan
Meta AI

nerblackbox:
A High-level Library for Named Entity Recognition in Python

AI SWEDEN

Specify a **dataset** and a **model** -
nerblackbox takes care of the rest!

2 lines of code for each step!

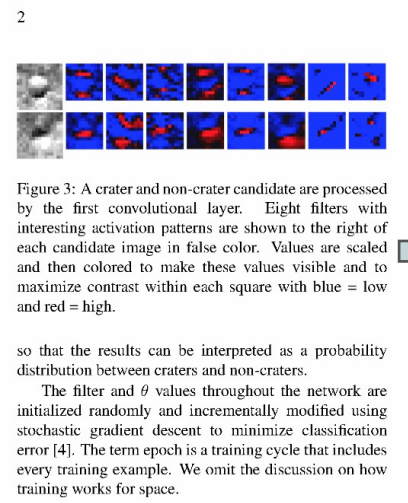
nerblackbox
flxst.github.io

LATEX RAINBOW:
OPEN-SOURCE LATEX TO PDF DOCUMENT SEMANTIC & LAYOUT ANNOTATION FRAMEWORK

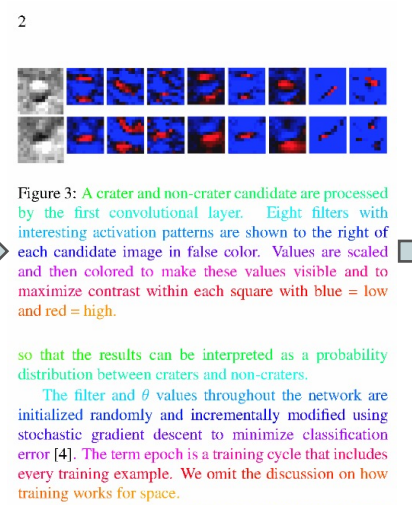
TECHNISCHE UNIVERSITÄT DARMSTADT

The framework proposes a color-based alignment method

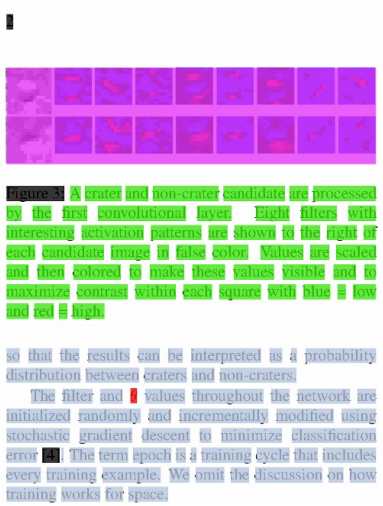
- Fetching semantic labels from LaTeX code
- Extracting element coordinates from PDF



1. Annotate LaTeX code



2. Compile rainbow PDF



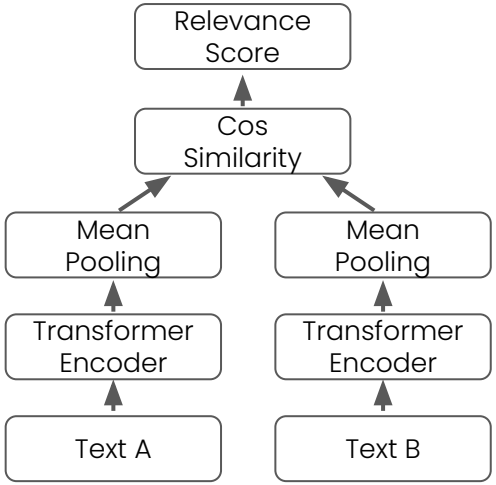
3. Align LaTeX & PDF

Annotations	Description
reading_order	Sequence order
label	Semantic Label (e.g., Caption, Footnote, Paragraph)
block_id	Number of block
section_id	Number of section
token	Text element extracted from PDF (e.g., θ)
latex	Corresponding LaTeX code (e.g., θ)
page_number	Number of page in PDF
Coordinates (x0, y0, x1, y1)	Element's position on PDF page

4. Extract annotations

Changxu Duan, Sabine Bartsch

Jina Embeddings: A Novel Set of High-Performance Sentence Embedding Models



Embedding Model Architecture

How to improve models in handling negated statements?

Should be similar

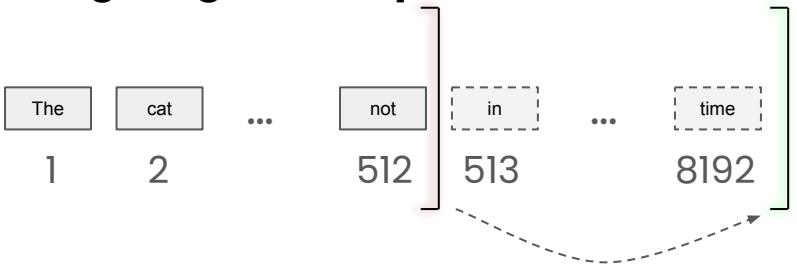
Statement: "Some dogs are running on a deserted beach."

Entailment: "There are multiple dogs present."

Contradiction: "There are no dogs present."

Should **not** be similar

How to train embedding models for encoding long text sequences?



Improving NER Research Workflows with SeqScore

Constantine Lignos, Maya Kruse, and Andrew Rueda

https://github.com/bltlab/seqscore

- **SeqScore** is a robust one-stop command line tool for working with NER data
- **Validation**: Validates label encoding (BIO, etc.) and offers options for automatic repair
- **Summarization**: Provides an overview of the types and entities in datasets
- **Conversion**: Converts between different label encodings (BIO to BIOES, etc.)
- **Scoring**: Scores system output with configurable handling of invalid label transitions
- **Error analysis**: Provides reports on errors to enable analysis

Brandeis

```
pip install seqscore
```

zilliz

GPT Cache

Key challenges in building LLM-based applications

1. Spiking costs with unnecessary API calls for semantically identical questions that the LLM has already answered, which will waste your money and resources.
2. Poor performance and scalability with high response latency. Additionally, LLM services enforce rate limits, restricting the number of API calls your applications can make to the server within a given timeframe.

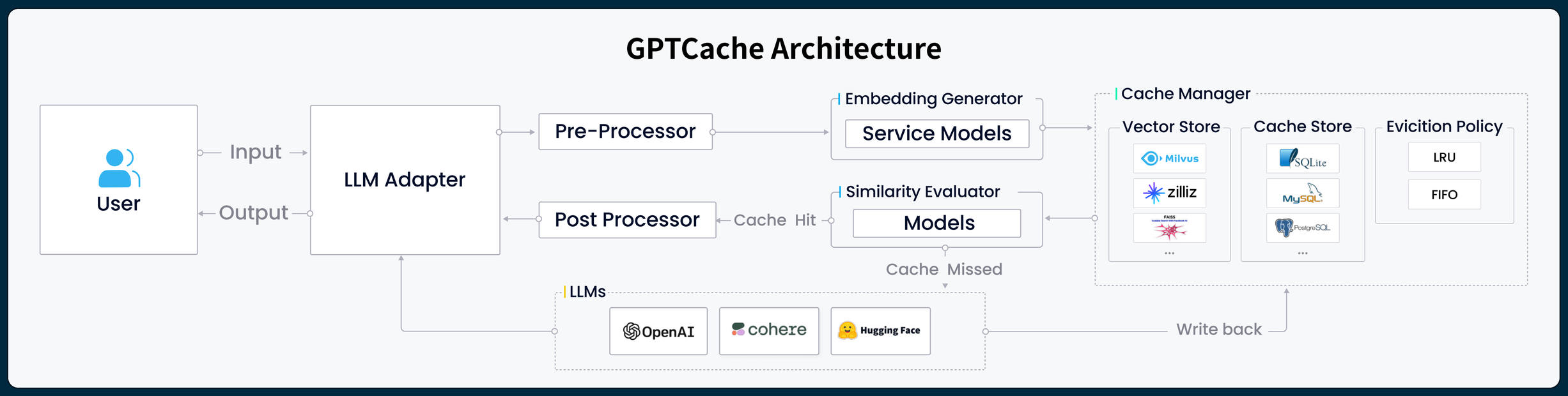
What is GPTCache?

GPTCache is an open-source semantic cache designed to improve the efficiency and speed of GPT-based applications by storing and retrieving the responses generated by language models. GPTCache allows users to customize the cache to their specific requirements, offering a range of choices for embedding, similarity assessment, storage location, and eviction policies. Furthermore, GPTCache supports both the OpenAI ChatGPT interface and the Langchain interface, with plans to support more interfaces in the coming months.

How does GPTCache work?

Simply put, GPTCache stores LLMs' responses in the cache. Therefore, when users make similar queries that LLMs had previously responded to, GPTCache searches and returns the results to the users without the need to call the LLM again. Unlike traditional cache systems such as Redis, GPTCache employs semantic caching, which stores and retrieves data through embeddings. It utilizes embedding algorithms to transform the user queries and LLMs' responses into embeddings and conducts similarity searches on these embeddings using a vector store such as Milvus.

GPTCache comprises six core modules: LLM Adapter, Pre-processor (Context Manager), Embedding Generator, Cache Manager, Similarity Evaluator, and Post-processor.



NOMIC

GPT4All: An Ecosystem of Open Source Compressed Language Models

Yuvanesh Anand, Zach Nussbaum, Adam Treat, Aaron Miller, Richard Guo,
Ben Schmidt, Planet Earth, Brandon Duderstadt*, Andriy Mulyar*

TL;DR

A technical report and case study of how an open source gpt-3.5-turbo clone became the 3rd fastest growing github repository of all time

55,000+
Github Stars

40,000+
Chat Client Monthly Active Users

66,000+
Python Package Downloads/Month

25,000+
GPT4All Discord Members

EDGAR-CRAWLER: Automating Financial Data Harvesting and Preprocessing for NLP

Lefteris Loukas | **Manos Fergadiotis** | **Ilias Stogiannidis** | **Prodromos Malakasiotis**

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

What's the problem?

Most **NLP datasets** are often behind paywalls. **EDGAR**, a notable free source, hosts comprehensive annual reports (10-K reports) from US publicly traded companies. However, these reports, stored as **complex PDF/HTML/TXT files** with numerous sections and pages, pose **challenges for researchers**. Extracting specific information becomes laborious, requiring downloading a vast number of reports for manual text extraction. This is an **impractical and time-consuming** process.

Our Solution:

EDGAR-CRAWLER, a free, open-source package that downloads and extracts information from annual reports (EDGAR 10-K documents) into an easy-to-manage JSON format.

Our software, EDGAR-CRAWLER, is made up of two modules:

```
1. python edgar_crawler.py
2. python extract_items.py
```

1. Responsible for crawling and downloading financial reports. Supports multiple input arguments.

2. Cleans and extracts the text of all or particular items from downloaded 10-K reports and saves them as JSON files.

Scientific Contributions in ML & NLP:

- Trusted by the community (**160+ stars** on Github!)
- **Multiple citations** in relevant literature.

Future Work:

Looking for contributors for these, send us a message if interested ....

- **Support more types of documents like quarterly reports.**
- **Create a GUI for more user-friendly configuration.**
- **Deploy a live demo to increase accessibility.**

edgar-crawler

Turn unstructured financial documents into clean JSON files.

DeepZensols

A deep learning NLP framework that:

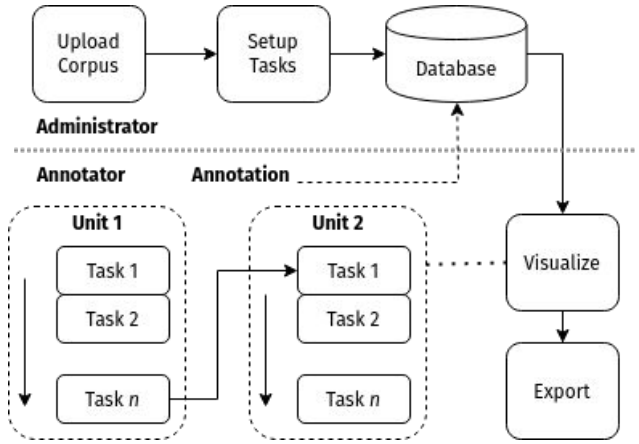
- Facilitates reproducing consistent results.
- Allows hot-swapping features and embeddings without further processing and re-vectorizing the dataset.
- Easily create, train and evaluate off-the-shelve models.
- Reproduces results, a issue that persists in the machine learning community.
- Includes easily configurable standard models allowing quick experimentation and without any coding.
- Thorough API and overview documentation with examples.

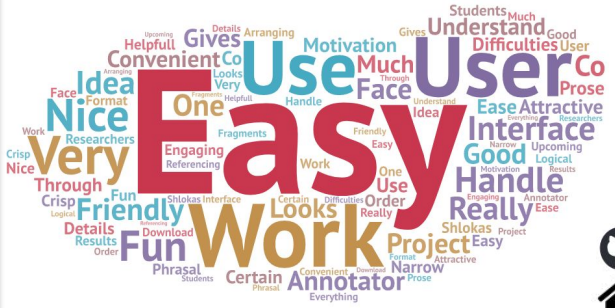
Antarlekhaka: A Comprehensive Tool for Multi-task Natural Language Annotation

~ 7000 Low-resource Language ⇒ Human annotation relevant!

- Ambiguous/Absent sentence boundaries
- Ordering, splitting, merging tokens
- Limited support in existing tools

- Corpora in poetry format
- Majority of Sanskrit literature in poetry





Eight Categories of NLP Tasks

1. **Sentence Boundary**
2. **Token Manipulation**
3. Token Text Annotation
4. Token Classification
5. Token Graph
6. Token Connection
7. Sentence Classification
8. Sentence Graph

- Sequential + Multi-task
- Pluggable heuristics
- Language agnostic
- Clone, Visualize, Export
- Administrative tasks

Hrishikesh Terdalkar
Arnab Bhattacharya

Antarlekhaka/code

hrishikeshrt.github.io