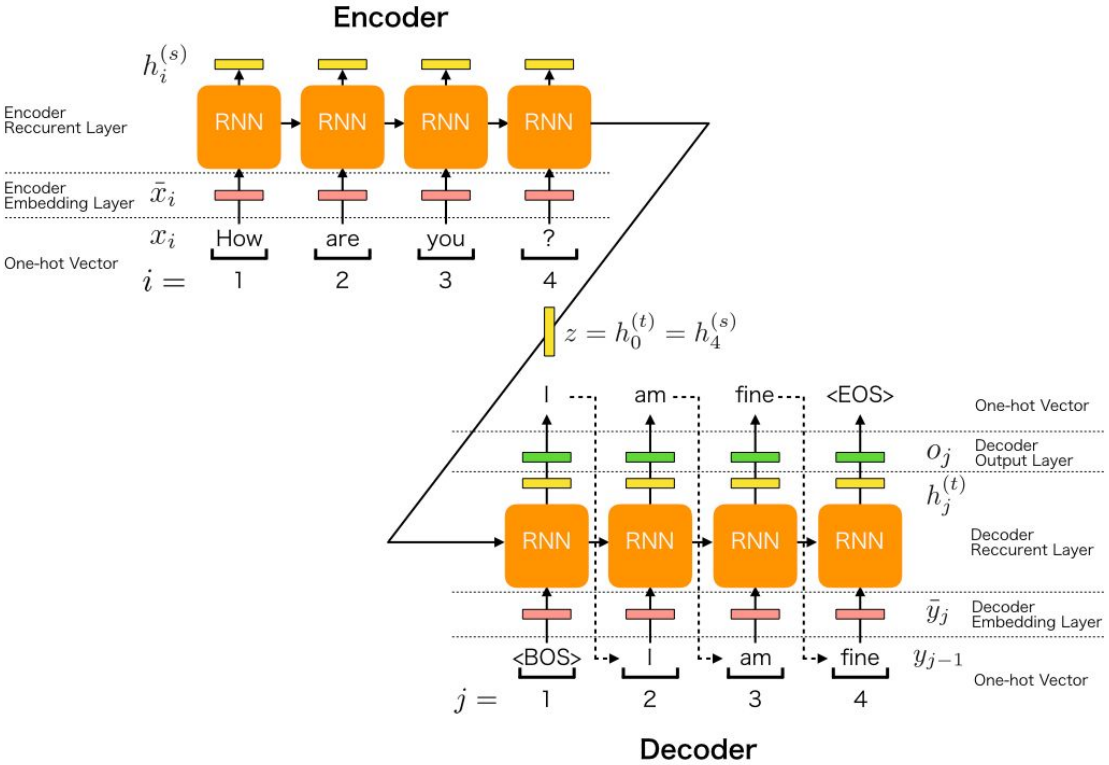
Transformers

Ηρακλής Βαρλάμης

Περιεχόμενα

- Transformers, Attention
- Encoders/decoders
- Pretrained transformers (BERT) και fine tuning
- LLM overview
- Retrieval Augmented Generation και Vector databases

Seq2seq tasks



- Μετάφραση
- Question answering
- Prompting

Η ακολουθία λέξεων στην έξοδο (decoder) λαμβάνει υπόψη της την ακολουθία στην είσοδο (encoder)

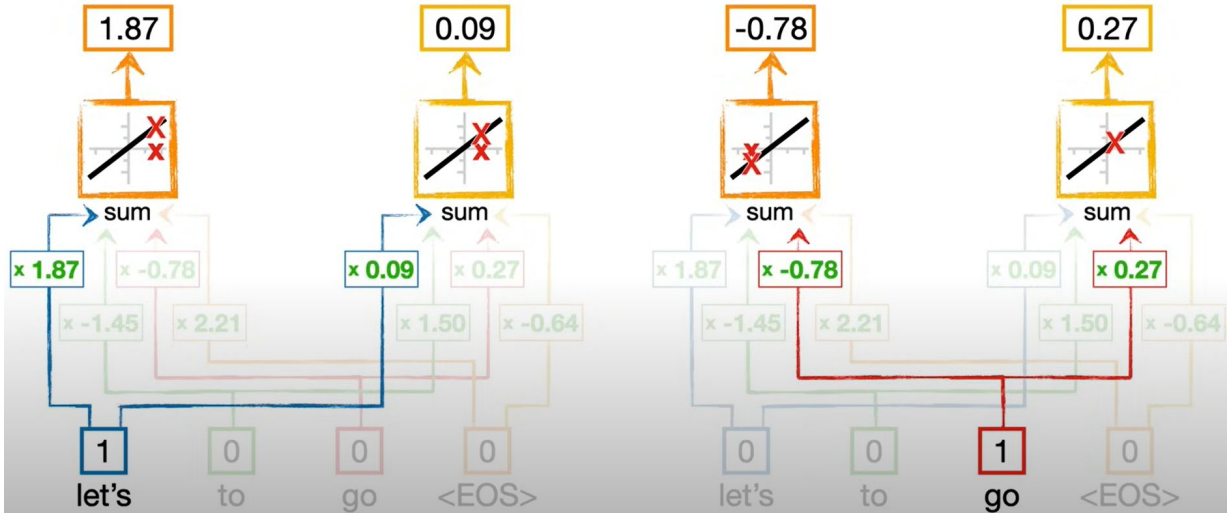
Κάθε λέξη (token) w εξαρτάται από τις λέξεις που προηγήθηκαν (conditional probabilities)

Embedding layer

Ένα δίκτυο με εισόδους όσο το μέγεθος του λεξικού και εξόδους όσο το επιθυμητό μέγεθος του διανύσματος αναπαράστασης

Τροφοδοτείται από one-hot vector embeddings για κάθε λέξη μιας πρότασης

Δίνει την dense αναπαράσταση της λέξης στην έξοδο



Προβλήματα των απλών νευρωνικών δικτύων

- Η αρχιτεκτονική encoder-decoder (seq2seq) μπορεί να θεωρηθεί ένα γλωσσικό μοντέλο βασισμένο σε πιθανότητες $P(y^{<1>}, \ldots, y^{<Ty>} \mid x^{<1>}, \ldots, x^{<Ty>})$
- Τα απλά νευρωνικά δίκτυα μαθαίνουν γνωρίσματα για τα tokens ανεξάρτητα από τη θέση τους
- Οι ακολουθίες εισόδου και εξόδου έχουν διαφορετικό μήκος
- Τα RNN είναι μια λύση στην κατεύθυνση αυτή

Transformers

Είναι στην ουσία αρχιτεκτονικές νευρωνικών δικτύων που προσπαθούν να αναπαραστήσουν όσο το δυνατόν περισσότερη πληροφορία από μια πρόταση

Να αναπαραστήσουν προτάσεις με τέτοιο τρόπο που να διακρίνονται οι διαφορές τους

Οι διαφορές:

- Οι ***λέξεις*** που απαρτίζουν κάθε πρόταση
- Η ***θέση*** των λέξεων στην κάθε πρόταση
- Οι ***σχέσεις*** μεταξύ των λέξεων στην πρόταση

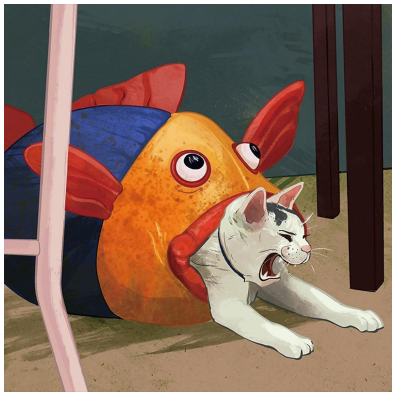
Word embeddings

...λέξεις....

- Τα **word embeddings** είναι η (static) αναπαράσταση κάθε λέξης
- Μπορούν να αντικατασταθούν από one-hot representations και ένα embedding layer στην είσοδο (ώστε να τα μαθαίνουμε κατά την εκπαίδευση)
- Τα word embeddings δεν λαμβάνουν υπόψη τη θέση της λέξης στην πρόταση

'cat eats fish'

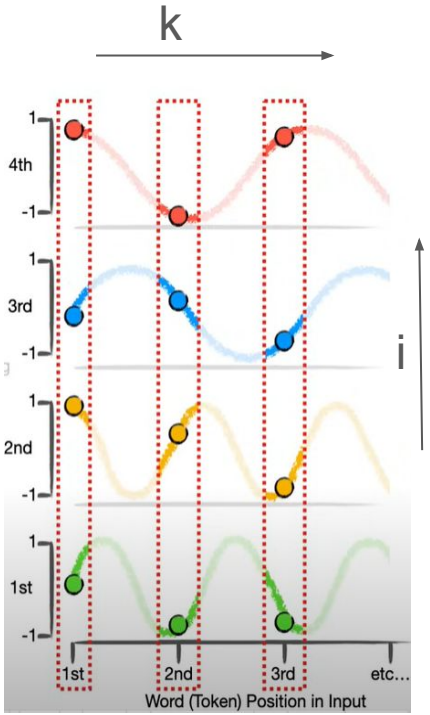
'fish eats cat'



Positional Encoding

...θέσεις....

- Χρησιμοποιούμε εναλλακτικά ημίτονα και συνημίτονα διαφορετικής συχνότητας
- Αν το positional vector θέλουμε να έχει διάσταση i (όσο και το word vector) θα χρησιμοποιήσουμε i συναρτήσεις (ημιτόνου/συνημιτόνου)
- Για κάθε λέξη της πρότασης, ανάλογα με τη θέση της k παίρνουμε τις αντίστοιχες τιμές (μεταξύ -1 και 1) από τις i συναρτήσεις
- Προσθέτουμε τα positional vector στο word embedding για να πάρουμε το positional encoding κάθε λέξης (που διαφέρει ανάλογα με τη θέση της)



For even indices (i):

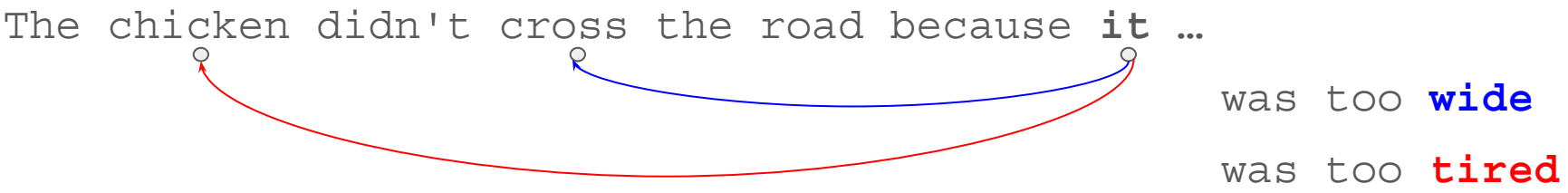
$$P(k, 2i) = \sin\left(\frac{k}{10000^{2i/d}}\right)$$

For odd indices (i):

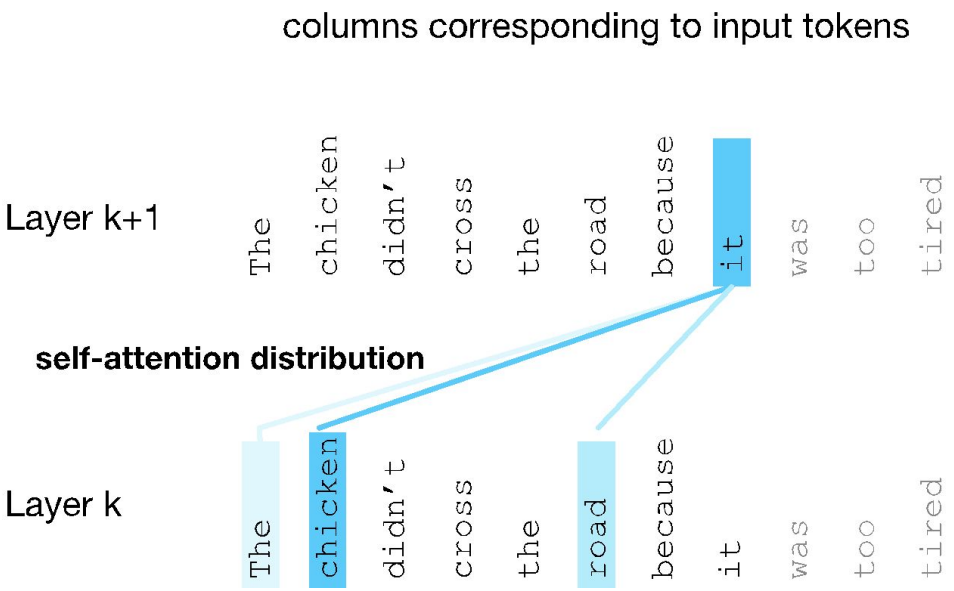
$$P(k, 2i+1) = \cos\left(\frac{k}{10000^{2i/d}}\right)$$

Contextual embeddings - Attention *...σχέσεις....*

- Ούτε τα word embeddings, ούτε τα positional encodings λαμβάνουν υπόψη το περιβάλλον της λέξης
- Η ιδέα του **attention** αλλάζει την αρχιτεκτονική του δικτύου ώστε να λαμβάνει υπόψη και το περιβάλλον κάθε λέξης στην αναπαράστασή της ⇒ **contextual embeddings**



Attention και Self-attention



- μια μέθοδος για να βρίσκουμε ένα ***weighted sum*** των word embeddings της πρότασης ***για κάθε λέξη στην πρόταση***

- είναι ένα weighted sum της ομοιότητας του word embedding της λέξης με όλες τις λέξεις της πρότασης (και με τον εαυτό της)

Attention masking

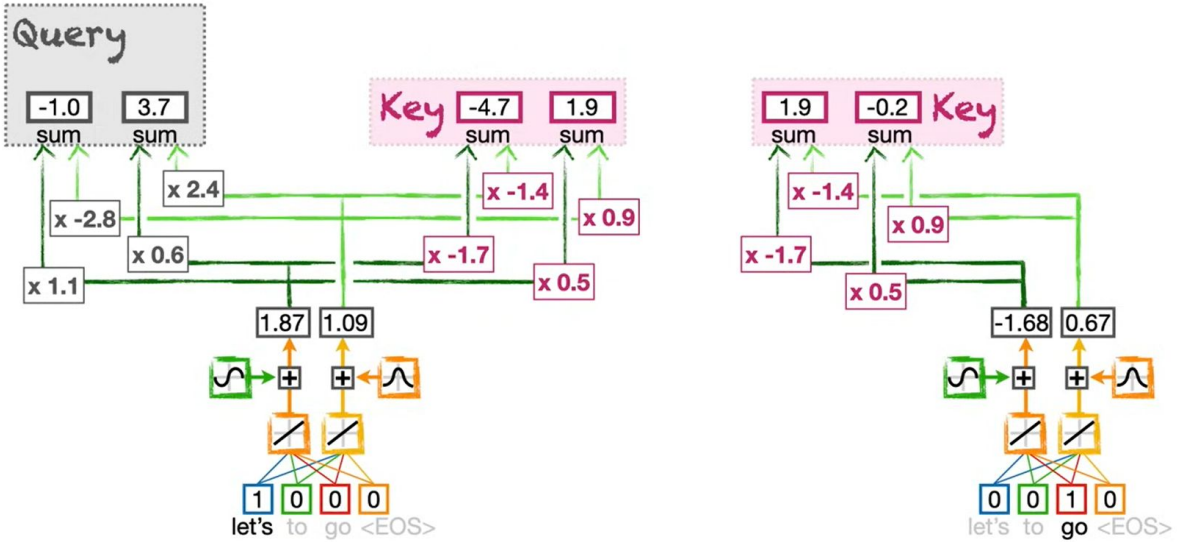
- Δε λαμβάνουμε πάντοτε υπόψη το attention από όλες τις λέξεις μιας πρότασης. π.χ.
  - Κατά την παραγωγή μιας απάντησης λαμβάνουμε υπόψη το cross-attention από τις λέξεις της ερώτησης
  - Κατά την παραγωγή μιας απάντησης λαμβάνουμε υπόψη μόνο το self-attention από τις λέξεις που προηγούνται του τελευταίου token στην απάντηση.

https://www.youtube.com/watch?v=zxQyTK8quyY&ab_channel=StatQuestwithJoshStarmer

(Self-)similarity

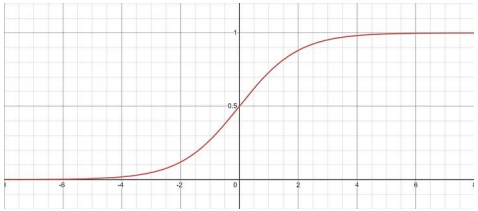
Αναπαριστούμε κάθε λέξη ως Query και ως Key

Υπολογίζουμε το Dot Product (QK^T) κάθε λέξης (query) και όλων των λέξεων (keys) της πρότασης



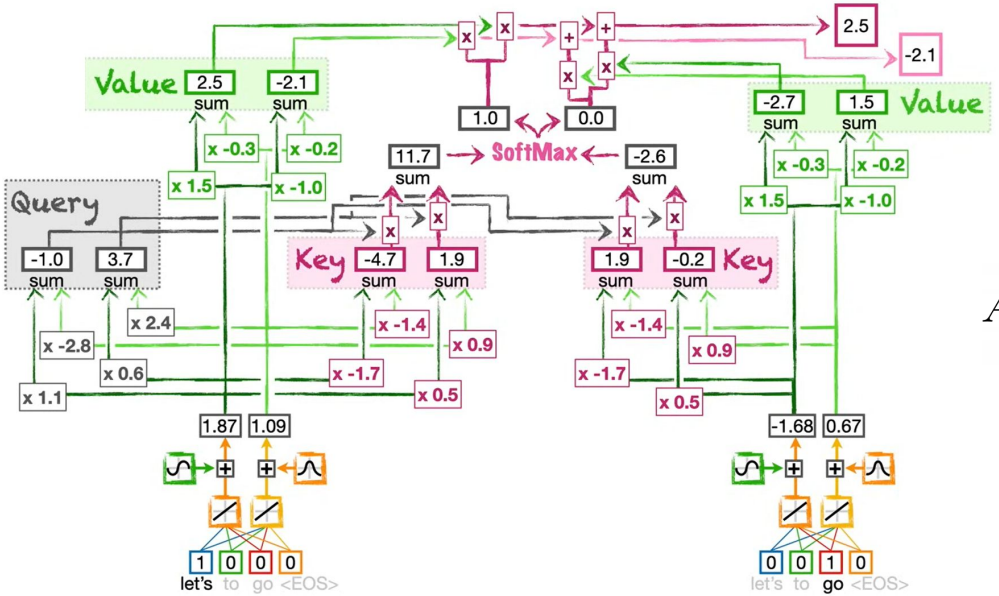
Self-attention

$$\frac{e^{z_i}}{\sum_{j=1}^{n} e^{z_j}}$$



Περνούμε το scaled self-similarity από μια SoftMax

Παίρνουμε το DotProduct της SoftMax με κάθε Value

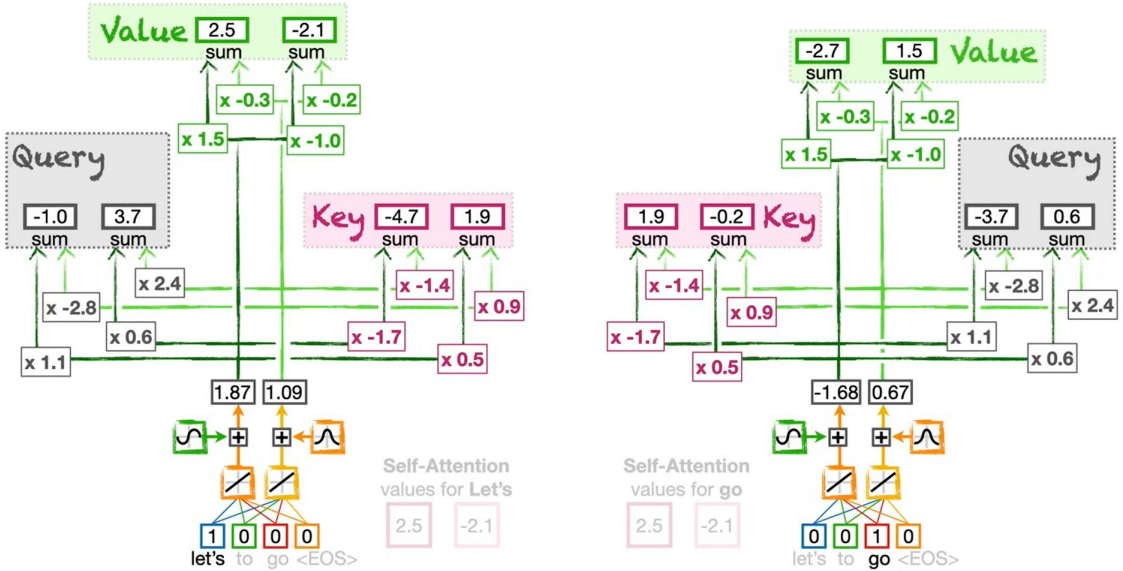


$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V$$

d_k οι διαστάσεις των embeddings

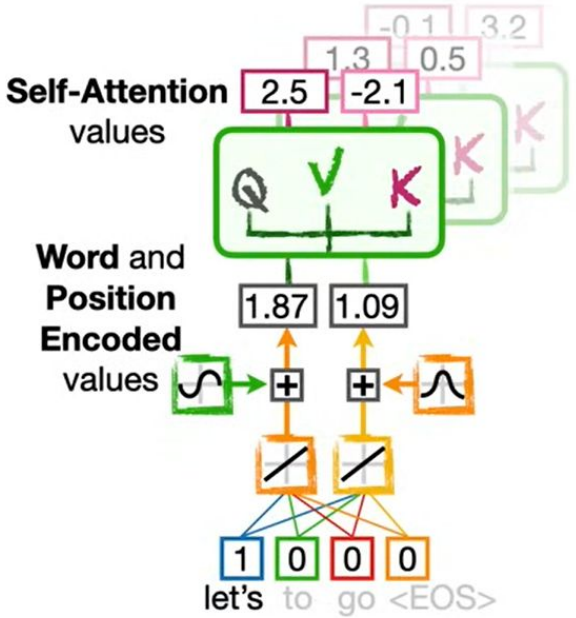
Παράλληλοι υπολογισμοί

Τα ίδια Q,K,V vectors παράγονται για κάθε λέξη κατά τον υπολογισμό των self-attention



Multi-head attention

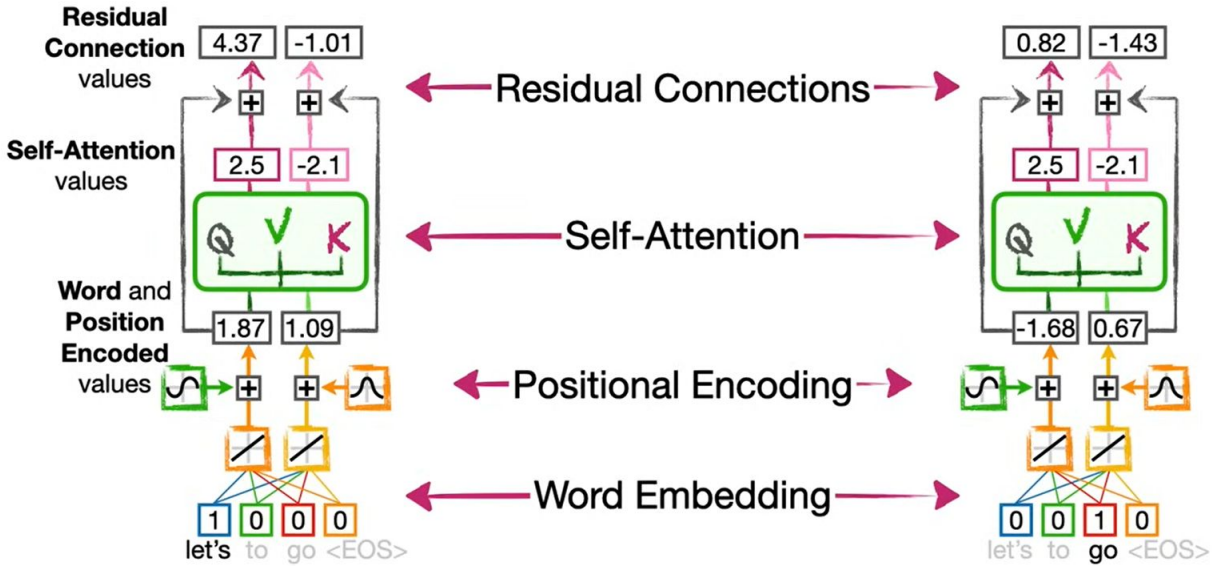
Επαναλαμβάνουμε τη διαδικασία για διαφορετικά βάρη ώστε να παράγονται άλλα Q,V,K άρα και επιπλέον self-attention values



Residual connections

Προσθέτουμε τα word και positional encodings απευθείας στα self-attention values

Έτσι διασφαλίζουμε ότι δεν “χάνονται” τα word και positional encodings

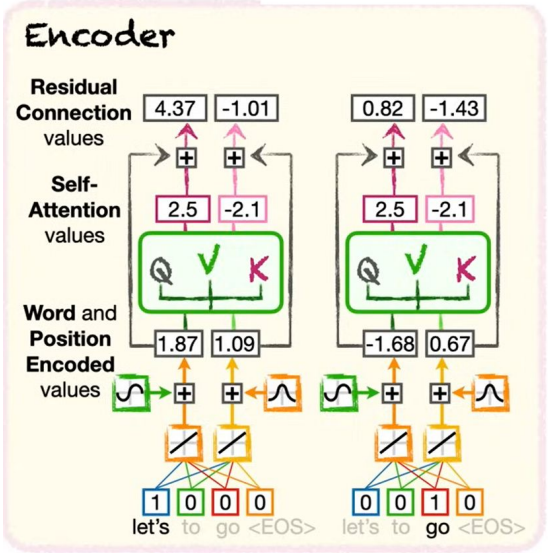


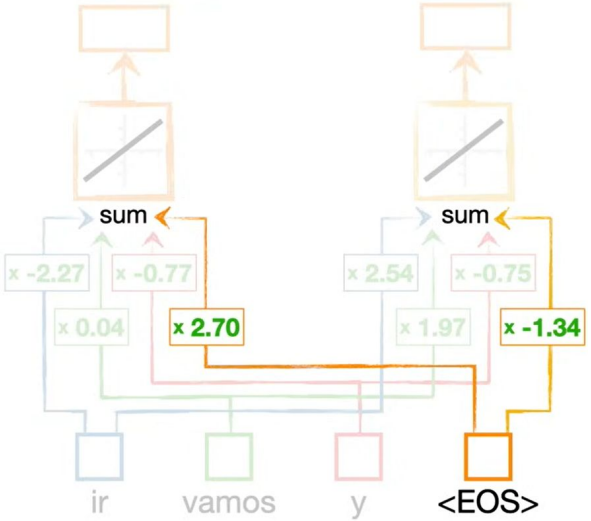
Περιεχόμενα

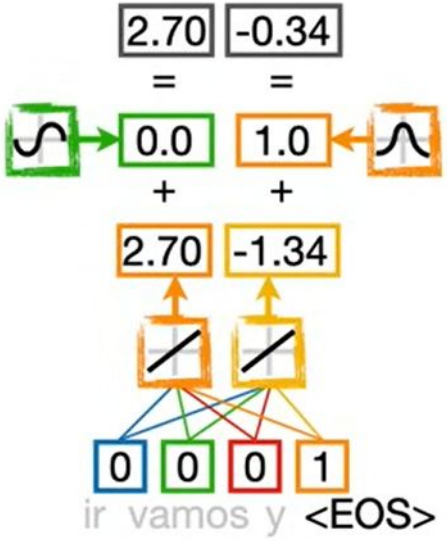
- Transformers, Attention
- Encoders/decoders
- Pretrained transformers (BERT) και fine tuning
- LLM overview
- Retrieval Augmented Generation και Vector databases

Encoder-Decoder (translation)

Αφού ολοκληρώσουμε με το self-attention του encoder, ξεκινάμε με τα word embeddings και τα position embeddings του τελευταίου token



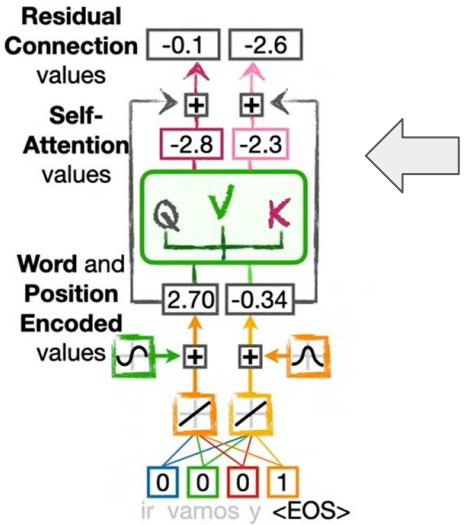


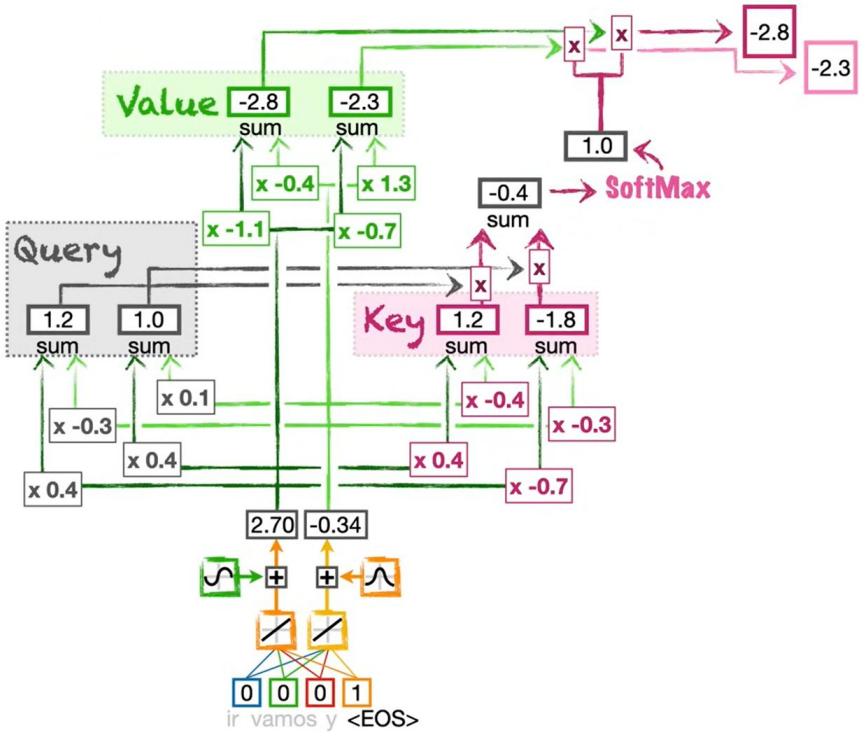


Decoder’s Self-attention

Χρησιμοποιώντας διαφορετικά βάρη υπολογίζουμε με τον ίδιο τρόπο τα Q,K,V και το self-attention στον decoder για το πρώτο token

Προσθέτουμε και τα residual connections

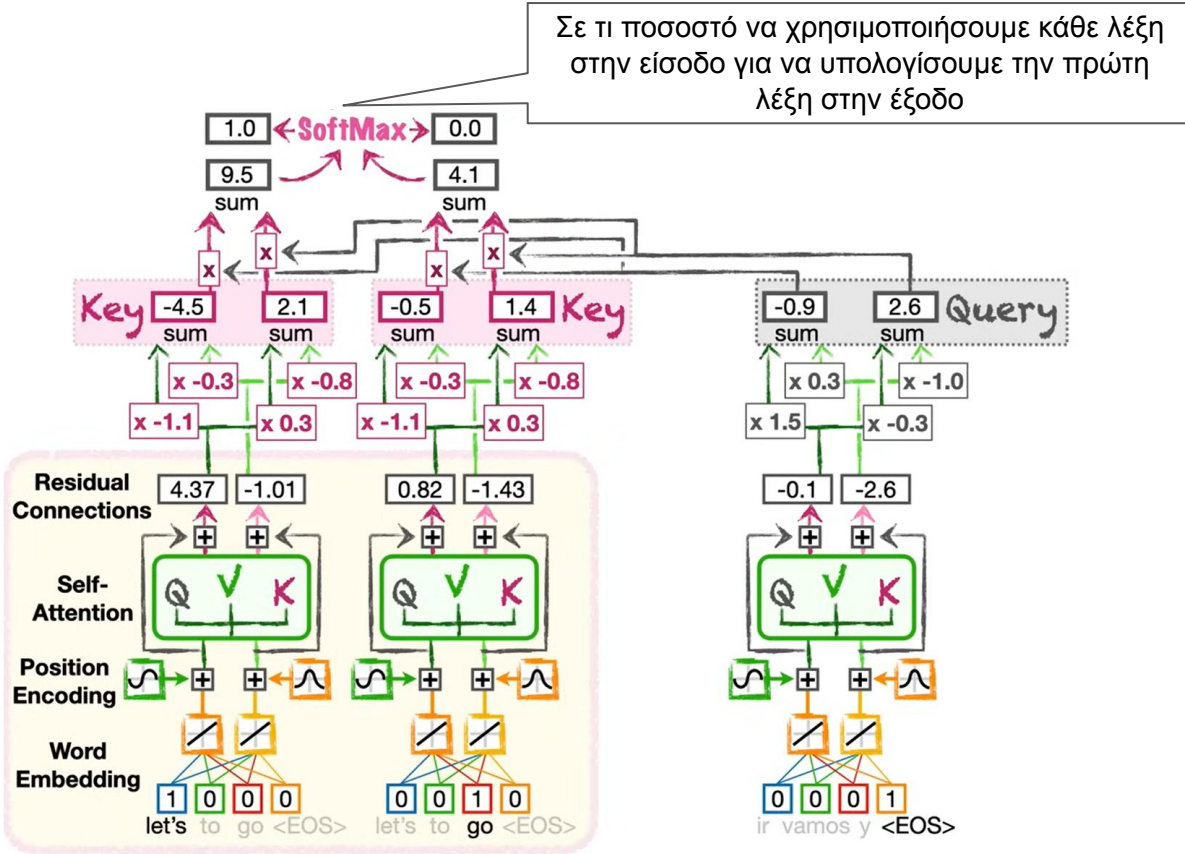




Χρήση του encoded sequence στον decoder

Χρησιμοποιώντας διαφορετικά βάρη υπολογίζουμε πόσο επηρεάζει κάθε token στην είσοδο (Key) την επόμενη λέξη στην έξοδο (Query)

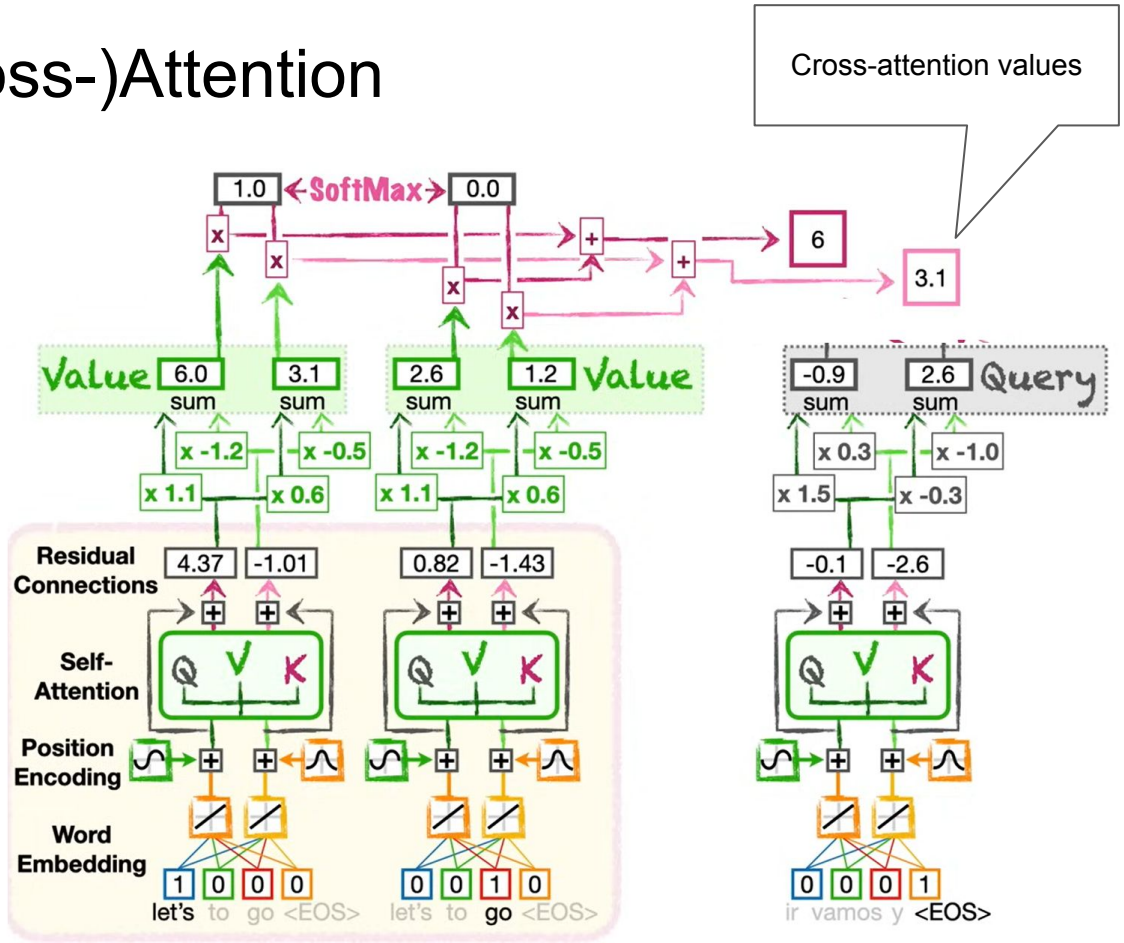
Περνάμε και πάλι από μια SoftMax



Encoder-Decoder (Cross-)Attention

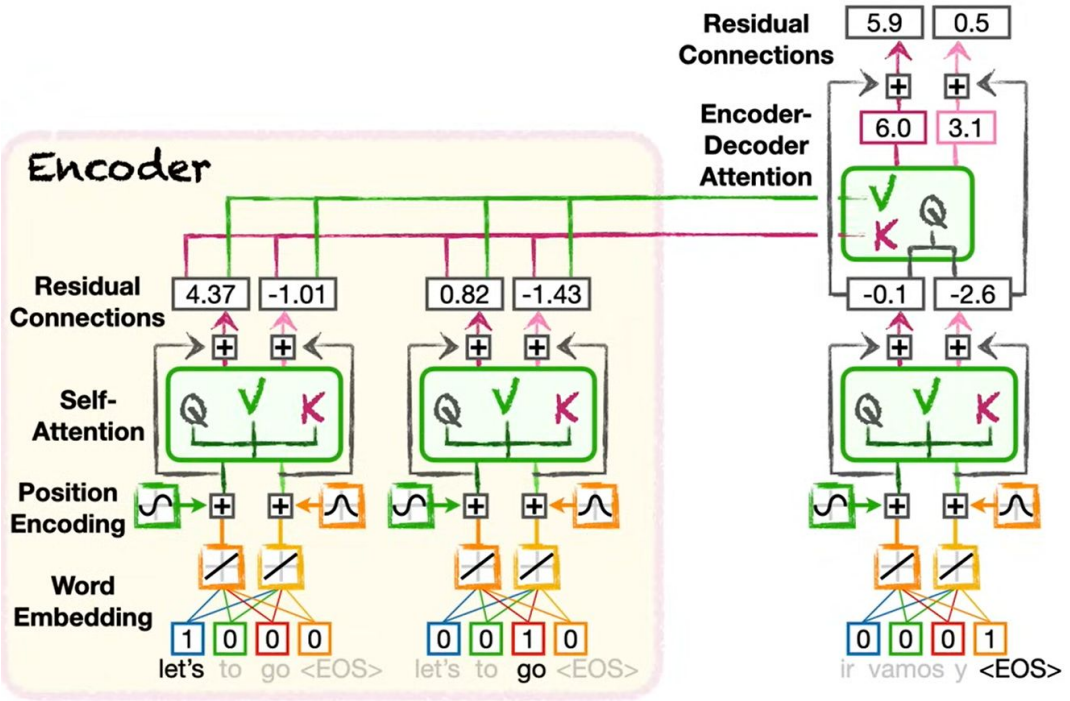
Υπολογίζουμε τα Values (V) για τα tokens στην είσοδο και αφού τα κάνουμε scale με την SoftMax τα προσθέτουμε για να πάρουμε το Encoder-Decoder Attention του πρώτου token στην έξοδο

Τα βάρη είναι διαφορετικά



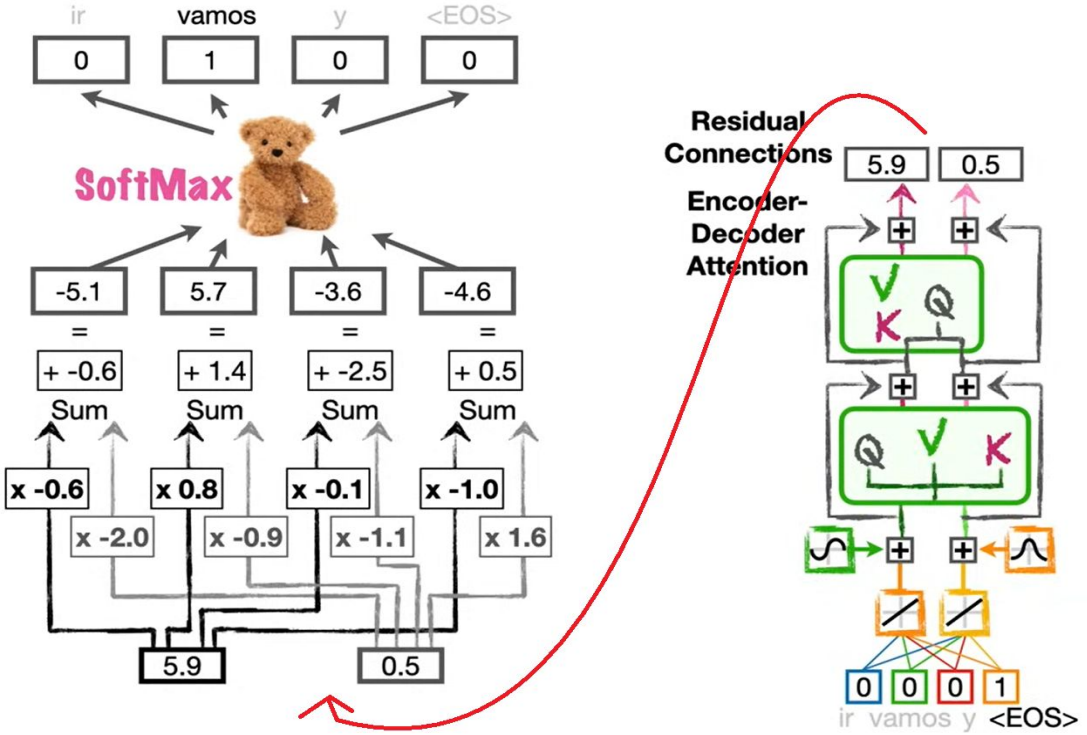
Τελική αναπαράσταση

Προσθέτουμε τα residual connections, ώστε το Encoder-Decoder attention να μην χαλάσει την πληροφορία του Self-attention του 1ου token αλλά και τα word και position encodings του.

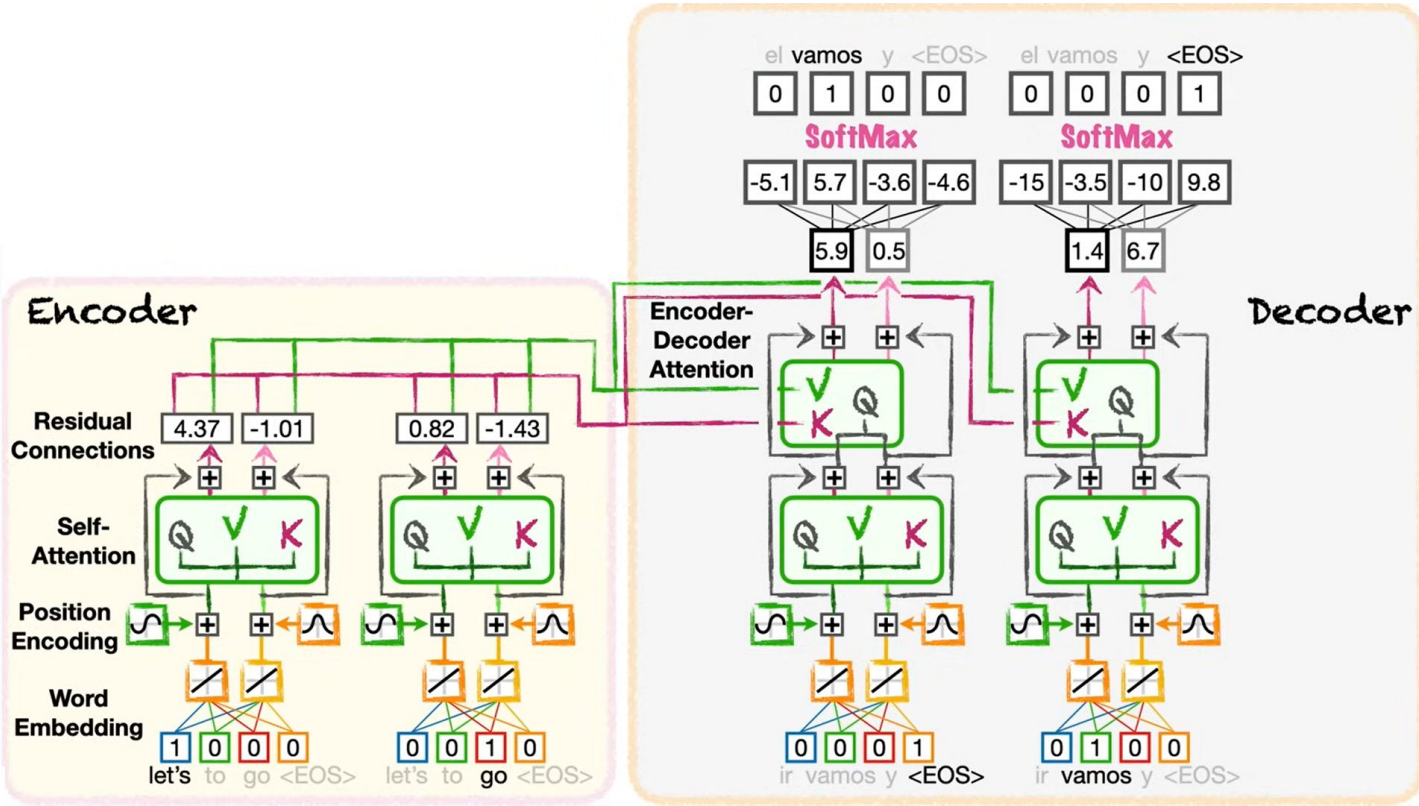


Υπολογισμός του επόμενου token

Η έξοδος του decoder πάει σε ένα dense network (με διαφορετικά βάρη) που με τη βοήθεια μιας SoftMax μετατρέπει την έξοδο σε ένα one-hot vector στο λεξικό των tokens της εξόδου (π.χ. σε άλλη γλώσσα ή στην γλώσσα απάντησης)



Τελικά

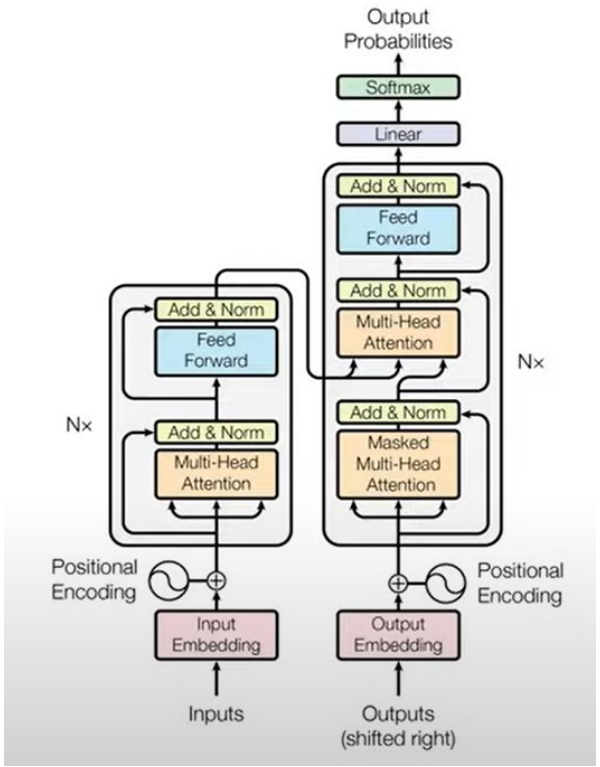

Encoder
Decoder
Encoder-Decoder Attention
Residual Connections
Self-Attention
Position Encoding
Word Embedding
4.37
-1.01
0.82
-1.43
let's to go <EOS>
let's to go <EOS>
el vamos y <EOS>
el vamos y <EOS>
SoftMax
SoftMax
-5.1 5.7 -3.6 -4.6
-15 -3.5 -10 9.8
5.9 0.5
1.4 6.7
ir vamos y <EOS>
ir vamos y <EOS>
Q
K
V

Περιεχόμενα

- Transformers, Attention
- Encoders/decoders
- Pretrained transformers (BERT) και fine tuning
- LLM overview
- Retrieval Augmented Generation και Vector databases

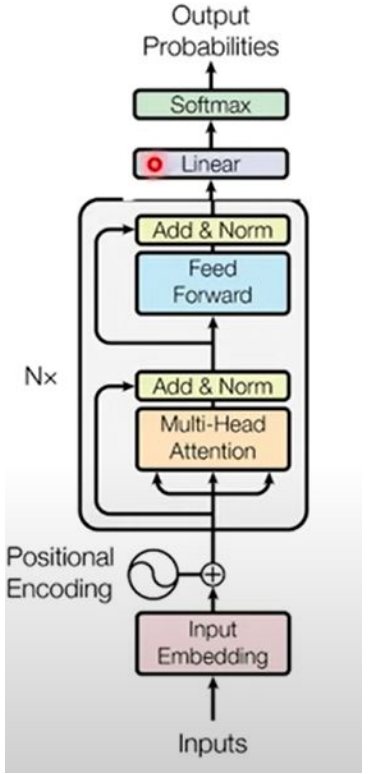
Transformers, GPT-2, and BERT

1. Ο transformer χρησιμοποιεί μια **στοίβα encoders** και **decoders**
2. **GPT:** Αν δεν έχουμε κάποια είσοδο και απλά θέλουμε να μοντελοποιήσουμε την “επόμενη λέξη” μπορούμε να αφαιρέσουμε τον encoder και να παράγουμε την “επόμενη λέξη” μια προς μία
3. **BERT:** Αν μας ενδιαφέρει να εκπαιδεύσουμε ένα γλωσσικό μοντέλο για ένα task, τότε δε χρειαζόμαστε τον decoder.



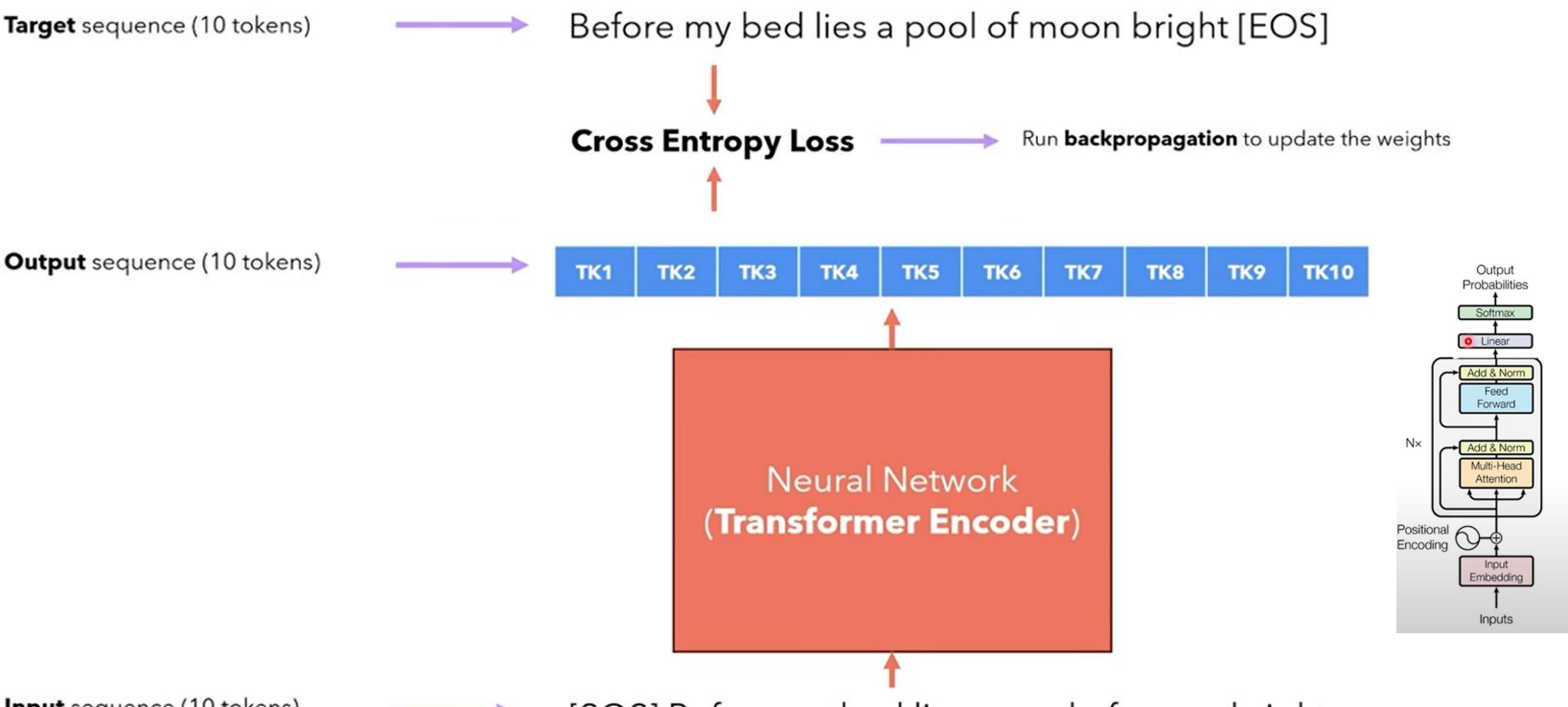
Εκπαίδευση ενός transformer encoder (LLM)

- Οι transformers συνήθως χρησιμοποιούν ημι-επιβλεπόμενη μάθηση με:
  - Μη επιβλεπόμενη προεκπαίδευση σε ένα πολύ μεγάλο σύνολο δεδομένων γενικού κειμένου
  - Ακολουθούμενη από επιβλεπόμενο fine-tuning σε ένα εστιασμένο σύνολο δεδομένων εισόδων και εξόδων για μια συγκεκριμένη εργασία
- Οι εργασίες που σχετίζονται με την προεκπαίδευση και τη λεπτομερή εκπαίδευση περιλαμβάνουν συνήθως:
  - Μοντελοποίηση γλώσσας (language modeling)
  - Πρόβλεψη επόμενης πρότασης (sentence completion)
  - Απαντήσεις σε ερωτήσεις (question answering)
  - Κατανόηση κειμένου (reading comprehension)
  - Ανάλυση συναισθήματος (sentiment analysis)
  - Παραφράσεις (paraphrasing)

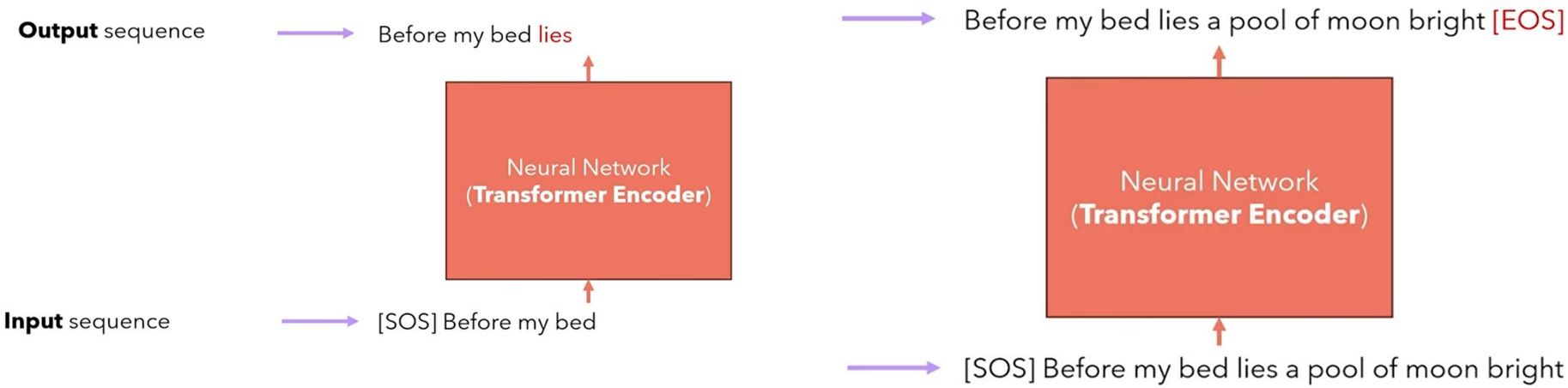


https://www.youtube.com/watch?v=90mGPxR2GgY&ab_channel=UmarJamil

Training a transformer



Προγνώσεις ενός εκπαιδευμένου transformer (LLM)



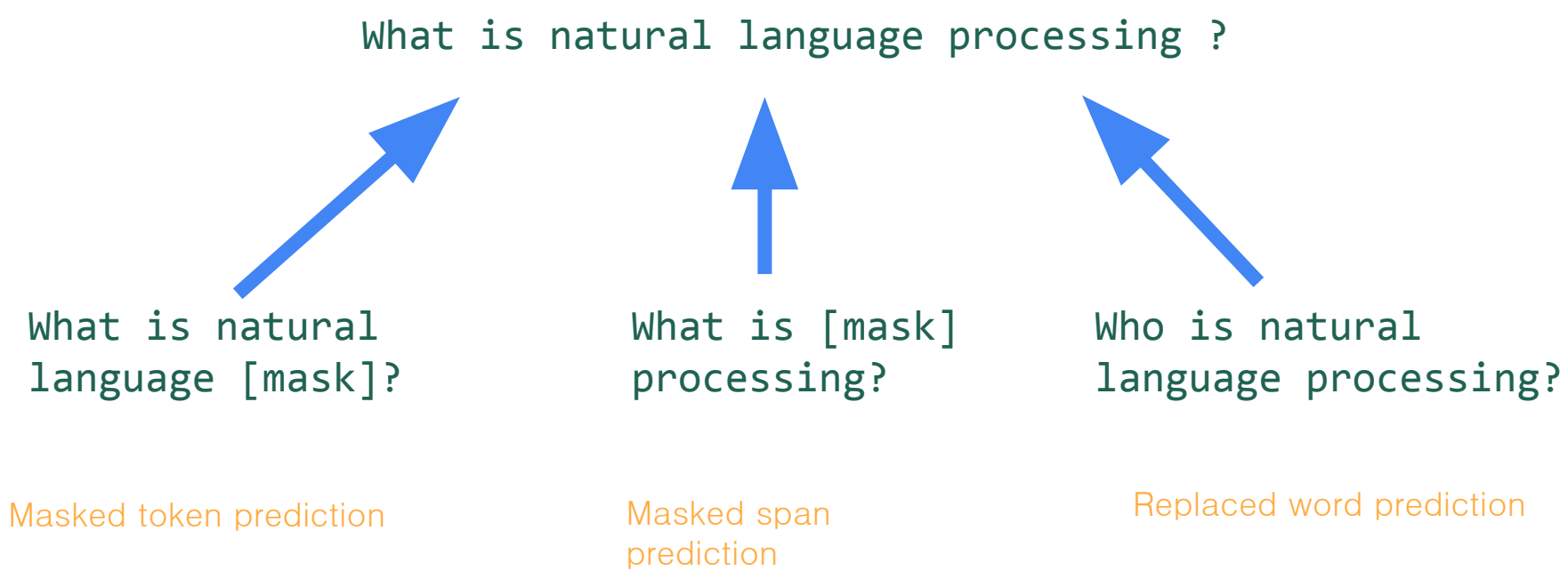
Next token prediction task

Έτοιμα προεκπαιδευμένα μοντέλα

- BERT: Bidirectional Encoder Representations from Transformers
- Είναι ένας transformer με stacked transformer layers προεκπαιδευμένο σε δύο unsupervised tasks: masked language modeling (fill-in-the-blank) και next sentence prediction (binary classification), σε κείμενα από την Αγγλική Wikipedia (2.5B words) και το BooksCorpus (800M words)
- $BERT_{BASE}$= 12 stacked transformer layers, 768 embedding vector, 12 attention heads
- $BERT_{LARGE}$= 12 stacked transformer layers, 1024 embedding vector, 16 attention heads
- Τα positional embeddings είναι learnable
- Χρησιμοποιεί τον WordPiece tokenizer (επιτρέπει sub-word tokens)
- Πλάτος λεξικού ~30.000 tokens

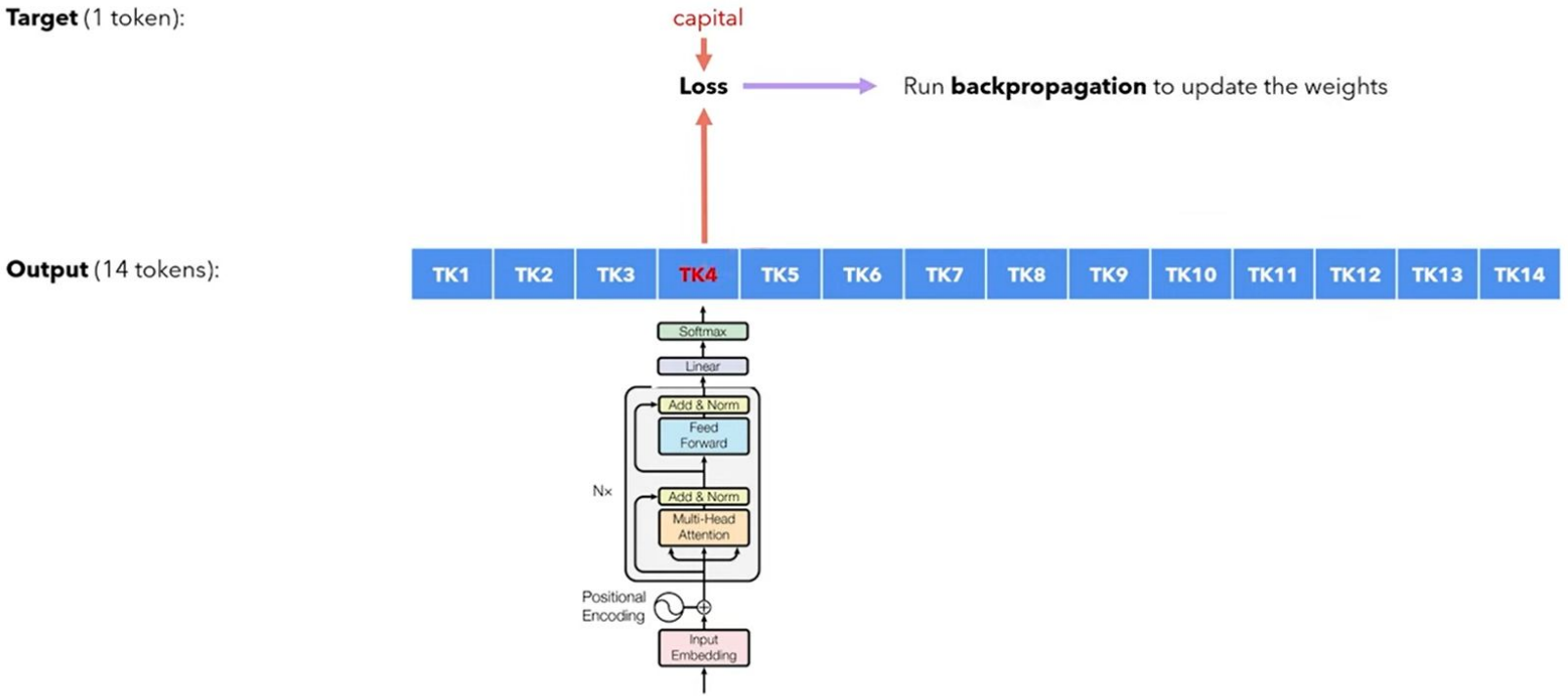
BERT masking tasks

A bidirectional training task



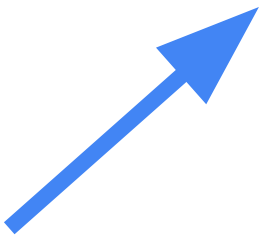
80% masking, 10% replacement, 10% nothing

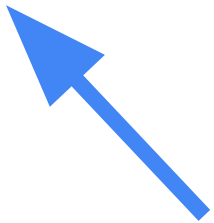
Training BERT (Masked Language Model task)



BERT next sentence prediction task

What is natural language processing ?





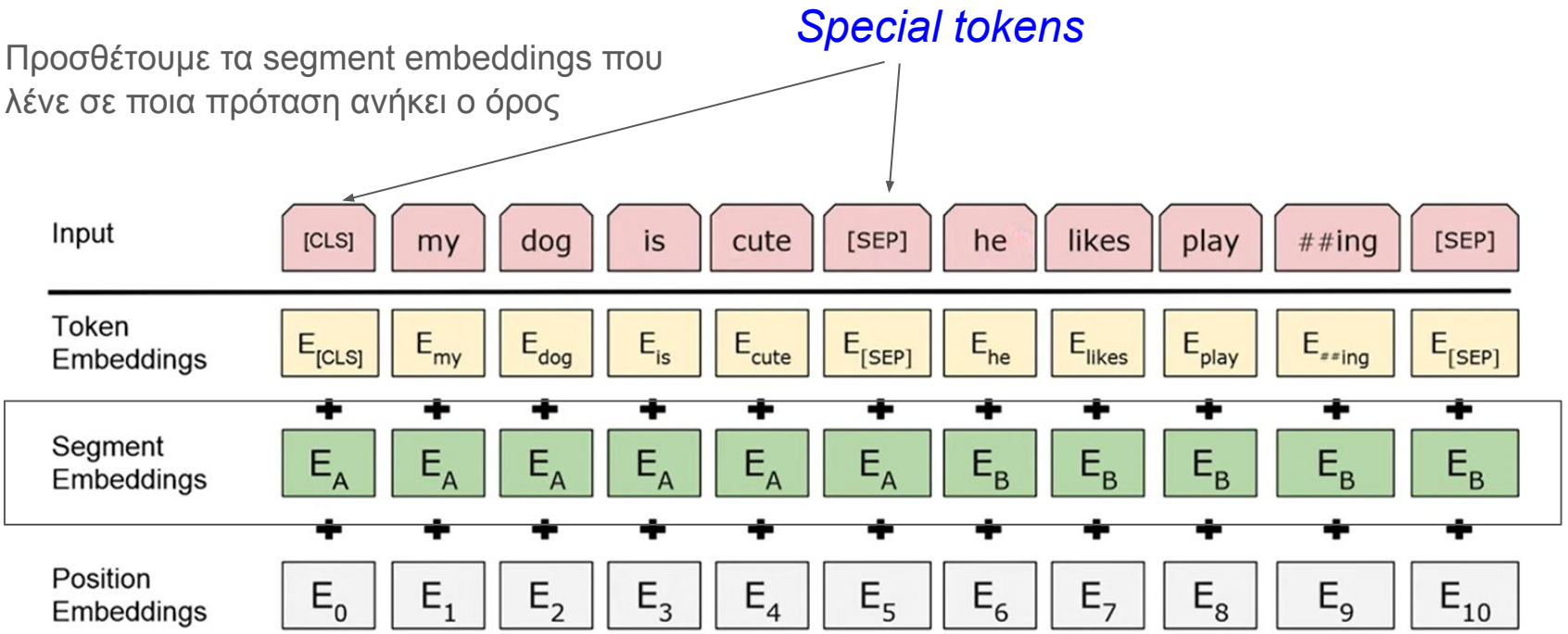
Natural Language Processing (NLP) is a field of AI that focuses on the interaction between computers and human language.

The sun sets over the calm ocean.

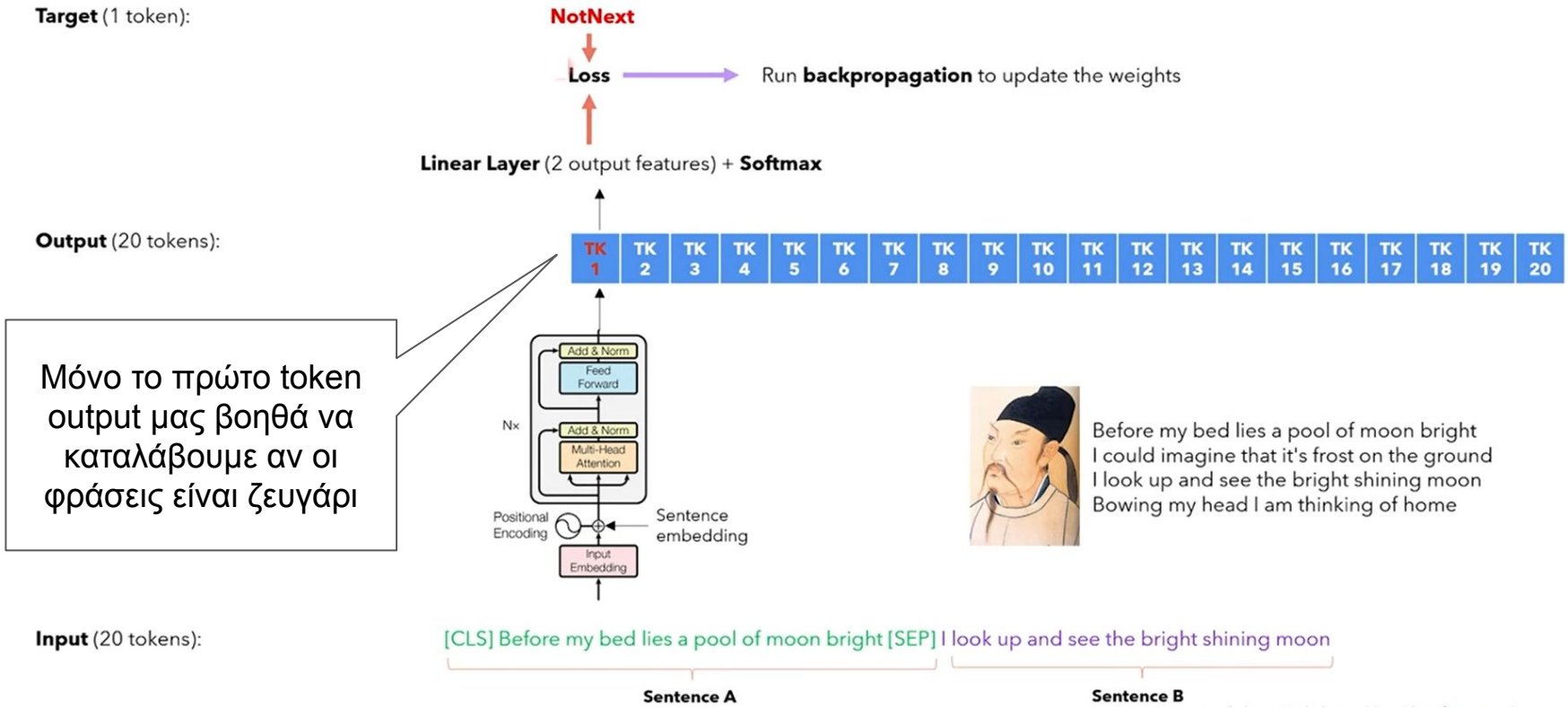
50% actual next sentence, 50% random sentence

Next sentence prediction

Προσθέτουμε τα segment embeddings που λένε σε ποια πρόταση ανήκει ο όρος



Training BERT (Next sentence task)



CLS token: self attention

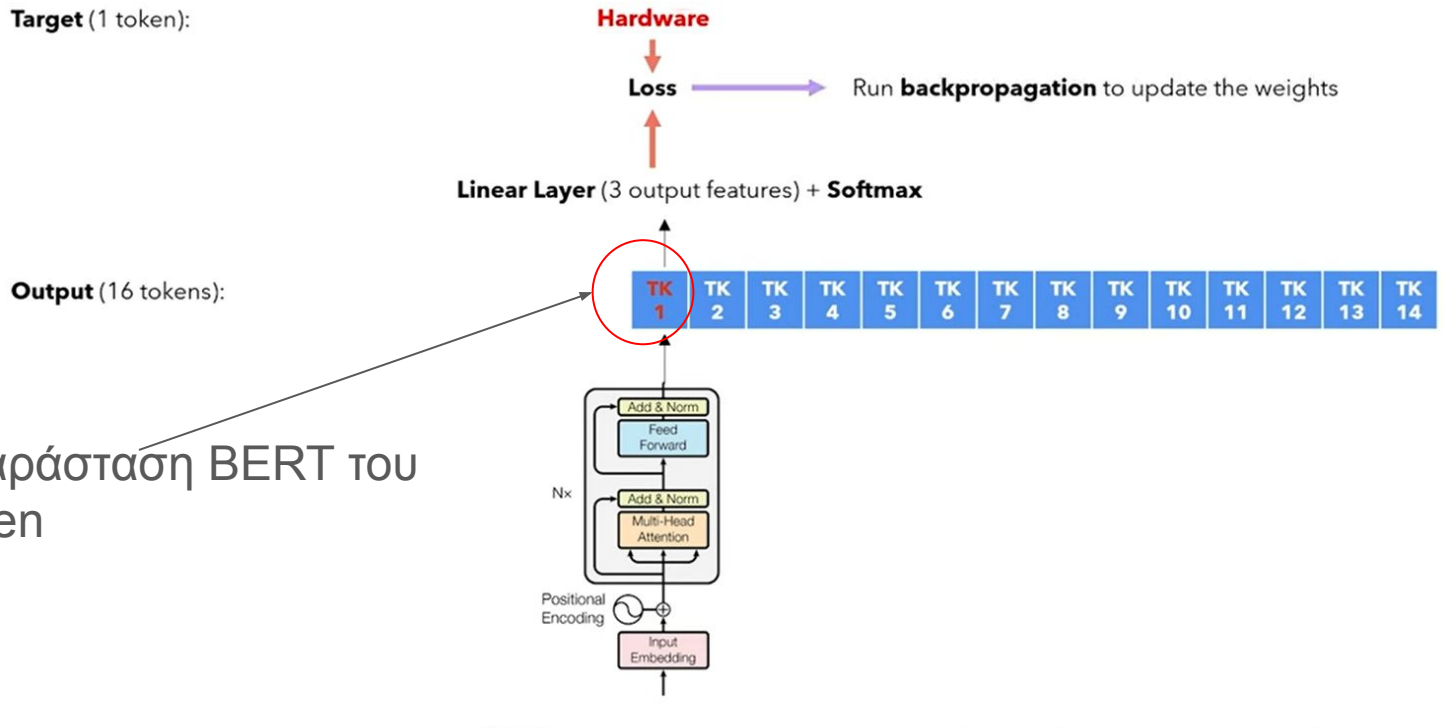
Χωρίς causal masking

Συλλέγει όλη την πληροφορία των tokens της πρώτης πρότασης και δεύτερης πρότασης

Κατηγοριοποιεί (classifies) το συνδυασμό των δύο προτάσεων ως αναμενόμενο

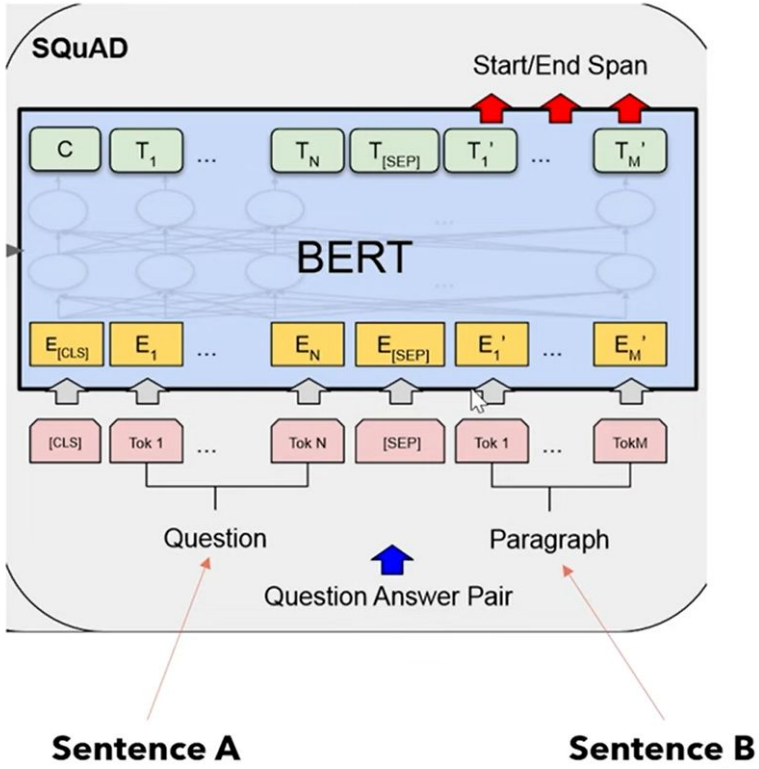
BERT fine-tuning (π.χ. classification)

Text classification - fine-tuning



Η αναπαράσταση BERT του CLS token

BERT question answering



Demo: https://rajpurkar.github.io/SQuAD-explorer/

Classify tokens as start and end positions

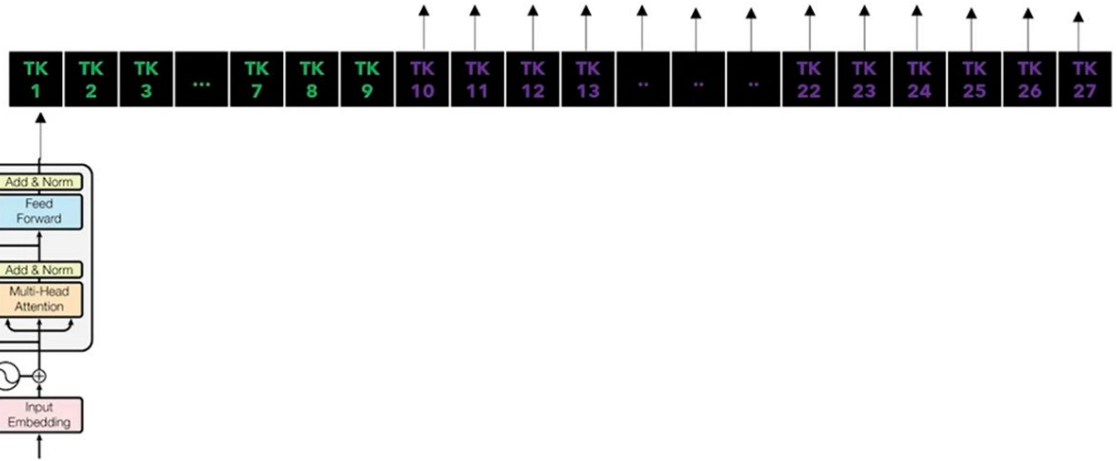
Target (1 token):

start=TK10, end=TK10

Loss → Run **backpropagation** to update the weights

Linear Layer (2 output features) + **Softmax**

Output (27 tokens):



Input (27 tokens):

[CLS] What is the fashion capital of China? [SEP] Shanghai is a City in China, it is also a financial center, its fashion capital and industrial city.

Hugging Face Transformers

https://huggingface.co/models

Μετάφραση: Helsinki-NLP/opus-mt-en-el, Helsinki-NLP/opus-mt-grk-en, lighteternal/SSE-TUC-mt-el-en-cased

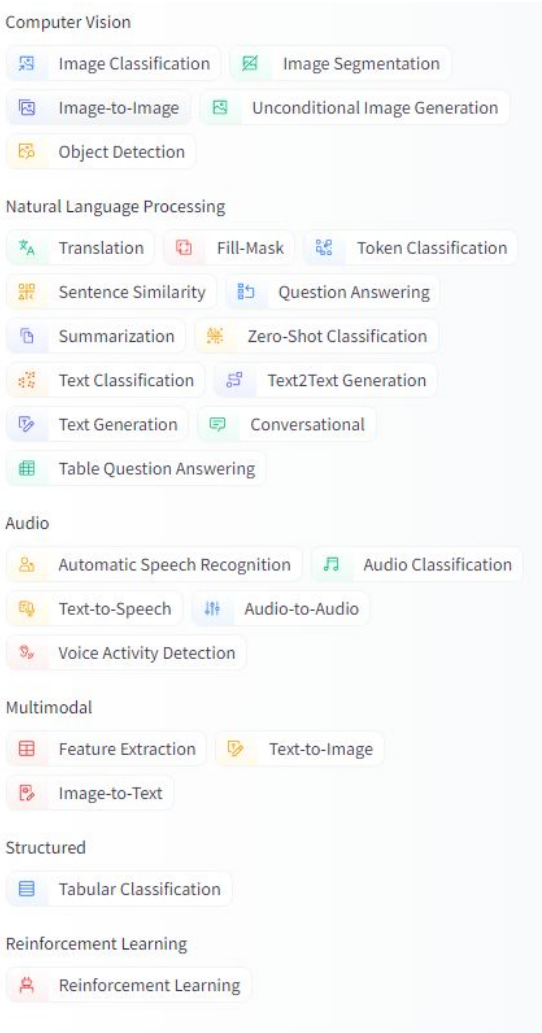
Επώνυμες οντότητες: spacy/el_core_news_md

Θεματικές κατηγορίες: cvcio/mediawatch-el-topics

Γνώμη/Fact: lighteternal/fact-or-opinion-xlmr-el

Παραγωγή κειμένου: sberbank-ai/mGPT, nikokons/gpt2-greek

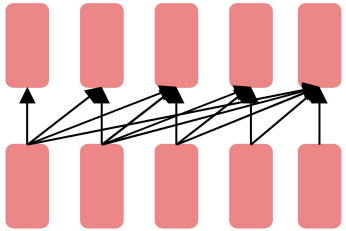
Διάλογοι στα Αγγλικά: facebook/blenderbot_small-90M



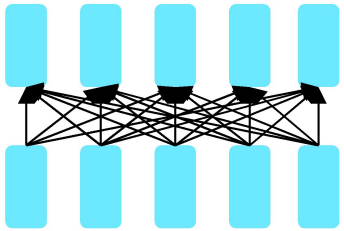
Περιεχόμενα

- Transformers, Attention
- Encoders/decoders
- Pretrained transformers (BERT) και fine tuning
- LLM overview
- Retrieval Augmented Generation και Vector databases

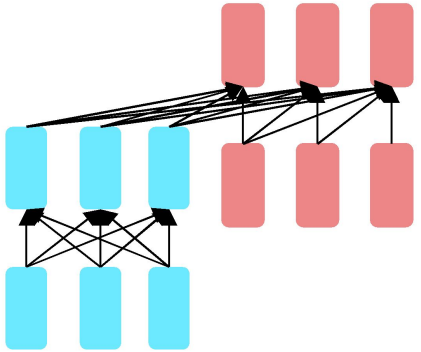
Τρεις βασικές αρχιτεκτονικές



Decoders
GPT, Claude,
Llama
Mixtral

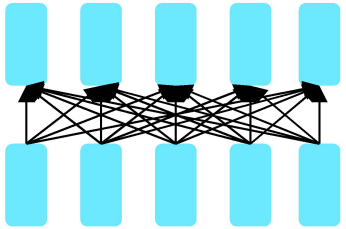


Encoders
BERT family,
HuBERT



Encoder-decoders
Flan-T5, Whisper

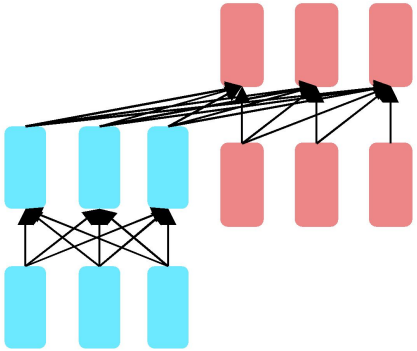
Encoders



- Αρκετά δημοφιλείς: Masked Language Models (MLMs)
- BERT family

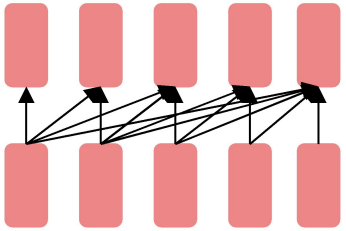
- Εκπαιδεύονται να βρίσκουν τις λέξεις που λείπουν (masked) με βάση τις λέξεις τριγύρω
- Συνήθως εκπαιδεύονται (finetuned) για classification tasks

Encoder-Decoders



- Εκπαιδεύονται να απεικονίζουν μια ακολουθία σε μια άλλη
- Πολύ δημοφιλή για:
  - machine translation (map from one language to another)
  - speech recognition (map from acoustics to words)

Decoder-only models



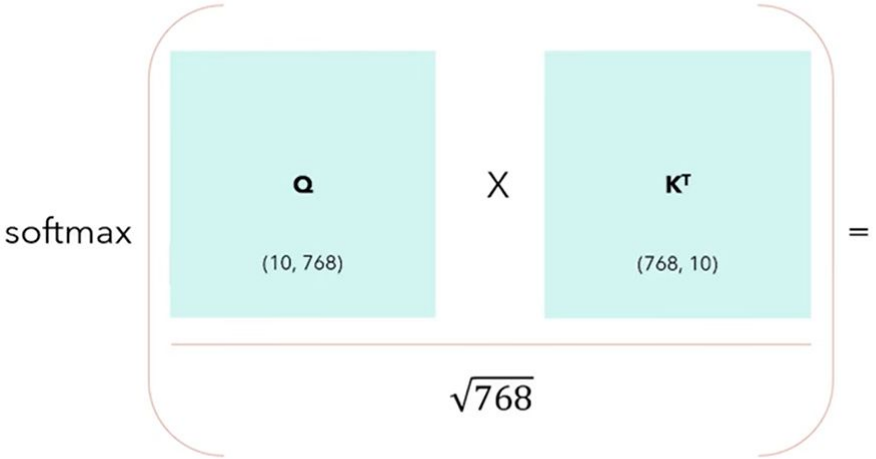
Γνωστά και ως

- Causal LLMs
- Autoregressive LLMs
- Left-to-right LLMs

- Προβλέπουν λέξεις από αριστερά προς τα δεξιά

Self-Attention κάθε token συγκρίνεται με κάθε token

$$Attention(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$



=

	[SOS]	Before	my	bed	lies	a	pool	of	moon	bright
[SOS]	0.62	0.19	0.02	0.02	0.04	0.01	0.00	0.09	0.00	0.02
Before	0.15	0.00	0.00	0.01	0.00	0.00	0.17	0.00	0.67	0.00
my	0.09	0.02	0.56	0.02	0.01	0.08	0.11	0.02	0.05	0.03
bed	0.10	0.06	0.03	0.00	0.53	0.12	0.01	0.11	0.00	0.04
lies	0.02	0.00	0.00	0.05	0.80	0.00	0.02	0.04	0.01	0.06
a	0.01	0.00	0.02	0.02	0.00	0.03	0.68	0.16	0.03	0.06
pool	0.00	0.16	0.02	0.00	0.03	0.56	0.00	0.00	0.22	0.01
of	0.22	0.00	0.01	0.05	0.19	0.44	0.00	0.00	0.04	0.04
moon	0.00	0.67	0.01	0.00	0.02	0.03	0.23	0.01	0.00	0.03
bright	0.06	0.00	0.03	0.03	0.43	0.21	0.03	0.06	0.13	0.03

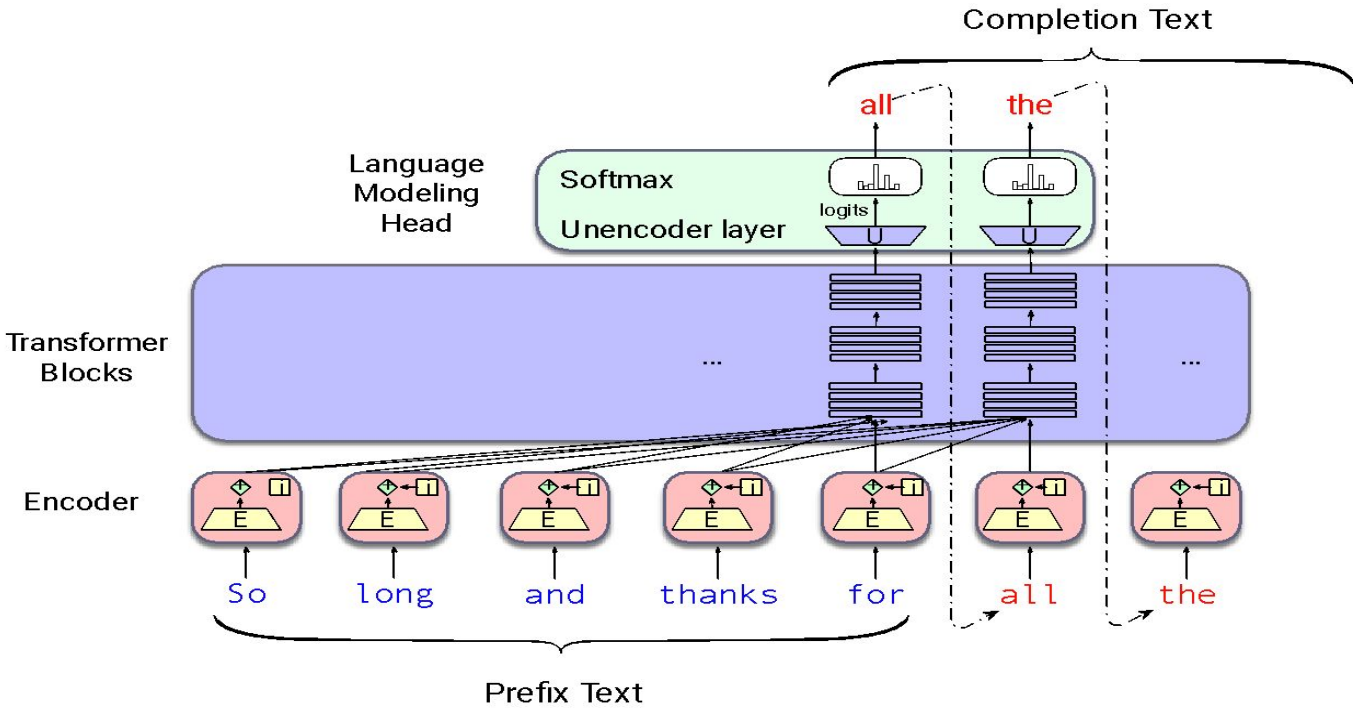
(10, 10)

Causal masking

Πώς διασφαλίζουμε ότι τα tokens στο self-attention επηρεάζονται μόνο από τα προηγούμενα tokens;

	[SOS]	Before	my	bed	lies	a	pool	of	moon	bright
[SOS] .	**5.45**	-∞	-∞	-∞	-∞	-∞	-∞	-∞	-∞	-∞
Before	4.28	**2.46**	-∞	-∞	-∞	-∞	-∞	-∞	-∞	-∞
my	8.17	3.56	**5.54**	-∞	-∞	-∞	-∞	-∞	-∞	-∞
bed	6.71	4.13	6.76	**0.79**	-∞	-∞	-∞	-∞	-∞	-∞
lies	5.43	7.59	3.91	6.14	**9.03**	-∞	-∞	-∞	-∞	-∞
a	4.42	4.35	7.55	3.14	1.35	**7.57**	-∞	-∞	-∞	-∞
pool	8.36	6.00	4.56	0.52	3.13	6.78	**9.00**	-∞	-∞	-∞
of	2.21	3.72	4.16	6.30	0.66	6.14	7.46	**6.77**	-∞	-∞
moon	4.08	6.22	5.00	4.20	5.72	5.35	7.46	3.55	**4.70**	-∞
bright	6.43	8.88	6.17	3.65	4.54	5.22	5.51	5.55	0.64	**1.38**

Παραγωγή υπό προϋποθέσεις (ότι προηγείται)



Χρήσεις

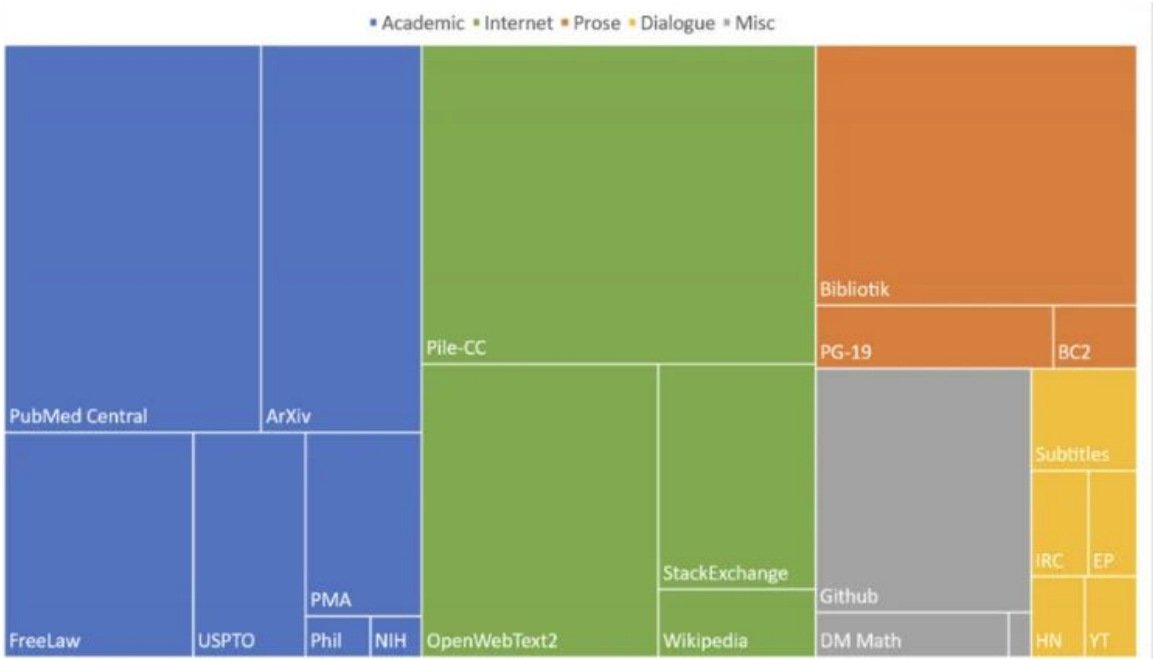
predicted next token(s)

- Sentiment analysis
  - The sentiment of the sentence "I like their customer service" is: ⇒ Positive <EOS>
- Question answering
  - Who wrote the book “Iluminati” : ⇒ Dan Brown <EOS>
- Summarization
  - The only thing crazier than a guy in snowbound Massachusetts boxing up the powdery white stuff and offering it for sale online? People are actually buying it. For $89, self-styled entrepreneur Kyle Waring will ship you 6 pounds of Boston-area snow in an insulated Styrofoam box – enough for 10 to 15 snowballs, he says.
    But not if you live in New England or surrounding states. “We will not ship snow to any states in the northeast!” says Waring’s website, ShipSnowYo.com. “We’re in the business of expunging snow!”
    His website and social media accounts claim to have filled more than 133 orders for snow – more than 30 on Tuesday alone, his busiest day yet. With more than 45 total inches, Boston has set a record this winter for the snowiest month in its history. Most residents see the huge piles of snow choking their yards and sidewalks as a nuisance, but Waring saw an opportunity.
    According to Boston.com, it all started a few weeks ago, when Waring and his wife were shoveling deep snow from their yard in Manchester-by-the-Sea, a coastal suburb north of Boston. He joked about shipping the stuff to friends and family in warmer states, and an idea was born. [...]

    ⇒ Kyle Waring will ship you 6 pounds of Boston-area snow in an insulated Styrofoam box – enough for 10 to 15 snowballs, he says. But not if you live in New England or surrounding states.

Εκπαίδευση - σώματα κειμένου

Σε μεγάλα σώματα κειμένου όπως το common crawl (386Tb), το refined web (2.8Tb), το pile (886Gb) που προέρχονται σε μεγάλο βαθμό από το web.



predict the next word

Perplexity

Μετρά πόσο καλά ένα γλωσσικό μοντέλο προβλέπει την επόμενη λέξη σε μια ακολουθία. Χαμηλότερες τιμές υποδηλώνουν ότι το μοντέλο είναι πιο σίγουρο και ακριβές στις προβλέψεις του, ενώ υψηλότερες τιμές υποδηλώνουν μεγαλύτερη αβεβαιότητα και χειρότερη απόδοση.

$$\begin{aligned} \text{Perplexity}_\theta(w_{1:n}) &= P_\theta(w_{1:n})^{-\frac{1}{n}} \\ &= \sqrt[n]{\frac{1}{P_\theta(w_{1:n})}} \end{aligned}$$

perplexity range is [1,∞]

Περιεχόμενα

- Transformers, Attention
- Encoders/decoders
- Pretrained transformers (BERT) και fine tuning
- LLM overview
- Retrieval Augmented Generation και Vector databases

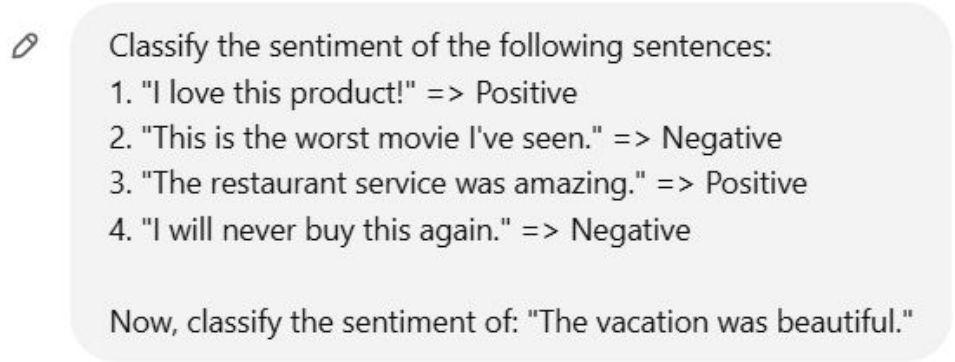
Χαρακτηριστικά των LLMs στην παραγωγή κειμένου

+ Μπορούν να παράξουν απαντήσεις σε μια ερώτηση (prompt) χρησιμοποιώντας το σύνολο του κειμένου της ερώτησης (attention)
+ Έχουν εκπαιδευτεί σε ένα μεγάλο σώμα κειμένων οπότε κάθε απάντηση που παράγουν είναι καλογραμμένη

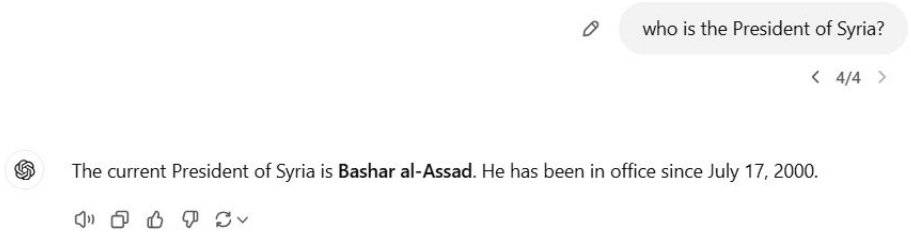
- Εκπαιδεύονται μια φορά σε ένα αρχικό dataset και μπορεί να μην έχουν σύγχρονη γνώση
- Παράγουν ψευδή αλλά αληθοφανή γνώση (hallucinations)
- Χρειάζονται fine-tuning για να προσαρμοστούν σε κάποιο task, αλλά τότε μπορεί να ξεχάσουν ότι γνώριζαν

Few shot prompting

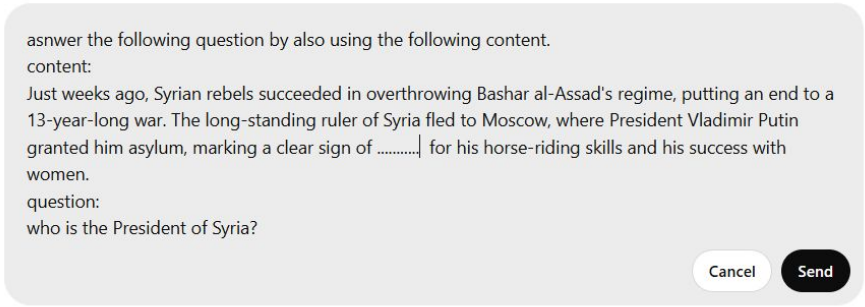
- Στο ίδιο μας το prompt προς το LLM δίνουμε παραδείγματα για τις απαντήσεις που περιμένουμε



QA με χρήση LLM

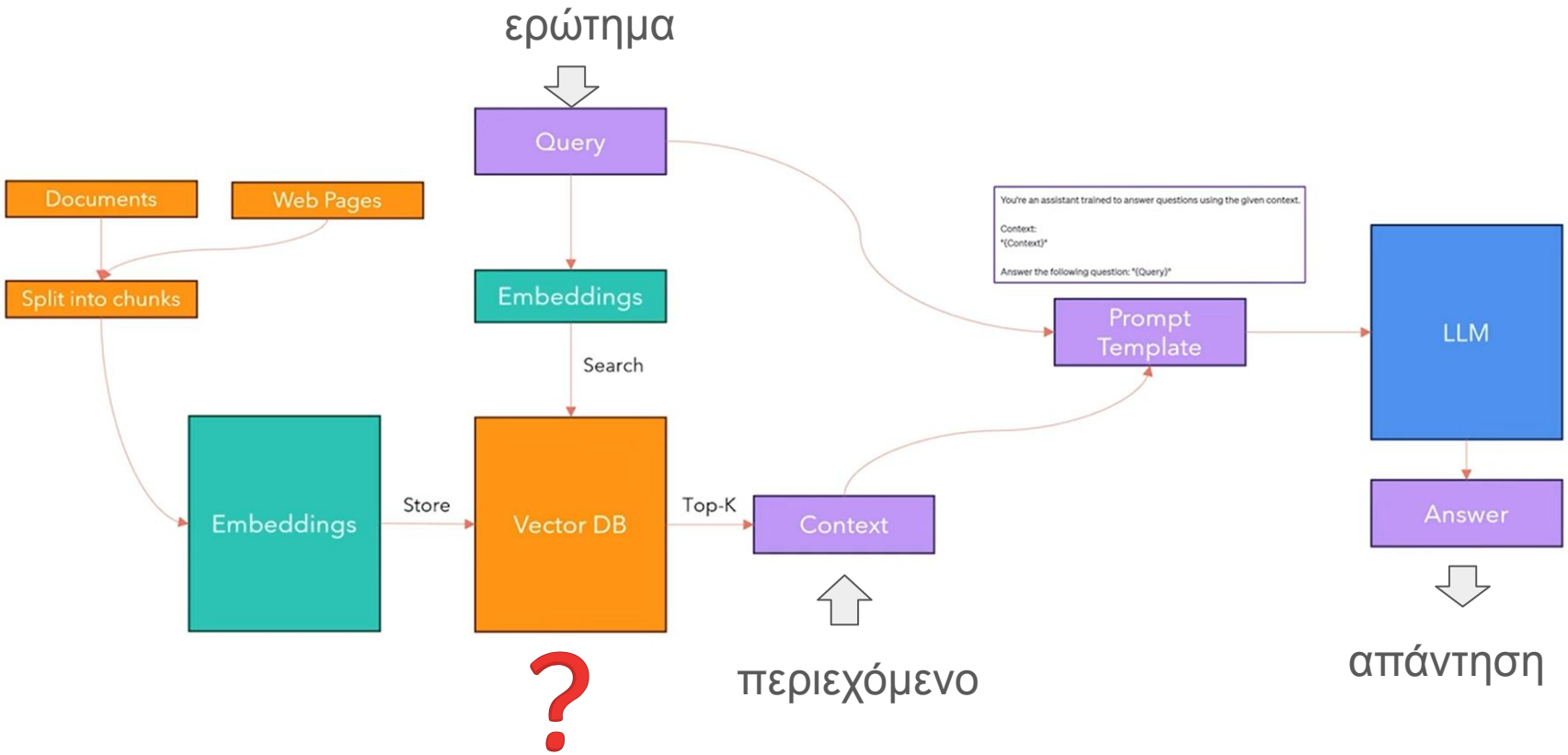


Χωρίς context

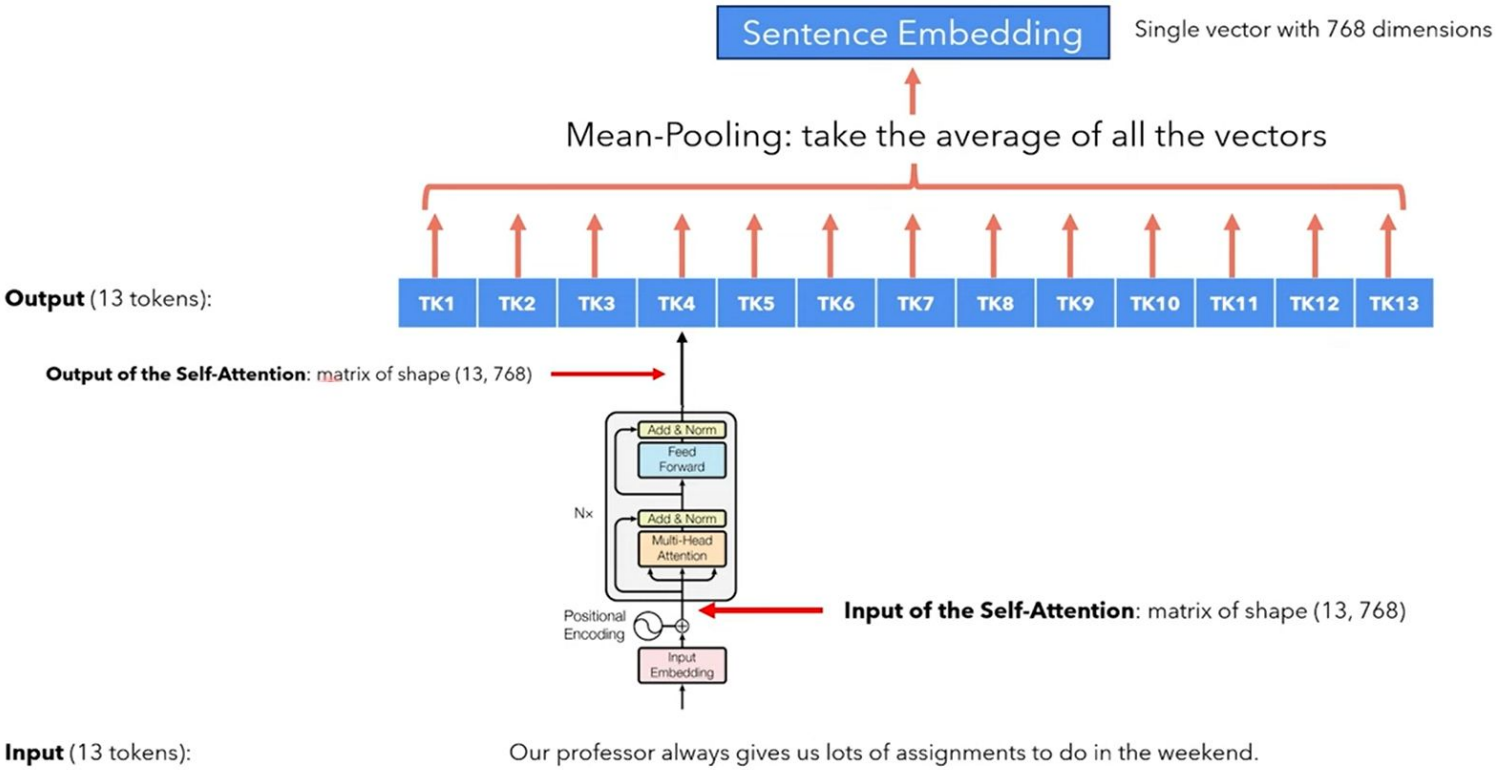


με context

QA with Retrieval Augmented Generation

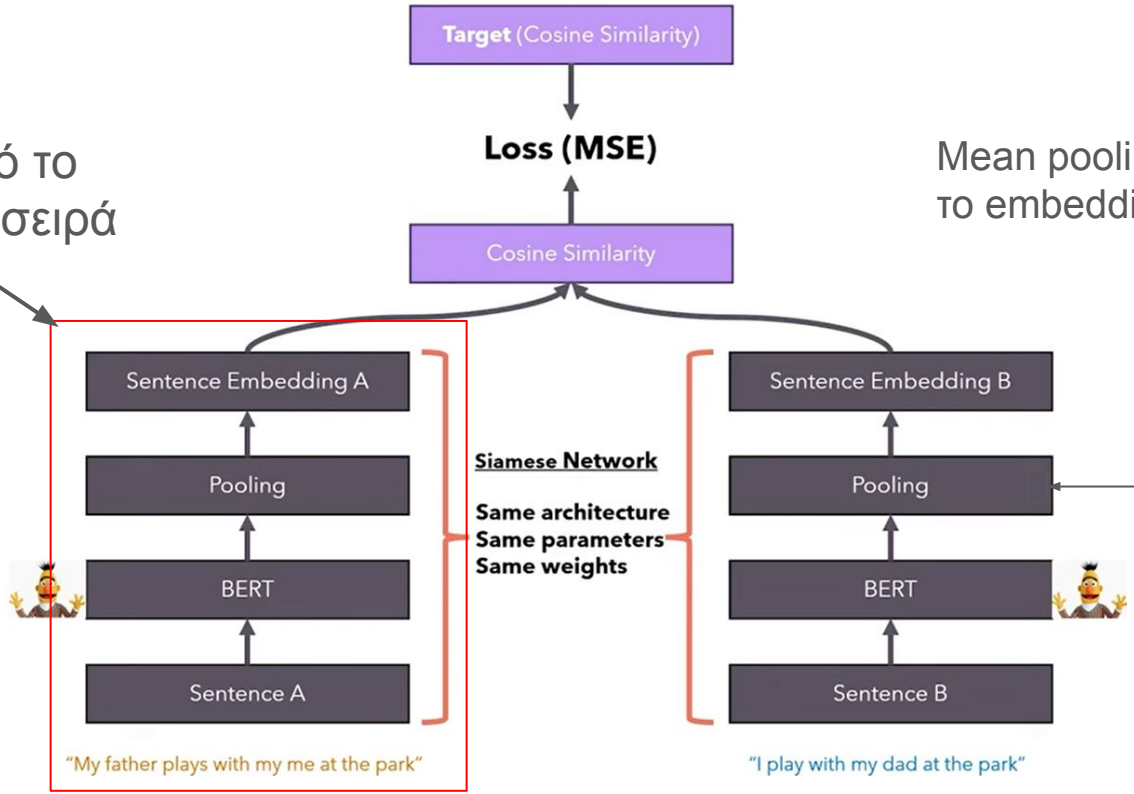


Sentence embeddings with BERT



Sentence BERT

Είναι ένα δίκτυο από το οποίο περνάμε στη σειρά τις δύο προτάσεις



Mean pooling, max pooling ή το embedding του CLS token

Vector DBs

- Ένας τύπος βάσης δεδομένων που αποθηκεύει και διαχειρίζεται διανύσματα (vectors) για την υποστήριξη αναζητήσεων και αναλύσεων σε δεδομένα υψηλής διάστασης.
- Ειδικά σχεδιασμένη για εφαρμογές μηχανικής μάθησης και τεχνητής νοημοσύνης, όπου η αναζήτηση παρόμοιων στοιχείων είναι κρίσιμη.
- Επιτρέπει γρήγορη αναζήτηση και ανάκτηση παρόμοιων στοιχείων μέσω αλγορίθμων όπως k-NN (k-nearest neighbors).
- Εφαρμογές
    - Επεξεργασία Φυσικής Γλώσσας: Χρήση σε μοντέλα γλώσσας για την αναγνώριση και κατηγοριοποίηση κειμένων.
    - Συστάσεις Προϊόντων: Χρησιμοποιείται σε συστήματα συστάσεων για την πρόταση παρόμοιων προϊόντων στους χρήστες.
    - Εικόνες και Πολυμέσα: Ανάλυση και αναγνώριση εικόνων μέσω χαρακτηριστικών που αποθηκεύονται ως διανύσματα.

Αναζήτηση για kNN

- Έχει τη μέγιστη ακρίβεια (precision)
- Απαιτεί σύγκριση με όλα τα δείγματα στη βάση (δλδ. O(N*D) για D διαστάσεις) και υπολογισμό π.χ. του cosine similarity

- Συνήθως μας ενδιαφέρει να έχουμε μεγάλο recall (να εντοπίσουμε δλδ όλα τα σχετικά δείγματα), καθώς μετά μπορούμε να υπολογίσουμε το cosine similarity με αυτά και να απορρίψουμε τα false positives.

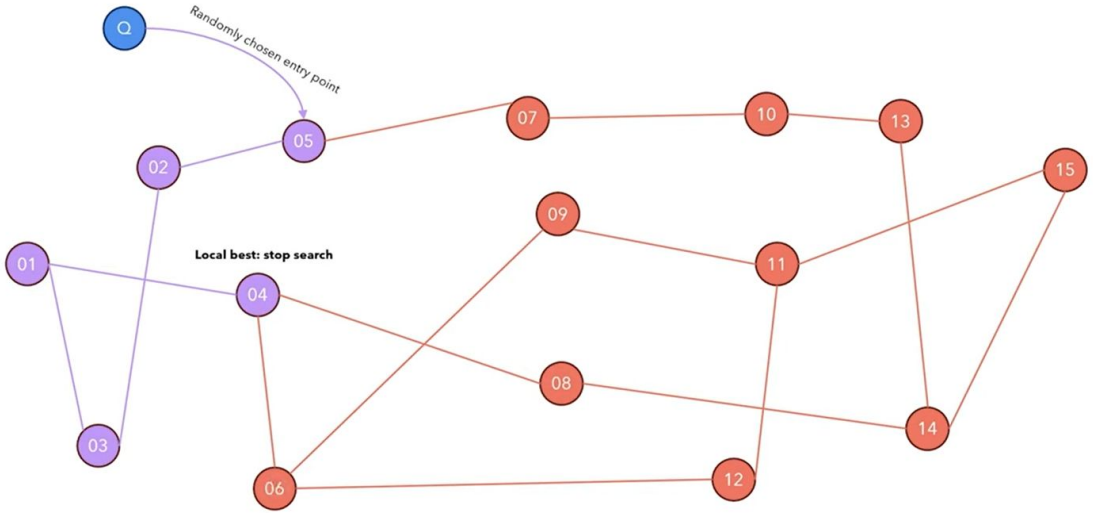
Hierarchical Navigable Small World (HNSW) Graphs

- Είναι μια μέθοδος αναζήτησης που χρησιμοποιεί ιεραρχικά διαχειρίσιμα μικρά υπο-γραφήματα για την αποτελεσματική εύρεση κοντινών γειτόνων (nearest neighbors) σε δεδομένα υψηλής διάστασης.
- Δομή Γραφήματος:
  - Δημιουργεί μια ιεραρχική δομή γραφήματος με πολλαπλά επίπεδα, όπου κάθε επίπεδο περιέχει υποσύνολα των αποθηκευμένων στοιχείων.
  - Κάθε κόμβος έχει λίγους γείτονες (topk-most similar σε ένα seed - στη λογική των small worlds)
  - Η επιλογή του επιπέδου γίνεται τυχαία με βάση μια εκθετική κατανομή πιθανότητας.
- Αναζήτηση:
  - Ξεκινά από το ανώτερο επίπεδο και χρησιμοποιεί τη διαχωρισμένη κλίμακα για να ενισχύσει την απόδοση.
  - Χρησιμοποιεί αλγορίθμους όπως k-NN για γρήγορη ανάκτηση αποτελεσμάτων.

Navigable Small World (HNSW) Graphs

Ξεκινάμε με ένα τυχαίο sentence και κινούμαστε μόνο στους γείτονές του μέχρι να βρούμε τον πιο κοντινό γείτονα. Επαναλαμβάνουμε με άλλο τυχαίο κόμβο

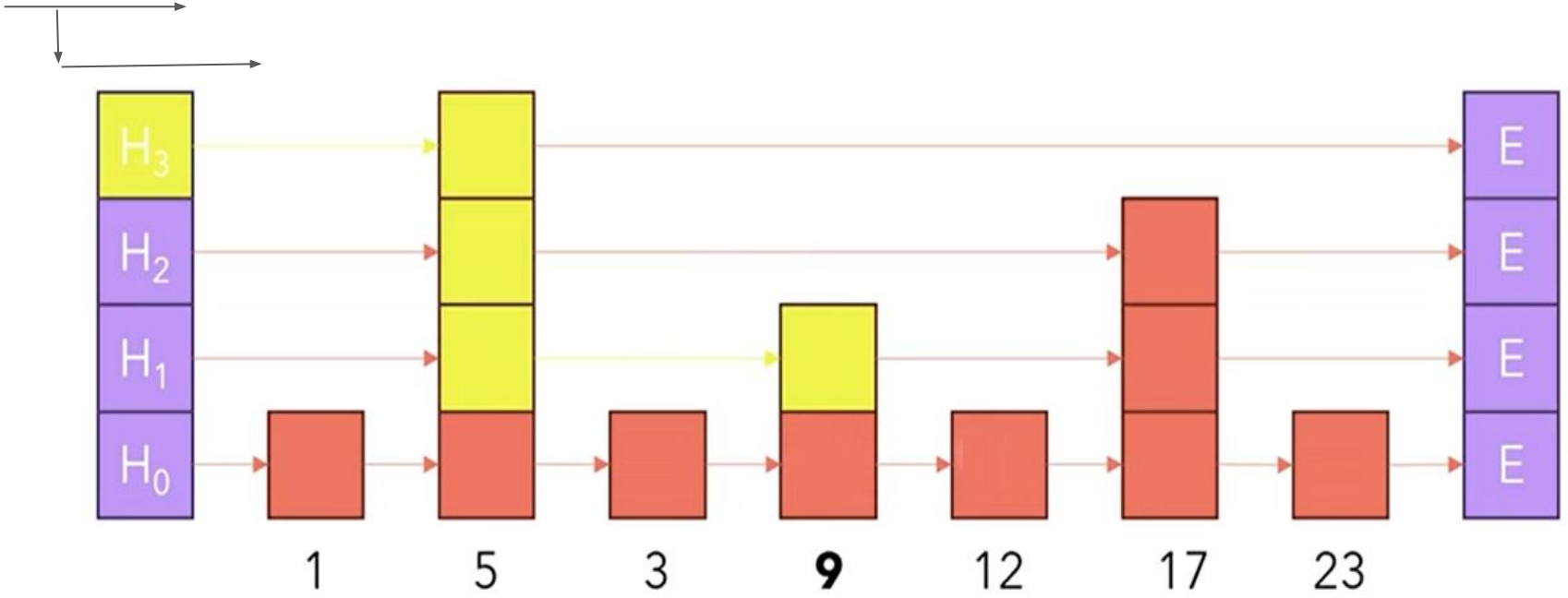
Node	Text
01	[…] The Transformer is a model […]
02	[…] Diagnose cancer with AI […]
03	[…] A transformer-based model […]
04	[…] The Transformer has 6 layers […]
05	[…] An MRI machine that costs 1$ […]
06	[…] The dot-product is a […]
07	[…] Big-Pharma is not so big […]
08	[…] Cross-Attention is a great […]
09	[…] To solve an ODE […]
10	[…] We are aging too fast […]
11	[…] Open-source models like […]
12	[…] MathBERT: a new model […]
13	[…] AI to control aging […]
14	[…] Attention is all you need […]
15	[…] LLaMA 2 has 7B params […]



https://www.youtube.com/watch?v=77QH0Y2PYKg&ab_channel=DataMListic

Skip Linked List

‘Εστω ψάχνουμε το 9. Ξεκινάμε από το 1ο επίπεδο και κατεβαίνουμε ώστε να μην προσπεράσουμε αυτό που ψάχνουμε



Hierarchical Navigable Small World (HNSW) Graphs

Σε κάθε επίπεδο είναι και πιο sparse ο γράφος. Ξεκινάμε από το ανώτερο επίπεδο. Βρίσκουμε τον πλησιέστερο κόμβο (με τη λογική του NSW) και κατεβαίνουμε στο επόμενο επίπεδο στον ίδιο κόμβο.