Παραγωγική Τεχνητή Νοημοσύνη: Generative AI

Κωνσταντίνος Καραμανής

The University of Texas at Austin & Archimedes/Athena RC

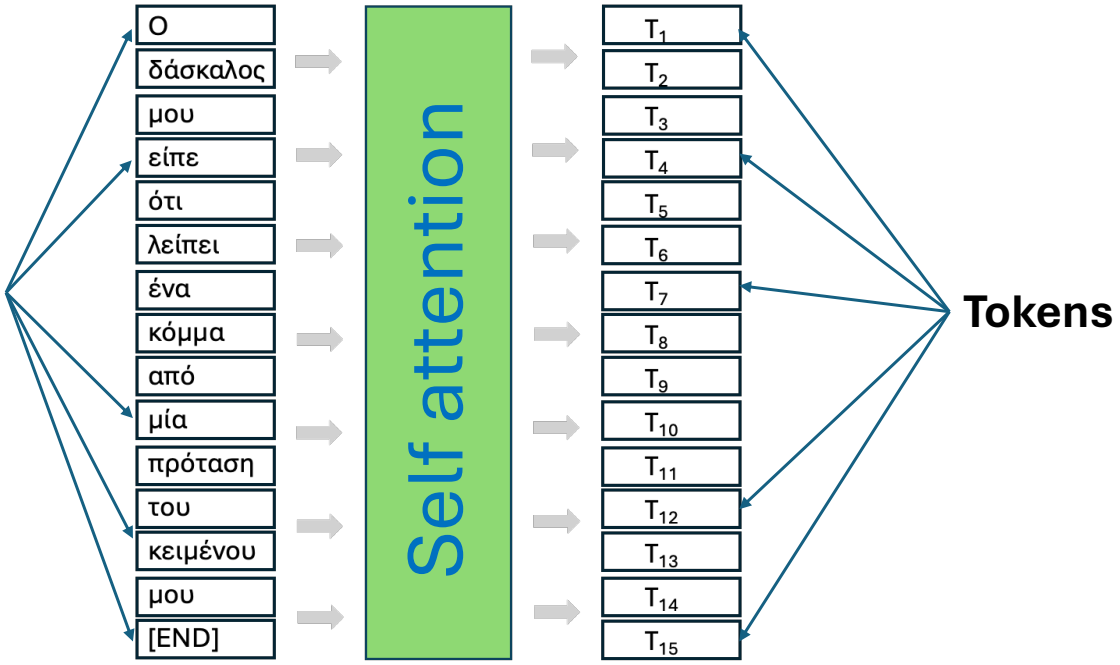
constantine@utexas.edu

https://caramanis.github.io/

Ας θυμηθούμε τα προηγούμενα...

Το «Transformer» Layer και Attention

“[Ο] [δάσκαλος] [είπε] [ότι] [λείπει] [ένα] [κόμμα] [από] [μια] [πρόταση] [του] [κειμένου] [μου].”



Self-supervision + σώμα κειμένων

Masked Language Modeling:

1. 15%: επιλέγουμε τυχαία 15% των «tokens»
2. 80%: αντικαθιστούμε 80% με [MASK]
3. 10%: αντικαθιστούμε 10% με τυχαία επιλεγμένη λέξη
4. 10%: τα υπόλοιπα 10% των tokens παραμένουν ίδια

X = κείμενο, Y = σωστά (αρχικά) tokens

Next Sentence Prediction:

1. Επιλέγουμε δύο προτάσεις (S1,S2) από το σώμα κειμένων. Στις μισές περιπτώσεις, η πρόταση S2 ακολουθεί την S1 στο κείμενο

X = (S1, S2), Y = 1 or 0 (ακόλουθη πρόταση)

Self-supervision + σώμα κειμένων

1. 15% των tokens επιλέγονται
2. 80% των επιλεγμένων token αντικαθίστανται με [MASK]
3. 10% των επιλεγμένων token αντικαθίστανται τυχαία
4. 10% των επιλεγμένων token παραμένουν απαράλλαχτα

TASK: Masked Language Modeling (MLM)

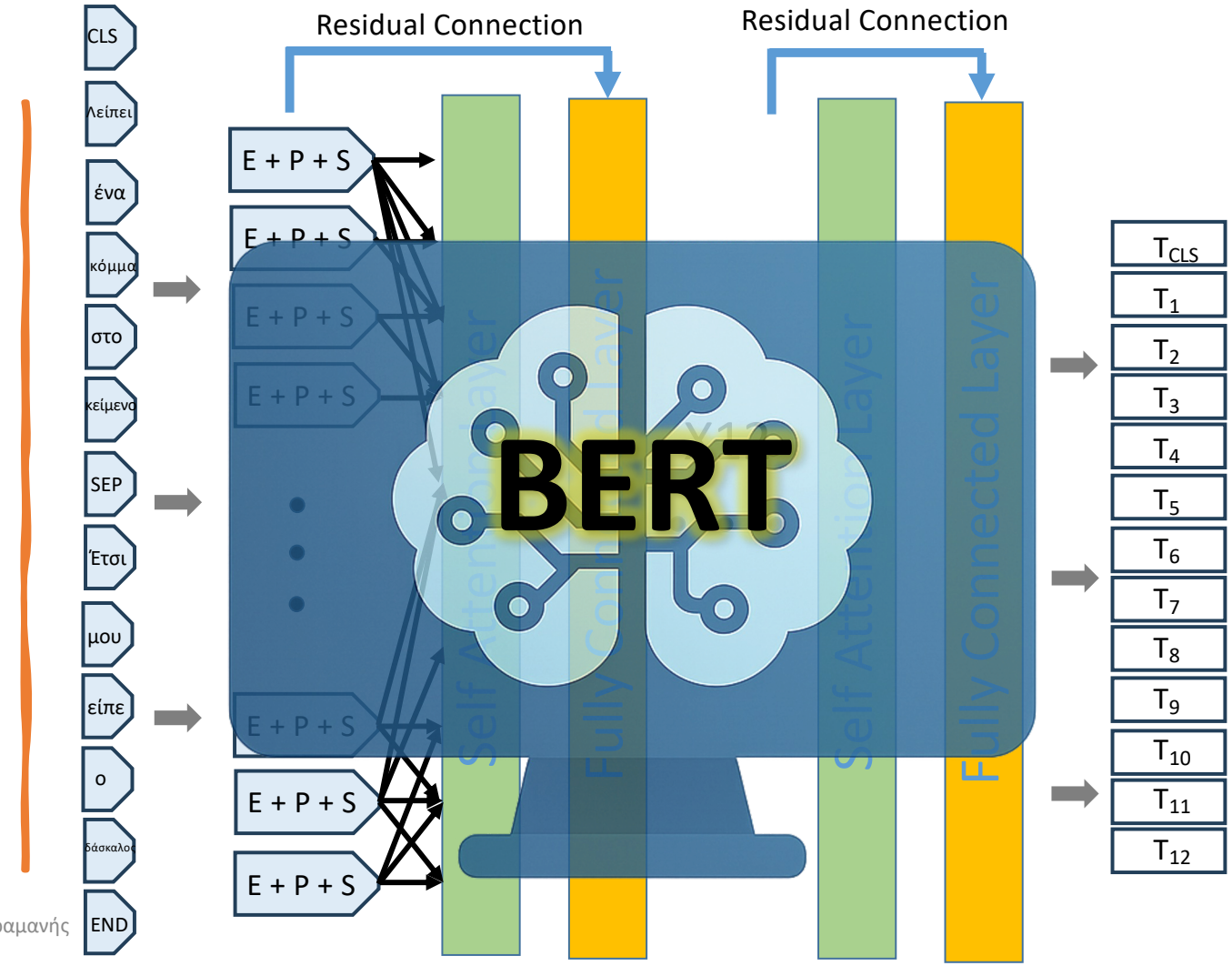
X_1: My current [redacted] interests focus on [redacted] decision-making in large-scale complex systems, with a focus on [redacted] and accordion.

Y_1: {research, autonomous, learning, computation}

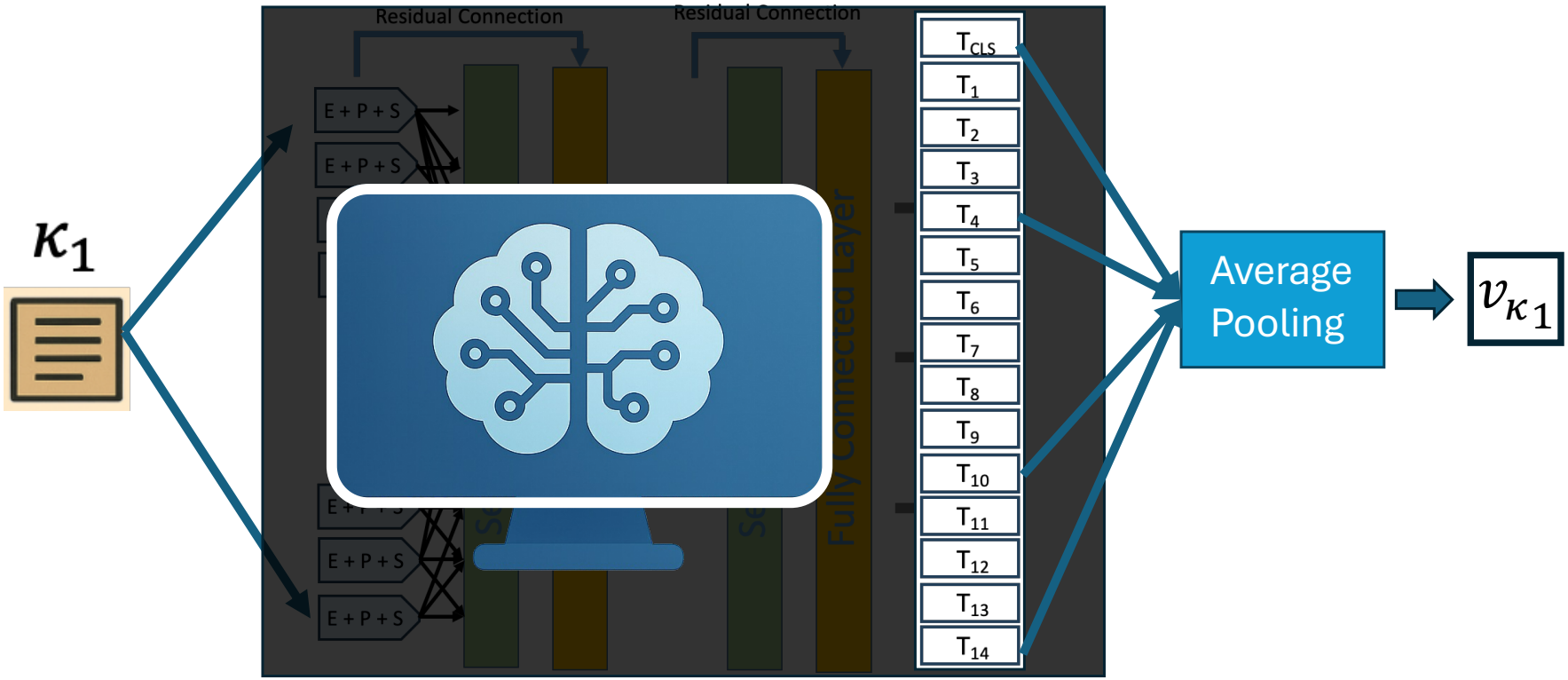
X_2: I am interested in [redacted] and adaptable optimization, high dimensional [redacted] and machine learning, reinforcement learning and [redacted]

Y_2: {robust, statistics, machine, agents}

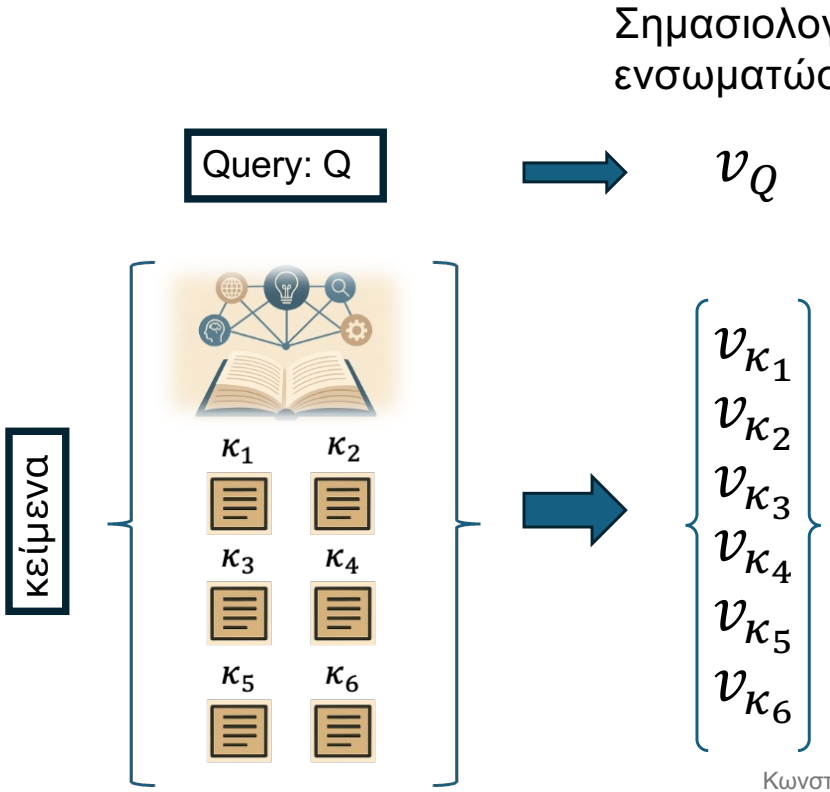
BERT Family: Αρχιτεκτονική του μοντέλου



Ενσωματώσεις με το BERT



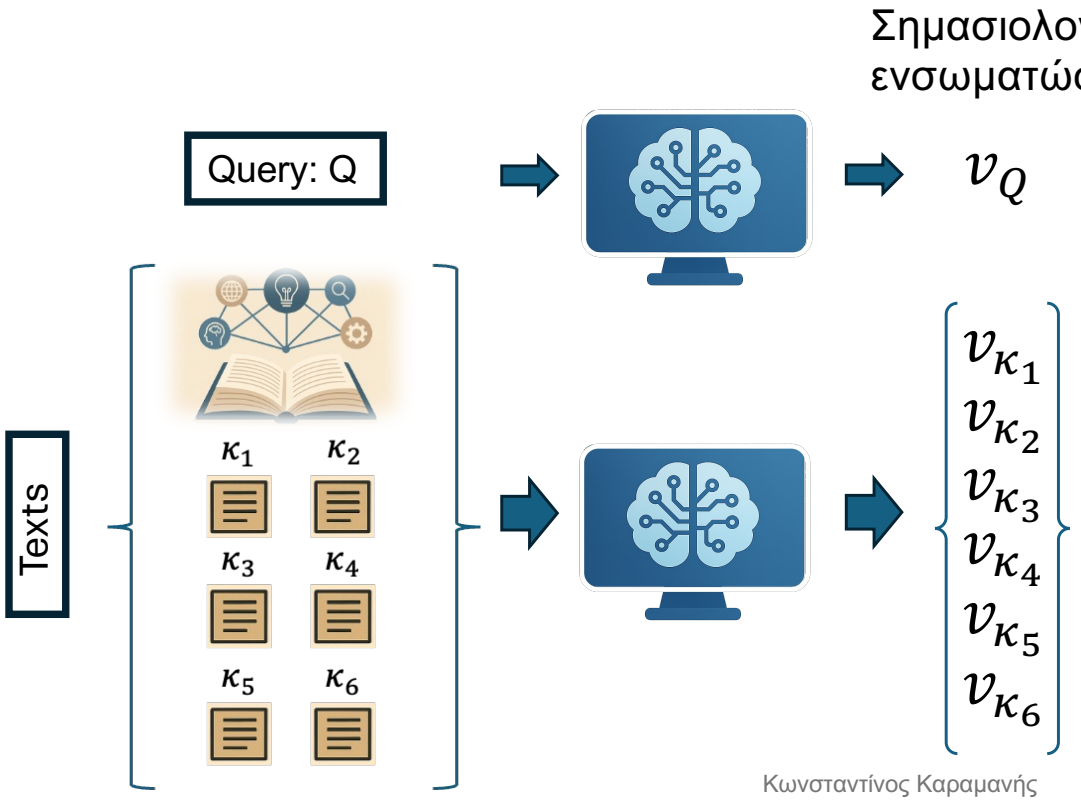
Αναζήτηση σε Βάση Γνώσεων



Κατατάσσουμε τα κείμενα σε φθίνουσα σειρά εσωτερικού γινομένου

$$\langle v_\kappa, v_Q \rangle$$

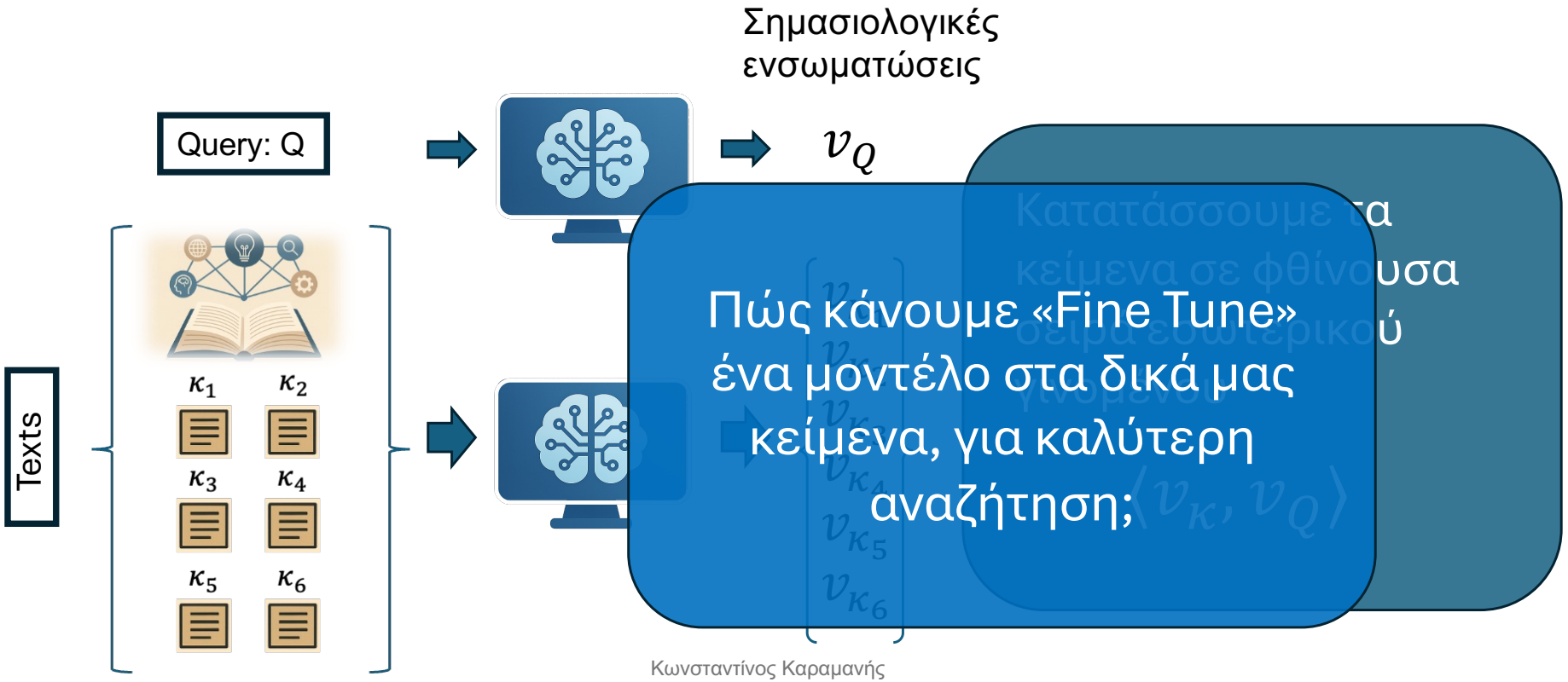
Αναζήτηση σε Βάση Γνώσεων



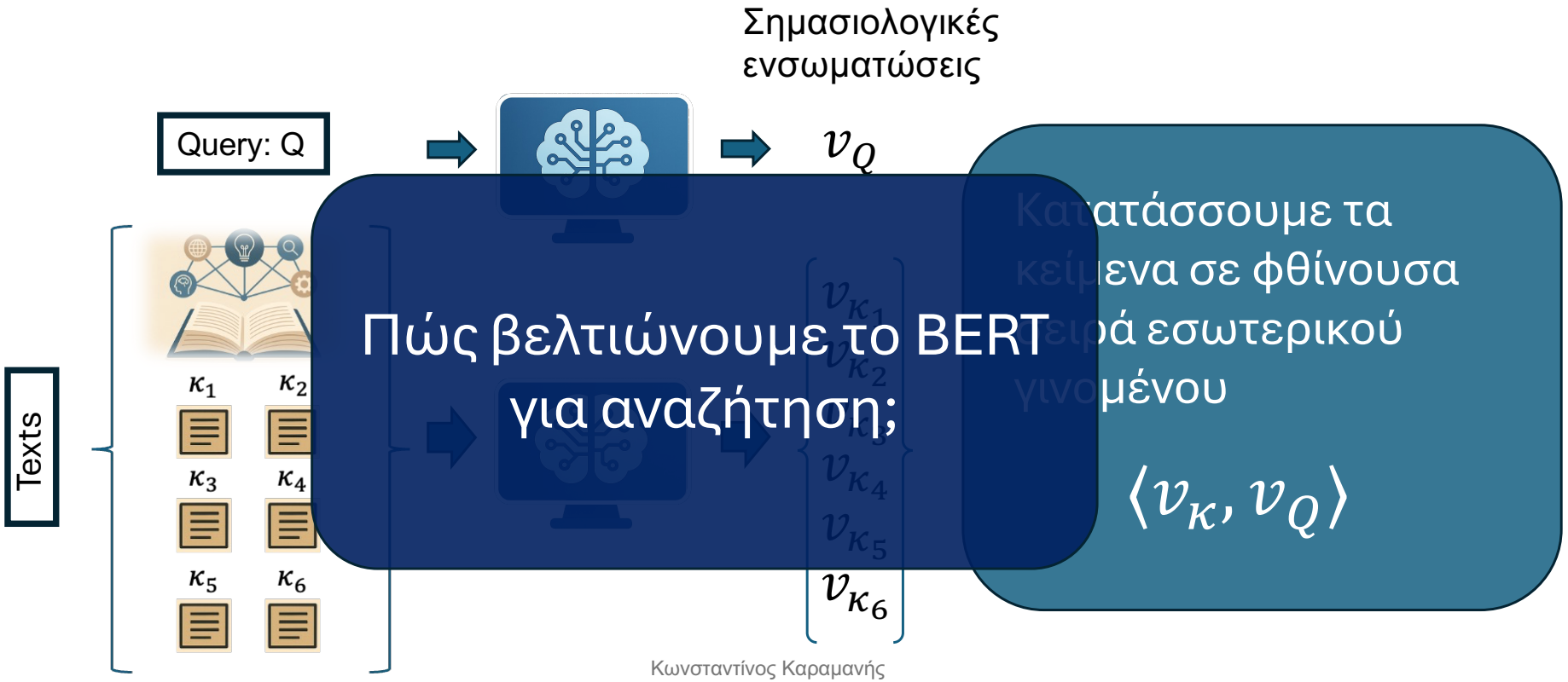
Κατατάσσουμε τα κείμενα σε φθίνουσα σειρά εσωτερικού γινομένου

$$\langle v_\kappa, v_Q \rangle$$

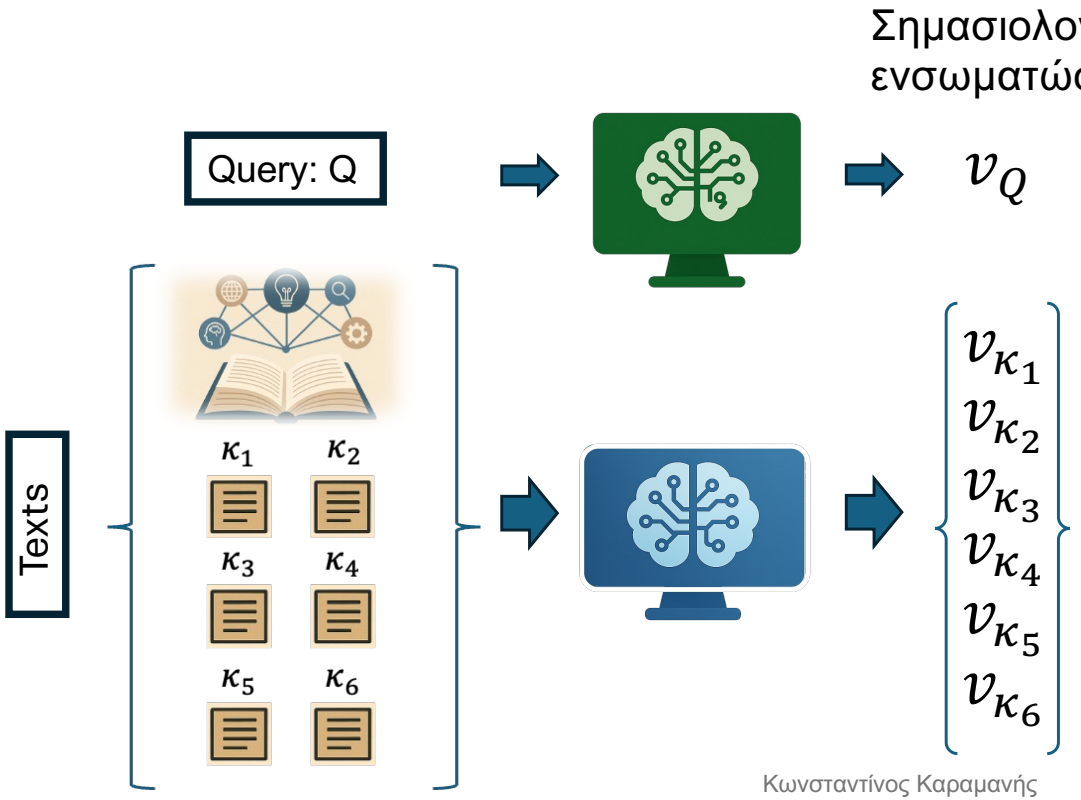
Αναζήτηση σε Βάση Γνώσεων



Αναζήτηση σε Βάση Γνώσεων



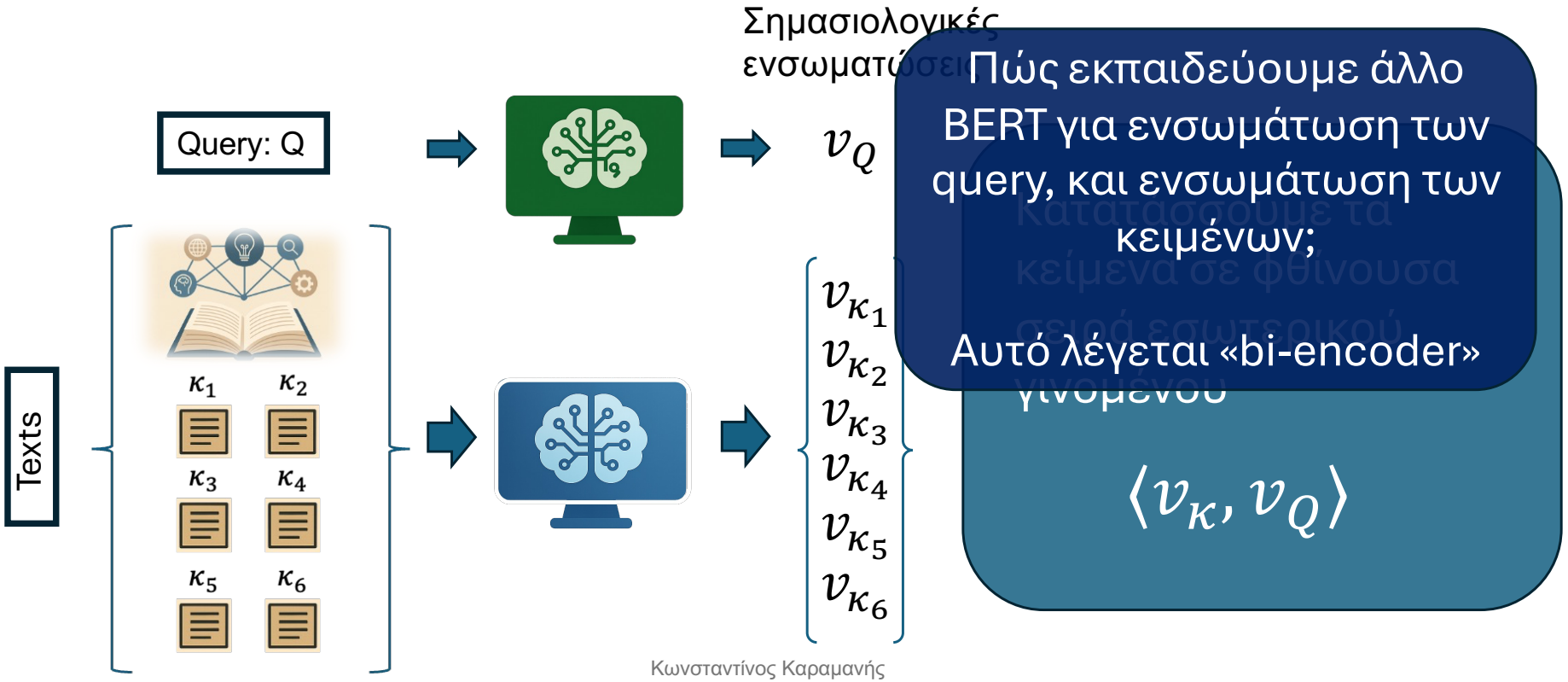
Αναζήτηση σε Βάση Γνώσεων



Κατατάσσουμε τα κείμενα σε φθίνουσα σειρά εσωτερικού γινομένου

$$\langle v_\kappa, v_Q \rangle$$

Αναζήτηση σε Βάση Γνώσεων



Fine Tuning: Μια πρώτη ιδέα

MLM Fine-tuning: το BERT έχει εκπαιδευτεί με την διαδικασία «Masked Language Modeling».

Θα μπορούσαμε να συνεχίσουμε αυτήν την διαδικασία, αλλά στα δικά μας δεδομένα. Όπως θα δούμε με πειράματα στο GPT2 / Llama, το MLM μας επιτρέπει να εξειδικεύσουμε τις ενσωματώσεις για το συγκεκριμένο λεξιλόγιο (και στατιστικές λεξιλογίου) των κειμένων μας.

Fine Tuning: Μια πρώτη ιδέα

MLM Fine-tuning: το BERT έχει εκπαιδευτεί με την διαδικασία «Masked Language Modeling».

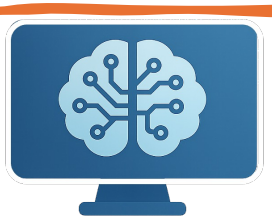
Το MLM είναι εργαλείο εκπαίδευσης που δημιουργεί ένα νευρωνικό δίκτυο με πολλές ικανότητες, όπως έχουμε ήδη δει.

Τώρα όμως θέλουμε να βελτιώσουμε το μοντέλο μας για συγκεκριμένο σκοπό: την σημασιολογική αναζήτηση

λεξιλογίου) των κειμένων μας.

Fine Tuning: Μια ~~πρώτη~~ εξειδικευμένη ιδέα

Αρχίζουμε με ένα MLM-εκπαιδευμένο μοντέλο BERT:

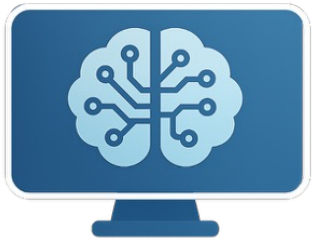


Από το σώμα κειμένων μας υποθέτουμε πως έχουμε δεδομένα εκπαίδευσης (Q_i,K_i): ερωτήματα Q_i και κείμενα K_i που ταιριάζουν (είναι σχετικά) με το ερώτημα

$$\{(Q_1, K_1), \vdots, (Q_N, K_N)\}$$

Περνάμε ερωτήσεις και κείμενα από το BERT

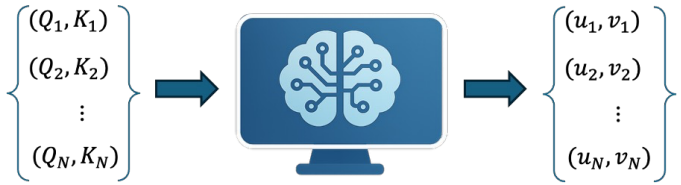
$$\{(Q_1, K_1), (Q_2, K_2), \vdots, (Q_N, K_N)\}$$



$$\{(u_1, v_1), (u_2, v_2), \vdots, (u_N, v_N)\}$$

Τι θα θέλαμε να ικανοποιούν τα δυανίσματα $\{u_i\}$και $\{v_j\}$

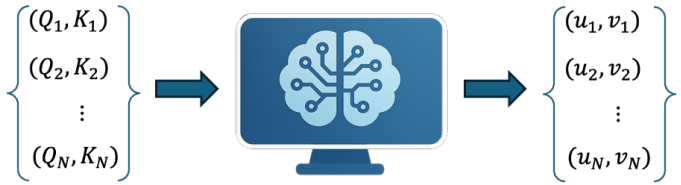
Fine Tuning: Μια ~~πρώτη~~ εξειδικευμένη ιδέα



Ο ιδανικός πίνακας πώς μοιάζει;

	v_1	v_2	v_3	$\cdots$	v_N
u_1	$u_1^\top v_1$	$u_1^\top v_2$	$u_1^\top v_3$	$\cdots$	$u_1^\top v_N$
u_2	$u_2^\top v_1$	$u_2^\top v_2$	$u_2^\top v_3$	$\cdots$	$u_2^\top v_N$
u_3	$u_3^\top v_1$	$u_3^\top v_2$	$u_3^\top v_3$	$\cdots$	$u_3^\top v_N$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
u_N	$u_N^\top v_1$	$u_N^\top v_2$	$u_N^\top v_3$	$\cdots$	$u_N^\top v_N$

Fine Tuning: Μια ~~πρώτη~~ εξειδικευμένη ιδέα

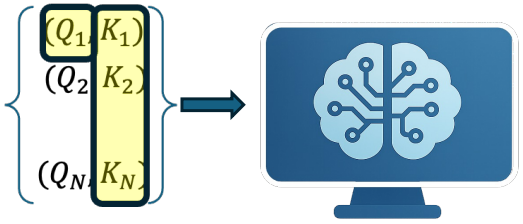


Ο ιδανικός πίνακας πώς μοιάζει;

	v_1	v_2	v_3	$\cdots$	v_N
u_1	$u_1^\top v_1$	$u_1^\top v_2$	$u_1^\top v_3$	$\cdots$	$u_1^\top v_N$
u_2	$u_2^\top v_1$	$u_2^\top v_2$	$u_2^\top v_3$	$\cdots$	$u_2^\top v_N$
u_3	$u_3^\top v_1$	$u_3^\top v_2$	$u_3^\top v_3$	$\cdots$	$u_3^\top v_N$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
u_N	$u_N^\top v_1$	$u_N^\top v_2$	$u_N^\top v_3$	$\cdots$	$u_N^\top v_N$

Μεγάλες τιμές

InfoNCE και Contrastive Loss



$$u_1, \{v_1, v_2, \dots, v_N\}$$

Data Loader

$$\begin{Bmatrix} (Q_1, K_1) \\ (Q_2, K_2) \\ \vdots \\ (Q_N, K_N) \end{Bmatrix}$$

$$p_1 = \frac{e^{u_1^\top v_1}}{\sum_j e^{u_1^\top v_j}}$$

$$p_2 = \frac{e^{u_1^\top v_2}}{\sum_j e^{u_1^\top v_j}}$$

$$\vdots$$

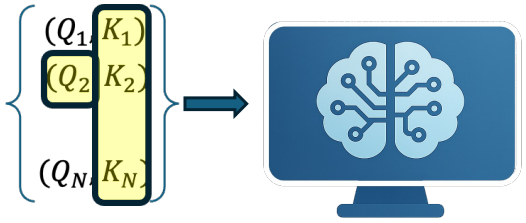
$$p_N = \frac{e^{u_1^\top v_N}}{\sum_j e^{u_1^\top v_j}}$$

Cross Entropy

$$\begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_N \end{pmatrix} \quad \text{vs} \quad \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

Μεγάλη απώλεια εάν δεν έχουμε μεγάλο p_1 (και μικρά τα υπόλοιπα)

InfoNCE και Contrastive Loss



$$u_2, \{v_1, v_2, \dots, v_N\}$$

Data Loader

$$\begin{Bmatrix} (Q_1, K_1) \\ (Q_2, K_2) \\ \vdots \\ (Q_N, K_N) \end{Bmatrix}$$

$$p_1 = \frac{e^{u_2^\top v_1}}{\sum_j e^{u_2^\top v_j}}$$

$$p_2 = \frac{e^{u_2^\top v_2}}{\sum_j e^{u_2^\top v_j}}$$

$$\vdots$$

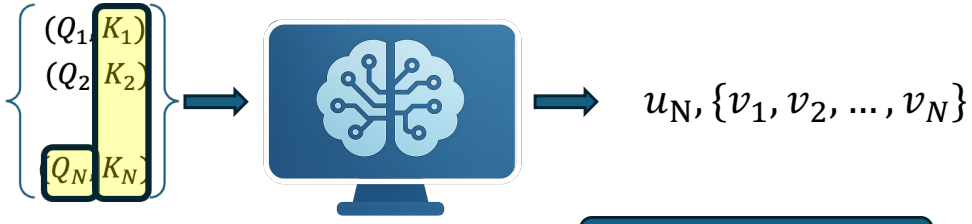
$$p_N = \frac{e^{u_2^\top v_N}}{\sum_j e^{u_2^\top v_j}}$$

Cross Entropy

$$\begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_N \end{pmatrix} \quad \text{vs} \quad \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{pmatrix}$$

Μεγάλη απώλεια εάν δεν έχουμε μεγάλο p_2 (και μικρά τα υπόλοιπα)

InfoNCE και Contrastive Loss



Data Loader → $\begin{Bmatrix} (Q_1, K_1) \\ (Q_2, K_2) \\ \vdots \\ (Q_N, K_N) \end{Bmatrix}$

$$p_1 = \frac{e^{u_N^T v_1}}{\sum_j e^{u_N^T v_j}}$$

$$p_2 = \frac{e^{u_N^T v_2}}{\sum_j e^{u_N^T v_j}}$$

$$\vdots$$

$$p_N = \frac{e^{u_N^T v_N}}{\sum_j e^{u_N^T v_j}}$$

Cross Entropy

$$\begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_N \end{pmatrix} \quad \text{vs} \quad \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix}$$

Μεγάλη απώλεια εάν δεν έχουμε μεγάλο p_N (και μικρά τα υπόλοιπα)

InfoNCE και Contrastive Loss

Step 1: $\{u_i, v_i\}$

Step 2: $S_{ij} = \langle u_i, v_j \rangle, \quad i, j \in B$

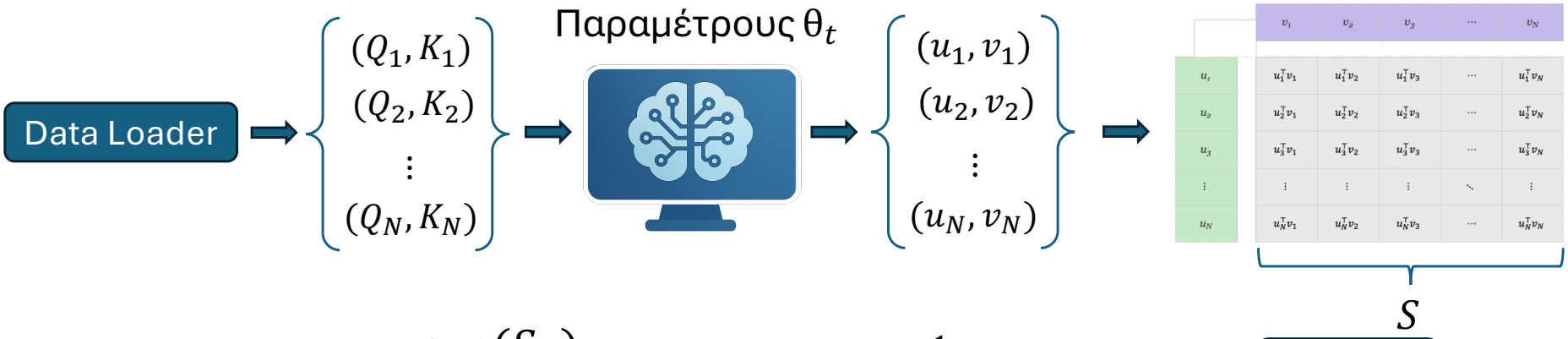
Step 3: $\mathcal{L}_i = -\log \frac{\exp(S_{ii})}{\sum_{k=1}^{B} \exp(S_{ik})}$

Step 4: $\mathcal{L} = \frac{1}{K} \sum_i \mathcal{L}_i$

	v_1	v_2	v_3	$\cdots$	v_N
u_1	$u_1^\top v_1$	$u_1^\top v_2$	$u_1^\top v_3$	$\cdots$	$u_1^\top v_N$
u_2	$u_2^\top v_1$	$u_2^\top v_2$	$u_2^\top v_3$	$\cdots$	$u_2^\top v_N$
u_3	$u_3^\top v_1$	$u_3^\top v_2$	$u_3^\top v_3$	$\cdots$	$u_3^\top v_N$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
u_N	$u_N^\top v_1$	$u_N^\top v_2$	$u_N^\top v_3$	$\cdots$	$u_N^\top v_N$

Η απώλεια είναι το άθροισμα της απώλειας cross entropy σε κάθε σειρά του πίνακα

InfoNCE και Contrastive Loss



$$\mathcal{L}_i(\theta_t) = -\log \frac{\exp(S_{ii})}{\sum_{k=1}^{B} \exp(S_{ik})} \rightarrow \mathcal{L}(\theta_t) = \frac{1}{K} \sum_i \mathcal{L}_i(\theta_t) \rightarrow \nabla_\theta \mathcal{L}(\theta_t)$$

$$\nabla_\theta \mathcal{L}(\theta_t) \rightarrow \texttt{optimizer.step()} \rightarrow \theta_{t+1}$$

InfoNCE και Contrastive Loss

Επεκτάσεις της μεθόδου

1. Θερμοκρασία: $\tilde{S}_{ij} = S_{ij}/\tau$
2. $(Q_i, K_i) \rightarrow \left(Q_i, K_{i_1}, \dots, K_{i_j}\right)$: Πάνω από 1 σχετικό αρχείο
3. $(Q_i, K_i) \rightarrow \left(Q_i, K_i, \tilde{K}_i\right)$: «αρνητικά» αρχεία
4. Βαθμός σχετικόκητας: $(Q_i, K_i) \rightarrow (Q_i, K_i, \rho_i)$

InfoNCE και Contrastive Loss

Επεκτάσεις της μεθόδου

1. Θερμοκρασία: $\tilde{S}_{ij} = S_{ij}/\tau$
2. $(Q_i, K_i) \rightarrow \left(Q_i, K_{i_1}, \dots, K_{i_j}\right)$: Πάνω από 1 σχετικό αρχείο
3. $(Q_i, K_i) \rightarrow \left(Q_i, K_i, \tilde{K}_i\right)$: «αρνητικά» αρχεία
4. Βαθμός σχετικόκητας: $(Q_i, K_i) \rightarrow (Q_i, K_i, \rho_i)$

Πώς δημιουργούμε καλά σύνολα δεδομένων $\{(Q_i, K_i)\}$