Transformers and foundation models

Arto Klami

April 9, 2025

Where are we?

- Last time we went through the transformer architecture
- BERT, GPT and Vision Transformer as example models with rather simple architecture but trained on massive amounts of data
- Earlier we talked about transfer learning: How we should leverage on already trained models to solve new tasks
- Today: A bit more about how transformers are trained and used
- Concept of foundation models
- NOTE: Most of the slides *intentionally* not updated from last year in terms of models, to show pace of progress. For instance, `DeepSeek chatbot` is from January 2025 and hence excluded

Self-attention layer

- In matrix form we have (omitting biases for clarity)

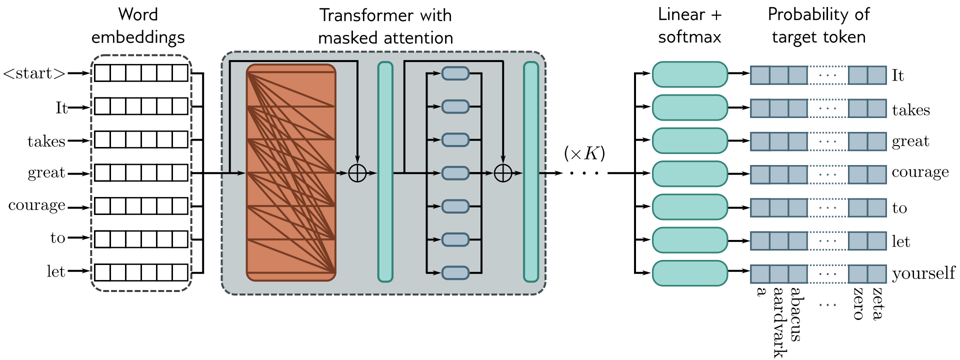
$$\mathbf{Q} = \mathbf{W}_q \mathbf{X}$$
$$\mathbf{K} = \mathbf{W}_k \mathbf{X}$$
$$\mathbf{V} = \mathbf{W}_v \mathbf{X}$$

- The whole computation is given by

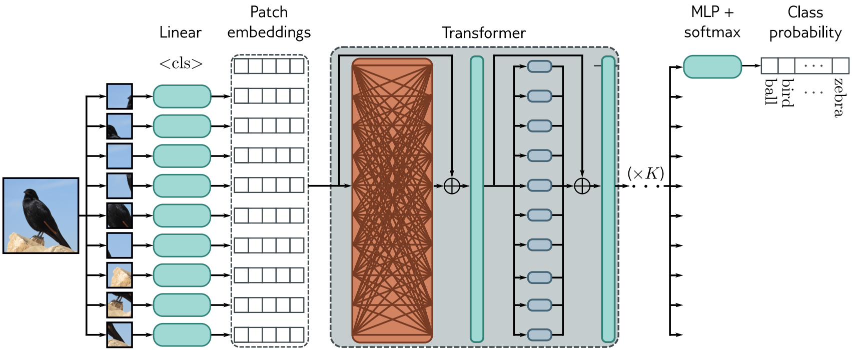
$$\mathbf{Y} = \mathbf{V} \cdot \mathrm{Softmax}\left(\frac{\mathbf{Q}^T \mathbf{K}}{\sqrt{D}}\right)$$

- A transformer block combines this with residual connection, layer normalization and input-specific MLPs
- A full model is (mostly) just a stack of these blocks

Examples: GPT3



Examples: Vision Transformer



GPT vs ViT

- Almost the same architecture
- Trained on data of similar scale:
  - GPT-3 used 300 billion tokens, but if a single example has roughly 1000 tokens then this is 300 million text snippets
  - ViT used 300 million labeled examples
- The supervision signal is completely different: ViT needed dedicated labeling effort, but GPT was trained to predict the next word in the sequence
- Both are very good models, so it is possible to train models this large in both ways

Self-supervised learning

- The training protocol for GPT is an example of *self-supervised learning* (more about this during the second half of the lecture)
- Basic idea:
  - Construct an auxiliary learning task that only requires the 'input' data
  - Use standard supervised learning tools to learn the model
  - Use the model for some new task
- Key advantage is that we do not need manual labeling since the true outputs are already known
- The auxiliary task of predicting the next word is a bit boring special case of self-supervised learning: People have studied it as an actual task since 90s (or perhaps 70s) and no new theory was needed to make it possible (trivial change of the self-attention layer was enough)

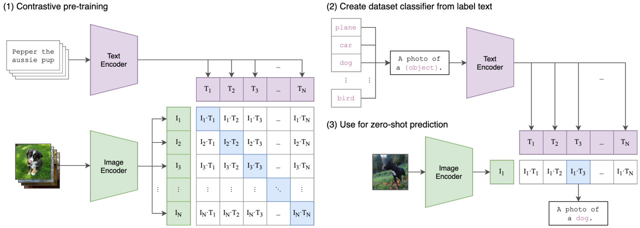
Self-supervised learning

- Even though we train the model to perform well in the training tasks, the goal is to use the model (also) for other tasks
- In some cases the auxiliary task can be completely artificial, in the sense that the model would never used for that task
- BERT is quite explicit on this, by splitting the training in two phases:
  - Pre-training: Learn to predict words that were masked out, from somewhere in the middle of a sentence. This is self-supervised learning.
  - Fine-tuning: Train the model further for a specific supervised task, like document classification or named entity recognition
- Note how we still need annotated labels for the second phase

Self-supervised learning: Re-visiting zero-shot learning

CLIP (Contrastive Language-Image Pre-Training)

- Train model on pairs of images and captions, to get representation space where images are close to their captions
- Zero-shot classifier for mythical beasts (hopefully not seen in training data):
    1. Fetch textual description of each beast from Wikipedia: *"A cockatrice is a mythical beast, essentially a two-legged dragon, wyvern, or serpent-like creature with a rooster's head."*
    2. Check which textual description is closest to a test image (nearest neighbor classifier)



Source: https://github.com/openai/CLIP

Foundation models

- *Foundation model* refers to a large model that can be used for solving diverse set of practical tasks
- Works as basis (a foundation) for new models
- Recent attempts of formal definition (largely for legislative purposes):
  - US: *"An AI model that is trained on broad data; generally uses self-supervision; contains at least tens of billions of parameters; is applicable across a wide range of contexts"*
  - EU: *"AI model that is trained on broad data at a scale, is designed for generality of output, and can be adapted to a wide range of distinctive tasks"*
  - UK: *"a type of AI technology that are trained on vast amounts of data that can be adapted to a wide range of tasks and operations"*

Foundation models

- Many of the current foundation models are transformer-based, especially in the language domain
- But not all: Diffusion models, state-space models, ...
- The concepts are on different levels: Transformer is an architecture family, whereas foundation model refers to a large model having wide capability
- We previously speculated on learning universal representations by multi-task learning, by pooling in all possible output modalities as tasks. Many of the current foundation models do exactly this, but in self-supervised manner, combining audio, video, text, images etc.

Foundation models

- Two streams of research and development:
  - Developing new/better foundation models
  - Developing services on top of them
- Extreme ongoing competition in developing both proprietary and open foundation models (Llama, Mistral, GPT, Poro, ...) especially for 'human-digestable data' (language, images, video, ...)
- Parallel activities in more specialized domains (drug molecules, time series, ...) that capture less public attention
- As hinted last time, this is an extremely costly arms race: GPT4 training cost was more than 100 million USD, many others report numbers around 1-10 million (not including salary costs, which probably trump all other costs)
- Smaller models are more feasible, but still costly: LLama-2-7b model still requires hardware that costs 30M, has training cost of tens of thousands and fine-tuning cost of thousands for each new task

Foundation models

- GPT3 can already predict the continuation of a text snippet, but is not in itself a (good) conversational agent
- `ChatGPT` uses GPT as the language model, but includes a lot of other components
  - Supervised fine-tuning: Human annotator writes desired output, used as training label
  - Reinforcement learning from human feedback: Annotator ranks alternative outputs, model learns to output answers that would get high reward by ranking high
  - Supervised training for safeguarding from harmful content, by manual annotation of undesired outputs etc.
- Similar story for the competing products (Gemini, Claude, ...)
- Many of the new capabilities in specific services are external for the 'neural network'. For example, *retrieval augmented generation* (RAG) refers to running explicit search to increase reliability of the generated content

Working with foundation models

- How should you work with these models?
- No good established practices yet, but we/you need to collectively figure it out:
  - What to build on? Common interfaces?
  - How to handle the (training and evaluation) cost?
  - How to use the models in research?
  - Open or closed models?
  - How to cope with disappearing models?
- Beyond this course; our focus is on the neural network component itself, but people will still be asking you for opinions