



Παραγωγική Τεχνητή Νοημοσύνη: Generative AI

Κωνσταντίνος Καραμανής

The University of Texas at Austin & Archimedes/Athena RC

constantine@utexas.edu

<https://caramanis.github.io/>



Αναζήτηση: λέξη σε κείμενο

Πώς γίνεται η αναζήτηση (search) σε κείμενο;

Αναζήτηση: λέξη σε κείμενο

Πώς γίνεται η αναζήτηση (search) σε κείμενο;

Query term: 'ποδήλατο'

Κείμενο: Το πρωί που οδηγούσα στο σχολείο είδα έναν άντρα ντυμένο Άγιο Βασίλη, καβάλα σε ένα ποδήλατο, και γέλασα πολύ.

Αναζήτηση: λέξη σε κείμενο

Πώς γίνεται η αναζήτηση (search) σε κείμενο;

Query term: ποδήλατο



Κείμενο: Το πρωί που οδηγούσα στο σχολείο είδα έναν άντρα ντυμένο Άγιο Βασίλη, καβάλα σε ένα ποδήλατο, και γέλασα πολύ.

Αναζήτηση: λέξη σε κείμενο

Πώς γίνεται η αναζήτηση (search) σε κείμενο;

Query term: ποδήλατο



Κείμενο: Το πρωί που οδηγούσα στο σχολείο είδα έναν άντρα ντυμένο Άγιο Βασίλη, καβάλα σε ένα ποδήλατο, και γέλασα πολύ.

Αναζήτηση: λέξη σε κείμενο

Πώς γίνεται η αναζήτηση (search) σε κείμενο;

Query term: ποδήλατο



Κείμενο: Το πρωί που οδηγούσα στο σχολείο είδα έναν άντρα ντυμένο Άγιο Βασίλη, καβάλα σε ένα ποδήλατο, και γέλασα πολύ.

Αναζήτηση: λέξη σε κείμενο

Πώς γίνεται η αναζήτηση (search) σε κείμενο;

Query term: ποδήλατο



Κείμενο: Το πρωί που οδηγούσα στο σχολείο είδα έναν άντρα ντυμένο Άγιο Βασίλη, καβάλα σε ένα ποδήλατο, και γέλασα πολύ.

Αναζήτηση: λέξη σε κείμενο

Πώς γίνεται η αναζήτηση (search) σε κείμενο;

Query term: ποδήλατο



Κείμενο: Το πρωί που οδηγούσα στο σχολείο είδα έναν άντρα ντυμένο Άγιο Βασίλη, καβάλα σε ένα ποδήλατο, και γέλασα πολύ.

Αναζήτηση: λέξη σε κείμενο

Πώς γίνεται η αναζήτηση (search) σε κείμενο;

Query term: 'ποδήλατο'



Κείμενο: Το πρωί που οδηγούσα **στο** σχολείο είδα έναν άντρα ντυμένο Άγιο Βασίλη, καβάλα σε ένα ποδήλατο, και γέλασα πολύ.

Αναζήτηση: λέξη σε κείμενο

Πώς γίνεται η αναζήτηση (search) σε κείμενο;

Query term: 'ποδήλατο'



Κείμενο: Το πρωί που οδηγούσα στο **σχολείο** είδα έναν άντρα ντυμένο Άγιο Βασίλη, καβάλα σε ένα ποδήλατο, και γέλασα πολύ.

Αναζήτηση: λέξη σε κείμενο

Πώς γίνεται η αναζήτηση (search) σε κείμενο;

Query term: ποδήλατο

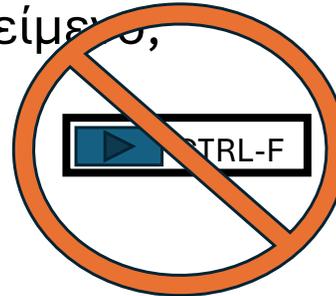


Κείμενο: Το πρωί που οδηγούσα στο σχολείο είδα έναν άντρα ντυμένο Άγιο Βασίλη, καβάλα σε ένα ποδήλατο, και γέλασα πολύ.

Αναζήτηση: λέξη έννοια σε κείμενο

Πώς γίνεται η αναζήτηση (search) σε κείμενο,

Query term: 'ποδήλατο'



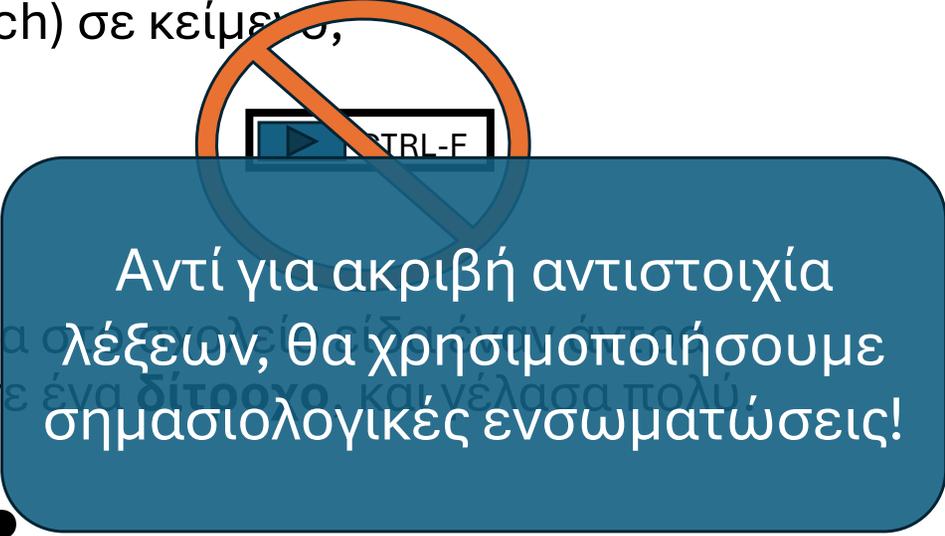
Κείμενο: Το πρωί που οδηγούσα στο σχολείο είδα έναν άντρα ντυμένο Άγιο Βασίλη, καβάλα σε ένα **δίτροχο**, και γέλασα πολύ.

Αναζήτηση: λέξη έννοια σε κείμενο

Πώς γίνεται η αναζήτηση (search) σε κείμενο,

Query term: 'ποδήλατο'

Κείμενο: Το πρωί που οδηγούσα
ντυμένο Άγιο Βασίλη, καβάλα σε



Αντί για ακριβή αντιστοιχία λέξεων, θα χρησιμοποιήσουμε σημασιολογικές ενσωματώσεις!



Ας θυμηθούμε τα
προηγούμενα...

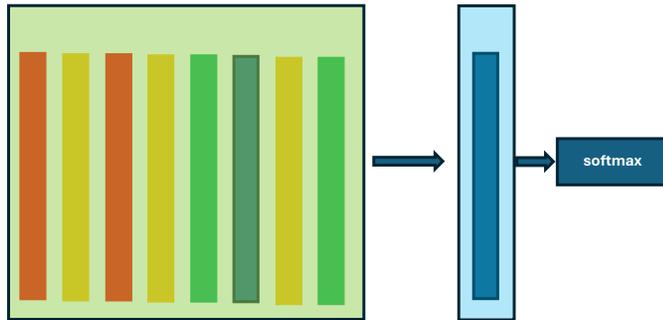


Εσωτερικό Γινόμενο

Το Εσωτερικό Γινόμενο $\langle W, X \rangle$ είναι μεγάλο όταν το W και το X :

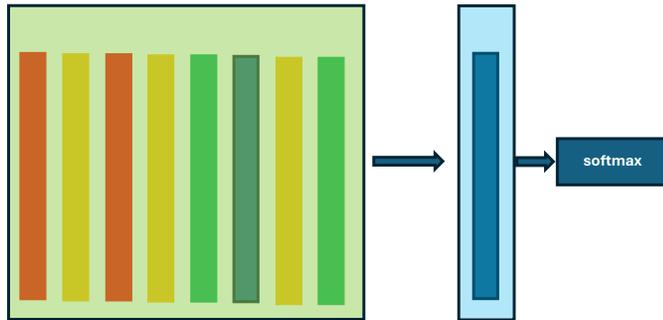
- «Μοιάζουν»
- Είναι κοντά στον γεωμετρικό χώρο
- Σχηματίζουν μικρή γωνία $\angle WX$

Convolutional Νευρωνικό Δίκτυο

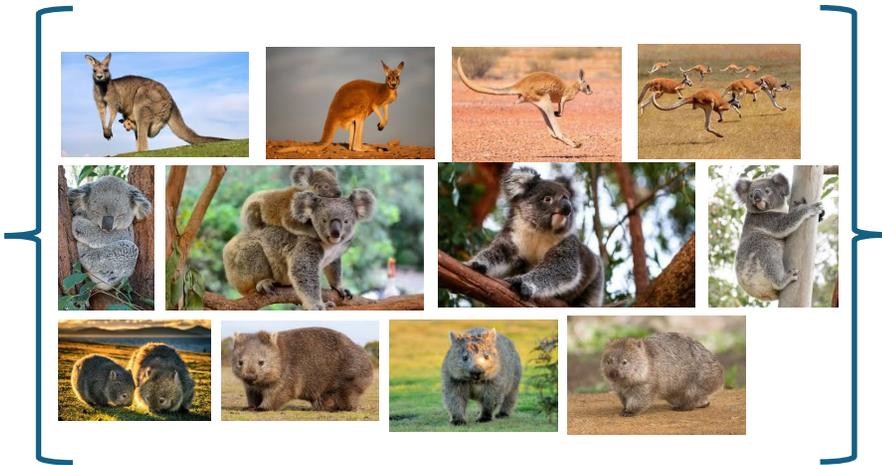


Δεδομένα {φασκλώμυς, κοάλα, καγκουρό}

Convolutional Νευρωνικό Δίκτυο

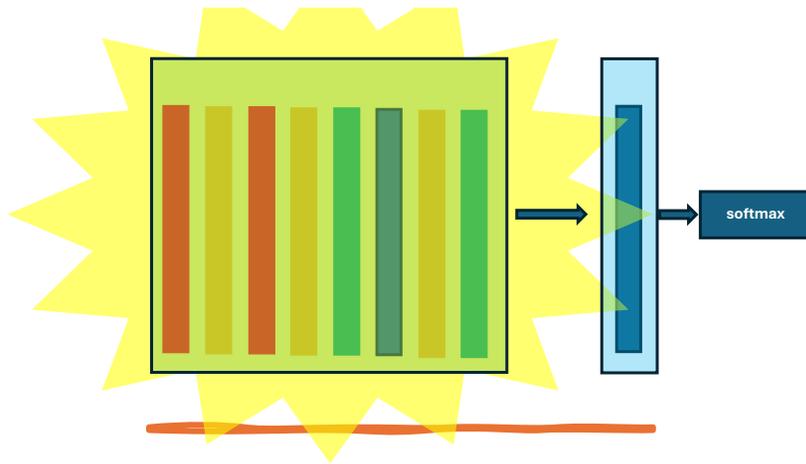


Δεδομένα {φασκλώμους, κοάλα, καγκουρό}

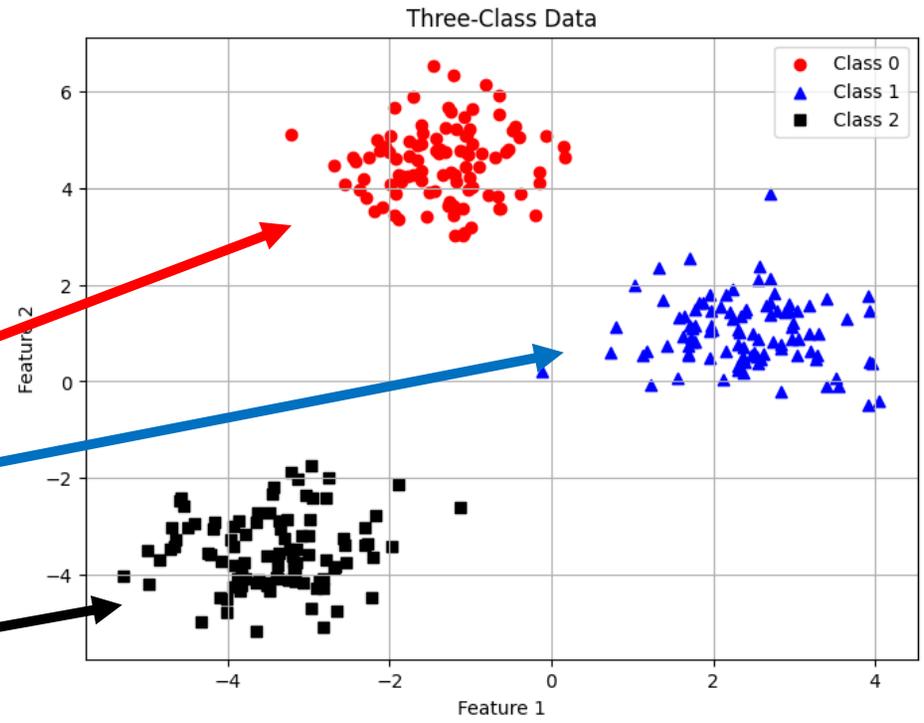
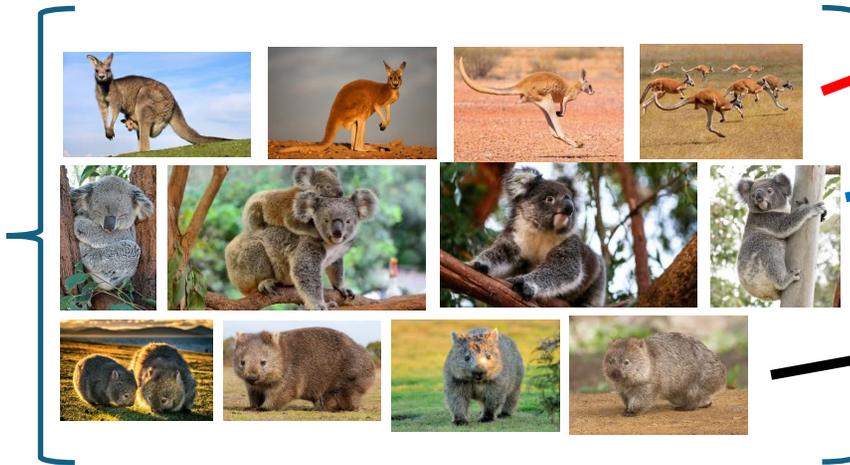


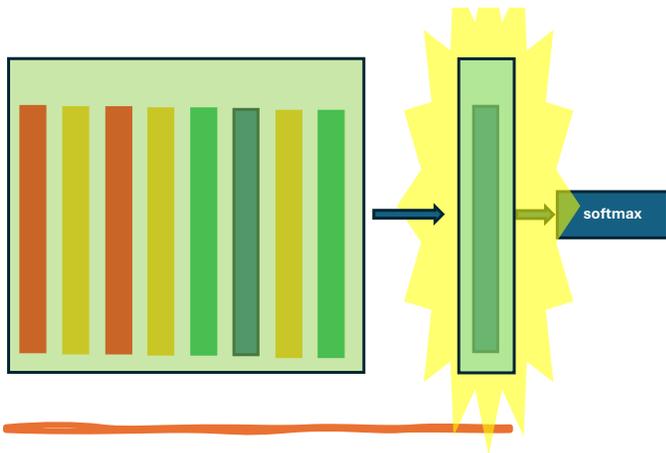
Κωνσταντίνος Καραμανής

Convolutional Νευρωνικό Δίκτυο



Δεδομένα {φασκλώμους, κοάλα, καγκουρό}



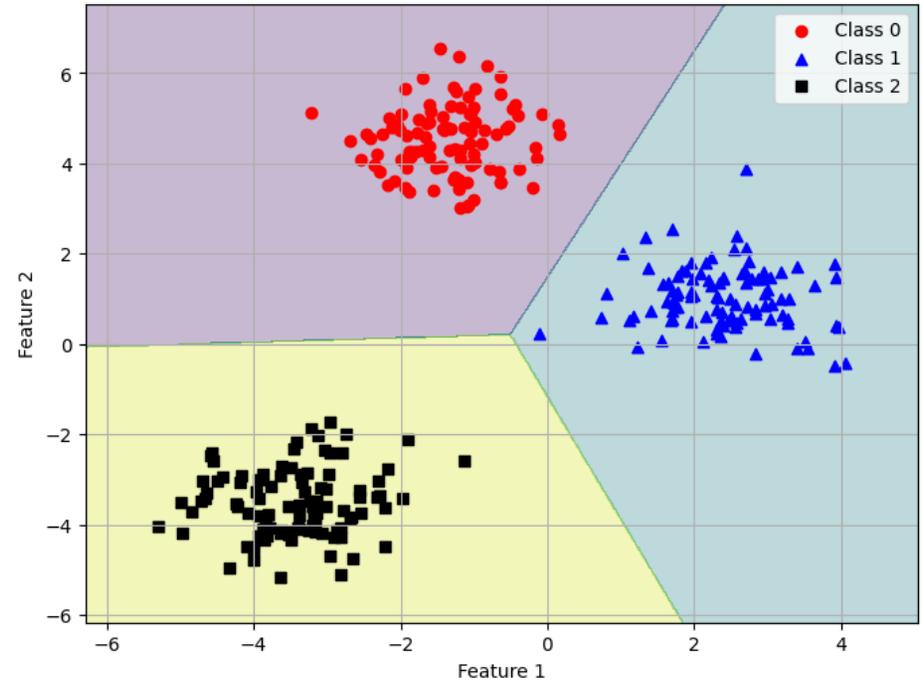


Δεδομένα {φασκλώμυς, κοάλα, καγκουρό}

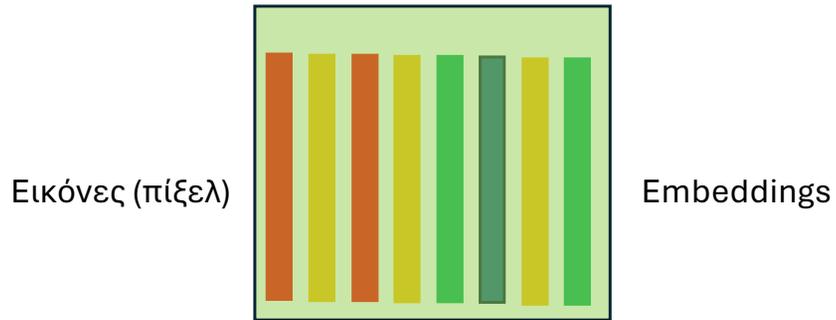


Convolutional Νευρωνικό Δίκτυο

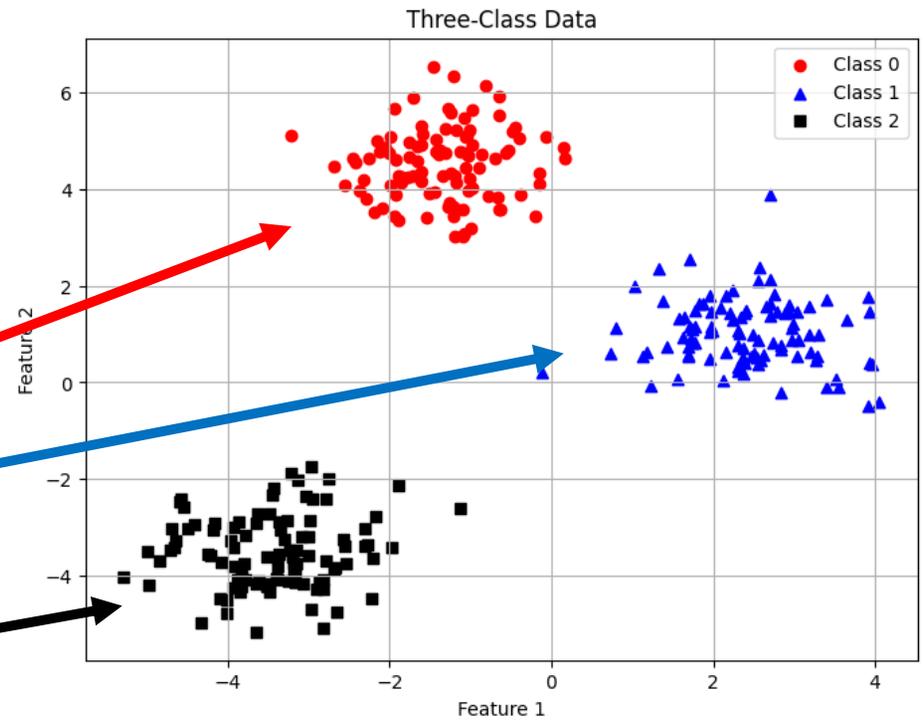
Three-Class Logistic Regression Decision Boundaries



Embeddings (Ενσωματώσεις)



Δεδομένα {φασκλώμους, κοάλα, καγκουρό}



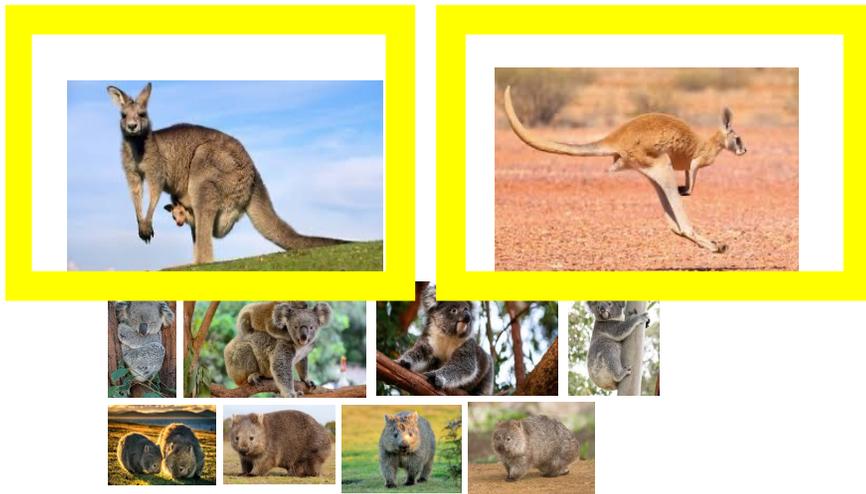
Embeddings: η απόσταση στον γεωμετρικό χώρο αντικαθιστά την απόσταση που αντιλαμβανόμαστε



Embeddings: η απόσταση στον γεωμετρικό χώρο αντικαθιστά την απόσταση που αντιλαμβανόμαστε

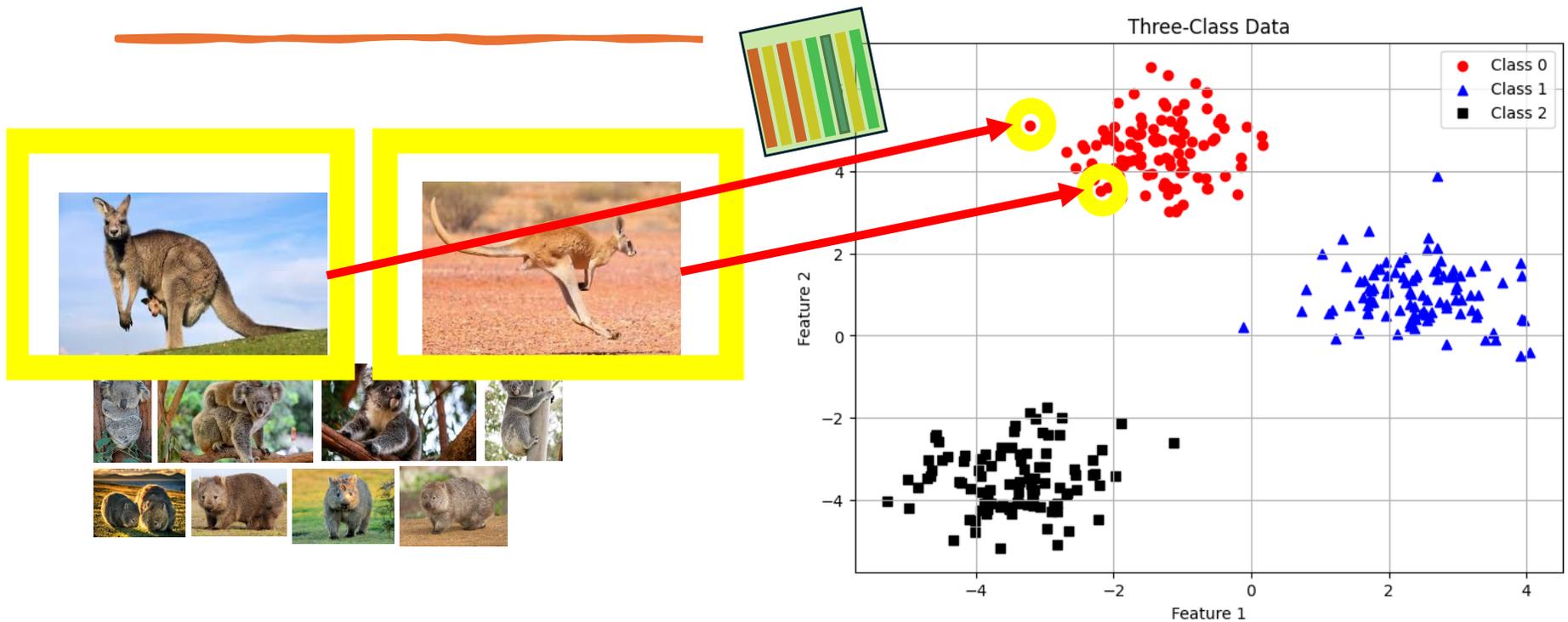


Embeddings/Ενσωματώσεις: η απόσταση στον γεωμετρικό χώρο αντικαθιστά την «σημασιολογική» απόσταση

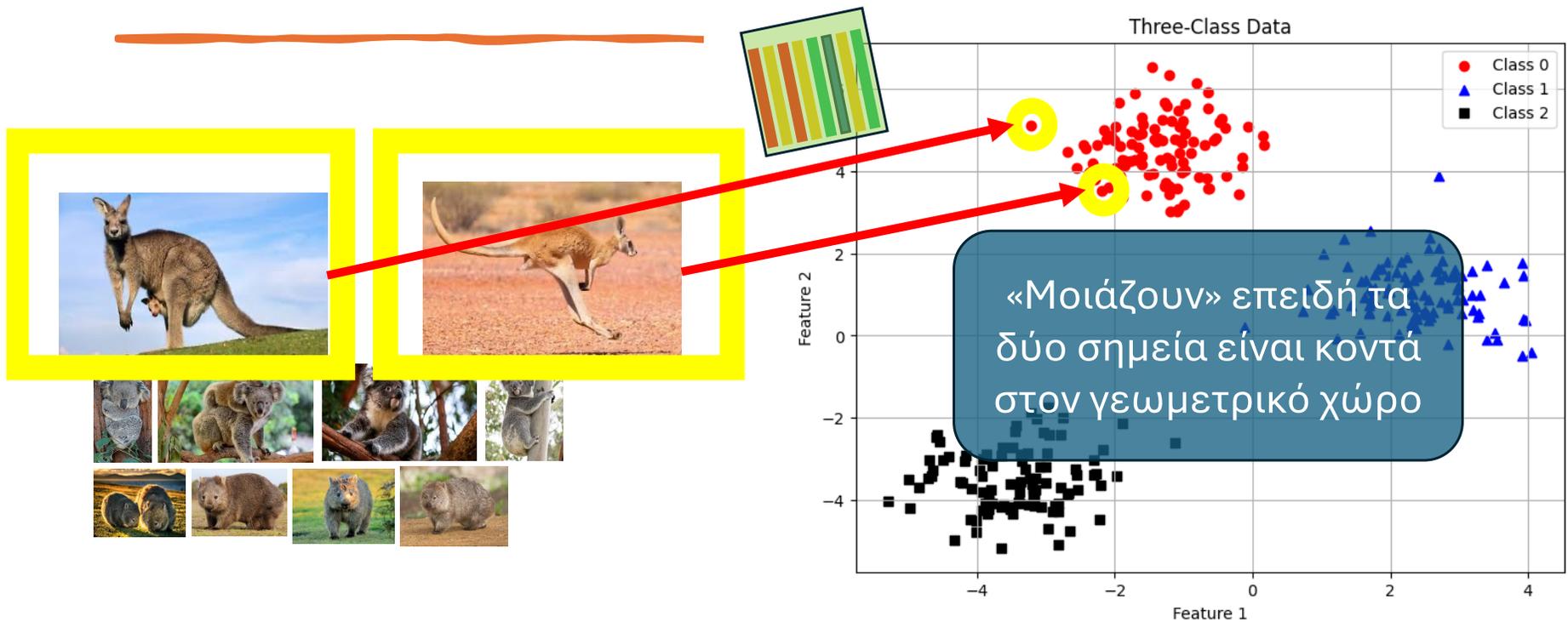


Πίξελ προς πίξελ δεν μοιάζουν καθόλου οι δύο εικόνες. «Μοιάζουν» επειδή απεικονίζουν και οι δύο καγκουρό

Embeddings/Ενσωματώσεις: η απόσταση στον γεωμετρικό χώρο αντικαθιστά την «σημασιολογική» απόσταση

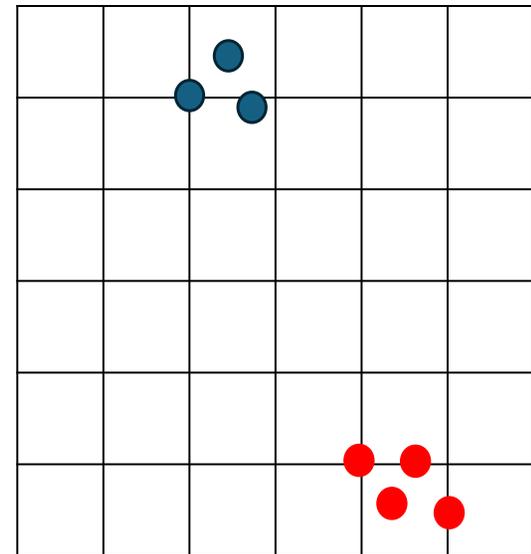
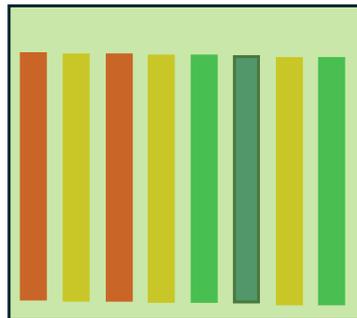


Embeddings/Ενσωματώσεις: η απόσταση στον γεωμετρικό χώρο αντικαθιστά την «σημασιολογική» απόσταση

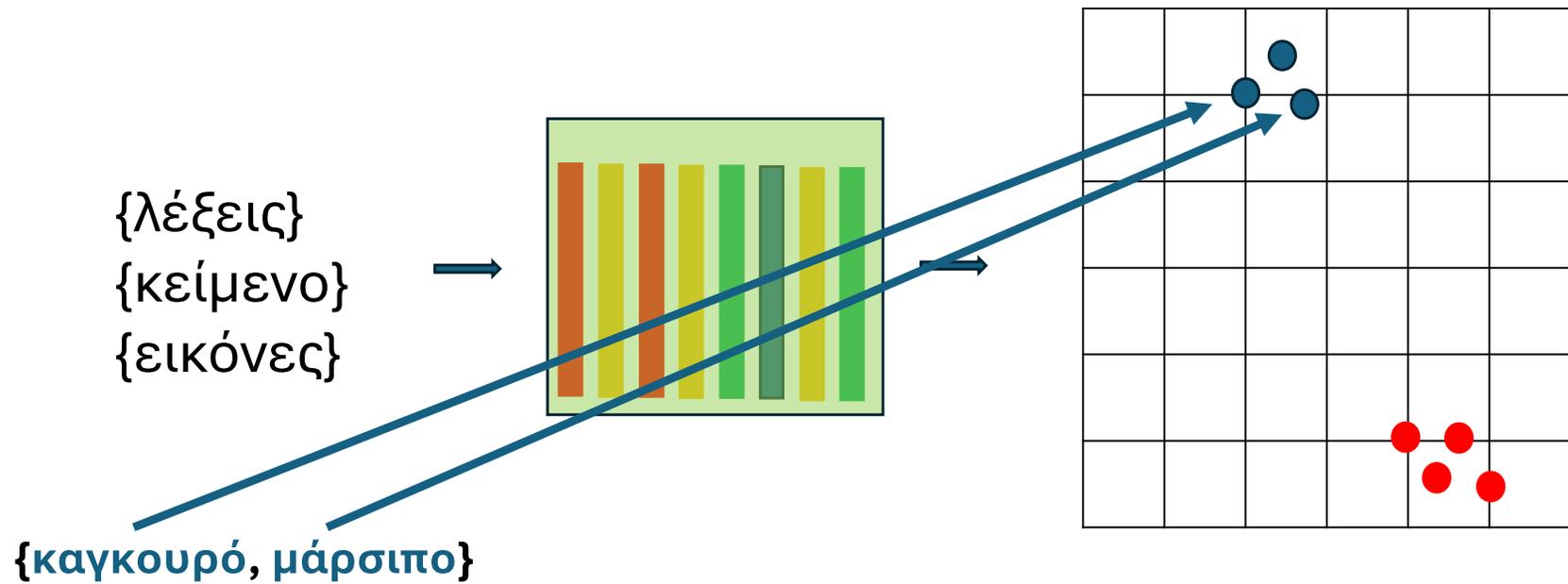


Embeddings/Ενσωματώσεις: η απόσταση στον γεωμετρικό χώρο αντικαθιστά την «σημασιολογική» απόσταση

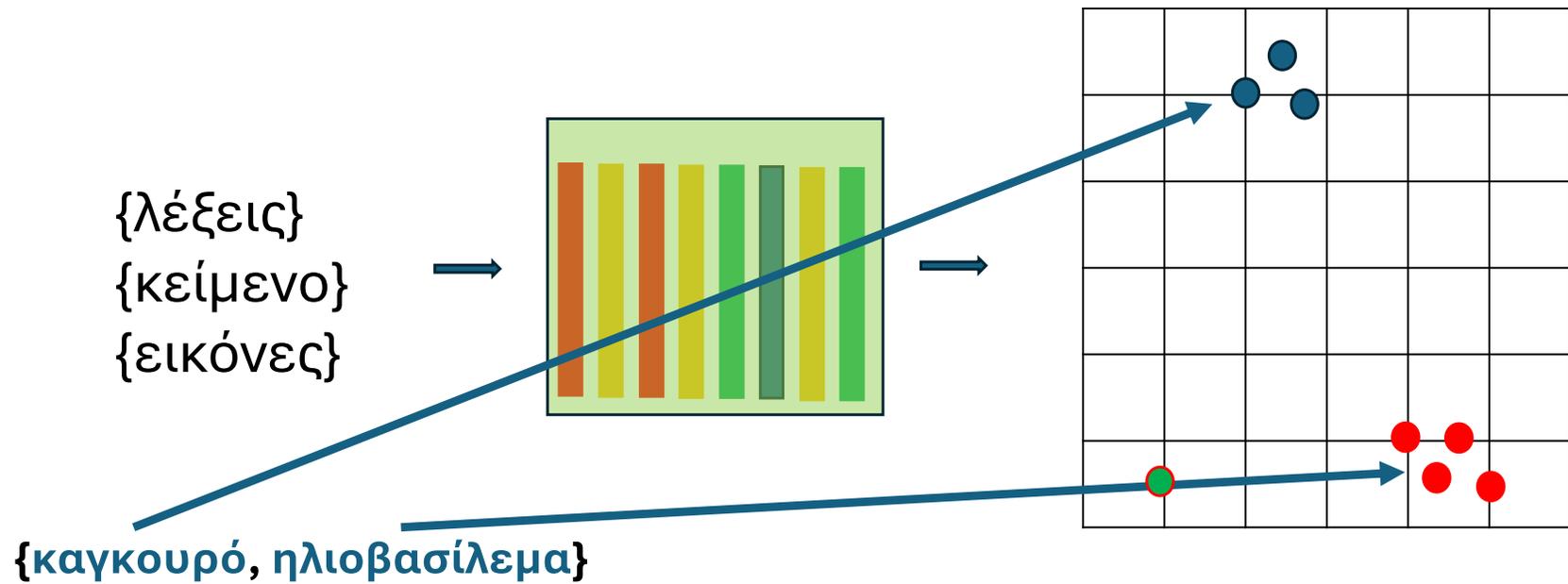
{λέξεις}
{κείμενο}
{εικόνες}



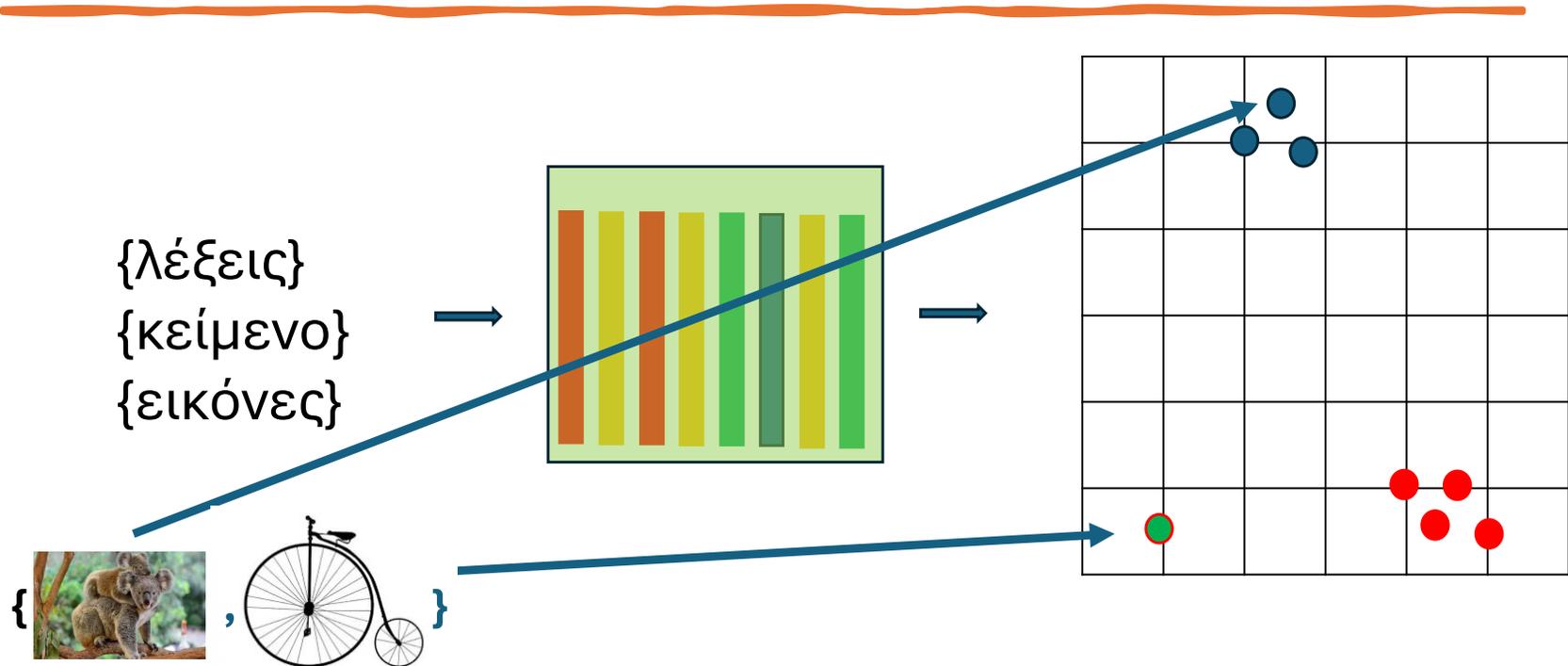
Embeddings/Ενσωματώσεις: η απόσταση στον γεωμετρικό χώρο αντικαθιστά την «σημασιολογική» απόσταση



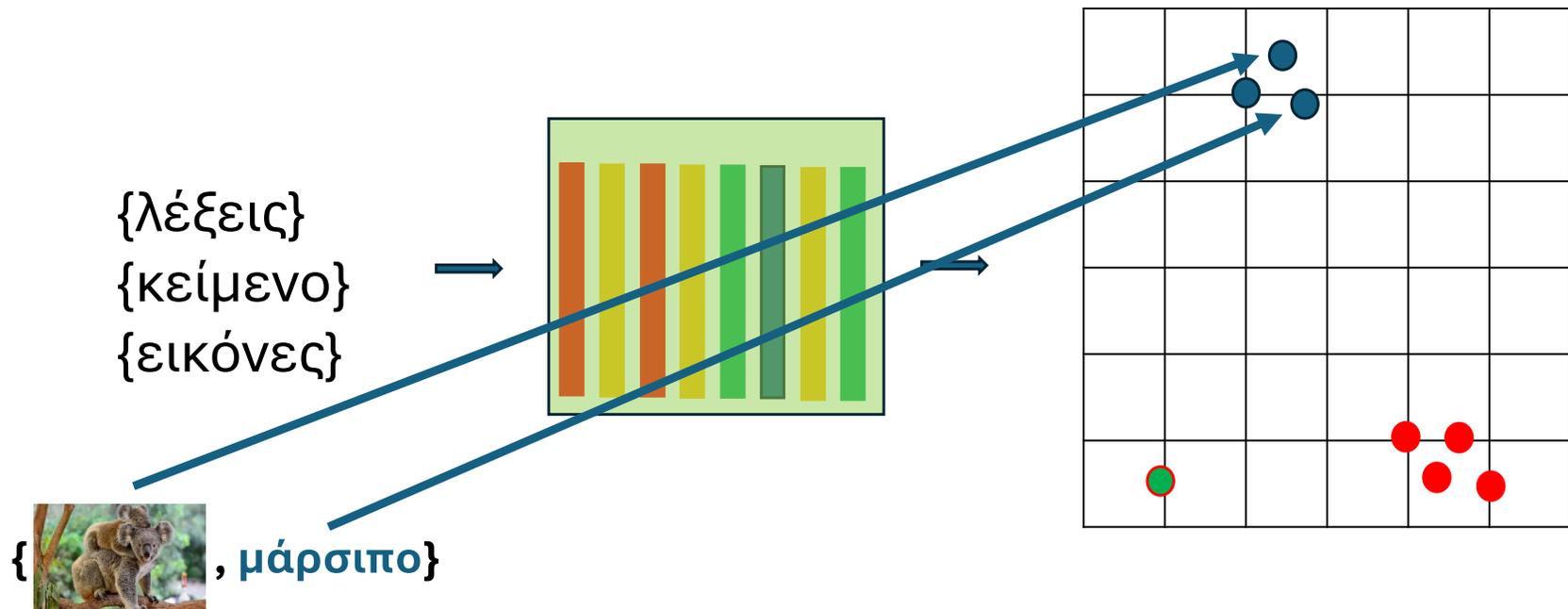
Embeddings/Ενσωματώσεις: η απόσταση στον γεωμετρικό χώρο αντικαθιστά την «σημασιολογική» απόσταση



Embeddings/Ενσωματώσεις: η απόσταση στον γεωμετρικό χώρο αντικαθιστά την «σημασιολογική» απόσταση



Embeddings/Ενσωματώσεις: η απόσταση στον γεωμετρικό χώρο αντικαθιστά την «σημασιολογική» απόσταση

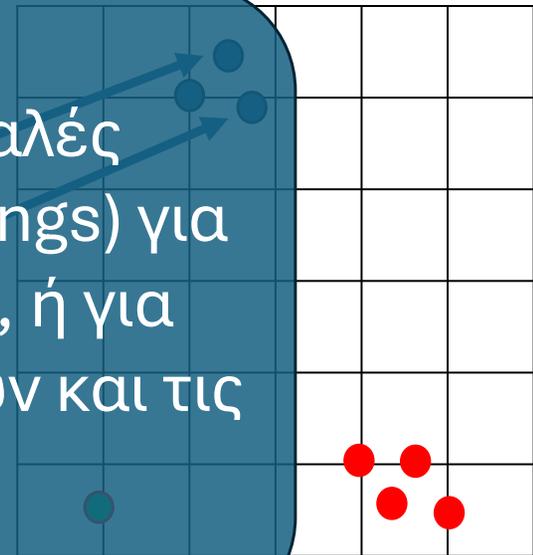


Embeddings/Ενσωματώσεις: η απόσταση στον γεωμετρικό χώρο αντικαθιστά την «σημασιολογική» απόσταση

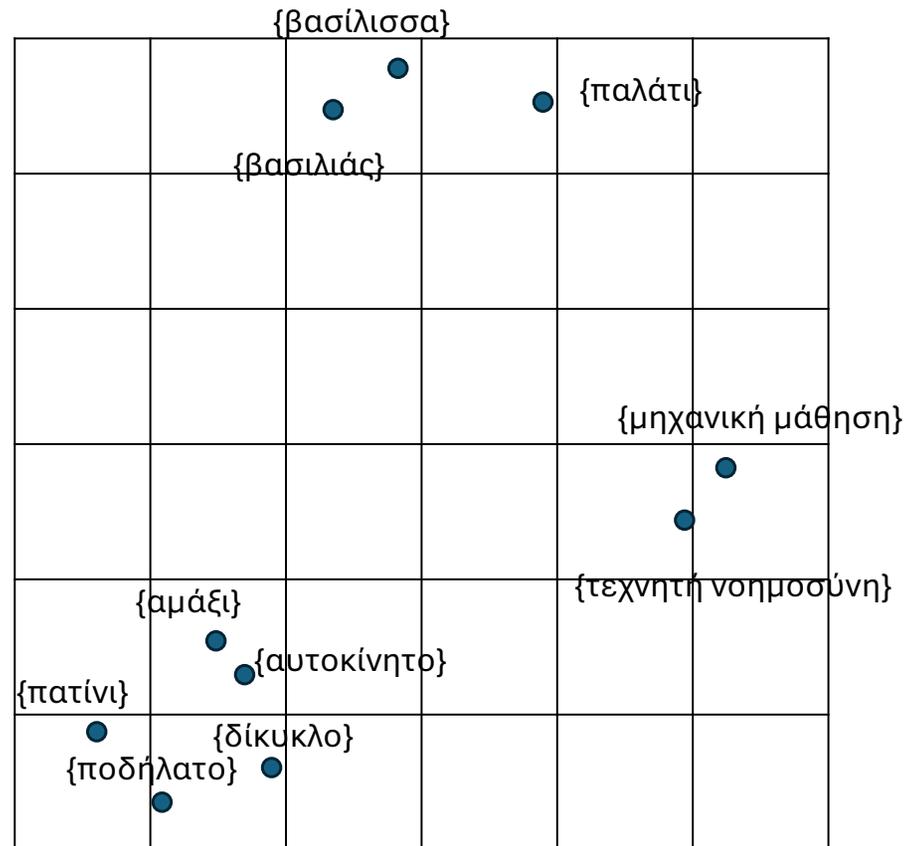
Πώς δημιουργούμε καλές ενσωματώσεις (embeddings) για {λέξεις}, {κείμενο}, {εικόνες} εικόνες, λέξεις, κείμενο, ή για δεδομένα που συνδυάζουν και τις δύο μορφές;



{, μάρσιπο}



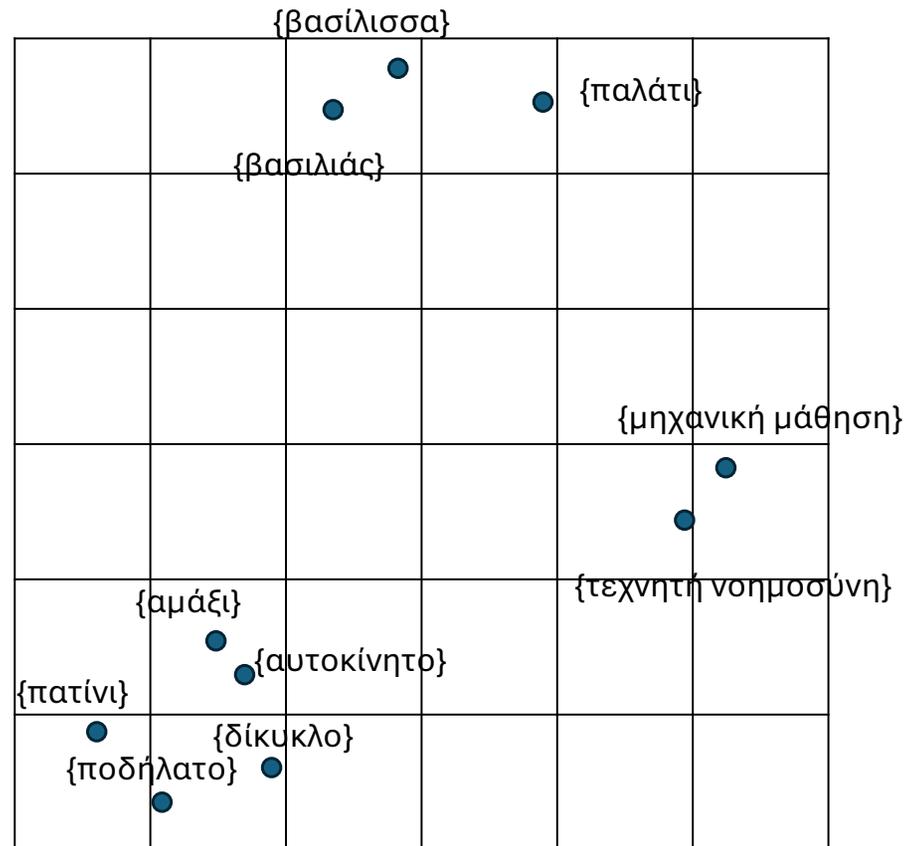
Word2Vec*: σημασιολογικές ενσωματώσεις λέξεων



*Mikolov, T., Chen, K., Corrado, G. and Dean, J., 2013. Efficient estimation of word representations in vector space.
arXiv preprint arXiv:1301.3781

Κωνσταντίνος Καραμανής

Word2Vec*: σημασιολογικές ενσωματώσεις λέξεων



*Mikolov, T., Chen, K., Corrado, G. and Dean, J., 2013. Efficient estimation of word representations in vector space.

[arXiv preprint arXiv:1301.3781](https://arxiv.org/abs/1301.3781)

Κωνσταντίνος Καραμανής

Πώς θα πετύχουμε τον στόχο μας;

- Word2Vec:**
1. Εάν έχουμε 30.000 λέξεις, έχουμε 900.000.000 ζεύγη. Δεν είναι πρακτικό (εφικτό) να χαρακτηρίσουμε όλα αυτά τα ζεύγη «με το χέρι».
 2. Και να είχαμε 900.000.000 θεμιτές γωνίες (αποστάσεις) $\theta_{\alpha\beta}$ από την λέξη α στην λέξη β , πως θα βρίσκαμε διανύσματα v_α, v_β ώστε $\angle v_\alpha v_\beta \approx \theta_{\alpha\beta}$ για κάθε ζεύγος (α, β) ;

*Mikolov, T., Chen, K., Corrado, G. and Dean, J., 2013. Efficient estimation of word representations in vector space.

[arXiv preprint arXiv:1301.3781](https://arxiv.org/abs/1301.3781)

Κωνσταντίνος Καραμανής

Η βασική ιδέα: self-supervision

- Θα λύσουμε ταυτόχρονα και τα δύο αυτά τα προβλήματα με την ευφυή ιδέα του **self-supervision**
- Θα εκμεταλλευτούμε μια σχεδόν ανεξάντλητη πηγή: το κείμενο στο διαδίκτυο
- Πρέπει να δημιουργήσουμε ένα πρόβλημα που να υπάγεται στο πρότυπο της *επιτηρούμενης μάθησης*

Πώς θα πετύχουμε τον στόχο μας;

1. Εάν έχουμε 30.000 λέξεις, έχουμε 900.000.000 ζεύγη. Δεν είναι πρακτικό (εφικτό) να χαρακτηρίσουμε όλα αυτά τα ζεύγη «με το χέρι».
2. Και να είχαμε 900.000.000 θεμιτές γωνίες (αποστάσεις) $\theta_{\alpha\beta}$ από την λέξη α στην λέξη β , πως θα βρίσκαμε διανύσματα v_α, v_β ώστε $\angle v_\alpha v_\beta \approx \theta_{\alpha\beta}$ για κάθε ζεύγος (α, β) ;

Επιτηρούμενη Μάθηση



Κωνσταντίνος Καραμανής

Επιτηρούμενη Μάθηση:
Βλέπουμε n παραδείγματα:
 $(X_1, y_1), (X_2, y_2), \dots, (X_n, y_n)$

Μαθαίνουμε τη σχέση $X \rightarrow y$,
δηλαδή μαθαίνουμε συνάρτηση
 $h(X)$ που να είναι «κοντά» στο y .

Επιτηρούμενη Μάθηση

Επιτηρούμενη Μάθηση:
Βλέπουμε n παραδείγματα:
 $(X_1, y_1), (X_2, y_2), \dots, (X_n, y_n)$

Το αρχικό μας πρόβλημα είναι να βρούμε σημασιολογικές ενσωματώσεις. Για να χρησιμοποιήσουμε όλα τα εργαλεία της επιτηρούμενης μάθησης πρέπει να διαμορφώσουμε το πρόβλημά μας στη σωστή μορφή.

Χρειαζόμαστε n «παραδείγματα» (X, y)

→ y ,
τήση
το y .



Η βασική ιδέα: self-supervision

*«Αποχαιρετά τη γη, φεύγει.
Περνάει την παγωμένη θάλασσα,
τρικυμία ξεσπάει, συντρίβεται το
καράβι σε άγρια χιονισμένα βράχια.»*

Η βασική ιδέα: self-supervision

«**Αποχαιρετά** τη **γη**, φεύγει.
Περνάει την παγωμένη θάλασσα,
τρικυμία ξεσπάει, συντρίβεται το
καράβι σε άγρια χιονισμένα βράχια.»

$X_1 = (\text{Αποχαιρετά}, \text{γη}), y_1 = P(\text{γη} \text{ ακολουθεί } \text{αποχαιρετά})$

Η βασική ιδέα: self-supervision

«**Αποχαιρετά** τη **γη**, φεύγει.
Περνάει την παγωμένη θάλασσα,
τρικυμία ξεσπάει, συντρίβεται το
καράβι σε άγρια χιονισμένα βράχια.»

$X_1 = (\text{Αποχαιρετά}, \text{γη}), y_1 = P(\text{γη} \text{ ακολουθεί } \text{αποχαιρετά})$

Υπολογίζεται από το
συνολικό σώμα
κειμένων που
χρησιμοποιούμε

Η βασική ιδέα: self-supervision

«Αποχαιρετά τη γη, φεύγει.
Περνάει την **παγωμένη θάλασσα**,
τρικυμία ξεσπάει, συντρίβεται το
καράβι σε άγρια χιονισμένα βράχια.»

$X_1 = (\text{Αποχαιρετά}, \text{γη}), y_1 = P(\text{γη ακολουθεί αποχαιρετά})$

$X_2 = (\text{παγωμένη}, \text{θάλασσα}), y_2 = P(\text{θάλασσα ακολουθεί παγωμένη})$

Υπολογίζεται από το
συνολικό σώμα
κειμένων που
χρησιμοποιούμε

Η βασική ιδέα: self-supervision

«Αποχαιρετά τη γη, φεύγει.
Περνάει την παγωμένη θάλασσα,
τρικυμία ξεσπάει, συντρίβεται το
καράβι σε άγρια χιονισμένα βράχια.»

$X_1 = (\text{Αποχαιρετά, γη}), y_1 = P(\text{γη ακολουθεί αποχαιρετά})$

$X_2 = (\text{παγωμένη, θάλασσα}), y_2 = P(\text{θάλασσα ακολουθεί παγωμένη})$

$X_3 = (\text{τρικυμία, ξεσπάει}), y_3 = P(\text{ξεσπάει ακολουθεί τρικυμία})$

$X_4 = (\text{συντρίβεται, καράβι}), y_4 = P(\text{καράβι ακολουθεί συντρίβεται})$

Υπολογίζεται από το
συνολικό σώμα
κειμένων που
χρησιμοποιούμε

Η βασική ιδέα: self-supervision

«Η κόρη της αγαπούσε πολύ { τις γάτες
τα σκυλιά
το παγωτό } »

		{σκύλος}	{γάτα}	
		●	●	
		{παγωτό}		
		●		

Η βασική ιδέα: self-supervision

«Η κόρη της αγαπούσε πολύ { τις γάτες
τα σκυλιά
το παγωτό } »

Εάν οι λέξεις {γάτα, σκύλος, παγωτό}
ακολουθούν τη λέξη {αγαπώ} με την ίδια
συχνότητα στο σώμα κειμένων μας,
θέλουμε τα ανάλογα διανύσματα να είναι
κοντά στο χώρο

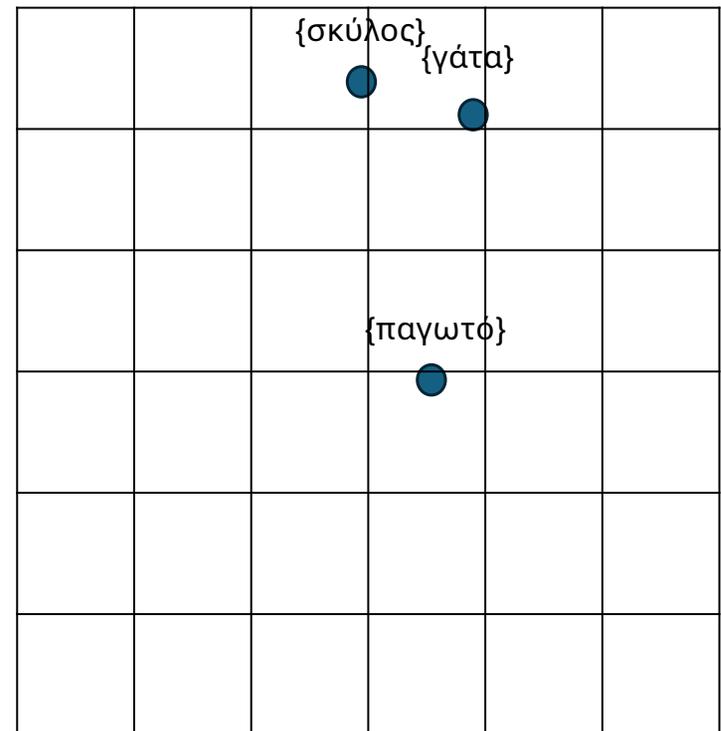
Κωνσταντίνος Καραμανής

		{σκύλος}	{γάτα}	
		●	●	
		{παγωτό}		
		●		

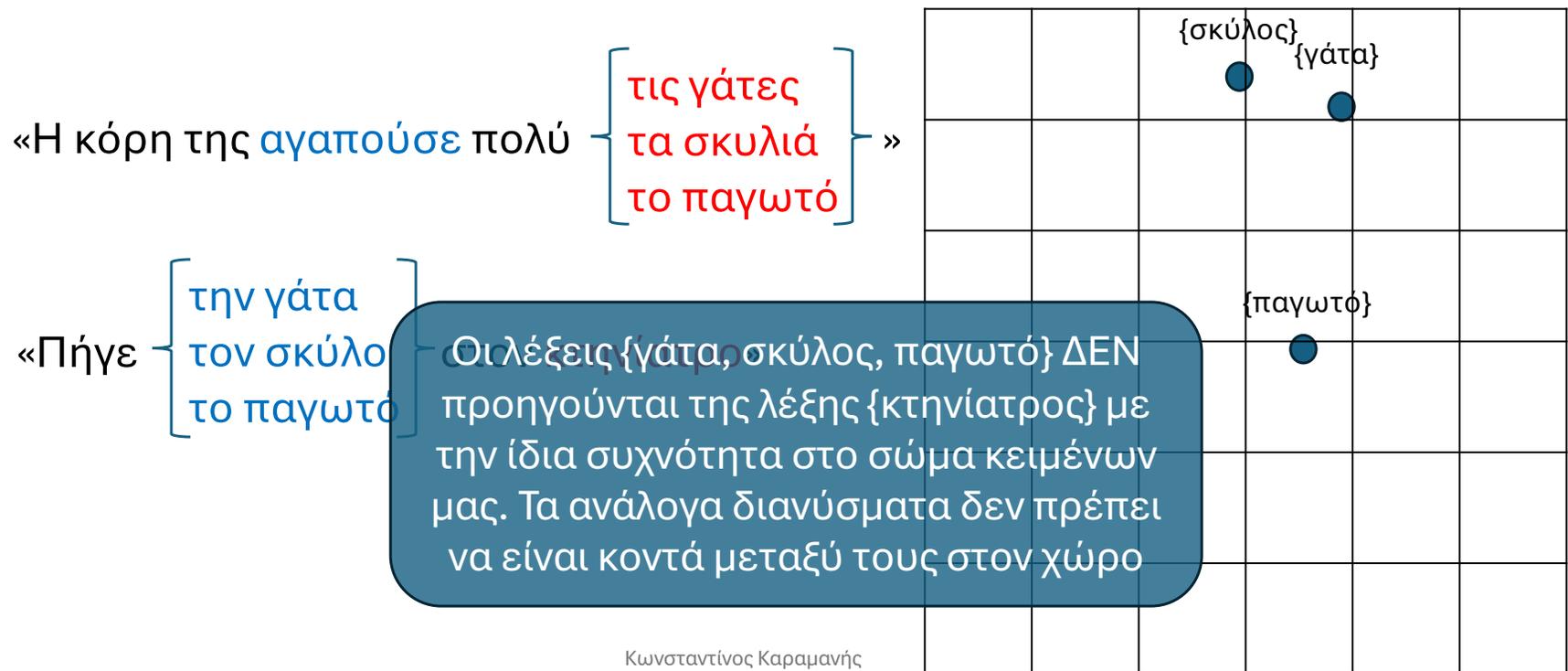
Η βασική ιδέα: self-supervision

«Η κόρη της αγαπούσε πολύ { τις γάτες
τα σκυλιά
το παγωτό } »

«Πήγε { την γάτα
τον σκύλο
το παγωτό } στον κτηνίατρο»



Η βασική ιδέα: self-supervision



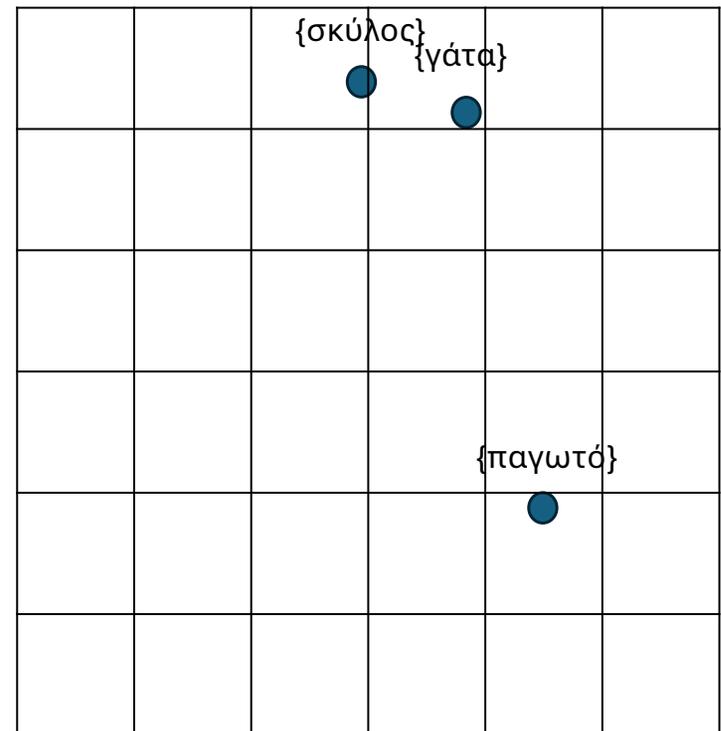
Η βασική ιδέα: self-supervision

«Η κόρη της αγαπούσε πολύ { τις γάτες
τα σκυλιά
το παγωτό } »

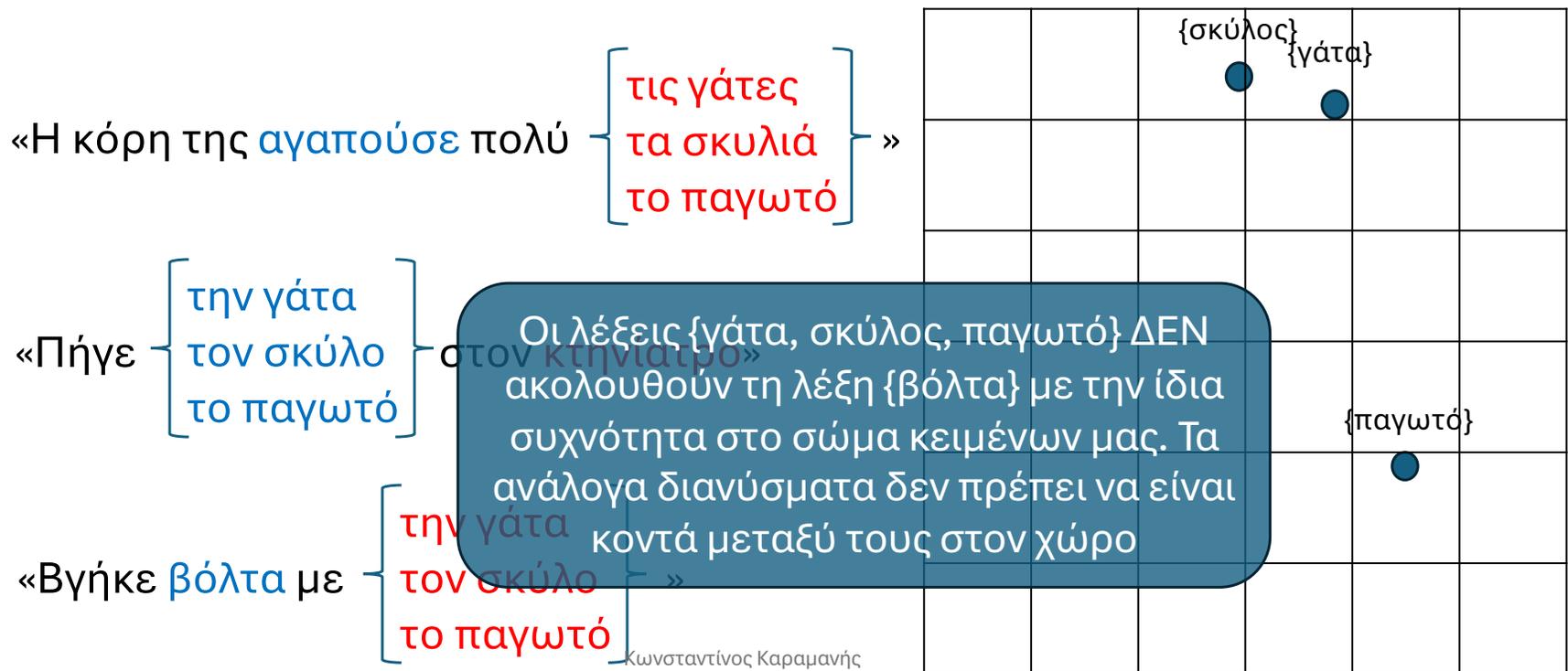
«Πήγε { την γάτα
τον σκύλο
το παγωτό } στον κτηνίατρο»

«Βγήκε βόλτα με { την γάτα
τον σκύλο
το παγωτό } »

Κωνσταντίνος Καραμανής



Η βασική ιδέα: self-supervision



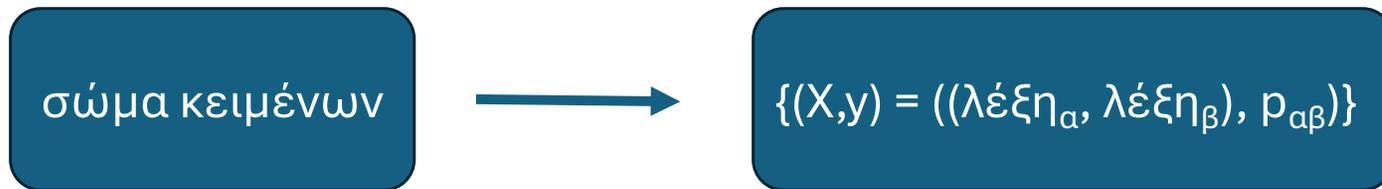
Self-supervision & Λογιστική Παλινδρόμηση

σώμα κειμένων



$\{(X,y) = ((\lambda\epsilon\chi\eta_\alpha, \lambda\epsilon\chi\eta_\beta), p_{\alpha\beta})\}$

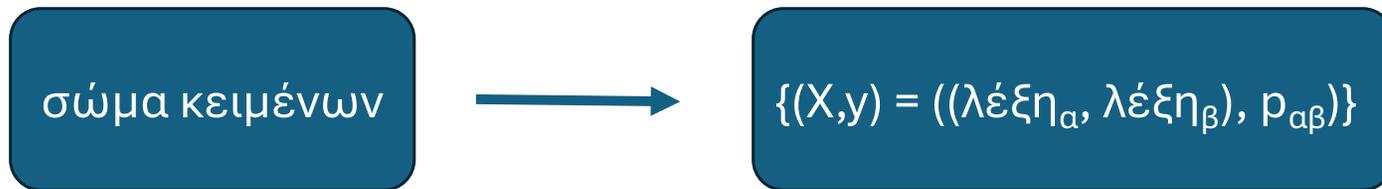
Self-supervision & Λογιστική Παλινδρόμηση



Για κάθε λέξη $\{\lambda\epsilon\chi\eta_\alpha\}$ θέλουμε να βρούμε διανύσματα $\mathbf{u}_\alpha, \mathbf{v}_\alpha$ ώστε:

$$p_{\alpha\beta} \approx \frac{\exp(\mathbf{u}_\alpha^T \mathbf{v}_\beta)}{\sum_{\alpha' \in A} \exp(\mathbf{u}_{\alpha'}^T \mathbf{v}_\beta)}$$

Self-supervision & Λογιστική Παλινδρόμηση



Για κάθε λέξη $\{\lambda\epsilon\chi\eta_\alpha\}$ θέλουμε να βρούμε διανύσματα $\mathbf{u}_\alpha, \mathbf{v}_\alpha$ ώστε:

$$p_{\alpha\beta} \approx \frac{\exp(\mathbf{u}_\alpha^T \mathbf{v}_\beta)}{\sum_{\alpha' \in A} \exp(\mathbf{u}_{\alpha'}^T \mathbf{v}_\beta)}$$

Υπολογίζεται από το κείμενο

Κωνσταντίνος Καραμανής

Υπολογίζεται από τις ενσωματώσεις

Self-supervision & Λογιστική Παλινδρόμηση

σώμα κειμένων



$\{(X,y) = ((\lambda\acute{\epsilon}\xi\eta_{\alpha}, \lambda\acute{\epsilon}\xi\eta_{\beta}), p_{\alpha\beta})\}$

Για κάθε λέξη $\{\lambda\acute{\epsilon}\xi\eta_{\alpha}\}$ θέλουμε να βρούμε διανύσματα $\mathbf{u}_{\alpha}, \mathbf{v}_{\alpha}$ ώστε:

$$p_{\alpha\beta} \approx \frac{\exp(\mathbf{u}_{\alpha}^T \mathbf{v}_{\beta})}{\sum_{\alpha' \in A} \exp(\mathbf{u}_{\alpha'}^T \mathbf{v}_{\beta})}$$

softmax

Νευρωνικό Δίκτυο για το Word2Vec

άα
άβακας
..
άλογο
..

λάδι
λαμπρός
λαός
λέξη
..
..

..
ωχρός
ωχρότητα
ωψ



Νευρωνικό Δίκτυο για το Word2Vec

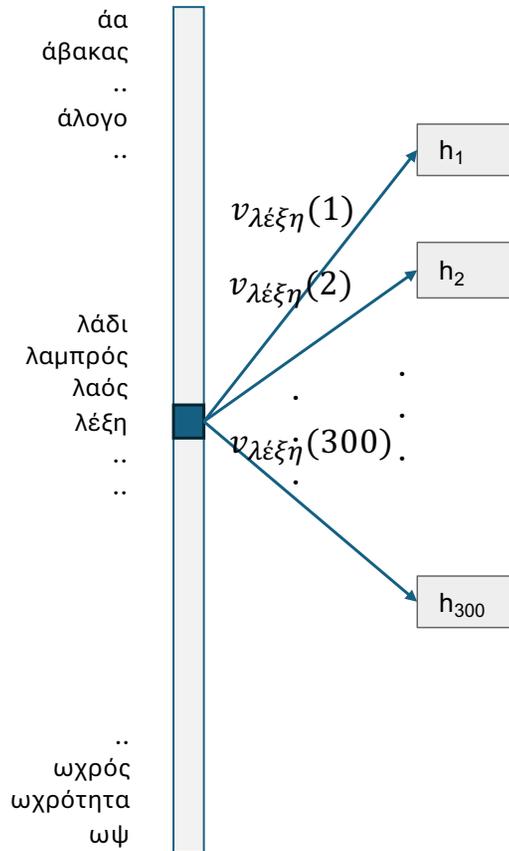
άα
άβακας
..
άλογο
..

λάδι
λαμπρός
λαός
λέξη
..
..

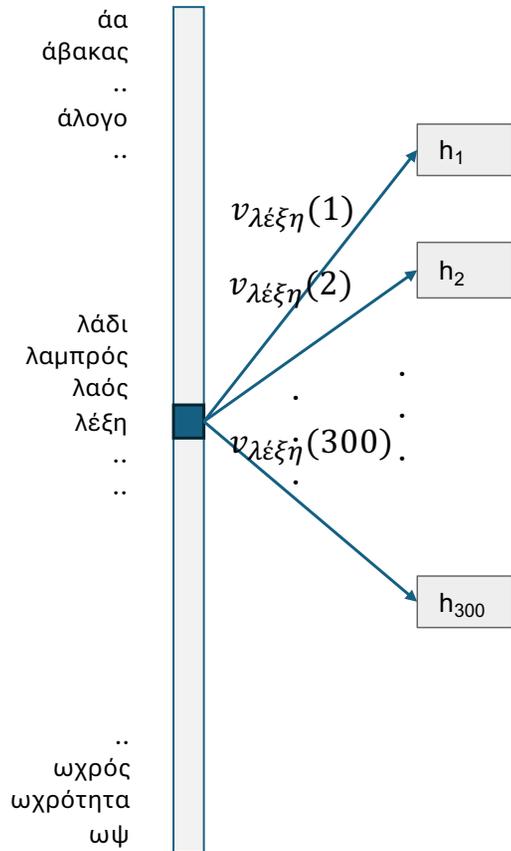
..
ωχρός
ωχρότητα
ωψ

Αρχίζουμε με όλες τις
λέξεις στο λεξικό μας – το
σύνολο λέξεων στα
κείμενά μας

Νευρωνικό Δίκτυο για το Word2Vec



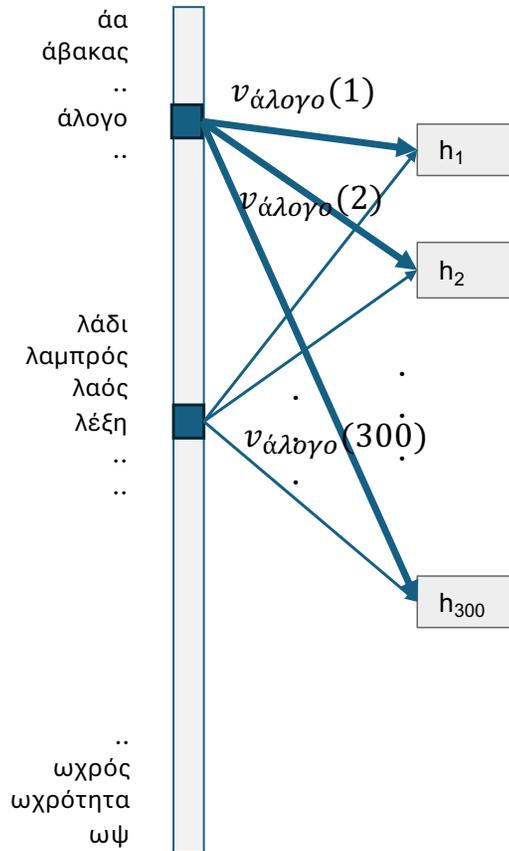
Νευρωνικό Δίκτυο για το Word2Vec



$v_{\text{λέξη}} = \begin{pmatrix} v_{\text{λέξη}}(1) \\ v_{\text{λέξη}}(2) \\ \vdots \\ v_{\text{λέξη}}(300) \end{pmatrix}$

Ενσωμάτωση (embedding) ως δεύτερη (ακόλουθη) λέξη σε 300 διαστάσεις.

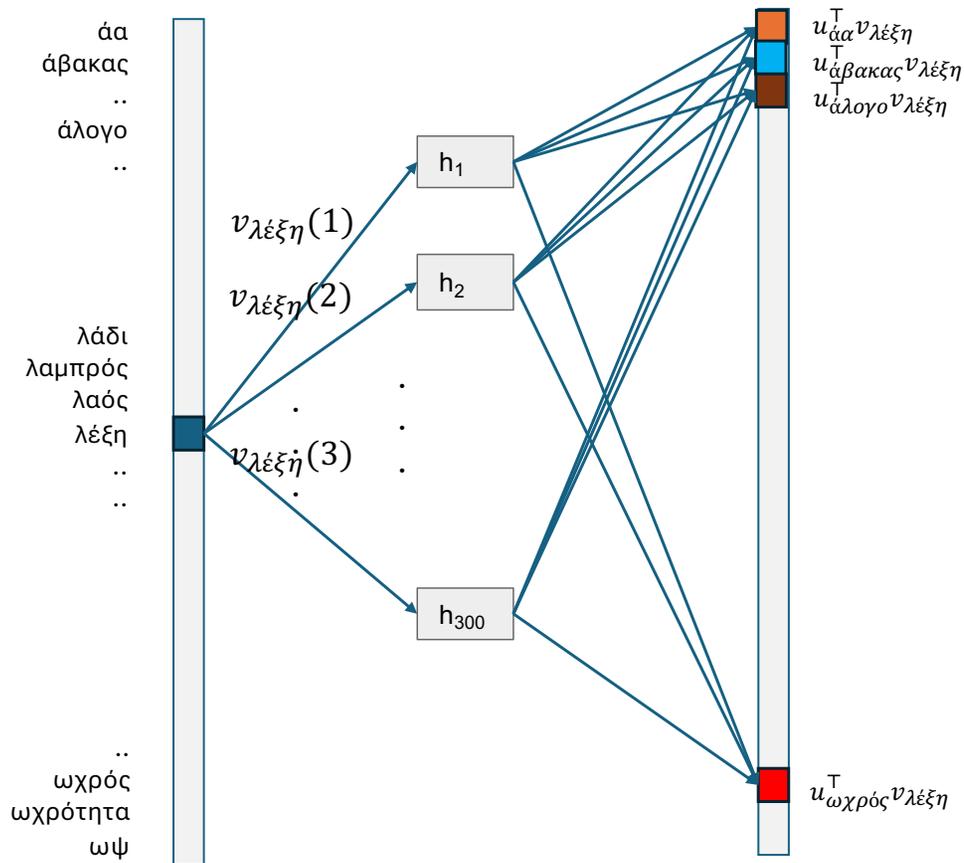
Νευρωνικό Δίκτυο για το Word2Vec



$$v_{\text{άλογο}} = \begin{pmatrix} v_{\text{άλογο}}(1) \\ v_{\text{άλογο}}(2) \\ \vdots \\ v_{\text{άλογο}}(300) \end{pmatrix}$$

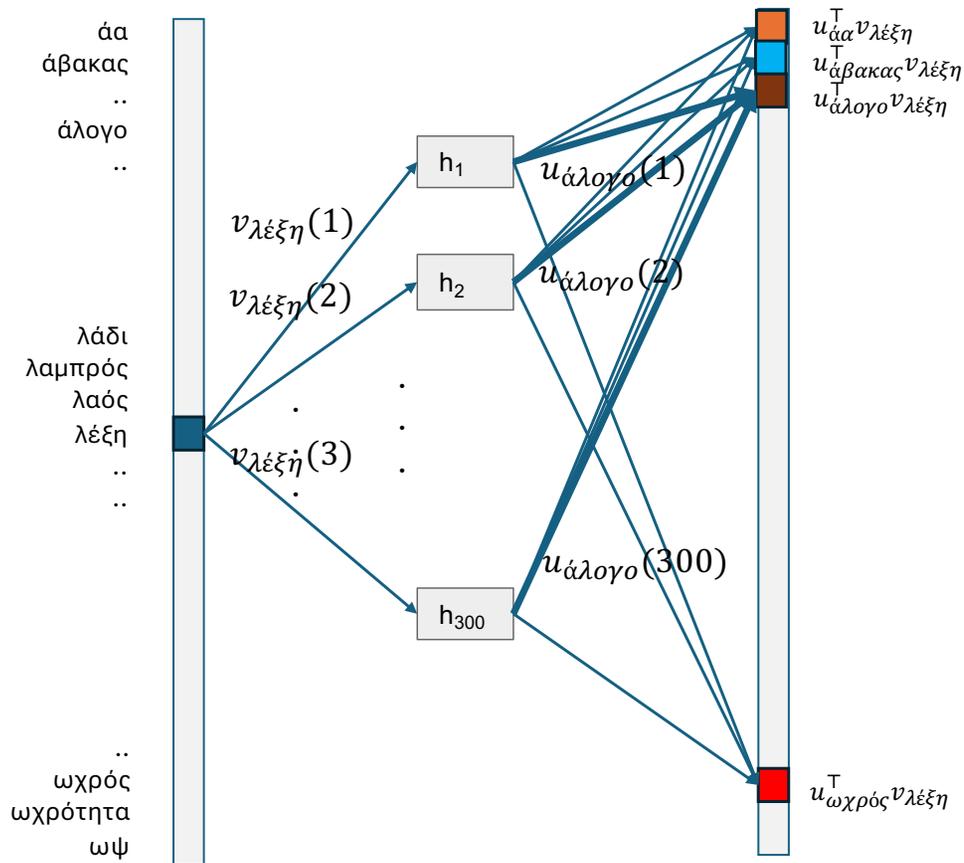
Ενσωμάτωση (embedding) ως δεύτερη (ακόλουθη) λέξη σε 300 διαστάσεις.

Νευρωνικό Δίκτυο για το Word2Vec



Κωνσταντίνος Καραμανής

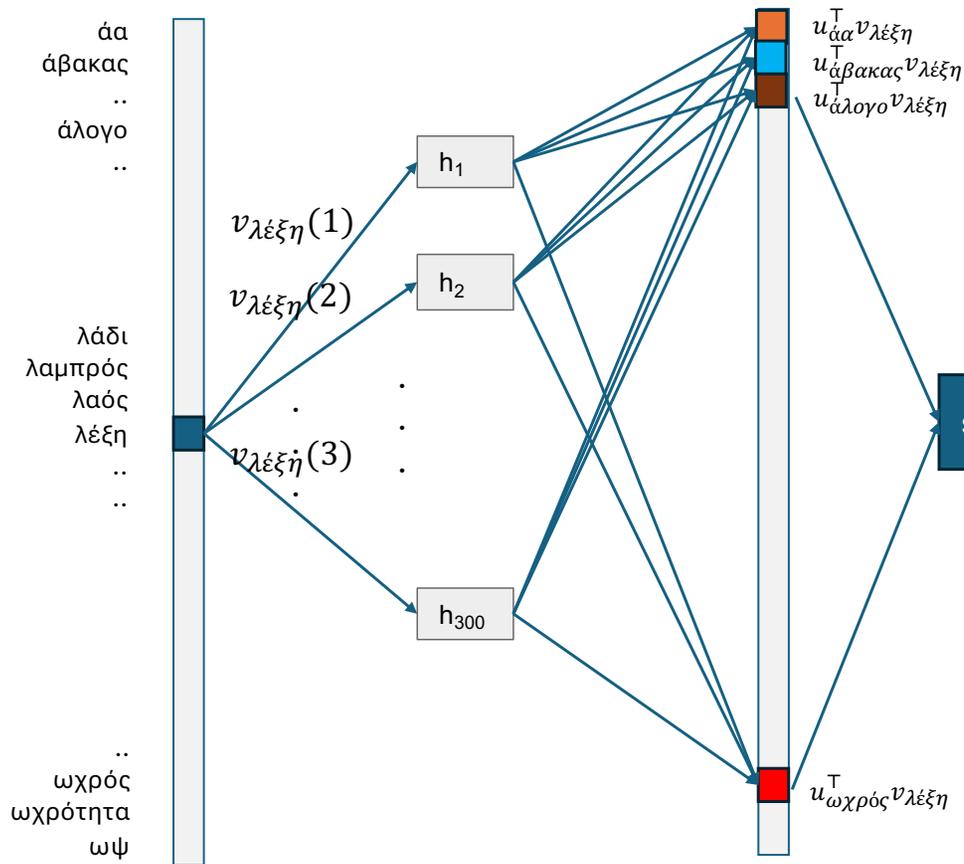
Νευρωνικό Δίκτυο για το Word2Vec



$$v_{\alpha\lambda\omicron\gamma\omicron} = \begin{pmatrix} v_{\alpha\lambda\omicron\gamma\omicron}(1) \\ v_{\alpha\lambda\omicron\gamma\omicron}(2) \\ \vdots \\ v_{\alpha\lambda\omicron\gamma\omicron}(300) \end{pmatrix}$$

Ενσωμάτωση (embedding) ως πρώτη (προηγούμενη) λέξη, πάλι σε 300 διαστάσεις

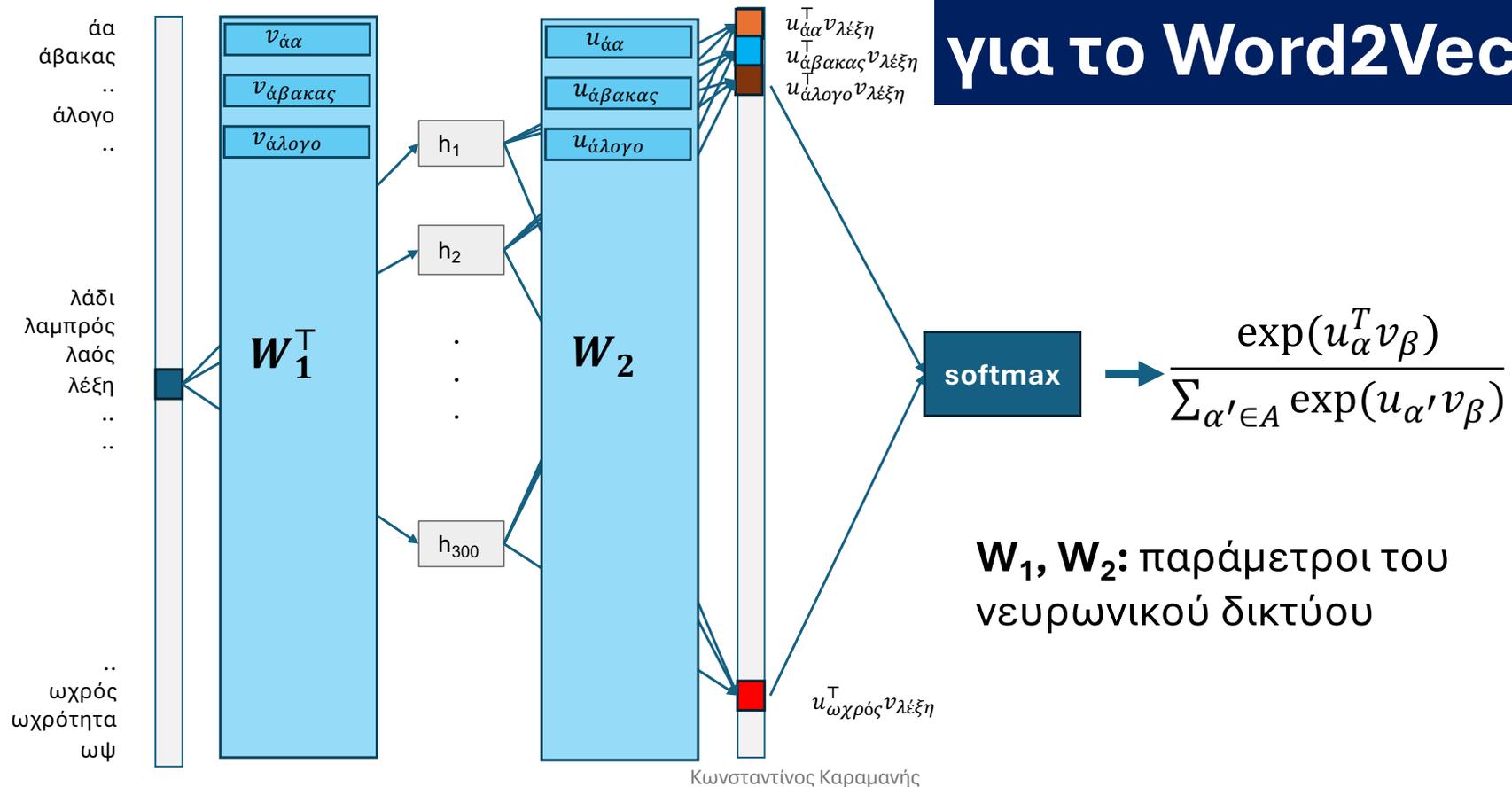
Νευρωνικό Δίκτυο για το Word2Vec



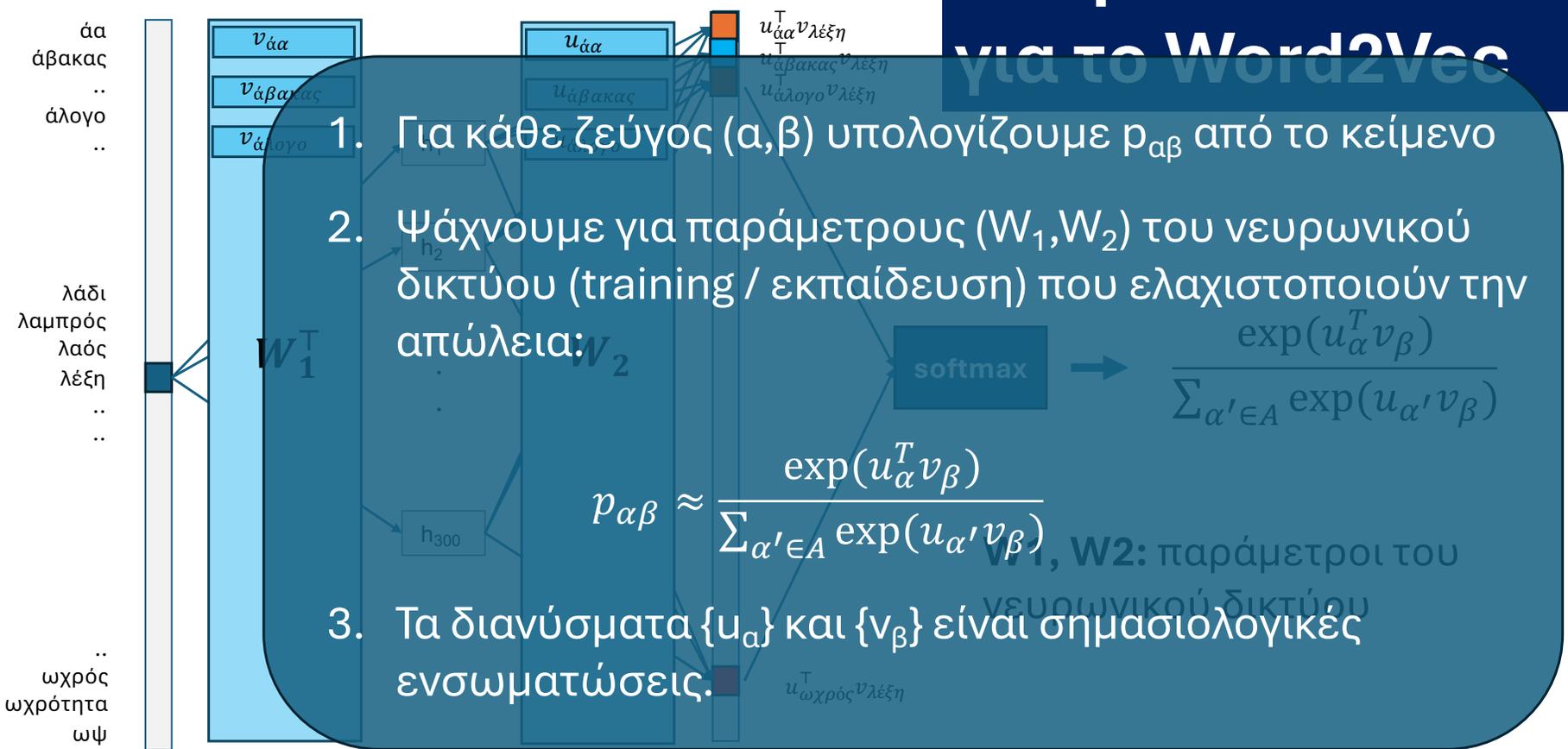
softmax

$$\frac{\exp(u_{\alpha}^T v_{\beta})}{\sum_{\alpha' \in A} \exp(u_{\alpha'}^T v_{\beta})}$$

Νευρωνικό Δίκτυο για το Word2Vec



Νευρωνικό Δίκτυο για το Word2Vec



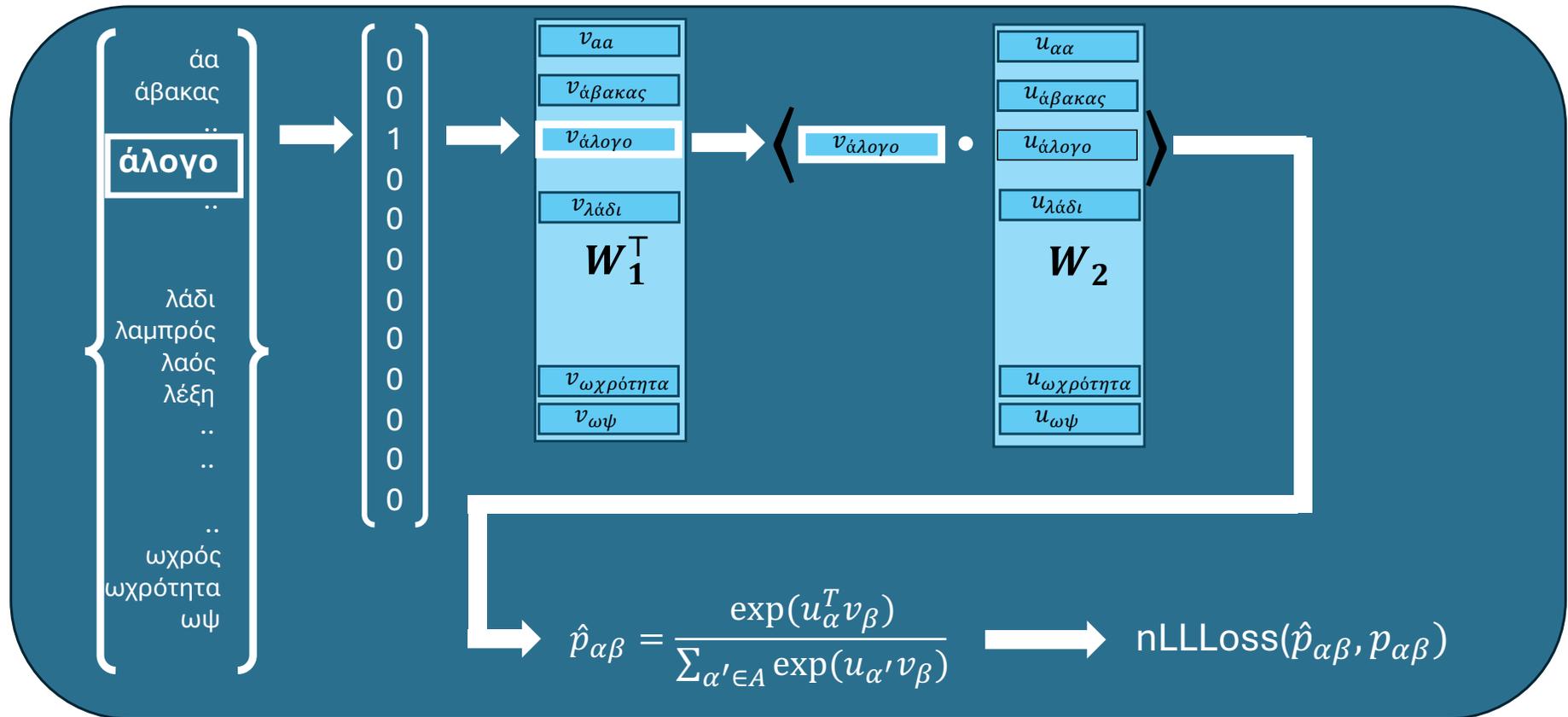
Word2Vec: Πώς Γίνονται οι Υπολογισμοί

άα
άβακας
..
άλογο
..

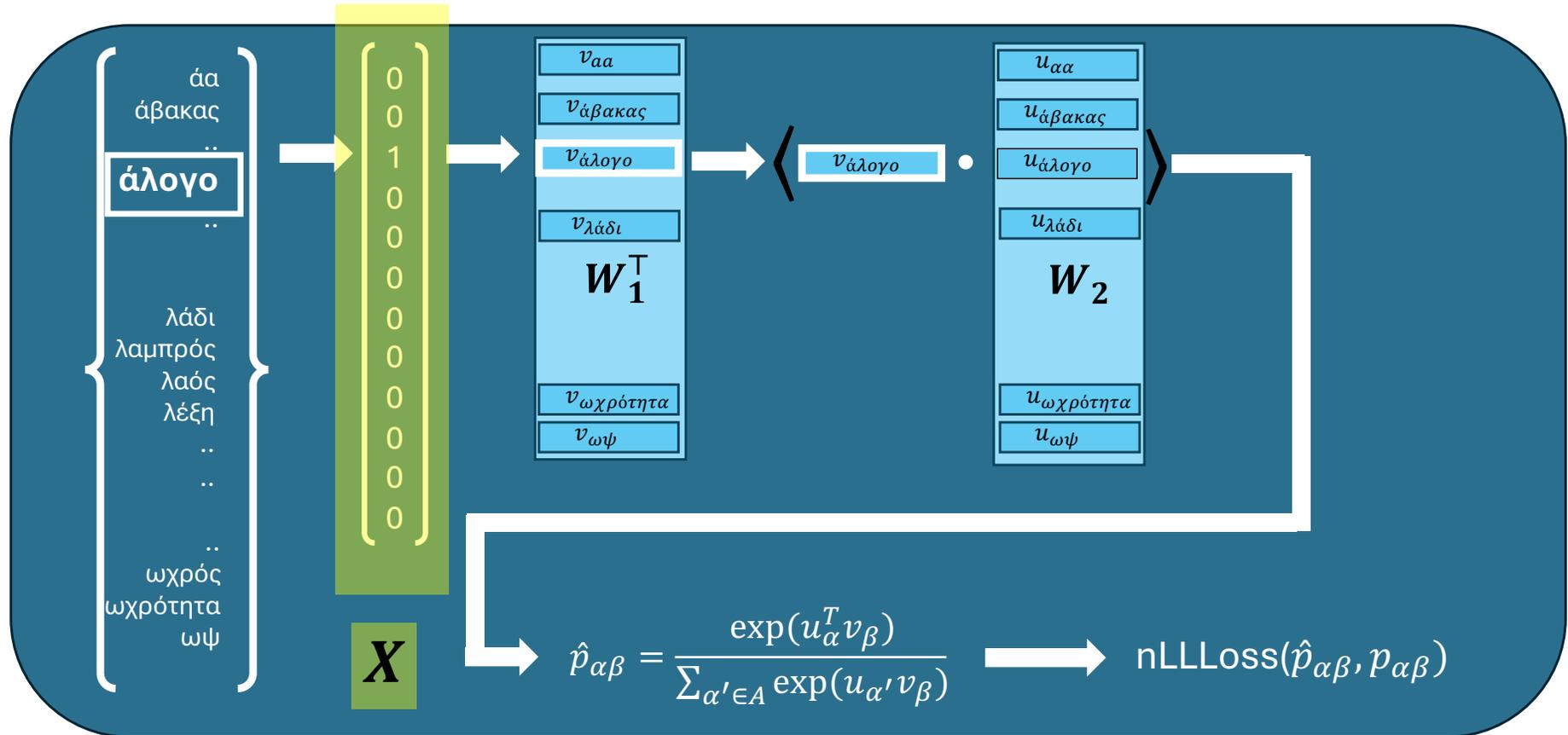
λάδι
λαμπρός
λαός
λέξη
..
..

..
ωχρός
ωχρότητα
ωψ

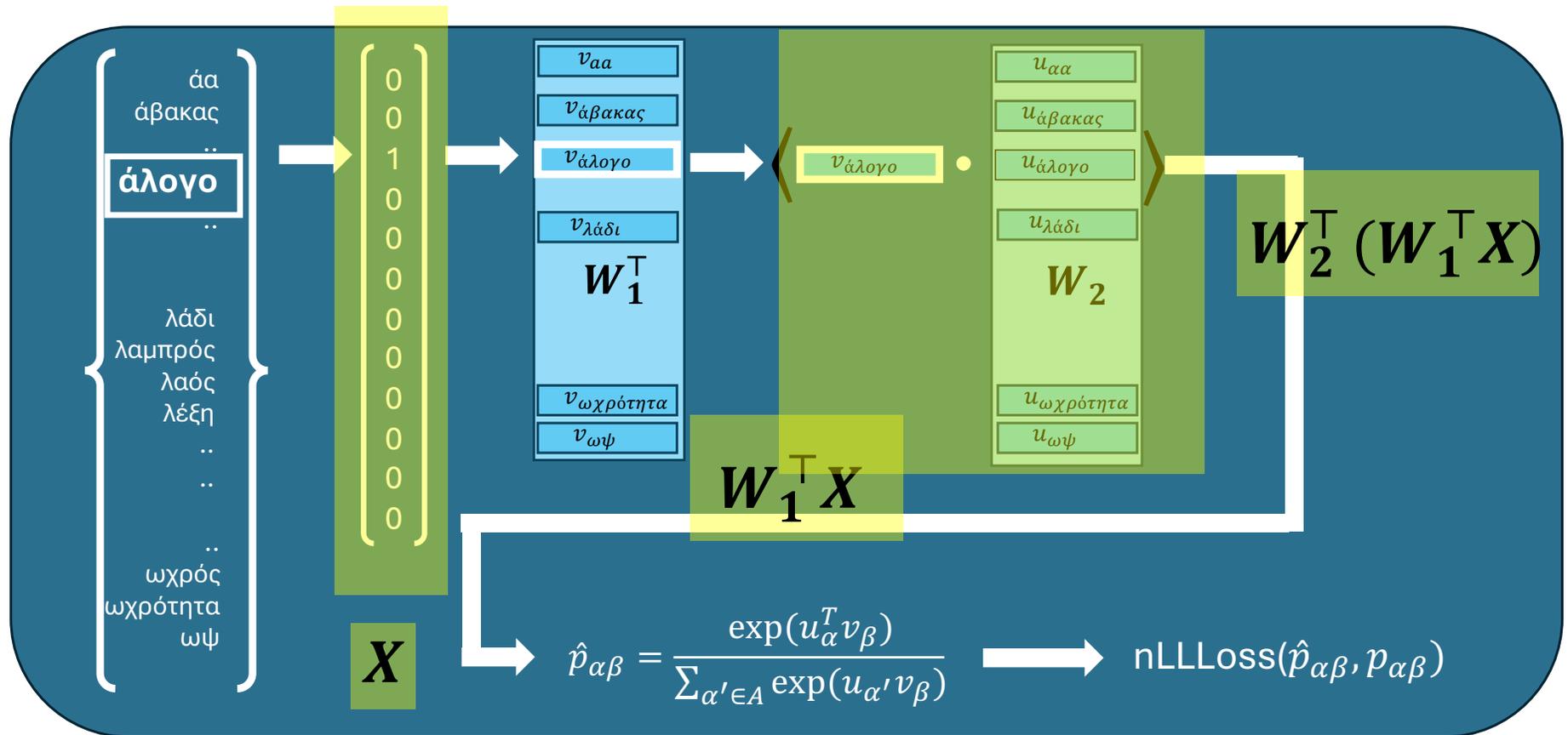
Word2Vec: Πώς Γίνονται οι Υπολογισμοί



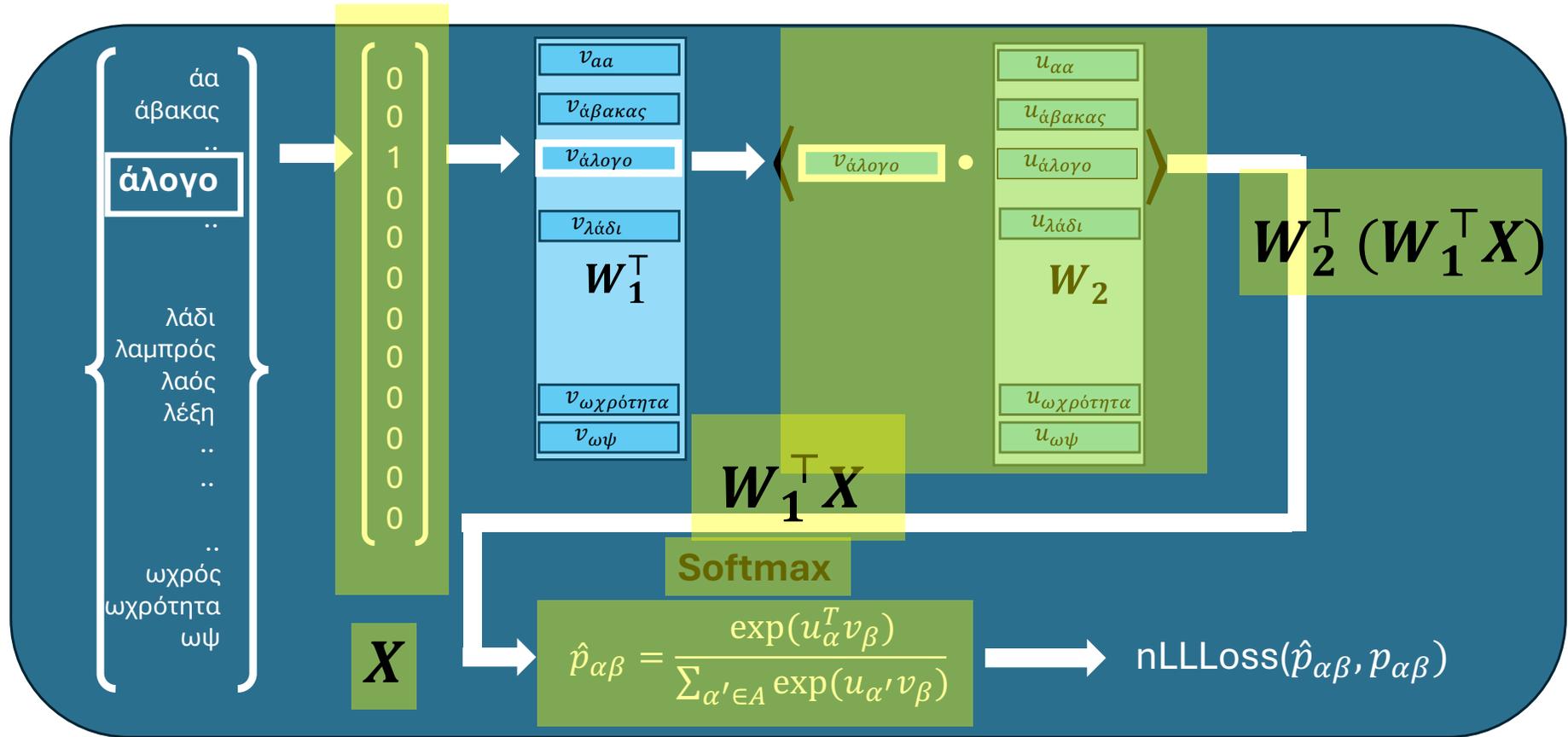
Word2Vec: Πώς Γίνονται οι Υπολογισμοί



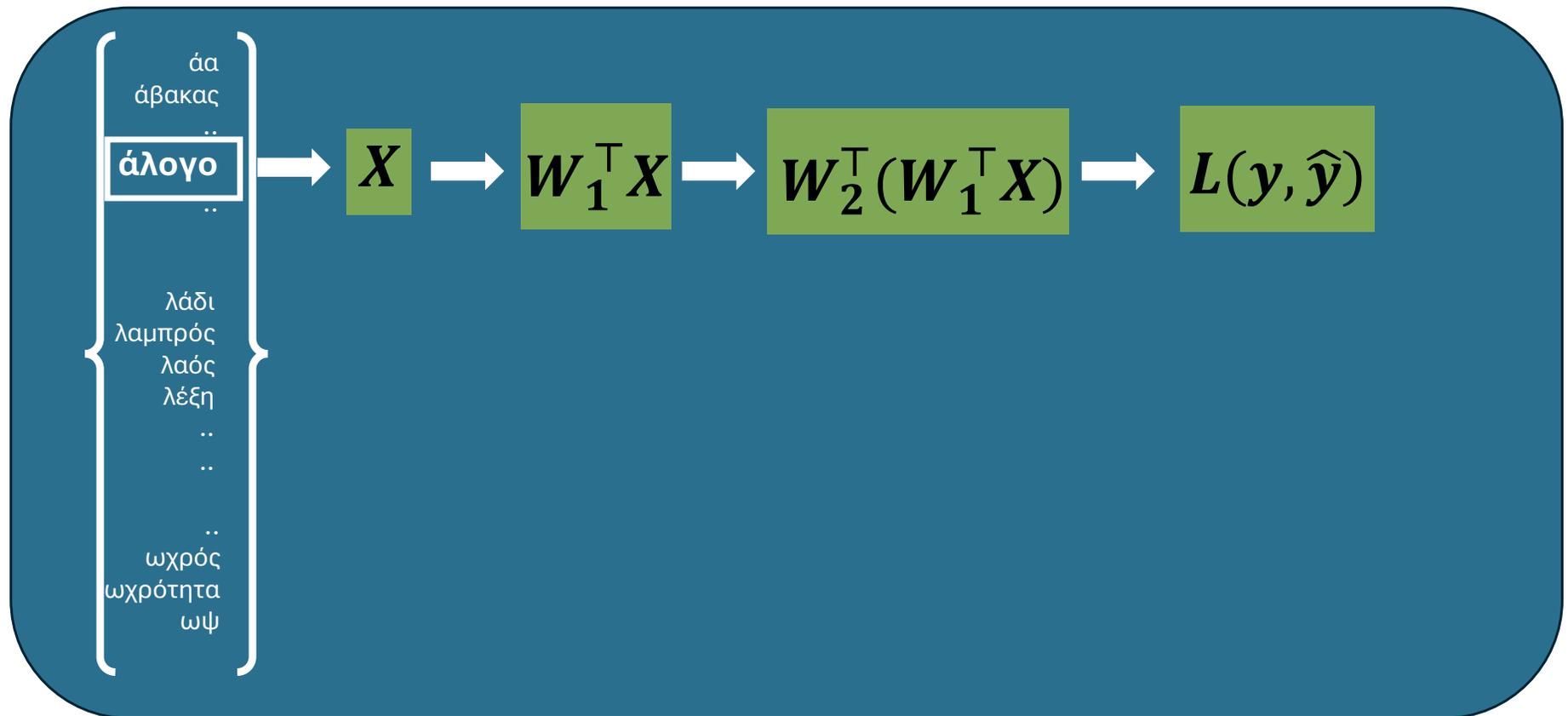
Word2Vec: Πώς Γίνονται οι Υπολογισμοί



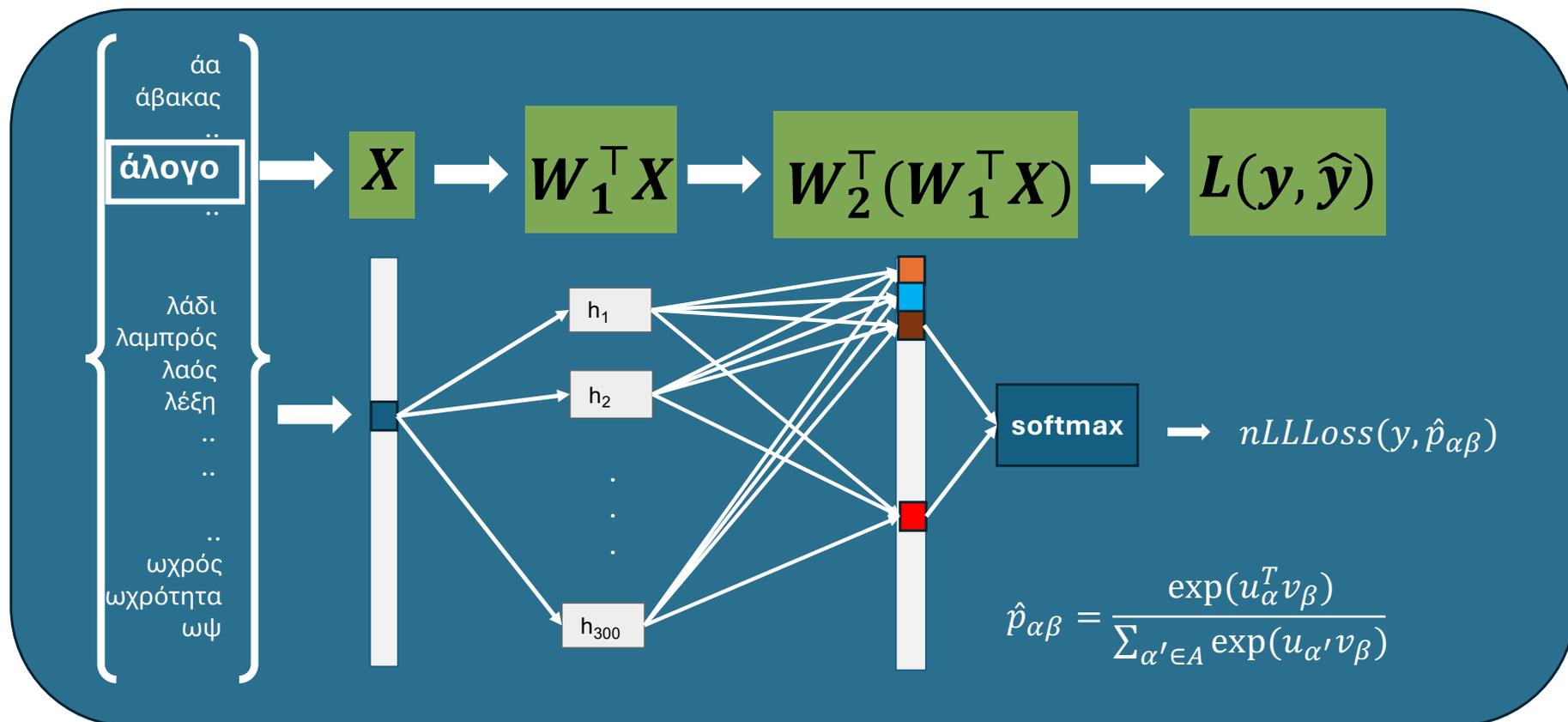
Word2Vec: Πώς Γίνονται οι Υπολογισμοί



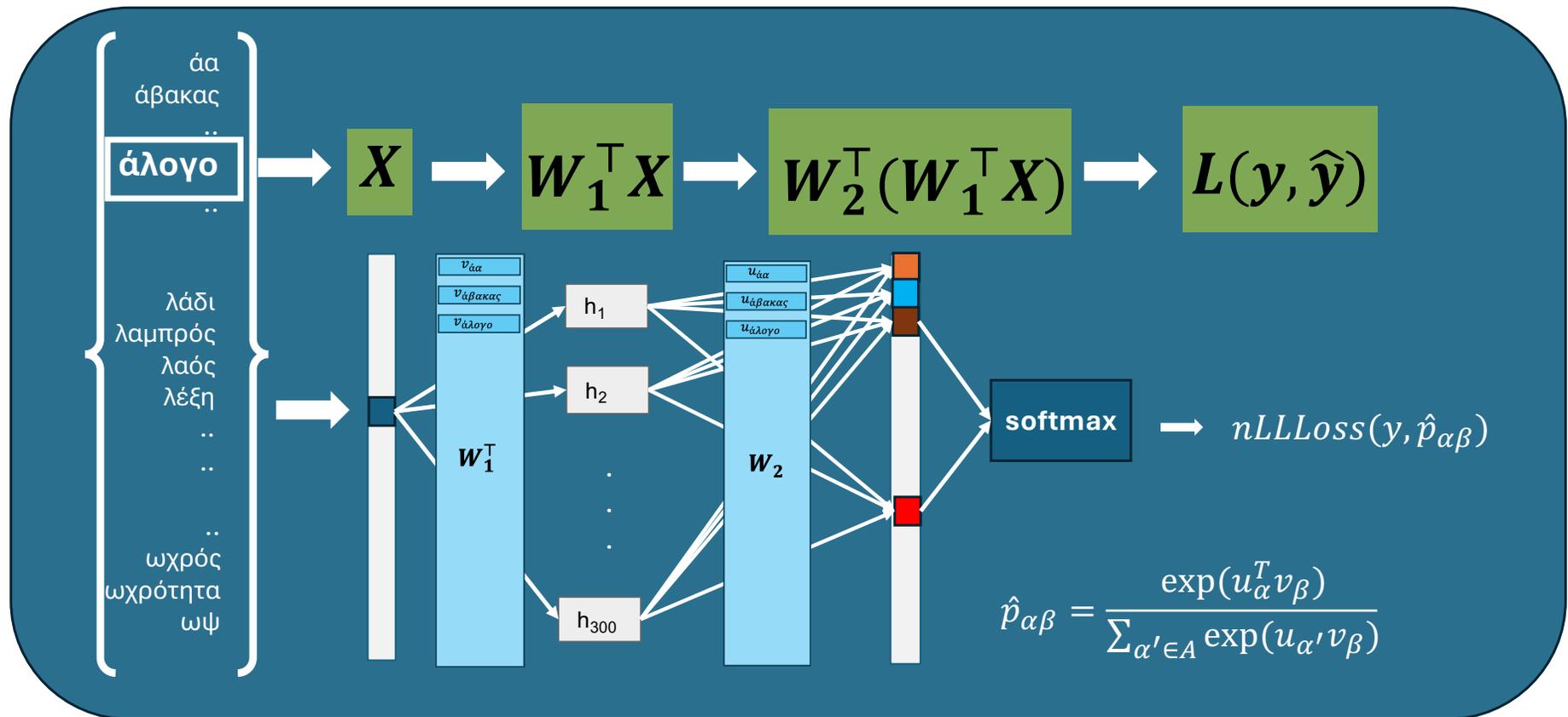
Word2Vec: Πώς Γίνονται οι Υπολογισμοί



Word2Vec: Πώς Γίνονται οι Υπολογισμοί



Word2Vec: Πώς Γίνονται οι Υπολογισμοί



Word2Vec: Demo

```
# Test word embedding
vector = model["king"] # Get embedding for "king"
print("Vector size:", len(vector))
```

Vector size: 300

Word2Vec: Demo

```
# Test word embedding
vector = model["king"] # Get embedding for "king"
print("Vector size:", len(vector))
```

Vector size: 300

```
# Find most similar words to "Parthenon"
for word in model.most_similar("Parthenon"):
    print(word)
```

```
('Acropolis', 0.5453481674194336)
('metopes', 0.5171760320663452)
('Parthenon_temple', 0.5158255100250244)
('Propylaea', 0.5151312351226807)
('Elgin_marbles', 0.4917999804019928)
('friezes', 0.4915829598903656)
('ancient_Athenian', 0.4904763996601105)
('Pergamon', 0.48820197582244873)
('Elgin_Marbles', 0.4869332015514374)
('Acropolis_hill', 0.484723836183548)
```

Κωνσταντίνος Καραμανής

Word2Vec: Demo

```
# Test word embedding
vector = model["king"] # Get embedding for "king"
print("Vector size:", len(vector))
```

Vector size: 300

```
# Find most similar words to "Parthenon"
for word in model.most_similar("Parthenon"):
    print(word)
```

```
('Acropolis', 0.5453481674194336) ←
('metopes', 0.5171760320663452)
('Parthenon_temple', 0.5158255100250244)
('Propylaea', 0.5151312351226807)
('Elgin_marbles', 0.4917999804019928)
('friezes', 0.4915829598903656)
('ancient_Athenian', 0.4904763996601105)
('Pergamon', 0.48820197582244873)
('Elgin_Marbles', 0.4869332015514374)
('Acropolis_hill', 0.484723836183548)
```

Συνημίτονο της γωνίας: $\angle u_{acropolis} u_{parthenon}$

Μεγαλύτερο συνημίτονο → μικρότερη γωνία
→ Πιο κοντά οι ενσωματώσεις

Word2Vec: οι βασικές ιδέες

Σημασιολογικές ενσωματώσεις λέξεων

Self-supervision: αξιοποιούμε μια σχεδόν ανεξάντλητη πηγή -- το κείμενο στο διαδίκτυο

Εκπαιδεύουμε ένα νευρωνικό δίκτυο που μας δίνει τις σημασιολογικές ενσωματώσεις κάθε λέξης

