DEPARTMENT OF KNOWLEDGE TECHNOLOGIES
Jožef Stefan Institute

FOUNDATION MODELS

Sašo Džeroski
Head of Department of knowledge technologies
Jožef Stefan Institute, Ljubljana, Slovenia

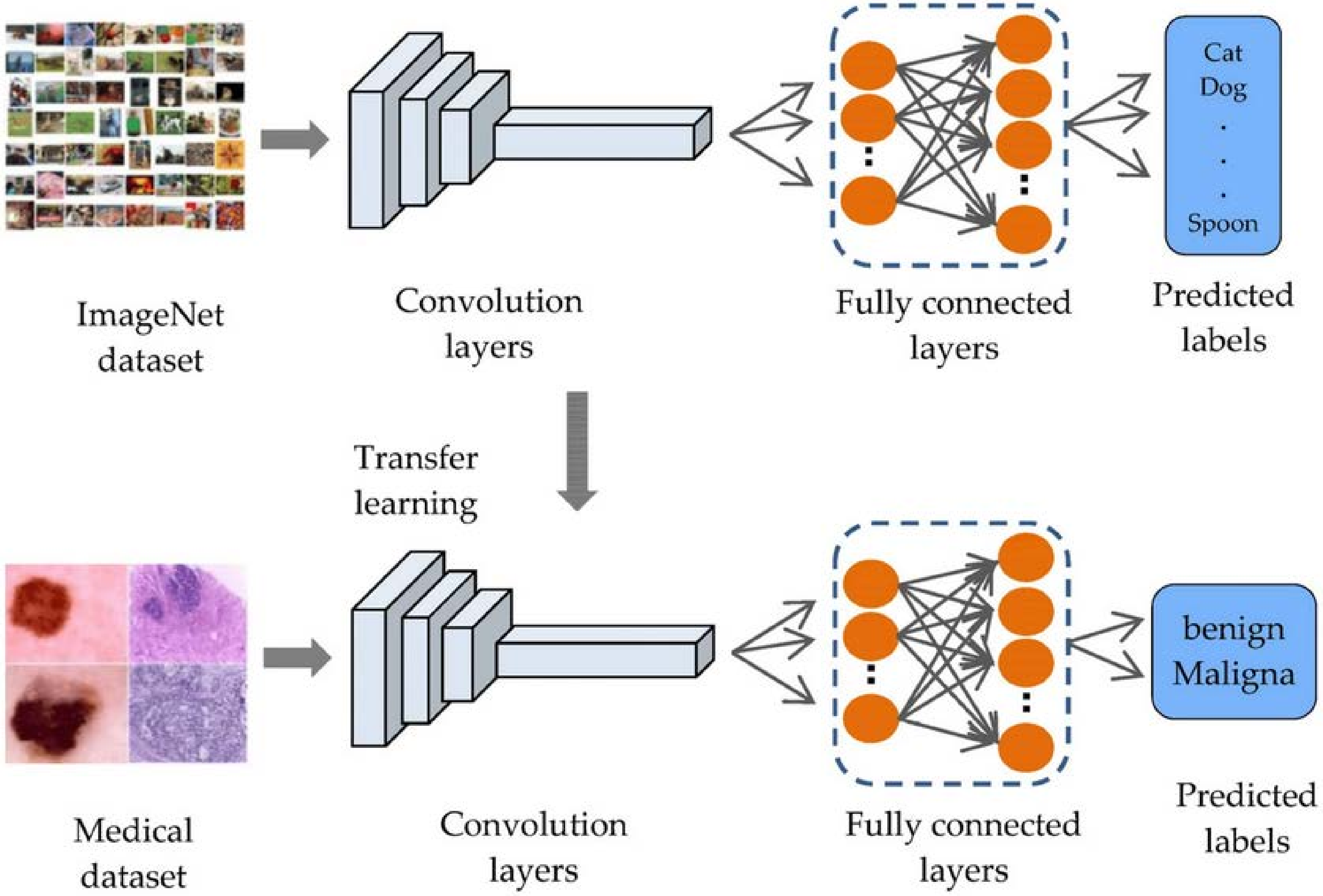
HOW TO DEAL WITH SMALL QUANTITIES OF LABELED DATA

- ***Transfer learning***
  - Learn **(pre-train)** a predictive model on a (similar) domain where large quantities of data are available
  - Use that model as a starting point and **fine-tune** it with the small quantity of data available
- ***Fully supervised learning:*** use only the small labelled dataset
- ***Unsupervised learning:*** no labels available at all
- ***Semi-supervised learning***
  - Use both labeled (few points) and unlabeled data (many points)
- ***Self-supervised learning***
  - Unuspervised: no labeled data points available, but
  - We make up a target/ supervision ourselves
  - And proceed to learn a predictive model

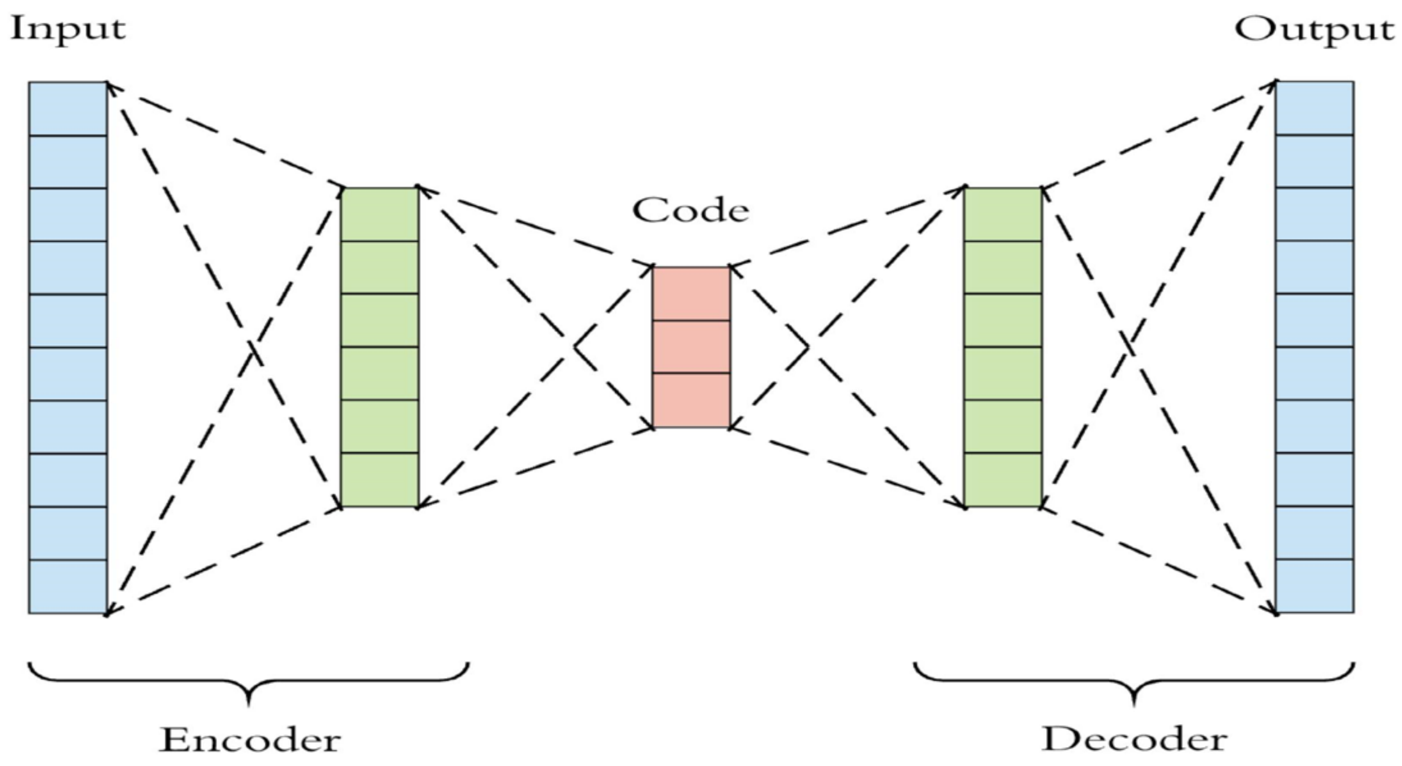
TRANSFER LEARNING

- To be able to learn from small data

- Exploit features/ representation learned on a large dataset
- Adapt the learned neural network (classification head)
  by using data from the small dataset



UNSUPERVISED LEARNING: AUTOENCODERS

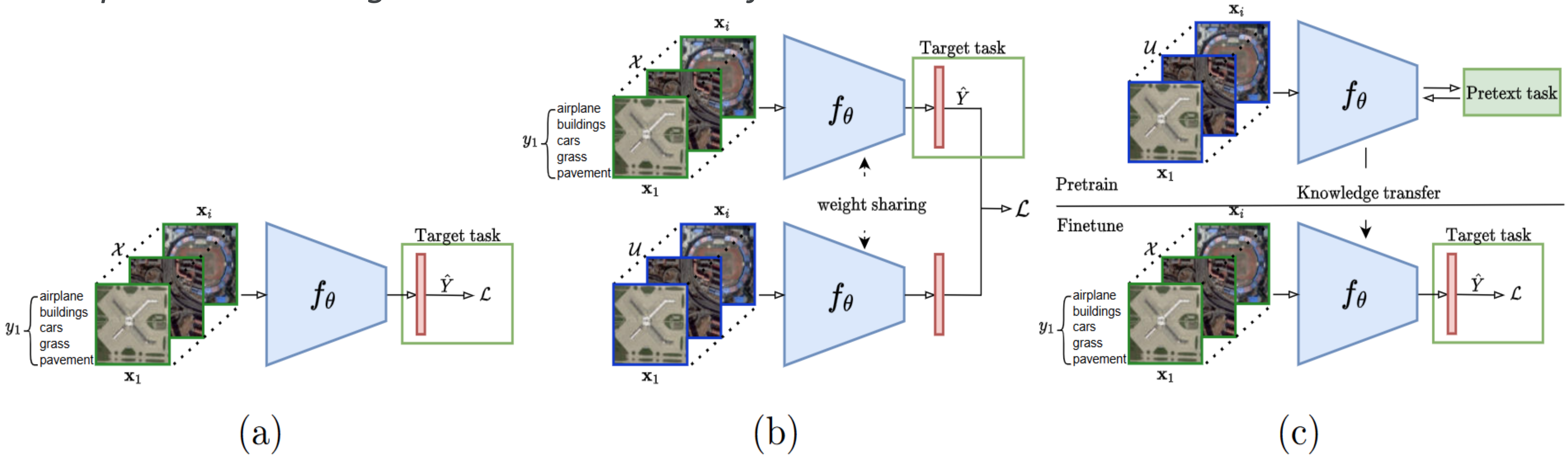


DIFFERENT KINDS OF SUPERVISION IN LEARNING FROM IMAGES

a) *(Fully) Supervised learning: All images are labeled*

b) *Semi-supervised learning: Labeled and unlabeled images used simultaneously*

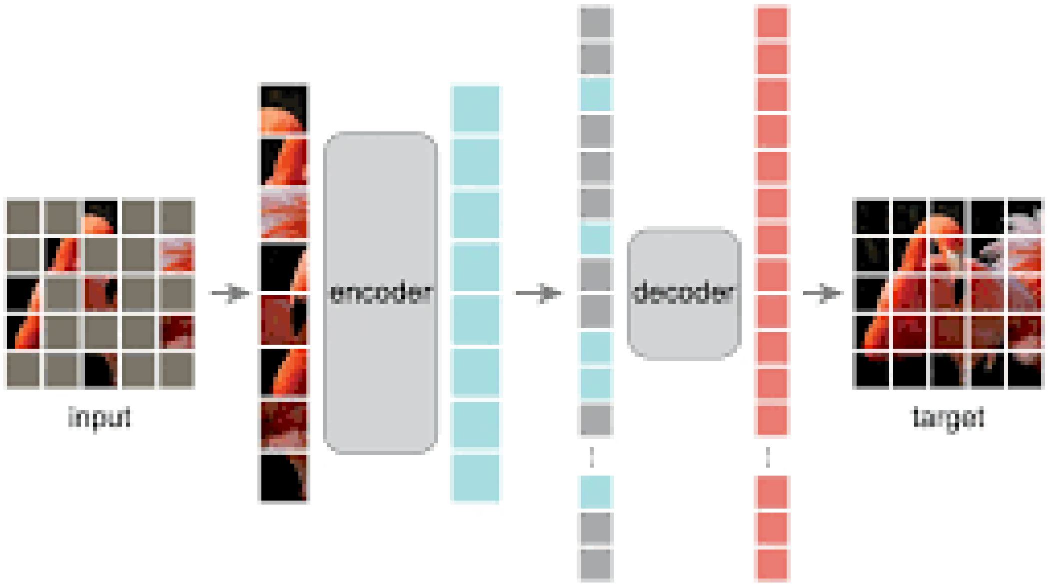
c) *SSL via Self-SL: Unlabeled data used first for self-SL (unsupervised), supervised learning on labeled data then follows*



FOUNDATION MODELS

Foundation models (FMs) *are large models generated by applying ML (deep learning) to a broad collection of data at scale that can be adapted for use in a wide range of downstream tasks*

Foundation models *are trained in a* ***self-supervised*** *manner, on pretext tasks, like reconstructing images from masked versions*

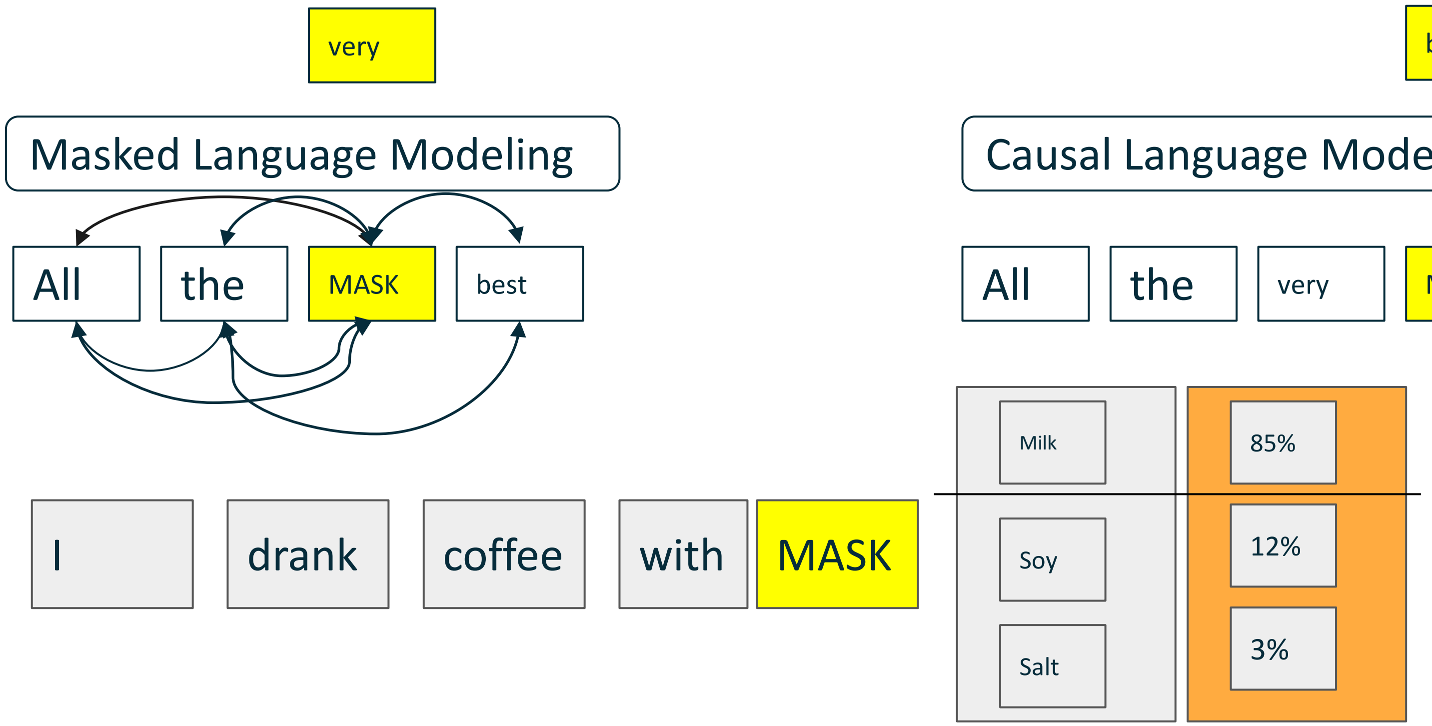


LARGE LANGUAGE MODELS

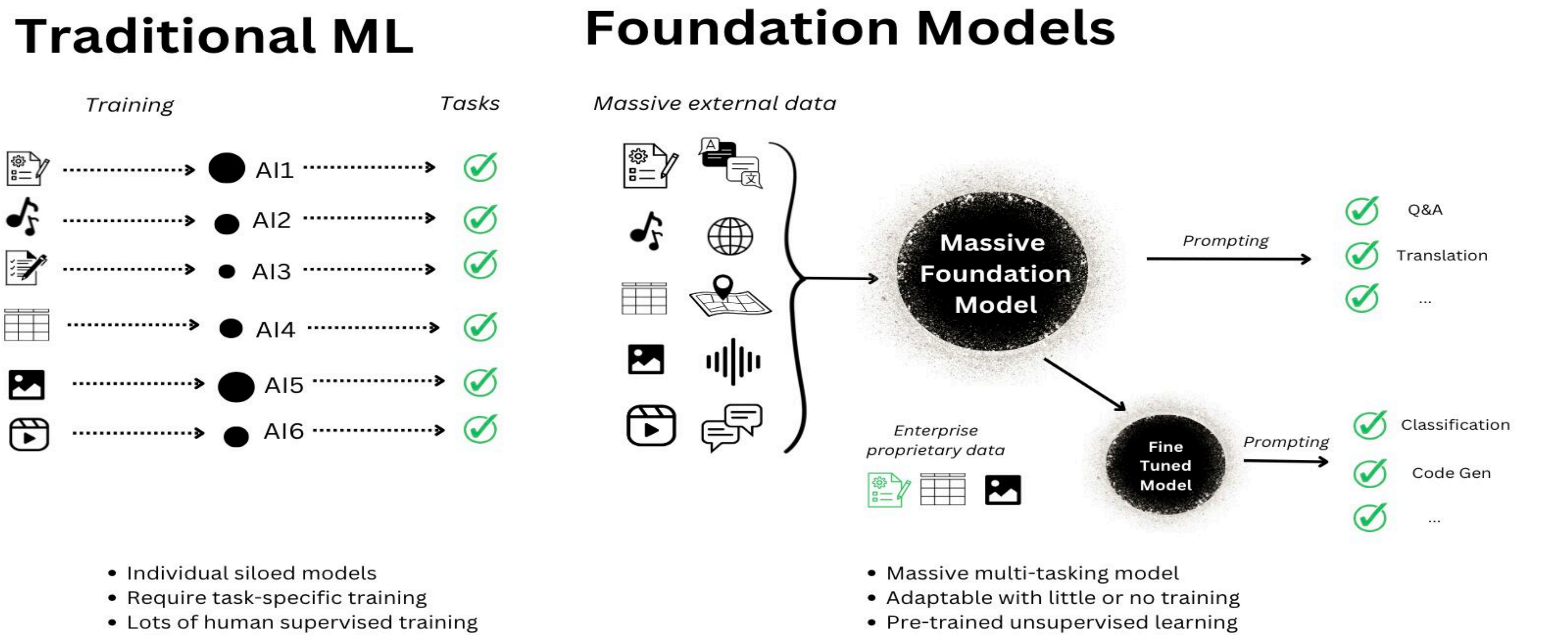
DEPARTMENT OF KNOWLEDGE TECHNOLOGIES
Jožef Stefan Institute

Large language models (LLMs) *are foundation models learned from large collections of text*

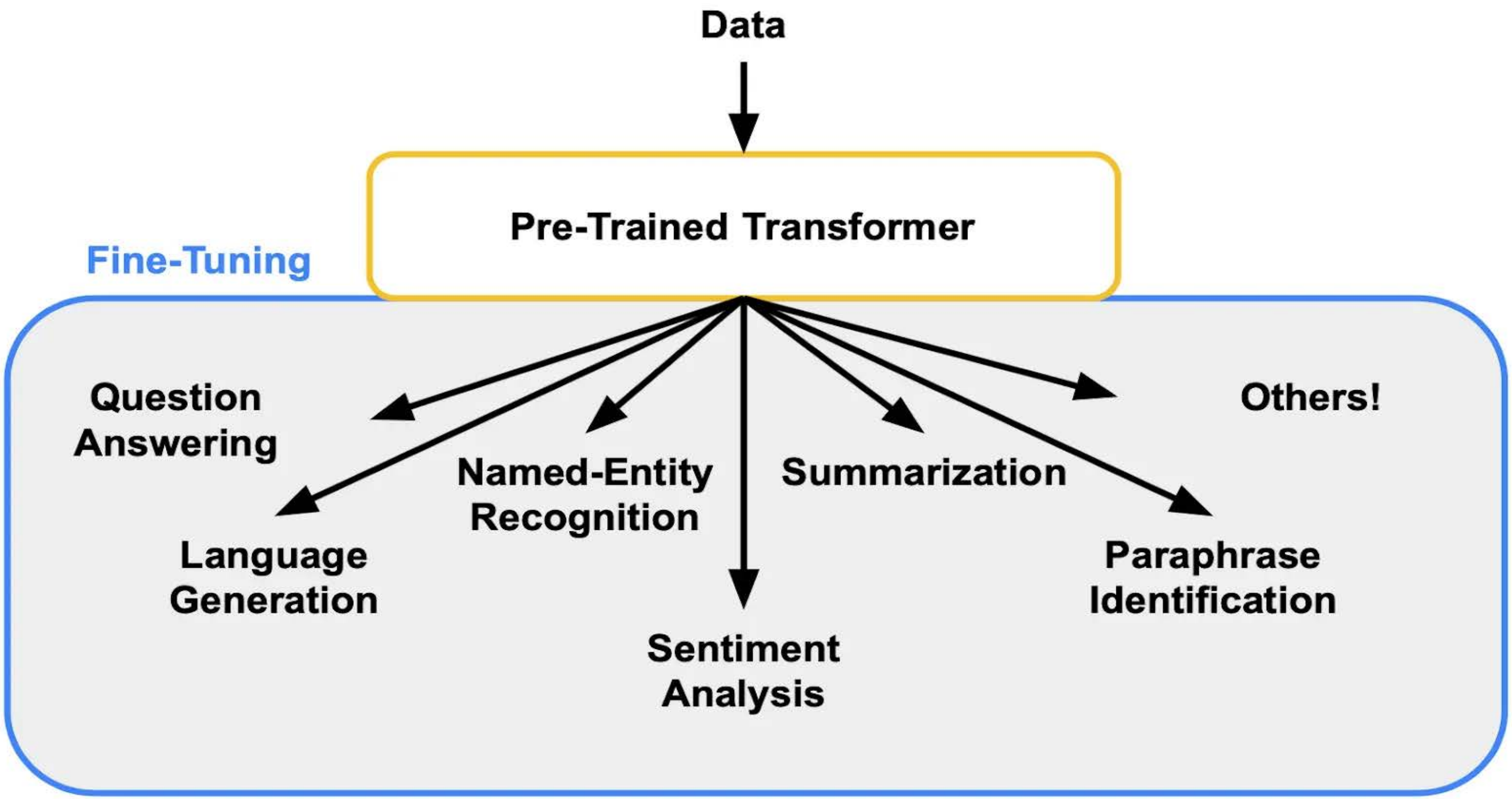
They are also trained on pretext tasks, like reconstructing text from masked versions or predicting the next word in a sequence



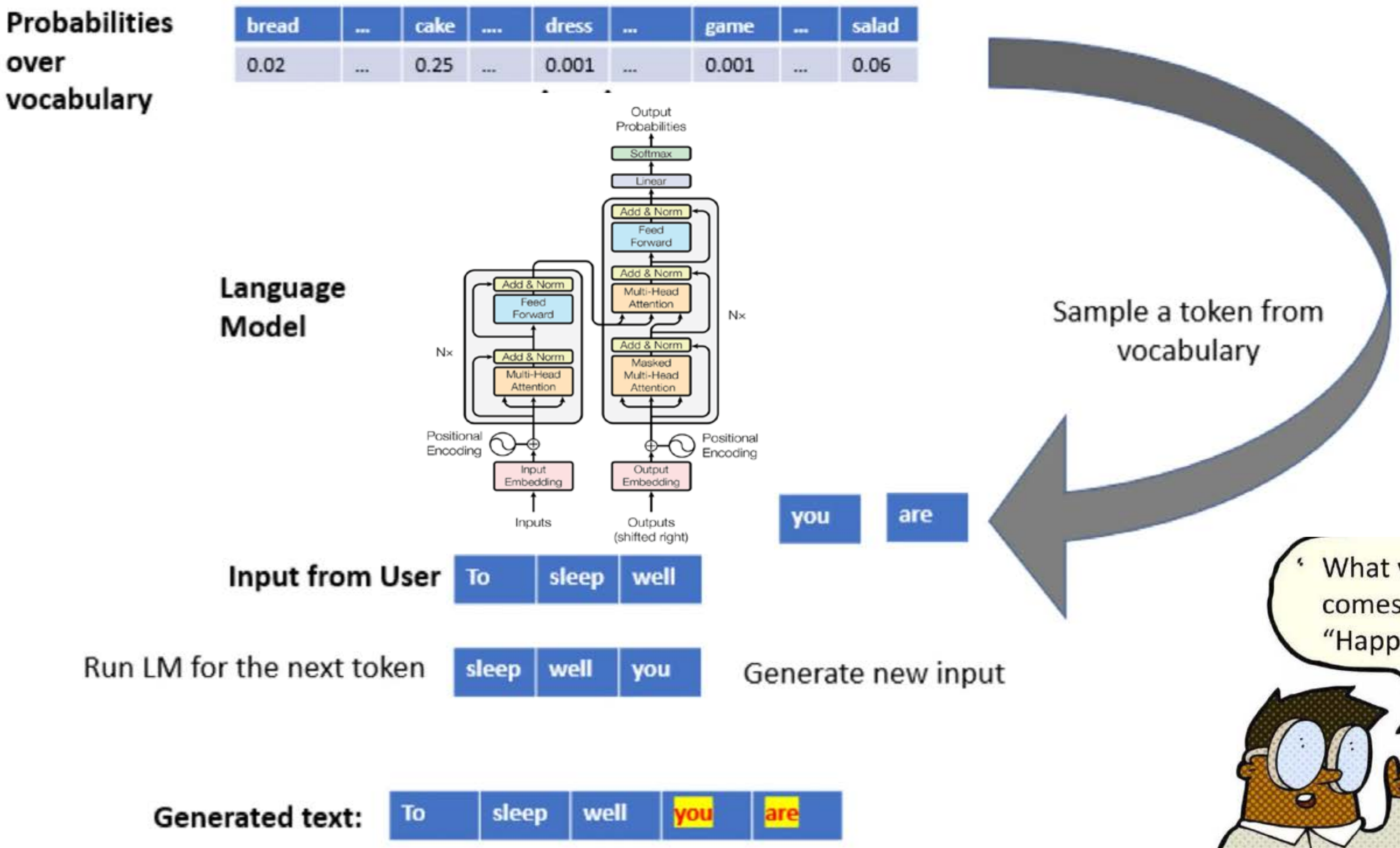
TRAINING MODELS vs. PRE-TRAINING AND FINE-TUNING FMs



DOWNSTREAM TASKS FOR LARGE LANGUAGE MODELS



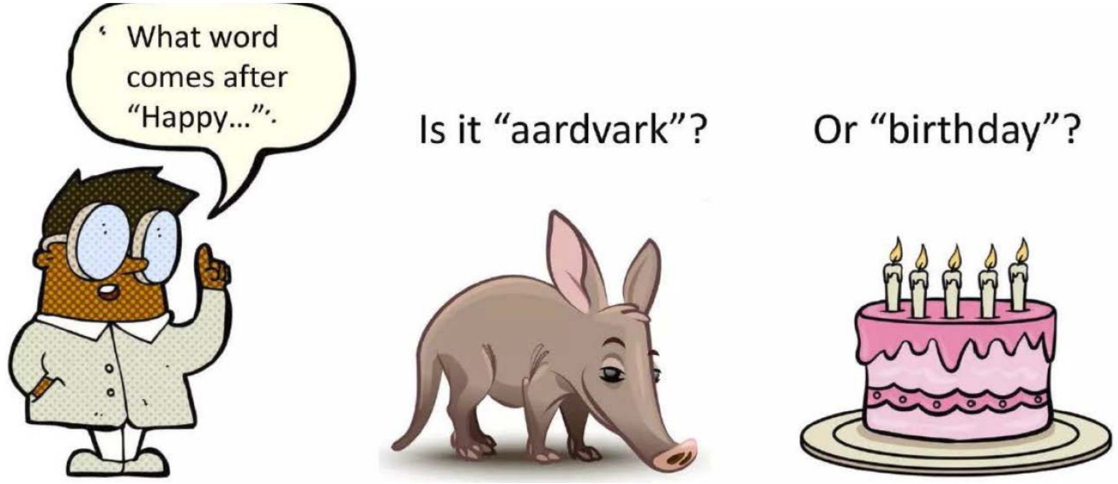
LARGE LANGUAGE MODELS AS GENERATIVE AI



LARGE LANGUAGE MODELS (LLMs) are huge deep NNs with billions of parameters

They are trained on huge amounts of text

They predict the next word after given partial text and can be used to **generate** new text



SEQUENTIAL WORD GUESSING

Me: ChatGPT, I want to explain sequential word guessing to someone. If a sentence starts with "The boy" can you tell me what your next 10 sequential guesses of words would be, one word at a time?

Sure! Here are 10 sequential word guesses, one at a time, based on the sentence starting with "The boy":

1. went, 2. to, 3. the, 4. park, 5. to, 6. play, 7. with, 8. his, 9.friends, 10. and

Keep in mind that as a language model, my guesses are based on patterns and probabilities derived from the text corpus I was trained on. The actual guesses may vary depending on the specific context and the model version being used.

What if the sentence instead begins with "The secret behind GPT-4 is"*?*

Here are 10 sequential word guesses:

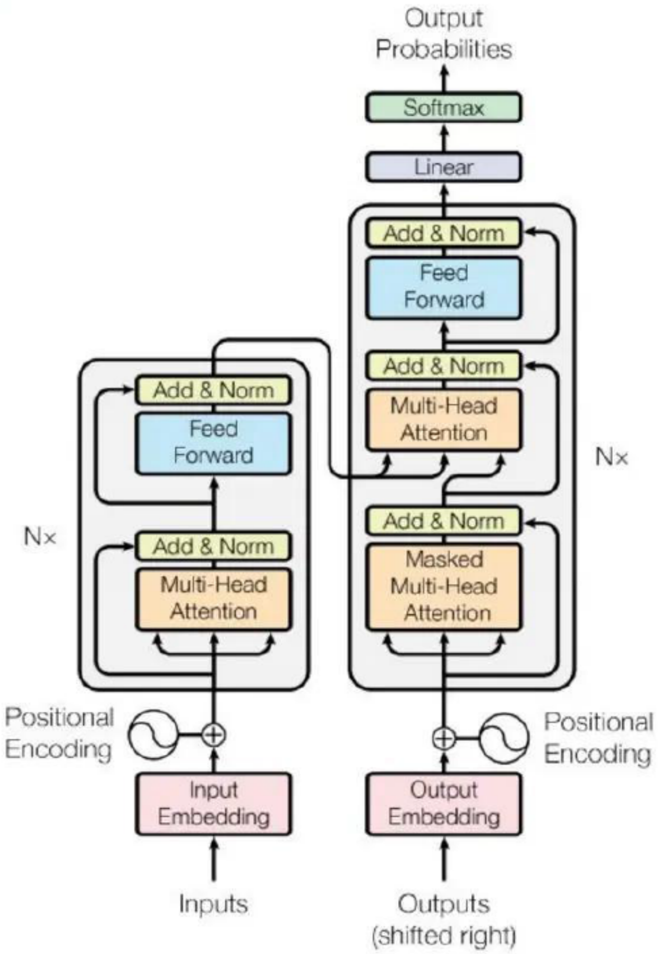
1. "its", 2. "ability", 3. "to", 4. "understand", 5. "and", 6. "generate", 7. "human-like", 8. "text", 9. "based", 10. "on“

Again, these are just guesses based on common patterns and do not represent an actual sentence.

Generative AI (LLMs) with Transformers

What Are Transformers?

Models that learn from a given dataset how to generate new data instances.



Source: Vaswani (2017), Attention Is All You Need (doi.org/10.48550/arXiv.1706.03762)

Generative (deep learning) **models** for understanding and generating text, images and many other types of data.

Transformers analyze chunks of data, called "tokens" and **learn to predict the next token** in a sequence, **based on previous and**, if available, **following tokens**.

The **auto-regressive** concept means that the output of the model, such as the prediction of a word in a sentence, is influenced by the previous words it has generated.

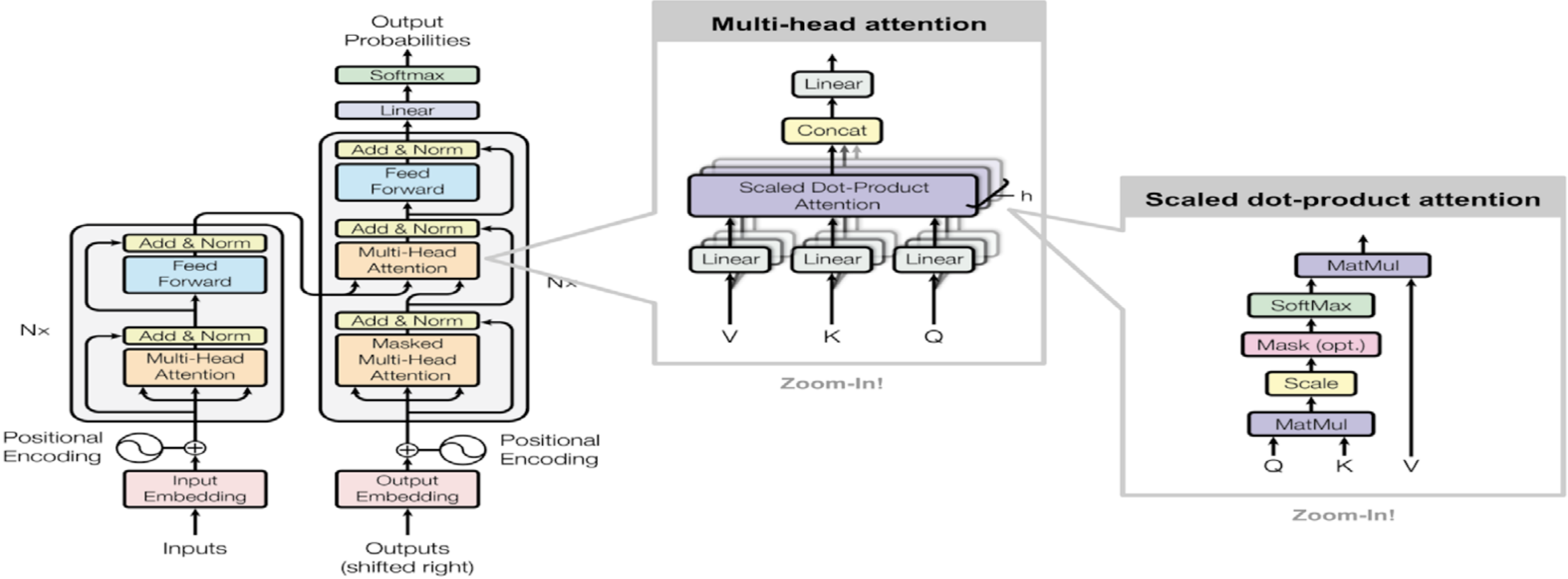
Music—MusicLM (Google) and Jukebox (OpenAI) generate music from text.

Image—Imagen (Google) and DALL.E (OpenAI) generate novel images from text.

Texte—OpenAI's GPT has become widely known, but other players have similar technology (including Google, Meta, Anthropic and others).

Others—Recommender (movies, books, flight destinations), drug discovery...

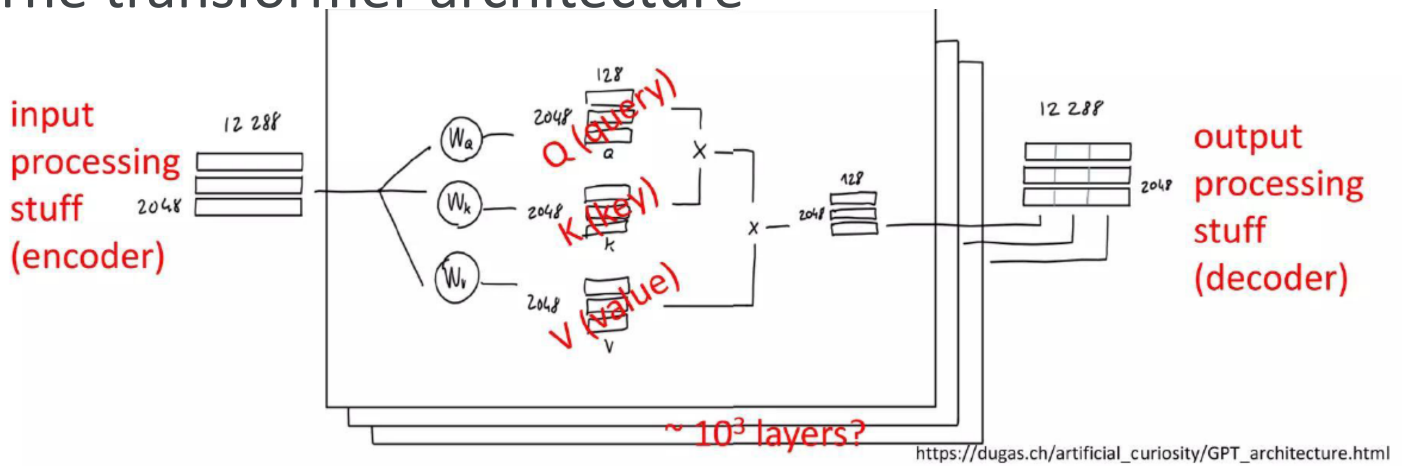
THE NEURAL NETWORKS BEHIND THE FOUNDATION MODELS: THE TRANSFORMER ARCHITECTURE



Very deep neural networks, with billions of parameters, require intensive computation (GPUs) for learning and inference

LARGE LANGUAGE MODELS: KEYS TO SUCCESS

- The transformer architecture



https://dugas.ch/artificial_curiosity/GPT_architecture.html

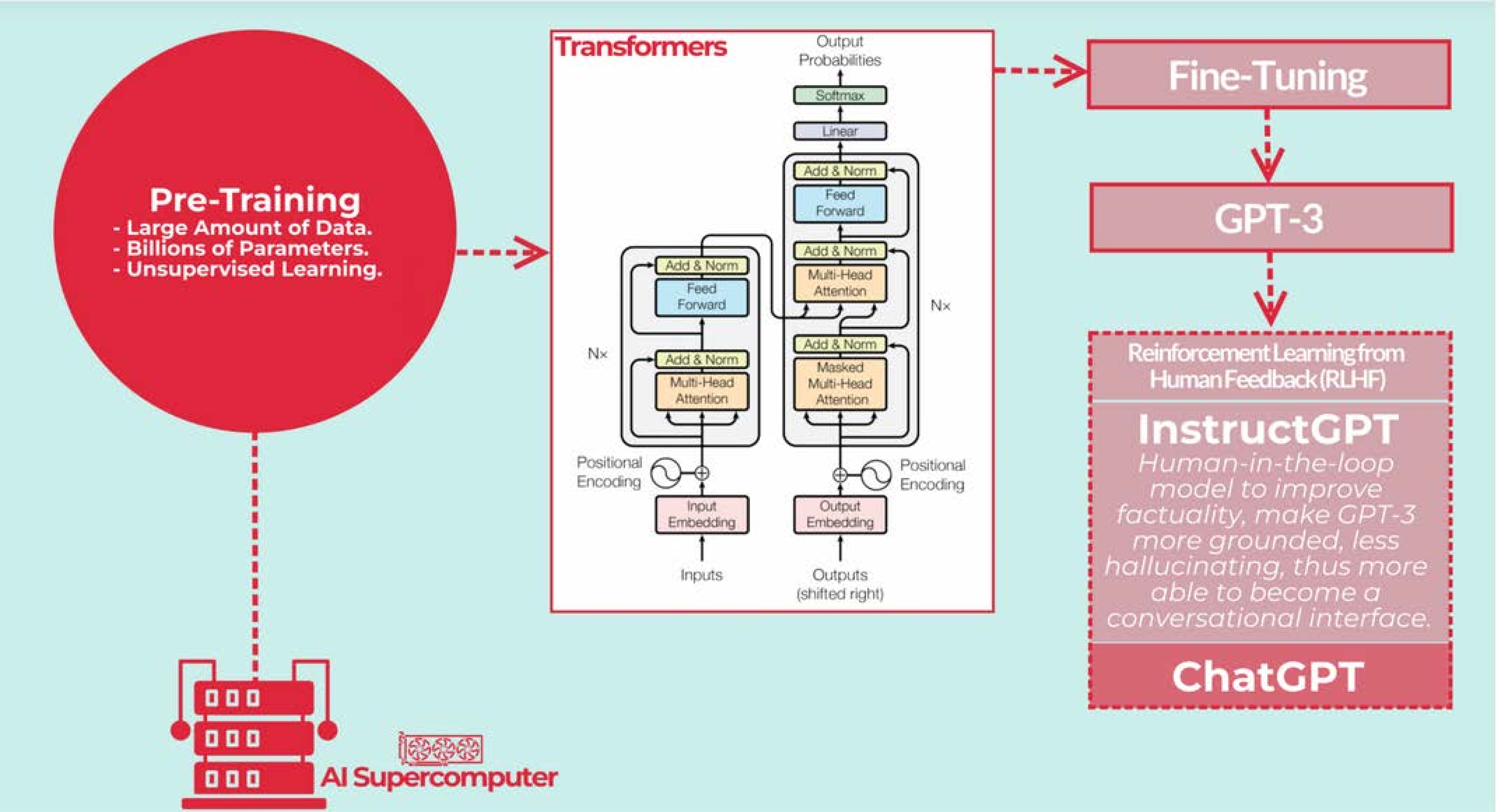
Document retrieval:

- Input text projected onto matrix of possible queries
- Matrix multiply to cross queries with keys (e.g., keywords)
- Matrix multiply to map result from keys to values (e.g., documents)

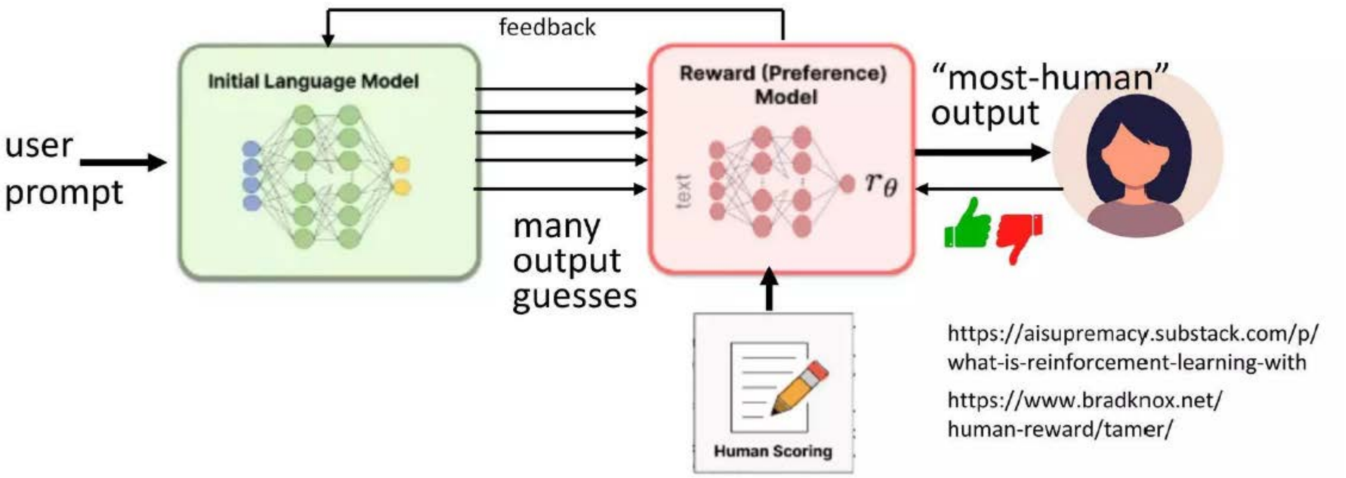
- The brilliant idea of Vaswani et al. (2017, “Attention Is All You Need”) is map all of Q, K, V from the same input.
  - This is “Self-Attention”
- And have all of Q, K, V learned.
- Many layers allows attention to many different levels of structure simultaneously
  - This is “Multi-headed”

- Sheer scale (Number of parameters in the model, Size of input/ training data)
- Reinforcement learning with human feedback (RLHF)

ChatGPT = LLM (GPT) + Conversational Interface

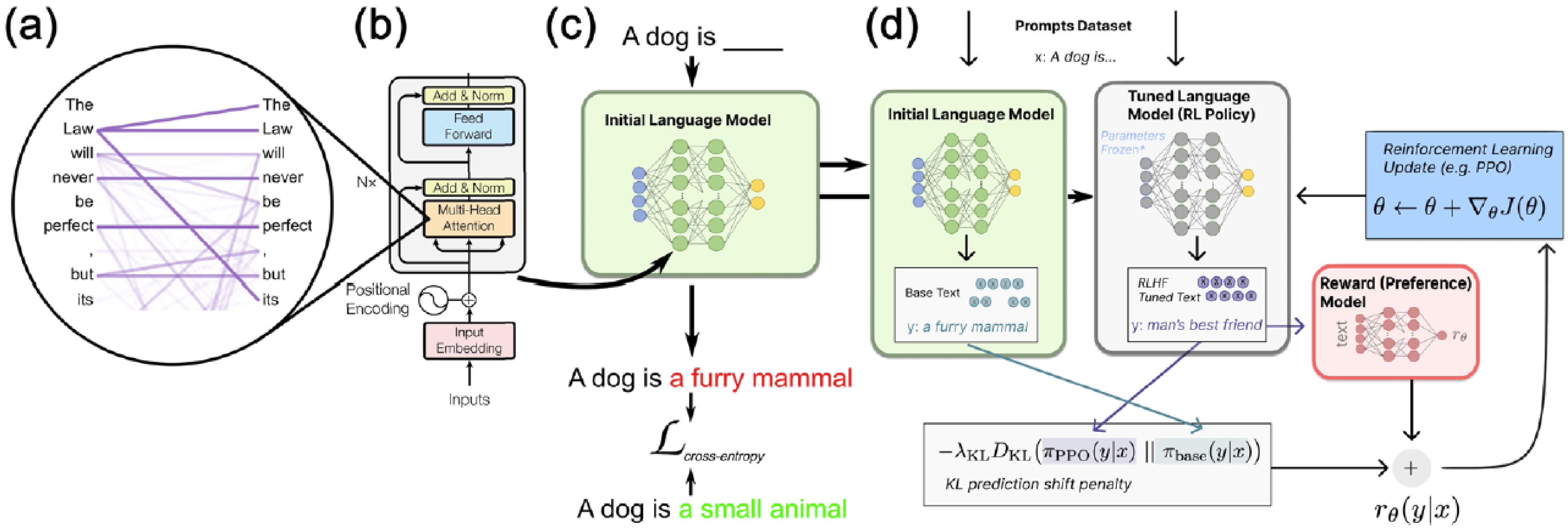


Fine-tuning by reinforcement learning from human feedback (RLHF)

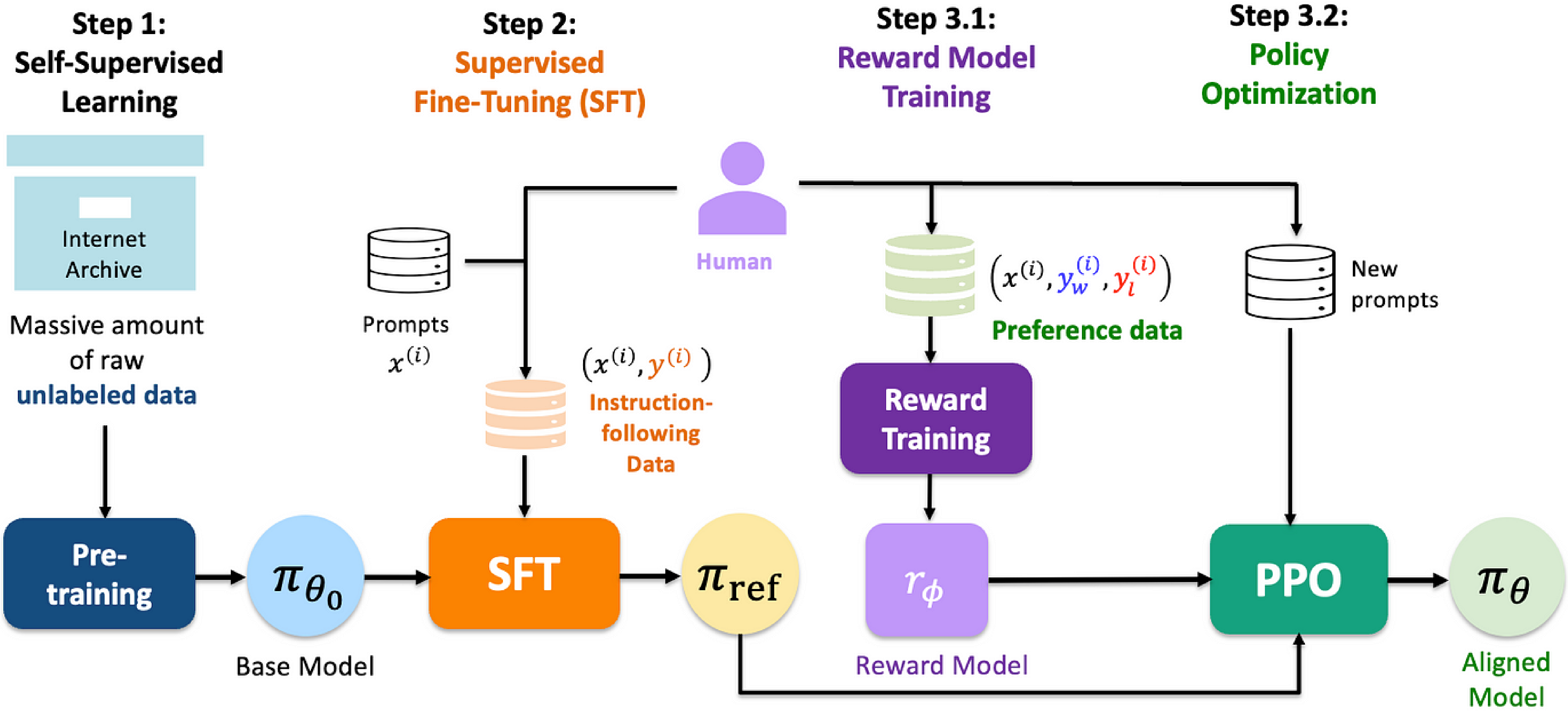


- 1. Humans generate an initial dataset of "typical" queries and "good" responses.
- 2. Humans rank a separate dataset of responses (some good, some bad).
- 3. The reward model is trained on this data.
- 4. The language model trains itself to satisfy the reward model. (How often?)
- 5. The reward model is retrained with (free!) input from hundreds of millions of users. (How often?)

Fine-tuning by reinforcement learning from human feedback (RLHF)

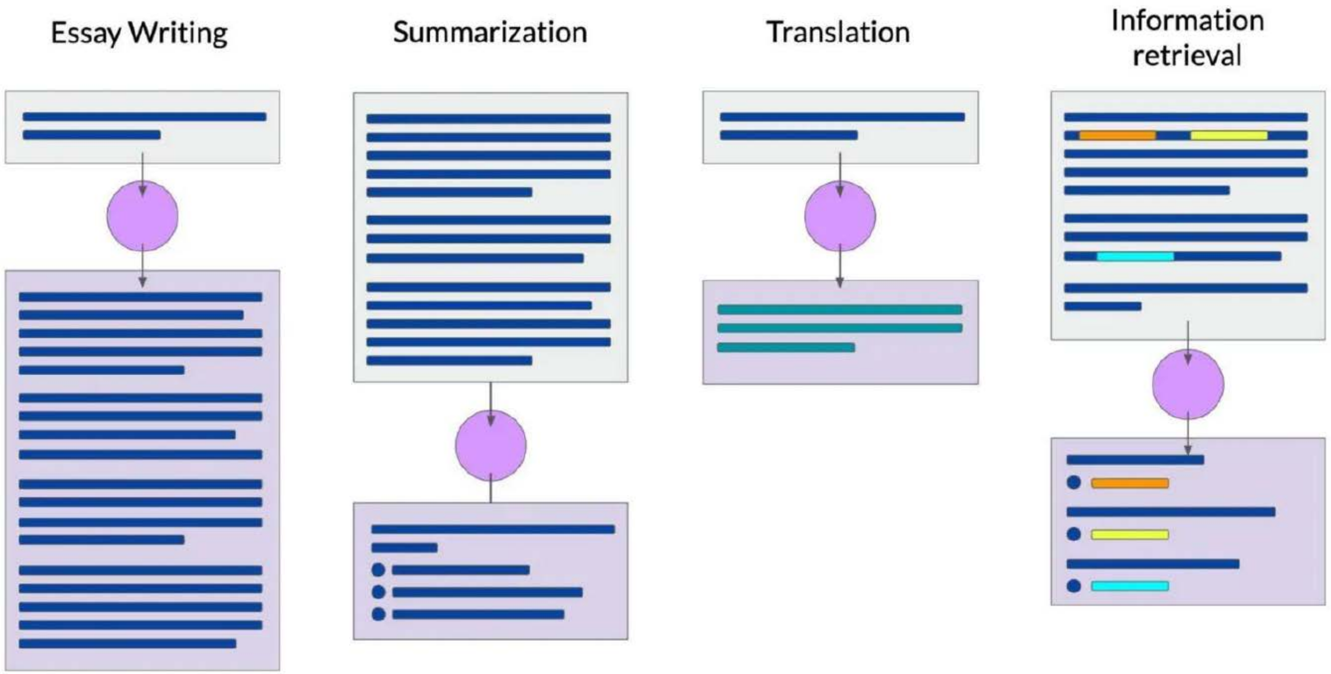


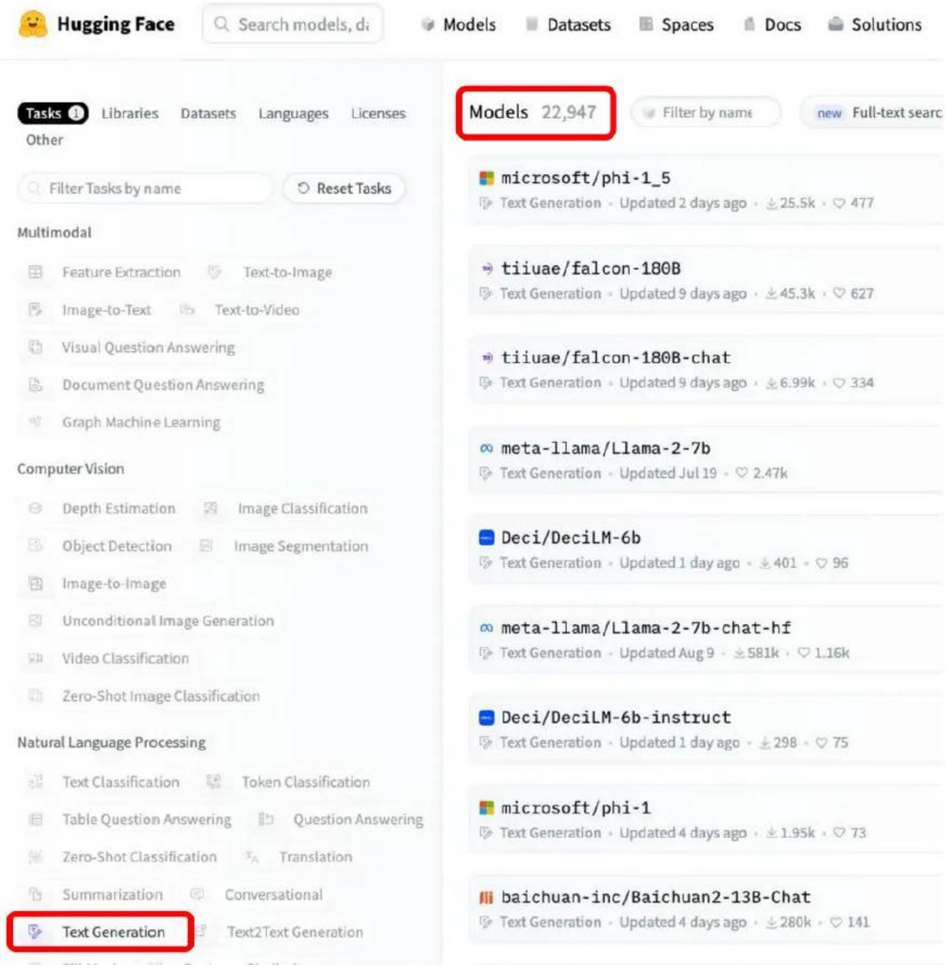
Reinforcement Learning from Human Feedback (RLHF)



TAILORING LLMs TO SPECIFIC PROBLEMS

Select a problem to solve





Select a LLM & use as is vs. *train from scratch*

- ***Adapt/ fine-tune***
- ***Retrieval-augmented generation***

USING LLMs AS IS: PROMPT ENGINEERING

Query crafting: *Improving the output of LLMs with actions like* ***phrasing queries, specifying style, providing context, assigning roles***

Chain of thought: *Guiding the LLM through intermediate steps towards a final answer*

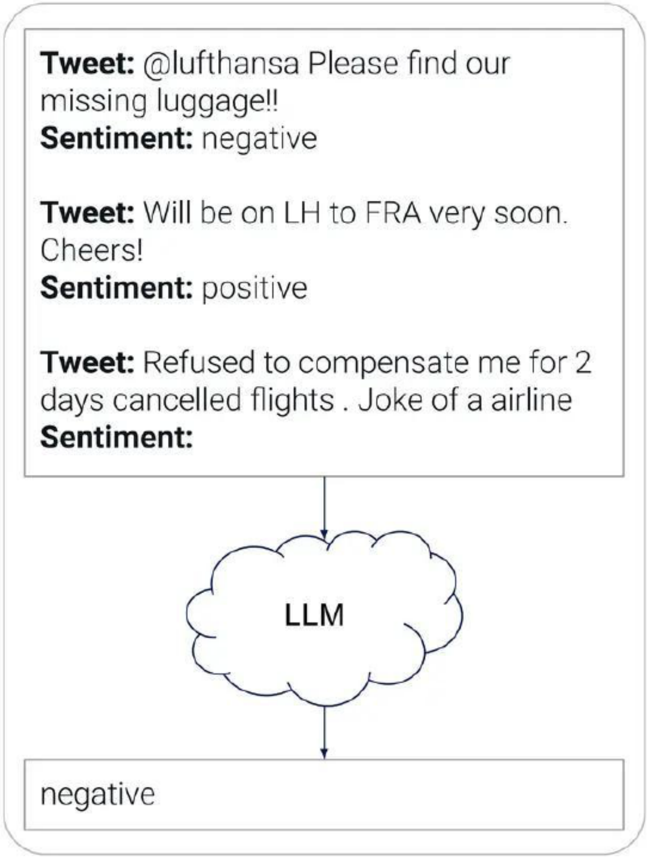
In-context learning (ICL): *Provide the LLM with one (or few) examples before asking it to solve a specific task*

0. ***Zero-shot prompting***

   No examples given

   Works for really large LMs

1. ***One-shot prompting:*** *One example*
2. ***Few-shot prompting:*** *Few examples*



Example of an input and output for two-shot prompting

FINE TUNING LLMs

Few-shot learning/ prompting *does not change the LLM*

It may not be sufficient for small models

In contrast, fine-tuning adapts an old model to obtain a new model

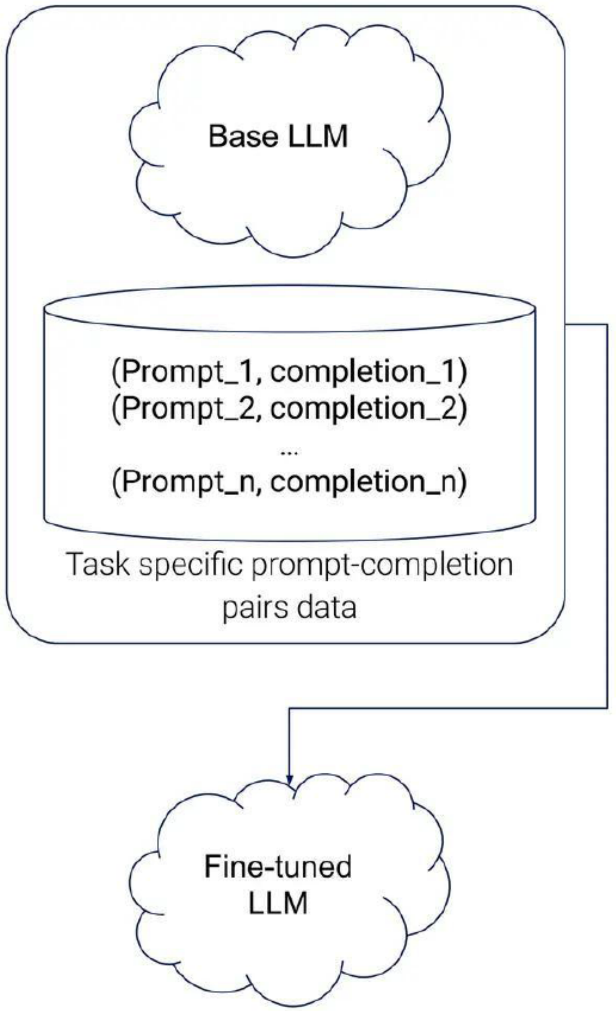
Fine-tuning *is a supervised learning process*

Which leads to a new model

It requires ***labeled data****, i.e., task specific (prompt,completion) pairs*

Instruction fine-tuning: each (prompt,completion) pair contains specific instruction (e.g., summarize this, translate that, classify)

We can fine-tune with data for one task only or for multiple-tasks



RETRIEVAL AUGMENTED GENERATION

Query Embedding

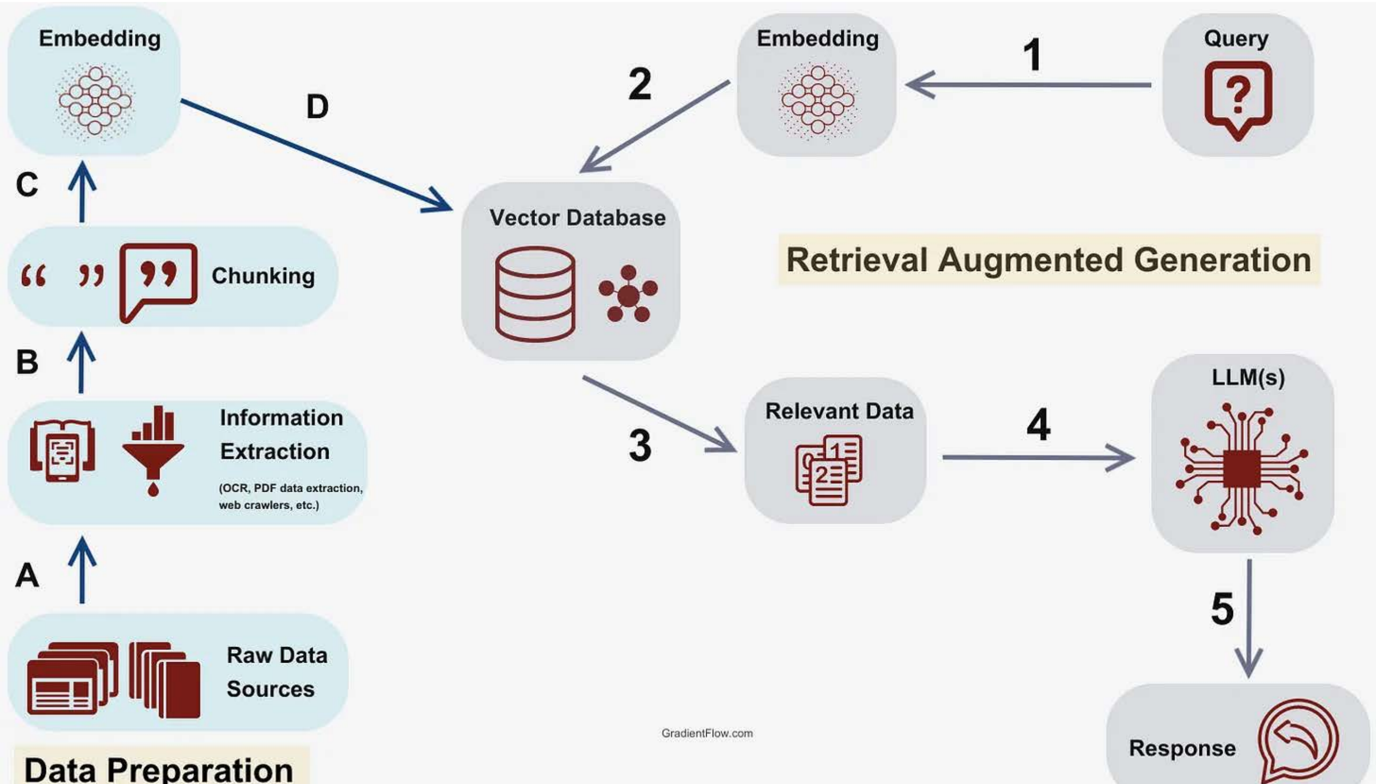
Transforms lawyer's question into a semantic vector that captures captures meaning and intent for precise searching.

Retriever

Searches firm's document chunks, case law, contracts, or regulations to find relevant information.

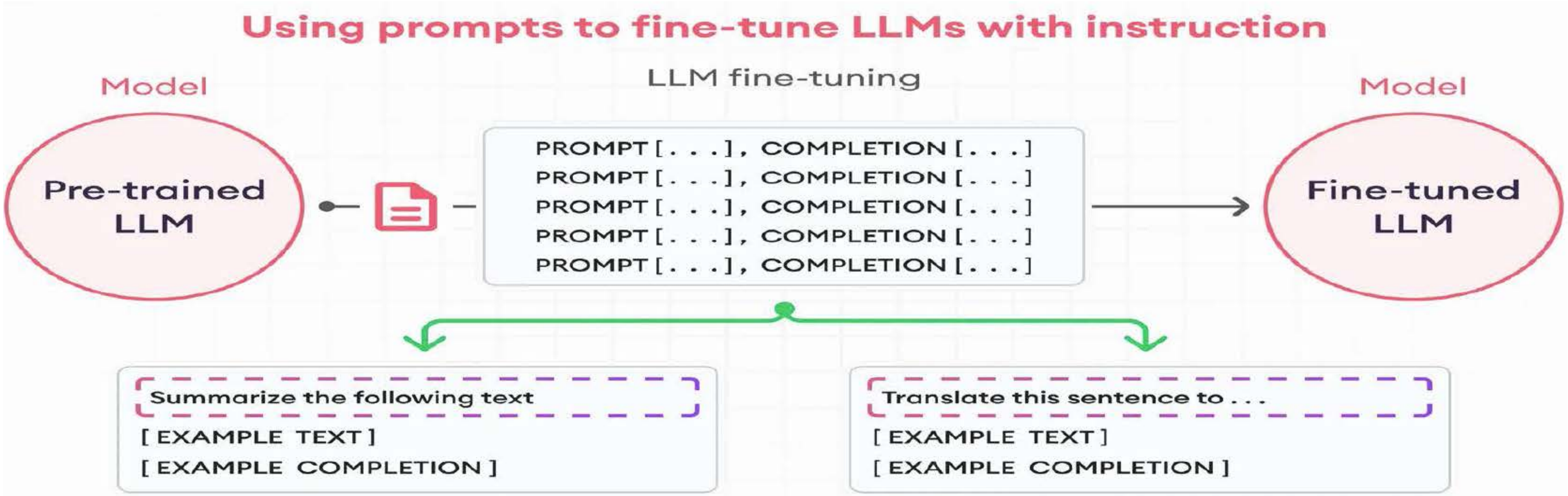
Generator

Synthesizes retrieved info with AI's language skills to produce accurate, context-aware responses.



FINE-TUNING FOUNDATION MODELS

Foundation models (FMs) are large models (e.g., large language models=LLMs) generated by applying ML (deep learning) to a broad collection of data at scale that can be adapted/fine-tuned for use in a wide range of downstream tasks



FOUNDATION MODELS FOR SCIENCE (Example from nutrition science)

LLMs can be adapted (with own data) to specific scientific domains:
As an example, we have fine-tuned the Llama 3 LLM model to nutrition science

The LLM was fine-tuned on several nutrition datasets to be able to:

- Extract food related named entities (NE): NER (recognition) / NEL (linking)
- Classify food entities according to several food taxonomies (e.g., FoodON)
- Retrieve nutritional values for ingredients and recipes

EXAMPLE:

Input: Compute the nutrient values per 100 grams in a recipe with the following ingredients: 250 g cream, whipped, cream topping, pressurized, 250 g yogurt, greek, plain, nonfat, 50 g sugars, powdered.

Output: Nutrient values per 100 g listed: energy - 179.00, fat - 10.28, protein - 6.09, salt - 0.05, saturates - 6.34, sugars - 14.00

FOUNDATION MODELS FOR SCIENCE (Example from nutrition science)

The tasks:

- ***Traffic Light Classification of Recipes:*** *healthy (green), moderate (orange), limit consumption (red)*
    - *Two datasets: Ingredients, Title+Ingredients*
    - *Four dimensions: Salt, Sugar, Fat, Saturates*
- ***Assessing Recipe Nutrient Values***
    - *Two datasets: Ingredients, Title+Ingredients*
- ***Making food data interoperable***
    - *Named entity linking (NEL)*
    - *Named entity recognition (NER) + NEL*
    - *Three datasets: CafeteriaFCD (recipes), CafeteriaSA (scientific abstracts), Artificial*
    - *Three ontologies: FoodOn, SnomedCT, Hansard*

EXAMPLE PROMPTS FOR THE THREE TASKS

Zero-shot prompt:
[INST]Review the nutrient values per 100 grams in a recipe using these ingredients: 180 g wheat flours, bread, unenriched, 180 g wheat flour, white, cake, enriched, 200 ml sugars, granulated, 200 g butter, without salt, 200 ml water, bottled, generic, 1 pinch salt, table.[/INST]
Nutritional values in each 100 g: energy - 364.03, fat - 17.96, protein - 4.09, salt - 0.05, saturates - 10.95, sugars - 18.33

One-shot prompt:
[INST]*The following are examples of questions (with answers) about nutrition.* Question: Determine the nutritional profile per 100 grams in a recipe that uses these ingredients: 1 cup cheese, gouda, 4 tablespoon butter, without salt, 3/4 cup wheat flour, white, all-purpose, unenriched, 1/2 teaspoon salt, table, 1/2 teaspoon spices, pepper, red or cayenne, 1 tablespoon cream, fluid, heavy whipping Answer: Nutrient values highlighted for 100 grams: energy - 426.12, fat - 29.33, protein - 14.52, salt - 1.92, saturates - 18.33, sugars - 1.18 *Respond to the following question in the same manner as seen in the examples above.* Question: Review the nutrient values per 100 grams in a recipe using these ingredients: 180 g wheat flours, bread, unenriched, 180 g wheat flour, white, cake, enriched, 200 ml sugars, granulated, 200 g butter, without salt, 200 ml water, bottled, generic, 1 pinch salt, table Answer:[/INST]
Nutritional values in each 100 g: energy - 364.03, fat - 17.96, protein - 4.09, salt - 0.05, saturates - 10.95, sugars - 18.33

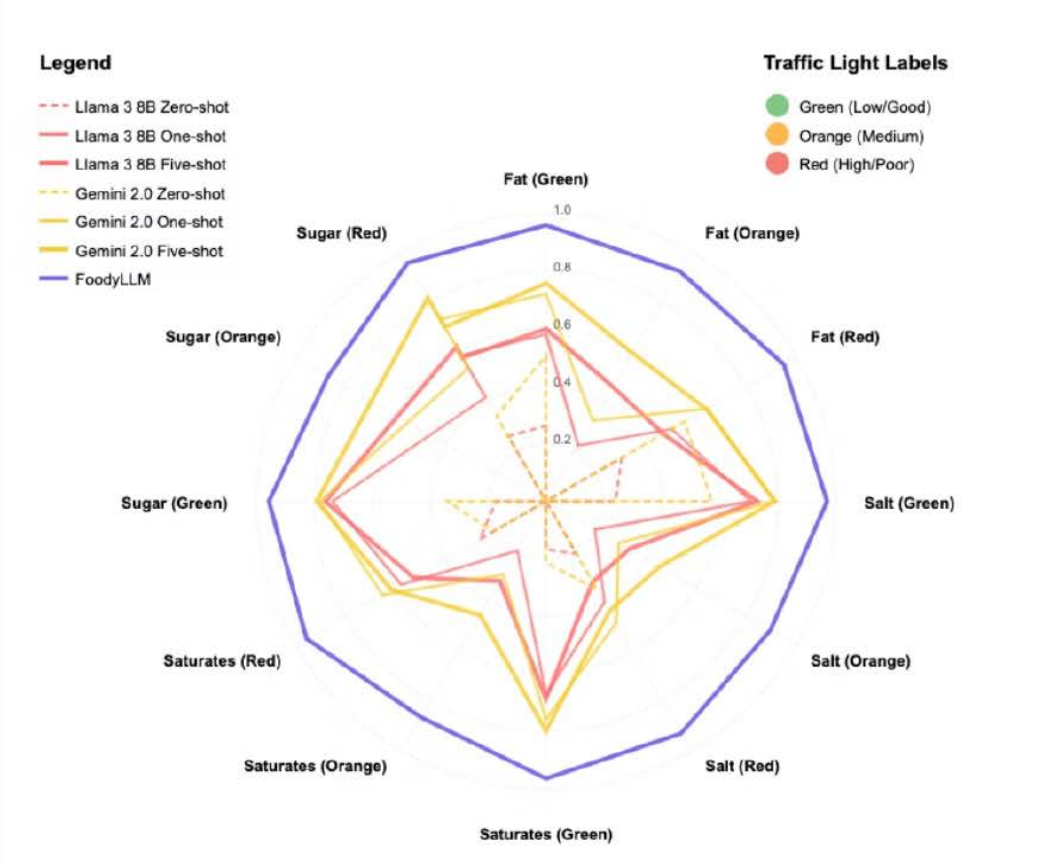
Five-shot prompt:
[INST]*The following are examples of questions (with answers) about nutrition.* Question: Gauge the nutrient values per 100 grams in a recipe prepared with the following ingredients: 2 cup cream, fluid, heavy whipping, 1 tablespoon spices, cardamom Answer: Per 100 g, the nutrient values are: energy - 339.02, fat - 35.36, protein - 3.04, salt - 0.07, saturates - 22.49, sugars - 2.85 Question: Establish the nutrient profile per 100 g in a recipe containing these ingredients: 1 tablespoon shallots, raw, 2 teaspoon spices, garlic powder, 12 cup peanut butter, smooth style, without salt, 3 tablespoon oil, sesame, salad or cooking, 2 tablespoon soy sauce made from soy (tamari), 1 teaspoon spices, ginger, ground, 1 teaspoon roland, seasoned rice wine vinegar, upc: 041224705142, 1/4-1/2 teaspoon spices, pepper, red or cayenne, 13 cup soup, chicken broth or bouillon, dry Answer: Nutrient profile for every 100 g: energy - 494.83, fat - 40.58, protein - 20.22, salt - 17.01, saturates - 8.28, sugars - 12.29 Question: Verify the nutrient values per 100 g in a recipe prepared with these ingredients: 16 ounce milk, fluid, 1% fat, without added vitamin a and vitamin d, 8 ounce beverages, almond milk, unsweetened, shelf stable, 13 cup sugars, granulated, 14 cup cornstarch, 12 teaspoon vanilla extract, 14 teaspoon shortening confectionery, coconut (hydrogenated) and or palm kernel (hydrogenated) Answer: Nutrient facts per 100 grams: energy - 340.40, fat - 1.30, protein - 0.41, salt - 0.03, saturates - 1.11, sugars - 50.82 Question: Identify the nutritional composition per 100 grams in a recipe with these ingredients: 500 g ground turkey, raw, 1 cup onions, raw, 12 cup bread crumbs, dry, grated, plain, 12 cup carrots, raw, 12 cup sauce, barbecue, 2 teaspoon sauce, worcestershire, 1 teaspoon spices, garlic powder, 34 teaspoon spices, pepper, black Answer: Nutrient profile for each 100 g: energy - 180.22, fat - 1.96, protein - 4.72, salt - 1.63, saturates - 0.42, sugars - 18.20 Question: Find the nutritional breakdown per 100 g in a recipe that uses the following ingredients: 12 pound pretzels, soft, unsalted, 21 ounce corn, sweet, white, raw, 12 ounce cookies, graham crackers, plain or honey, lowfat, 32 ounce nuts, walnuts, english, 7 ounce cookies, graham crackers, plain or honey, lowfat, 12 ounce cookies, graham crackers, plain or honey, lowfat, 1 pound butter, without salt, 12 drop sauce, ready-to-serve, pepper, tabasco, 8 tablespoon sugars, brown, 2 teaspoon spices, chili powder, 2 teaspoon sauce, worcestershire, 2-3 tablespoon spices, garlic powder Answer: The nutrient breakdown per 100 g is: energy - 383.98, fat - 14.10, protein - 7.92, salt - 0.59, saturates - 3.85, sugars - 3.92 *Respond to the following question in the same manner as seen in the examples above.* Question: Review the nutrient values per 100 grams in a recipe using these ingredients: 180 g wheat flours, bread, unenriched, 180 g wheat flour, white, cake, enriched, 200 ml sugars, granulated, 200 g butter, without salt, 200 ml water, bottled, generic, 1 pinch salt, table Answer:[/INST]
Nutritional values in each 100 g: energy - 364.03, fat - 17.96, protein - 4.09, salt - 0.05, saturates - 10.95, sugars - 18.33

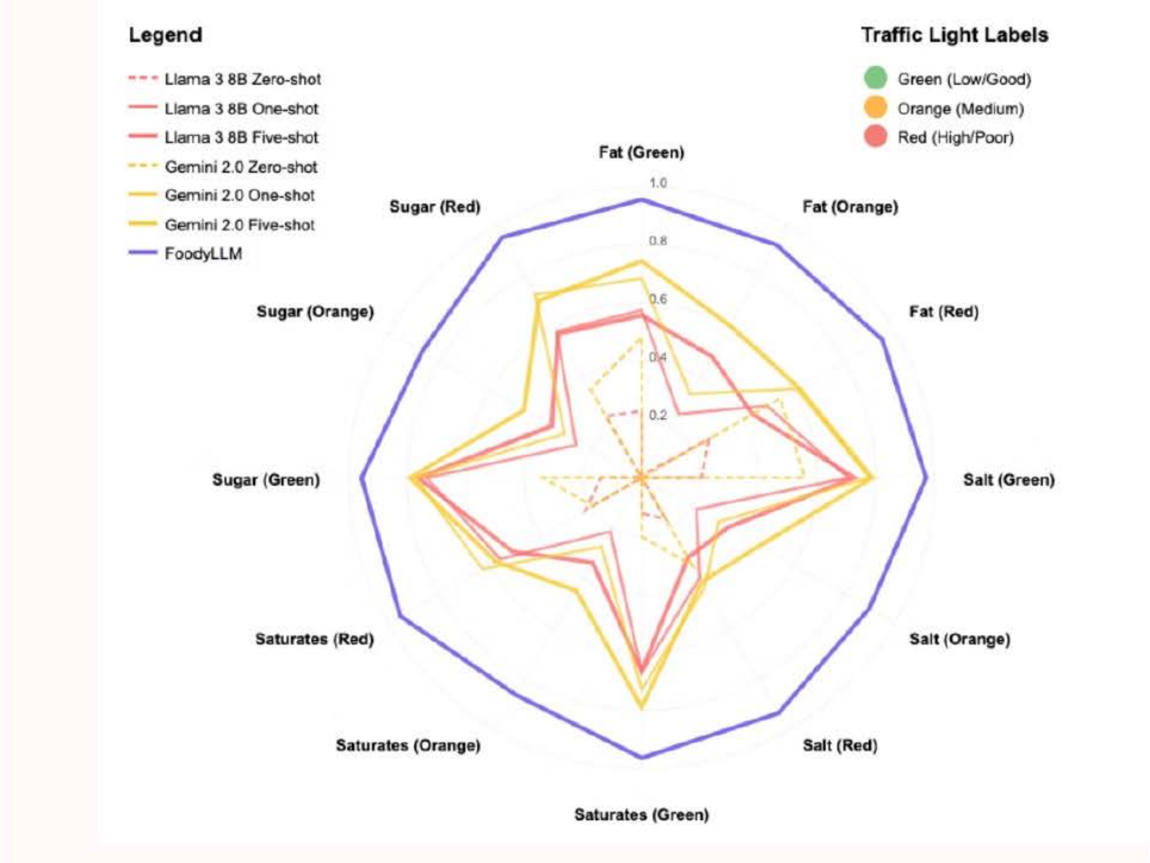
[INST] Verify the FSA traffic light labels per 100 grams in a recipe that has these ingredients: 1 cup water, bottled, generic, 1 tablespoon oil, olive, salad or cooking, 34 teaspoon salt, table, 3 cup wheat flours, bread, unenriched, 2 teaspoon sugars, granulated, 1 teaspoon wheat flours, bread, unenriched, 2 teaspoon spices, parsley, dried, 1 teaspoon semolina, unenriched [/INST]
Food Standards Agency traffic lights per 100 g presented: fat - green, salt - red, saturates - green, sugars - green

[INST] Evaluate the FSA traffic light labels per 100 grams in *Easy Pizza Dough for Bread Machine* using these ingredients: 1 cup water, bottled, generic, 1 tablespoon oil, olive, salad or cooking, 34 teaspoon salt, table, 3 cup wheat flours, bread, unenriched, 2 teaspoon sugars, granulated, 1 teaspoon wheat flours, bread, unenriched, 2 teaspoon spices, parsley, dried, 1 teaspoon semolina, unenriched [/INST]
The fsa traffic lights per 100 g listed are: fat - green, salt - red, saturates - green, sugars - green

[INST] Find all food entities present in the text: Mix the cream cheese, beef, olives, onion, and Worcestershire sauce together in a bowl until evenly blended. Keeping the mixture in the bowl, scrape it into a semi-ball shape. Cover and refrigerate until firm, at least 2 hours. Place a large sheet of waxed paper on a flat surface. Sprinkle with walnuts. Roll the cheese ball in the walnuts until completely covered. Transfer the cheese ball to a serving plate, or rewrap with waxed paper and refrigerate until needed. [/INST]
Certainly, the entities connected with food are outlined as follows: cream cheese, beef, olives, onion, Worcestershire sauce, walnuts, cheese ball.
[INST] Is it within your capabilities to link the extracted food entities to a Hansard taxonomy? [/INST]
Certainly, the entities are associated properly: cream cheese - AG.01.e [Dairy produce]; AG.01.e.02 [Cheese]; AG.01.n [Dishes and prepared food]; AG.01.n.18 [Preserve], beef - AG.01.d.03 [Beef], olives - AG.01.h.01.e [Fruit containing stone], onion - AG.01.h.02.e [Onion/leek/garlic], Worcestershire sauce - AG.01.h [Fruit and vegetables]; AG.01.l.04 [Sauce/dressing], walnuts - AG.01.h.01.f [Nut], cheese ball - AG.01.e.02 [Cheese]; AG.01.n.18 [Preserve].
[INST] Might I trouble you to connect the extracted food entities to a FoodOn ontology, if possible? [/INST]
Definitely, the entities are linked suitably: cream cheese - FOODON-03301889; FOODON-00001013, onion - FOODON-03301704; NCBITaxon-4679, worcestershire sauce - FOODON-03305003; FOODON-03311146, walnuts - NCBITaxon-16718, cheese - FOODON-00001013.
[INST] Might it be possible for you to link them to a SNOMEDCT ontology, if needed? [/INST]
Absolutely, the entities are related as such: cream cheese - SNOMEDCT-226849005; SNOMEDCT-255621006; SNOMEDCT-102264005, beef - SNOMEDCT-226916002, olives - SNOMEDCT-227436000, onion - SNOMEDCT-735047000, worcestershire sauce - SNOMEDCT-443701000124100; SNOMEDCT-227519005, cheese - SNOMEDCT-102264005.

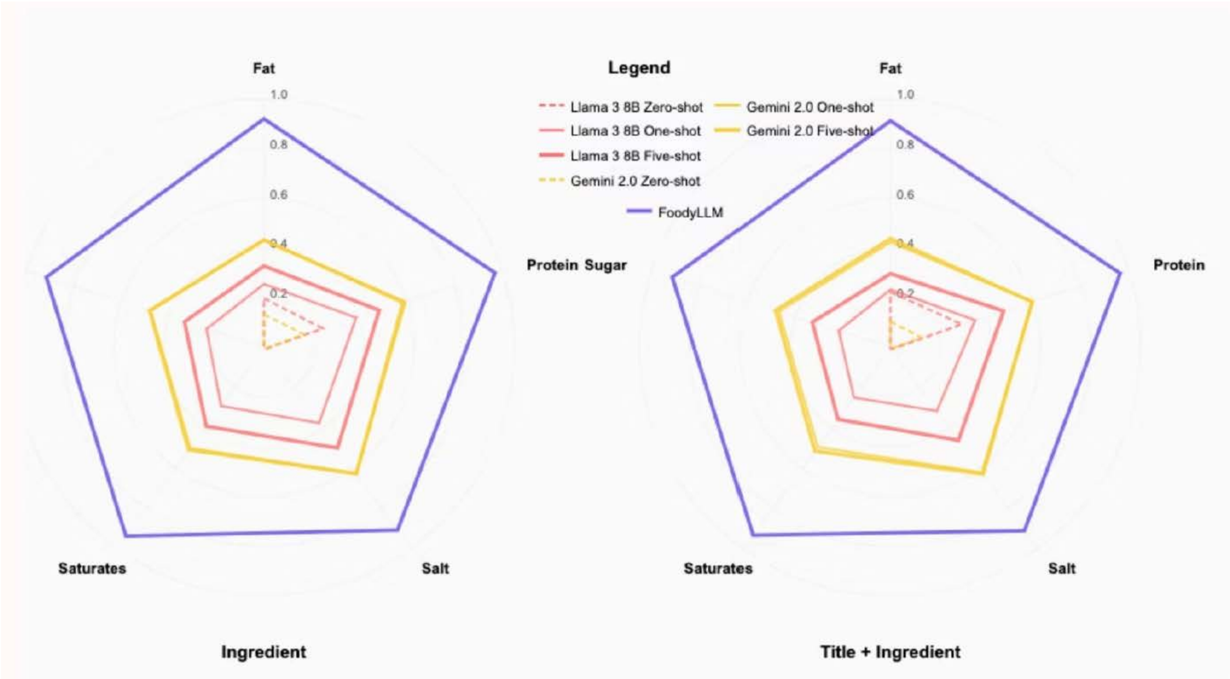
FoodyLLM: Results for Traffic Light Classification



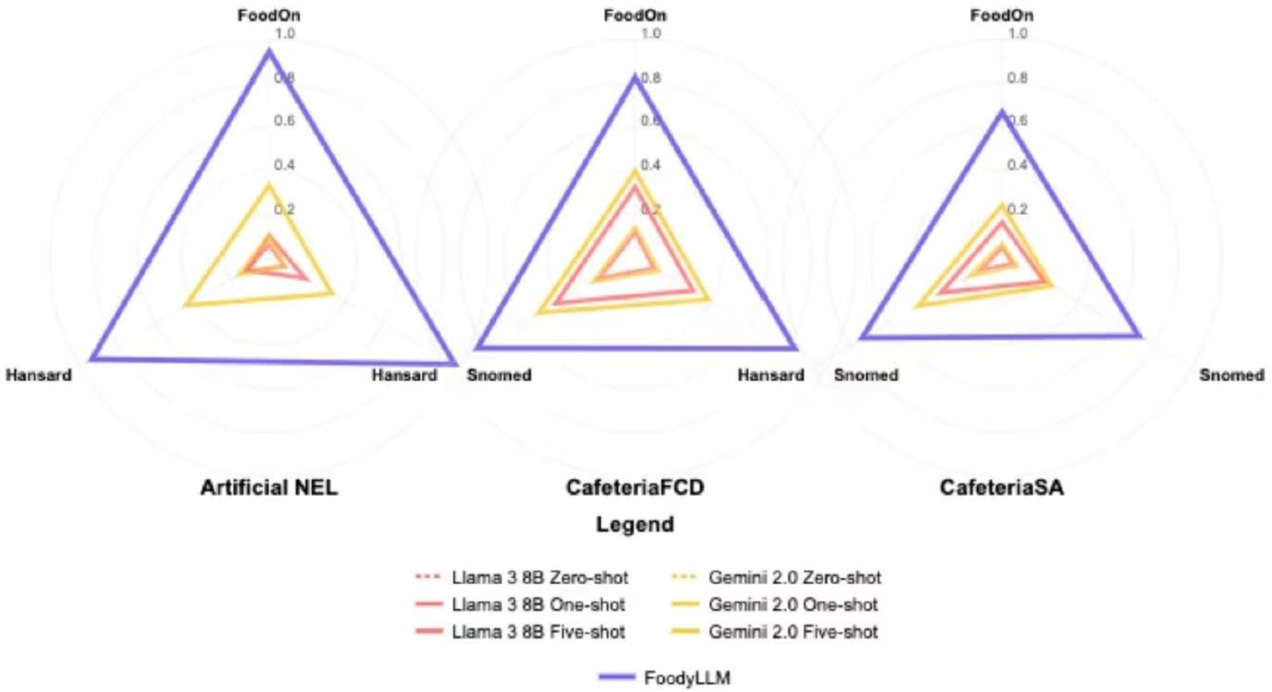


Ingredient dataset; Title+Ingredient dataset

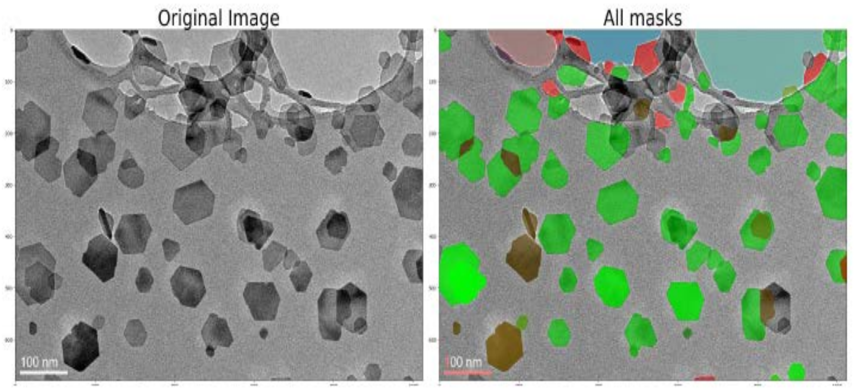
FoodyLLM:
Results for
Assessing Recipe Nutrient Values



Named Entity Linking



USING VISUAL FOUNDATION MODELS IN MATERIALS SCIENCE



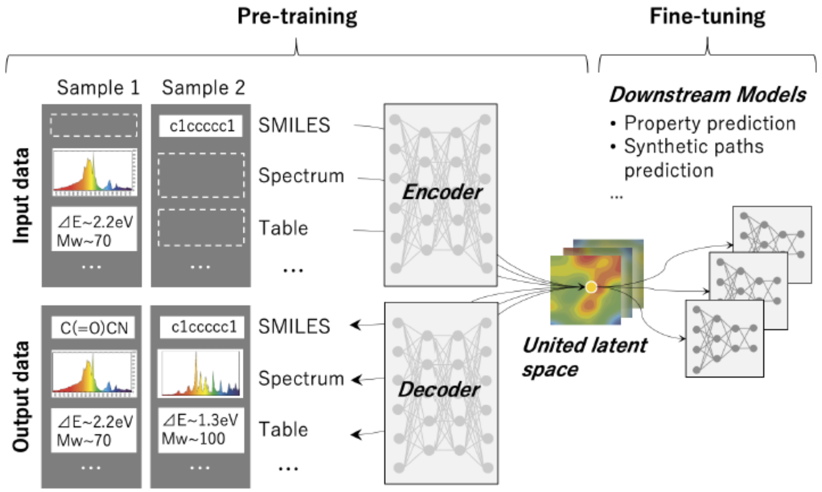
Segment Anything Model: *Used to analyze Electron Microscopy Images*

Designing magnetic materials: *Transmission Electron Microscopy images of Barium hexaferrite nanoplatelets*

***Particle diameter distribution predicts** magnetic properties*

Coming up next

Multi-modal FMs for materials, e.g., visual-language models



MULTI-MODAL FOUNDATION MODELS

Are built and operate on different modalities (text, audio, images, video, …)

- *Align the representations of different modalities*
- *Most typically bimodal (e.g. vision-language models)*

Multi-modal FM learned from remotely sensed data

Include DOFA and AlphaEarth-Foundations