Hvordan fungerer store språkmodeller?

Pierre Lison

Norsk Regnesentral (NR)

& Universitetet i Oslo

10. juni 2024

NORSK REGNESENTRAL
NORWEGIAN COMPUTING CENTER

UiO : University of Oslo

Store språkmodeller

= Maskinlæringsmodeller optimert til å gjette det neste ordet i en tekst

Dype nevrale nettverk
med mange milliarder
parametere

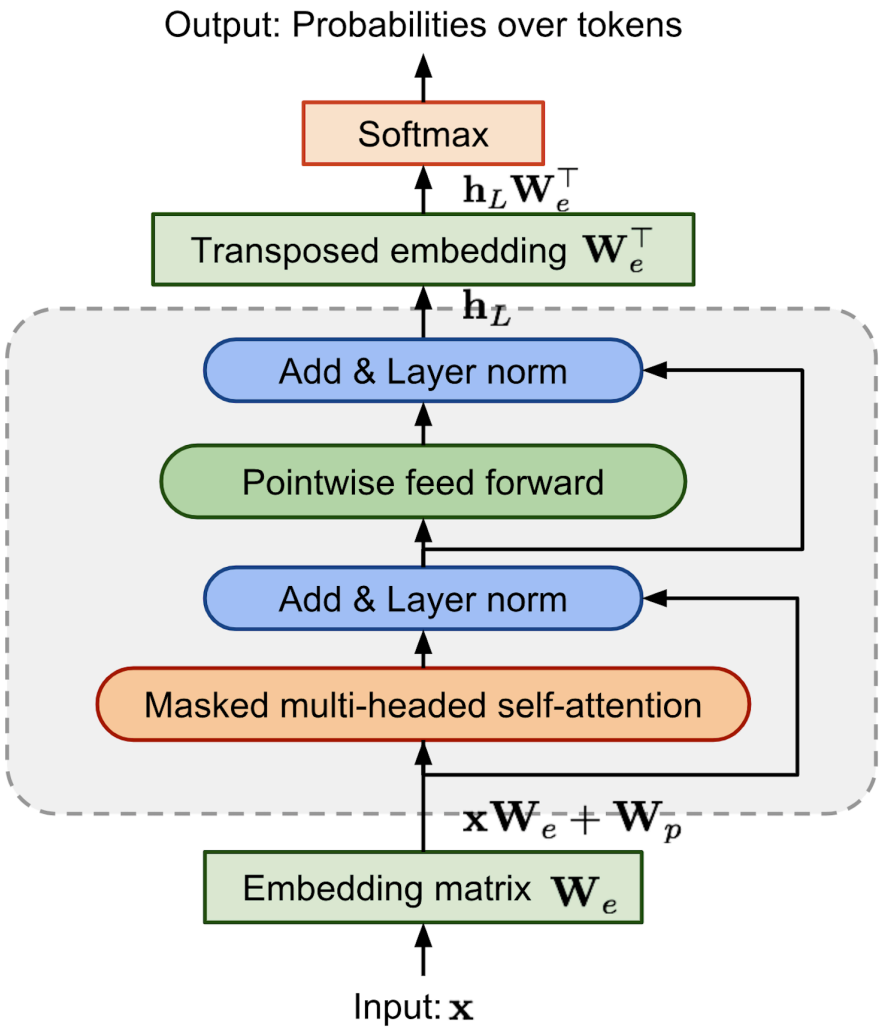
Trent på store tekstsamlinger
hentet fra nettkilder
(Wikipedia, bøker, nettfora, etc.)

Som et biprodukt av denne «gjetteleken» vil den nevrale modellen gradvis bygge opp **representasjoner** av hvordan språket fungerer, samt mer generelt *bakgrunnskunnskap*

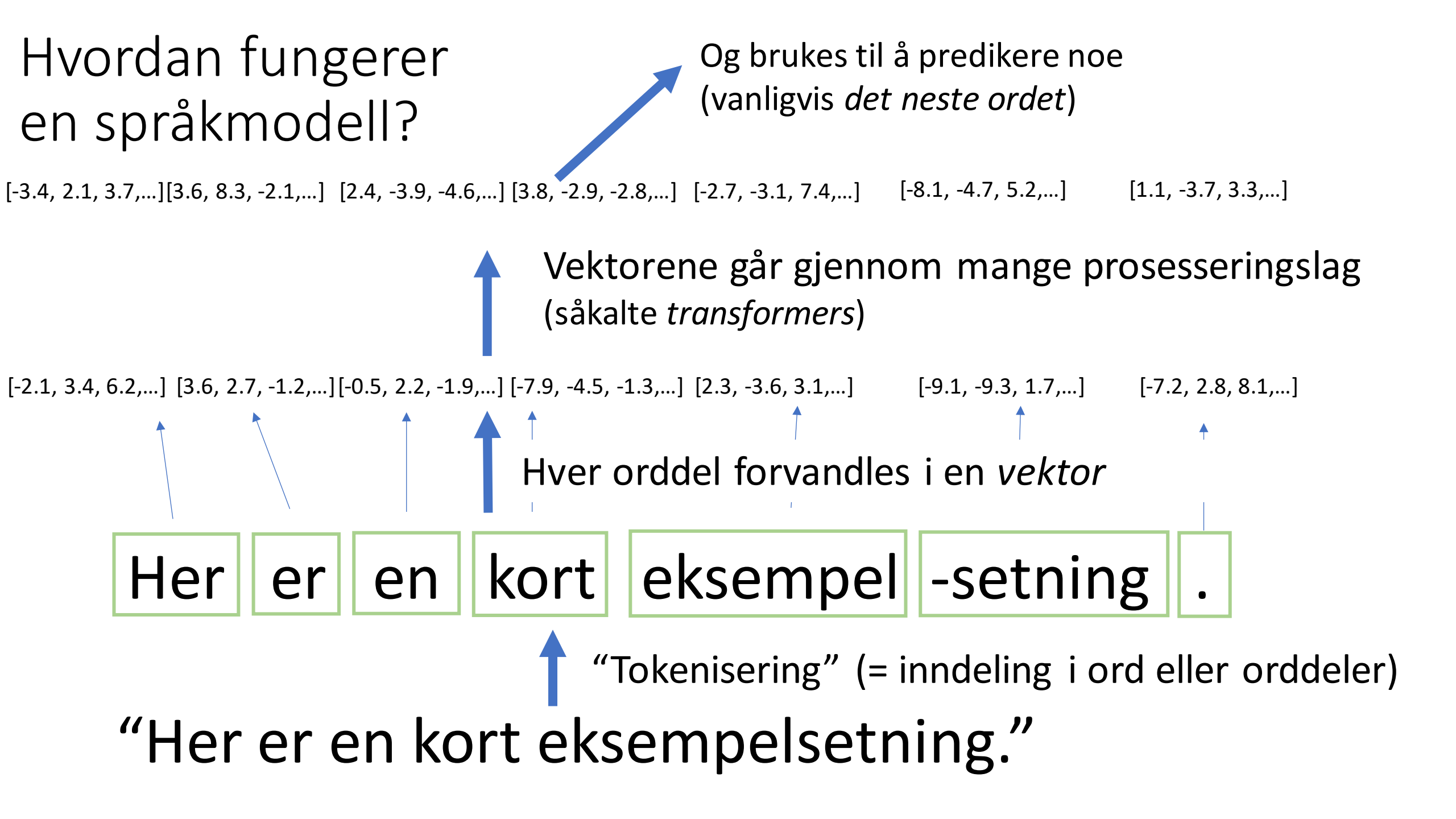
Store språkmodeller

Dagens språkmodeller er alle varianter av den samme grunnarkitektur: *transformers*

- Hver token (ord eller del av et ord) representeres med en lang rekke tall, såkalte *ordvektorer*
- Tallrekkene går gjennom mange *prosesseringslag*
- **Utdata**: sannsynlighet for neste token

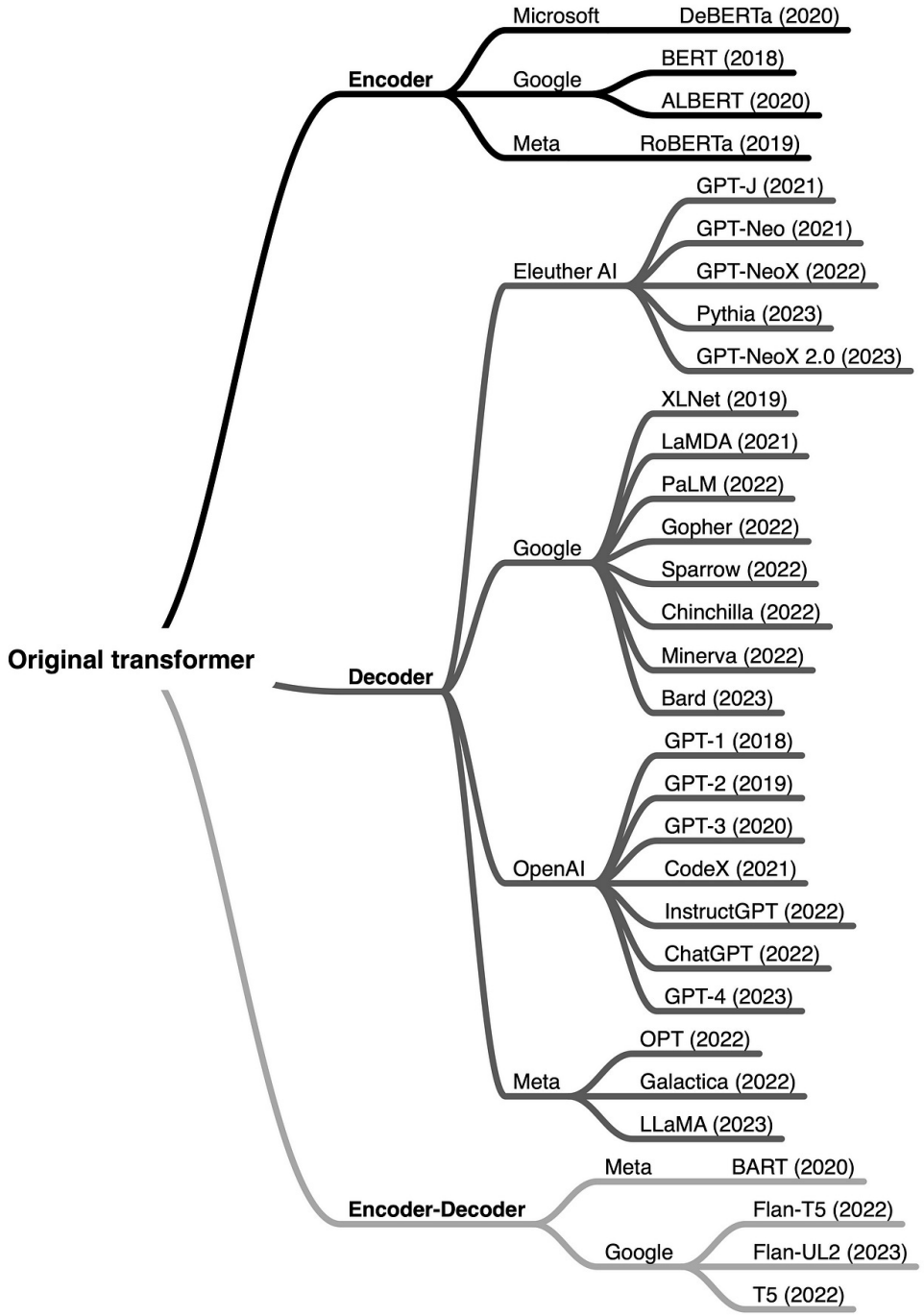


Hvordan fungerer en språkmodell?

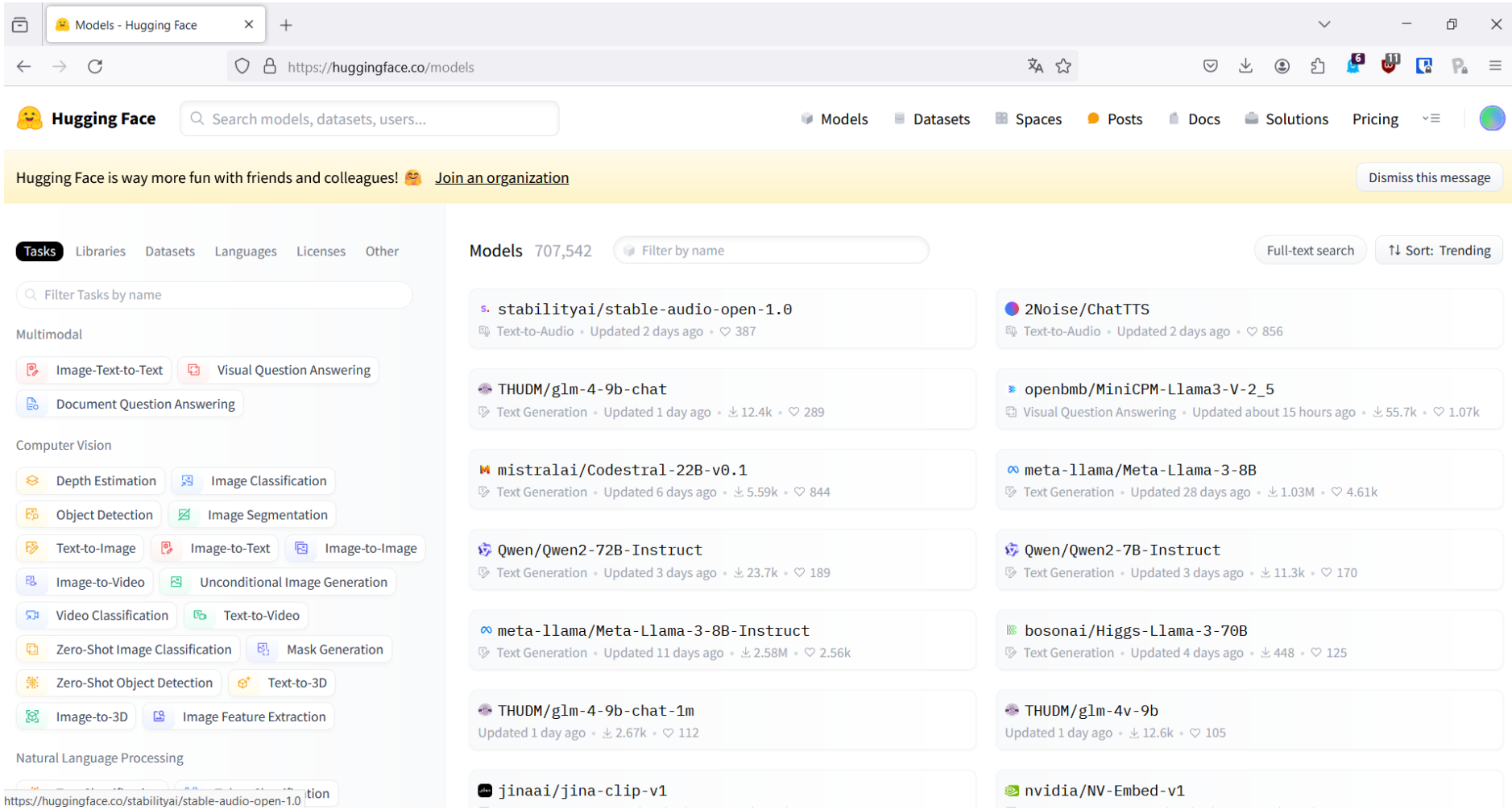


Varianter av transformers

- **Encoder**:
  - Mest egnet for tekstklassifisering og informasjongjenfinning
  - «konteksten» av et ord på begge sider
- **Decoder**:
  - Mest egnet for tekstgenerering
  - «Konteksten» er kun de forrige ordene
  - Meste populære tilnærming!
- **Encoder-decoder**:
  - Ulike språkmodeller for input og output
  - Brukt i bl.a. maskinoversettelse



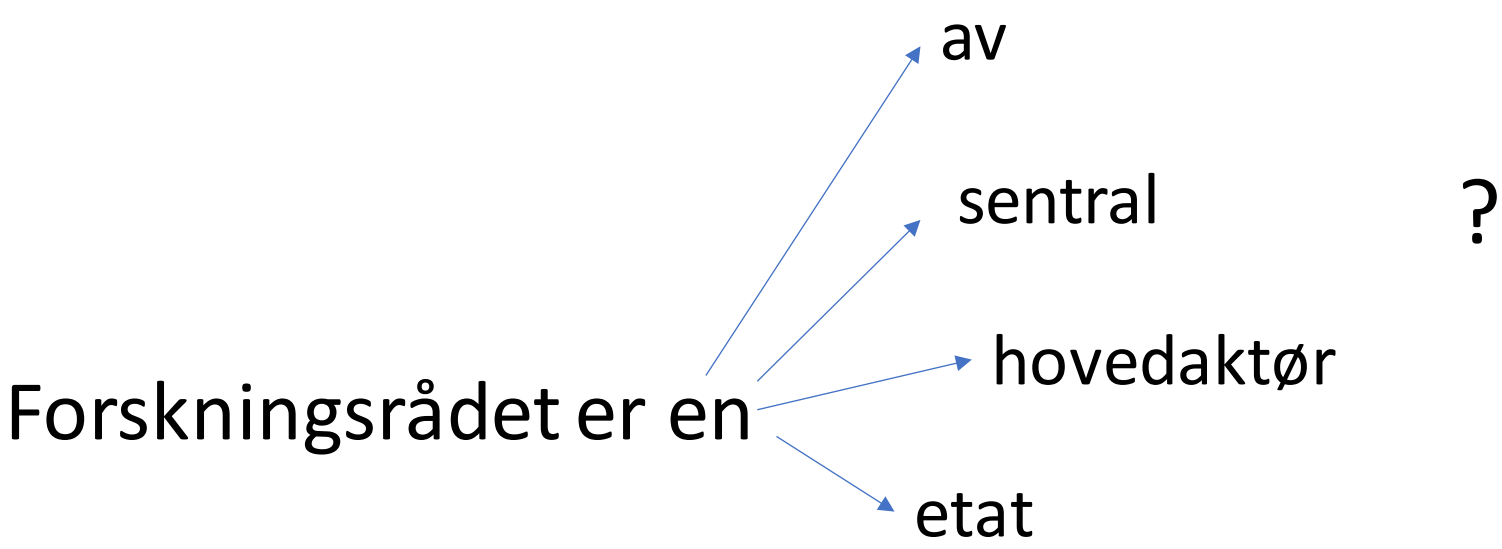
HuggingFace



Hva med norsk?

- Flere språkmodeller trent på norske data
  - NorBERT, NB-BERT, NorGPT, NorMistral – aktiv utvikling!
  - *Norwegian Collossal Corpus* som treningsgrunnlag: 49 GB med norskspråklige nyhetsartikler, bøker, offentlige rapporter, nettdata, osv.
- Også mulig å bruke *flerspråklige* modeller
  - Forholdsvis godt språk
  - … men en del kunnskapshull om norske forhold og kulturelle referanser

Trening



- Språkmodeller er optimert til å predikere *det neste ordet* i en tekst
- **Trening** = gradvis endring av parametere slik at modellprediksjonene kommer seg litt nærmere «fasiten»
- Starter med en «dum» modell som predikere vilkårlige ord og forbedrer den litt etter litt

Trening

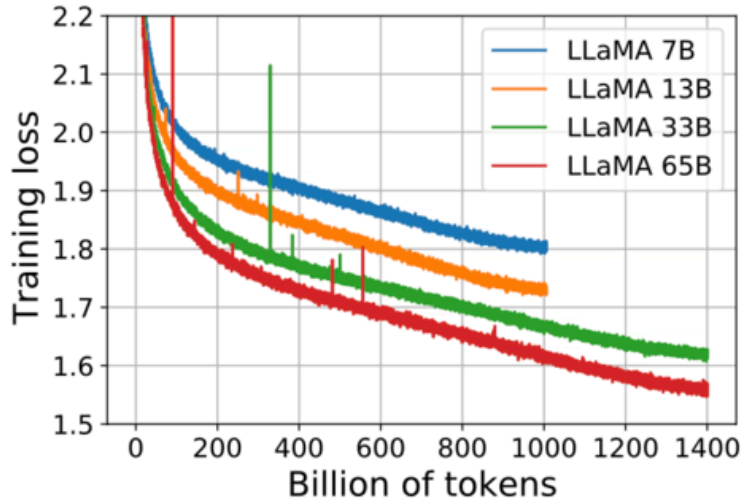


Figure 1: **Training loss over train tokens for the 7B, 13B, 33B, and 65 models.** LLaMA-33B and LLaMA-65B were trained on 1.4T tokens. The smaller models were trained on 1.0T tokens. All models are trained with a batch size of 4M tokens.

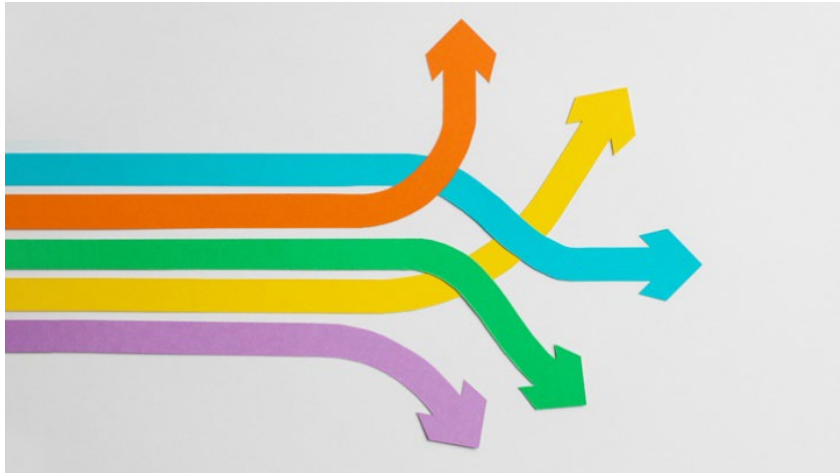
Moderne språkmodeller:

- Mange milliarder parametere (1.5 billioner for Gemini)
- Petabytes av nettdata som treningsgrunnlag
- Trenes med tusenvis av spesialisert maskinvare (GPUer) i mange dager

Tilpasning

- Etter trening kan språkmodellen *tilpasses* andre oppgaver enn å predikere det neste ordet:
  - F.eks. svare brukeren i en samtale, følge instruksjoner, klassifisere dokumenter, oppsummere eller oversette tekster osv.

- Flere tilpasningsstrategier:
  - **Finjustering**
  - **Prompting**
  - **Reinforcement learning**



Finjustering (fine-tuning)

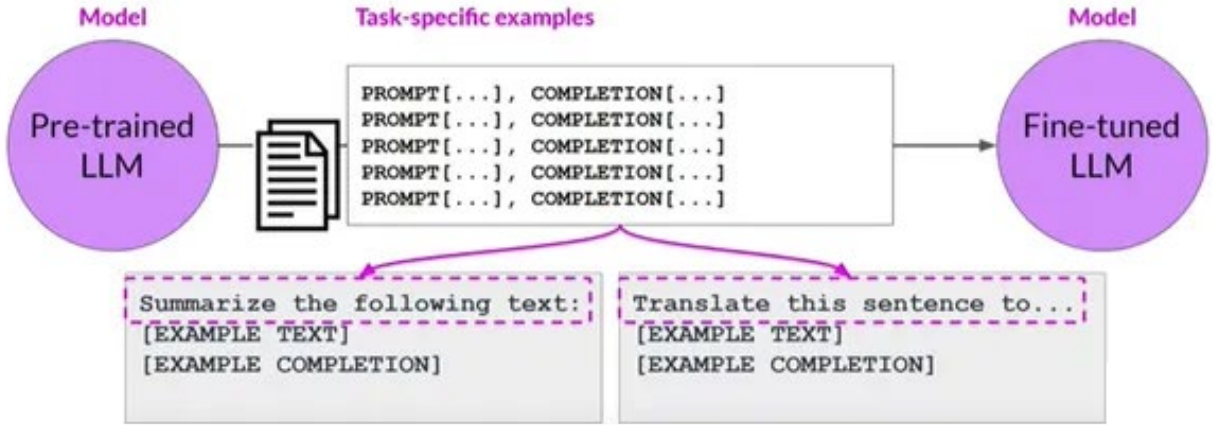
= “videretrening” av LLM for utføre *en bestemt oppgave*

- For eksempel klassifisere tekster eller svare på brukerspørsmål

- Typisk basert på *domenespesifikke data*
- Mange metoder, avhengig av språkmodellen, hvor mye data man har samlet inn, og regneressurser man har tilgjengelig
  - Se bl.a. “Parameter-efficient fine-tuning” (PEFT)

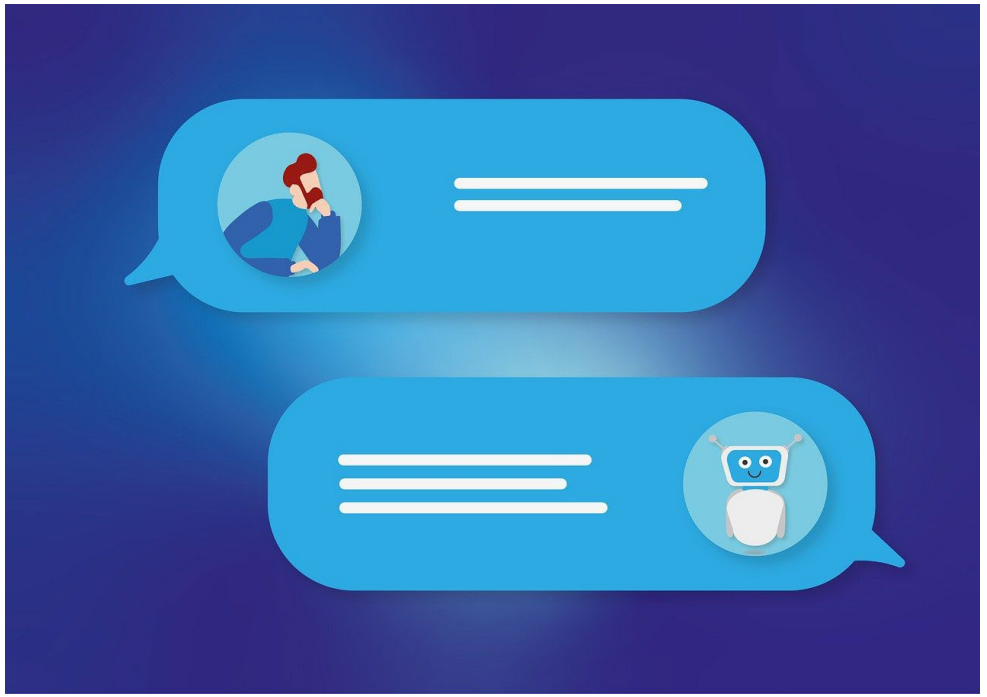
Instruction fine-tuning

- Systems like ChatGPT are not raw LLMs, they are specifically *fine-tuned* to follow instructions and/or engage in a dialogue with the user
- Many open-source LLMs have downloadable models that are instruction fine-tuned



Prompting

- En *prompt* er bare en instruksjon i tekstform som gis som input til en språkmodell
- Ulike formuleringer vil gi svært ulike resultater!
- Man kan også legge til *eksempler* av <input, output> i prompt



Såkalt “In-context learning”

Preference optimisation

- Vi kan også justere språkmodellen ved å *belønne* gode svar og *straffe* dårlige svar
- **Steg 1**: estimering av en *belønningsmodell*
  - Fra store mengder menneskelige vurderinger
  - Kan også brukes for å hindre uønskede produksjoner, f.eks. hatefull innhold
- **Steg 2**: optimering av språkmodellen for å maksimere sannsynligheten for å få "tommel opp" for svaret

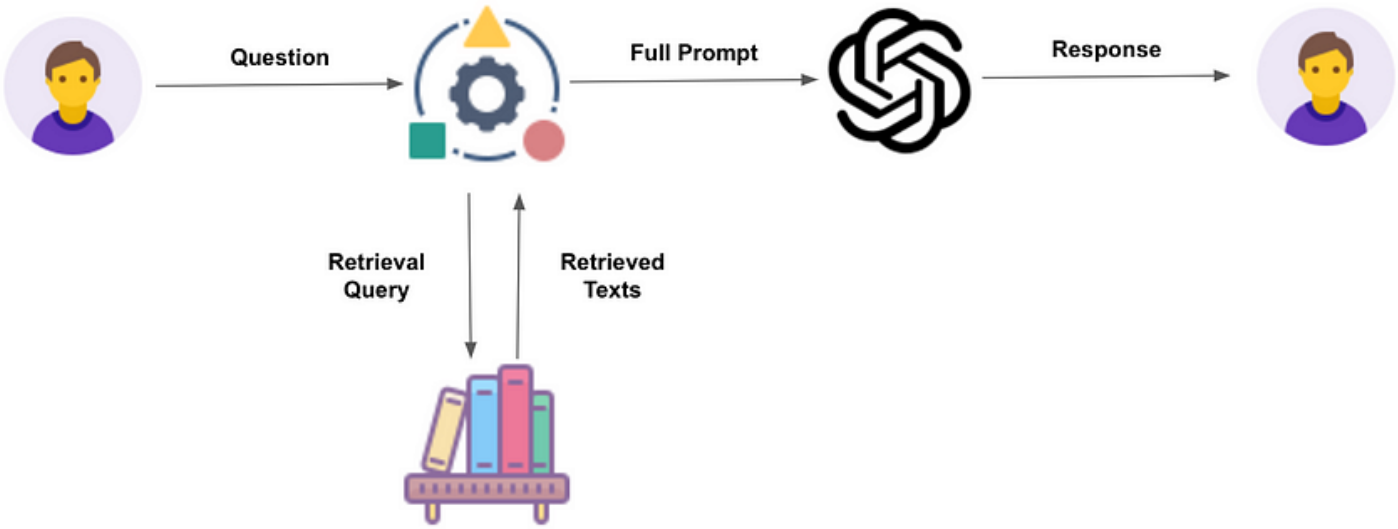
Nyere utviklinger

1. **Multimodale modeller**: trene språkmodellen med flere og rikere data enn bare tekst: **bilder**, videoer, lyd, sensordata, osv.

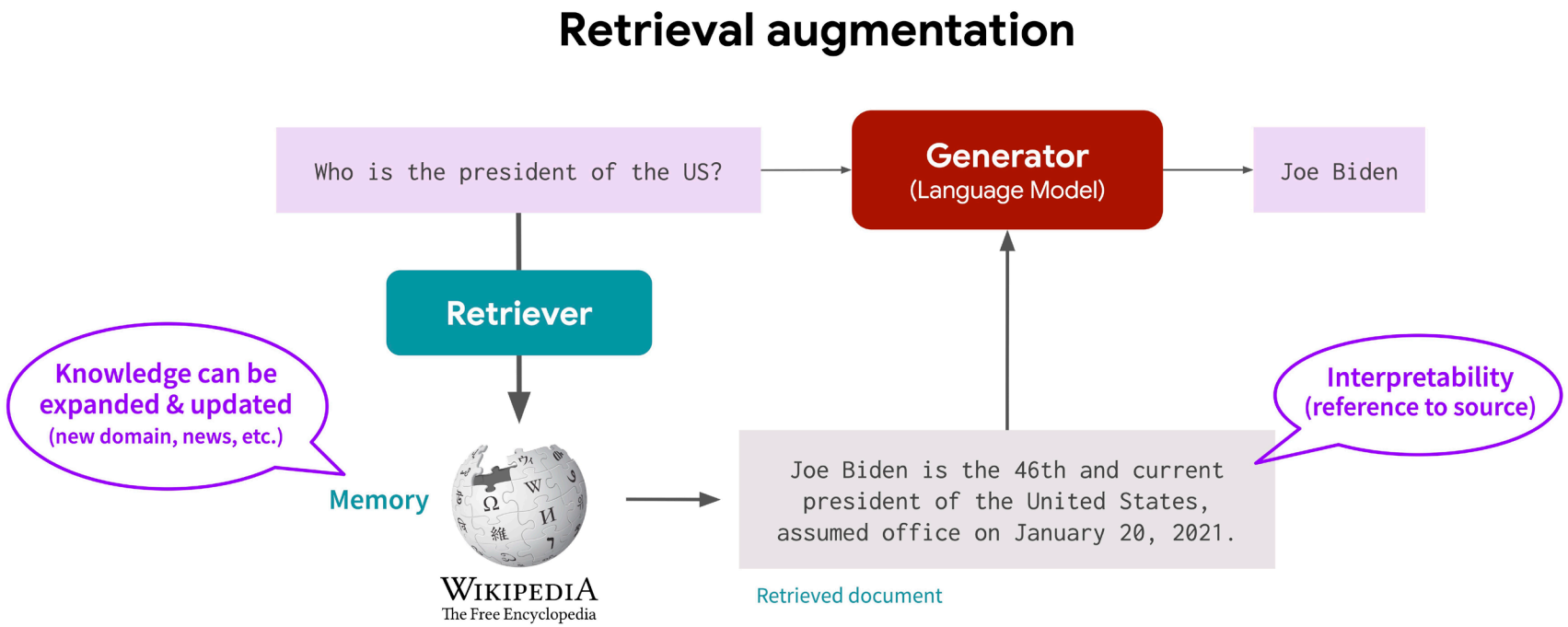
Nyere utviklinger

2. Retrieval-augmented Generation (RAG)

Idéen: språkmodellen leter først etter relevante opplysninger i en tekstdatabase og produsere sine svar basert på hva som ble funnet



Retrieval-augmented models



Benefits:

- Knowledge base can be easily inspected and updated (just add or remove documents)
- Can help reduce hallucinations
- Can operate on in-house data

Nyere utviklinger

Gemma

MISTRAL AI_

3. Stadig bedre **open-source språkmodeller**

→ Mulig å bruke store språkmodeller uten å være avhengig av skybaserte API-er som sender data ut av virksomheten!

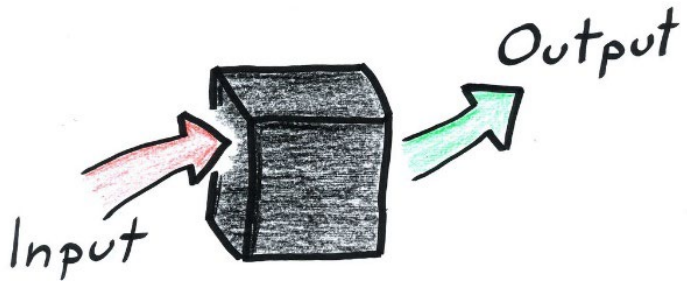
Write a short paragraph about the Norwegian Computing Center, and highlight one specific contribution in Norway.

The Norwegian Computing Center (Norsk Regnesentral or NR) is a renowned research institute and hub for computational science and technology in Norway. Established in 1952, NR has played a pivotal role in advancing the field of computer science and mathematics within the country. One of its notable contributions is the development of the NORHEIMSUND model, which has been instrumental in improving the efficiency and accuracy of weather forecasting in Norway. This model utilizes advanced computational algorithms to simulate and predict weather patterns, aiding meteorologists in providing more precise and timely weather forecasts for the region. The Norwegian Computing Center's dedication to innovation and cutting-edge research has had a lasting impact on various sectors in Norway, including meteorology and environmental sciences.

Utfordring 1: manglende saklighet

- Språkmodeller er optimert til å produsere *plausible* tekster, ikke nødvendigvis *korrekte* tekster!
  - Modellen har ingen forhold til «sannhet» som sådan
- Feil svar kan komme fra treningsdata, som kan inneholde ulike typer feil eller desinformasjon.....
- Men modellen kan fortsatt *hallusinere* med et "perfekt" treningssett!
- Og vil ofte gjøre det i en skråsikker tone

Utfordring 2: manglende styring

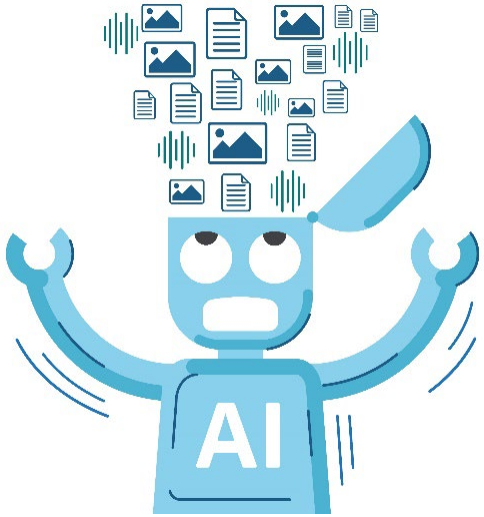


- Språkmodeller er "**svarte bokser**": vi forstår egentlig ikke hvorfor de genererer en viss respons
- Vi kan til en visst grad «styre» modellen mot en ønsket adferd
  - Gjennom finjustering eller «prompting» med egne eksempler, eller ved å belønne gode svar og straffe dårlige

➔ Men modellen kan fortsatt produsere uforutsigbare tekster …

- Språkmodeller er også nærmest umulige å redigere
  - hvordan kan f.eks. vi håndheve GDPRs *rett til å bli glemt*?

Bruk av språkmodeller



- Store språkmodeller er kraftige IT-verktøy som kan brukes for de fleste oppgaver som har med *tekst/tale* å gjøre
  - Fra skrivestøtte til oppsummering og informasjongjenfinning
- Men vi må også være klar over deres *begrensninger*
  - Bl.a. *manglende styring* og fare for *hallusinering*
  - Resonneringsevne er ofte overfladisk

Self-attention

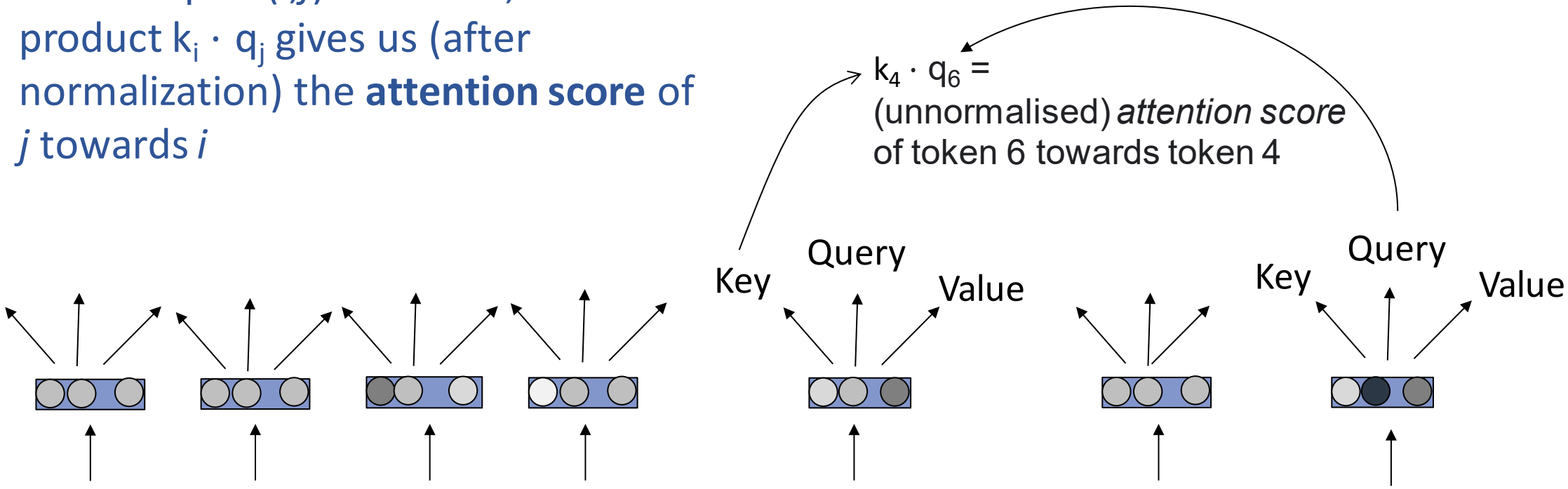
The text is first tokenized (into *wordpieces*)

And each token is mapped to a (learned) vector,
which is a numeric array like [-2.42, 0.566, 3.239, …]

Self-attention

Each token vector is then mapped via (learned) linear transformations to three vectors: the ***key*** $\mathbf{k}_i$, ***query*** $\mathbf{q}_i$ and ***value*** $\boldsymbol{v}_i$

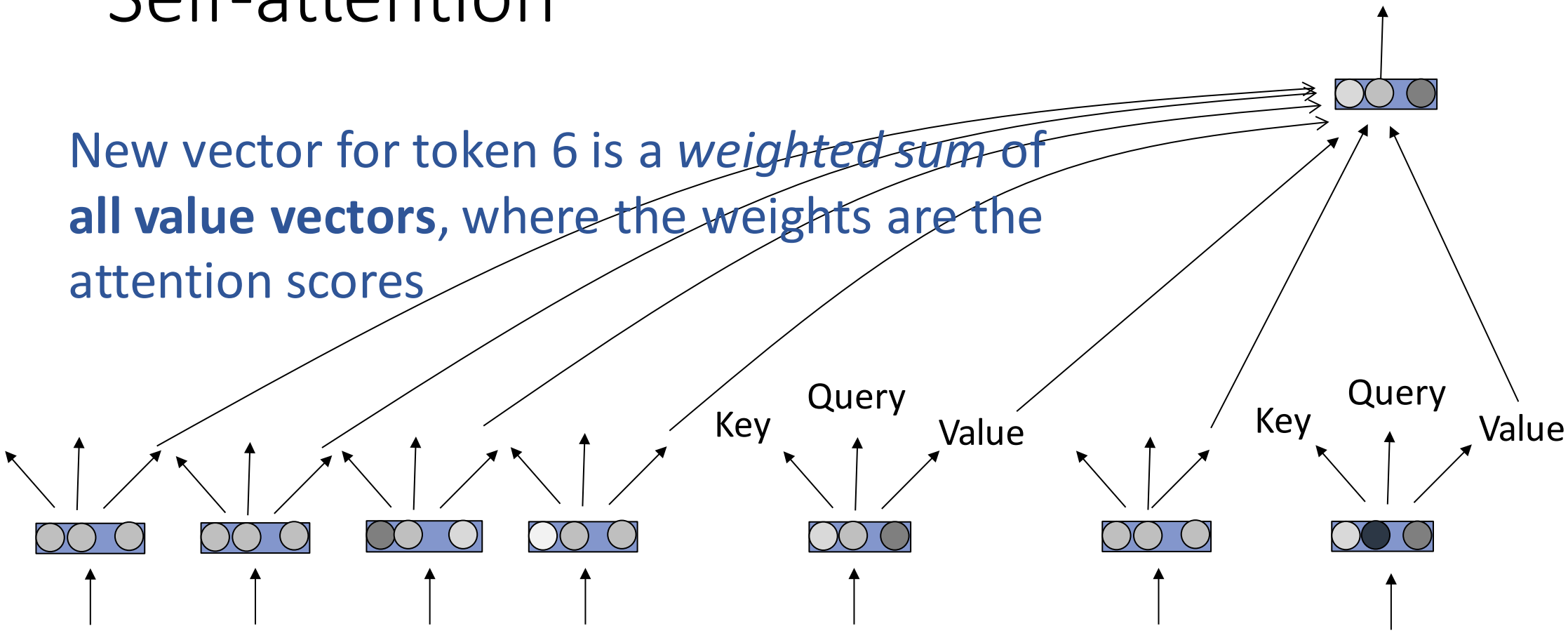
For each pair (*i*,*j*) of tokens, the dot product $k_i \cdot q_j$ gives us (after normalization) the **attention score** of *j* towards *i*



Self-attention

This vector is then further processed (classical feedforward network)

New vector for token 6 is a *weighted sum* of **all value vectors**, where the weights are the attention scores



jeg lurer på om forsikringen jeg tegnet …

0 1 2 3 4 5 6