Building and Deploying Foundation Models

BIODS 271 / CS 277

Tanveer Syeda-Mahmood

Model development processes

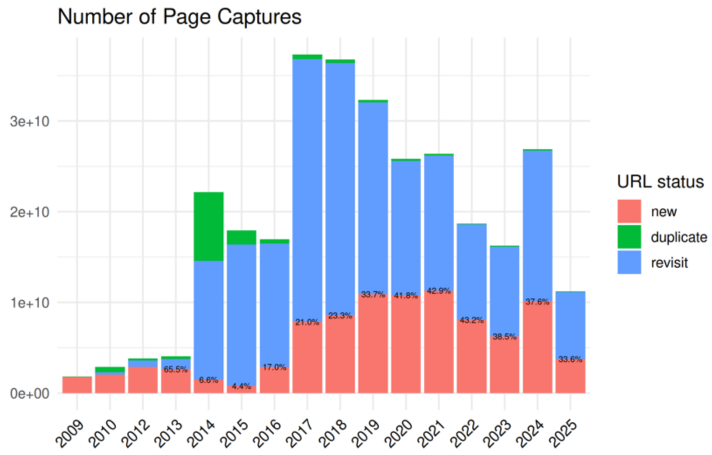
- Identify data sources → Obtain legal approvals → Records data in Lakehouse
- Cleanse data → Prepare Data → Design model architecture
- Schedule training resources → Train model over epochs → Select benchmarks
- Evaluate model on benchmarks → Prepare model cards and papers → Release in open source or proprietary platforms

Documentation & Governance

Data acquisition considerations

- How much data to acquire?
- Is there enough variety in the data?
- What is the inherent bias in the data sampling?
  - Under-sampling or over sampling
- Addressing ethical concerns
- Usually sourced from multiple collections
  - petabytes of data and trillions of tokens.
- Usually with the help of partners for enterprise data
  - E.g. NextData
- Labeling datasets is a discipline in itself

Common crawl dataset



Types of datasets used to train FMs

- Raw text data, image data, video data, domain-specific data
- Data from pdfs – multimodal data
- Question-answer pairs on chunks derived from
  - Text, Images, Videos
  - Domain-specific content
- Manual, semi-automatic to fully-automatic ground truth labeling
  - For medical imaging from companion reports

Datasets used for LLMs available in model cards

Dataset	Description
Common Crawl	Open repository of web crawl data.
Webhose	Unstructured web content converted into machine-readable data feeds acquired by IBM.
arXiv	Over 1.8 million scientific paper pre-prints posted to arXiv.
Wikimedia	Eight English Wikimedia projects (enwiki, enwikibooks, enwikinews, enwikiquote, enwikisource, enwikiversity, enwikivoyage, enwiktionary) containing extracted plain text from pages and articles.
OpenWeb Text	Open-source version of OpenAI's Web Text corpus containing web pages through 2019.
Stack Exchange	Anonymized set of all user-contributed content on the Stack Exchange network, a popular collection of websites centered around user-contributed questions and answers.
Hacker News	News on computer science and entrepreneurship, taken between 2007-2018.
Project Gutenberg PG19	A repository of free e-books with focus on older works for which U.S. copyright has expired.
GitHub Clean	Code data from CodeParrot covering a variety of coding languages.
Pubmed Central	Biomedical and life sciences papers.
Free Law	Public-domain legal opinions from US federal and state courts.
SEC Filings	10-K/Q filings from the US Securities and Exchange Commission (SEC) for the years 1934-2022.

Patents	US patents granted from 1975 to May 2023, excluding design patents.
DeepMind Mathematics	Mathematical question and answer pairs data.
Earning Calls Transcript	Transcripts from the quarterly earnings calls that companies hold with investors. The dataset reports a collection of earnings call transcripts, the related stock prices, and the sector index.
EDGAR	This corpus comprises of annual reports from all the publicly traded companies in the US spanning a period of more than 25 years.
FDIC	The data is from the annual submissions of the FDIC.
Finance Textbooks	This corpus is from Open Textbook Library which is UMN's free textbook library, and this dataset includes the dump of all textbooks tagged as finance.
Financial Research Papers	Publicly available financial research paper corpus.
IBM Documentation	IBM redbooks and product documents.

GPT-3:

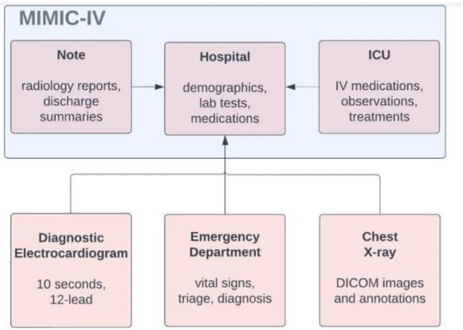
(1) a version of the CommonCrawl dataset, filtered based on similarity to high-quality reference corpora, (2) an expanded version of the Webtext dataset, (3) two internet-based book corpora, and (4) English-language Wikipedia.

Datasets for VLMs

- Paired Image-text
  - LAION-5b dataset
  - MS-COCO
  - Flickr30k
- Diversity in visual concepts, languages, and contexts, which requires datasets covering multiple domains (e.g., nature, urban environments), languages beyond English, and varied lighting or object configurations.
  - CAULDRON
    - 50 datasets
- Structured and grounded datasets:
  - bounding boxes in COCO or Flickr30K enable models to localize objects within images
- Healthcare datasets need special considerations for data assembly

VQAv2	82,772	443,757	1,595,929
COCO-QA	46,287	78,736	286,982
Visual7W	14,366	69,817	279,268
A-OKVQA	16,539	17,056	236,492
TallyQA	98,680	183,986	738,254
OK-VQA	8,998	9,009	38,853
HatefulMemes	8,500	8,500	25,500
VQA-RAD	313	1,793	8,418
Captioning			
LNarratives	507,444	507,444	21,328,731
Screen2Words	15,730	15,743	143,103

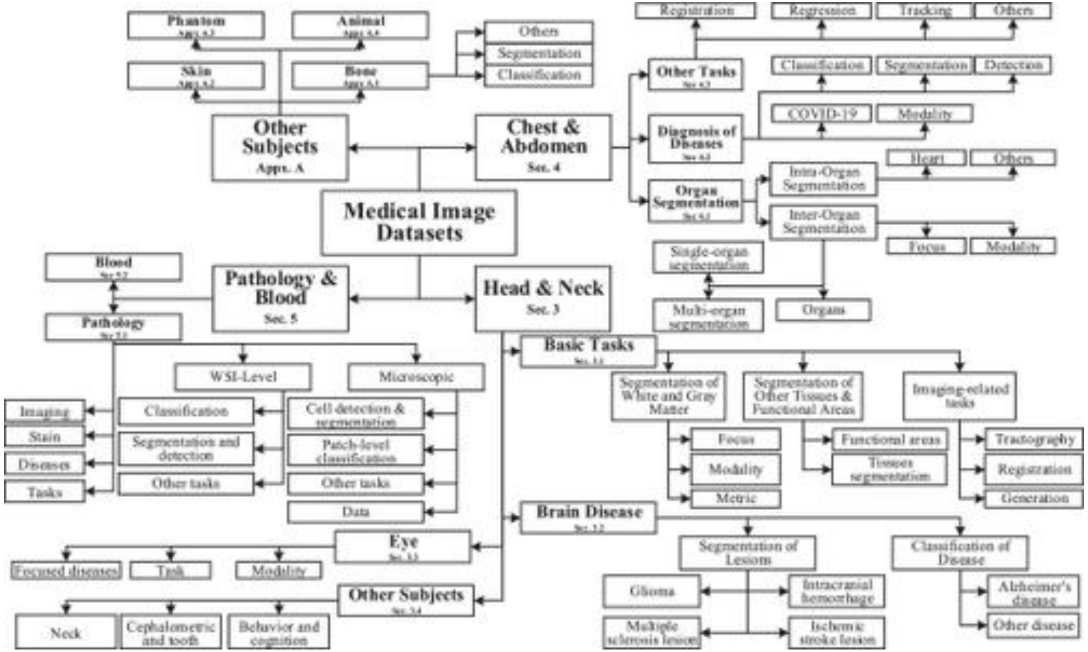
https://huggingface.co/datasets/HuggingFaceM4/the_cauldron



https://www.nature.com/articles/s41597-022-01899-x

Datasets for training healthcare FM

- Access to large collections still an issue
- Popular datasets
  - MIMIC (60k patients, 400,000+images), Chexpert (64k+patients, 200k images+reports), PadChest, NIH-8 (30k patients, 100,000+images), ChestImagenome
  - TCIA collections
  - MedPix (12,000 patients, 9000 topics, 59K images)
  - MURA (14,000+ studies, 40k+ images)
  - OpenNeuro datasets (1200+ datasets, 51K patients)
  - https://github.com/sfikas/medical-imaging-datasets
  - https://dl.acm.org/doi/10.1145/3615862



https://dl.acm.org/doi/10.1145/3615862

Considerations for legal approvals

- Legal and Licensing issues
  - Who owns the data
  - Volume and Variety: Ensure the dataset is large and diverse enough to train robust models.
  - Specific Use: Clarify whether the data can be used for commercial purposes, research, or both.
  - Exclusivity: Determine if the rights are exclusive or non-exclusive.
- Sensitivity and bias in the data
  - Representational bias (under or over representation) leads to homogenization
  - Abuse (chat bots use toxic content)
- Business relevance
  - Training data not aligned with business values.

Recording model development in a Lakehouse

- Design of schema to record all details associated with the development of the models.
- Schema covers:
  - Data provenance (source, date acquired, etc.)
  - Approval chains and data clearances
  - Parameter variations for the various runs (e.g. context length)
  - Model training logs (last epoch date, etc.) and run status
  - Model checkpoint details, base model details
  - Datasets generated from the model
  - Intended use
  - Model family hierarchy details

Namespace	Model ID	Model Label	Base Model	Model Type	Size	Revision	Variant	Product Name	Access

Data cleansing

- Protecting privacy & data leakage
- Removing objectionable content
  - Hate and profanity, PII removal
  - Toxicity & biases in the content
  - Stereotyping
- Copyright infringement
- Confidential/sensitive data
- Many of the cleansing operations use AI models underneath

Deduplication	HAP filter	People detector	HIPAA field removal
Data quality filtering	Logo removal	Face Blur	PHI removal
License filtering	Confidential data removal	Children removal	DICOM cleaner

Cleansing operations examples

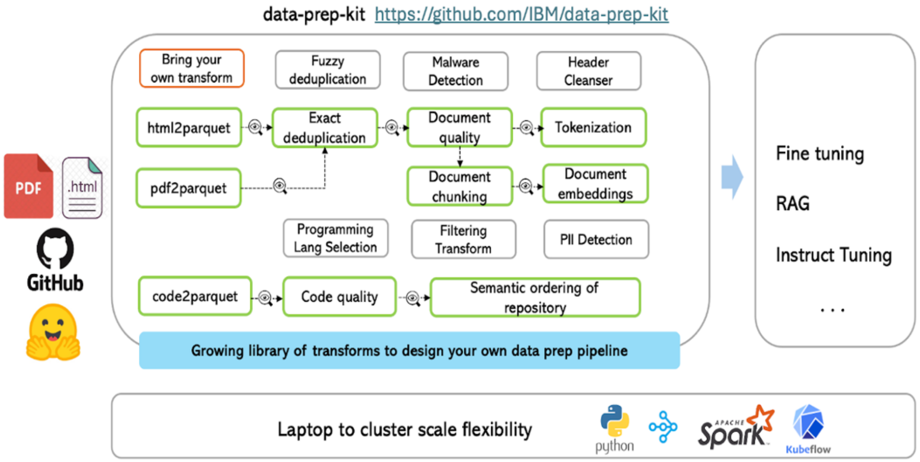
yolov8m-seg.pt people detection

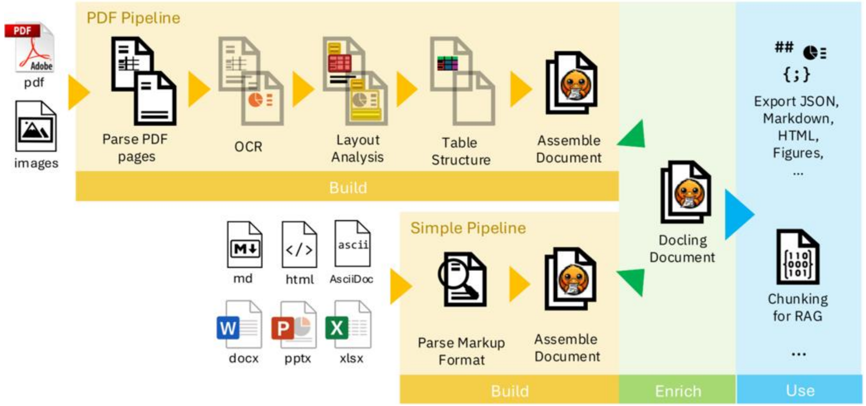
Data Cleaning: Before vs. After

Method	Before	After
Deduplication (Sentence-Level, Document Level)	We offer a variety of services. Our services include web design, SEO, and social media management. Our services include web design, SEO, and social media management.	We offer a variety of services. Our services include web design, SEO, and social media management.
Quality Filters (Language, Keyword, Statistic)	This document contains important information. この文書には重要な情報が含まれています。 중요한 정보가 포함되어 있습니다.	This document contains important information.
Content Filters (Toxic, Bias)	I can't believe how stupid this idea is. Only an idiot would think this is good.	I have concerns about this idea. It might be worth exploring other options.
Privacy Reduction (Personality Identifiable Information)	John Doe's phone number is 123-456-7890, and he lives at 1234 Elm Street, Springfield.	[Name]'s phone number is [redacted], and he lives at [redacted address].
Rule-based Cleansing	This is an exmaple text!! with some TYPOs and unnecessary punctuations,, and spaces .	This is an example text with some typos and unnecessary punctuations and spaces.

Data preparation

- Need AI models underneath:
  - Document shredding
  - Chunking
  - Text extraction
- Selecting relevant chunks
- Token generation
  - Word-piece tokenization
- Image-text association
  - Caption-image association
- Ground truth labeling for QA pair generation
- Platforms for large-scale parallel processing
  - E.g. Ray parallelism, Spark, etc.

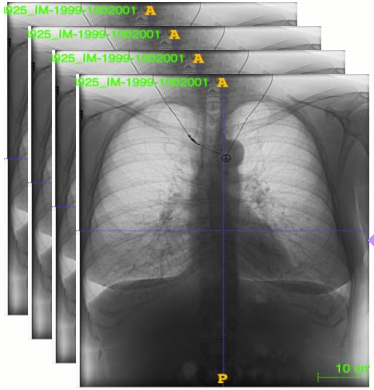


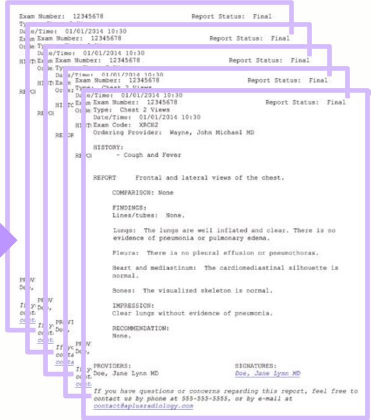


Self-supervised labeling approaches

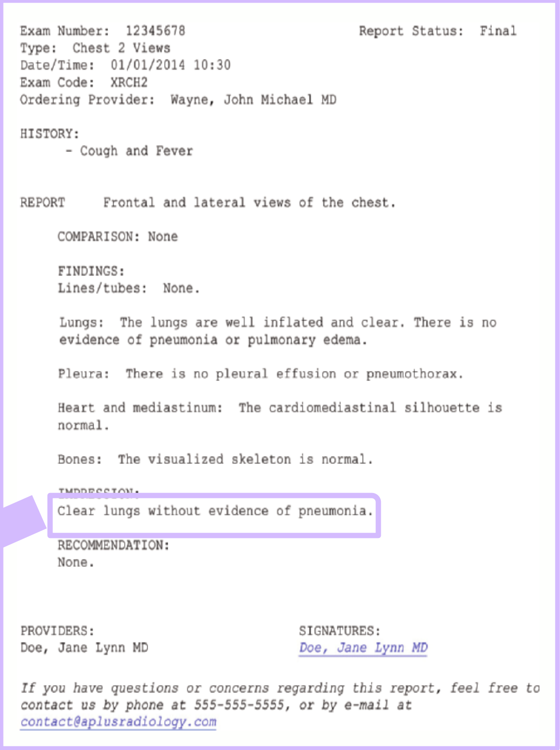
- Using companions reports to label images
- Using LLM to summarize the data
- Using LLM to extract QA
- Manual oversight for ground truth generation

Labeling images from reports





Associated reports



Exam Number: 12345678 Report Status: Final
Type: Chest 2 Views
Date/Time: 01/01/2014 10:30
Exam Code: XRCH2
Ordering Provider: Wayne, John Michael MD

HISTORY:
- Cough and Fever

REPORT Frontal and lateral views of the chest.

COMPARISON: None

FINDINGS:
Lines/tubes: None.

Lungs: The lungs are well inflated and clear. There is no evidence of pneumonia or pulmonary edema.

Pleura: There is no pleural effusion or pneumothorax.

Heart and mediastinum: The cardiomediastinal silhouette is normal.

Bones: The visualized skeleton is normal.

IMPRESSION:
Clear lungs without evidence of pneumonia.

RECOMMENDATION:
None.

PROVIDERS: Doe, Jane Lynn MD
SIGNATURES: Doe, Jane Lynn MD

If you have questions or concerns regarding this report, feel free to contact us by phone at 555-555-5555, or by e-mail at contact@aplusradiology.com

<u>Clear</u> <u>lungs</u> without evidence of <u>pneumonia.</u>

Anatomical finding

Anatomy

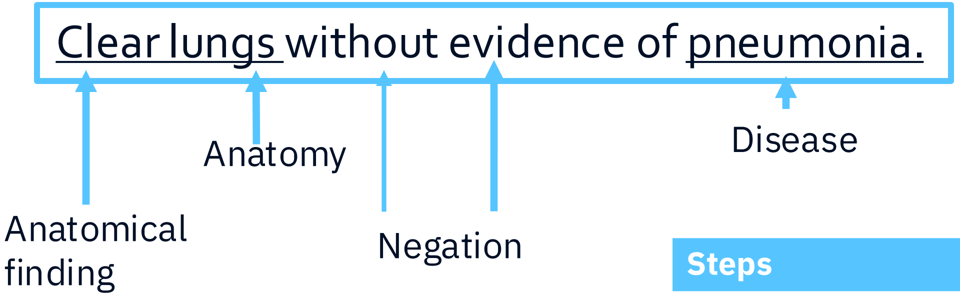
Negation

Disease

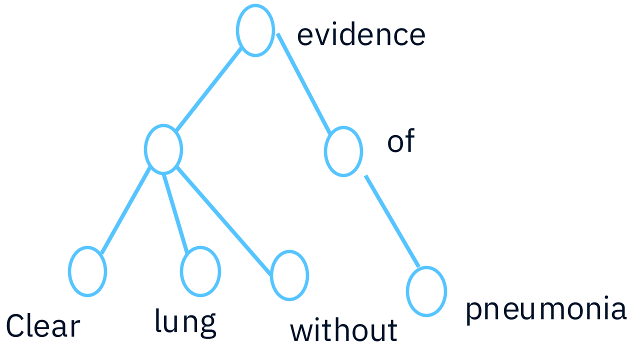
Fine-grain modifiers:

- Anatomy affected, Sub-anatomy, Location, Laterality, Severity, Size, Shape, Character, Correlation, Cause, Symptom, Hedge

Label extraction from reports



Dependency parse tree



ESG Parser Output

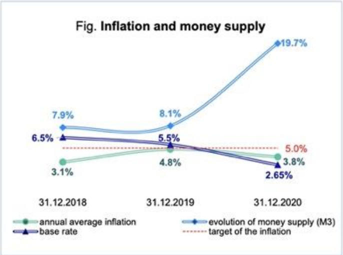
```
-------------------------------------
  .- nadj       clear1(1,2,u)     adj e
.-+- subj(n)    lung1(2,u)        noun
| `- nadjp      without2(3,u)     adv r
o--- top        evidence2(4,2,u)  verb
`--- vprep      of1(5,4,6)        prep
  `- objprep(n) pneumonia1(6,u,u) noun
-------------------------------------
```

Steps	Action
Initial groups given by dependency parser	[(1,2,u)]-> clear lung -> (core group)
Phrasal grouping using connected component analysis	[(2,u)(4,2,u)(5,4,6)(6,u,u))]-> lung evidence of pneumonia -> (core group)
Negation detection	[(3,u)] -> without -> (negation span, helper group)
Assembled FFL patterns	**anatomical finding|no|clear lung|lung|||clear disease|no|pneumonia|lung|**

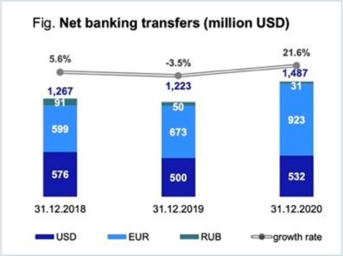
FFL pattern F=< T|N|C|M* >

Natural language analysis of the sentence using ESG parser ("Deep parsing in Watson", M.Mc Cord, J.W. Murdock, B.K. Boguraev, IBM Journal of Research and Development (Volume: 56 , Issue: 3.4 , May-June 2012)

Data Prep example- Automatic QA



me of remittances from abroad to individuals, on a net b
to 2019 and amounted to 1,487 million US dollars, rec
2015 to present.



- Prompt for Summary generation: Please describe this image
- Prompt for QA generation:
  - Here are the rules for question and answer generation. 1) The question should not be a multiple choice question and answer. 2) The answers should be in a single paragraph (no bullet points). 3) The questions should be tagged as Question: and the answers should be tagged as Answer: 4) Do not generate any other text before and after the questions and answers. 5) If you are unable to generate question and answers your response should be - Unable to generate questions and answers. 6) Do not repeat the same question. Using these rules, generate 5 questions and answers based on the following

Generated Summary

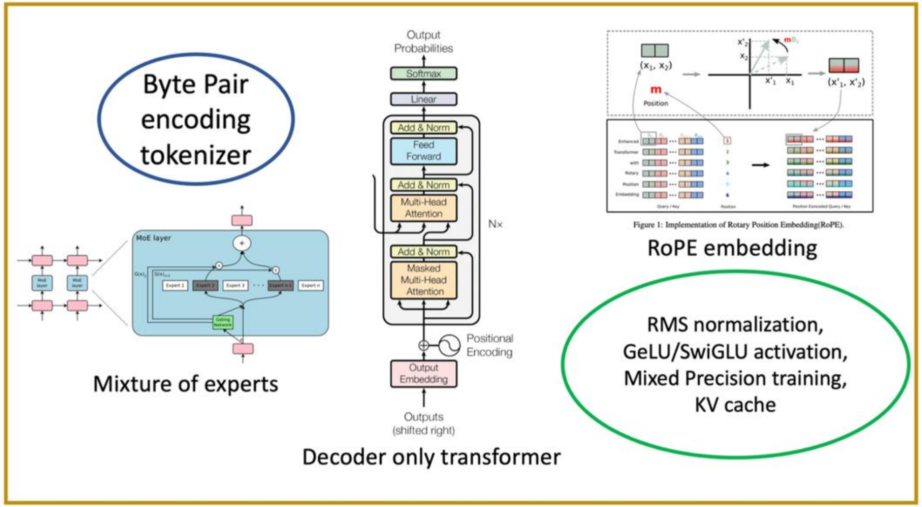
```
Generated text: The image is a composite of two graphs, each representing different economic indicators. The first graph is titled "Fig. In
flation and money supply," and the second graph is titled "Fig. Net banking transfers (million USD)."

### First Graph: "Fig. Inflation and money supply"
- **Title**: "Fig. Inflation and money supply"
- **Axes**:
  - **X-axis**: Labeled as "31.12.2018," "31.12.2019," and "31.12.2020," indicating the time series data points.
  - **Y-axis**: Labeled as "Inflation rate (%)" and "Money supply (M3)," indicating the percentage of inflation and the money supply in mil
lion M3 (Million Three).
- **Data Points**:
  - **Inflation Rate**:
    - 31.12.2018: 6.5%
    - 31.12.2019: 8.1%
    - 31.12.2020: 19.7%
  - **Money Supply**:
    - 31.12.2018: 5.0%
    - 31.12.2019: 5.5%
    - 31.12.2020: 3.8%
- **Trends**:
  - The inflation rate shows a general upward trend from 6.5% in 2018 to 19.7% in 2020.
  - The money supply also shows an upward trend from 5.0% in 2018 to 3.8% in 2020
```

Generated QA

```
'Question: What is the trend of inflation rate and money supply in the given time series data points?\nAnswer: The inflation rate shows a g
eneral upward trend from 6.5% in 2018 to 19.7% in 2020. The money supply also shows an upward trend from 5.0% in 2018 to 3.8% in 2020.\n\nQ
uestion: What is the difference between the inflation rate and money supply in the given time series data points?\nAnswer: The difference b
etween the inflation rate and money supply in the given time series data points is that the inflation rate is the percentage of inflation,
while the money supply is the amount of money in circulation.\n\nQuestion: What is the highest inflation rate and money supply in the given
time series data points?\nAnswer: The highest inflation rate in the given time series data points is 19.7% in 2020, and the highest money s
upply is 5.5% in 2019.\n\nQuestion: What is the lowest inflation rate and money supply in the given time series data points?\nAnswer: The l
owest inflation rate in the given time series data points is 6.5% in 2018, and the lowest money supply is 3.8% in 2020.\n\nQuestion: What i
s the difference between the inflation rate and money supply in the given time series data points?\nAnswer: The difference between the infl
ation rate and money supply in the given time series data points is that the inflation rate is the percentage of inflation, while the money
supply is the amount of money in circulation.'
```

Designing model architecture

- Basic architectures
  - Transformers
  - Llava
  - Newer (more details in later lectures)
    - State-space
    - Mamba
    - Bamba
    - RAG for training
    - Tree of thought

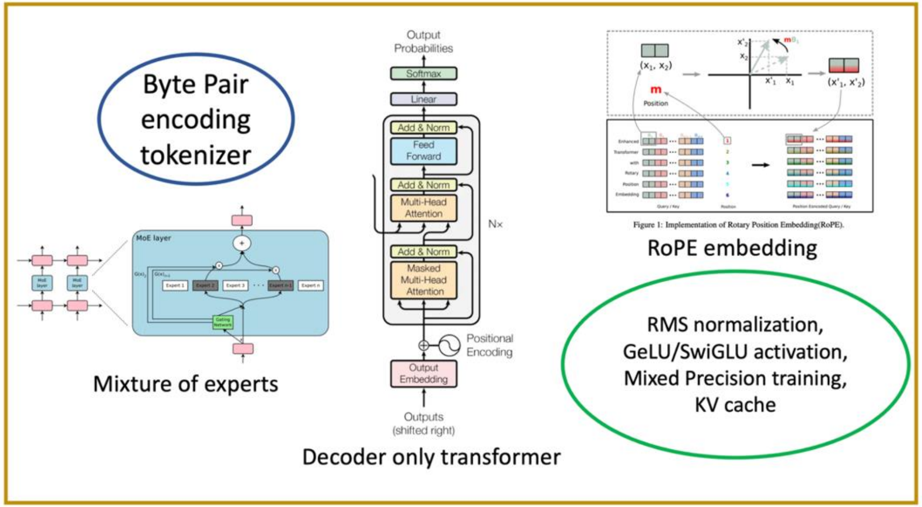


A generic LLM architecture recipe

https://www.linkedin.com/pulse/llm-end-to-end-resources-part-1-model-architecture-vivek-madan-1a0ic/

Designing model architecture

- Byte pair encoding
  - E.g. aaabdaaabac
  - Compressed: XdXac
    - X=ZY Y=ab Z=aa

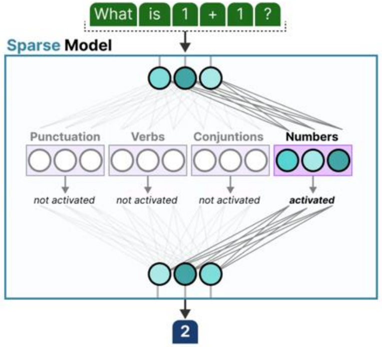


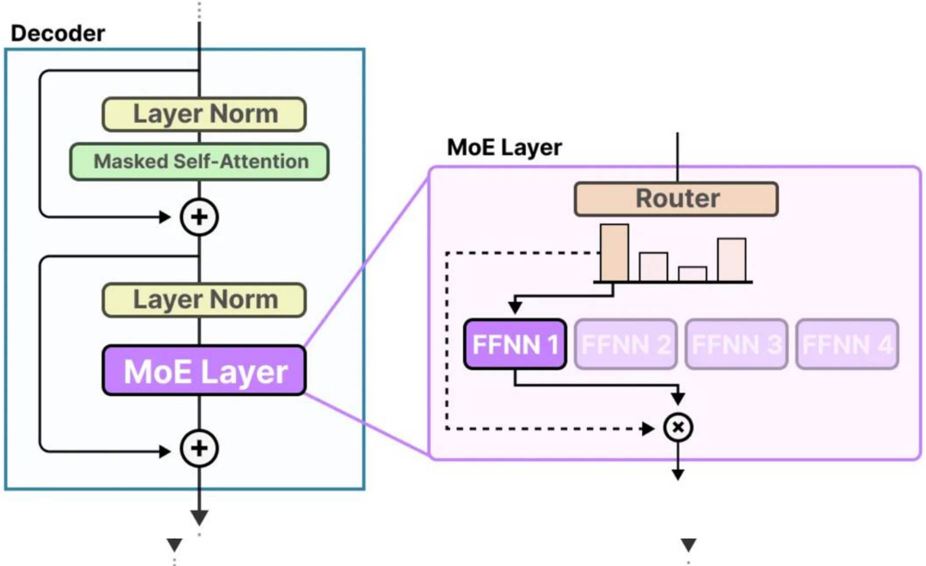
A generic LLM architecture recipe

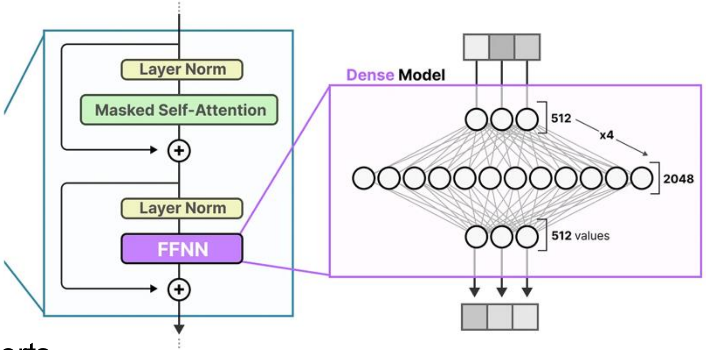
https://www.linkedin.com/pulse/llm-end-to-end-resources-part-1-model-architecture-vivek-madan-1a0ic/

Designing model architecture

- Mixture of experts models







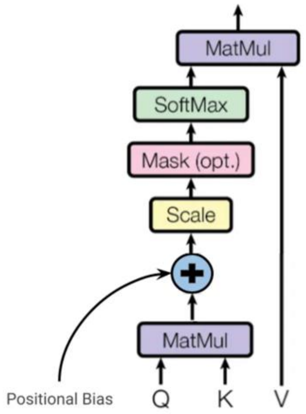
https://newsletter.maartengrootendorst.com/p/a-visual-guide-to-mixture-of-experts

Fedus, William, Barret Zoph, and Noam Shazeer. "Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity." *Journal of Machine Learning Research* 23.120 (2022): 1-39

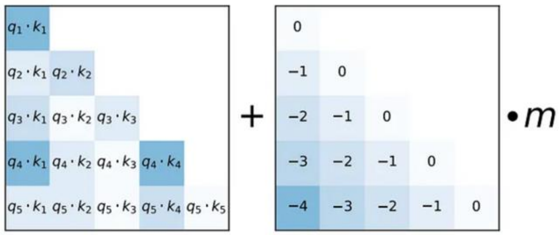
Load balancing to prevent bias to one expert

Designing model architecture

- Changing positional embeddings
  - From fixed sinusoidal to relative embeddings
  - RoPE (rotational position embeddings) (used by Llama, Llama2,..)
    - RoPE rotates the embedding vector of each token based on its position in the sequence. The rotation angle is proportional to the token's position.
      - two tokens with the same relative distance will have the same rotation angle, regardless of their absolute position in the sentence.
      - Allows to handle longer context length
    - Captures **relative distance between the tokens encoded**
  - ALiBi (Attention with linear biases) (used by BLOOM)
    - Much simpler – adds a constant bias term to the attention computation



Scaled dot product adaptation in ALiBi.

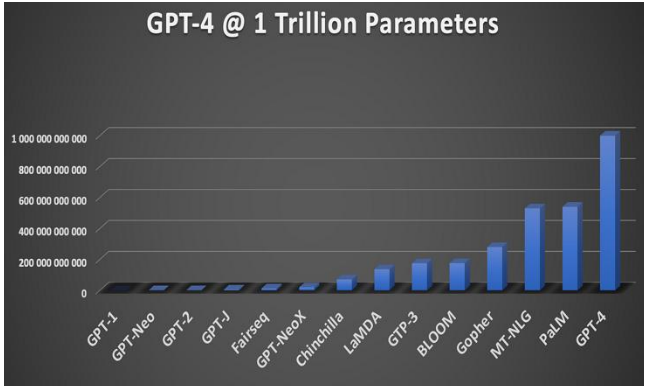


Linear biased attention in ALiBi.

Training Resources/Costs for Foundational Models

https://neptune.ai/blog/nlp-models-infrastructure-cost-optimization

- High computational requirements
- **Smaller Models (7B and below):** A single GPU with 16GB VRAM (like an RTX 4080) might suffice.
- **Larger Models (13B+):** Consider GPUs with 24GB+ VRAM (like NVIDIA A100, H100, or RTX 4090).
- **Extremely Large Models (175B+):** Thousands of GPUs are typically required, such as those used for training GPT-3.
- Storage capacity:
  - Multiple copies of the whole model in a single storage device is difficult
  - Distributed inference is needed
  - **OpenAI's GPT-3** model, with 175B parameters, requires over **300GB** of storage for its parameters
- Bandwidth requirements pos problems
- Energy consumption can be huge
- Cost
  - running cost of the chatGPT is around **$100,000** per day or **$3M** per month.



Environmental and financial costs

Consumption	CO_2e (lbs)
Air travel, 1 passenger, NY↔SF	1984
Human life, avg, 1 year	11,023
American life, avg, 1 year	36,156
Car, avg incl. fuel, 1 lifetime	126,000
Training one model (GPU)	
NLP pipeline (parsing, SRL)	39
w/ tuning & experimentation	78,468
Transformer (big)	192
w/ neural architecture search	626,155

Table 1: Estimated CO_2 emissions from training common NLP models, compared to familiar consumption.[1]

		Estimated cost (USD)	
Models	Hours	Cloud compute	Electricity
1	120	$52–$175	$5
24	2880	$1238–$4205	$118
4789	239,942	$103k–$350k	$9870

Table 4: Estimated cost in terms of cloud compute and electricity for training: (1) a single model (2) a single tune and (3) all models trained during R&D.

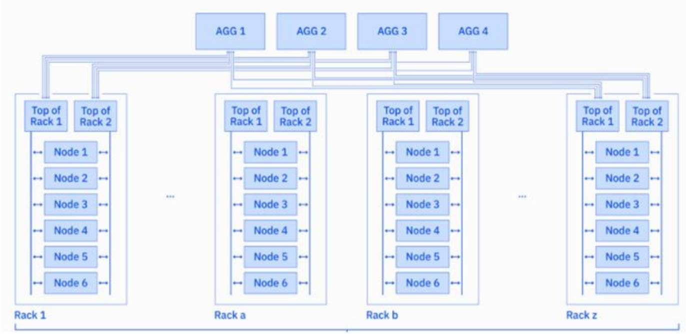
Energy and Policy Considerations for Deep Learning in NLP. Strubell et al. ACL 2019. https://arxiv.org/abs/1906.02243
On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? 🦜. Bender et al. FAccT 2021. https://dl.acm.org/doi/10.1145/3442188.3445922

Training infrastructures

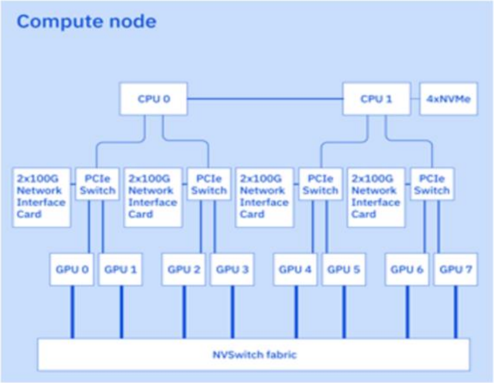
- See report for details:
- Vela architecture
- https://arxiv.org/pdf/2407.05467

Compute node

- Dual 48-core 4th Gen Intel Xeon Scalable Processors
- 2TB of RAM
- 8 NVIDIA H100 GPUs with 80GB High Bandwidth Memory (HBM)
- 10 NVIDIA ConnectX-7 NDR 400 gigabits per second (Gb/s) InfiniBand Host Channel Adapters (HCA)
  - 8 dedicated to compute fabric
  - 2 dedicated to storage fabric
- 8 3.4TB Enterprise NVMe U.2 Gen4
- Dual 25G Ethernet Host links
- 1G Management Ethernet Port



(a) Overall system view



(b) Compute node view

Deploying model architecture – Inference costs

- KV Cache is the dominant factor

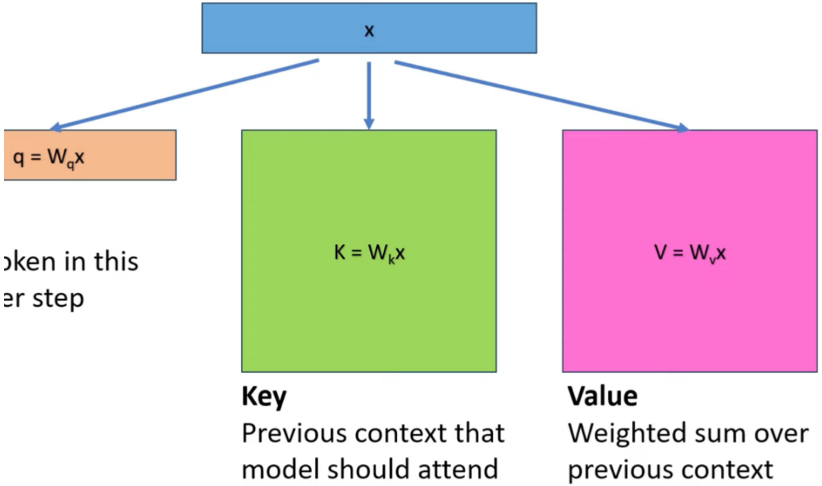
GPT-4 API Pricing

With broad general knowledge and domain expertise, GPT-4 can follow complex instructions in natural language and solve difficult problems with accuracy.

Learn about GPT-4

Model	Input	Output
8K context	$0.03 / 1K tokens	$0.06 / 1K tokens
32K context	$0.06 / 1K tokens	$0.12 / 1K tokens

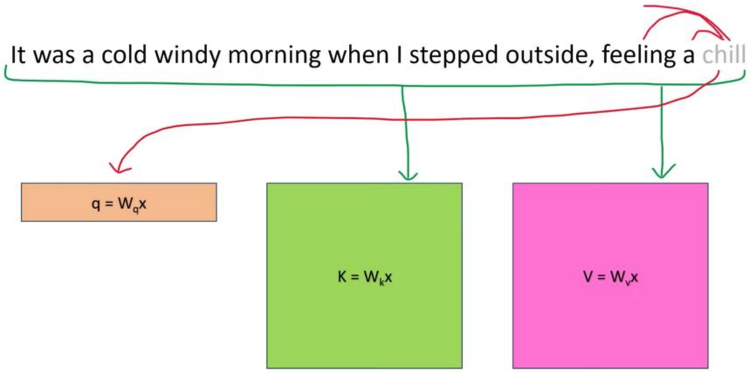
$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

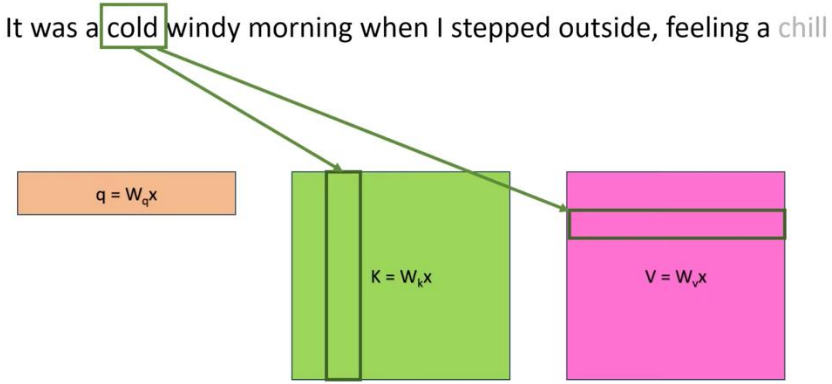


https://www.youtube.com/watch?v=80bIUggRJf4

Deploying model architecture

- KV Cache is the dominant factor

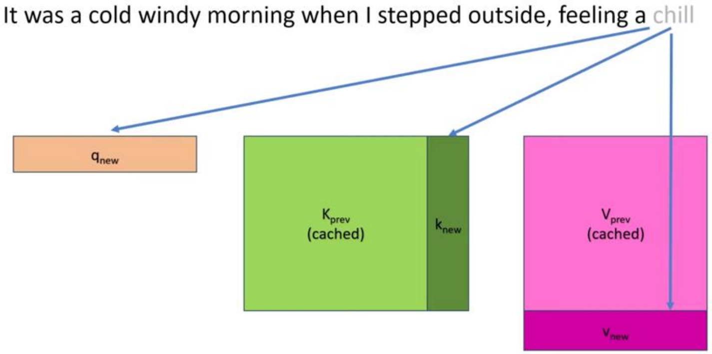


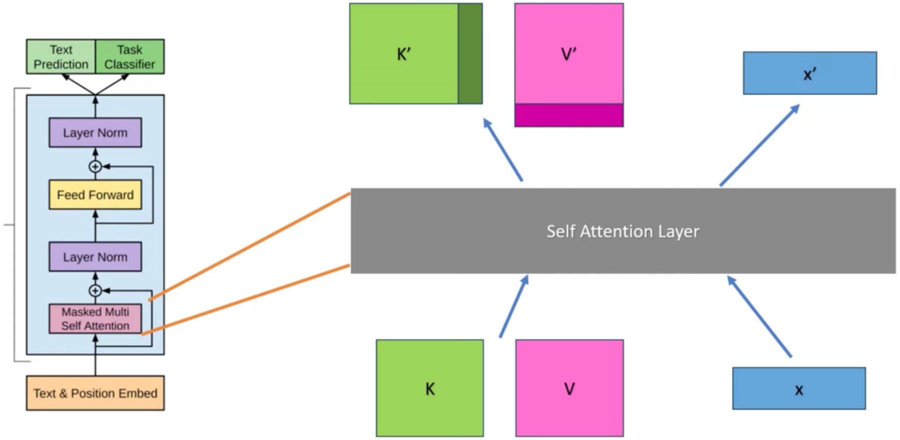


https://www.youtube.com/watch?v=80bIUggRJf4

Deploying model architecture

- KV Cache is the dominant factor





https://www.youtube.com/watch?v=80bIUggRJf4

Deploying model architecture

- KV Cache is the dominant factor

Memory Usage

$$2 * precision * n_{layers} * d_{model} * seqlen * batch$$

2 = two matrices for K and V

precision = bytes per parameter (eg: 4 for fp32)

n_{layers} = layers in the model

d_{model} = dimension of embeddings

seqlen = length of context in tokens

batch = batch size

Example: OPT-30B

$$2 * precision * n_{layers} * d_{model} * seqlen * batch$$

2 = two matrices for K and V

precision = 2 (use fp16 inference)

n_{layers} = 48

d_{model} = 7168

seqlen = 1024

batch = 128

KV cache: 180 GB

Model: 2*30B = 60GB

https://www.youtube.com/watch?v=80bIUggRJf4

Evaluation benchmarks

- Evaluation on standard benchmarks is critical for reporting performance.
- Benchmarks tests for skills:
  - language understanding, question-answering, math problem-solving, and coding tasks
- Different benchmarks for different model types:
  - LLM, VLM, Embedding models, Speech, Video, etc.
- Limitations of LLM benchmarks :
  - data contamination
  - Training and test on same data
  - narrow focus,
  - loss of relevance over time as model capabilities surpass benchmarks.
  - Applicability to enterprise situation

MMLU (Massive Multitask Language Understanding) benchmark

Conceptual Physics

When you drop a ball from rest it accelerates downward at 9.8 m/s². If you instead throw it downward assuming no air resistance its acceleration immediately after leaving your hand is
(A) 9.8 m/s² ✓
(B) more than 9.8 m/s² ✗
(C) less than 9.8 m/s² ✗
(D) Cannot say unless the speed of throw is given. ✗

College Mathematics

In the complex z-plane, the set of points satisfying the equation $z^2 = |z|^2$ is a
(A) pair of points ✗
(B) circle ✗
(C) half-line ✗
(D) line ✓

HellaSwag

Assets: HellaSwag dataset (GitHub), HellaSwag leaderboard
Paper: HellaSwag: Can a Machine Really Finish Your Sentence? by Zellers et al. (2019)

HellaSwag is a benchmark designed to test commonsense natural language inference. It requires the model to predict the most likely ending of a sentence. Similar to ARC, HellaSwag is structured as a multiple-choice task. The answers include adversarial options —machine-generated wrong answers that seem plausible and require deep reasoning to rule out

AI2 Reasoning Challenge (ARC)

Assets: ARC dataset (HuggingFace), ARC leaderboard
Research: Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge by Clark et al. (2018)

The AI2 Reasoning Challenge (ARC) benchmark evaluates the ability of AI models to answer complex science questions that require logical reasoning beyond pattern matching. It was created by the Allen Institute for AI (AI2) and consists of over 7700 grade-school level,

Evaluation benchmarks

VLM text generation benchmarks

	Molmo-E	InternVL2	Phi3v	Phi3.5v	Granite Vision
Document benchmarks					
DocVQA	0.66	0.87	0.87	**0.88**	**0.88**
ChartQA	0.60	0.75	0.81	0.82	**0.86**
TextVQA	0.62	0.72	0.69	0.7	**0.76**
AI2D	0.63	0.74	**0.79**	**0.79**	0.78
InfoVQA	0.44	0.58	0.55	0.61	**0.63**
OCRBench	0.65	**0.75**	0.64	0.64	**0.75**
LiveXiv VQA	0.47	0.51	**0.61**	-	**0.61**
LiveXiv TQA	0.36	0.38	0.48	-	**0.55**
Other benchmarks					
MMMU	0.32	0.35	0.42	**0.44**	0.35
VQAv2	0.57	0.75	0.76	0.77	**0.81**
RealWorldQA	0.55	0.34	0.60	0.58	**0.65**
VizWiz VQA	0.49	0.46	0.57	0.57	**0.64**
OK VQA	0.40	0.44	0.51	0.53	**0.57**

- Elevator toolkit has 20 datasets for VLM embeddings

We support the downstream evaluation of image classification on 20 datasets: `Caltech101`, `CIFAR10`, `CIFAR100`, `Country211`, `DTD`, `EuroSat`, `FER2013`, `FGVCAircraft`, `Food101`, `GTSRB`, `HatefulMemes`, `KittiDistance`, `MNIST`, `Flowers102`, `OxfordPets`, `PatchCamelyon`, `SST2`, `RESISC45`, `StanfordCars`, `VOC2007`. Our toolkit also supports `ImageNet-1K` evaluation, whose result is shown as reference on the

Reporting performance on benchmarks

- Gives indication of the level of difficulty
  - Prompt: "A photo of {}."

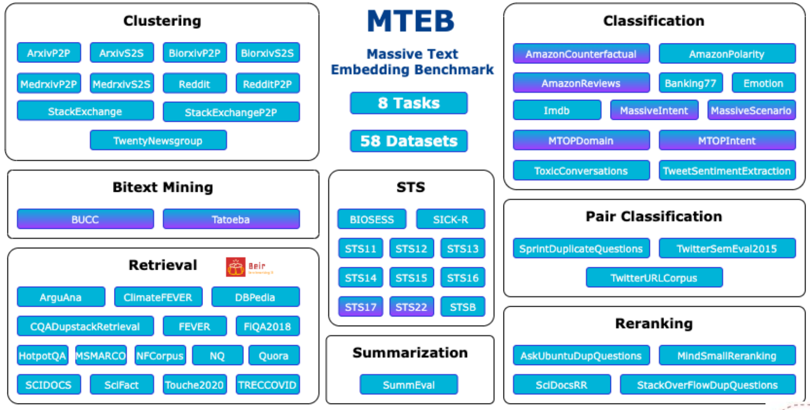
	caltech101		cifar10		cifar100		country211		dtd		eurosat		fgvc		flowers102	
	NDCG@1	NDCG@10	NDCG@1	NDCG@10	NDCG@1	NDCG@10	NDCG@1	NDCG@10	NDCG@1	NDCG@10	NDCG@1	NDCG@10	NDCG@1	NDCG@10	NDCG@1	NDCG@10
CLIP	94.1	92.4	100.0	99.1	88.0	82.4	39.3	30.7	46.8	45.1	60.0	63.7	17.0	19.5	58.8	56.7
CLIP336	97.0	95.1	100.0	99.3	91.0	87.6	63.0	52.6	55.3	57.5	70.0	68.0	47.0	36.5	67.6	69.3
OpenCLIP	95.0	94.9	100.0	100.0	91.0	90.7	50.7	39.4	68.1	59.5	70.0	61.2	25.0	25.6	67.6	64.7
SigLIP	95.0	96.1	100.0	99.3	89.0	91.1	40.3	33.7	70.2	69.5	60.0	60.5	44.0	43.1	87.3	85.8

	food101		gtsrb		mnist		oxfordpet		pcam		sst2		voc2007	
	NDCG@1	NDCG@10	NDCG@1	NDCG@10	NDCG@1	NDCG@10	NDCG@1	NDCG@10	NDCG@1	NDCG@10	NDCG@1	NDCG@10	NDCG@1	NDCG@10
CLIP	100.0	96.8	30.2	29.5	30.0	36.0	83.8	86.6	50.0	78.9	50.0	43.7	95.0	97.5
CLIP336	99.0	98.0	53.5	51.0	80.0	83.5	97.3	91.9	100.0	74.1	50.0	56.8	100.0	99.6
OpenCLIP	98.0	97.7	46.5	44.4	80.0	80.3	91.9	92.9	100.0	78.4	50.0	45.5	100.0	99.3
SigLIP	99.0	99.0	58.1	57.4	100.0	98.6	97.3	96.2	100.0	76.6	100.0	68.8	100.0	99.3

	Average	
	NDCG@1	NDCG@10
CLIP	62.9	63.9
CLIP336	78.1	74.7
OpenCLIP	75.6	71.6
SigLIP	82.7	78.3

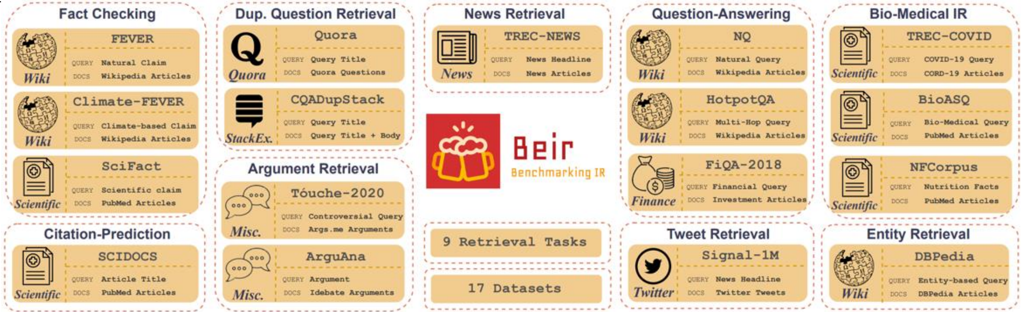
Evaluation benchmarks

- Embedding benchmarks



Evaluate on many tasks:

- Entity extraction
- Clustering
- Classification
- Sentence completion
- Question answering
- Retrieval
- Re-ranking



Evaluating using benchmarks

- **Classification Metrics** like accuracy.
  - These metrics are ideal for tasks with a single correct answer.
- **Overlap-based metrics**
  - **Lexical matching methods e.g. BLEU, ROUGE**
  - Semantic scoring methods, e.g. cosine similarity
  - Perplexity metrics ->coherence, conciseness, readability
- **Functional code quality.**
  - Some coding benchmarks, like HumanEval, use unique metrics such as pass@k, which reflects how many generated code samples pass unit tests for given problems.
- **Fine-tuned evaluator models.**
  - The TruthfulQA benchmark uses a fine-tuned evaluator called "GPT-Judge" (based on GPT-3) to assess the truthfulness of answers by classifying them as true or false.
- **LLM-as-a-judge**.
  - MT-bench introduced LLM-based evaluation to approximate human preferences. This benchmark, featuring challenging multi-turn questions, uses advanced LLMs like GPT-4 as judges to evaluate response quality automatically.

Preparing model cards and papers

- https://huggingface.co/ibm-granite/granite-vision-3.2-2b

ibm-granite / **granite-vision-3.2-2b** like 87 Follow IBM Granite 1.66k

Image-Text-to-Text · Transformers · Safetensors · English · llava_next · conversational · text-generation-inference · arxiv:2502.09927 · License: apache-2.0

Model card · Files and versions · Community 9 · Train · Deploy · Use this model

granite-vision-3.2-2b

Model Summary: granite-vision-3.2-2b is a compact and efficient vision-language model, specifically designed for visual document understanding, enabling automated content extraction from tables, charts, infographics, plots, diagrams, and more. The model was trained on a meticulously curated instruction-following dataset, comprising diverse public datasets and synthetic datasets tailored to support a wide range of document understanding and general image tasks. It was trained by fine-tuning a Granite large language model with both image and text modalities.

Evaluations:

We evaluated Granite Vision 3.2 alongside other vision-language models (VLMs) in the 1B-4B parameter range using the standard llms-eval benchmark. The evaluation spanned multiple public benchmarks, with particular emphasis on document understanding tasks while also including general visual question-answering benchmarks.

	Molmo-E	InternVL2	Phi3v	Phi3.5v	Granite Vision
Document benchmarks					
DocVQA	0.66	0.87	0.87	0.88	**0.89**
ChartQA	0.60	0.75	0.81	0.82	**0.87**

Downloads last month
22,563

Safetensors
Model size 2.98B params · Tensor type BF16 · Chat template · Files info

Inference Providers NEW
Image-Text-to-Text
This model isn't deployed by any Inference Provider. 2 Ask for provider support

Model tree for ibm-granite/granite-vision-3.2-2b

Base model	ibm-granite/granite-3.1-2b-base
Finetuned	ibm-granite/granite-3.1-2b-instruct
Finetuned (11)	this model
Finetunes	5 models
Quantizations	6 models

Space using ibm-granite/granite-vision-3.2-2b 1
ibm-granite/granite-vision-demo

Collection including ibm-granite/granite-vision-3.2-2b

https://arxiv.org/abs/2502.09927

Releasing models in open source

- Most popular site is HuggingFace
- It is a git-based repository to track all versions
- Can be done as individual or through an organization umbrella
- Need to clear internal open source processes before upload.
- Models can be used from open source if they are designed for a library that has built-in support.
  - Custom models that use trust_remote_code=True can also leverage these methods.
- In case your model is a custom PyTorch model, one can leverage the PyTorchModelHubMixin class as it allows to add from_pretrained, push_to_hub to any nn.Module class, just like models in the Transformers, Diffusers and Timm libraries.
- In addition to programmatic uploads, you can always use the web interface or the git command line.
- More details on:
  - https://huggingface.co/docs/hub/en/models-uploading#upload-from-a-library-with-built-in-support
- Other open source platforms:
  - DeepSeek, Tensorflow, PyTorch, Keras, Scikit-learn
- You can provide training code and inference code or inference only