



AGENDA



Con la collaborazione incondizionata della
Associazione Italiana di Intelligenza Artificiale



- **CONVEGNO ON LINE: Mercoledì 7 Maggio, ore 15.00 – 18.00**
- Metodi dell'Intelligenza Artificiale Generativa: architettura, apprendimento e metodi di prompt engineering
- **OVERVIEW**
 - Foundation Models for Natural Language Processing.
 - Internals of Encoder-Decoder architectures.
 - Chat GPT.
 - Prompt Engineering e Few Shot Learning.
 - Tendenze recenti.
- **USE CASES:**
 - Process management in banking.
 - Information Extraction per la medicina,
 - Modelli di forecasting.

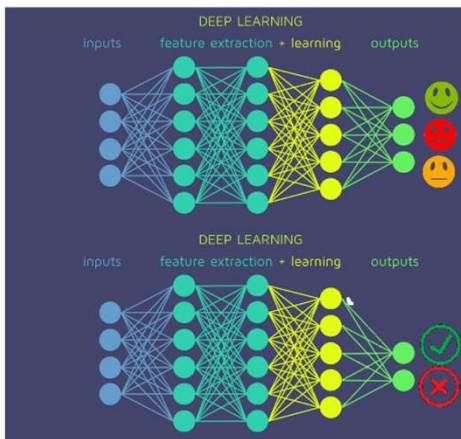
OVERVIEW

- Le Reti Neurali: dai perceptron ai Transformers
 - Il ruolo dei Foundation Models in NLP
 - Internals of Encoder-decoder architectures
- Modelli Generativi e Large Language Models: la famiglia GPT, e chatGPT
 - Chat GPT: principi di funzionamento
- Few-shot Learning
 - 0-shot learning models
 - Prompt Engineering
- Use cases: process management nel sistema bancario, information extraction per la medicina, modelli previsionali (forecasting)

RETI NEURALI (RECAP)

PERCETTRONI E MULTILAYER PERCEPTRONS, CONVOLUTIONAL NEURAL NETWORKS E RECURRENT NETWORKS

MULTILAYER PERCEPTRON FOR CLASSIFICATION TASKS



- Multi-classification MLPs
 - there will be an output unit for each of the labels
 - Ex: n -way topic classification
 - 3 labels in Sentiment Analysis: Positive, Negative, Neutral
- Direct Classification MLPs
 - Binary TASK (True/False)

CONVOLUTIONAL NEURAL NETWORKS (LE CUN, 1998)

- Mainly used for images related tasks
 - image classification
 - face detection
 - etc...
- Learn feature representations
 - by **convolving** over the input
 - with a **filter**, that slides over the input image
- **Compositionality** (local)
 - Each filter composes a local patch of lower-level features into a higher-level representation
- **Location Invariance**
 - the detection of specific patterns is independent of where it occurs

1	0	1
0	1	0
1	0	1

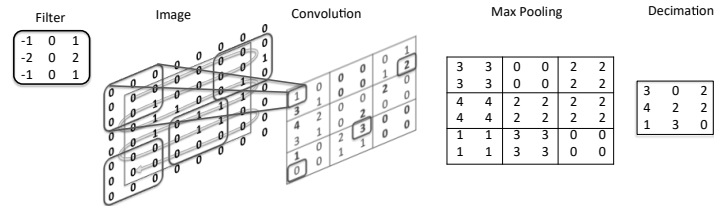
1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

Image

4		

Convolved Feature

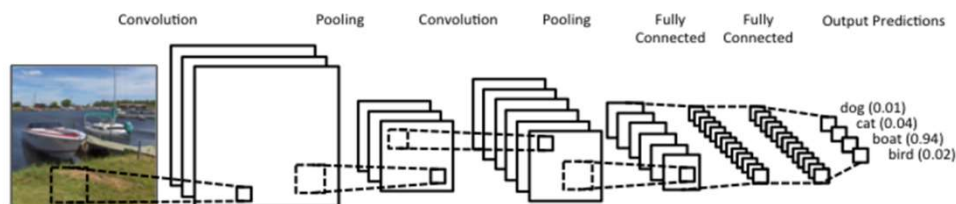
A FURTHER EXAMPLE OF: CONVOLUTION WITH POOLING, AND DECIMATION OPERATIONS



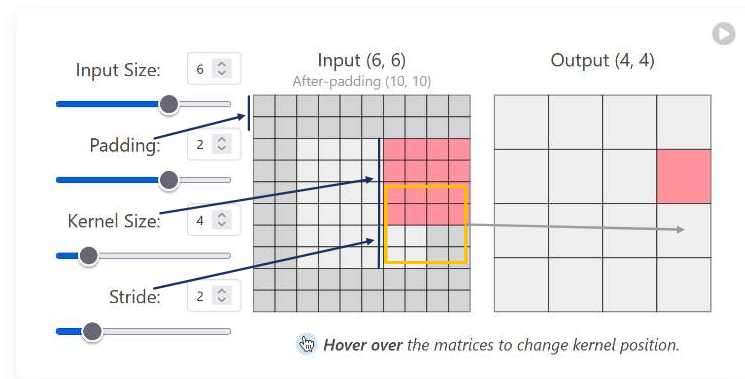
- An image is convolved with a filter; curved rectangular regions in the first large matrix depict a random set of image locations
- Maximum values within small 2x2 regions are indicated in bold in the central matrix
- The results are pooled, using max-pooling then decimated by a factor of two, to yield the final matrix

CONVOLUTIONAL NEURAL NETWORKS

- CNNs automatically learn the parameters of the filters
 - a filter is a matrix of parameters
 - the key aspect is that a filter is adopted for the whole image
- Convolution can be applied in **multiple** layers
 - a layer $l+1$ is computed by convolving over output produced in layer l
 - Pooling is an operation often adopted for taking the most informative features that are learned after a convolution step

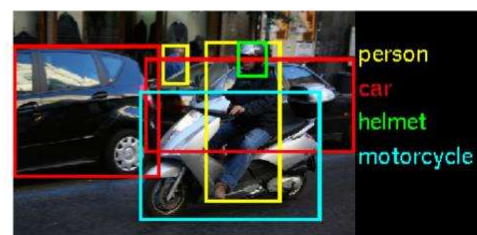
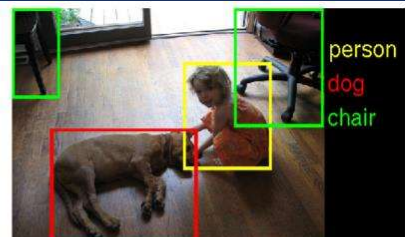
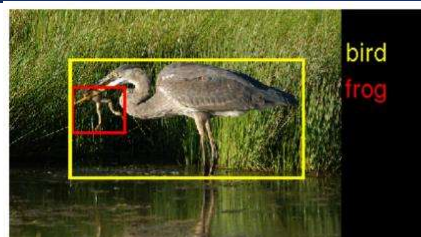


TRAINING A CNN: TERMINOLOGY



$$O = \frac{InputD - KernelD + 2PaddingD}{StrideD} + 1$$

ILSVRC2014 EXAMPLES



WHAT HAS BEEN LEARNT?

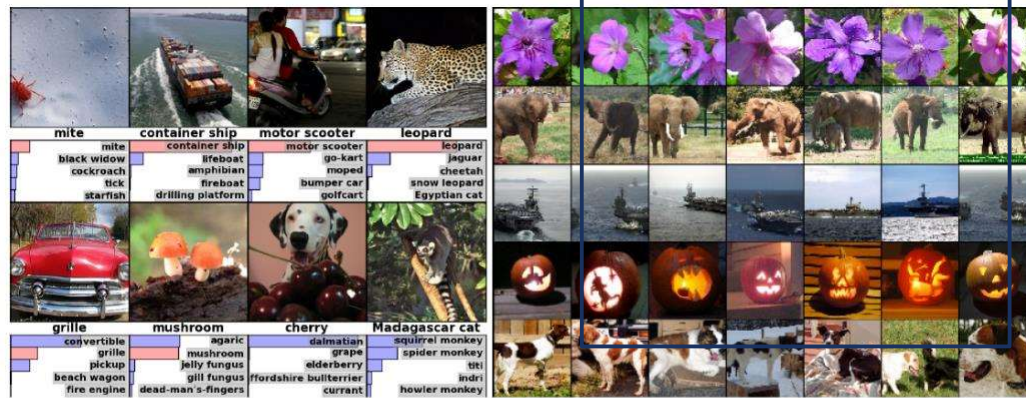


Figure 4: **(Left)** Eight ILSVRC-2010 test images and the five labels considered most probable by our model. The correct label is written under each image, and the probability assigned to the correct label is also shown with a red bar (if it happens to be in the top 5). **(Right)** Five ILSVRC-2010 test images in the first column. The remaining columns show the six training images that produce feature vectors in the last hidden layer with the smallest Euclidean distance from the feature vector for the test image.

RETI NEURALI RICORRENTI (RECAP)

LE RETI RICORRENTI



RECURRENT NEURAL NETWORKS

For example, consider the classical form of a dynamical system:

$$\mathbf{s}^{(t)} = f(\mathbf{s}^{(t-1)}; \boldsymbol{\theta}), \quad (10.1)$$

where $\mathbf{s}^{(t)}$ is called the state of the system.

Equation 10.1 is recurrent because the definition of \mathbf{s} at time t refers back to the same definition at time $t - 1$.

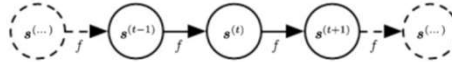
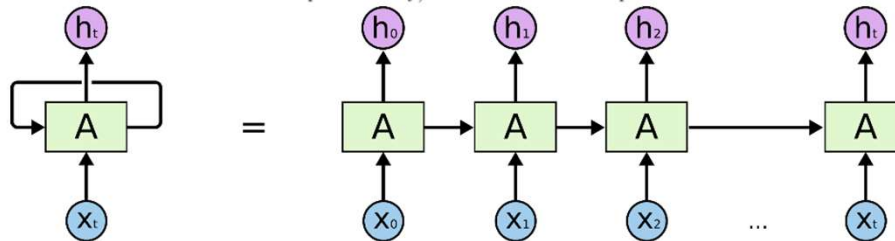


Figure 10.1: The classical dynamical system described by equation 10.1, illustrated as an unfolded computational graph. Each node represents the state at some time t , and the function f maps the state at t to the state at $t + 1$. The same parameters (the same value of $\boldsymbol{\theta}$ used to parametrize f) are used for all time steps.



TYPES OF RNNs

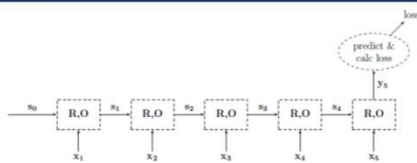


Figure 7: Acceptor RNN Training Graph.

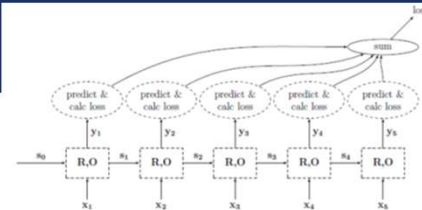


Figure 8: Transducer RNN Training Graph.

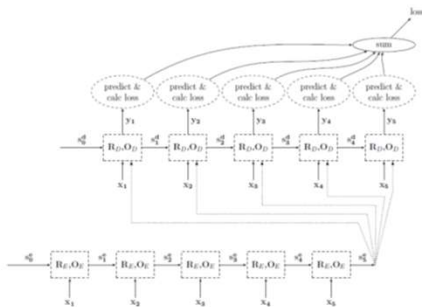


Figure 9: Encoder-Decoder RNN Training Graph.

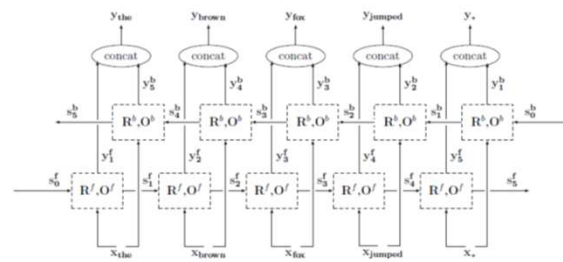


Figure 11: biRNN over the sentence "the brown fox jumped .".

EXAMPLES: LANGUAGE UNDERSTANDING

THE MS COGNITIVE TOOLKIT

Task: Slot tagging with an LSTM

# show	# 0
# flight	# 0
# from	# 0
# burbar	# B-fromloc.city_name
# to	# 0
# st.	# B-to1oc.city_name
# louis	# I-to1oc.city_name
# on	# 0
# monday	# B-depart_date.day_name



<https://learn.microsoft.com/en-us/cognitive-toolkit/Hands-On-Labs-Language-Understanding>

EXAMPLES: LANGUAGE UNDERSTANDING

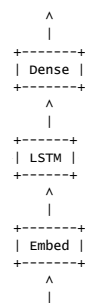
THE MS COGNITIVE TOOLKIT

Task: Slot tagging with an LSTM

```

19 |x 178:1 |# BOS      |y 128:1 |# 0
19 |x 770:1 |# show    |y 128:1 |# 0
19 |x 429:1 |# flights |y 128:1 |# 0
19 |x 444:1 |# from    |y 128:1 |# 0
19 |x 272:1 |# burbank |y 48:1  |# B-fromloc.city_name
19 |x 851:1 |# to      |y 128:1 |# 0
19 |x 789:1 |# st.     |y 78:1  |# B-to1oc.city_name
19 |x 564:1 |# louis   |y 125:1 |# I-to1oc.city_name
19 |x 654:1 |# on      |y 128:1 |# 0
19 |x 601:1 |# monday  |y 26:1  |# B-depart_date.day_name
19 |x 179:1 |# EOS      |y 128:1 |# 0

```



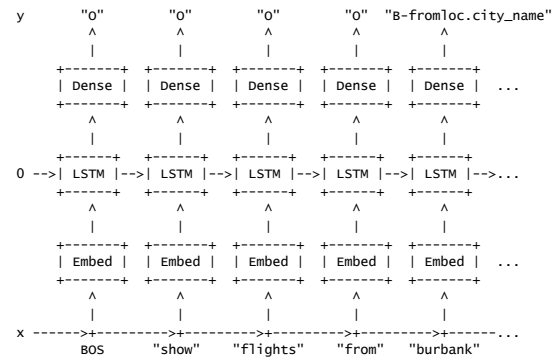
EXAMPLES: LANGUAGE UNDERSTANDING

THE MS COGNITIVE TOOLKIT

Task: Slot tagging with an LSTM

```

19 |x 178:1 |# BOS      |y 128:1 |# 0
19 |x 770:1 |# show    |y 128:1 |# 0
19 |x 429:1 |# flights |y 128:1 |# 0
19 |x 444:1 |# from     |y 128:1 |# 0
19 |x 272:1 |# burbank  |y 48:1  |# B-fromloc.city_name
19 |x 851:1 |# to       |y 128:1 |# 0
19 |x 789:1 |# st.      |y 78:1  |# B-toloc.city_name
19 |x 564:1 |# louis     |y 125:1 |# I-toloc.city_name
19 |x 654:1 |# on       |y 128:1 |# 0
19 |x 601:1 |# monday   |y 26:1  |# B-depart_date.day_name
19 |x 179:1 |# EOS      |y 128:1 |# 0
  
```



MODELLI FONDAZIONALI PER IL NLP

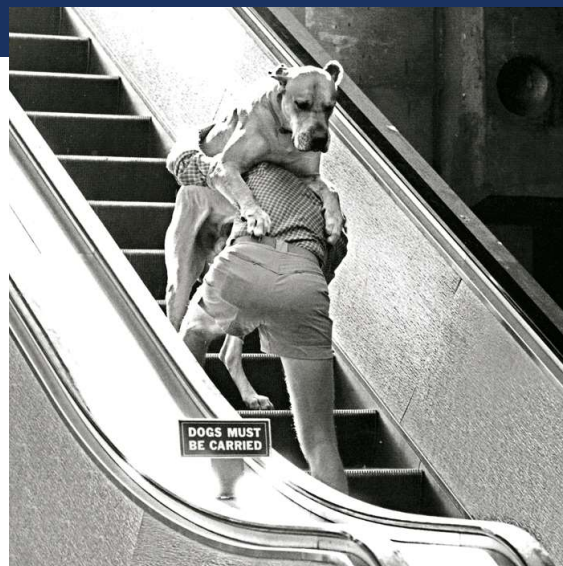
NATURAL LANGUAGE UNDERSTANDING, PROBABILISTIC LANGUAGE MODELS, TRASFORMERS



NATURAL LANGUAGE PROCESSING: AT THE HEART OF GENERATIVE AI SYSTEMS

- Syntax, Semantics and Pragmatics in Artificial Intelligent Agents
- Language Modeling:
 - Statistical approaches
 - Neural approaches to NL semantics
 - Neural Probabilistic Language Models
- Encoder-Decoder architectures

NATURAL LANGUAGE & AMBIGUITY



AMBIGUITY: AN EXAMPLE

- *"Dogs must be carried on this escalator"*
can be consistently interpreted in a number of ways:



- *All dogs should have a chance to go on this wonderful escalator ride*
- *This escalator is for dog-holders only*
- *You can't carry your pet on the other escalators*
- *When riding with a pet, carry it*

THE NLP CHAIN: LEVELS OF LINGUISTIC ANALYSIS

- Given an **valid utterance** such as

John, I am freezing

Pragmatics: what does it do?

- vs.

I, John, freezing am

Semantics: what does it mean?

Syntax: what is grammatical?

ANALOGY WITH ARTIFICIAL LANGUAGES

Syntax: no compiler errors

Semantics: no implementation bugs

Pragmatics: implemented the right algorithm

Different **syntax**, same **semantics** (5):

$$2 + 3 \Leftrightarrow 3 + 2$$

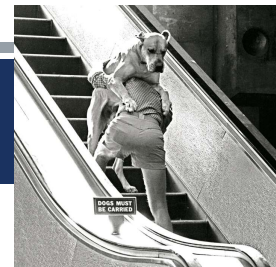
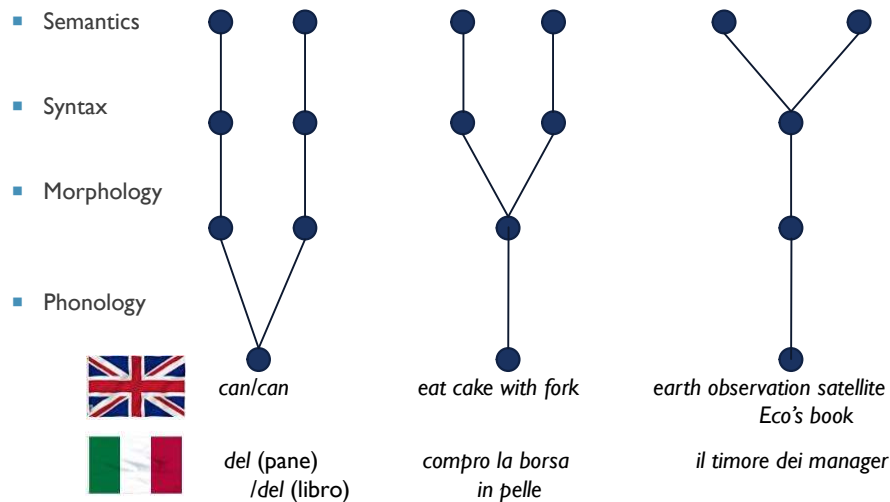
Same **syntax**, different **semantics** (1 and 1.5):

$$3 / 2 \text{ (Python 2.7)} \not\Leftrightarrow 3 / 2 \text{ (Python 3)}$$

Good **semantics**, bad **pragmatics**:

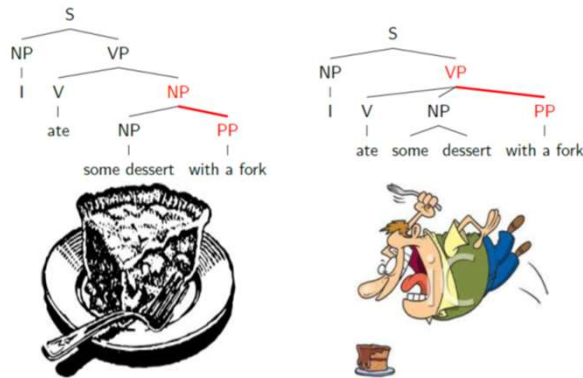
correct implementation of deep neural network
for estimating coin flip prob.

AMBIGUITY AND LINGUISTIC LEVELS



GRAMMARS & AMBIGUITY

I ate some dessert with a fork.



PARSING & AMBIGUITY

- The parser search space is huge as for the effect of several forms of ambiguity that interacts in a combinatorial way

- e.g. *La vecchia porta la sbarra,*
- or *Buffalo buffalo Buffalo buffalo buffalo*

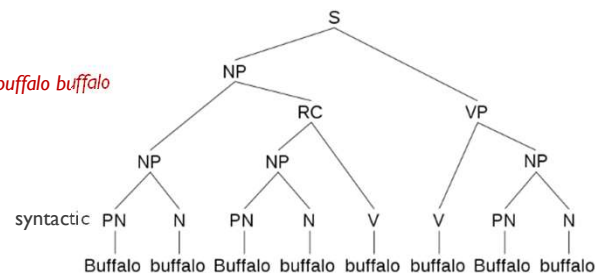
- Notice the strong relationship with semantics
 - Most of the ambiguities cannot be solved at the sole level
 - Lexical information (e.g. word senses) are crucial:



- To operate in a market* viz. *To operate a body part*



- Operare in un mercato* ≠ *Operare un paziente*



Bison from Buffalo, New York who are intimidated by other bison in their community also happen to intimidate other bison in their community



SEMANTICS

- What is the meaning of the sentence

John saw Kim?

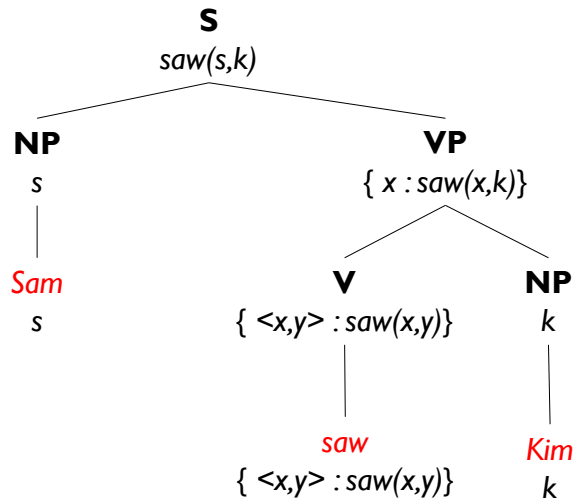


- Desirable Properties:
 - It should be **derivable as a function of the individual constituent parts**, i.e. the meanings of constituents such as *Kim*, *John* and *see*
 - Independent from syntactic phenomena**, e.g. *Kim was seen by John* is a paraphrase as *it has the same semantics*
 - It must be directly used **to trigger some inferences**:
 - Who* was seen by *John*? *Kim*!
 - John* saw *Kim*. *He* started running to *her*.

A TRUTH CONDITIONAL SEMANTICS



John saw Kim



THE DISTRIBUTIONAL HYPOTHESIS

STUDIES IN
LINGUISTIC ANALYSIS

John Rupert Firth

IV

The *placing* of a *text* as a constituent in a context of situation contributes to the statement of meaning since situations are set up to recognize *use*. As Wittgenstein says, 'the meaning of words lies in their use.'⁴ The day to day practice of playing language games recognizes customs and rules. It follows that a text in such established usage may contain sentences such as 'Don't be such an ass!', 'You silly ass!', 'What an ass he is!' In these examples, the word *ass* is in familiar and habitual company, commonly collocates with *you silly—, he is a silly—, don't be such an—, You shall know a word by the company it keeps!* One of the meanings of *ass* is its habitual collocation with such other words as those above quoted.⁵ Though Wittgenstein was dealing with another problem, he also recognizes the plain face-value, the physiognomy of words. They look at us!⁶ 'The sentence is composed of the words and that is enough.'

⁴ Many of Deane Berman's 'lectures' follow the pattern of the *Idiot*. See especially 'Pick the Winner' in *Paradoxes*, Cambridge, 1988.
⁵ See 'General Linguistics and Descriptive Grammar', pp. 80-4.
⁶ 'Linguistics: An Introduction to the Study of Language', pp. 10-11.
⁷ 'Philosophical Investigations', pp. 80, 106.
⁸ See 'Moral of Meaning', pp. 101-7. In this essay, collocation is first suggested as a technical term.
⁹ See *Philosophical Investigations*, p. 101.

Firth, J.R. (1957). "A synopsis of linguistic theory 1930-1955". *Studies in Linguistic Analysis*: 1-32. Reprinted in F.R. Palmer, ed. (1968). *Selected Papers of J.R. Firth 1952-1959*. London: Longman.

<https://cs.brown.edu/courses/csci2952d/readings/lecture1-firth.pdf>

LINGUISTICS AND COMPUTATIONAL SEMANTICS

- **Foundation:** Linguistic theory positing that **words with similar contexts have similar meanings**.
 - ... and **representation** from our computational perspective
- **Computational Leap:** tied to the Vector Space Model (Salton, 1975); represents documents and words as **vectors in a metric space**.
 - **Key Idea:** Documents are characterized by their words, and words by the documents they appear in.
 - 🧠 Initially a Bag of Words model

APPROACHES FOR REPRESENTING WORDS

Distributional Semantics (Count)

- Used since the 90's
- Sparse word-context PMI/PPMI matrix
- Decomposed with SVD

Word Embeddings (Predict)

- Inspired by deep learning
- word2vec (Mikolov et al., 2013)
- GloVe (Pennington et al., 2014)



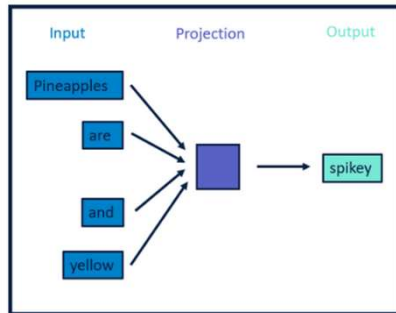
Underlying Theory: **The Distributional Hypothesis** (Harris, '54; Firth, '57)
 "Similar words occur in similar contexts"

(Baroni et al, 2014) Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors – ACL
<https://aclanthology.org/P14-1023/>

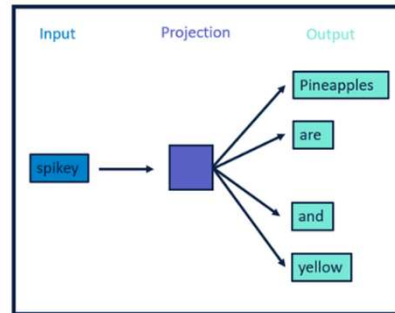
THE TWO MODELS BEHIND WORD2VEC

Contextual Bag Of Word:

Predicts a target word based on context words.

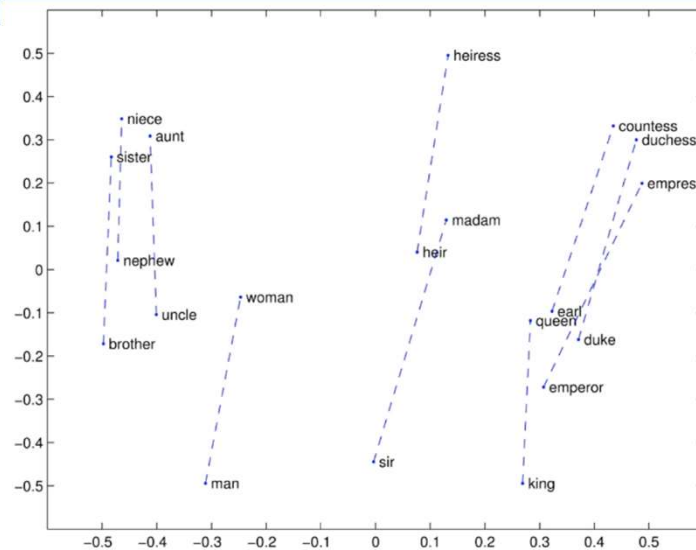


CBOW

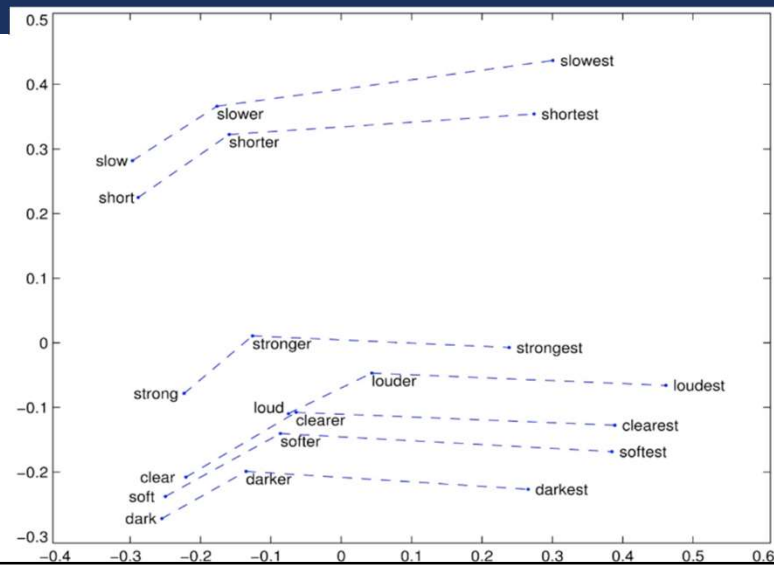


Skip-gram

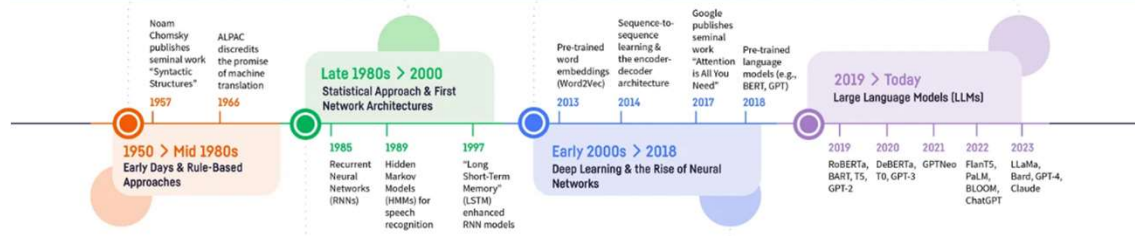
GEOMETRY AND MEANING ...



GEOMETRY AND MEANING ...



LANGUAGE MODELING



- Language Modeling:
 - Statistical approaches
 - Neural approaches to NL semantics

NATURAL LANGUAGE AS A MARKOV PROCESS

Output



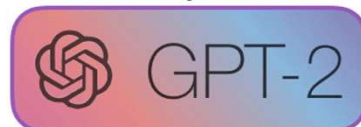
Fundamental Questions for Probabilistic Language Models

- What is **the most likely word** given the left most recent context?
- What is the **probability of an entire sentence**?
- What is the **most likely (inner/hidden) state sequence** given the (observable) sentence?

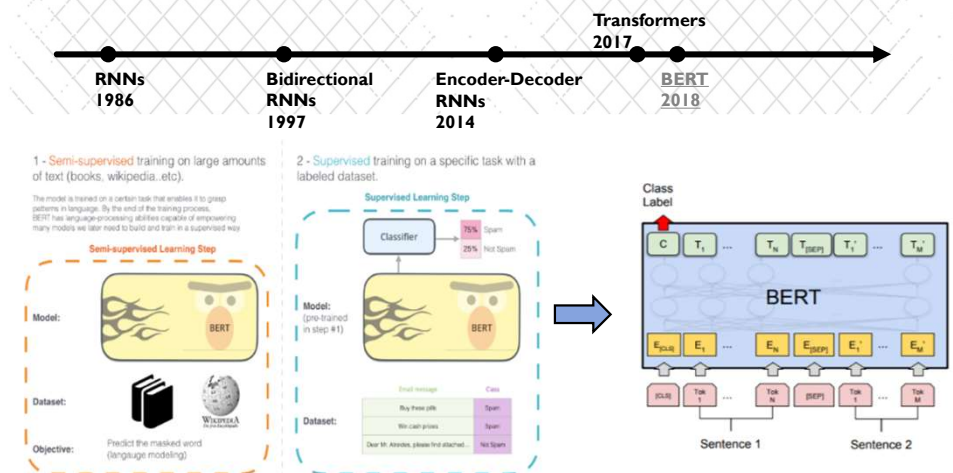
↑
GENERATIVE
LANGUAGE
MODEL

LANGUAGE MODELING AS A NEURAL DECODING

Output



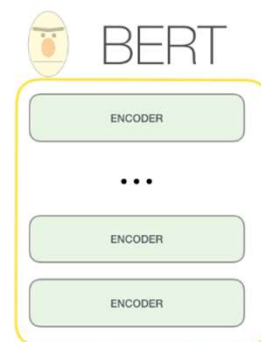
BERT: Encoding Natural Language Semantics through Transformers



BERT (DEVLIN ET AL, 2018)

Bidirectional Encoder Representations from Transformers

- Only the encoder is used
- Designed to generate **contextual meaningful representation** of input words
 - Representations are **context sensitive**, thanks to self-attention
 - Understand the context of a word in a sentence from **both left and right sides** (bidirectionally).
- Representations are embeddings
 - not suitable for text generation
 - ... but for many other tasks



Images from <https://jalammar.github.io/illustrated-bert/>

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805.

BERT (DEVLIN ET AL, 2018)

Why should it work?

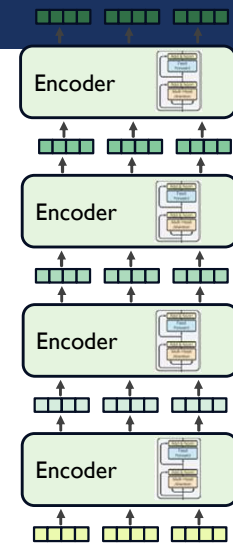
- It is just a piece of the Transformer architecture (next in few slides)

The GREAT IDEA: Pre-Training the encoder

- Pre-trained on a large corpus of text and then fine-tuned for specific tasks like question answering, sentiment analysis, etc.

Images from <https://jalammar.github.io/illustrated-bert/>

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805.



NO PRE-TRAINING NO PARTY! THE REVOLUTION OF PRE-TRAINING IN NLP

- Simple idea:** train a (possibly large) model on a different task and re-use it on your task
 - circumventing the need for training from scratch
 - facilitating “quicker”, more effective deployment of the model
- Precedent in Computer Vision:**
 - This strategy mirrors developments in computer vision
 - Architectures pre-trained on classification tasks using datasets like ImageNet
 - When applied on related task, these “starting point” achieve very good results
- Addressing Overfitting in Large Models:**
 - With **increasing model sizes** and parameter counts, the **risk of overfitting grows**
 - Pre-training on vast datasets mitigates this by providing a broad learning base.

TOWARDS FOUNDATION MODELS

- **Emergence of Foundation Models in NLP:**
 - Large-scale models trained on linguistic tasks, forming a versatile base that can be fine-tuned for various specific applications.
- **Everybody worked on customizing Foundation Models:**
 - Leverage the extensive knowledge encapsulated in Foundation Models by fine-tuning them for particular NLP tasks.
- If you are interested in foundation models
 - [Zhou et al, 2023] A Comprehensive Survey on Pretrained Foundation Models: A History from BERT to ChatGPT
 - <https://arxiv.org/abs/2302.09419>

PRETRAINING BERT

- BERT takes a sequence of tokens as input
- Utilizes **self-attention across layers** to generate context-aware representations of each token in the sequence.
- In each layer, $h=12$ $\mathbf{W}^Q, \mathbf{W}^K, \mathbf{W}^V$ matrices
- **Pre-training tasks:**
 - **Masked-language modeling**

Use the output of the masked word's position to predict the masked word

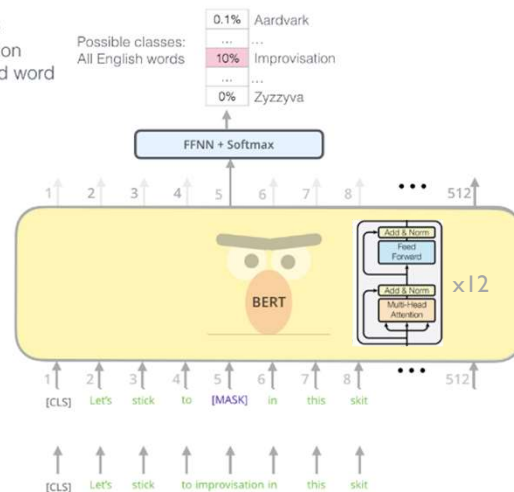
Possible classes:
All English words

0.1%	Aardvark
...	...
10%	Improvisation
...	...
0%	Zyzyva



Randomly mask 15% of tokens

Input

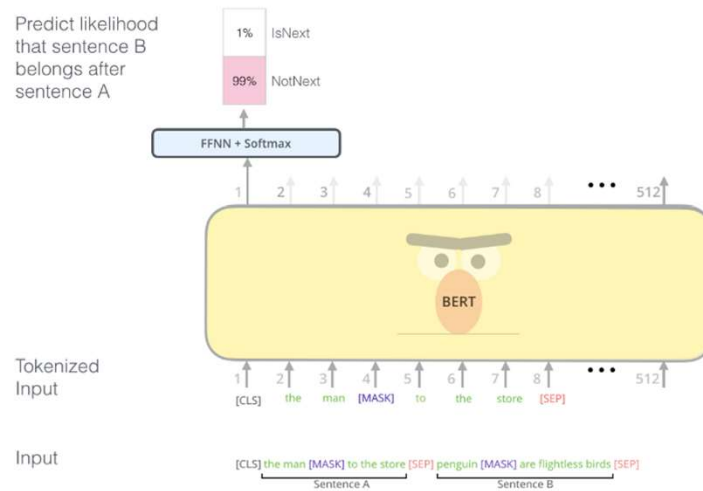


PRETRAINING BERT (2)

- BERT takes a sequence of tokens as input
- Utilizes **self-attention across layers** to **generate context-aware** representations of each token in the sequence.
- In each layer, $h=12$ W^Q, W^K, W^V matrices
- **Pre-training tasks:**
 - Masked-language modeling
 - **Next sentence prediction**

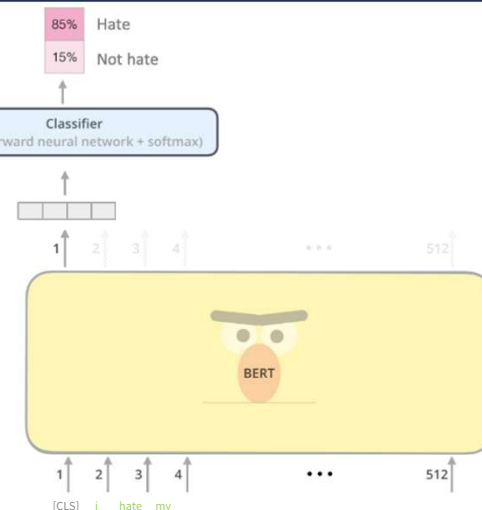
Pretrained using the Toronto BookCorpus (800M words) and English Wikipedia (2,500M words)

Predict likelihood that sentence B belongs after sentence A

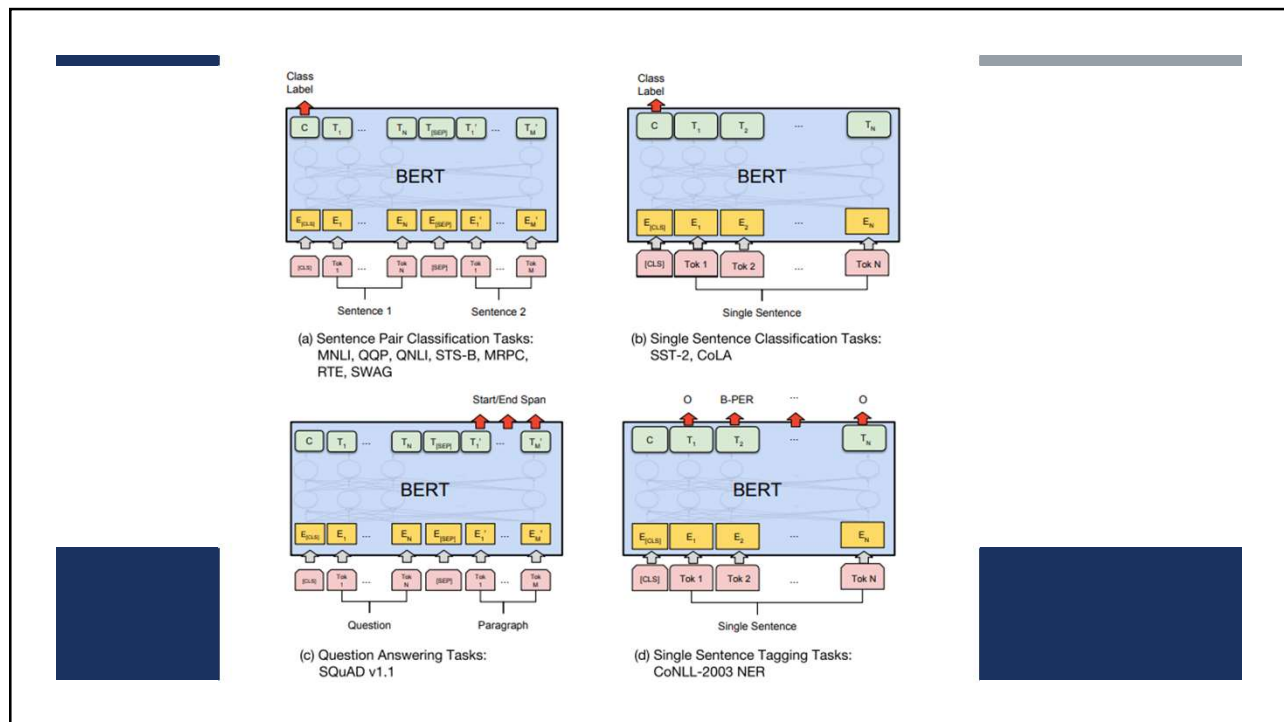


BERT AND FINE-TUNING

- Once pretrained, we can **apply it to new sentences**
- BERT will **produce encoded representations** for each input symbol
- And it can be used in **different classification** just adding a new (linear) classifier...
- ... through fine-tuning of the entire architecture
- not trivial to forget what learned during the pre training



Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805.



LANGUAGE MODELING AND REASONING

- Logical Entailment: the axiomatic «logical» view
- Training Automatic Entailment systems
 - From formal logic to NL
 - Recognizing Textual Entailment as a learning modality without any training example
- Applied Textual Entailment
 - Classification of Sentence Pairs as a new task
 - New Task description as Prompting
- Prompting Applications



ENTAILMENT: THE «LOGICAL» VIEW

- Logical implication is used to express the **entailment relationship** between two subformulas

$$A \rightarrow B \qquad \forall x A(x) \rightarrow B(x)$$

- Logics helps in expressing **logical reasoning schemata** through normalized forms, e.g.,

$$A \rightarrow B \equiv \neg A \vee B \qquad \forall x A(x) \rightarrow B(x) \equiv \neg A(e) \vee B(e) \quad (\text{after Skolemization})$$

- or equivalent variants

$$A \rightarrow B \equiv \neg(A \wedge \neg B) \qquad \forall x A(x) \rightarrow B(x) \equiv \forall x \neg(A(x) \wedge \neg B(x))$$

ENTAILMENT: SEMANTICS

- Logical implication is tightly related to **semantics**, as it is the basis for an efficient approach to logical reasoning.

- In fact $\{A\} \models B$ iff $\{\} \models (A \rightarrow B)$ (Worlds where A is true also make B true, i.e. $A \rightarrow B$ is a tautology)

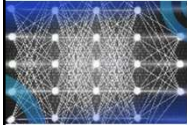
- B is semantically implied by A (only) if $(A \rightarrow B)$ is a tautology. This is used for the algorithms based on **proof by contradiction**, i.e.,

$$\{A\} \models B \text{ iff } \{A, \neg B\} \models \perp \text{ or } \quad (\text{with } \perp \text{ denoting the always false formula})$$

$$\{\Delta, A\} \models B \text{ iff } \{\Delta, A, \neg B\} \models \perp$$

HOW TO DECIDE ABOUT ENTAILMENT THROUGH TRANSFORMERS

- Logical implication (such as $\{A\} \models B$) is usually managed through **a chain of deductive steps** (as in logic programming) from the input query (i.e. a theorem to be demonstrated) to its fully resolved facts, or through contradictions
- Limitations: not formal treatment of uncertainty, poor coverage (the axiomatic system Δ is not fully known a priori), pervasive complexity within large knowledge bases.



- Neural Networks can be adopted to limit the impact of incompleteness or noise in the reference rules and minimize the risk of mistakes in the entailment decision.**
 - LANGUAGE KNOWLEDGE** allows to employ linguistic semantics for approximating logical deductions
 - The **deduction chain** can be successful or not: this implies that **the entire inference can be mapped into a BINARY CLASSIFICATION TASK**
 - The input is a pair A and B of the sentences corresponding respectively to the hypothesis (A) and to the thesis (B)

ENTAILMENT & TRANSFORMERS



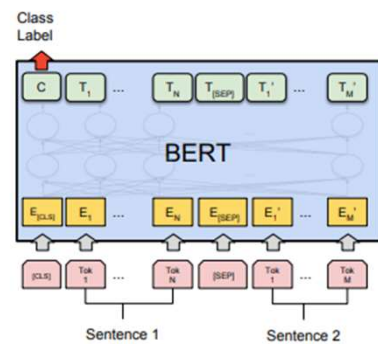
A possible process is

- Map the logical rules (as axioms) into a training dataset
- Map a new potential theorem into a natural language sentence
- Make the sentence the input of a NNs
- Solve the inference task of accepting/rejecting the entailment as a **binary classification task**
- In other words, given a training set of axioms such as
 - $\Delta: \{A_1 \rightarrow B_1, \dots, A_n \rightarrow B_n\}$
 - Induce a function RTE such that for every future pair (A_i, B_j)
 - $h(A_i, B_j) = \text{true}$ iff $\{\Delta, A_i\} \models B_j$ or alternatively $h(A_i \rightarrow B_j) = \text{true}$ iff $\{\Delta, A_i\} \models B_j$

THE ROLE OF TRANSFORMERS

■ First setting

- $h(A_i, B_j) = \text{true}$ iff $\{\Delta, A_i\} \models B_j$
- Input given by 2 sentences
- BERT used as the encoder
- A stacked classifier is trained on labeled pairs
- Type of Inference:
 - PARAPHRASING
 - TEXTUAL ENTAILMENT

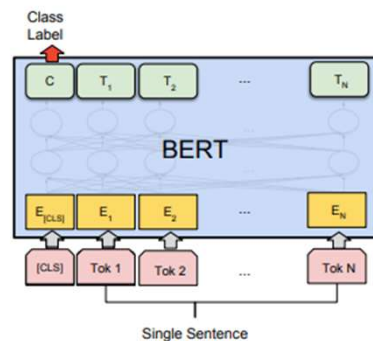


(a) Sentence Pair Classification Tasks:
MNLI, QQP, QNLI, STS-B, MRPC,
RTE, SWAG

THE ROLE OF TRANSFORMERS (2)

■ Second setting

- $h(A_i \rightarrow B_j) = \text{true}$ iff $\{\Delta, A_i\} \models B_j$
- Input given 1 sentence expressing the task over A_i and B_j
- BERT used as the encoder
- A stacked classifier is trained on labeled pairs
- Example (PARAPHRASING):
 - «The sentence B_j has the same meaning of sentence A_i »
 - «Sentence A_i means the same as B_j »



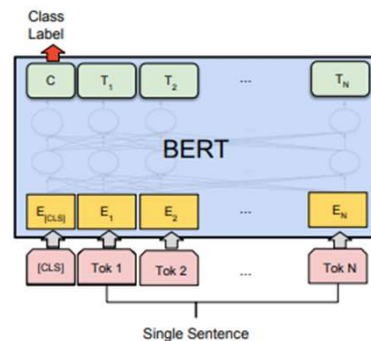
(b) Single Sentence Classification Tasks:
SST-2, CoLA

THE ROLE OF TRANSFORMERS (3)

■ Second setting

- $h(A_i \rightarrow B_j) = \text{true}$ iff $\{\Delta, A_i\} \models B_j$
- Input given 1 sentence expressing the task over A_i and B_j
- BERT used as the encoder
- A stacked classifier is trained on labeled pairs

- Example (TEXTUAL ENTAILMENT):
- «The sentence B_j is implied by sentence A_i »
- «Sentence A_i guarantees the truth of B_j »



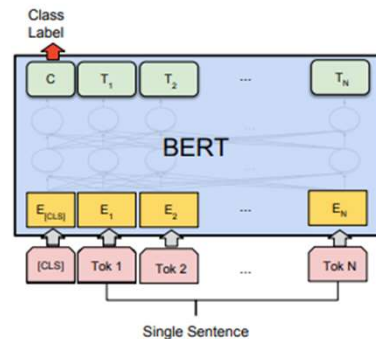
(b) Single Sentence Classification Tasks:
SST-2, CoLA

NEURAL ENTAILMENT: APPLICATIONS

■ The setting

$$h(A_i \rightarrow B_j) = \text{true} \text{ iff } \{\Delta, A_i\} \models B_j$$

- corresponds to expressions that depend on complex interactions between A_i and B_j mapped into an individual sentence
 - **BERT can be always used as the encoder**
 - The **stacked classifier is an automatic entailment recognition tool**
- Future TEXTUAL ENTAILMENT tasks, e.g., :
 - TOPICAL CLASSIFICATION
 - «The sentence B_j is classified by label A_i », «Label A_i corresponds to the topic of B_j »
 - SENTIMENT ANALYSIS:
 - « A_i implies the sentiment label B_j », « A_i expresses sentiment B_j »



(b) Single Sentence Classification Tasks:
SST-2, CoLA

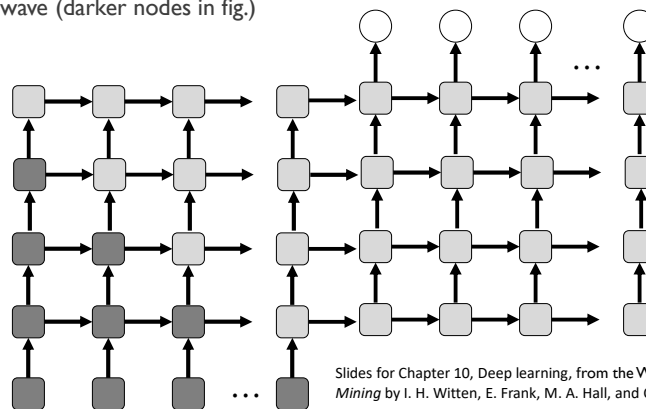
RETI NEURALI AVANZATE: INTERNALS

LA ATTENZIONE ED I TRANSFORMERS

METODI E ARCHITETTURE

ENCODER-DECODER DEEP ARCHITECTURES

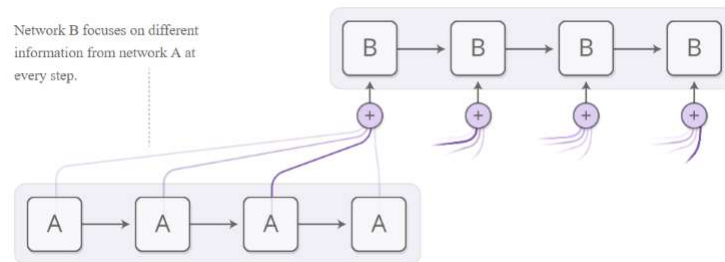
- Given enough data, a deep encoder-decoder architecture (see below) can yield results that compete with hand-engineered translation systems.
- The connectivity structure means that partial computations in the model can flow through the graph in a wave (darker nodes in fig.)



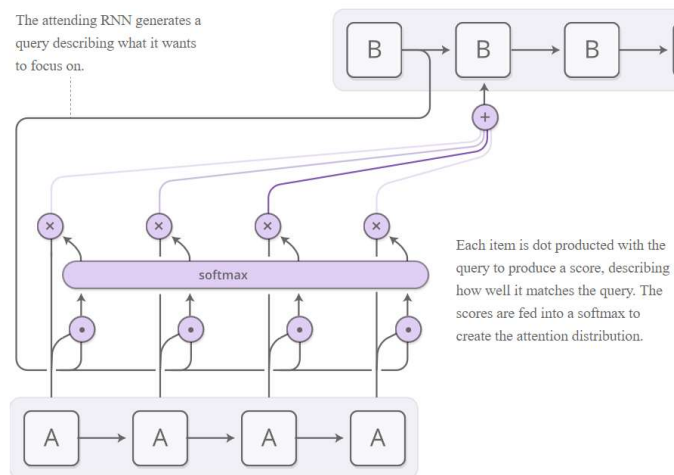
Slides for Chapter 10, Deep learning, from the Weka book, *Data Mining* by I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal

ATTENTION-BASED RNNs

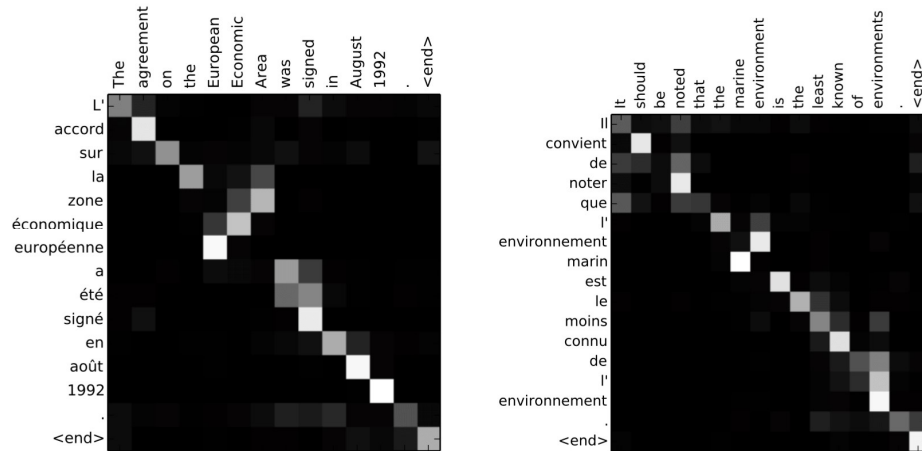
- A NN (e.g. B) is used to attend the outcome of a second network A, e.g. (Vaswani et al., 2017)



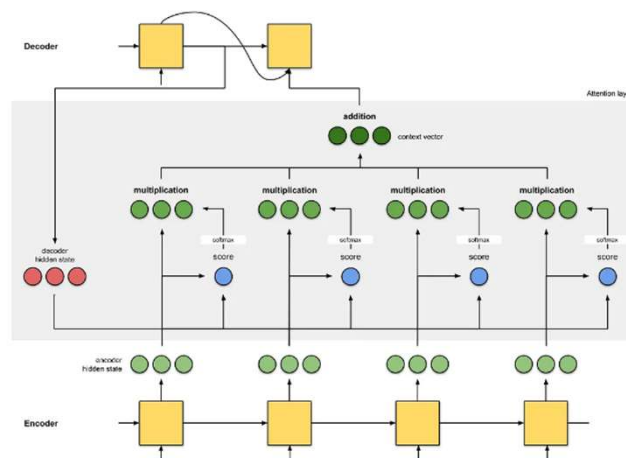
ATTENTION-BASED RNNs



ATTENTION IN MACHINE TRANSLATION

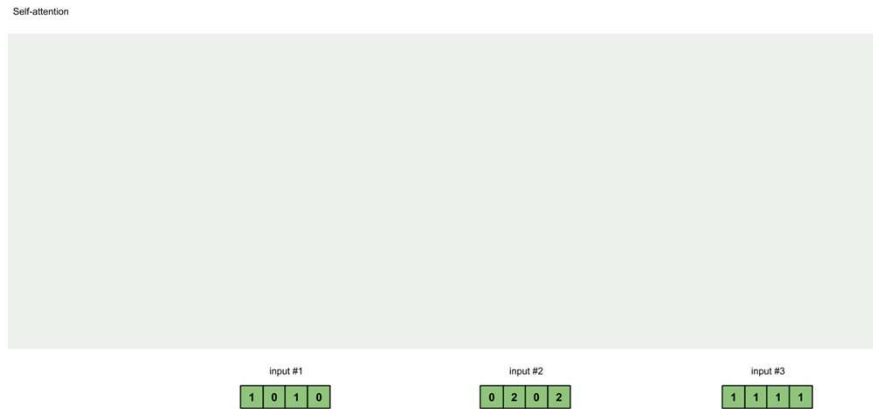


ATTENTION IN SEQ2SEQ MODELS



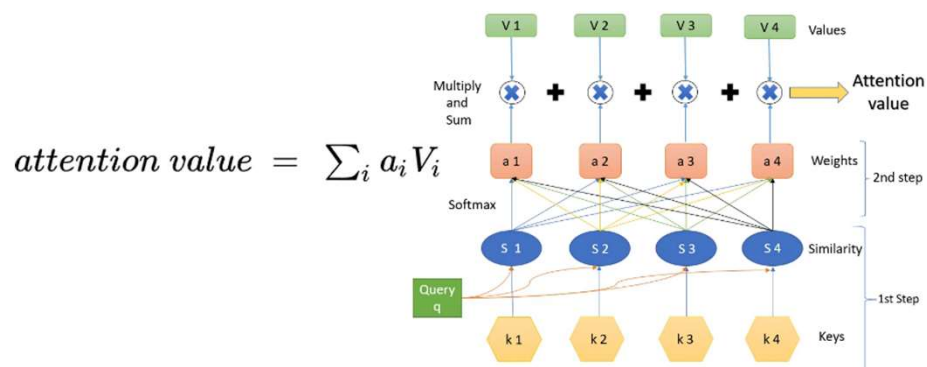
by Manuel Romero: from *Attn: Illustrated Attention*, by Raimi Karim, [Towards Data Science](#), Jan 20, 2019

SELF-ATTENTION

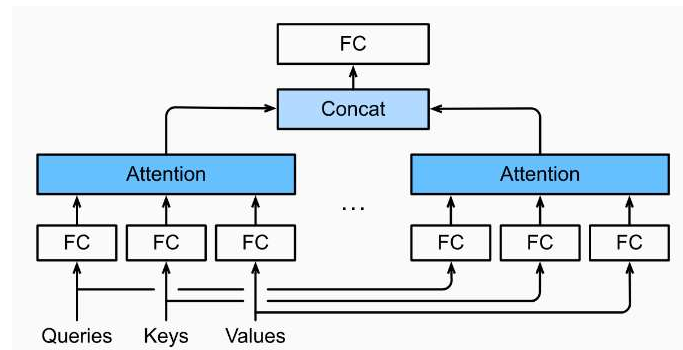


From <https://colab.research.google.com/drive/1rPk3ohrmVclqhH7uQ7qys4oznDdAhpzF> by Manuel Romero

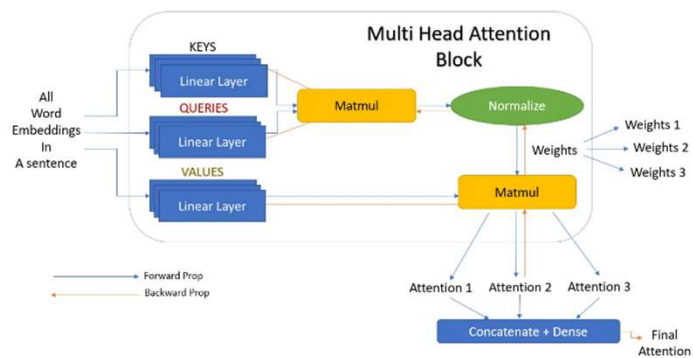
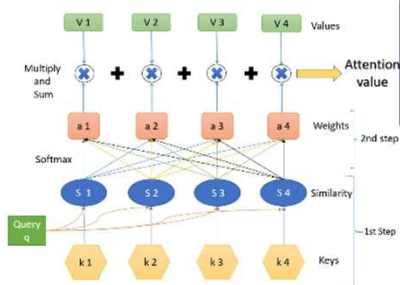
THE ATTENTION INFORMATION FLOW



ATTENTION: MULTIHEAD



MULTIHEAD ATTENTION AND TRAINING



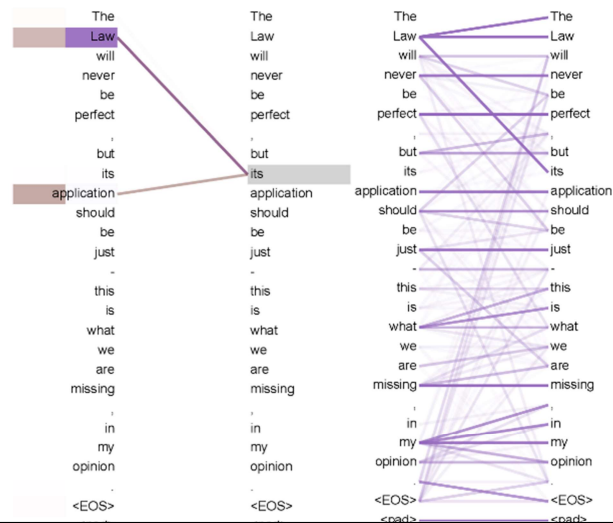
ATTENTION IN MACHINE TRANSLATION

- Multihead attention is first captured at the encoding level between words in the input
- The different levels encode attention across multiple groups of word
- During Decoding the overall attention is used to condition individual emissions left to right
- As a results, emissions are made dependent on the entire input sequence and all dependencies are captured
- Queries are individual words embeddings, while keys are trained so that attention weights are learned from examples during training
- All attentions are thus targeted to minimize (decoding) errors

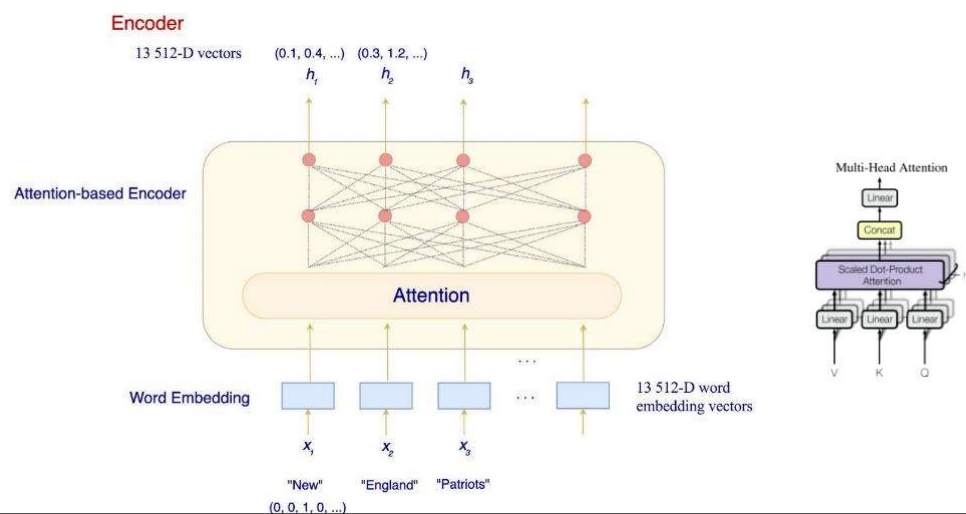
ATTENTION & ENCONDING

- In a decoding process (e.g. machine translation) there are **three** kinds of dependencies for neural architectures
- Dependencies are **independently** established between
 1. the **input and output** tokens
 2. the **input tokens themselves**
 3. the **output tokens themselves**
- Examples:
 - Machine Translation
 - QA where the query the answer paragraph is the input and the matched answer is the output

ATTENTION AND ANAPHORA



BERT: EXPLOITING ATTENTION FOR NLP



BERT & NLP: TRAINING THE ENCODER (ONLY)

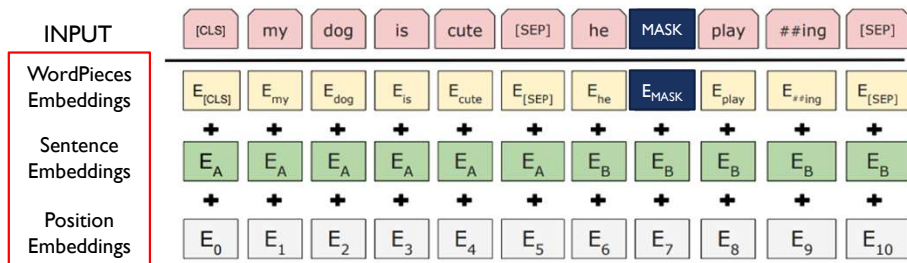
- How to *train* (i.e. *optimize*) the encoding?
- **Two General and complex** tasks are proposed in (Devlin et al., 2018) are
 - **Masked Language Modeling** (15%)
 - Inspired by Distributional Hypothesis
 - Can be Simulated and does not require any labeling
 - **Next Sentence Prediction**
 - Inspired by Textual Inference tasks (e.g. Textual Entailment)
 - Can be Simulated and does not require any labeling
- **Source Representations**
 - Words? And why not *subword*? (in the BERT jargon) Word Pieces!!
 - Useful to deal with out-of-vocabulary phenomena

BERT (DEVLIN ET AL. '18)

Pretraining on two unsupervised prediction tasks:

- **Masked Language Model:** given a sentence s with missing words, reconstruct s
 - Example: Amazon <MASK> amazing \rightarrow Amazon is amazing
 - In BERT the language modeling is deeply Bidirectional, while in ELMo the forward and backward LMs were two independent branches of the NN
- **Next Sentence Prediction:** given two sentences s_1 and s_2 , the task is to understand whether s_2 is the actual sentence that follows s_1
 - 50% of the training data are positive examples: s_1 and s_2 are actually consecutive sentences
 - 50% of the training data are negative examples: s_1 and s_2 are randomly chosen from the corpus

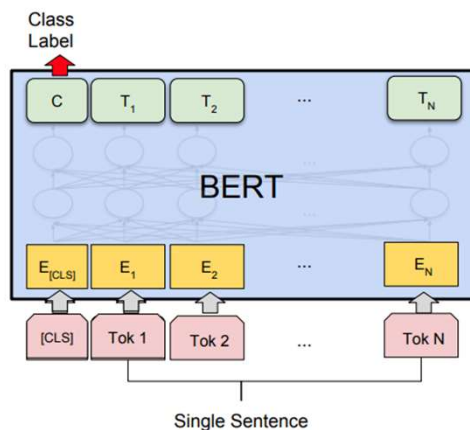
BERT PRETRAINING: INPUT REPRESENTATIONS



All these embeddings are learned during the (pre)training process

In pre-training 15% of the input tokens are masked for the masked LM task

BERT (DEVLIN ET AL. '18) TASKS: INDIVIDUAL SENTENCE CLASSIFICATION



BERT for single sentence classification (Sentiment analysis, Intent Classification, etc.)



Jon Gordon
@JonGordon11

I love this time of year between Christmas and the New Year. It's a time of reflection, hope, vision and possibility. A good time to think about who you want to be and the life you want to create.



shauna
@wednesdaysadums · Follow

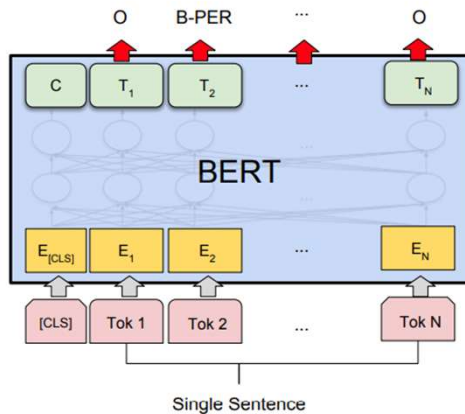
the ryanair bag policy would actually make you suicidal

5:21 PM · Jan 9, 2023

♥ 50 💬 Reply ↗ Share



BERT (DEVLIN ET AL. '18) TASKS: SEQUENCE TAGGING

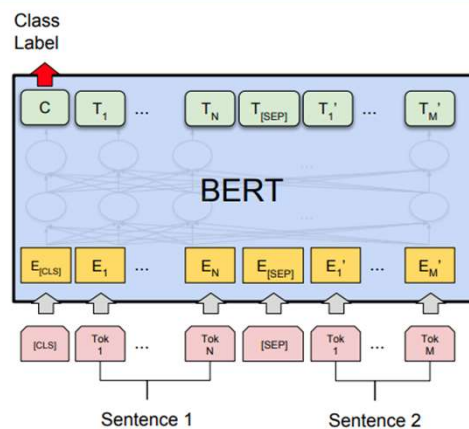


Task: Slot tagging

x 178:1 # BOS	y 128:1 # O
x 770:1 # show	y 128:1 # O
x 429:1 # flights	y 128:1 # O
x 444:1 # from	y 128:1 # O
x 272:1 # burbank	y 48:1 # B-fromloc.city_name
x 851:1 # to	y 128:1 # O
x 789:1 # st.	y 78:1 # B-toloc.city_name
x 564:1 # louis	y 125:1 # I-toloc.city_name
x 654:1 # on	y 128:1 # O
x 601:1 # monday	y 26:1 # B-depart_date.day_name
x 179:1 # EOS	y 128:1 # O

BERT for **Sequence Tagging Tasks** (e.g., POS tagging, Named Entity Recognition, etc.)

BERT (DEVLIN ET AL. '18) TASKS: SENTENCE PAIRS CLASSIFICATION



Answer selection in QA: Decide if A contains an **answer** to Q:

Q: "What is the Capital of Italy?"

A: "Rome, as the capital of Italy, is located"

RTE: Given T decide if H is **true** (or not)

T: "Rome is the Capital of Italy."

H: "Rome is in Italy."

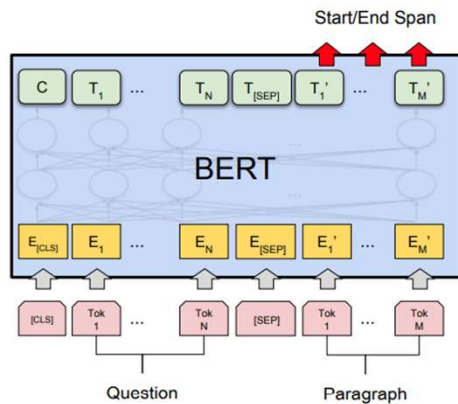
PI: Given S1 and S2 decide if they are **paraphrases** (or not)

S1: "Rome is the Capital of Italy."

S2: "Italy has Rome as its own Capital town."

BERT for sentence pairs classification (answer selection in QA, Recognizing Textual Entailment, Paraphrase Identification)

BERT (DEVLIN ET AL. '18) TASKS: QUESTION ANSWERING



Answer Span Selection in QA:

Decide which part of the text A corresponds to the answer to the query Q:

Q: "What is the Capital of Italy?"

A: "<Start>Rome<End>, as the capital of Italy,"

BERT for Answer Span Selection in Question Answering

A QA EXAMPLE ON SQUAD

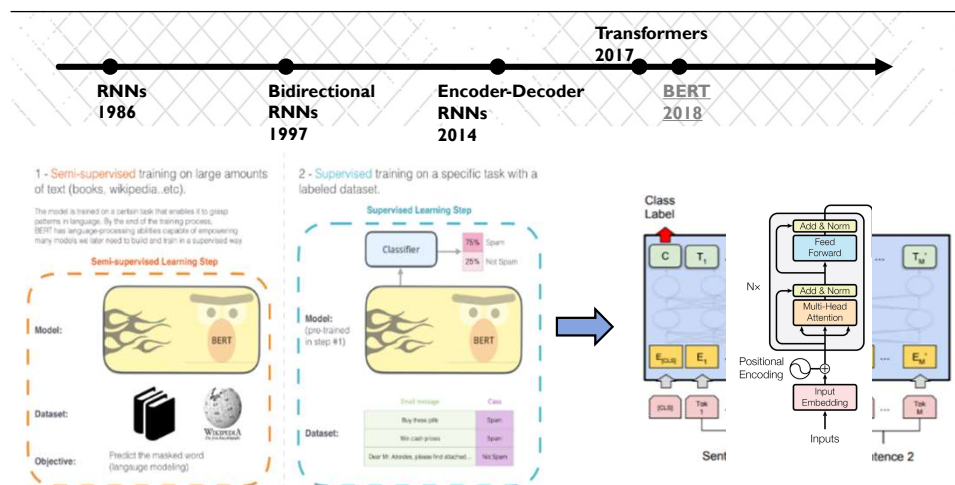
- Question Answering even across languages
 - Query in Italian
 - Answer span over English Texts
 - from Danilo Croce, Alexandra Zelenanska, Roberto Basili: *Neural Learning for Question Answering*, in Italian. AI²IA 2018, Springer Verlag, 389-402, 2018.

The screenshot shows the COVID-19 HQ interface. At the top, it says "COVID-19 HQ" and "Insert your question here:". Below this is a search bar with the query "How is Covid-19 transmitted?". To the right of the search bar is a red button labeled "Q SEARCH". Below the search bar, there are three radio buttons: "Order by Solr", "Order by BERT", and "Order by Solr-BERT". The "Order by BERT" button is selected. Below the radio buttons is a search result snippet titled "In-flight Transmission Cluster of COVID-19: A Retrospective Case Series Running title: In-flight Transmission Cluster of COVID-19 figure". The snippet includes a list of authors and a paragraph of text discussing the transmission of COVID-19.

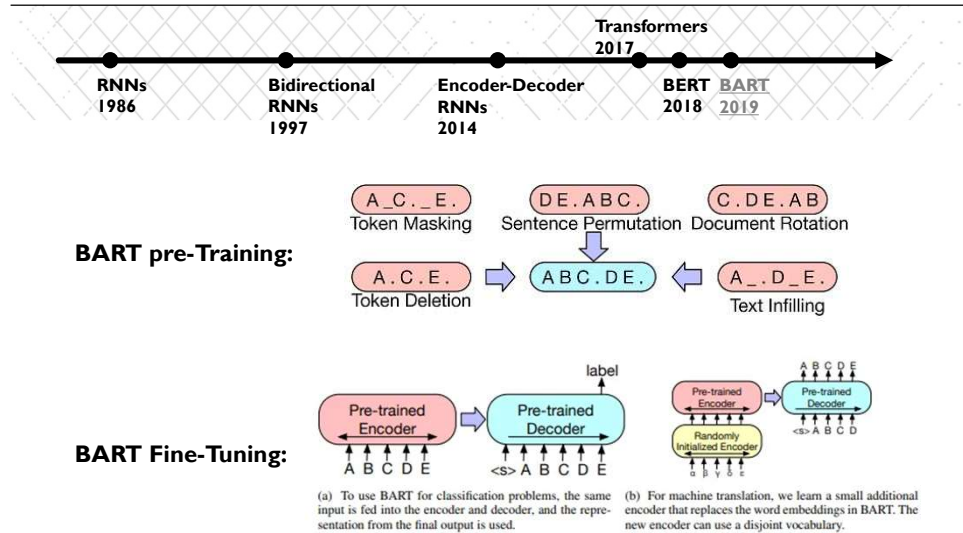
RETI NEURALI AVANZATE: DALL'AUTOENCODING ALLA IA GENERATIVA

METODI E ARCHITETTURE

Machine learning paradigms underlying ChatGPT



Machine learning paradigms underlying ChatGPT: BART



GPT-2: DECODER ONLY ARCHITECTURES (RADFORD ET AL., 2019)

- “We demonstrate that language models begin to learn these tasks without any explicit supervision when trained on a new dataset of millions of webpages called WebText”
- GPT-2 is a large transformer-based language model with 1.5 billion parameters, trained on a dataset of 8 million web pages.
- GPT-2 is trained with a simple objective: predict the next word, given all of the previous words within some text.
- The diversity of the dataset causes this simple goal to contain naturally occurring demonstrations of many tasks across diverse domains.
- GPT-2 is a direct scale-up of GPT, with more than 10X the parameters and trained on more than 10X the amount of data

GPT-2: SOURCES OF INSPIRATION

- Multitask QA Networks (MQAN) (McCann et al, 2018)

Examples

Question	Context	Answer	Question	Context	Answer
What is a major importance of Southern California in relation to California and the US?	...Southern California is a major economic center for the state of California and the US...	major economic center	What has something experienced?	Areas of the Baltic that have experienced eutrophication ,	eutrophication
What is the translation from English to German?	Most of the planet is ocean water.	Der Großteil der Erde ist Meerwasser	Who is the illustrator of Cycle of the Werewolf?	Cycle of the Werewolf is a short novel by Stephen King, featuring illustrations by comic book artist Bernie Wrightson .	Bernie Wrightson
What is the summary?	Harry Potter star Daniel Radcliffe gains access to a reported £320 million fortune ...	Harry Potter star Daniel Radcliffe gets £320M fortune...	What is the change in dialogue state?	Are there any Erftrean restaurants in town?	Food Erftrean
Hypothesis: Product and geography are what make cream skinning work. Entailment , neutral, or contradiction?	Premise: Conceptually cream skinning has two basic dimensions - product and geography.	Entailment	What is the translation from English to SQL?	The table has column names... Tell me what the notes are for South Australia	SELECT notes from table WHERE 'Current Slogan' = 'South Australia'
Is this sentence positive or negative ?	A stirring, funny and finally transporting re-imagining of Beauty and the Beast and 1950s horror film.	positive	Who had given help?	Joan made sure to thank Susan for all the help she had given.	Susan

Figure 1: Overview of the decaNLP dataset with one example from each decaNLP task in the order presented in Section 2. They show how the datasets were pre-processed to become question answering problems. Answer words in red are generated by pointing to the context, in green from the question, and in blue if they are generated from a classifier over the output vocabulary.

- Our speculation is that **a language model with sufficient capacity will begin to learn to infer and perform the tasks demonstrated in natural language sequences in order to better predict them, regardless of their method of procurement**. If a language model is able to do this it will be, in effect, performing unsupervised multitask learning.

GPT-2: ARCHITECTURE AND TASKS

- From (Radford et al., 2017, GPT paper)

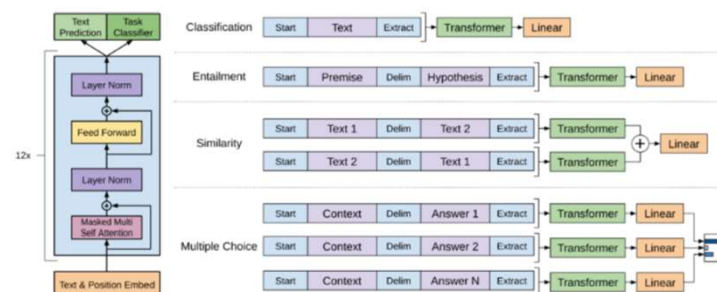


Figure 1: (left) Transformer architecture and training objectives used in this work. (right) Input transformations for fine-tuning on different tasks. We convert all structured inputs into token sequences to be processed by our pre-trained model, followed by a linear+softmax layer.

GPT-2: RESULTS OVER DIFFERENT TASKS

Language Models are Unsupervised Multitask Learners										
	LAMBADA (PPL)	LAMBADA (ACC)	CBT-CN (ACC)	CBT-NE (ACC)	WikiText2 (PPL)	PTB (PPL)	enwik8 (BPB)	text8 (BPC)	WikiText103 (PPL)	1BW (PPL)
SOTA	99.8	59.23	85.7	82.3	39.14	46.54	0.99	1.08	18.3	21.8
117M	35.13	45.99	87.65	83.4	29.41	65.85	1.16	1.17	37.50	75.20
345M	15.60	55.48	92.35	87.1	22.76	47.33	1.01	1.06	26.37	55.72
762M	10.87	60.12	93.45	88.0	19.93	40.31	0.97	1.02	22.05	44.575
1542M	8.63	63.24	93.30	89.05	18.34	35.76	0.93	0.98	17.48	42.16

Table 3. Zero-shot results on many datasets. No training or fine-tuning was performed for any of these results. PTB and WikiText-2 results are from (Gong et al., 2018). CBT results are from (Bajgar et al., 2016). LAMBADA accuracy result is from (Hoang et al., 2018) and LAMBADA perplexity result is from (Grave et al., 2016). Other results are from (Dai et al., 2019).

- The LAMBADA dataset (Paperno et al., 2016)
 - It tests the ability of systems to model long-range dependencies in text.
 - The task is to predict the final word of sentences which require at least 50 tokens of context for a human to successfully predict.

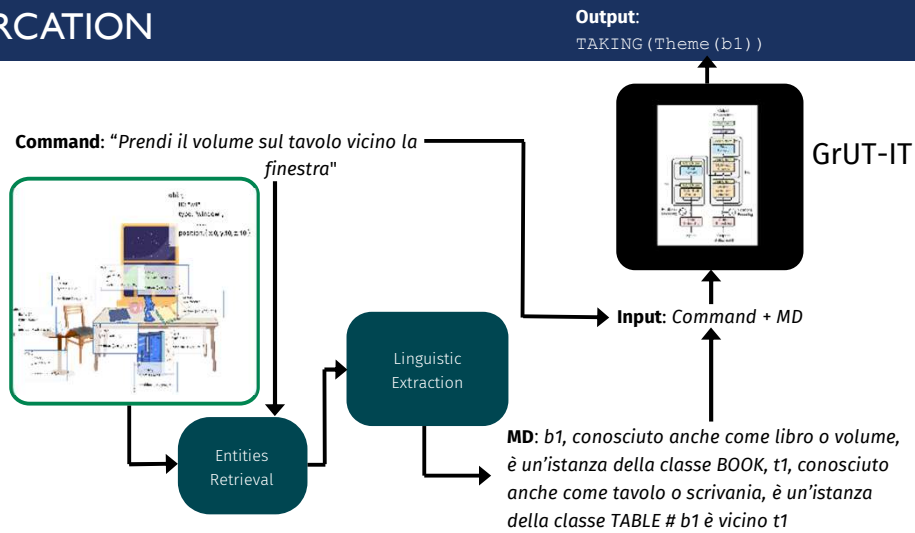
GPT-2: RESULTS ON LAMBADA

- The LAMBADA dataset (Paperno et al., 2016)
 - It tests the ability of systems to model long-range dependencies in text.
 - The task is to predict the final word of sentences which require at least 50 tokens of context for a human to successfully predict.

(1)	Context: "Yes, I thought I was going to lose the baby," "I was scared too," he stated, sincerity flooding his eyes. "You were?" "Yes, of course. Why do you even ask?" "This baby wasn't exactly planned for." Target sentence: "Do you honestly think that I would want you to have a?" Target word: miscarriage
(2)	Context: "Why?" "I would have thought you'd find him rather dry," she said. "I don't know about that," said Gabriel. "He was a great craftsman," said Heather. "That he was," said Flannery. Target sentence: "And Polish, to boot," said Target word: Gabriel
(3)	Context: Preston had been the last person to wear those chains, and I knew what I'd see and feel if they were slipped onto my skin-the Reaper's unending hatred of me. I'd felt enough of that emotion already in the amphitheater. I didn't want to feel anymore. "Don't put those on me," I whispered. "Please." Target sentence: Sergei looked at me, surprised by my low, raspy please, but he put down the Target word: chains
(4)	Context: They tuned, discussed for a moment, then struck up a lively jig. Everyone joined in, turning the courtyard into an even more chaotic scene, people now dancing in circles, swinging and spinning in circles, everyone making up their own dance steps. I felt my feet tapping, my body wanting to move. Target sentence: Aside from writing, I've always loved Target word: dancing

- GPT-2 improves the state of the art from 99.8 (Grave et al., 2016) to 8.6 perplexity and increases the accuracy of LMs on this test from 19% (Dehghani et al., 2018) to 52.66%. Adding a stop-word filter as an approximation to this further increases accuracy to 63.24%.
- Investigating GPT-2's errors showed most predictions are valid sentence continuations, but are not valid final words

GRUT: BART FOR HUMAN ROBOT INTERACTION



Hromei et al, 2022, "Embedding Contextual Information in Seq2seq Models for Grounded Semantic Role Labeling"

GRUT: EXPERIMENTAL EVALUATION

FP = Frame Prediction
AIC = Argument Identification and Classification
EM = Exact Match
HM = Head Match

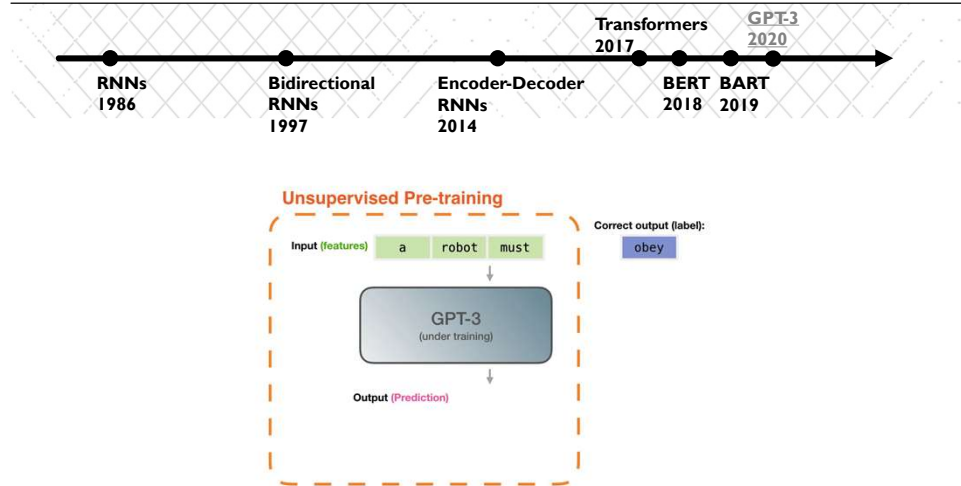
Model	Learning Rate	FP	AIC-Exact Match	AIC-Head Match
LU4R	-	95.32%	77.67%	86.35%
GrUT-IT	$5 \cdot 10^{-5}$	96.86%	82.30%	85.19%

LU4R: TAKING (Theme ("libro"))
GrUT-IT: TAKING (Theme (b1))

Results here are reported as F1 values on 10-fold cross-validation schema with 80/10/10 data split.
Performance for LU4R is reported in *italic* as it is not entirely comparable with.



Machine learning paradigms underlying ChatGPT



GPT3: NOVELTY

- «Language Models are Few-Shot Learners” (Brown et al., 2020)

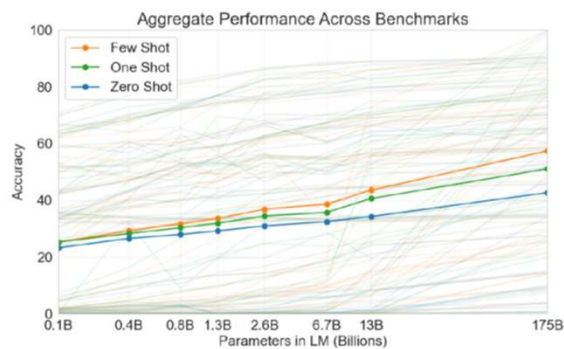


Figure 1.3: Aggregate performance for all 42 accuracy-denominated benchmarks While zero-shot performance improves steadily with model size, few-shot performance increases more rapidly, demonstrating that larger models are more proficient at in-context learning. See Figure 3.8 for a more detailed analysis on SuperGLUE, a standard NLP benchmark suite.

PROMPTING VS. LEARNING

The three settings we explore for in-context learning

Zero-shot

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.



One-shot

In addition to the task description, the model sees a single example of the task. No gradient updates are performed.



Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.



Traditional fine-tuning (not used for GPT-3)

Fine-tuning

The model is trained via repeated gradient updates using a large corpus of example tasks.



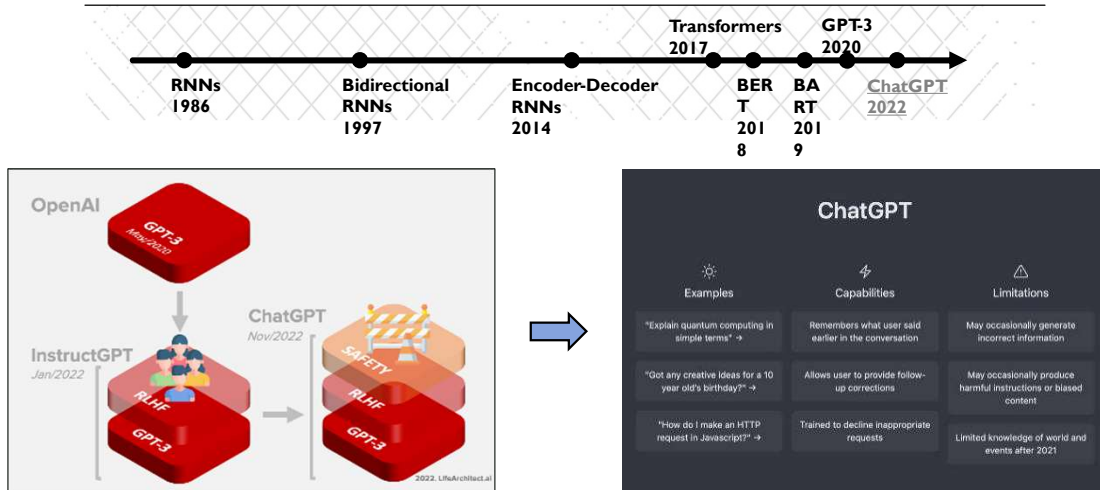
GPT-3: SIZE

Model Name	n_{params}	n_{layers}	d_{model}	n_{heads}	d_{head}	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	6.0×10^{-4}
GPT-3 Medium	350M	24	1024	16	64	0.5M	3.0×10^{-4}
GPT-3 Large	760M	24	1536	16	96	0.5M	2.5×10^{-4}
GPT-3 XL	1.3B	24	2048	24	128	1M	2.0×10^{-4}
GPT-3 2.7B	2.7B	32	2560	32	80	1M	1.6×10^{-4}
GPT-3 6.7B	6.7B	32	4096	32	128	2M	1.2×10^{-4}
GPT-3 13B	13.0B	40	5140	40	128	2M	1.0×10^{-4}
GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128	3.2M	0.6×10^{-4}

Table 2.1: Sizes, architectures, and learning hyper-parameters (batch size in tokens and learning rate) of the models which we trained. All models were trained for a total of 300 billion tokens.

- Here n_{params} is the total number of trainable parameters, n_{layers} is the total number of layers, d_{model} is the number of units in each bottleneck layer (we always have the feedforward layer four times the size of the bottleneck layer, $d_{\text{ff}}=4 \times d_{\text{model}}$), and d_{head} is the dimension of each attention head.
- All models use a context window of $n_{\text{ctx}} = 2048$ tokens

Machine learning paradigms underlying ChatGPT



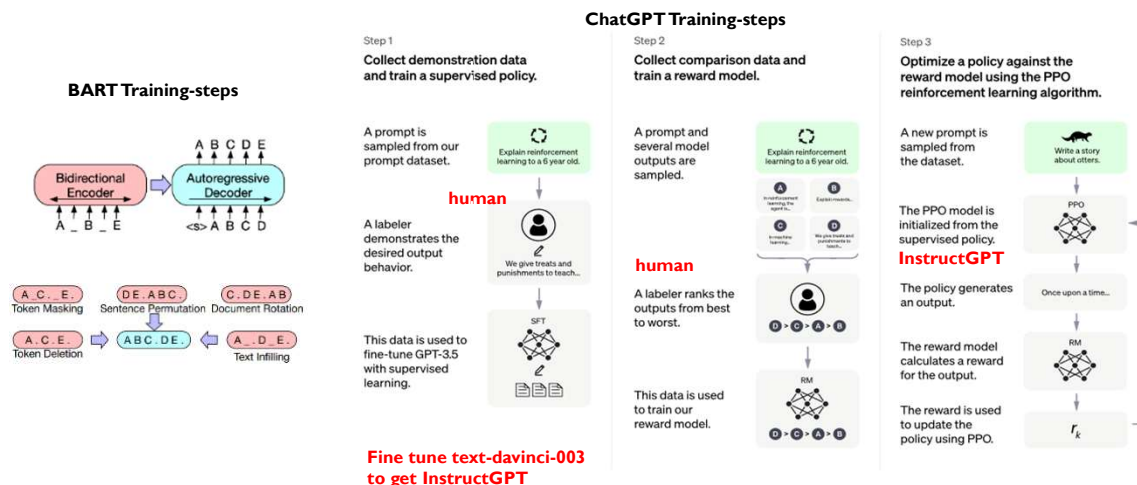
LIMITATIONS OF GPT-3

- Large language models often express unintended behaviors such as making up facts, generating biased or toxic text, or simply not following user instructions. This is because the language modeling objective is **misaligned**.
- The idea: aligning language models by **training them to act in accordance with the user's intention** (Leike et al., 2018).
 - explicit intentions such as following instructions
 - implicit intentions such as staying truthful, and not being biased, toxic, or otherwise harmful.
- Overall Objective: language models should be helpful (they should help the user solve their task), honest (they shouldn't fabricate information or mislead the user), and harmless (they should not cause physical, psychological, or social harm to people or the environment).

INSTRUCT GPT

- **Step 1:** Collect demonstration data, and train a supervised policy. Labelers provide demonstrations of the desired behavior on the input prompt distribution. Then, fine-tuning of a pretrained GPT-3 model on this data using supervised learning is carried out.
- **Step 2:** Collect comparison data, and train a reward model. A dataset of comparisons between model outputs is collected: labelers indicate which output they prefer for a given input. A reward model to predict the human-preferred output is then trained.
- **Step 3:** Optimize a policy against the reward model using PPO. We use the output of the RM as a scalar reward. We fine-tune the supervised policy to optimize this reward using the proximal policy optimization (PPO) algorithm (Schulman et al., 2017).

At the heart of ChatGPT (from BART to ChatGPT)



from Ouyang, L., Wu, J., Jiang, et al. (2022). Training language models to follow instructions with human feedback



MORE ON PROMPTING

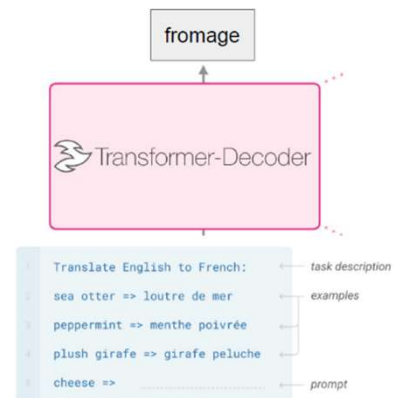


LEARNING MODALITIES

- Fine Tuning (as BERT/BART)
- In-context learning
- Prompting

IN-CONTEXT LEARNING

- Pretrain a large language model on a task
- Manually design a «prompt» that shows how to define a novel task as a generation task
- There is no need to train further the model, i.e. update model weights



[Brown et al. 2020](#)

PROMPTING

- “A good prompt is one that is specific and provides enough context for the model to be able to generate a response that is relevant to the task.” (GPT-3)
- Earliest work in prompts traces back to GPT-1/2 (Radford et al., 2018,2019)
- If LMs are given good prompts they can achieve significant zero-shot performance on NLP tasks ranging from sentiment classification to reading comprehension

PROMPT BASED FINE TUNING

FINE TUNING: more parameters for the stacked classifier, more examples (even in few-shot scenarios)

PROMPT-BASED FINE TUNING: need for good prompts, no further parameters to tune

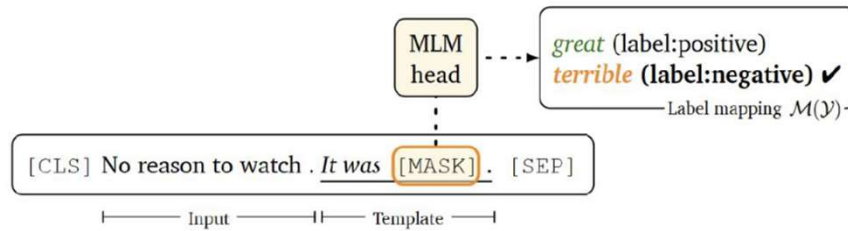


Image Source: Making Pre-trained Language Models Better Few-shot Learners, Gao, et al. 2021

PROMPT-BASED FINE TUNING: THE PROCESS

Input: \mathcal{X}_1 = No reason to watch.

Step 1. Formulate the downstream task into a (Masked) LM problem using a *template*:

`[CLS] No reason to watch . It was [MASK] . [SEP]`

Input Template

Step 2. Choose a *label word mapping* \mathcal{M} , which maps task labels to individual words.

`great (label:positive)`
`terrible (label:negative) ✓`
 Label mapping $\mathcal{M}(\mathcal{Y})$

Image Source: Making Pre-trained Language Models Better Few-shot Learners, Gao, et al. 2021

11

PROMPT-BASED FINE TUNING: THE PROCESS

Step 3. Fine-tune the LM to fill in the correct label word.

$$p(y \mid x_{\text{in}}) = p([\text{MASK}] = \mathcal{M}(y) \mid x_{\text{prompt}})$$

$$= \frac{\exp(\mathbf{w}_{\mathcal{M}(y)} \cdot \mathbf{h}_{[\text{MASK}]})}{\sum_{y' \in \mathcal{Y}} \exp(\mathbf{w}_{\mathcal{M}(y')} \cdot \mathbf{h}_{[\text{MASK}]})},$$

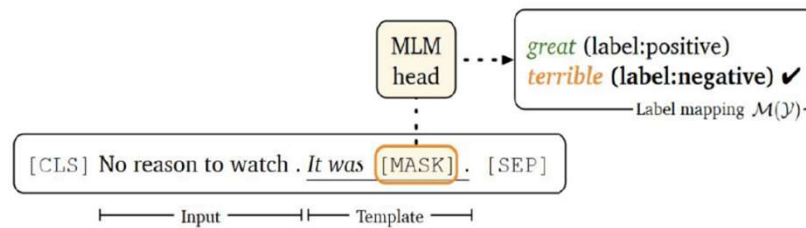


Image Source: Making Pre-trained Language Models Better Few-shot Learners, Gao, et al. 2021

PROMPT BASED FINE TUNING: TASKS

SST-2: sentiment analysis.

- E.g. **S1** = "The movie is ridiculous". **Label**: negative.
- Manual prompt:

Template	Label words
<code><S₁> It was [MASK] .</code>	great/terrible

SNLI: Natural Language Inference

- **S1** = "A soccer game with multiple males playing". **S2** = "Some men are playing sport". **Label**: Entailment.
- Manual prompt:

Template	Label words
<code><S₁> ? [MASK] , <S₂></code>	Yes/Maybe/No

PROMPTING

GPT-3 

Very very large language
Unchanged Model Parameters
Usually human-designed **prompts** and **demonstrations**

PET 

LM-BFF 

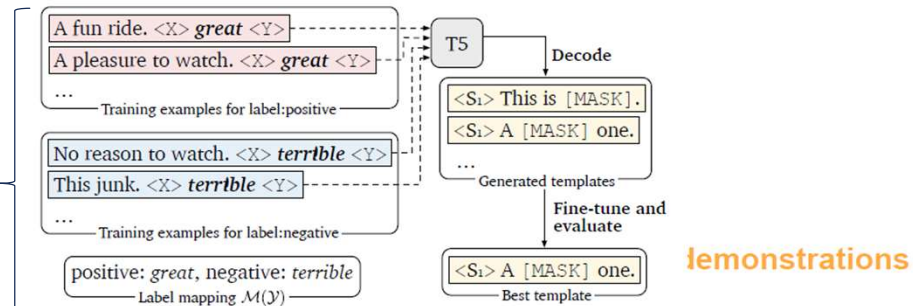


Figure 2: Our approach for template generation.

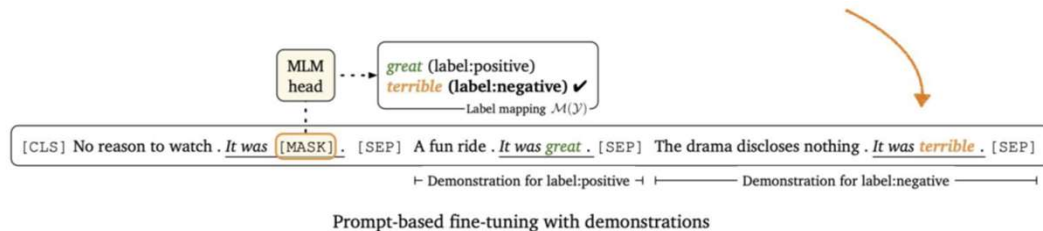
DATASETS

Category	Dataset	$ \mathcal{Y} $	Type	Labels (classification tasks)
single-sentence	SST-2	2	sentiment	positive, negative
	SST-5	5	sentiment	v. pos., positive, neutral, negative, v. neg.
	MR	2	sentiment	positive, negative
	CR	2	sentiment	positive, negative
	MPQA	2	opinion polarity	positive, negative
	Subj	2	subjectivity	subjective, objective
	TREC	6	question cls.	abbr., entity, description, human, loc., num.
	CoLA	2	acceptability	grammatical, not_grammatical
sentence-pair	MNLI	3	NLI	entailment, neutral, contradiction
	SNLI	3	NLI	entailment, neutral, contradiction
	QNLI	2	NLI	entailment, not_entailment
	RTE	2	NLI	entailment, not_entailment
	MRPC	2	paraphrase	equivalent, not_equivalent
	QQP	2	paraphrase	equivalent, not_equivalent
	→ STS-B	\mathcal{R}	sent. similarity	-

Source: Making Pre-trained Language Models Better Few-shot Learners, Gao, et al. 2021

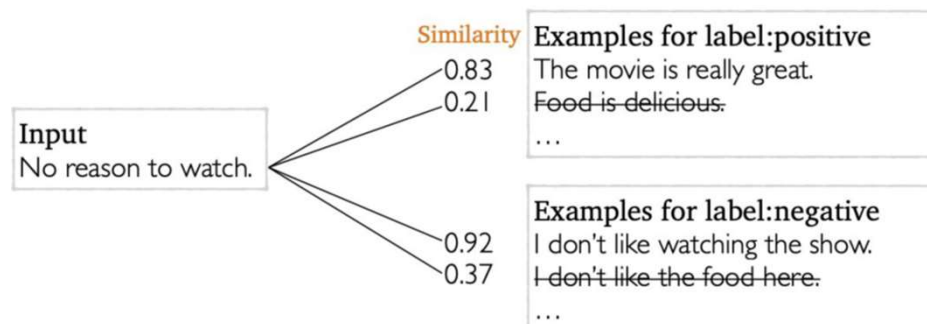
PROMPT BASED ON DEMONSTRATION

- Demonstration is based on the idea that in few-shot learning you can exemplify a task by using instances from the training set that demonstrate how to solve a task

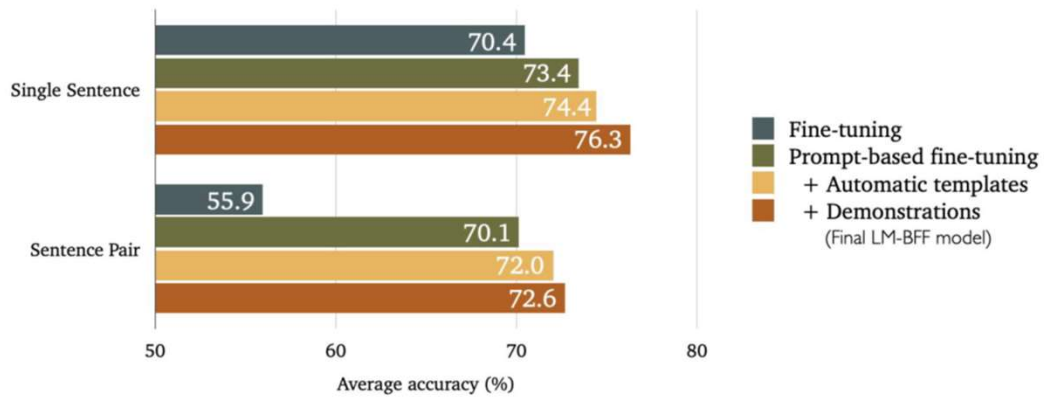


- Selective demonstration (INTUITION): Apply **demonstrations** that are **semantically close** to the input for optimal results

EXAMPLES OF DEMONSTRATIONS



PROMPTING WITH DEMONSTRATIONS



APPLICAZIONI DELLE RETI NEURALI: USE CASES

IMMAGINI: OBJECT DETECTION, ENCODING, MAP COLOURING, RAGS



TASK: TABLES-TO-TEXT

3 Datasets

- DART: Triplets similar to WebNLG, but bigger and on all Wikipedia tables. (Open domain)
- [\(Nan et al. 2021\)](#)

Parent-child relations provided by internal annotator →

Surface realization provided by internal / MTurk annotator →


[TITLE]: NFL Europe Stadiums

Team	Stadium	Capacity	Opened	City
Amsterdam Admirals	Amsterdam Arena	51,859	1996	Amsterdam, The Netherlands
Amsterdam Admirals	Olympisch Stadion	31,600	1928	Amsterdam, The Netherlands
Barcelona Dragons	Mini Estadi	15,276	1982	Barcelona, Spain

→ "The Amsterdam Admirals play in the Olympisch Stadion, which opened in 1928."

TASK: NEURAL AUTHORING

T write a biblical verse in the style of the king james bible explaining how to remove a peanut butter sandwich from a VCR

 And it came to pass that a man was troubled by a peanut butter sandwich, for it had been placed within his VCR, and he knew not how to remove it.

And he cried out to the Lord, saying, "Oh, Lord, how can I remove this sandwich from my VCR, for it is stuck fast and will not budge?"

And the Lord spoke unto him, saying, "Fear not, my child, for I shall guide thy hand and show thee the way. Take thy butter knife, and carefully insert it between the sandwich and the VCR, and gently pry them apart. And with patience and perseverance, the sandwich shall be removed, and thy VCR shall be saved."

And the man did as the Lord commanded, and lo and behold, the sandwich was removed from the VCR, and the man was saved.

And the Lord said, "Verily I say unto thee, seek not to put thy peanut butter sandwiches in thy VCR, for it is not a suitable place for such things. Rather, keep thy sandwiches in thy refrigerator or on thy plate, where they belong."

And the man heeded the Lord's words, and from that day forth, he kept his sandwiches in their proper place, and was saved from trouble and woe. Amen.

SUMMARIZATION

- Given a long (fragment of) document generate few summary sentences
- XSUM data set
 - BBC News articles
 - Depends on several parts of the text, i.e., not just few sentence selection
 - Designed to amplify the abstractive power of the transformer decoder
 - (Narayan et al, 2018)

SUMMARY: *A man and a child have been killed after a light aircraft made an emergency landing on a beach in Portugal.*

DOCUMENT: Authorities said the incident took place on Sao Joao beach in Caparica, south-west of Lisbon.

The National Maritime Authority said a middle-aged man and a young girl died after they were unable to avoid the plane.

[6 sentences with 139 words are abbreviated from here.]

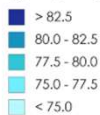
Other reports said the victims had been sunbathing when the plane made its emergency landing.

[Another 4 sentences with 67 words are abbreviated from here.]

Video footage from the scene carried by local broadcasters showed a small recreational plane parked on the sand, apparently intact and surrounded by beachgoers and emergency workers.

[Last 2 sentences with 19 words are abbreviated.]

Life expectancy in Europe (years, 2016)



Created with mapchart.net

CHAT GPT4

Describe the image for me:
https://en.wikipedia.org/wiki/List_of_European_countries_by_life_expectancy#/media/File:Life_expectancy_in_Europe.png

The image depicts a color-coded map of Europe, with different shades of green and yellow representing the life expectancy of each country. The map is titled "Life expectancy in Europe" and shows a gradient from lighter to darker shades of green, representing higher life expectancies.

The countries with the highest life expectancy, such as Switzerland, Spain, Italy, and France, are shown in the darkest shades of green, indicating life expectancies of over 80 years. The countries with lower life expectancies, such as Russia, Belarus, and Ukraine, are depicted in yellow and lighter shades of green, indicating life expectancies of under 75 years.

The map also includes a legend that provides a color-coded scale and corresponding life expectancy ranges, allowing viewers to easily interpret the data. Overall, the image provides a visual representation of the variation in life expectancy across different European countries, highlighting the disparities between countries and regions.

OBJECT DETECTION WITH CNNs

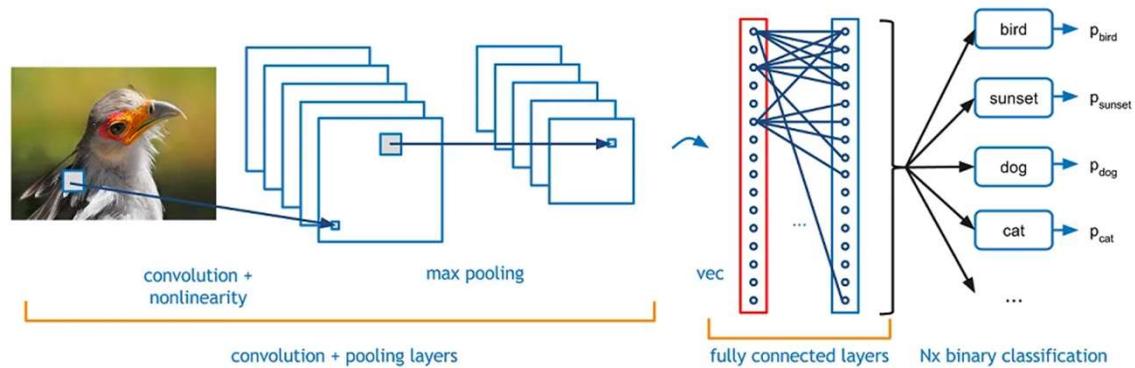
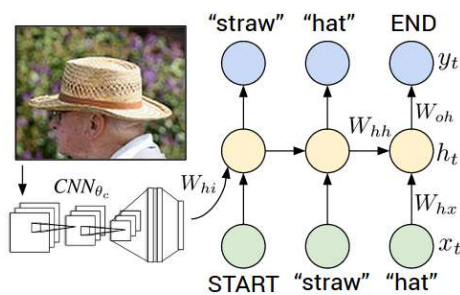
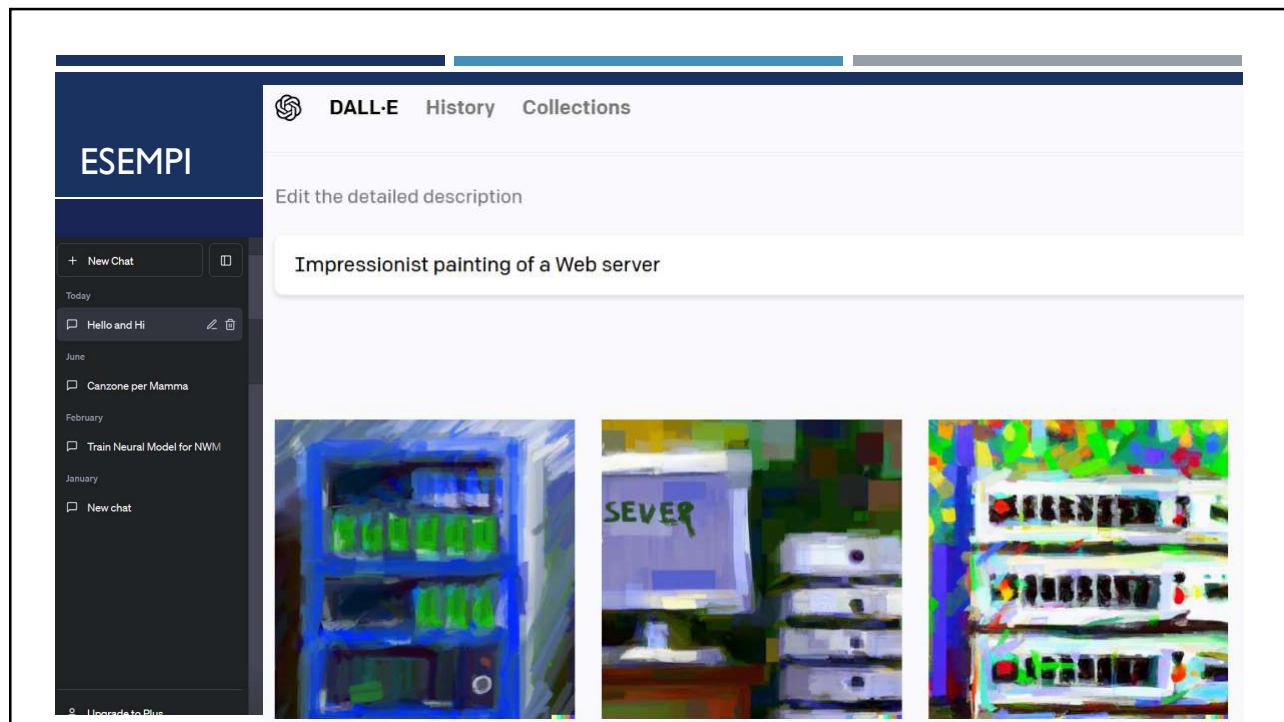


IMAGE CAPTIONING: ADVANCED ARCHITECTURES

- Image to captions
 - Convolutional Neural Network to learn a representation of the image
 - (Bi-directional) Recurrent Neural Network to generate a caption describing the image
 - its input is the representation computed from the CNN
 - its output is a sequence of words, i.e. the caption



“baseball player is throwing ball in game.”



NEURAL ENCODING-DECODING FOR DALL-E

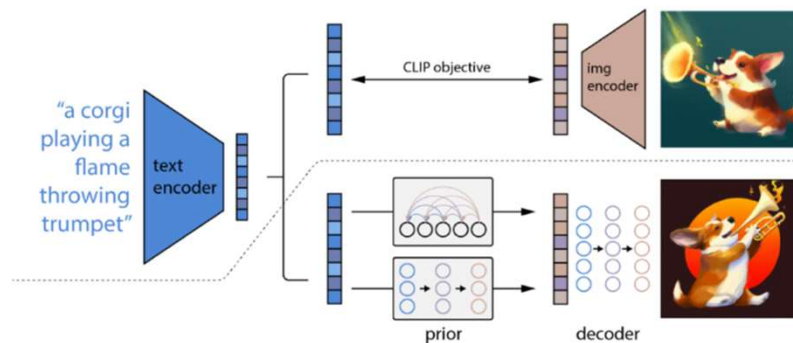


Figure 2: A high-level overview of unCLIP. Above the dotted line, we depict the CLIP training process, through which we learn a joint representation space for text and images. Below the dotted line, we depict our text-to-image generation process: a CLIP text embedding is first fed to an autoregressive or diffusion prior to produce an image embedding, and then this embedding is used to condition a diffusion decoder which produces a final image. Note that the CLIP model is frozen during training of the prior and decoder.

MULTIMODAL NNS: INTEGRATING IMAGE AND TEXTS IN CLIP

- Object Recognition usually employs ad hoc training data sets implying ad hoc CNN models
- The paper (*) demonstrates that the simple pre-training task of predicting **which caption goes with which image** is an efficient and scalable way to learn SOTA image representations from scratch on a dataset of 400 million (image, text) pairs collected from the internet.
- After pre-training**, natural language is used to reference learned visual concepts (or describe new ones) enabling zero-shot transfer of the model to downstream tasks.
- Zero-shot learning**: solving an object recognition task without ANY training example
- The IDEA: Optimizing the behaviours of image classifiers trained with natural language supervision at large scale.**

(*) Learning Transferable Visual Models From Natural Language Supervision, Redford et al, 2021, <https://arxiv.org/abs/2103.00020v1>

CLIP (CONTRASTIVE LANGUAGE-IMAGE PRE-TRAINING)

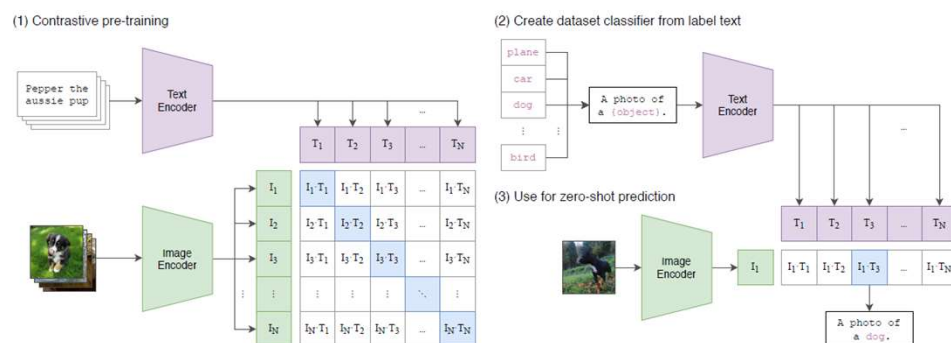
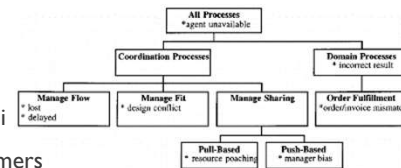


Figure 1. Summary of our approach. While standard image models jointly train an image feature extractor and a linear classifier to predict some label, CLIP jointly trains an image encoder and a text encoder to predict the correct pairings of a batch of (image, text) training examples. At test time the learned text encoder synthesizes a zero-shot linear classifier by embedding the names or descriptions of the target dataset's classes.

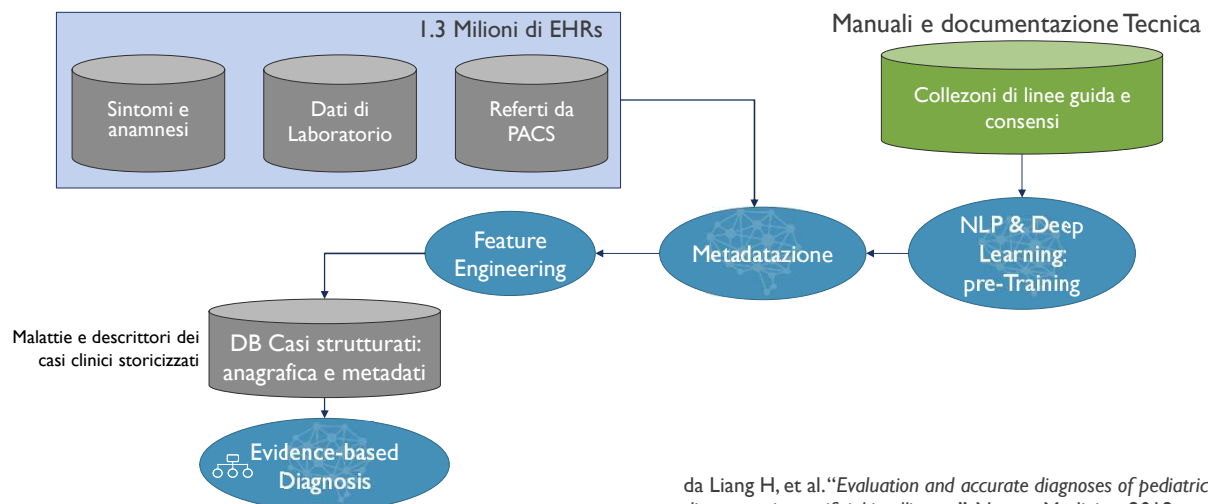
(*) Learning Transferable Visual Models From Natural Language Supervision, Redford et al, 2021, <https://arxiv.org/abs/2103.00020v1>

BANKING: ABILABERT IN DECODE

- 5 banche coordinate da ABILAB
- Una Process Taxonomy condivisa e differenti Basi di Dati Documentali
- Automatic Text-driven Process Mapping basato su reti neurali Trasformers



DIAGNOSI MALATTIE PEDIATRICHE: UN WORKFLOW ORIENTATO AL ML



da Liang H, et al. "Evaluation and accurate diagnoses of pediatric diseases using artificial intelligence", Nature Medicine, 2019

MEDICAL INFORMATION EXTRACTION

INPUT: "Si **osserva una lesione** nel **lobo superiore sinistro** del **polmone del paziente** ..."

INPUT

Word Embedding

QUERIES

Q: Is the left upper lobe of the lung detectable?

Q: Is there a mass in the left upper lobe?

Q: Is there a detectable lesion in the left upper lobe?

Q: Is there a detectable obstruction in the bronchus?

Q: Is there an abnormality in the bronchus?

ANSWERS

A: 1

A: 1

A: 1

A: 0

A: 0

EVIDENCE BASED DIAGNOSIS: RISULTATI (11,926 PAZIENTI)

Table 2 | Illustration of diagnostic performance of our AI model and physicians

Disease conditions	Our model	Physicians				
		Physician group 1	Physician group 2	Physician group 3	Physician group 4	Physician group 5
Asthma	0.920	0.801	0.837	0.904	0.890	0.935
Encephalitis	0.837	0.947	0.961	0.950	0.959	0.965
Gastrointestinal disease	0.865	0.818	0.872	0.854	0.896	0.893
Group: 'Acute laryngitis'	0.786	0.808	0.730	0.879	0.940	0.943
Group: 'Pneumonia'	0.888	0.829	0.767	0.946	0.952	0.972
Group: 'Sinusitis'	0.932	0.839	0.797	0.896	0.873	0.870
Lower respiratory	0.803	0.803	0.815	0.910	0.903	0.935
Mouth-related diseases	0.897	0.818	0.872	0.854	0.896	0.893
Neuropsychiatric disease	0.895	0.925	0.963	0.960	0.962	0.906
Respiratory	0.935	0.808	0.769	0.89	0.907	0.917
Systemic or generalized	0.925	0.879	0.907	0.952	0.907	0.944
Upper respiratory	0.929	0.817	0.754	0.884	0.916	0.916
Root	0.889	0.843	0.863	0.908	0.903	0.912
Average F1 score	0.885	0.841	0.839	0.907	0.915	0.923

TRANSFORMER BASED FORECASTING

- Informer architecture proposed in (Zhou et al., 2020)
- Adopt an Encoder-Decoder architecture
- Force self attention to capture most of the dependencies within the input time series
- Use a variant ProbSparse of the attention mechanism for minimizing parameter size
- Stacks a number of encoding layers (in a CNN style) to reduce the size of the synthetic output to the decoder

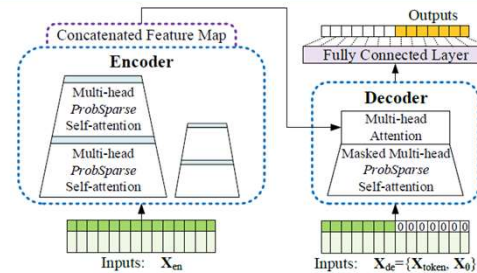


Figure 2: Informer model overview. Left: The encoder receives massive long sequence inputs (green series). We replace canonical self-attention with the proposed *ProbSparse* self-attention. The blue trapezoid is the self-attention distilling operation to extract dominating attention, reducing the network size sharply. The layer stacking replicas increase robustness. Right: The decoder receives long sequence inputs, pads the target elements into zero, measures the weighted attention composition of the feature map, and instantly predicts output elements (orange series) in a generative style.

TRANSFORMER BASED FORECASTING

- Uniform Representation of temporal data in the Informer architecture
 - Input values
 - Local time stamp embeddings
 - Global Time stamp embeddings

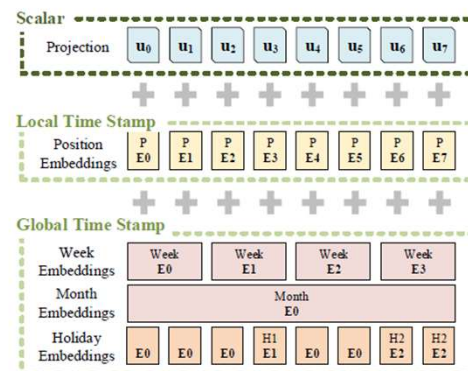
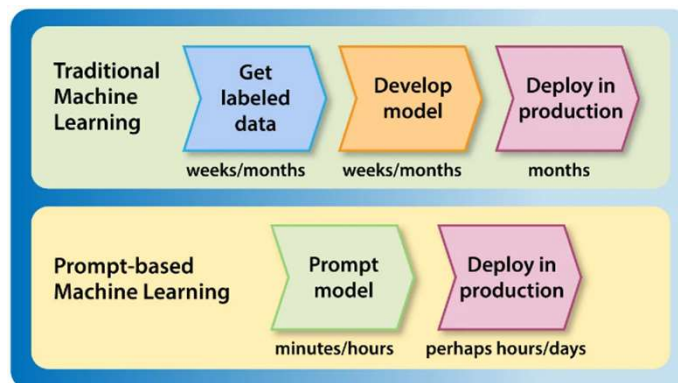


Figure 6: The input representation of Informer. The inputs's embedding consists of three separate parts, a scalar projection, the local time stamp (Position) and global time stamp embeddings (Minutes, Hours, Week, Month, Holiday etc.).

LARGE LANGUAGE MODELS

TRENDS

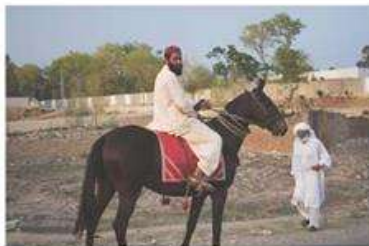
TRENDS ...



RIFLESSIONI

■ Competenza, Razionalità ed Onniscienza

- Un sistema di AI generativa ha una **SIGNIFICATIVA COMPETENZA LINGUISTICA** in analogia con i parlanti delle diverse lingue in cui esso è stato addestrato
- E' **RAZIONALE** in senso linguistico poiché conosce le regole della comunicazione e le usa in modo *utile*
- **NON è ONNISCENTE**
 - Errori di *senso comune*
 - Mostra talvolta incompetenza
 - Non è **esperto** dei diversi domini
- **NON** è sempre **completamente coerente**
 - Allucinazioni



A person on a horse



A person on a horse ?

Raphael - Saint George Fighting the Dragon

Raphael, Public domain, via Wikimedia Commons

LLMS: POTENZIALITÀ E RISCHI

- 💡 Enorme flessibilità nella comprensione e generazione linguistica
- 💡 Capacità di affrontare nuovi task attraverso il **prompting**
- 💡 Forte capacità di specializzazione verso fenomeni semantici specifici (domini, enciclopedie, dati in tempo reale)
- 💡 Facile integrazione con competenze in altri ambiti cognitivi (machine vision)
- 🚫 Forti limiti nella capacità di certificare i comportamenti linguistici
- 🚫 Bulimia computazionale
- 🚫 Limitata analogia con i processi cognitivi

INTEGRAZIONE DI CONOSCENZA E LLMS: RAG MODELS

- Retrieval Augmented Generation
 - A *generation time* si rende disponibile una informazione di contesto che qualifica la risposta
 - Essenziale per task **knowledge intensive**
 - Si applica sia al *pre-training* che al *fine-tuning* ed al *prompting*
 - Ha mostrato di mitigare le allucinazioni

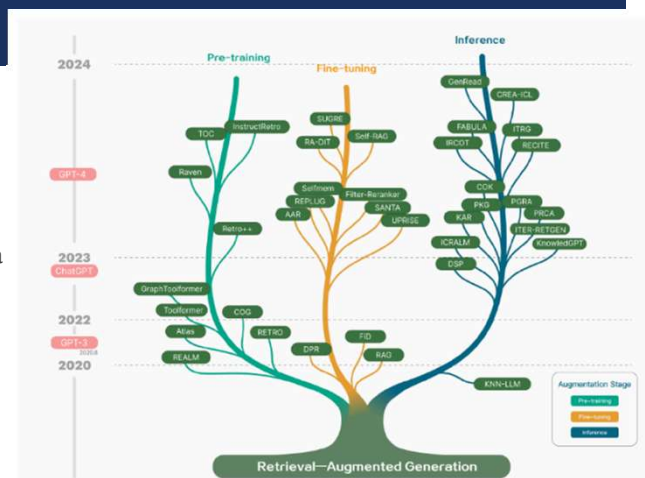


Figure 1: Technology tree of RAG research development featuring representative works

(Lewis et al, 2020) [Retrieval-augmented generation for knowledge-intensive NLP tasks. Proceedings of NIPS, Advances in Neural Information Processing Systems, 2020.](#)

RAG MODELS: IL PROBLEMA E L'IDEA

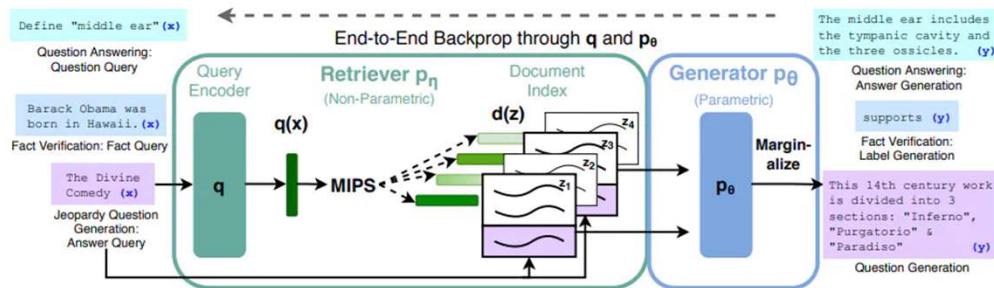
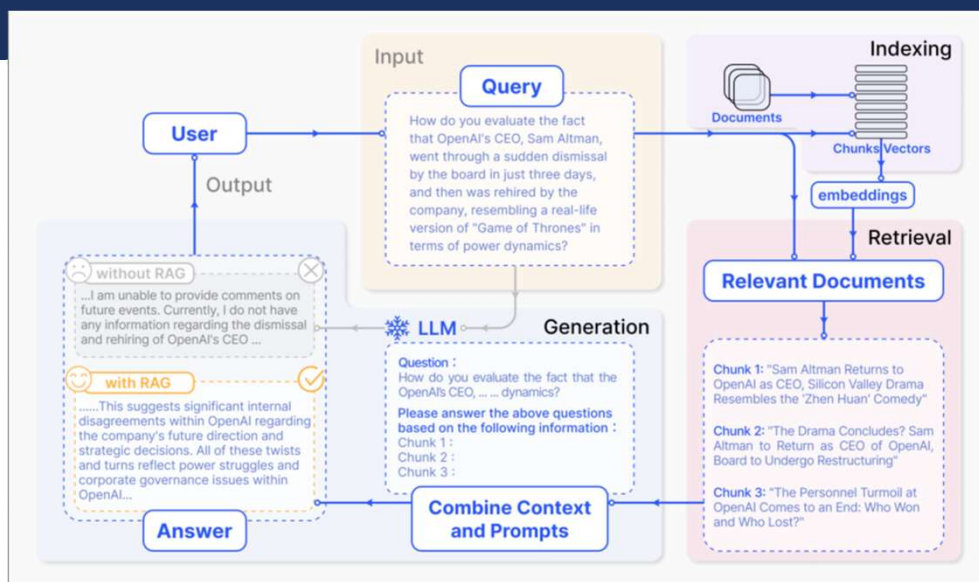


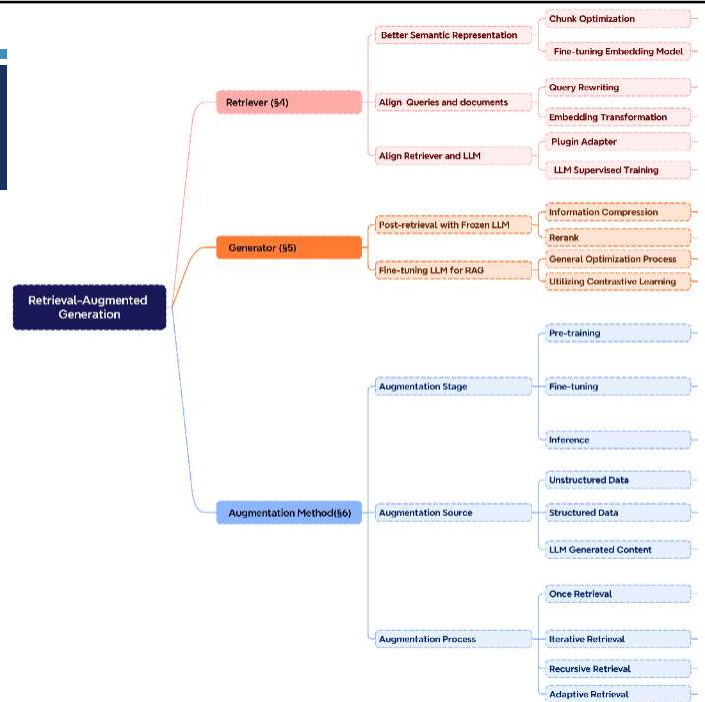
Figure 1: Overview of our approach. We combine a pre-trained retriever (*Query Encoder + Document Index*) with a pre-trained seq2seq model (*Generator*) and fine-tune end-to-end. For query x , we use Maximum Inner Product Search (MIPS) to find the top-K documents z_i . For final prediction y , we treat z as a latent variable and marginalize over seq2seq predictions given different documents.

RAG MODELS: IL FLUSSO INFORMATIVO

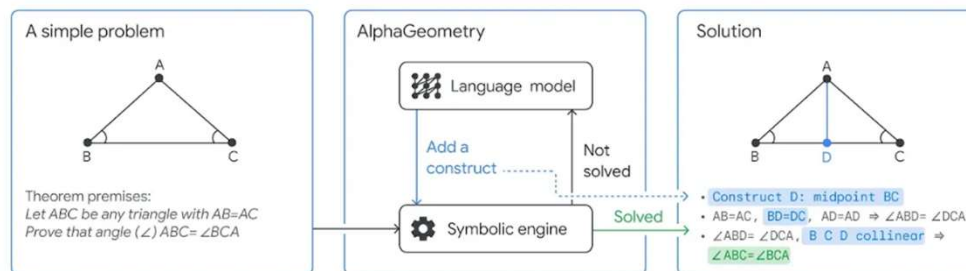


A RAG TAXONOMY

- Research is active in different directions
 - Retrieval
 - Generation
 - Textual, Logical and Procedural Augmentation
- DBs or KG are often explored as information sources



ALPHAGEOMETRY (GOOGLE DEEPMIND, JAN 2024)



Trinh, Trieu H., Wu Yuhuai, Le Quoc V., He He, Luong Thang, Solving olympiad geometry without human demonstrations, *Nature*, 625, 2024.

ALPHAGEOMETRY (GOOGLE DEEPMIND, JAN 2024)

IMO 2015 P3

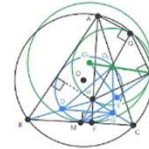
Let ABC be an acute triangle. Let (O) be its circumcircle, H its orthocenter, and F the foot of the altitude from A . Let M be the midpoint of BC . Let Q be the point on (O) such that $QH \perp QA$ and let K be the point on (O) such that $KH \perp KO$. Prove that the circumcircles (O_1) and (O_2) of triangles FKM and GKH are tangent to each other.



AlphaGeometry

Solution

```
[...]
Construct D: midpoint BH [a]
[a], O, midpoint HQ  $\Rightarrow$  BQ  $\parallel$  O, D [20]
[...]
Construct G: midpoint HC [b]
 $\angle GMD = \angle GO, D \Rightarrow M, O, G, D$  cyclic [26]
[...]
[a], [b]  $\Rightarrow$  BC  $\parallel$  DG [30]
[...]
Construct E: midpoint MK [c]
[c]  $\Rightarrow$   $\angle KFC = \angle KO, E$  [104]
[...]
 $\angle PKO_1 = \angle PKO_2 \Rightarrow KO_1 \parallel KO_2$  [109]
[109]  $\Rightarrow O, O_1, K$  collinear  $\Rightarrow (O_1), (O_2)$  tangent
```



Problem 3 of the 2015 International Mathematics Olympiad (left) and a condensed version of AlphaGeometry's solution (right). The blue elements are added constructs. AlphaGeometry's solution has 109 logical steps.

Trinh, Trieu H., Wu Yuhuai, Le Quoc V., He He, Luong Thang. Solving olympiad geometry without human demonstrations. *Nature*, 625, 2024.

BIBLIOGRAFIA: TRANSFORMERS

- (Vaswani 2017), Attention is all you need, <https://arxiv.org/abs/1706.03762>
- (Devlin et al 2018), BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, <https://arxiv.org/abs/1810.04805>
- An interesting introduction to the attention mechanism:
 - **All you need to know about 'Attention' and 'Transformers' — In-depth Understanding — Part I**, A. Sarkar, URL: <https://towardsdatascience.com/all-you-need-to-know-about-attention-and-transformers-in-depth-understanding-part-1-552f0b41d021#4c16>
- Other Task specific works:
 - Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. CoRR, abs/1409.0473, 2014.
 - Effective Approaches to Attention-based Neural Machine Translation, Minh-Thang Luong Hieu Pham Christopher D. Manning, 2015, <https://arxiv.org/abs/1508.04025v5>
 - Yoon Kim, Carl Denton, Luong Hoang, and Alexander M. Rush. Structured attention networks. In International Conference on Learning Representations, 2017.

BIBLIOGRAFIA: BEYOND TRANSFORMER

- (Vaswani 2017), Attention is all you need, <https://arxiv.org/abs/1706.03762>
- (Devlin et al 2018), BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, <https://arxiv.org/abs/1810.04805>
- Rocktaschel et al., "Reasoning About Entailment With Neural Attention" (ICLR 2016)
- T5: (Wolf et al, 2019) Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. ArXiv, abs/1910.03771.
- BART Encoding-Decoding: Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., ... & Zettlemoyer, L. (2019). Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. arXiv preprint arXiv:1910.13461. <https://arxiv.org/abs/1910.13461>
- Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, "Improving Language Understanding by Generative Pre-Training", 2019
- Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, Dario Amodei: Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. Advances in neural information processing systems, 33, 1877-1901. <https://arxiv.org/abs/2005.14165>, NeurIPS 2020.
- Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, Paul F. Christiano: Learning to summarize with human feedback. NeurIPS 2022



GRAZIE
DELL'ATTENZIONE

BASILI@INFO.UNIROMA2.IT