AI Applications Lecture 7

Embedding Layer and Distances/Similarities

SUZUKI, Atsushi
Jing WANG

Outline

Introduction

1.1 Review of the Previous Lecture

In the previous lecture, we focused on the **token generator**.

1.1 Review of the Previous Lecture

In the previous lecture, we focused on the **token generator**.

We learned about **sampling**, which maps the **continuous output of a neural network** (like a probability distribution for the next token) to a **discrete object in natural language** (a specific token).

1.1 Review of the Previous Lecture

In the previous lecture, we focused on the **token generator**.

We learned about **sampling**, which maps the **continuous output of a neural network** (like a probability distribution for the next token) to a **discrete object in natural language** (a specific token).

In this lecture, we reverse the perspective.

1.1 Review of the Previous Lecture

In the previous lecture, we focused on the **token generator**.

We learned about **sampling**, which maps the **continuous output of a neural network** (like a probability distribution for the next token) to a **discrete object in natural language** (a specific token).

In this lecture, we reverse the perspective.

We will make explicit the fact that **when using a neural network as the core of a token generator, it implicitly converts discrete objects (tokens) into continuous objects (real-valued vectors)**.

1.1 Review of the Previous Lecture

In the previous lecture, we focused on the **token generator**.

We learned about **sampling**, which maps the **continuous output of a neural network** (like a probability distribution for the next token) to a **discrete object in natural language** (a specific token).

In this lecture, we reverse the perspective.

We will make explicit the fact that **when using a neural network as the core of a token generator, it implicitly converts discrete objects (tokens) into continuous objects (real-valued vectors)**.

The component responsible for this is the **embedding layer**.

1.2 Learning Outcomes for This Lecture

By the end of this lecture, you should be able to:

- Explain **what** an embedding layer does.

1.2 Learning Outcomes for This Lecture

By the end of this lecture, you should be able to:

- Explain **what** an embedding layer does.
- **Calculate** the **distances and similarities** (Euclidean distance, standard inner product, cosine similarity) between token representations (vectors) obtained through embedding.

Preliminaries: Mathematical Notations

2. Preliminaries: Mathematical Notations

Set & Function:

- Sets: $\mathcal{A}$
- Membership: $x \in \mathcal{A}$
- Integer range: $[1, k]_{\mathbb{Z}} := \{1, \ldots, k\}$
- Real numbers: $\mathbb{R}, \mathbb{R}_{>0}, \mathbb{R}_{\geq 0}$
- Function: $f : \mathcal{X} \to \mathcal{Y}$

Vector: Denoted by $\boldsymbol{v}$.

- A column of numbers, $\boldsymbol{v} \in \mathbb{R}^n$.
- i-th element is v_i.
- Standard inner product: $\langle \boldsymbol{u}, \boldsymbol{v} \rangle = \boldsymbol{u}^\top \boldsymbol{v}$

2. Preliminaries: Mathematical Notations

Sequence: Denoted by $\boldsymbol{a} = (a_1, a_2, \dots)$.

- A function $\boldsymbol{a} : [1, n]_{\mathbb{Z}} \to \mathcal{A}$.
- Length is denoted by $|\boldsymbol{a}|$.

Matrix: Denoted by $\boldsymbol{A}$.

- $m \times n$ matrix: $\boldsymbol{A} \in \mathbb{R}^{m,n}$.
- (i, j)-th element is $a_{i,j}$.

Tensor: Denoted by $\underline{\boldsymbol{A}}$.

- Simply a multi-dimensional array.
- Vector $\to$ 1st-order, Matrix $\to$ 2nd-order.

One-Hot Encoding

3.1 Motivation

The **appropriate distance relationships between tokens** in natural language are not known in advance.

3.1 Motivation

The **appropriate distance relationships between tokens** in natural language are not known in advance.

Therefore, to **treat all tokens symmetrically (equally)**, we use the simplest and most neutral representation, the **one-hot vector**.

Definition (One-Hot Encoding)

Consider a finite set $\mathcal{V} = \{1, \ldots, d_{\mathrm{in}}\}$ with a vocabulary size of $d_{\mathrm{in}} \in \mathbb{Z}_{>0}$. For $i \in \mathcal{V}$, the one-hot vector $\mathrm{e}_i \in \{0,1\}^{d_{\mathrm{in}}}$ is defined as:

$$(\mathrm{e}_i)_j := \begin{cases} 1 & (j = i) \\ 0 & (j \neq i) \end{cases} \quad (j \in [1, d_{\mathrm{in}}]_{\mathbb{Z}}). \tag{1}$$

3.1 Motivation

Remark

For any $i \neq j$, we have $\|\mathrm{e}_i - \mathrm{e}_j\|_2 = \sqrt{2}$.

Remark

For any $i \neq j$, we have $\|\mathrm{e}_i - \mathrm{e}_j\|_2 = \sqrt{2}$. This expresses neutrality (lack of prior assumptions) by ensuring that the **distance between all pairs of tokens is equal**.

Mapping to the Next Layer and the Emergence of Embedding

4.1 Symmetry and Fully-Connected Layer

When the **input is one-hot**, we have no choice but to treat **each input node equally.** The natural connection to the next set of nodes is a **fully-connected** layer.

When the **input is one-hot**, we have no choice but to treat **each input node equally.** The natural connection to the next set of nodes is a **fully-connected** layer.

This part of the network is called an **embedding layer**.

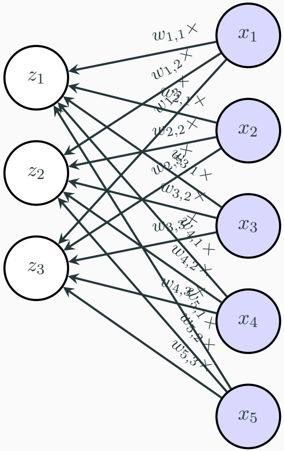

Figure 1: A fully-connected layer.

4.2 Matrix Representation and Equivalence to Linear Regression

A fully-connected layer is a **linear map** and can be written using a matrix $\boldsymbol{W} \in \mathbb{R}^{d_{\text{emb}}, d_{\text{in}}}$:

$$\boldsymbol{z} := \boldsymbol{W}\,\boldsymbol{x} \quad (\boldsymbol{x} \in \mathbb{R}^{d_{\text{in}}},\ \boldsymbol{z} \in \mathbb{R}^{d_{\text{emb}}}). \tag{2}$$

4.3 Active Edges for a One-Hot Input

When the input is a one-hot vector,
$\boldsymbol{x} = \mathrm{e}_i$, something interesting happens.

4.3 Active Edges for a One-Hot Input

When the input is a one-hot vector, $\boldsymbol{x} = \mathrm{e}_i$, something interesting happens. The multiplication $\boldsymbol{W}\,\mathrm{e}_i$ simply selects the i**-th column of the matrix** $\boldsymbol{W}$.

$$\boldsymbol{W}\,\mathrm{e}_i = \boldsymbol{w}_i \tag{3}$$

(where $\boldsymbol{w}_i$ is the i-th column of $\boldsymbol{W}$).

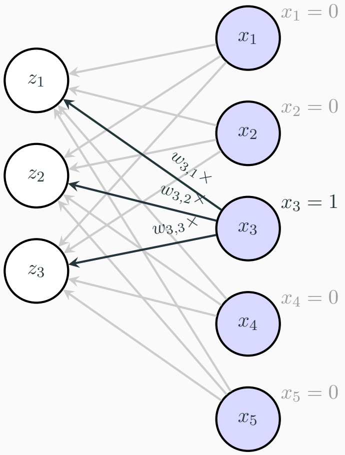

Figure 2: Active edges for input e_3.

4.4 Equivalent Understanding of Embedding

This means:

- **"Feeding a one-hot vector e_i into the first fully-connected layer"...**

4.4 Equivalent Understanding of Embedding

This means:

- "**Feeding a one-hot vector e_i into the first fully-connected layer**"...

...is **equivalent** to...

- "**Directly looking up the column vector w_i in the weight matrix**".

4.4 Equivalent Understanding of Embedding

This means:

- "**Feeding a one-hot vector e_i into the first fully-connected layer**"...

...is **equivalent** to...

- "**Directly looking up the column vector w_i in the weight matrix**".

The column vector $\boldsymbol{w}_i$ is a **learnable parameter** vector that represents token i.

4.4 Equivalent Understanding of Embedding

Definition (Embedding)

The map defined by each column $\boldsymbol{w}_i$ of the matrix $\boldsymbol{W} \in \mathbb{R}^{d_{\mathrm{emb}},\, d_{\mathrm{in}}}$,

$$\iota : \ i \in \{1, \ldots, d_{\mathrm{in}}\} \mapsto \boldsymbol{w}_i \in \mathbb{R}^{d_{\mathrm{emb}}} \tag{4}$$

is called an **embedding** or **embedding representation** [1, 2, 3].

Remark

It is customary to call w_i the **representation** of the i-th token or the **embedding** of the i-th token.

4.4 Equivalent Understanding of Embedding

Remark

It is customary to call w_i the **representation** of the i-th token or the **embedding** of the i-th token.

Mathematically, an "embedding" is an **injective map** that preserves structure.

4.4 Equivalent Understanding of Embedding

Remark

It is customary to call w_i the **representation** of the i-th token or the **embedding** of the i-th token.

Mathematically, an "embedding" is an **injective map** that preserves structure.

This name is used with the expectation that the relationships between tokens will be **reflected** in the relationships between their vector representations in the new space.

4.5 Vocabulary Extension (Adding Columns)

How do we add new tokens to our vocabulary?

4.5 Vocabulary Extension (Adding Columns)

How do we add new tokens to our vocabulary?

- We increase the dimension of the one-hot vectors (i.e., increase d_{in}).

4.5 Vocabulary Extension (Adding Columns)

How do we add new tokens to our vocabulary?

- We increase the dimension of the one-hot vectors (i.e., increase d_{in}).
- In the matrix representation, we simply **add new column vectors to W**.

4.5 Vocabulary Extension (Adding Columns)

How do we add new tokens to our vocabulary?

- We increase the dimension of the one-hot vectors (i.e., increase d_{in}).
- In the matrix representation, we simply **add new column vectors to $\boldsymbol{W}$**.
- It is possible to **train only the new columns**, or fine-tune the entire matrix.

Distances and Similarities in the Embedding Space

5. Distances and Similarities in the Embedding Space

5. Distances and Similarities in the Embedding Space

If the embedding reflects semantic information, calculating the **relationships between vectors** can have practical applications [2, 1].

5. Distances and Similarities in the Embedding Space

If the embedding reflects semantic information, calculating the **relationships between vectors** can have practical applications [2, 1].

Here, we will strictly define the most fundamental measures.

Definition (Euclidean Distance)

For $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^{d_{\mathrm{emb}}}$, the **Euclidean distance** $d_{\mathrm{E}}(\boldsymbol{u}, \boldsymbol{v})$ is defined as:

$$d_{\mathrm{E}}(\boldsymbol{u}, \boldsymbol{v}) := \|\boldsymbol{u} - \boldsymbol{v}\|_2 = \sqrt{(\boldsymbol{u} - \boldsymbol{v})^\top (\boldsymbol{u} - \boldsymbol{v})}. \tag{5}$$

5. Distances and Similarities in the Embedding Space

Definition (Euclidean Distance)

For $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^{d_{\text{emb}}}$, the **Euclidean distance** $d_{\mathrm{E}}(\boldsymbol{u}, \boldsymbol{v})$ is defined as:

$$d_{\mathrm{E}}(\boldsymbol{u}, \boldsymbol{v}) := \|\boldsymbol{u} - \boldsymbol{v}\|_2 = \sqrt{(\boldsymbol{u} - \boldsymbol{v})^\top (\boldsymbol{u} - \boldsymbol{v})}. \tag{5}$$

The **smaller** the distance, the more similar the concepts.

Definition (Standard Inner Product)

For $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^{d_{\text{emb}}}$, the **Standard Inner Product** $\langle \boldsymbol{u}, \boldsymbol{v} \rangle$ is defined as:

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle := \boldsymbol{u}^\top \boldsymbol{v} = \sum_{k=1}^{d_{\text{emb}}} u_k v_k. \tag{6}$$

Definition (Standard Inner Product)

For $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^{d_{\mathrm{emb}}}$, the **Standard Inner Product** $\langle \boldsymbol{u}, \boldsymbol{v} \rangle$ is defined as:

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle := \boldsymbol{u}^\top \boldsymbol{v} = \sum_{k=1}^{d_{\mathrm{emb}}} u_k v_k. \tag{6}$$

The **larger** the inner product, the more similar the concepts. The Transformer architecture [4] uses this measure extensively.

Definition (Cosine Similarity)

For $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^{d_{\mathrm{emb}}} \setminus \{\mathbf{0}\}$, the **cosine similarity** $\cos(\boldsymbol{u}, \boldsymbol{v})$ is defined as:

$$\cos(\boldsymbol{u}, \boldsymbol{v}) := \frac{\langle \boldsymbol{u}, \boldsymbol{v} \rangle}{\|\boldsymbol{u}\|_2 \, \|\boldsymbol{v}\|_2}. \tag{7}$$

Definition (Cosine Similarity)

For $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^{d_{\text{emb}}} \setminus \{\boldsymbol{0}\}$, the **cosine similarity** $\cos(\boldsymbol{u}, \boldsymbol{v})$ is defined as:

$$\cos(\boldsymbol{u}, \boldsymbol{v}) := \frac{\langle \boldsymbol{u}, \boldsymbol{v} \rangle}{\|\boldsymbol{u}\|_2 \, \|\boldsymbol{v}\|_2}. \tag{7}$$

This is the cosine of the angle between the two vectors. The **larger** the similarity (closer to 1), the more similar the concepts.

The following identity holds:

$$\begin{aligned} \|\boldsymbol{u} - \boldsymbol{v}\|_2^2 &= \|\boldsymbol{u}\|_2^2 + \|\boldsymbol{v}\|_2^2 - 2\langle \boldsymbol{u}, \boldsymbol{v} \rangle \\ &= \|\boldsymbol{u}\|_2^2 + \|\boldsymbol{v}\|_2^2 \\ &\quad - 2\|\boldsymbol{u}\|_2 \|\boldsymbol{v}\|_2 \cos(\boldsymbol{u}, \boldsymbol{v}) \end{aligned} \tag{8}$$

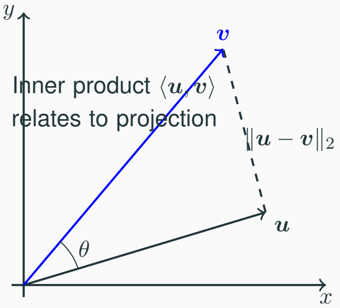

Figure 3: Geometry of measures.

Example (Euclidean Distance)

For $\boldsymbol{u} = \begin{bmatrix} 2 \\ -1 \\ 3 \end{bmatrix}$ and $\boldsymbol{v} = \begin{bmatrix} 1 \\ 2 \\ -1 \end{bmatrix}$, find the Euclidean distance.

Step 1: Calculate the difference vector

$$\boldsymbol{u} - \boldsymbol{v} = \begin{bmatrix} 2 - 1 \\ -1 - 2 \\ 3 - (-1) \end{bmatrix} = \begin{bmatrix} 1 \\ -3 \\ 4 \end{bmatrix} \tag{9}$$

Step 2: Calculate the L2 norm of the difference

$$\|\boldsymbol{u} - \boldsymbol{v}\|_2 = \sqrt{1^2 + (-3)^2 + 4^2} \tag{10}$$
$$= \sqrt{1 + 9 + 16} = \sqrt{26}. \tag{11}$$

The Euclidean distance is $\sqrt{26}$.

Exercise

Find the Euclidean distance $d_{\mathrm{E}}(\boldsymbol{a}, \boldsymbol{b})$ for $\boldsymbol{a} = \begin{bmatrix} -1 \\ 4 \\ 2 \end{bmatrix}$ and $\boldsymbol{b} = \begin{bmatrix} 3 \\ 0 \\ -2 \end{bmatrix}$.

Answer

The difference vector is:

$$\boldsymbol{a} - \boldsymbol{b} = \begin{bmatrix} -1 - 3 \\ 4 - 0 \\ 2 - (-2) \end{bmatrix} = \begin{bmatrix} -4 \\ 4 \\ 4 \end{bmatrix} \tag{12}$$

Answer

The difference vector is:

$$\boldsymbol{a} - \boldsymbol{b} = \begin{bmatrix} -1 - 3 \\ 4 - 0 \\ 2 - (-2) \end{bmatrix} = \begin{bmatrix} -4 \\ 4 \\ 4 \end{bmatrix} \tag{12}$$

The Euclidean distance is:

$$\|\boldsymbol{a} - \boldsymbol{b}\|_2 = \sqrt{(-4)^2 + 4^2 + 4^2} = \sqrt{16 + 16 + 16} = \sqrt{48} = 4\sqrt{3}. \tag{13}$$

Example (Inner Product)

For $\boldsymbol{u} = \begin{bmatrix} 2 \\ -1 \\ 3 \end{bmatrix}$ and $\boldsymbol{v} = \begin{bmatrix} 1 \\ 2 \\ -1 \end{bmatrix}$, find the inner product.

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle = 2 \cdot 1 + (-1) \cdot 2 + 3 \cdot (-1) \tag{14}$$

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle = 2 \cdot 1 + (-1) \cdot 2 + 3 \cdot (-1) \tag{14}$$

$$= 2 - 2 - 3 = -3. \tag{15}$$

The inner product is -3.

Exercise

Find the inner product $\langle \boldsymbol{a}, \boldsymbol{b} \rangle$ for $\boldsymbol{a} = \begin{bmatrix} -1 \\ 4 \\ 2 \end{bmatrix}$ and $\boldsymbol{b} = \begin{bmatrix} 3 \\ 0 \\ -2 \end{bmatrix}$.

Answer

$$\langle \boldsymbol{a}, \boldsymbol{b} \rangle = (-1) \cdot 3 + 4 \cdot 0 + 2 \cdot (-2) \tag{16}$$

Answer

$$\langle \boldsymbol{a}, \boldsymbol{b} \rangle = (-1) \cdot 3 + 4 \cdot 0 + 2 \cdot (-2) \tag{16}$$

$$= -3 + 0 - 4 = -7. \tag{17}$$

Example (Cosine Similarity)

For $\boldsymbol{u} = \begin{bmatrix} 2 \\ -1 \\ 3 \end{bmatrix}$ and $\boldsymbol{v} = \begin{bmatrix} 1 \\ 2 \\ -1 \end{bmatrix}$, find the cosine similarity.

Step 1: Reuse the inner product

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle = -3 \quad \text{(from previous example)} \tag{18}$$

Step 2: Calculate the norms

$$\|\boldsymbol{u}\|_2 = \sqrt{2^2 + (-1)^2 + 3^2} = \sqrt{4 + 1 + 9} = \sqrt{14} \tag{19}$$

$$\|\boldsymbol{v}\|_2 = \sqrt{1^2 + 2^2 + (-1)^2} = \sqrt{1 + 4 + 1} = \sqrt{6} \tag{20}$$

Step 3: Divide inner product by product of norms

$$\cos(\boldsymbol{u}, \boldsymbol{v}) = \frac{-3}{\sqrt{14}\,\sqrt{6}} = \frac{-3}{\sqrt{84}} = \frac{-3}{2\sqrt{21}}. \tag{21}$$

Exercise

Find the cosine similarity for $\boldsymbol{a} = \begin{bmatrix} -1 \\ 4 \\ 2 \end{bmatrix}$ and $\boldsymbol{b} = \begin{bmatrix} 3 \\ 0 \\ -2 \end{bmatrix}$.

Answer

$$\langle \boldsymbol{a}, \boldsymbol{b} \rangle = -7 \quad \text{(from previous exercise)} \tag{22}$$

$$\|\boldsymbol{a}\|_2 = \sqrt{(-1)^2 + 4^2 + 2^2} = \sqrt{1 + 16 + 4} = \sqrt{21} \tag{23}$$

$$\|\boldsymbol{b}\|_2 = \sqrt{3^2 + 0^2 + (-2)^2} = \sqrt{9 + 0 + 4} = \sqrt{13} \tag{24}$$

$$\cos(\boldsymbol{a}, \boldsymbol{b}) = \frac{-7}{\sqrt{21}\sqrt{13}} = \frac{-7}{\sqrt{273}}. \tag{25}$$

Summary

6. Summary

Let's summarize the key takeaways from today's lecture.

- An **embedding layer** is a fully-connected layer that takes a one-hot vector as input. This is equivalent to a lookup operation, where the i-th token is mapped to the i-th column vector $\boldsymbol{w}_i$ of the layer's weight matrix.

6. Summary

Let's summarize the key takeaways from today's lecture.

- An **embedding layer** is a fully-connected layer that takes a one-hot vector as input. This is equivalent to a lookup operation, where the i-th token is mapped to the i-th column vector $\boldsymbol{w}_i$ of the layer's weight matrix.
- This vector $\boldsymbol{w}_i$ is called the **representation** or **embedding** of token i. It's a dense, continuous vector in a lower-dimensional space.

6. Summary

Let's summarize the key takeaways from today's lecture.

- An **embedding layer** is a fully-connected layer that takes a one-hot vector as input. This is equivalent to a lookup operation, where the i-th token is mapped to the i-th column vector $\boldsymbol{w}_i$ of the layer's weight matrix.
- This vector $\boldsymbol{w}_i$ is called the **representation** or **embedding** of token i. It's a dense, continuous vector in a lower-dimensional space.
- We defined three key measures to quantify relationships between embeddings: **Euclidean distance**, **standard inner product**, and **cosine similarity**. These allow us to capture the notion of "similarity" in the embedding space.

References i

[1] Ian Goodfellow, Yoshua Bengio, and Aaron Courville.
Deep Learning.
MIT Press, Cambridge, MA, 2016.

[2] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean.
Efficient estimation of word representations in vector space.
In Proceedings of Workshop at ICLR 2013, 2013.

[3] Jeffrey Pennington, Richard Socher, and Christopher D. Manning.
Glove: Global vectors for word representation.
In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1532–1543, 2014.

[4] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin.
Attention is all you need.
In Proceedings of NeurIPS 2017, 2017.