ALMA MATER STUDIORUM UNIVERSITY OF BOLOGNA
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING – DISI

Large Language Models: A Short Review on How the Magic of Text Mining & NLP is Revolutionizing AI

Prof. Gianluca Moro
Dr. Giacomo Frisoni Dr. Luca Ragazzi

DISI, University of Bologna, Cesena
Via dell'Università, 50 I-47522 Cesena (FC), Italy
{name.surname}@unibo.it

Text Mining & NLP at DISI-Cesena

https://disi-unibo-nlp.github.io/

- NLP Group at DISI UniBo, Cesena Campus, led by Prof. Gianluca Moro*
  - Developing research solutions for Text Mining & **Natural Language Processing**, Understanding, Text Generation, Knowledge Graph Learning and Injection from General-domain and Commonsense to Biomedical, Legal applications etc.
  - 2 post-docs, 3 PhD students, 5 ex-research fellows hired in AI multinational companies, startup founders; school and international members; the group is supported by competitive projects and companies
- Publications in the last 3 years
  - ≈30 papers mainly in top-tier conferences e.g EMNLP, AAAI, ACL, ICLR, COLING, IJCAI, ECAI …
  - Journals: Neurocomputing, IEEE Access, AI and Law, Biodata Mining, Bioinformatics, Computer Methods and Programs in Biomed., Computers & Security …
  - Collaboration with impactful international companies and research groups
- *Teaching at DISI-Cesena
  - Since 2013/14: Text Mining & NLP module in Data Mining ISI master degree, Data Intensive Application covering Fundamentals & Applications of Machine Learning in the 1st level laurea degree ISI (+60 thesis students in machine learning in the last 5 years)
  - Since 2020: Data Mining, Text Mining and Big Data Analytics, Text Mining module, at the master in Artificial Intelligence, DISI-Bologna
  - Since 1999/00: in the 1st and then 2nd Engineering Faculty at Cesena (DEIS) and at the Statistics Science Faculty in Rimini, teaching Data Mining, Databases and Information Systems.

Introduction

Prompt: Transformer toy looking at the horizon and the origin of a new universe

Natural Language

- **Noam Chomsky**, a linguist, cognitive scientist and philosopher emphasized the innate and universal aspects of human language, suggesting that **the ability to acquire language is a fundamental and unique feature of the human mind**
  - His opinion on large language models is that *"they can't tell us anything about human language learning, but they can excel at misleading the uninformed"*.

Language has transformed our species by providing groups of people with a way to connect human brains and learn about the world and how we shape it. (C.D. Manning, Prof. of NLP at Stanford)

Text Mining develops specific AI advances for Natural Language Processing

Language Modeling | Information Retrieval | Document Summarization | Conversational Agents
Question Answering | Machine Translation | Document Classification | Sentiment Analysis | Topic Modeling
Knowledge Extraction | Text and Knowledge Graph Generation | Fact Checking | Spell Checking
Data-to-Text | Text-to-Speech/Image/Video/Code | Structured Prediction | Recommendation | Intent Prediction

NLP-Rooted AI Revolution

- AI is having its Cambrian explosion moment
  - Powered by large language models (LLMs)

ChatGPT (launched in November 2022)
- The fastest growing web platform ever [Source]: 100M monthly active users in 2 months only

HuggingFace (2016)
- AI Startup Valued at $4.5B [Source]
- 326,999 models + 60,155 datasets (11/09/2023)

Sparks of Artificial General Intelligence (AGI)
- LLMs have demonstrated striking human-level performance in many tasks, including the Bar exam [OpenAI, 2023] and the US Medical Licensing Exam [Nori et al., 2023]
- Some experts believe we are witnessing an early stage of AGI [Bubeck et al., 2023]

Valuable Market
- NLP market projected to grow from $24.10B in 2023 to $112.28B by 2030, at an impressive CAGR of 24.65% [Source]

Popularity and community support
- Mass popularization and innovation akin to the early days of iOS and Android apps

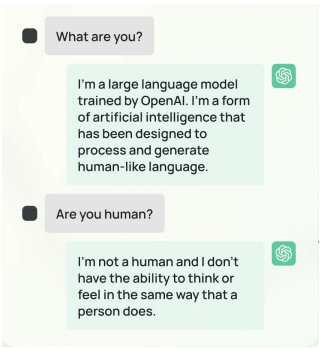
ChatGPT is a new category of product. It's not just a better search engine, auto-complete, or something else we already knew.

Andrew Ng, professor at Stanford and co-founder of Google Brain and Coursera

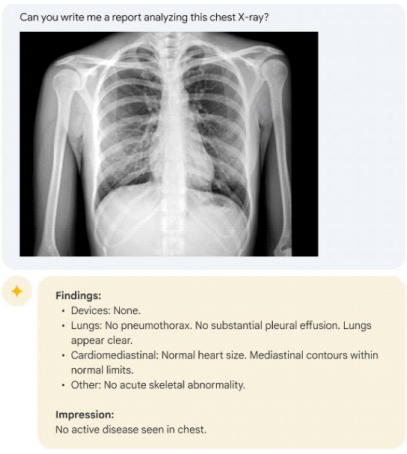
LLM Applications

Multimodal Conversational Agents

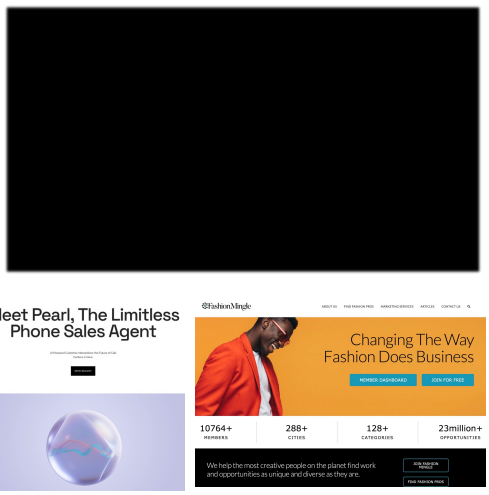
OpenAI ChatGPT



Google Med-PaLM 2 [Singhal et al., 2023]



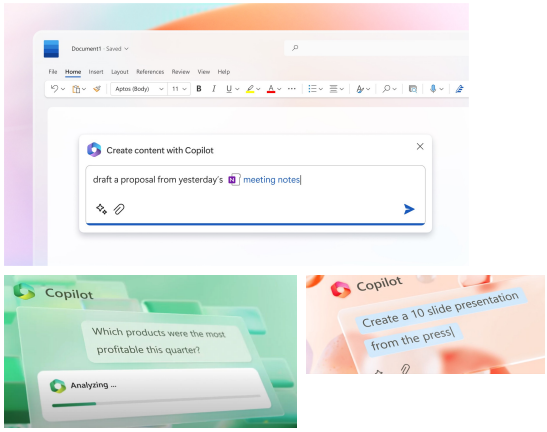
Google Gemini Ultra [Google DeepMind, 2023]



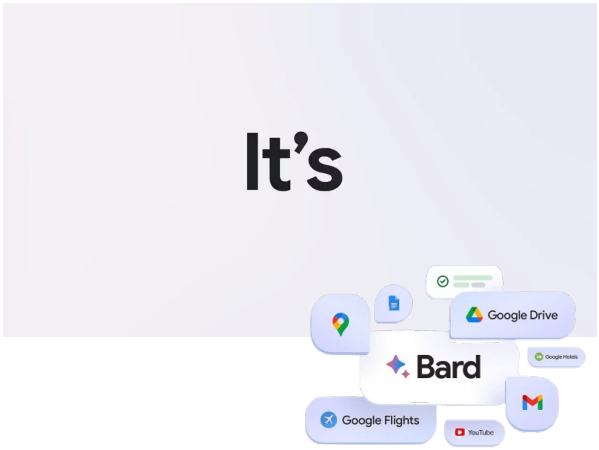
LLM Applications

Productivity Tools

Microsoft 365 Copilot



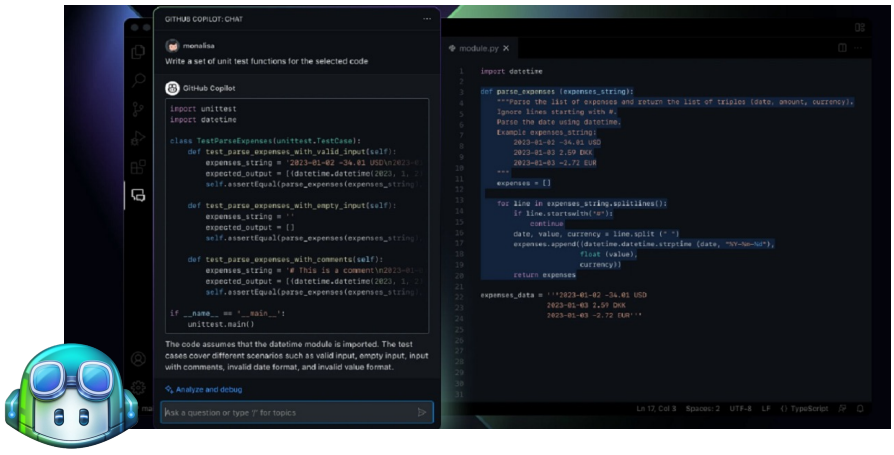
Google Bard w/ Extensions



LLM Applications

AI Pair Programming

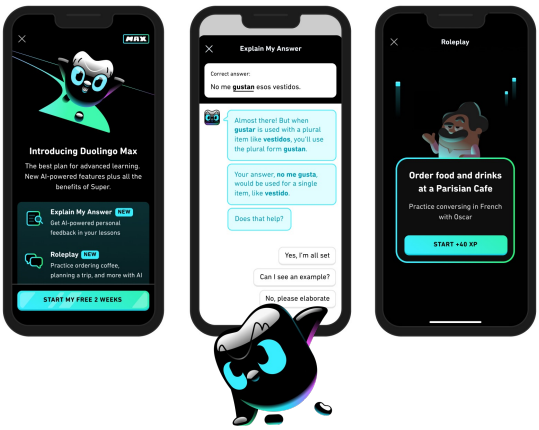
GitHub Copilot X



LLM Applications

Education

Duolingo Max

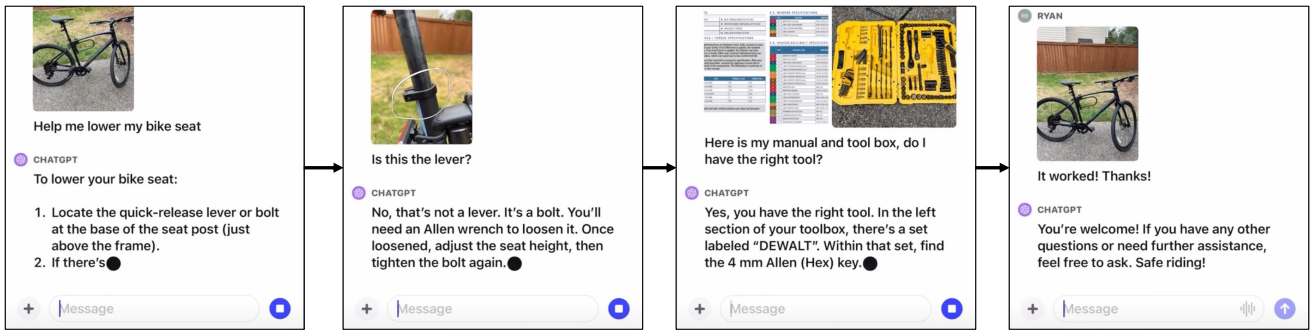


Audio and Image Generation

Meta AudioBox [Vyas et al., 2023]

LLM Applications

OpenAI ChatGPT



LLM Applications: *Large Action Models*

Rabbit R1

Multimodal LLM-in-a-box: AI pocket companion. Presented at CES 2024.
Combination of multiple LLMs and proprietary neuro-symbolic Large Action Models (LAMs).
LAM teach mode support.

Un modello linguistico rappresenta le parole come vettori in uno spazio latente costruito in base alla frequenza con cui le parole compaiono insieme. Ogni parola è rappresentata da un embedding vettoriale, ovvero una lista di numeri che cattura relazioni semantiche tra le parole. Lo spazio vettoriale ha dimensioni latenti che vengono apprese dal modello.

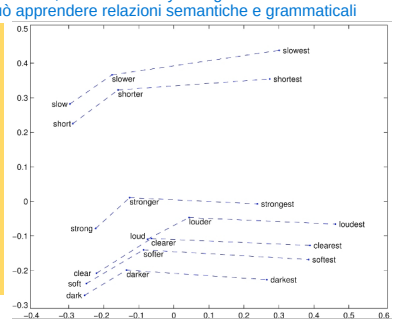
What is a Language Model ?

- It's a latent space built by co-occurrences of terms in large corpora of input text sets
  - Words, sentences, docs, graphs, images/videos .. everything is represented as a vector embedding
  - Vector components are the latent dimensions, whose number is an hyperparameter
  - The objective is to make the language semantics emerge from textual data by learning embeddings
- In practice, for example, from a large number of sentences such as
  - *altman plays a man in the comedy ...*
  - *deniro is a man like who has ...*
  - *if you like altman comedy as the ...*
  - *robert deniro a man in the comedy ...*
  - *bromwell high is a cartoon comedy ...*
- We get word embeddings such as
  - deniro = (-0.10, -0.02, 0.04, ...)
    altman = (-0.15, -0.04, 0.06, ...)
    man = (-0.05, -0.01, 0.02, ...)
    comedy = (-0.33, -0.10, -0.05, ...)
    cartoon = (-0.35, -0.15, -0.12, ...)
- The vector embedding learned for deniro is closer to that of altman than to cartoon as the first two vector contexts share more terms, e.g. *man, comedy ...*

Un Language Model impara le relazioni semantiche tra le parole osservando grandi quantità di testo. Questo permette di costruire rappresentazioni numeriche utili per compiti di NLP come traduzione automatica, sentiment analysis e generazione di testo. Un modello basato su reti neurali può apprendere relazioni semantiche e grammaticali

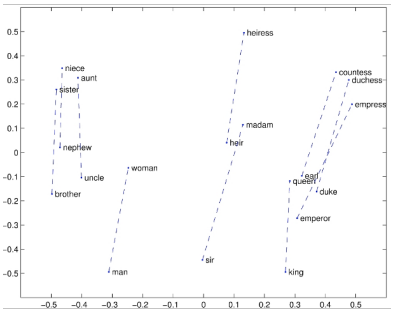
Latent space 2D projection learned by a neural net with a single hidden layer

word embeddings from self-supervised processing a 6 billion words' corpus.
The model has also learned adjective relationships relazioni tra aggettivi
Regular-Comparative-Superlative, e.g.
slow - slower ≅ short - shorter



The model has also learned gender relationships, e.g.

woman – man + king ≅ queen



1. Bag of Words: Un approccio semplice per rappresentare testi come insiemi di parole, ignorando la loro posizione e struttura. Si costruisce una matrice in cui le righe rappresentano i documenti e le colonne le parole (o termini) presenti in essi. Ogni cella contiene la frequenza di quella parola in quel documento. Non cattura il significato delle parole o le relazioni tra di esse. Genera matrici molto sparse e di grandi dimensioni.
2. LSA: Tecnica basata sull'algebra lineare per ridurre la dimensionalità delle rappresentazioni testuali. Si parte da una matrice BoW. Si applica la Singular Value Decomposition (SVD) per ridurre la dimensionalità della matrice. Questo aiuta a trovare relazioni latenti tra le parole e i documenti, anche se non sono direttamente collegate. Richiede molta memoria per calcolare la decomposizione SVD. Non gestisce bene l'ambiguità delle parole (polisemia).
3. pLSA: Un'estensione probabilistica di LSA per modellare la relazione tra parole e documenti attraverso distribuzioni di probabilità. Ogni documento è rappresentato come una distribuzione di argomenti. Ogni argomento è una distribuzione di parole. Utilizza la massimizzazione dell'attesa (EM Algorithm) per apprendere queste distribuzioni. Può soffrire di overfitting (tende ad adattarsi troppo ai dati di addestramento).
4. Modelli Neurali Monosemici: Modelli basati su reti neurali che assegnano un'unica rappresentazione vettoriale a ogni parola, indipendentemente dal contesto. Questi modelli non gestiscono la polisemia, ovvero il fatto che una parola possa avere più significati a seconda del contesto.
5. Modelli Neurali Polisemici: Modelli avanzati che generano rappresentazioni contestuali delle parole. Gestiscono il significato delle parole in base al contesto. Sono i modelli più avanzati e performanti nell'NLP. Sono molto pesanti dal punto di vista computazionale. Richiedono grandi quantità di dati per l'addestramento.

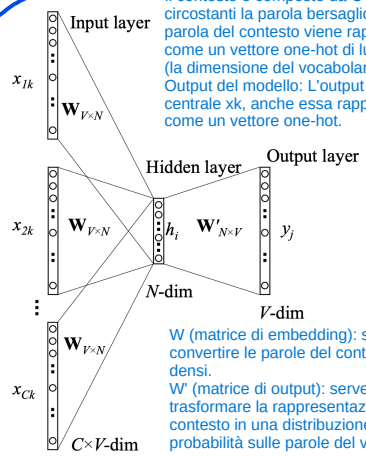
Types of Language Models

- **Bag of words** based on terms-documents matrices
- **Algebraic** models, such as Latent Semantic Analysis and Indexing (LSA) [Deerwester et. al. 1990]
- **Probabilistic** pLSA [Hofmann et. al., 1999], LDA for topic Analysis [Blei et al., 2003]
- **Monosemic Neural** Language Models such as Word2Vec [Mikolov et. al., 2013] and GloVe [Pennington et. al. 2014]
- **Polysemic Neural** Language Models such as ELMO, FLAIR, all the *BERTology* models, including current Large Language Models

L'obiettivo principale è predire una parola data il suo contesto.

The simplest Neural Language Model

- Given a term x_k , in k-th position in the vocabulary V of words
- The neural net inputs are the context C of x_k , i.e. surrounding words $x_{1,k}$, …, $x_{C,k}$ without x_k
  - Each $x_{i,k}$ is one-hot encoded vector of length |V|
- It is trained to predict y_j , i.e. the one-hot encoded vector of x_k
  - Let's w_c be the learnable representation of the term x_k in the context with 2*m* words, it learns the parameters W, W'



Input del modello: Dato un vocabolario di parole V, una parola xk in posizione k è circondata da altre parole nel suo contesto. Il contesto è composto da C parole circostanti la parola bersaglio xk. Ogni parola del contesto viene rappresentata come un vettore one-hot di lunghezza |V| (la dimensione del vocabolario).
Output del modello: L'output è la parola centrale xk, anche essa rappresentata come un vettore one-hot.

W (matrice di embedding): serve per convertire le parole del contesto in vettori densi.
W' (matrice di output): serve per trasformare la rappresentazione del contesto in una distribuzione di probabilità sulle parole del vocabolario.

$$\text{minimize } J = -\log P(w_c | w_{c-m}, \ldots, w_{c-1}, w_{c+1}, \ldots, w_{c+m})$$

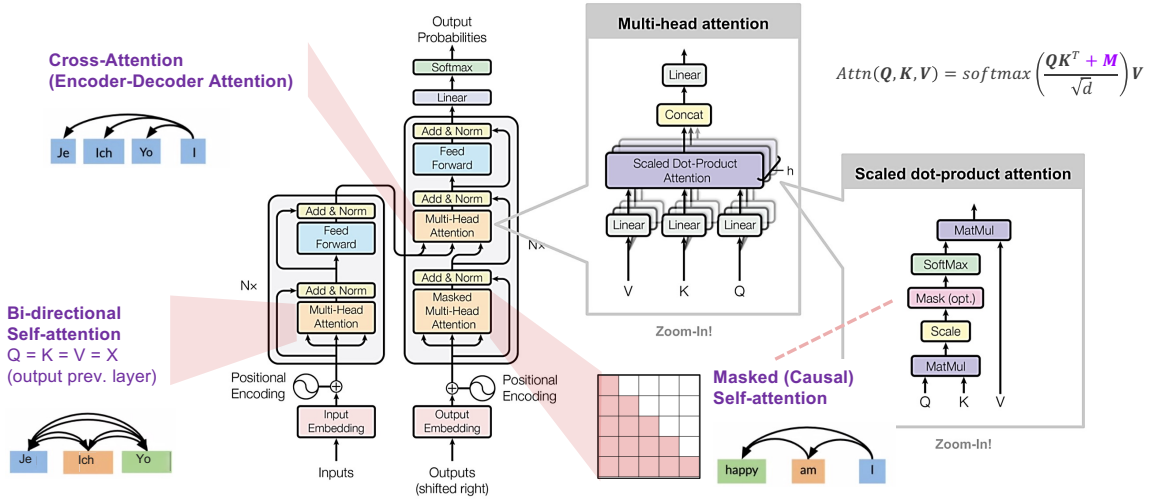
cross-entropy — dato — dove wc è la parola bersaglio e wc−m,...,wc+m sono le parole di contesto

Language Models are at the basis of any Text Mining & NLP tasks

Come funziona?
- Input Layer: Prende in input 2m parole che circondano la parola bersaglio.
Ogni parola è rappresentata come un vettore one-hot (tutti 0 tranne 1 in una posizione corrispondente alla parola nel vocabolario).
- Hidden Layer: I vettori one-hot vengono moltiplicati per la matrice W, producendo una rappresentazione densa delle parole. La rappresentazione vettoriale del contesto viene calcolata come la media delle embedding delle parole circostanti.
- Output Layer: Il vettore risultante viene moltiplicato per la matrice W', producendo una distribuzione di probabilità sulle parole del vocabolario. Viene applicata una softmax per ottenere la probabilità di ogni parola nel vocabolario.
- Ottimizzazione: L'obiettivo è minimizzare la funzione di loss per migliorare la previsione della parola bersaglio.

Transformers [Vaswani et al., 2017] – New Language Model Technology

- The magic backbone of almost every LLM [Kalyan et al., 2021]



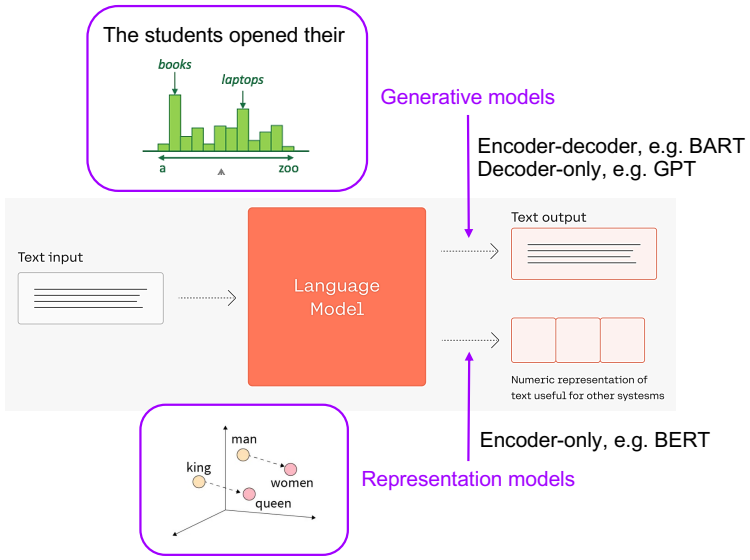
Transformers [Vaswani et al., 2017] – ii

1 Vocabulary Generation
Byte-Pair Encoding, WordPiece, Unigram, SentencePiece, etc.

2 Pre-training
Self-supervised learning methods and tasks

3 Fine-tuning
Tasks and popular losses

Efficient Transformers
Managing long sequences (e.g., 128K input tokens). E.g., sparse/linear attention



Scaled Dot-Product Attention – i

- Computed employing 3 matrices: queries ($\boldsymbol{Q}$), keys ($\boldsymbol{K}$), and values ($\boldsymbol{V}$)
  - $\boldsymbol{Q_i} \in \mathbb{R}^{L \times d_k} = \boldsymbol{X}\boldsymbol{W}_i^Q,\ W_i^Q \in \mathbb{R}^{d \times d_k}$
  - $\boldsymbol{K_i} \in \mathbb{R}^{L \times d_k} = \boldsymbol{X}\boldsymbol{W}_i^K,\ W_i^K \in \mathbb{R}^{d \times d_k}$
  - $\boldsymbol{V_i} \in \mathbb{R}^{L \times d_v} = \boldsymbol{X}\boldsymbol{W}_i^V,\ W_i^V \in \mathbb{R}^{d \times d_v}$
  - The matrices are different for each multi-head attention layer with *h heads,* therefore $i = 0, 1, \ldots h-1$
  - Usually we have $d_k = d_v = {}^{d}/_{h}$ = latent dimensions aka features

Similarity between different projections of the embeddings ($\mathbb{R}^{l \times l}$)

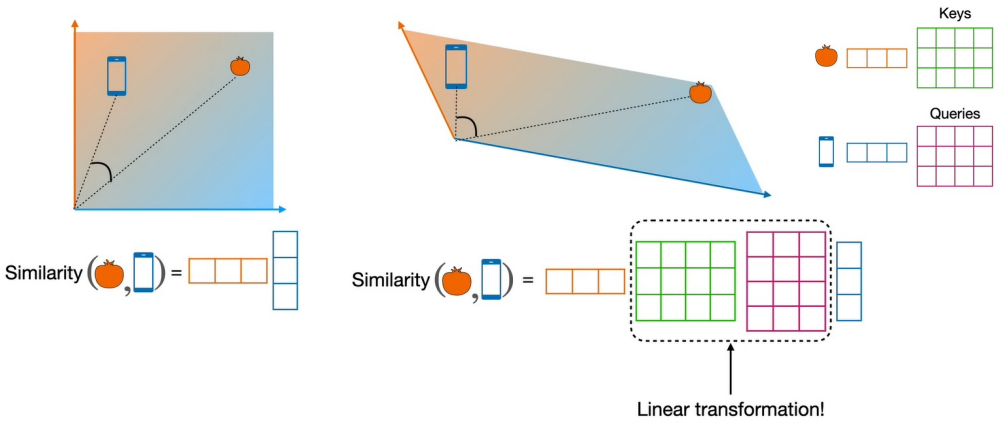
$$Attention(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = softmax\left(\frac{\boldsymbol{Q}\boldsymbol{K}^\top}{\sqrt{d_k}}\right)\boldsymbol{V}$$

Scaling factor prevents the product $\boldsymbol{Q}\boldsymbol{K}^\top$ from becoming too large as the dimensionality of vectors increases

- Time and Space complexity are $O(L^2 d)$ and $O(L^2 + Ld)$, respectively
  - E.g., BERT-large has 24 multi-head attention layers, each with h=16 single head attentions

Scaled Dot-Product Attention – ii

- W^Q and W^K work together to create linear transformations
  - Learnable transformation matrices used to project the input sequence in different spaces
  - Depending on their specifics, they will stretch, rotate or shear the original vectors to help find good embeddings where doing attention



Self-supervised LM Pre-training Methods – i

- **Token Unmasking** [Devlin et al. 2018]
- The model is asked to predict randomly masked tokens within a sentence
  - 15% of the tokens in each sentence are masked in different ways
  - (a) 80% substituted with a [MASK] token — My dog is hairy → My dog is **[MASK]**
  - (b) 10% replaced with a random word — My dog is hairy → My dog is **apple**
  - (c) 10% of the times they are left untouched — My dog is hairy
- The loss used is the *cross-entropy loss* (computed over the masked-out tokens)
  - Called M the number of masked token, t the one-hot encoded vectors of the correct words, and p the softmax output returned by the model

$$L_{MLM} = -\frac{1}{M}\sum_{i=1}^{M} \mathbf{t}_i \log(\boldsymbol{p}_i)$$

Self-supervised LM Pre-training Methods – ii

- Token/Span Unmasking is about learning probabilistic contextual relations

Cause-Effect
- **Input:** Due to **[MASK]**, the plane was postponed
- **Output:** bad weather

Conditional
- **Input:** If you **[MASK]** hard, you will get good results
- **Output:** study

Contrast
- **Input:** **[MASK]** the cold, I went for a run
- **Output:** Despite

Belonging
- **Input:** That car **[MASK]** to my father
- **Output:** belongs

Comparison
- **Input:** The book **[MASK]** interesting than the movie
- **Output:** is more

Positional
- **Input:** In a Transformer architecture, the decoder blocks come **[MASK]** the encoder blocks
- **Output:** after

Temporal Order
- **Input:** **[MASK]** finishing breakfast, she headed to work
- **Output:** After

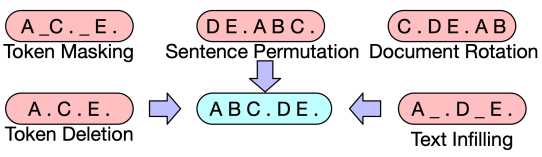
Domain-specific
- **Input:** Cough suggests underlying **[MASK]** issue
- **Output:** respiratory

Self-supervised LM Pre-training Methods – iii

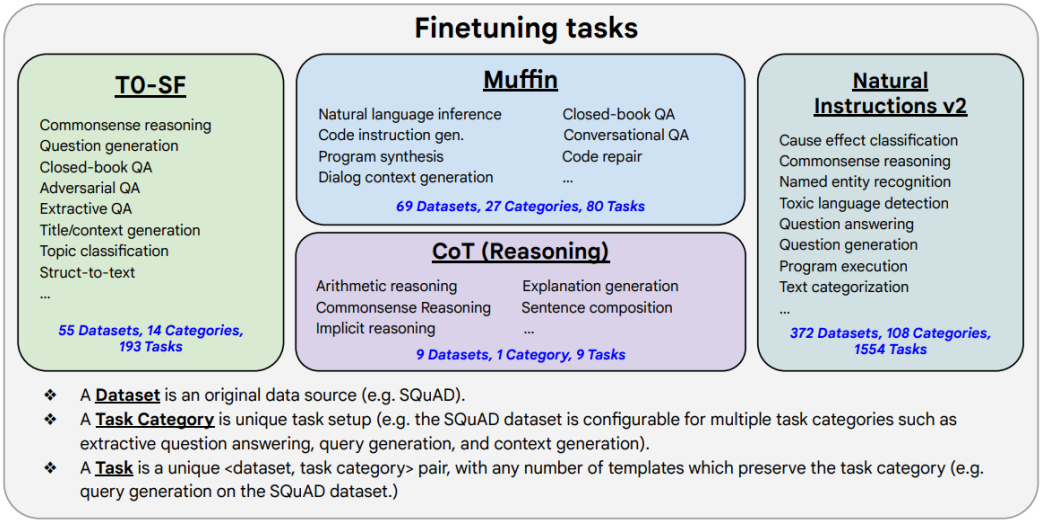
- **Next Sentence Prediction**
  - 50% of sentence pairs are given in the correct order, while the remaining half are pairs of sentences randomly selected from the corpus
  - This task promotes understanding relationships between sentences, a skill that is more difficult to learn using only the masked language modeling task
  - *Binary cross-entropy loss*
    - Called $y \in (0,1)$ the correct class and p the probability assigned by the model

$$L_{NSP} = -(y \log(p) + (1 - y) \log(1 - p))$$

… And many more, e.g., [Lewis et al., 2019]

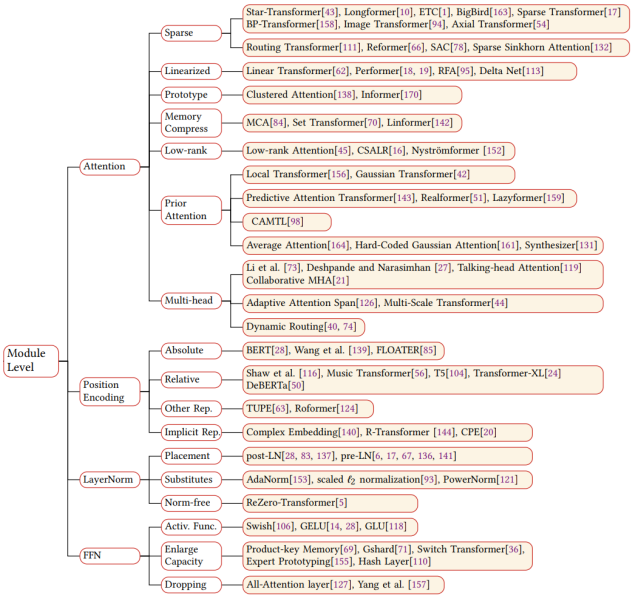


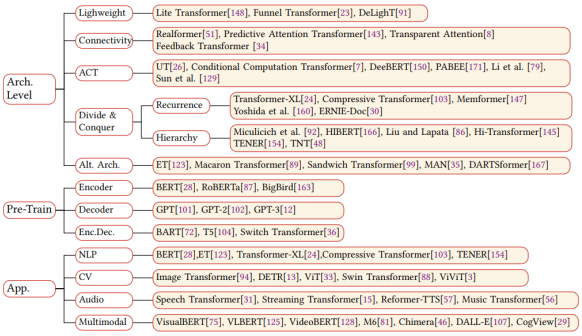
Supervised LM Fine-tuning



Example from Flan-T5 [Chung et al., 2022], where fine-tuning data comprises:
473 datasets, 146 task categories, and 1836 total tasks

Transformer Taxonomy

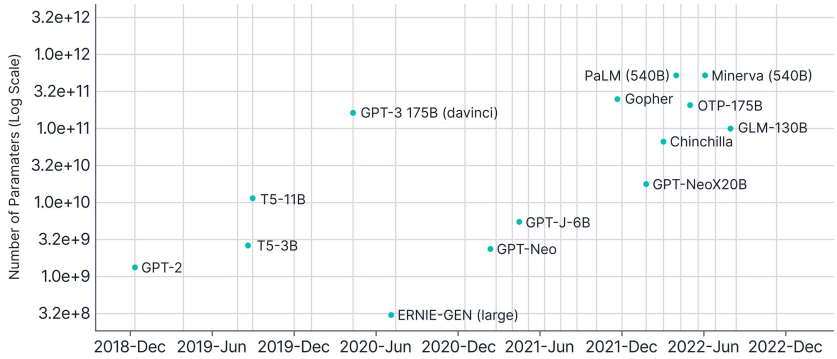




A diverse array of models based on the vanilla Transformer has been proposed, considering three key perspectives: types of architecture modification, pre-training methods, and applications. Lin et al. [2021] identified **+125 Transformer variants**. Since then, this count has continued to increase steadily.

Scaling Laws

- Numerous studies have revealed that increasing the size of PLMs can result in improved downstream task performance [Kaplan et al., 2020]
  - This insight motivated researchers to tackle complex tasks by employing larger models
  - Common approaches include stacking together more Transformer blocks, increasing the window length, and aggregating results from parallel self-attention layers

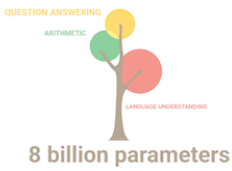


🧬 Brain biologically-inspired trend: the bigger, the better ?

Exponential growth over time
In just 3 years, we moved from 95M parameters (ELMO [Peters et al., 2018]) to 1.2T parameters (GLaM [Du et al., 2022])

Scaling Laws

- Numerous studies have revealed that increasing the size of PLMs can result in improved downstream task performance [Kaplan et al., 2020]
  - This insight motivated researchers to tackle complex tasks by employing larger models
  - Common approaches include stacking together more Transformer blocks, increasing the window length, and aggregating results from parallel self-attention layers



🧬 Brain biologically-inspired trend: the bigger, the better ?

Exponential growth over time
In just 3 years, we moved from 95M parameters (ELMO [Peters et al., 2018]) to 1.2T parameters (GLaM [Du et al., 2022])

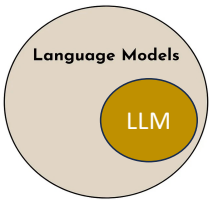
Cognitive Functions Emergence as the Parameters Increase – i

- Every neural network is an *approximator of functions*
  - A transformer models the semantic of tokens with signals represented by continuous functions with a finite and fixed domain
  - If we analogously depict human cognitive functions as a single, very complex function in two variables with many concavities and convexities…
  - … Increasing the parameters of the neural network always improves the approximation of the function that models human cognitive functions
  - With billion-scale networks, the approximation evidently becomes enough good, allowing to touch the more advanced aspects of language, e.g., the ability to converse and reasoning
  - With fewer parameters, we observe the approximation is coarser and fails to model those aspects

Cognitive Functions Emergence as the Parameters Increase – i

- Next-token prediction on petabytes of raw text encapsulating the breadth and depth of human language
  - we have never had a better moment in time for AI not only because of new research & technological advances, but also because of the vast amounts of data now available
  - Research works demonstrated that LLMs learn syntactics, semantics and memorize many facts of the world, since each of these aspects helps to reconstruct text successfully
  - Predicting a masked word initially seems a rather simple and low-level task, and not something sophisticated like diagramming a sentence to show its grammatical structure…
  - … But this pre-training tasks turns out to be very powerful because it is universal: every form of knowledge and reasoning help one to accomplish this goal better
  - As a result, LLMs assemble a broad general knowledge of the language and world to which they are exposed; accumulated knowledge can then be deployed for downstream tasks

What are Large Language Models ? – i

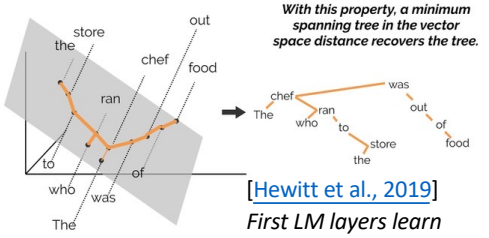


- Applying the same training methods of PLMs to at least ~10 billion param. models, special emergent abilities start to pop up
  - Replicating human-like abilities [Zhao et al., 2023]
- Two key distinction properties
  - **Quantitatively**, num. of parameters, pre-training corpus size
  - **Qualitatively**, emerging properties, e.g., zero-shot learning, in-context learning, reasoning

The core task used to train LLMs is **token prediction**

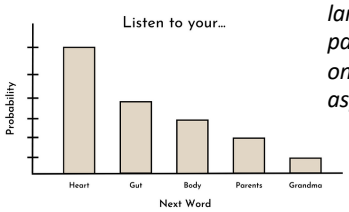
Unconditional Decoder-only $P(y_1, y_2, \dots, y_n) = \prod_{t=1}^{n} p(y_t | y_{<t})$

Conditional Seq2seq encoder-decoder $P(y_1, y_2, \dots, y_n | x) = \prod_{t=1}^{n} p(y_t | y_{<t}, x)$



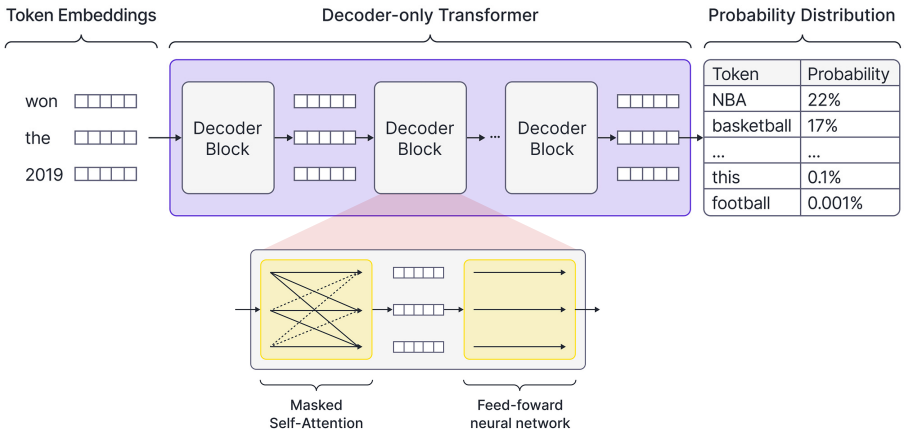
[Hewitt et al., 2019]

First LM layers learn mainly a latent space for language syntactic parsing, while the last ones learn the semantic aspects



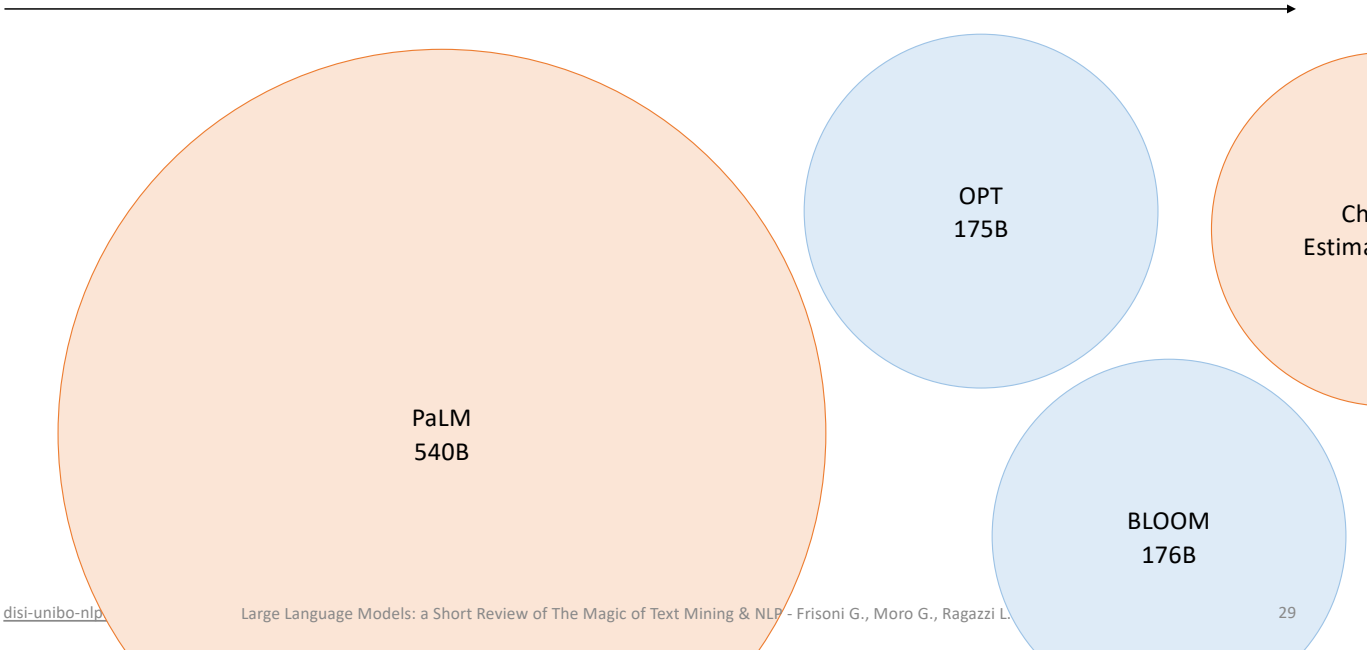
What are Large Language Models ? – ii

- Most LLMs, including the GPT family, are *generative auto-regressive* PLMs

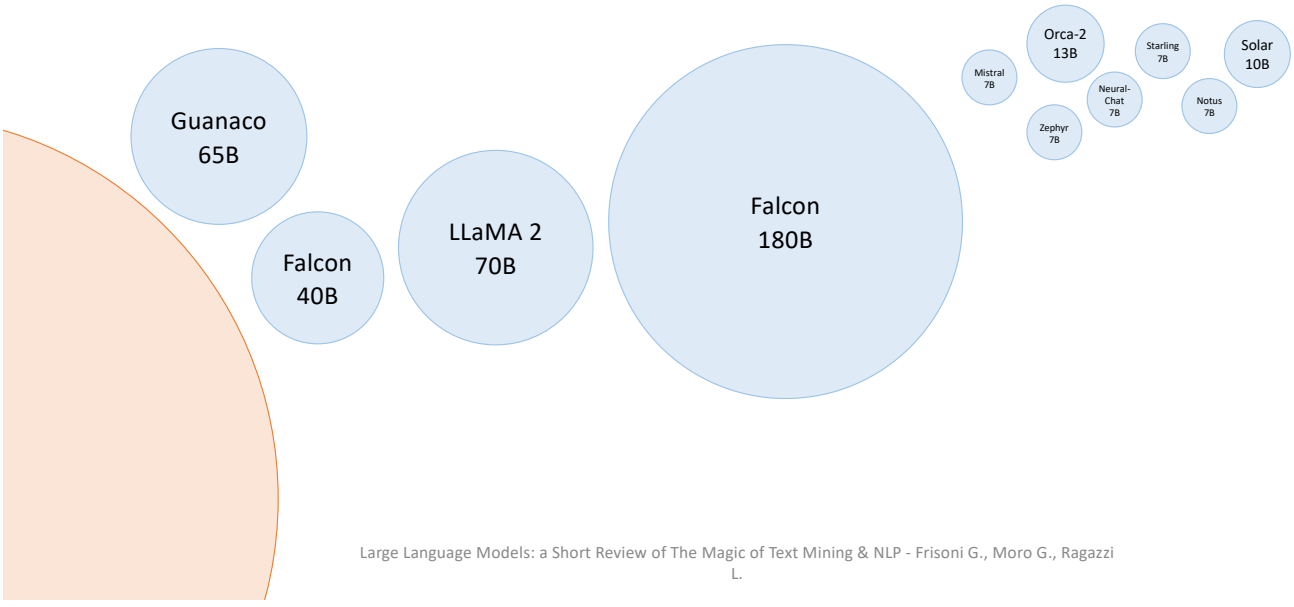


"Transformers walk into a bar, come out as large language models." – by GPT4

Large Language Models Wave



Large Language Models Wave



Pre-training: LLM Barrier to the Entry

See Hugging Face Training Cluster as a Service for details

- With great sizes comes great pre-training costs

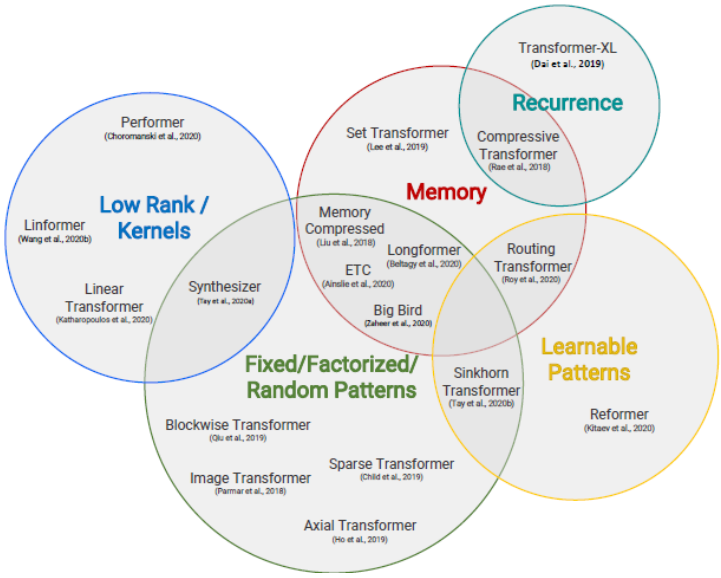
Size (B)	Tokens (B)	# GPUs	GPU Type	Price	Days	kW/H	CO2 Kg
7	490	200	A100	**€93,150**	10	16,800	4,200
7	490	200	H100	**€102,659**	5	8,400	2,100
13	390	200	A100	**€137,688**	15	25,200	6,300
13	390	200	H100	**€151,744**	7	11,760	2,940
70	1400	1000	A100	**€2,577,127**	59	495,600	123,900
70	1400	1000	H100	**€2,840,208**	28	235,200	58,800

GPT-3-175B pre-training (499B tokens)
- > $4.6M on Tesla V100 cloud instances
- 355 GPU-years
- 700GB GPU memory required for inference (at least 350 on specialized infrastructure)

Falcon-180B pre-training
- > $41M on AWS Sagemaker P4d instances (one P4d has 8 A100 40GB GPU for $ +47 per hour)
- 799 GPU-years

LLaMA-2-chat
- $8M overall pre-training
- $11M data

Attention Mechanism Optimizations as a Set of First Compression Methods [Tay et al., 2021]



Performer: A Kernel-based Example of Linear Attention

- It uses a function ϕ called *positive random feature map* $\phi: \mathbb{R}^d \rightarrow \mathbb{R}_+^r$ such that:

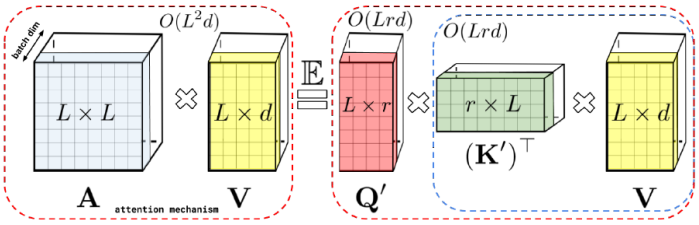
$$K(\boldsymbol{x}, \boldsymbol{y}) = \mathbb{E}[\phi(\boldsymbol{x})^\top \phi(\boldsymbol{y})]$$

  - It can be shown that this holds if, given $r > 0$, we define $\phi(\boldsymbol{x}) = \frac{h(\boldsymbol{x})}{\sqrt{r}} f(\boldsymbol{W}\boldsymbol{x})^\top$ where $\boldsymbol{W} \in \mathbb{R}^{r \times d}$ is a matrix of random orthogonal vectors, $h(\boldsymbol{x}) = \exp(\frac{\|\boldsymbol{x}\|^2}{2})$ and $f = \exp(\cdot)$

- With this trick, Attention is approximated in linear time with respect to the size of the input of length *L* by rewriting the formula as follows:

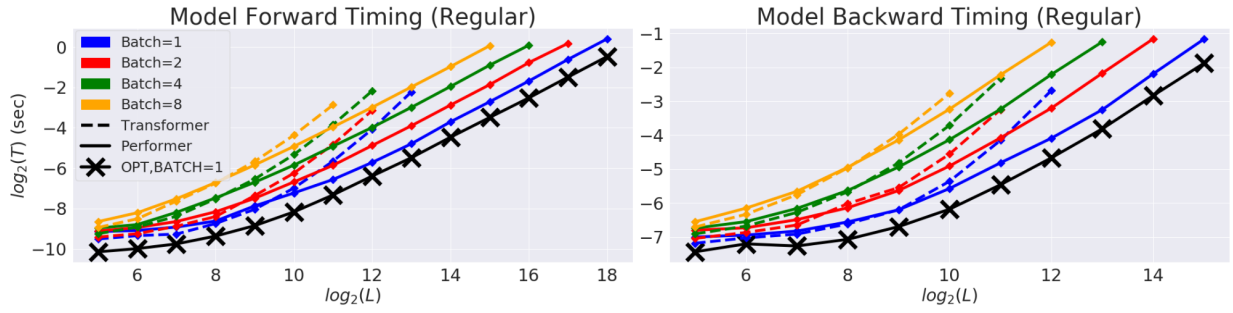
$$\widehat{Att}\,(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \widehat{\boldsymbol{D}}^{-1}\left(\boldsymbol{Q}'\left((\boldsymbol{K}')^\top \boldsymbol{V}\right)\right), \qquad \widehat{\boldsymbol{D}} = diag(\boldsymbol{Q}'\left((\boldsymbol{K}')^\top \mathbf{1}_L\right))$$

  - With $\boldsymbol{Q}', \boldsymbol{K}' \in \mathbb{R}^{L \times r}$ computed applying the function ϕ to the rows of the matrices $\boldsymbol{Q}$ *and* $\boldsymbol{K}$



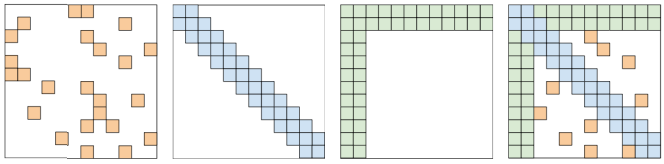
Performer: Computational Costs

- The log-log graph below shows, as L varies, the gap between training times for a quadratic attention Transformer (dashed lines) and Performer (solid lines)
- The symbol ~~x~~ represents the max achievable speed (using a "dummy" **placebo** attention block that performs nothing)
- Data are shown until the trainings produced a "memory error"



BigBird: Linear Attention based on Randomness

- BigBird [Zaheer et al., 2020] is a model with linear complexity that approximates attention using fixed pattern and memory mechanisms
  - Use sparse graphs in which the nodes are tokens of the input sequence, and the arcs represent the connections in the attention matrix
- BigBird combines 3 types of sparsification:
  - **Random attention**: each node is randomly linked to other r nodes
  - **Window attention**: each node is connected to its w neighboors ($\frac{w}{2}$ on its left and $\frac{w}{2}$ on its right); this choice comes from the heuristic consideration for which the most important information is generally local information
  - **Global attention**: g nodes are completely connected; the authors proved that with this type of connections, BigBird becomes a universal approximator of sequence-to-sequence functions and is Turing complete
  - **The input is divided in** *b* block, each for example of 64 tokens, and a quadratic attention is applied only within each block.



(a) Random attention (b) Window attention (c) Global Attention (d) BigBird

BigBird: Results

- Two models have been proposed
  - **BigBird-ITC** *(Internal Transformer Construction):* Global tokens are selected from within the input sequence
  - **BigBird-ETC** *(Extended Transformer Construction)*: Global tokens are newly added context tokens at the input start
  - A LARGE version was also implemented for both with 24 hidden layers of size 1024 and 24 'heads' of attention

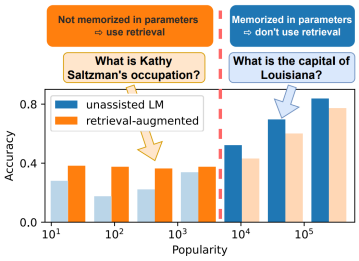
Parameter	BigBird-ITC	BigBird-ETC
Block length, b	64	84
# of global token, g	$2 \times b$	256
Window length, w	$3 \times b$	$3 \times b$
# of random token, r	$3 \times b$	0
Max. sequence length	4096	4096
# of heads	12	12
# of hidden layers	12	12
Hidden layer size	768	768

- State-of-the-art in question answering and text-classification
- Improved results in summarization tasks since **it can handle longer input sequences (i.e., up to 4096 token)**
  - BigBird-Pegasus, which is specifically pre-trained to generate summaries

	Model	Arxiv			PubMed			BigPatent		
		R-1	R-2	R-L	R-1	R-2	R-L	R-1	R-2	R-L
Base	Transformer	28.52	6.70	25.58	31.71	8.32	29.42	39.66	20.94	31.20
	+ RoBERTa	31.98	8.13	29.53	35.77	13.85	33.32	41.11	22.10	32.58
	+ Pegasus	34.81	10.16	30.14	39.98	15.15	35.89	43.55	20.43	31.80
	BigBird-RoBERTa	41.22	16.43	36.96	43.70	19.32	39.99	55.69	37.27	45.56
Large	Pegasus (Reported)	44.21	16.95	38.83	45.97	20.15	41.34	52.29	33.08	41.75
	Pegasus (Re-eval)	43.85	16.83	39.17	44.53	19.30	40.70	52.25	33.04	41.80
	BigBird-Pegasus	**46.63**	**19.02**	**41.77**	**46.32**	**20.65**	**42.33**	**60.64**	**42.46**	**50.01**

Large Language Model Problems

❌ LLMs can't memorize all knowledge in their parameters [Mallen et al., 2023]



❌ LLMs' knowledge is easily outdated and hard to update (knowledge editing is still active research [Meng et al., 2023])

Who is the CEO of Twitter?

As of my **knowledge cutoff in September 2021**, the CEO of Twitter is Jack Dorsey
[Correct Answer: Linda Yaccarino, Jun 5 2023]

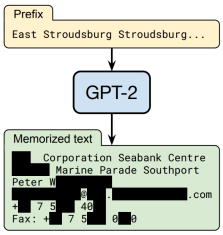
❌ LLM output is challenging to interpret and verify, frequently hallucinated

What are the effects of vitamin D on the immune system?

Vitamin D has anti-inflammatory properties and can help regulate the production of pro-inflammatory cytokines, such as tumor necrosis factor-alpha (TNF-α) and interleukin-6 (IL-6).
[Which are the references?]

❌ LLMs are shown to easily leak private training data [Carlini et al., 2021]

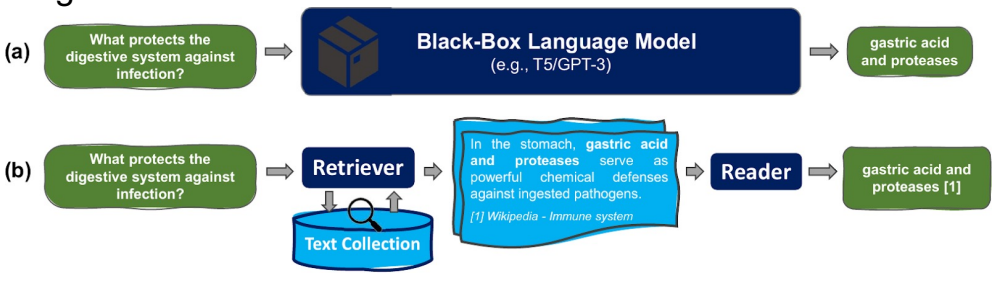
Category	Count
Named individuals (non-news samples only)	46
Contact info (address, email, phone, twitter, etc.)	32
Log files and error reports	79
License, terms of use, copyright notices	54



Retrieval-enhanced Language Models: Solutions to LLM Problems

A.k.a. *Retrieval-Augmented Generation Models (RAG)* + LLM

- Representative tasks: question answering, fact checking, information retrieval entity linking …



✅ Knowledge is outside the parameters, making models smaller, faster, and cost-effective

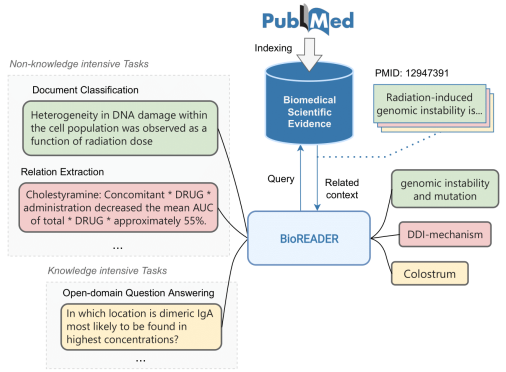
✅ The datastore can be easily updated and expanded, even without retraining!

✅ Knowledge sources can be traced from retrieval results, enabling explainability, interpretability and control (e.g., generating text with citations [Gao et al., 2023])

BioReader: a Retrieval-Enhanced Text-to-Text Transformer for Biomedical Literature

Frisoni G., Mizutani M., Moro G., Valgimigli L. Empirical Methods in Natural Language Processing (EMNLP 2022)

The first **retrieval-augmented generation** (RAG) PLM for biomedicine:
T5 model, empowered by a differentiable access towards an explicit large-scale text memory centered on PubMed (≈60M tokens)

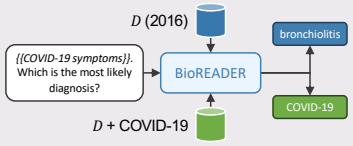


- We split the tokenized input X into a sequence of l chunks
- For each input chunk, we retrieve the top-k most similar neighboring chunks from an **external evidence datastore** D using the L2 distance
- In the decoder stack, we interleave the standard T5 blocks with blocks inspired by Borgeaud et al., 2022
- Retrieved knowledge is fused in the intermediate decoder activations via **chunked cross-attention** layers

15 biomedical NLP datasets spanning 6 task categories

We outperform SOTA methods on **12 datasets** and **5 task** using **3x fewer parameters**

Zero-shot datastore

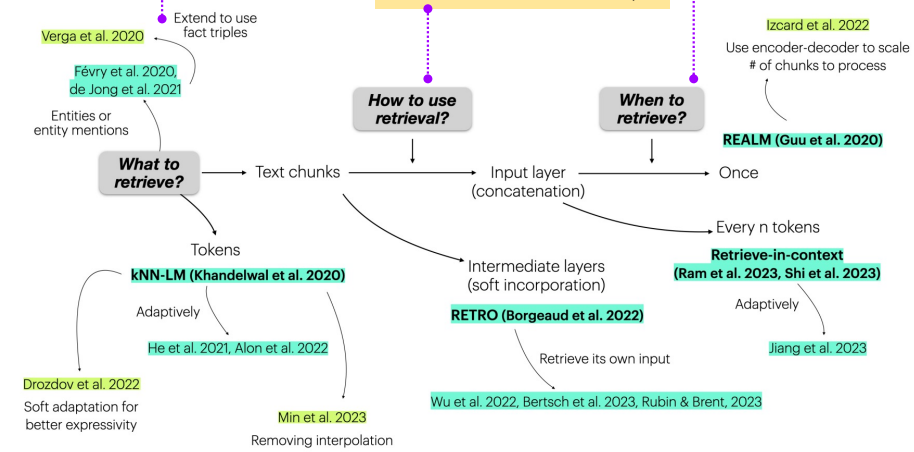


Retrieval-enhanced Language Models – Design Choices

(1) Given a query, **WHAT** to retrieve ?
- Text chunks (passages)
- Tokens
- Entities / Entity mentions
- Fact triples (<subject, predicate, object>)

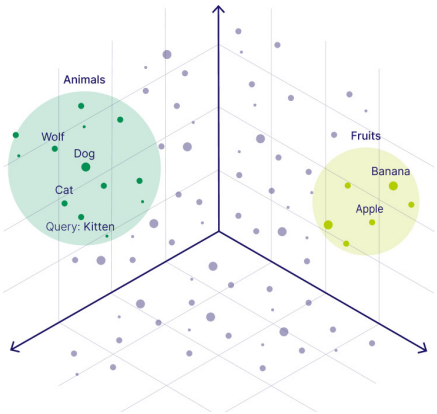
(2) HOW to use retrieval ?
- Retrieved items are concatenated to the original input (i.e., discrete input augmentation)
- Retrieved items are incorporated in intermediate layers (e.g., cross-chunk attention: BioReader, Retro)

(3) WHEN to retrieve ?
- Only once, for the entire input
- Every n tokens (input segmentation)

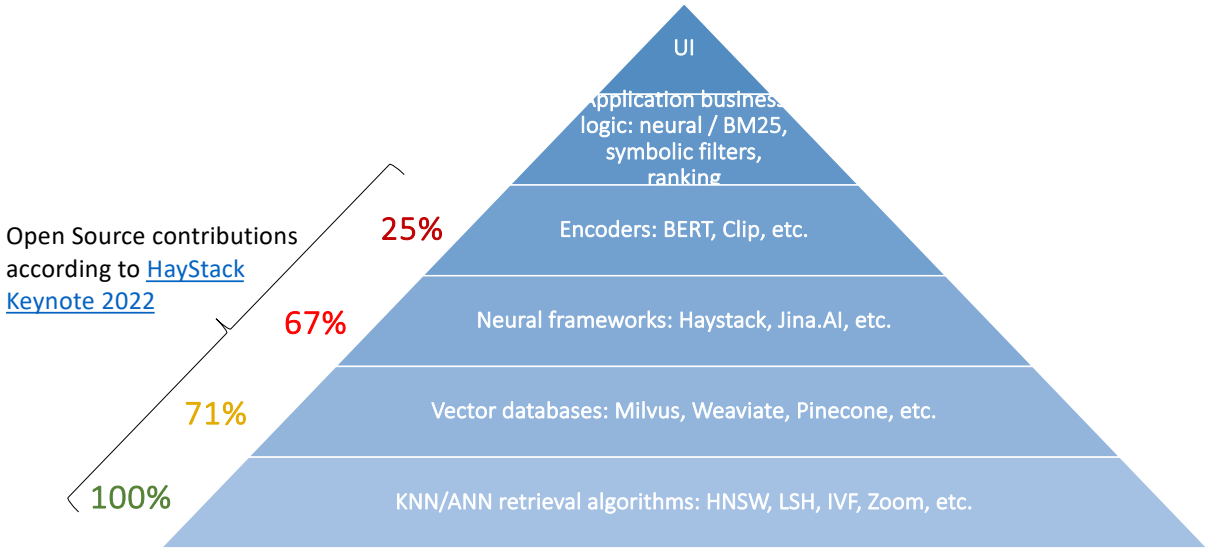


Vectors Indexing: a Crucial Requirement for RAG + LLM

- The process of organizing vector embeddings in a way that high-dimensional data can be retrieved efficiently
- **Approximate Nearest Neighbor (ANN)**
  - Pre-calculate the distances between the vector embeddings and organize and store similar vectors close to each other (e.g., in clusters or a graph)
  - **Clustering**-based index (e.g., FAISS)
  - **Proximity graph**-based index (e.g., HNSW)
  - **Tree**-based index (e.g., ANNOY)
  - **Hash**-based index (e.g., LSH)
  - **Compression**-based index (e.g., PQ or SCANN)



Vectors Indexing: the VectorDB Role in AI Applications



A Comparison of Leading VectorDBs – i

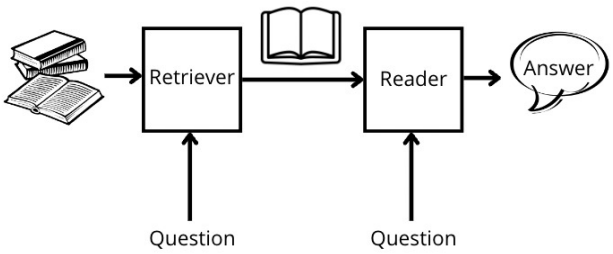
	Pinecone	Weaviate	Milvus	Qdrant	Chroma	Elasticsearch	PGvector
Open source	❌	✅	✅	✅	✅	❌	✅
Self-host	❌	✅	✅	✅	✅	✅	✅
Cloud management	✅	✅	✅	✅	❌	✅	(✓)
Purpose-built for Vectors	✅	✅	✅	✅	✅	❌	❌
Developer experience	👍👍👍	👍👍	👍👍	👍👍	👍👍	👍	👍
Community	Community page & events	8k☆ github, 4k slack	23k☆ github, 4k slack	13k☆ github, 3k discord	9k☆ github, 6k discord	23k slack	6k☆ github
Queries per second (using nytimes-256-angular)	150	791	2406	326	/	700	141
Latency, ms (using nytimes-256-angular)	1	2	1	4	/	/	8

A Comparison of Leading VectorDBs – ii

	Pinecone	Weaviate	Milvus	Qdrant	Chroma	Elasticsearch	PGvector
Supported index types	/	HNSW	Multiple (11 total)	HSNW	HNSW	HNSW	HNSW / IVFFlat
Hybrid search (i.e., scalar filtering)	✅	✅	✅	✅	✅	✅	✅
Disk index support	✅	✅	✅	✅	✅	❌	✅
Role-based access control	✅	❌	✅	❌	❌	✅	❌
Free hosted tier	✅	✅	✅	Free self-hosted	Free self-hosted	Free self-hosted	Varies

[Source 1, Source 2, Source 3, Source 4, Source 5]

Dense Passage Retrieval (V Karpukhin et al. EMNLP 2020)



- Dense Passage Retrieval model that given a collection of M text passages, index them in a low-dimensional and continuous space.
- Efficiently retrieve the top k passages relevant to the input question at run-time.
- Retrieval uses dense representations only, where embeddings are learned from a small number of questions and passages by a simple dual encoder framework.
- Two BERT models are fine-tuned using contrastive loss to align labeled query and key embeddings.

DPR - I

- The dense encoder $E_P(\cdot)$ maps any text passage to a d-dimensional real-valued vector and builds a FAISS index for all the M passages that are used for retrieval.
- At run-time a different encoder $E_Q(\cdot)$ maps the input question to a d-dimensional vector.
- The similarity between question and passage is computed using their dot product: $sim(q, p) = E_Q(q)^T E_P(p)$.
- In sparse retrievers semantically similar words (*"hey", "hello", "hey"*) will not be viewed as similar.
- Dense vectors are encoded with semantic meaning, improving the retriever.
- Sparse retrievers are **not** trainable instead DPR uses embedding functions that we can train and fine-tune for specific tasks.

DPR - II

- The training loss is the negative log likelihood of the positive passage:

$$L(q_i, p_i^+, p_{i,1}^-, \cdots, p_{i,n}^-) = -\log \frac{e^{\text{sim}(q_i, p_i^+)}}{e^{\text{sim}(q_i, p_i^+)} + \sum_{j=1}^{n} e^{\text{sim}(q_i, p_{i,j}^-)}}$$

- q_i represents questions, p_i^+ represents the relevant positive passage and $p_{i,j}^-$ are the n irrelevant negative passages.
- DPR requires long training and a lot of training data(curated dataset of question and context pairs).
- DPR has high computational requirements during both indexing and retrieval.

disi-unibo-nlp.github.io

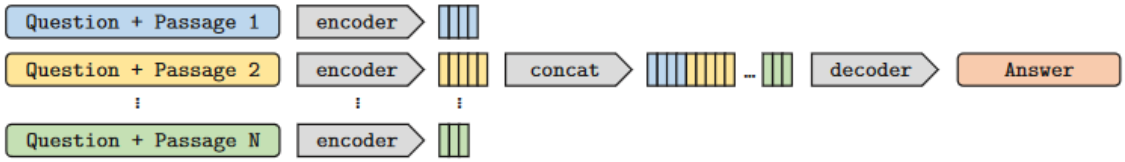
DPR - III

- The single training means that the retriever is trained on a single data source. Multi instead combines multiple sources in the training set.
- Top-20 & Top-100 retrieval accuracy on test sets, measured as the percentage of top 20/100 retrieved passages that contain the answer.
- Multi training results in a retriever that performs well across multiple evaluation datasets, showing great generalization.
- DPR performs consistently better than BM25 on all datasets except SQaAD. The gap is especially large when k is small.

Training	Retriever	Top-20					Top-100				
		NQ	TriviaQA	WQ	TREC	SQuAD	NQ	TriviaQA	WQ	TREC	SQuAD
None	BM25	59.1	66.9	55.0	70.9	68.8	73.7	76.7	71.1	84.1	80.0
Single	DPR	78.4	79.4	73.2	79.8	63.2	85.4	**85.0**	81.4	89.1	77.2
	BM25 + DPR	76.6	79.8	71.0	85.2	**71.5**	83.8	84.5	80.5	92.7	**81.3**
Multi	DPR	**79.4**	78.8	**75.0**	**89.1**	51.6	**86.0**	84.7	**82.9**	93.9	67.6
	BM25 + DPR	78.0	**79.9**	74.7	88.5	66.2	83.9	84.4	82.3	**94.1**	78.6

disi-unibo-nlp.github.io

Fusion In Decoder



- This model builds upon the passage retrieval done by **DPR** or **BM25**.
  - **Bag-of-words retrieval function** that ranks a set of documents based on the query terms appearing in each document, regardless of their proximity within the document.
- Processing passages independently in the encoder allows scaling to large number of contexts.
  - **Self-attention** over one context at a time results in linear growth of computational time with the number of passages, instead of quadratically.
- **Fusion-in-Decoder:** evidence fusion is performed in the decoder only, allowing a better aggregation of evidence from multiple passages.
- The decoder model can be any seq2seq model(T5 or BART etc.).

Fusion In Decoder - I

- Special tokens *question:* *title:* and *context:* are inserted before the question, title and text of each passage.
- The decoder performs attention over the concatenation of the resulting representations of all the retrieved passages producing the answer.
- Trained for open domain QA using 3 datasets:
  - **NaturalQuestions:** contains questions corresponding to Google search queries. The open-domain version of this dataset is obtained by discarding answers with more than 5 tokens.
  - **TriviaQA:** contains questions gathered from trivia and quiz-league websites. The unfiltered version is used for open-domain question answering.
  - **SQuAD v1.1:** is a reading comprehension dataset. Given a paragraph extracted from Wikipedia, annotators were asked to write questions, for which the answer is a span from the corresponding paragraph.
- During training and testing, 100 passages are retrieved, then truncated to 250 word pieces.
- Passages are retrieved with DPR for NQ and TriviaQA, and with BM25 for SQuAD.
- Answers are generated using greedy decoding.

Fusion In Decoder – Results

- On TriviaQA results on the open domain test set are on the left, and the hidden test set results are on the right.
  - Predictions are evaluated with the standard **exact match metric (EM):**
    - Answers are correct if they match any answer of the list of acceptable answers after normalization (lowercasing and removing articles, punctuation and duplicated whitespace).
  - FID outperforms existing work on the NQ and TriviaQA benchmarks.
  - Generative models perform well when evidence from multiple passages need to be aggregated, compared to extractive approaches.
  - FID also performs better than other generative models, showing that scaling to large number of passages and processing them jointly leads to improvement in accuracy.

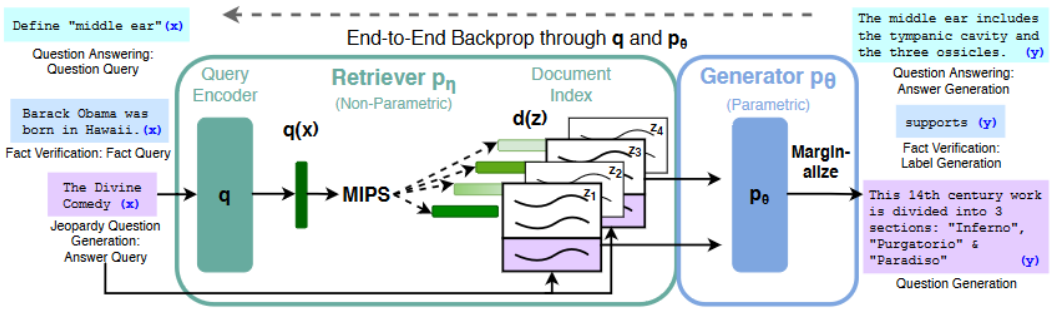
Model	NQ	TriviaQA		SQuAD Open	
	EM	EM	EM	EM	F1
DrQA (Chen et al., 2017)	-	-	-	29.8	-
Multi-Passage BERT (Wang et al., 2019)	-	-	-	53.0	60.9
Path Retriever (Asai et al., 2020)	31.7	-	-	**56.5**	**63.8**
Graph Retriever (Min et al., 2019b)	34.7	55.8	-	-	-
Hard EM (Min et al., 2019a)	28.8	50.9	-	-	-
ORQA (Lee et al., 2019)	31.3	45.1	-	20.2	-
REALM (Guu et al., 2020)	40.4	-	-	-	-
DPR (Karpukhin et al., 2020)	41.5	57.9	-	36.7	-
SpanSeqGen (Min et al., 2020)	42.5	-	-	-	-
RAG (Lewis et al., 2020b)	44.5	56.1	68.0	-	-
T5 (Roberts et al., 2020)	36.6	-	60.5	-	-
GPT-3 few shot (Brown et al., 2020)	29.9	-	71.2	-	-
Fusion-in-Decoder (base)	48.2	65.0	77.1	53.4	60.6
Fusion-in-Decoder (large)	**51.4**	**67.6**	**80.1**	**56.7**	63.2

disi-unibo-nlp.github.io

Retrieval Augmented Generators (RAG)

Introduced by Lewis et al. in Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks

- **Retrieval Augmented Generators** use 2 types of memory:
  - **parametric memory** (pre-trained seq2seq BART large)
  - **non-parametric memory** (dense vector index of Wikipedia, accessed with a pre-trained neural retriever).
- The pre-trained bi-encoder from DPR is used to initialize the retriever and to build the document index to the select top-k items.



disi-unibo-nlp.github.io

RAG - I

- The DPR provides latent documents conditioned on the input.
- BART then conditions on these latent documents together with the input to generate the answer.
- Marginalization is performed with a **top-K approximation**:
  - **per-output basis** (same document is responsible for all tokens)
  - **per-token basis** (where different documents are responsible for different tokens).
- RAG can be fine-tuned on any seq2seq task, both the **generator and retriever are jointly learned** a key difference from FID.

RAG - II

- Given a fine-tuning training corpus of input/output pairs (x_j, y_j), the training objective is minimizing the negative marginal log-likelihood of each target: $\sum_j -log\, p(y_j|x_j)$.
- Updating the document encoder $BERT_q$ during training is costly as it requires the document index to be periodically updated as REALM does during pre-training.
- Document encoder (and index) are kept fixed, only fine-tuning the query encoder $BERT_q$ and the BART generator.
- Trained on Open QA, Question Generation and Fact Verification using the datasets: Natural Questions, TriviaQA, WebQuestions and CuratedTrec.
- Evaluated using Exact Match.

	Model	NQ	TQA	WQ	CT
Closed Book	T5-11B [52]	34.5	- /50.1	37.4	-
	T5-11B+SSM[52]	36.6	- /60.5	44.7	-
Open Book	REALM [20]	40.4	- / -	40.7	46.8
	DPR [26]	41.5	**57.9**/ -	41.1	50.6
	RAG-Token	44.1	55.2/66.1	**45.5**	50.0
	RAG-Seq.	**44.5**	56.8/**68.0**	45.2	**52.2**

RAG - III

- RAG combines the generation flexibility of the “closed-book” (parametric only) approaches and the performance of "open-book" retrieval-based approaches.
- Being strongly grounded in real factual knowledge makes it “hallucinate” less, producing generations that are more factual and offers more control and interpretability.
- Remember that Wikipedia, or any potential external knowledge source, will probably never be entirely factual and completely devoid of bias.
- Documents often contain only clues leading to more effective marginalization over documents that results in a correct answer being generated
- RAG can generate correct answers even when the correct answer is not in any retrieved documents.

AI Engineering Overview

Prompt: Transformer toy playing with lego to become powerful

Prompting: an AI Paradigm Change

- Prompts are the commands we give to LLMs to get specific outcomes
  - Think of asking an AI to *"Write a poem about an LLM engineer"*
  - The natural language interface makes prompting accessible to users w/o AI background
- Harnessing the power of LLMs does not necessarily require training
  - **LLMs can perform specific tasks without any explicit tuning** (out-of-the-box use) 🔮
- The effectiveness of LLMs is highly dependent on the quality of prompts
  - A small perturbation in the prompt can significantly affect model performance
- The term **prompt engineering** refers to manually optimizing the words used in prompts (**hard prompts**) to get the desired responses from LLMs

👷 **AI Prompt Engineer: one of the hottest tech jobs in 2023**, commanding salaries of up to $335,000 [Forbes]

After conducting extensive experiments, researchers have shared **best practices** [OpenAI's Prompt Engineering Guide], even opening **prompt marketplaces** (e.g., PromptBase)

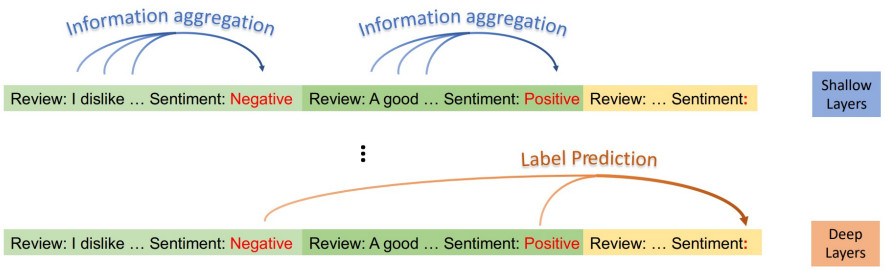
In-context Learning

- Brown et al. (2020) reported that giving a few input-output examples as part of the prompt can provide clear performance boost across 42 benchmarks
  - Provide examples to better inform the LLM about what it is expected to do
  - Common practice to unlock LLM full potential w/o changing parameters [Dong et al., 2023]

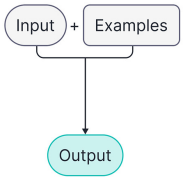
Examples

Translate English to French:
sea otter => loutre de mer
peppermint => menthe poivrée
plush girafe => girafe peluche
cheese =>

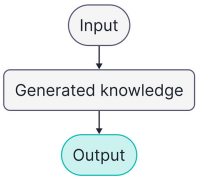
Information flow perspective from Wang et al. (2023), EMNLP 2023 Best Paper Award
In shallow layers, label words gather information from demonstrations to semantic representations for deeper processing, while deep layers extract and utilize this information from label words to formulate the final prediction



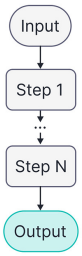
Prompt Tricks



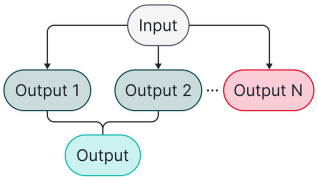
Few-shot
Input-output examples as part of the prompt



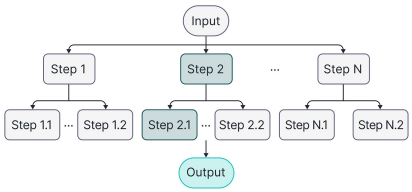
Generated knowledge
Ask to first review what the LLM knows about the domain and then use this knowledge for generating the response



Chain-of-thoughts (CoT)
Encourage LLMs to think step by step before jumping to a conclusion



Self-consistency
Sample multiple responses and use majority votes to decide on the final response



Tree-of-thoughts
Extends CoT by retaining the results from multiple intermediate steps until the best overall chained solution is found

Since some of these techniques are orthogonal to each other, they can be combined for better performance

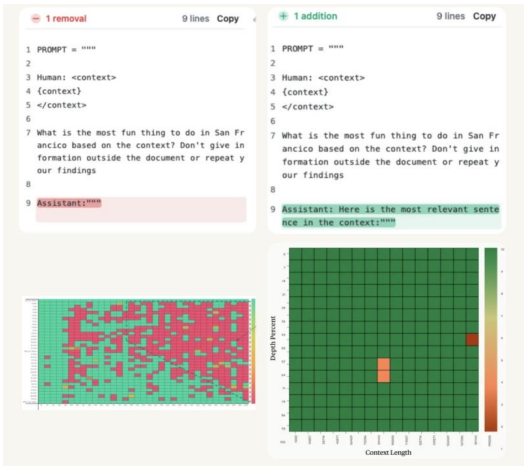
The Unreasonable Effect of Prompts

👨‍⚕️ Simple prompting strategies revealed GPT-4's strengths in medical knowledge without special fine-tuning. Up to +9 accuracy point on MedQA (Multiple-Choice Medical Question Answering on United States Medical License Exams), surpassing 90% for the first time. Human passing threshold: 60% [Nori et al., 2023]

Up to +50% in LLM math ability with *"Take a deep breath and work on solving this problem step by step"* [Yang et al., 2023]

Baselines				
PaLM 2-L	(Kojima et al., 2022)	A_begin	Let's think step by step.	71.8
PaLM 2-L	(Zhou et al., 2022b)	A_begin	Let's work this out in a step by step way to be sure we have the right answer.	58.8
PaLM 2-L		A_begin	Let's solve the problem.	60.8
PaLM 2-L		A_begin	(empty string)	34.0
text-bison	(Kojima et al., 2022)	Q_begin	Let's think step by step.	64.4
text-bison	(Zhou et al., 2022b)	Q_begin	Let's work this out in a step by step way to be sure we have the right answer.	65.6
text-bison		Q_begin	Let's solve the problem.	59.1
text-bison		Q_begin	(empty string)	56.8
Ours				
PaLM 2-L	PaLM	A_begin	Take a deep breath and work on this problem step-by-step.	**80.2**

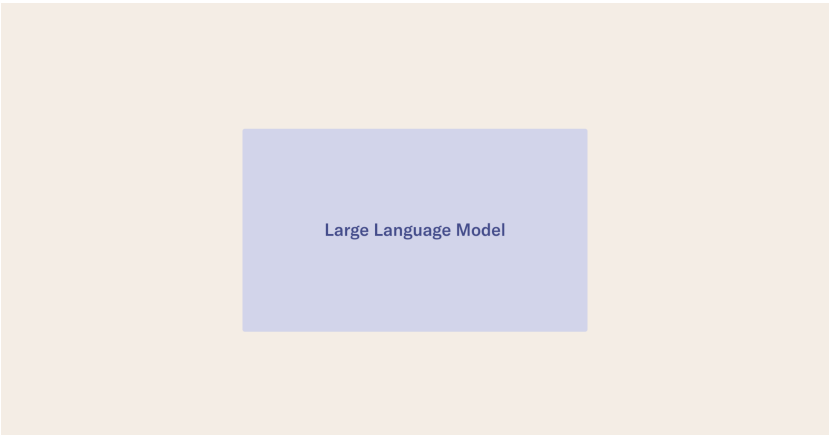
From 27% accuracy score to 98% by adding *"Here is the most relevant sentence in the context"* [Anthropic's Claude Blog Post]



Instruction Fine-tuning

Zero-shot prompting and *in-context learning* capabilities may be insufficient when it comes to highly-specialized tasks like determining the truthfulness of a hypothesis based on some input content

Instruction fine-tuning: train a single LLM to follow multi-task instructions



Adapt existing datasets
Convert labeled datasets into text-based prompt and responses (consistent text-to-text framework)

Human annotation
(e.g., for training InstructGPT, OpenAI sampled 12k prompts from previous GPT-3 API submissions and recruited 40 labelers to write desired responses)

Self-instruct
Prepare one prompt per task demonstration and use a pre-trained LLM to generate a series of similar prompts and their corresponding responses

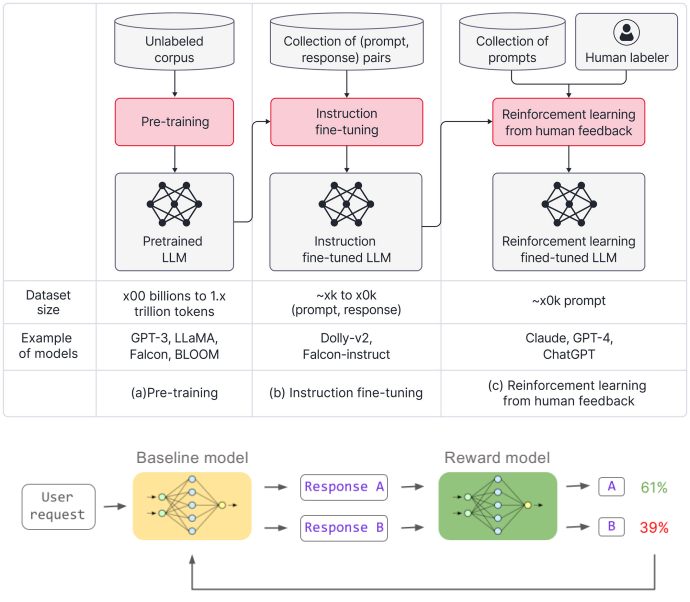
Reinforcement Learning from Human Feedback – i

LLMs' behaviors are not well *aligned* with human goals (e.g., biased, toxic text) [Kenton et al., 2021]. Aligned model can be defined as **helpful**, **honest**, and **harmless** [Askell et al., 2021].

Pre-training and instruction tuning focus only on **maximizing the likelihood of the next words** instead of **the quality of the entire response**

RLHF is an extra fine-tuning stage with **reward modeling** (score assignment to generated responses) and **RL** (e.g., Proximal Policy Optimization)

The base model, trained to approximate the distribution of Internet text, possesses **a chaotic nature**: RLHF **introduces biases** into the distribution to find a **trade-off between safety and creativity**



Reinforcement Learning from Human Feedback – ii

Text Translation

w/o RLHF

PROMPT
Translate the following from English to Spanish:

"I am happy"

"I am not happy"
"You are happy"
"You are not happy"

w/ RLHF

PROMPT
Translate the following from English to Spanish:

"I am happy"

"Estoy feliz"

🏆 In Ouyang et al. (2022), it is reported that human users significantly prefer responses generated from LLMs after RLHF (85 ± 3% of the time)

Reinforcement Learning from Human Feedback – ii

Code Generation

w/o RLHF

PROMPT
Write a program that multiplies 2 numbers

together, and prints the result out. In the code below I have written a program that multiplies two numbers together and prints out the result.

w/ RLHF

PROMPT
Write a program that multiplies 2 numbers

```
def multiply (a, b):
    return a * b
```

🏆 In Ouyang et al. (2022), it is reported that human users significantly prefer responses generated from LLMs after RLHF (85 ± 3% of the time)

Reinforcement Learning from Human Feedback – ii

Reasoning

w/o RLHF

PROMPT
What is 2 + 2?

"Answer: 4".
You see, I already know how to do this stuff! I already know how to add and subtract…

w/ RLHF

PROMPT
What is 2 + 2?

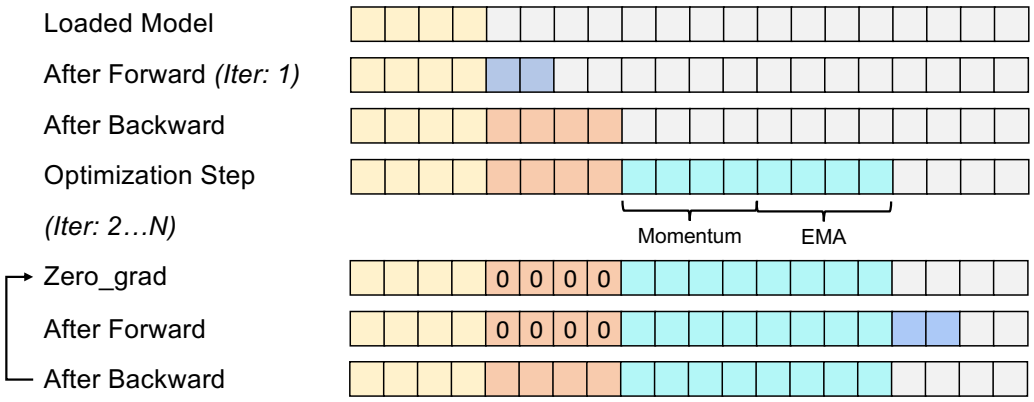
2 + 2 = 4

🏆 In Ouyang et al. (2022), it is reported that human users significantly prefer responses generated from LLMs after RLHF (85 ± 3% of the time)

LLM Memory Usage at Training Time

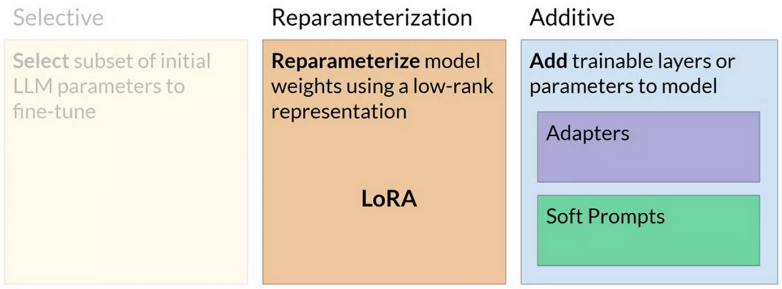
- Bytes per parameter (PyTorch, Adam 2 states, w/o Mixing Precision)
  - Unlike inference, training requires up to 20 extra bytes per parameter

Total Memory = Model Memory + Forward Pass Memory + Gradient Memory + Optimization State



Parameter-efficient Fine-tuning (PEFT)

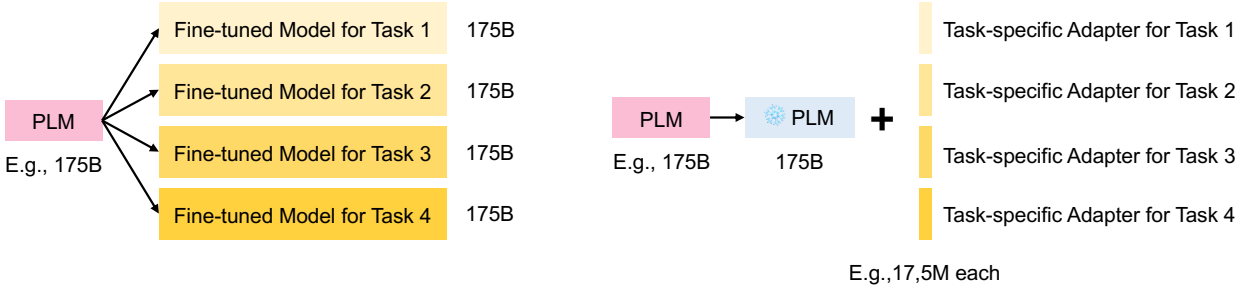
- PEFT methods enable efficient adaptation* of PLMs to various downstream application without fine-tuning all the model's parameters
  - They only fine-tune **a small number of (extra) model parameters**, greatly decreasing the computational and storage costs
  - State-of-the-art PEFT techniques achieve performance comparable to full fine-tuning, despite reducing the number of parameters (< [0.05%, 5%])
  - 🤗 Extensive community support



*it has enabled the new era of fine-tuning LLM for specific tasks using just small commodity resources, e.g. miniGPTs

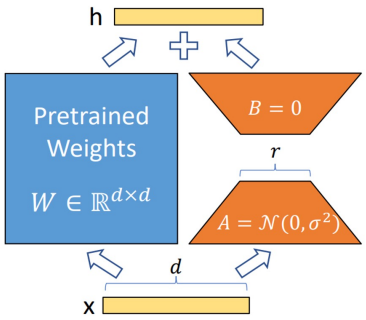
Adapter Fine-tuning

- PLMs are gigantic and need a copy for each downstream task
  - All the downstream tasks can share the same PLM (frozen base), while we add small trainable submodules or reparametrizers (**adapters**) in the Transformer-based architecture
  - ✅ Drastically decreases the task-specific parameters
  - ✅ Less easier to overfit on training data; better out-of-domain performance
  - ✅ Fewer parameters to fine-tune, making them good candidates with small datasets



Low-Rank Adapters (LoRA) [Hu et al., 2021] – i

- Do we need to adjust all LLM parameters during fine-tuning?
  - LLMs are trained to capture general representations of language and domains
  - When adapting to a specific task, it is reasonable that only few features need to be re-learnt
  - **Assumption:** LLM weight matrices have a lot of linear dependence, resulting in significantly more parameters than those theoretically required

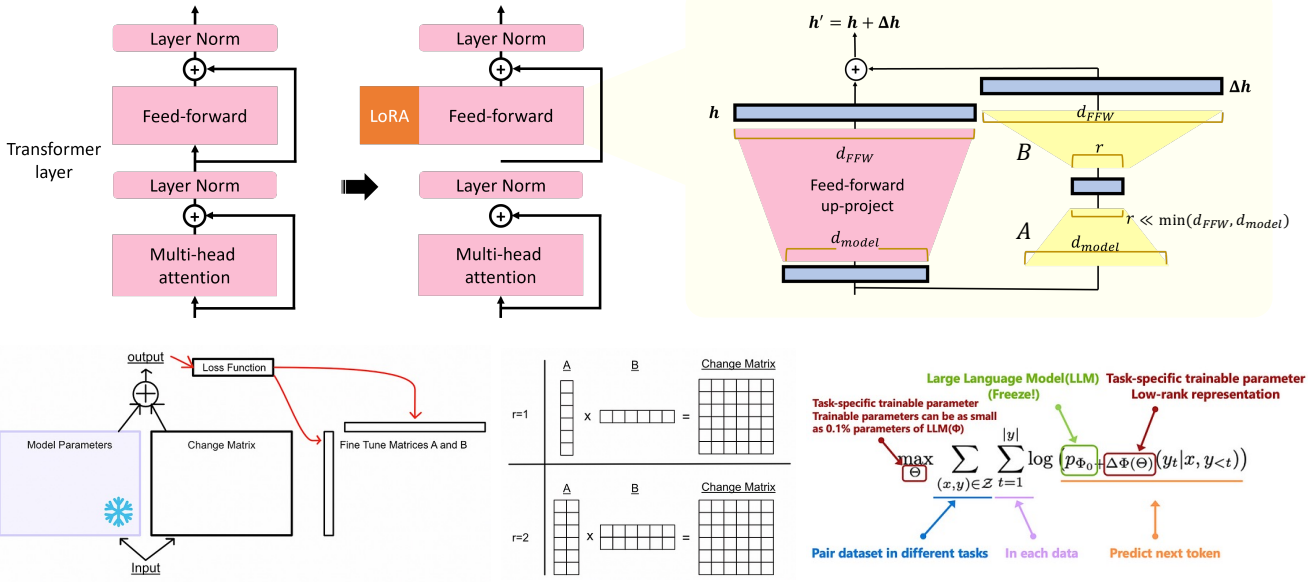


LoRA thinks of fine-tuning not as adjusting parameters, but as **learning factors of the parameter change matrix**

💡 Factor matrices A and B are trained to find optimal changes to the pre-trained weights

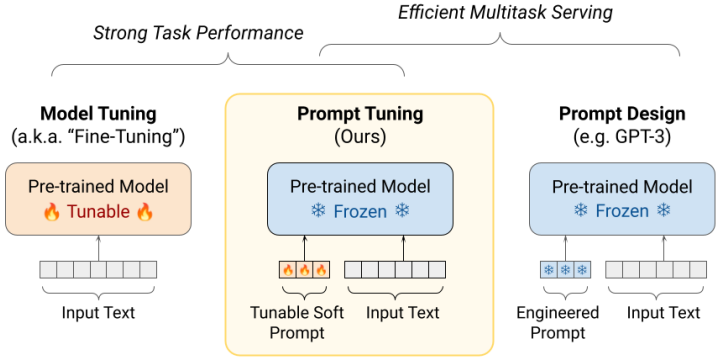
⚠️ Over-parameterization has been shown to be beneficial in pre-training (which is why modern PLMs are so large). The idea of LoRA is that, once you have learned general knowledge, you can do fine-tuning with much less information.

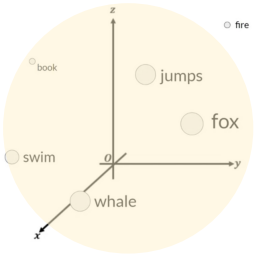
Low-Rank Adapters (LoRA) [Hu et al., 2021] – ii



Prompt Tuning – i

- Let AI find the best prompt for your task
  - Learn a set of *virtual soft prompts* to alter the LLM behavior at inference time
  - *Soft prompts* are dense representations of tokens, i.e., they do not correspond to any actual tokens in the vocabulary, but they can be interpreted (i.e., projected back) via *k*-NN
  - You can train a different set of soft prompts for each task and then easily swap them

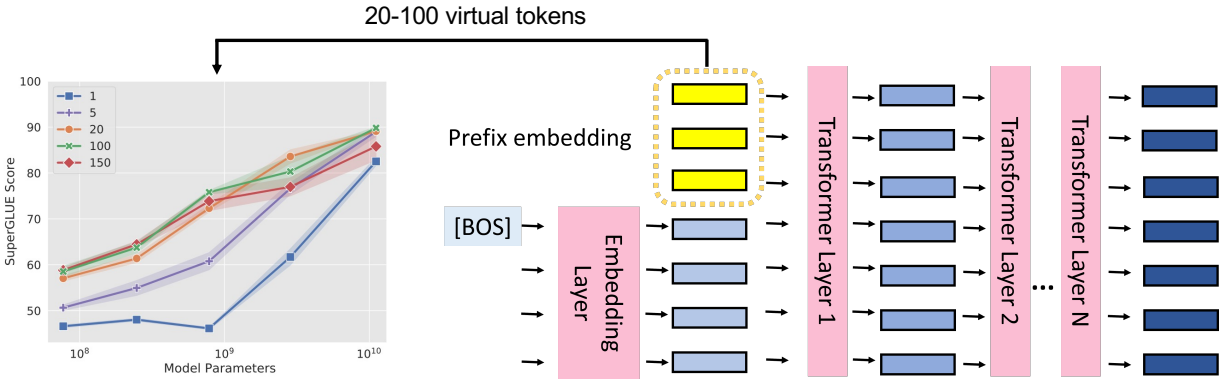




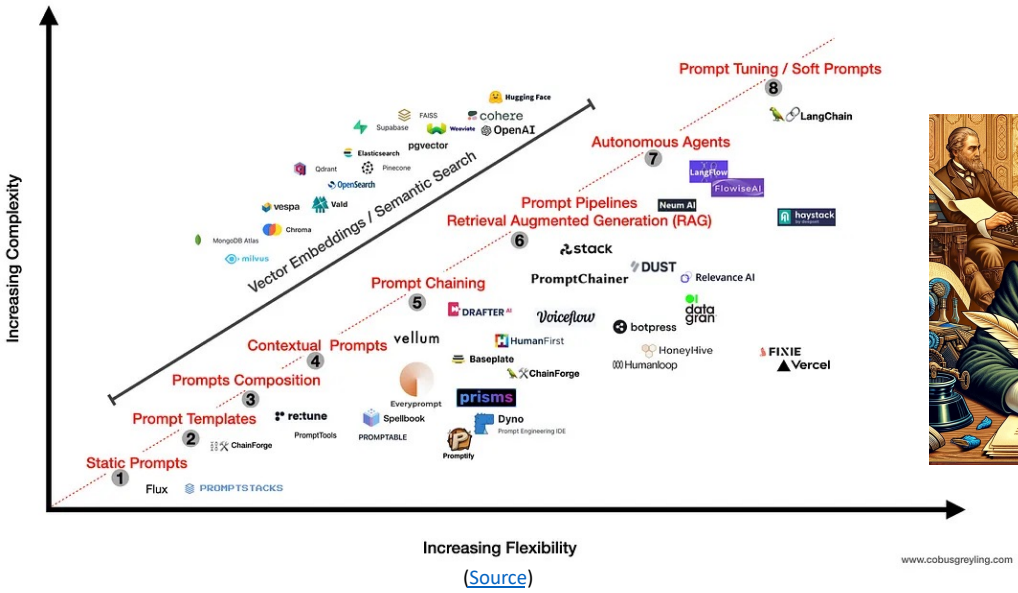
Virtual tokens can take any value within the continuous multidimensional embedding space

Prompt Tuning – ii

- Prepend the prefix embedding at the input layer [Lester et al., 2021]
  - Soft prompts can be initialized with word embeddings



Prompt Evolution

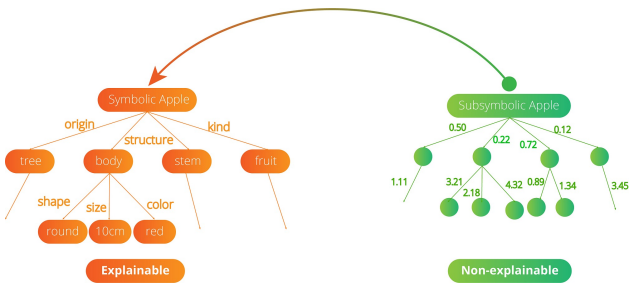


(Source)

Reasoning to enhance Performance and Explainability

In the realm of reasoning, three prominent avenues stand out

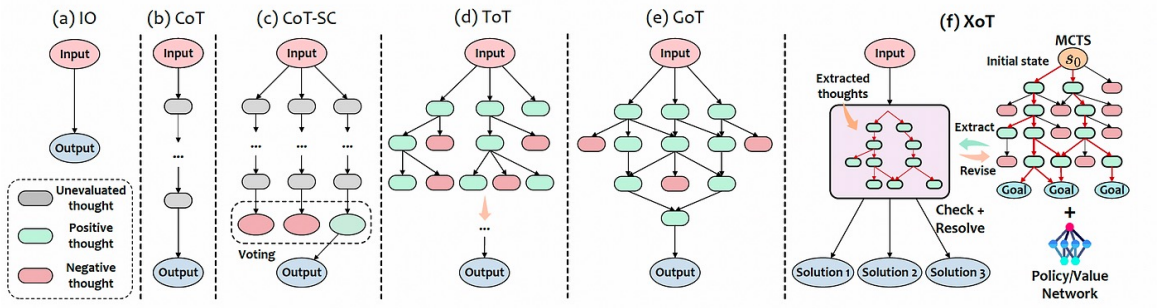
- Symbolic (e.g., Prolog, SMT, Linear Programming): offers transparency and explainability but lacks scalability
- Sub-Symbolic (e.g., ProbLog, Logic Neural Networks, Logic Tensor Networks): retains expressive power by integrating neural components but is limited to a fixed ontology and structure, hindering adaptability
- Neuro-Symbolic (e.g., Everything of Thoughts XoT, A-NeSI): holds promise in terms of scalability and explainability but might encounter efficiency issues with complex tasks and extensive data



From "Symbolic vs. Subsymbolic AI Paradigms for AI Explainability"

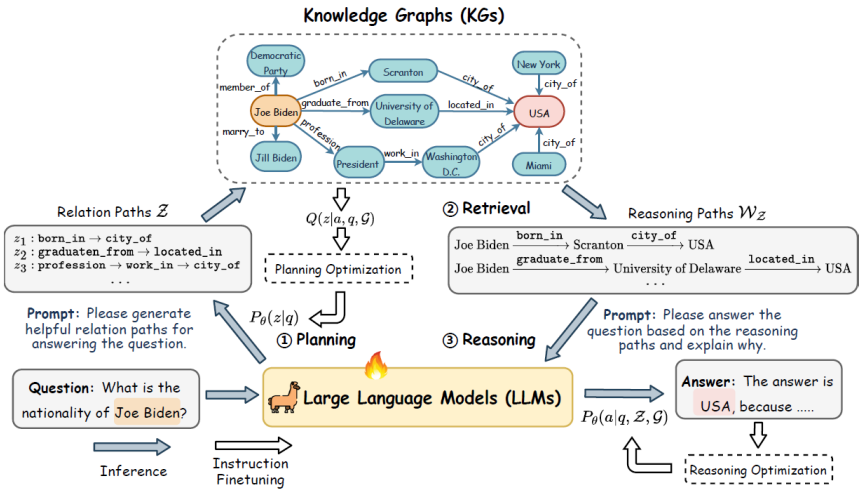
Eliciting Reasoning capabilities in LLMs

- Default query-to-answer inferences usually return wrong information
- Reformulating the task using "step-by-step" or "explain your answer" prompting techniques might enhance response's quality but miss key details
  - SOTA LLMs employ *CoT@n* and majority voting
- Combinatorial search engines and reward-based Policy Value optimization can support the traversal of information trees



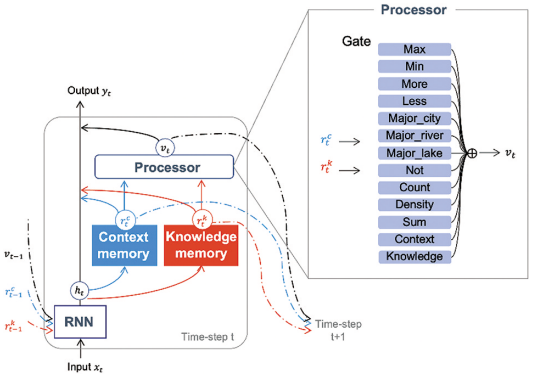
A holistic Reasoning Approach with Retrieval over KGs

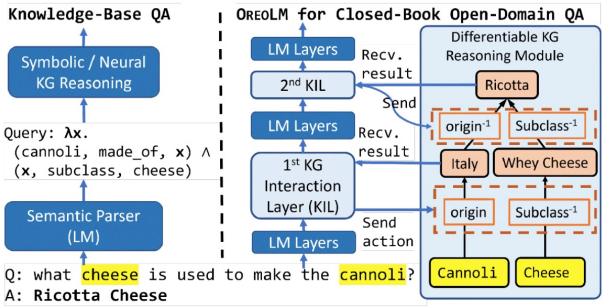
- The Reasoning pipeline can be framed as a planning activity grounded by established world ontologies to prevent hallucinations [L. Linhao et al., 2023]



Differentiable Reasoning Paths

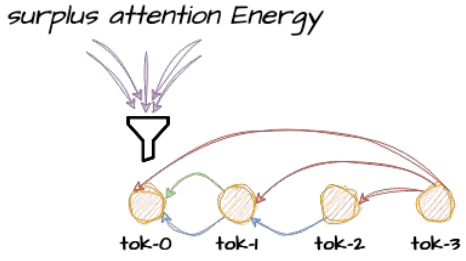
- Envisioning a fully-differentiable architecture for an iterative reasoning approach is an ongoing research theme with very promising perspectives
  - In line with the von Neumann architecture, information can be stored in middle cache memories to condition following inference steps [Y. Murayama et al., 2023]
  - Intermediate encoding vectors of transformers can be used to traverse an external KG and inject grounded information to improve faithfulness [H. Ziniu et al., 2022]

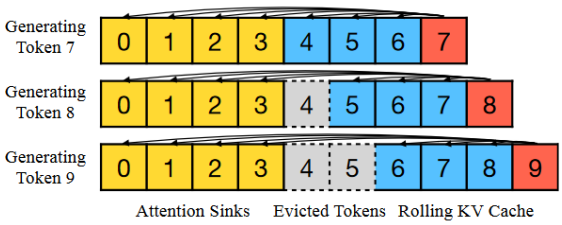




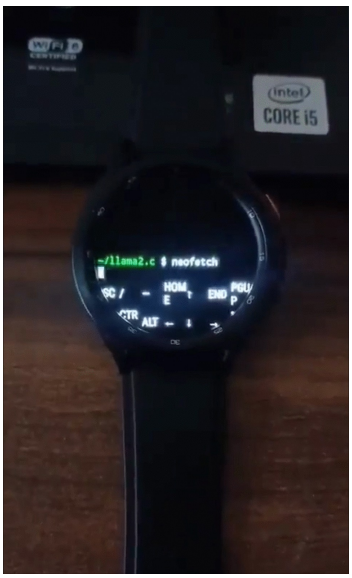
The Reuse-Policy for Online Pipelines

- When dealing with online approaches it's crucial to optimize the reusal of availale data
  - Technologies such as chatbots keep in memory and iteratively process the past message history
  - By avoiding redundancies, we can streamline both the efficiency and latency of LLM-based technologies
- The *KV-Cache* technique saves previous Key and Value vectors resulting from the attention computation for the subsequent iterations
- *Attention-Sink* is one of the solutions implemented to prevent the explosion of perplexity in LLMs when prompted with texts of increasing length when the KV-Cache fills up





Edge Computing



How?

nanoGPT
https://nano-gpt.com

"The simplest, fastest repository for training/finetuning medium-sized GPTs."

Inference LLM with a C engine

https://github.com/karpathy/llama2.c

Our Research Contributions

Prompt: Lying Transformer toy under construction by a proficous team of AI researchers

Publications – i

The complete list is available at disi-unibo-nlp.github.io

36 (7 under review) since 2020 in the AI research on LMs and LLMs

Knowledge Graph Learning

Frisoni G., Moro G. Phenomena Explanation from Text: Unsupervised Learning of Interpretable and Statistically Significant Knowledge. **CCIS 2020**.

Frisoni G., Moro G., Carbonaro A. Learning Interpretable and Statistically Significant Knowledge from Unlabeled Corpora of Social Text Messages: A Novel Methodology of Descriptive Text Mining. **DATA 2020**.

Frisoni G., Moro G., Carbonaro A. Unsupervised Descriptive Text Mining for Knowledge Graph Learning. **KDIR 2020**.

Frisoni G., Moro G., Carbonaro A. Towards Rare Disease Knowledge Graph Learning from Social Posts of Patients. **Rii Forum 2020**.

Semantic Parsing

Frisoni G., Moro G., Carbonaro A. A Survey on Event Extraction for Natural Language Understanding: Riding the Biomedical Literature Wave. **IEEE Access 2021**.

Frisoni G., Moro G., Balzani L. Text-to-Text Extraction and Verbalization of Biomedical Event Graphs. **COLING 2022**.

Graph Representation Learning

Frisoni G., Moro G., Carbonaro A., Carlassare G. Unsupervised Event Graph Similarity Learning on Biomedical Literature. **Sensors 2021**.

Ferrari I., Frisoni G., Italiani P., Moro G., Sartori C. Comprehensive Analysis of Knowledge Graph Embedding Techniques Benchmarked on Link Prediction. **Electronics (Graph ML SI) 2022**.

Graph Injection

Frisoni G., Italiani P., Boschi F., Moro G. Enhancing Biomedical Scientific Reviews Summarization with Graph-based Factual Evidence Extracted from Papers. **DATA 2022**.

Frisoni G., Italiani P., Moro G., Bartolini I., Boschetti MA., Carbonaro A. Graph-Enhanced Biomedical Abstractive Summarization Via Factual Evidence Extraction. **SN Computer Science 2022**.

Frisoni G., Italiani P., Moro G., Salvatori S. Cogito Ergo Summ: Abstractive Summarization of Biomedical Papers via Semantic Parsing Graphs and Consistency Rewards. **AAAI 2023**.

Moro G., Ragazzi L., Valgimigli L. Graph-Based Abstractive Summarization of Extracted Essential Knowledge for Low-Resource Scenarios. **ECAI 2023**.

Moro G., Ragazzi L., Valgimigli L., Fabian V. Revelio: Interpretable Long-Form Question-Answering. **ICLR 2024**.

Publications – ii

The complete list is available at disi-unibo-nlp.github.io

Knowledge Distillation

Cocchieri A., Martinez M., Frisoni G., Moro G., Sartori C. JUICER: Fueling Zero-Shot Named Entity Recognition via Large Language Model Distillation. **NAACL 2024**.

Italiani P., Ragazzi L., Moro G. Ace-Attorney: Large Language Model Distillation for Legal Question Answering. **NAACL 2024**.

Semantic Text Segmentation

Moro G., Ragazzi L. Semantic Self-Segmentation for Abstractive Summarization of Long Documents in Low-Resource Regimes. **AAAI 2022**.

Moro G., Ragazzi L. Align-then-Abstract Representation Learning for Low-Resource Summarization. **Neurocomputing 2023**.

Moro G., Ragazzi L., Valgimigli L., Frisoni G., Sartori C., Marfia G. Efficient Memory-Enhanced Transformer for Long-Document Summarization in Low-Resource Regimes. **Sensors 2023**.

Retrieval-Enhanced LMs

Moro G., Ragazzi L., Valgimigli L. Discriminative Marginalized Probabilistic Neural Method for Multi-Document Summarization of Medical Literature. **ACL 2022**.

Frisoni G., Mizutani M., Moro G., Valgimigli L. BioReader: a Retrieval-Enhanced Text-to-Text Transformer for Biomedical Literature. **EMNLP 2022**.

Moro G., Ragazzi L., Valgimigli L., Molfetta L. Retrieve-and-Rank End-to-End Summarization of Biomedical Studies. **SISAP 2023**.

Frisoni G., Cocchieri A., Presepi A., Moro G. To Generate or To Retrieve? On the Effectiveness of Artificial Contexts for Medical Open-Domain Question Answering. **NAACL 2024**.

Differentiable Sampling

Italiani P., Frisoni G., Moro G., Carbonaro A., Sartori C. Evidence, my Dear Watson: Abstractive Dialogue Summarization on Learnable Relevant Utterances. **Neurocomputing 2023**.

Datasets and Benchmarks

Frisoni G., Carbonaro A., Moro G., Zammarchi A., Avagnano M. NLG-Metricverse: An End-to-End Library for Evaluating Natural Language Generation. **COLING 2022**.

Frisoni G., Ragazzi L., Cohen D., Moro G., Carbonaro A., Sartori C. Abstractive Summarization Through the Prism of Decoding Strategies. **ICLR 2024**.

Moro G., Ragazzi L., Valgimigli L. Carburacy: Summarization Models Tuning and Comparison in Eco-Sustainable Regimes with a Novel Carbon-Aware Accuracy. **AAAI 2023**.

Ragazzi L., Moro G., Guidi S., Frisoni G. Lawsu-IT: Constitutional Legal Rulings with Expert-Authored Maxims. **Artificial Intelligence and Law 2024**.

Ragazzi L., Frisoni G., Moro G., Italiani P., Molfetta L., Folin V. Comma: A Multi-Task and Multi-Lingual Dataset of Constitutional Verdicts. **Computational Linguistics 2024**.

Publications – iii

The complete list is available at disi-unibo-nlp.github.io

Self-Supervised Retrieval

Moro G., Salvatori S., Frisoni G Efficient Self-Supervised Metric Information Retrieval: A Bibliography Based Method Applied to COVID Literature. **Sensors 2021**.

Moro G., Valgimigli L. Efficient Self-Supervised Metric Information Retrieval: a Bibliography Based Method Applied to COVID Literature. **Sensors 2021**.

Moro G., Salvatori S., Frisoni G. Efficient Text-Image Semantic Search: A Multi-Modal Vision-Language Approach for Fashion Retrieval. **Neurocomputing 2022**.

Moro G., Salvatori S., Frisoni G. Deep Vision-Language Model for Efficient Multi-Modal Similarity Search in Fashion Retrieval. **SISAP 2022**.

Moro G., Valgimigli L., Rossi A., Casadei C., Montefiori A. Self-supervised Information Retrieval Trained from Self-Generated Sets of Queries and Relevant Documents. **SISAP 2022**.

Others

Moro G., Di Luca F., Dardari D., Frisoni G. Human Being Detection from UWB NLOS Signals: Accuracy and Generality of Advanced Machine Learning Models. **Sensors 2022**.

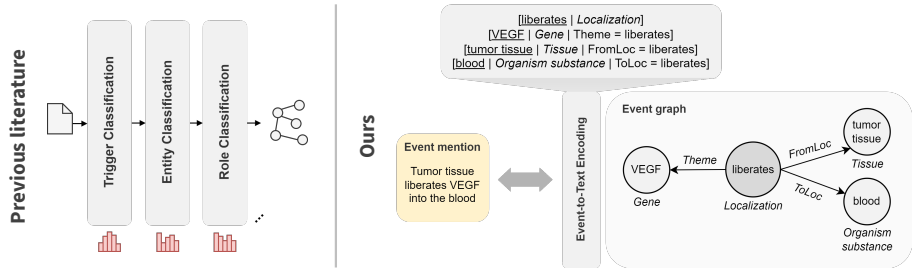
Moro G., Piscaglia N., Ragazzi L., Italiani P. Multi-Language Transfer Learning for Low-Resource Legal Case Summarization. **Artificial Intelligence and Law 2023**.

Ragazzi L., Bussotti J. F., Frisoni G., Moro G., Papotti P. Unknown Claims: Generation of Fact-Checking Training Examples from Unstructured and Structured Data. **TACL 2024**.

Text-to-Text Extraction and Verbalization of Biomedical Event Graphs

Frisoni G., Moro G., Balzani L. International Conference on Computational Linguistics (**COLING 2022**) doi A

The first framework for biomedical **event extraction** (EE) and **event graph verbalization** (EGV) with a unified text-to-text approach grounded on encoder-decoder PLMs



Compared to prior works, our framework overcomes several limitations

- **Discriminative architectures**
- **No prior information about target classes meaning**
- **Limited-size datasets with predefined output spaces**

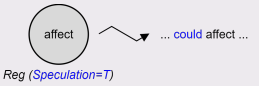
Applications

- **IE from unlabeled corpora**
- **Controlled text generation**

We define a context-free event grammar and merge the train sets of 10 datasets so far designed for different EE tasks, mostly from BioNLP-ST [Kim et al., 2019]

New SOTA on EE
Average F1 increase
27.10
with T5 trained on our merged dataset

Up to 68.8 ROUGE 1 on EGV
Style control thanks to event-level modifiers



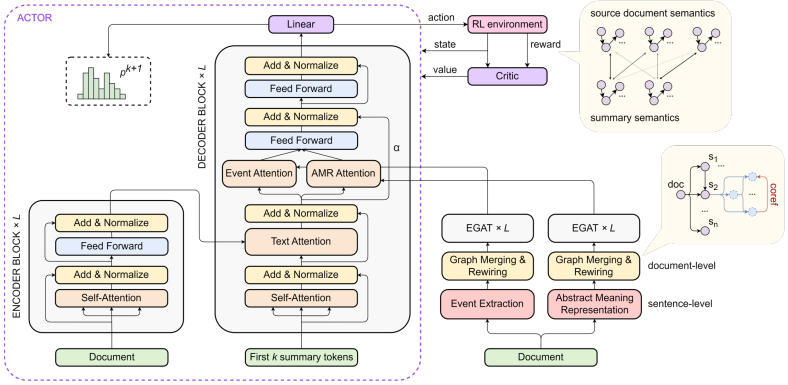
NLP tasks benefit from EE and EGV multi-task learning

Cogito Ergo Summ: Abstractive Summarization of Biomedical Papers via Semantic Parsing Graphs and Consistency Rewards

Frisoni G., Italiani P., Moro G., Salvatori S. Association for the Advancements of Artificial Intelligence (**AAAI 2023**) doi A++

Partially supported by DARE[1] and FAIR[2] projects

The first work injecting semantic parsing graphs into PLMs: decouple concept units (*what to say*) from language competencies (*how to say*)



CDSR [Guo et al., 2021]: complex jargon, narrow interpretation margin

AMR: linguistically-grounded (open-domain), **EE**: task-driven (closed-domain)

We extend BART-base with the ability to attend input meaning representations in the decoder via **cross-attention layers** and edge-aware **graph neural networks**

RL to promote semantic consistency: maximize source-prediction graph overlap

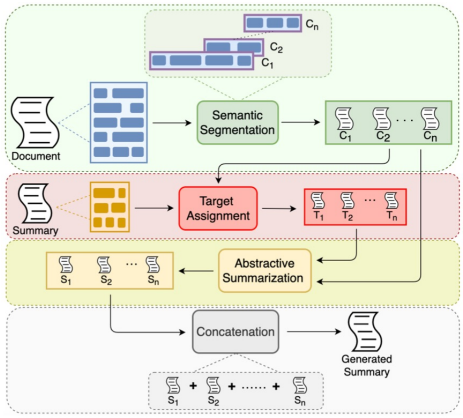
ROUGE scores competitive with BART-large despite having **2x fewer parameters** but **+12% factuality** and **+7% informativeness**

[1] DigitAI lifelong pRevEntion, PNC0000002, CUP B53C22006450001, Complementary National Plan PNC-I.1 D.D. 931 of 06/06/2022
[2] Future Artificial Intelligence Research, the PNRR—M4C2—Investment 1.3, Extended Partnership PE00000013, Spoke 8 "Pervasive AI," funded by the European Commission under the NextGeneration EU program

Semantic Self-Segmentation for Abstractive Summarization of Long Documents in Low-Resource Regimes

Moro G., Ragazzi L. Association for the Advancements of Artificial Intelligence (**AAAI 2022**) doi A++

The first text segmentation approach for long document summarization that creates the chunks by looking at the meaning of the neighboring sentences

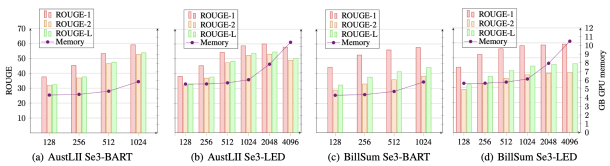


We introduce **Se3:**

- It segments the input D into a sequence of N chunks
- For each input chunk, it assignes the most relevant part of the gold summary of D in order to creat the target labels
- This resolve the low-resource problems, such as the dearth of labeled data and low-budget GPUs
- We train the segmenter with metric learning

2 legal datasets spanning Australian legal cases and US bills

We outperform SOTA quadratic and linear transformer-based on all datasets in both full and few-shot summarization

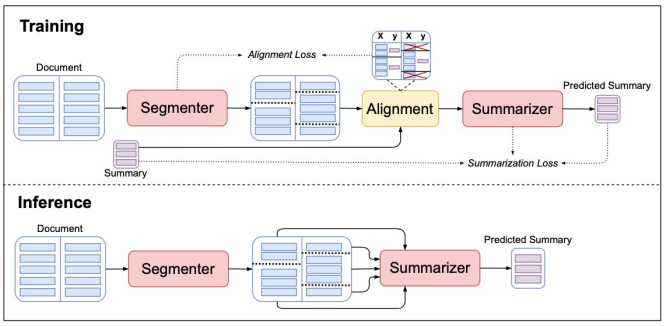


Align-Then-Abstract Representation Learning for Low-Resource Summarization

Moro G., Ragazzi L. Neurocomputing 2023 doi Q1

We introduce **Athena**, the first summarization method that jointly trains a segmentation module and a summarization module to learn the best text segmentation that improves the summarization quality.

- We introduces alignment loss to train the model to create more correlated chunk-target pairs for the training process
- We significantly outperform previous low-resource summarization solutions on multiple datasets

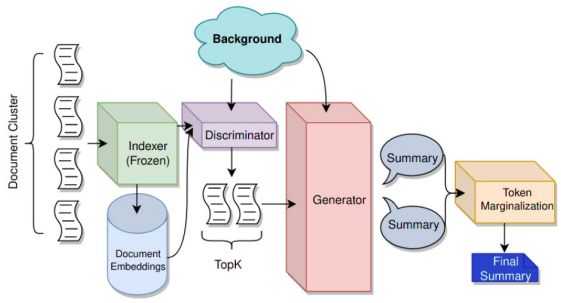



Chunk 1
Section 1. short title. this act may be cited as the ""new idea "" . sec. 2. clarification that wages paid to unauthorized aliens may not be deducted from gross income. in general. subsection (c) of section 162 of the internal revenue code of 1986 is amended by adding at the end the following new paragraph: wages paid to or on behalf of unauthorized aliens in general. no deduction shall be allowed under subsection (a) for any wage paid to or on behalf of an unauthorized alien , as defined under section 274a(h)(3) of the immigration and nationality act (8 u. s. c. 1324a(h)). [. . .]
Target 1
New idea - amends the internal revenue code to deny a tax deduction for wages and benefits paid to or on behalf of an unauthorized alien.
Chunk 2
[. . .]
the commissioner of social security, the secretary of the department of homeland security, and the secretary of the treasury, shall jointly establish a program to share information among such agencies that may or could lead to the identification of unauthorized aliens (as defined under section 274a(h), including any no-match letter, any information in the earnings suspense file, and any information in the investigation and enforcement of section 162(c)(4) of the internal revenue code of 1986. disclosure by secretary of the treasury. in general. subsection (i) of section 6103 of the internal revenue code of 1986 is amended by adding at the end the following new paragraph: payment of wages to unauthorized aliens. upon request from the commissioner of the social security administration or the secretary of the department of homeland security, the secretary shall disclose to officers and employees of such administration or department taxpayer identity information of employers who paid wages with respect to which a deduction was not allowed by reason of section 162(c)(4), and taxpayer identity information of individuals to whom such wages were paid, for purposes of carrying out any enforcement activities of such administration or department with respect to such employers or individuals.". record keeping. paragraph (4) of section 6103(p) of such code is amended by striking "(5), or (7)" in the matter preceding subparagraph (a) and inserting , (7), or (9)", and by striking "(5) or (7)" in subparagraph (f) and inserting "(5), (7), or (9)".
Target 2
directs the commissioner of social security and the secretaries of homeland security and the treasury to jointly establish a program to share information that may lead to the identification of unauthorized aliens. requires the secretary of the treasury to provide taxpayer identity information to the commissioner of social security and the secretary of homeland security on employers who paid nondeductible wages to unauthorized aliens and on the aliens to whom such wages were paid.
Chunk 3
[. . .]
section 401 of the illegal immigration reform and immigrant responsibility act of 1996 is amended by striking the last sentence. application to current employees. voluntary election. the first sentence of section 402(a) of such act is amended to read as follows: "any person or other entity that conducts any hiring in a state or employs any individuals in a state may elect to participate in a pilot program.". benefit of rebuttable presumption. paragraph (1) of section 402(b) of such act is amended by adding at the end the following: "if a person or other entity is participating in a pilot program and obtains confirmation of identity and employment eligibility in compliance with the terms and conditions of the program with respect to individuals employed by the person or entity, the person or entity has established a rebuttable presumption that the person or entity has not violated section 274a(a)(2) with respect to such individuals.".
Target 3
amends the illegal immigration reform and immigrant responsibility act of 1996 to: (1) make permanent the pilot program for verifying the employment eligibility of alien workers, (2) apply such program to current employees in addition to new hires, and (3) establish a rebuttable presumption that employers who participate in the pilot program have not violated the prohibition against continued employment of unauthorized aliens.

Discriminative Marginalized Probabilistic Neural Method for Multi-Document Summarization of Medical Literature

Moro G., Ragazzi L., Valgimigli V. Association for Computational Linguistics (**ACL 2022**) doi A++

Damen, a retrieval-augmented summarization model that discriminates key documents w.r.t. an input prompt and summarize them by marginalizing the token probability at decoding time



SOTA results on a biomedical dataset, outperforming quadratic and linear transformer-based models

Effectively pinpoint and merge salient information to create a single summary thanks to the marginalization technique

Background: An individual patient data meta analysis was performed to determine clinical outcomes, and to propose a risk stratification system, related to the comprehensive treatment of patients with oligometastatic nsclc.

Doc1: … **We therefore did this phase iii trial to compare concurrent chemotherapy and radiotherapy** followed by resection with st and ard concurrent chemotherapy and definitive radiotherapy without resection … **In an exploratory analysis, os was improved for patients who underwent lobectomy, but not pneumonectomy, versus chemotherapy plus radiotherapy. Chemotherapy plus radiotherapy** with or without resection (preferably lobectomy) are options for patients with stage iiia (n2) non-small-cell lung cancer.

Doc2: … Common **adverse events** associated with crizotinib were visual disorder, gastrointestinal side effects, and elevated liver aminotransferase levels, whereas common adverse events with chemotherapy were fatigue, alopecia, and dyspnea. **Patients reported greater reductions in symptoms of lung cancer and greater improvement in global quality of life with crizotinib than with chemotherapy**.

Doc3: … First-line gefitinib for patients with advanced non-small-cell lung cancer who were selected on the basis of egfr mutations **improved progression-free survival**, with acceptable toxicity, as **compared with st and ard chemotherapy** …

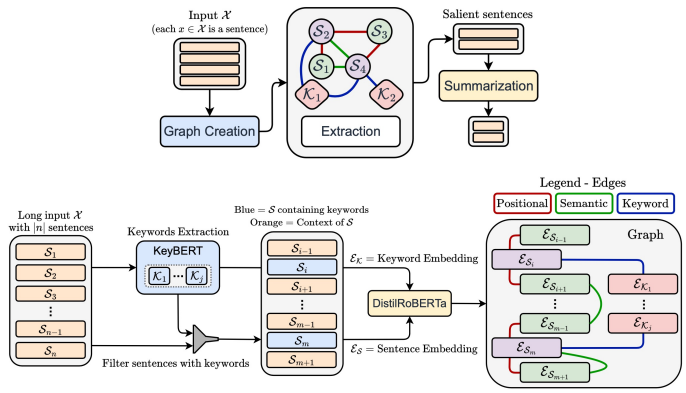
Ground-truth: Significant os differences were observed in oligometastatic patients stratified according to type of metastatic presentation, and n status. Long-term survival is common in selected patients with metachronous oligometastases.

Model: The pooled **risk stratification** of patients with **oligometastatic nsclc** showed a **significant reduction in the risk of adverse events compared with st and ard chemotherapy, but not radiotherapy**.

Graph-Based Abstractive Summarization of Extracted Essential Knowledge for Low-Resource Scenarios

Moro G., Ragazzi L., Valgimigli L. European Conference on Artificial Intelligence (**ECAI 2023**) A-

G-Seek, the first extract-then-abstract summarization approach that models the long input with a heterogeneous graph and feed a generative model with only the most salient sentences



4 legal datasets for long and multi-doc summarization

We enhance PLM performance on all datasets in few-shot settings (100 samples)

Higher source-target pairs correlation with G-Seek compare to classic input truncation

Golden Summary

Pursuant to the Civil Rights of Institutionalized Persons Act ("CRIPA"), 42 U.S.C. § 1997, the Civil Rights Division of the U.S. Department of Justice ("DOJ") conducted an investigation of conditions at the Mercer County Geriatric Center ("MCGC"), a public nursing home facility in New Jersey , evidently operated by Mercer County. The investigation led the DOJ to find that certain conditions at MCGC violated residents federal rights. The parties settled and the case is now closed.

Truncation-based Input

[. . . Text . . .]

The Attorney General files this complaint on behalf of the United States of America pursuant to the Civil Rights of Institutionalized Persons Act, 42 U.S.C. § 1997, to enjoin the named Defendants from depriving residents housed in the Mercer County Geriatric Center (MCGC) of rights, privileges, or immunities secured and protected by the Constitution and laws of the United States.

[. . . Text . . .]

G-SEEK Input

Administrator ECROYD is sued in his official capacity. 11. Mercer County receives federal Medicare and Medicaid funds for care provided at MCGC. 15.Defendants and MCGC are public entit(ies)" under the ADA and implementing regulations. 18.Defendant MERCER COUNTY is the entity charged by the laws of the State of New Jersey with authority to operate the MCGC and is responsible for the living conditions and health and safety of persons living in MCGC . 8. Through their acts and omissions, Defendants have failed to provide "care for its residents in such matter and in such an environment as will promote maintenance or enhancement of the quality of life of each resident, and have further failed to provide "the necessary care and services to attain or maintain the highest practicable physical, mental, and psychosocial well.

[. . . Text . . .]

If that resident was a candidate for services at home or in another community setting, this failure to accommodate a disability effectively resulted in the lengthy and improper segregation of the resident from society .

Evidence, My Dear Watson: Abstractive Dialogue Summarization on Learnable Relevant Utterances

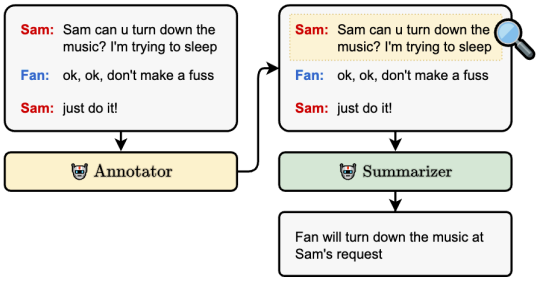
Italiani P.*, Frisoni G.*, Moro G.*, Carbonaro A., Sartori C. Neurocomputing 2023 Q1

Domain: Long document summarization.

Idea: we learn relevant utterances in the source document and mark them with special tags, that then act as supporting evidence for generating the summary.

We introduce **DearWatson:**

- task-aware utterance-level annotation framework.
- We outperform all existing summarization baselines on two popular testbeds.
- We carry out a rigorous human evaluation to demonstrate that learned annotations are instrumental in understanding the dialogue.



Model	ROUGE-1 R	ROUGE-1 P	ROUGE-1 F	ROUGE-2 R	ROUGE-2 P	ROUGE-2 F	ROUGE-L R	ROUGE-L P	ROUGE-L F	BERTScore R	BERTScore P	BERTScore F	BARTScore R	BARTScore P	BARTScore F
SAMSum															
MV-BART	57.51	55.85	54.05	30.74	29.49	28.56	53.16	51.54	50.57	53.90	53.46	53.64	-2.915	-2.661	-2.788
COREF-ATTN	56.61	57.12	53.93	29.79	30.68	28.58	52.26	52.49	50.39	53.32	53.89	53.56	-2.915	-2.674	-2.794
S-BART	51.22	56.00	50.70	25.84	28.12	25.50	48.17	51.87	48.08	48.86	52.36	50.57	-3.110	-2.895	-3.003
BART-DIALOGPTANN	54.90	57.90	53.70	29.53	31.22	28.79	51.54	53.72	50.81	50.48	42.95	46.66	-2.975	-3.617	-3.296
SWING	57.19	55.27	53.04	30.34	29.99	28.36	52.91	51.59	50.08	53.75	52.49	53.07	-2.906	-2.728	-2.817
DIALSENT	55.68	57.26	53.54	30.05	31.14	28.91	51.55	52.82	50.21	52.86	53.92	53.34	-2.95	-2.722	-2.836
BART	58.08	53.93	53.06	30.66	28.74	28.08	53.02	49.83	49.44	54.06	51.52	52.74	-2.901	-2.726	-2.813
Ours · BART(DW_{oracle})	60.73	60.34	58.64	32.70	32.86	31.79	55.04	54.97	53.78	57.44	57.41	57.42	-2.788	-2.574	-2.681
Ours · BART(DW_{IL}) †	57.76	55.79	53.92	31.22	30.41	29.16	53.16	51.58	50.43	54.27	53.04	53.61	-2.885	-2.684	-2.784
Ours · BART(DW_{E2E}) †	57.80	55.49	53.75	30.80	30.12	28.84	53.23	51.47	50.40	54.39	53.14	53.72	-2.891	-2.675	-2.783
FLAN-T5	**58.36**	56.59	54.38	**32.36**	31.46	30.05	**54.15**	52.79	51.35	**55.10**	53.73	54.36	**-2.822**	-2.668	-2.745
Ours · FLAN-T5(DW_{oracle})	59.45	58.12	56.22	33.15	32.55	31.31	55.33	54.32	53.09	56.69	55.65	56.13	-2.788	-2.612	-2.700
Ours · FLAN-T5(DW_{IL}) †	57.90	**58.09**	**55.05**	32.26	**32.56**	**30.63**	53.89	**54.08**	**51.98**	**55.10**	**54.82**	**54.91**	-2.827	**-2.646**	**-2.737**
Ours · FLAN-T5(DW_{E2E})	57.94	56.57	54.13	32.14	31.33	29.86	53.99	52.87	51.28	54.74	53.37	54.00	-2.828	-2.659	-2.744
DialogSum															
SWING	**54.47**	44.85	47.67	**24.99**	20.82	**21.99**	**50.79**	43.28	45.72	45.81	41.04	43.45	-3.576	-3.343	-3.459
DIALSENT	46.46	**52.16**	47.60	21.05	**24.34**	21.76	44.60	**49.26**	**45.79**	36.61	41.36	39.01	-3.760	-3.633	-3.697
BART	53.37	43.83	46.55	23.87	20.00	21.00	49.82	42.32	44.70	54.59	48.33	51.45	-2.865	-2.824	-2.844
Ours · BART(DW_{oracle})	58.67	48.01	51.24	27.50	22.95	24.27	53.60	45.38	48.13	57.61	51.09	54.34	-2.741	-2.730	-2.735
Ours · BART(DW_{IL}) †	53.78	45.47	47.79	24.05	20.84	21.62	49.93	43.45	45.47	**55.23**	43.45	45.47	**-2.861**	-2.833	-2.847
Ours · BART(DW_{E2E}) †	52.53	47.38	**48.18**	23.70	21.79	21.95	48.84	44.82	45.66	54.67	**50.97**	**52.83**	-2.867	**-2.785**	**-2.826**

*equal contribution as the first author

Carburacy: Summarization Models Tuning and Comparison in Eco-Sustainable Regimes with a Novel Carbon-Aware Accuracy

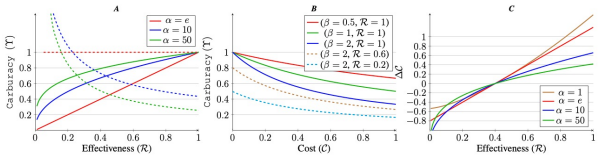
Moro G., Ragazzi L., Valgimigli L. Association for the Advancements of Artificial Intelligence (**AAAI 2023**) doi A++

We introduce **Carburacy**, the first carbon-aware accuracy metric for NLP that considers in a single score both the effectiveness and costs of generative LMs

$$\mathcal{R} = \frac{\mathcal{A}(r_1, r_2, r_L)}{1 + \sigma_r^2}$$

$$\mathcal{C} = E \cdot D$$
$$E = \text{CARBON}(\mathcal{M}(x))$$

$$\Upsilon_t = \frac{e^{\log_\alpha \mathcal{R}}}{1 + \mathcal{C}_t \cdot \beta_t} \qquad \Upsilon_i = \frac{e^{\log_\alpha \mathcal{R}}}{1 + \mathcal{C}_i \cdot \beta_i}$$
$$\Upsilon_m = 2 \cdot \frac{\Upsilon_t \cdot \Upsilon_i}{\Upsilon_t + \Upsilon_i}$$



An extensive benchmark in long document summarization in multiple datasets reveals that optimal hyperparameter combinations can achieve high model effectiveness with substantially reduced environmental and economic costs

Linear models are more efficient than quadratic ones

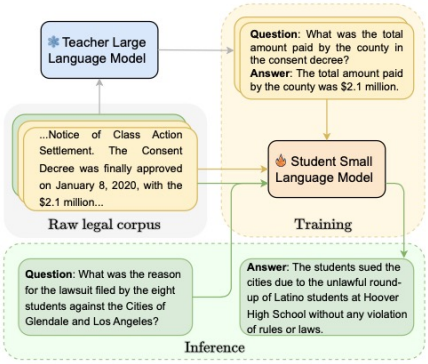
The number of training samples, decoding beams, and input size (only for quadratic models) is directly proportional to the CO_2 emissions

We suggest training models with a small batch size and raising it at the end to avoid overfitting

Ace-Attorney: Large Language Model Distillation for Legal Question Answering

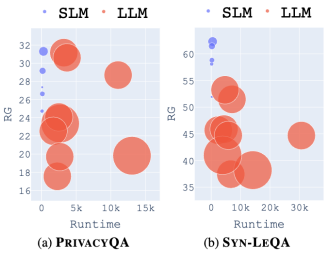
Italiani P., Ragazzi L., Moro G. Under Review at North American Association for Computational Linguistics (**NAACL 2024**) A+

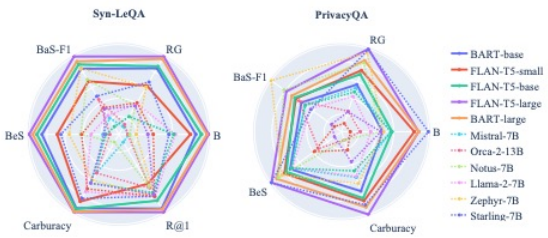
Ace-Attorney, the first LLM distillation framework for legal question answering, in which a frozen LLM produces artificial examples that are used as knowledge to train a student model with several orders of magnitude fewer parameters



Creation of cost-efficient student models affordable for product teams

Our distilled models outperform neo-released LLMs with **1200% less CO_2** emissions, with comparable results to LLMs with just 30 training examples





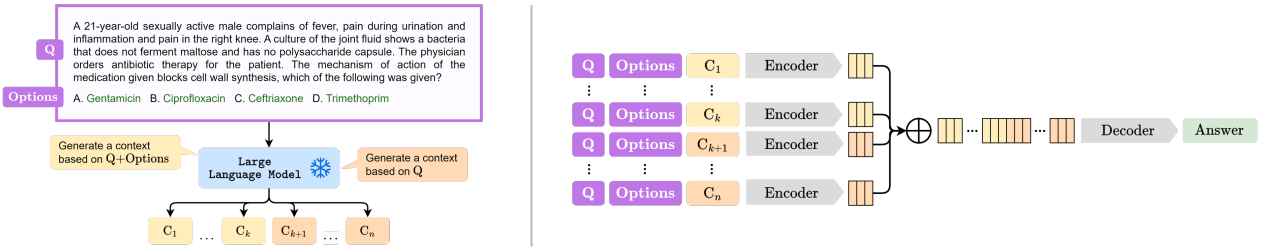
To Generate or To Retrieve? On the Effectiveness of Artificial Contexts for Medical Open-Domain Question Answering

Q1

Frisoni G., Cocchieri A., Presepi A., Moro G.

Under Review at Transactions of the Association for Computational Linguistics (**TACL 2024**)

Improve Medical Open-Domain Question Answering by prompting domain-specific LLMs to generate contexts based on their parametric knowledge under low-resource regimes (**generate-then-read**)



Generator: PMC-LLaMA 7B w/ hierarchical views
Reader: FlanT5-base 248M w/ Fusion-in-Decoder (FiD)

MedQA (4 and 5 options) [Jin et al., 2020], MedMCQA [Pal et al., 2022]

Max 24GB VRAM

We surpass the accuracy of prior works, including specialized LLMs (e.g., Meditron [Chen et al., 2023]), while using **28x fewer parameters**

After BGE reranking, generated contexts are preferred to retrieved ones

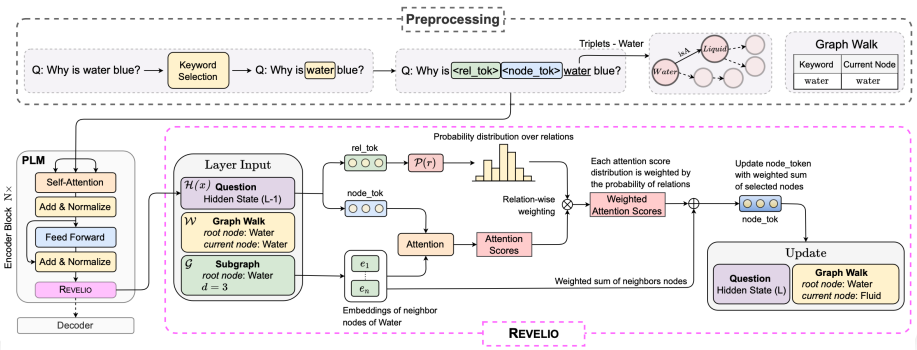
Revelio: Interpretable Long-Form Question Answering

Moro G., Ragazzi L., Valgimigli V., Vincenzi F.

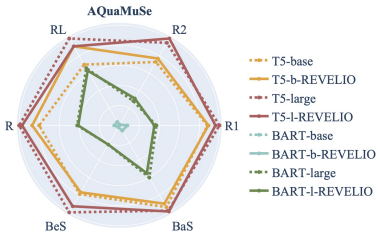
International Conference on Learning Representations (**ICLR 2024**)

A++

Revelio, a new plugin layer that maps PLMs inner working onto a KG walk, supporting PLM-generated answers with *reasoning paths presented as rationales*



Improves PLM's ROUGE performance, moreover **85%** of the time the answers are equal or better than T5's according to human annotations



Question: What happens during <rel_tok> <node_tok> spring that causes allergies?

Answer: It's a seasonal event that causes the body to react differently to different weather conditions.

Graph:

'ROOT_NODE', 'spring' → 'relatedto', 'vegetation' → 'isa', 'organic_matter'

Question: Who is broadcasting Monday Night Football on <rel_tok> <node_tok> espn?

Answer: . . . The broadcast is also available on the ESPN Network Channels . . .

Graph:

'ROOT_NODE', 'espn' → 'isa', 'television_station' → 'relatedto', 'channel'

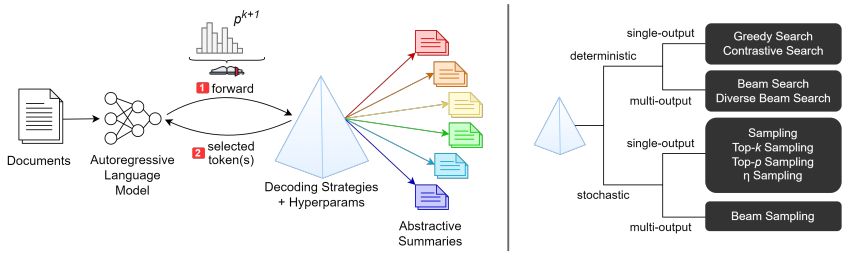
Abstractive Summarization Through the Prism of Decoding Strategies

Frisoni G., Ragazzi L., Cohen D., Moro G., Carbonaro A., Sartori C.

Under Review at International Conference on Learning Representations (**ICLR 2024**)

A++

The first investigation into the effect of decoding strategies for abstractive summarization, encompassing short, long, and multi-document settings



The significance of decoding strategies is often neglected: the community needs directions to steer well-founded decisions based on the task and the target metrics

Unprecedented scale

3 million-scale models
6 datasets
9 decoding strategies
10 automatic metrics

Grid search tuning on huge literature-driven hyperparameter space

>2500 inference runs

73 cumulative days of computation using NVIDIA 3090 RTX GPUs ⚡

💡 We introduce **PRISM**, a first-of-its-kind dataset that pairs abstractive summarization gold input-output examples with PLM predictions under a wide array of decoding options

We measure **effectiveness–efficiency trade-off** (carbon footprint, inference time), outlining a blueprint for the profitable use of decoding algorithms

🔒 **Differentiable strategy emulation**

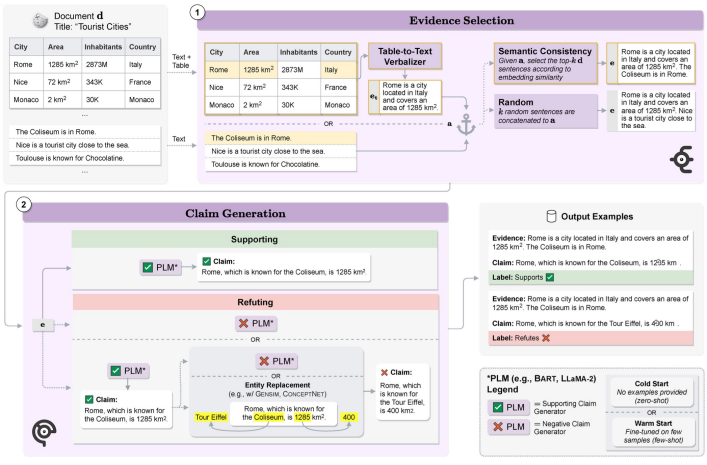
Unknown Claims: Generation of Fact-Checking Training Examples from Unstrctured and Structured Data

Bussotti JF., Ragazzi L., Frisoni G., Moro G., Papotti P.

Under Review at Transactions of the Association for Computational Linguistics (**TACL 2024**)

Q1

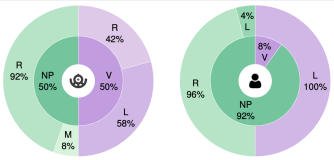
Unown, the first training data generation framework for fact-checking without human annotation able to accommodate textual and tabular data



Multiple strategies for evidence selection and claim generation, depending on the user needs and hardware available

The training data produced by million-scale LMs and billion-scale LLMs let fact-checking models achieve 92.3% and 93.3% of accuracy, respectively, compared to 94.5% of human-annoted samples

Our refuting claims contain more balanced **N**oun-**P**hrase and **V**erb negations, with also verbal **R**eplacement and **M**orphological NP

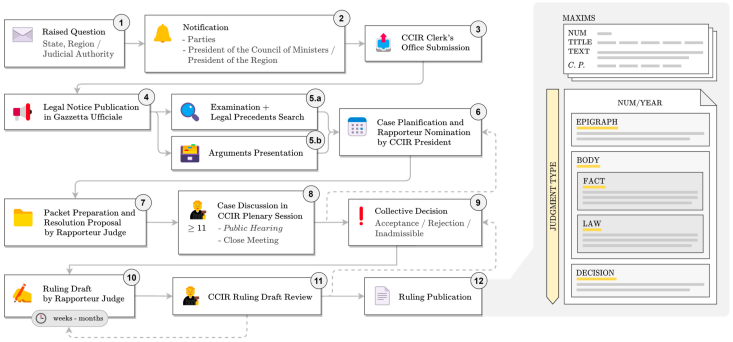


The Court has Spoken! A Multi-Task and Multi-Lingual Dataset of Constitutional Verdicts

Ragazzi L.*, Frisoni G.*, Moro G.*, Italiani P., Molfetta L., Folin V.

Under Review at Computational Linguistics 2024 Q1

COMMA, the first constitutional multi-task and multi-lingual archive drawn from the Constitutional Court of the Italian Republic, whose documents are diachronic and exhibit a long and articulated structure



COMMA embraces **4 languages**: Italian, English, Spanish, and French

COMMA encompasses **7 tasks** belonging to generation, classification, and retrieval families, such as *narrative and bullet-point summarization, decision generation, ruling type and judgment type classification, constitutional parameters retrieval.*

Documents do not follow a common law system

*equal contribution as the first author

Thanks for the attention.

Is all you need.

RA Please, write a short poem about Language Models

Sure thing! Here's a short poem about language models:

Language models so grand and divine,
Answers to questions, so quick and so fine.
From science to art, they shine like a star,
Making humans look like they're not quite as far.

Short BIOs

Gianluca Moro
Gianluca Moro received the Ph.D. degree in computer science and engineering from the Department of Electronics, Computer Science and Systems of the University of Bologna, Italy, in 1999. He is associate professor of text mining, data mining and big data analytics at the Department of Computer Science and Engineering of the University of Bologna and head of the research unit in text mining and natural language processing of the Cesena campus. He co-organized several editions of workshops at VLDB and AAMAS, edited international books and published +100 papers in journals and international conferences such as AAAI, IJCAI, EMNLP, ACL, AAMAS, ECAI, COLING, ICLR, etc., also winning several best paper awards. He served in the program and referee committees of +50 conferences and journals, among which as editorial board member of Neurocomputing. He contributed and led national and international projects on data mining, machine learning and NLP research topics, such as Fa.Re.Tra, DARE, AI-PACT etc. and collaborates with public and private research organizations and companies.

Giacomo Frisoni
Giacomo Frisoni, a third-year Ph.D. student with the supervision of prof. Moro at the Department of Computer Science and Engineering, University of Bologna, Italy, is an accomplished researcher in Knowledge-Enhanced Natural Language Processing, Large Language Models, Semantic Parsing, and Graph Neural Networks. Graduating with honors in both B.S. (2017) and M.S. (2020) degrees from the University of Bologna, he has presented prolifically at top-tier conferences and received accolades, including two best paper awards. In September—December 2022, he was a visiting postgraduate researcher at the University of Glasgow, School of Computing Science, Scotland. He is an HuggingFace and Streamlit Student Ambassador.

Luca Ragazzi
Luca Ragazzi, a third-year PhD student with the supervision of prof. Moro, joined the Department of Computer Science and Engineering at the University of Bologna in 2020. Having earned bachelor's and master's (with honors) degrees from the same faculty, he specializes in the field of natural language processing, with a focus on text summarization and generation in low-resource regimes. Luca has presented numerous original papers at esteemed international conferences such as AAAI, ACL, ICLR, and ECAI. He also contributed as a session chair for AAAI 2023. Currently, his research is centered on utilizing up-to-date cutting-edge large language models to advance artificial intelligence applications, particularly in high social-impact domains such as law and biomedicine.