



# Sisteme de Recomandare

Laborator 4: Recomandări content-based folosind  
*cosine similarity* (29.10.2025)

Dr. ing. Ș.I. Gabriel Guțu-Robu  
gabriel.gutu@upb.ro

# About Cosine Similarity

# Cosine Similarity

- A measure used to determine **how similar two pieces of text are**, regardless of their length.
- It represents the *cosine of the angle between two vectors in a multi-dimensional space* — the closer the angle is to  $0^\circ$ , the more similar the texts are. The text is usually converted into numerical vectors (e.g., using term frequency or **TF-IDF**).
- In real applications, more advanced techniques are used, such as TF-IDF weighting, word embeddings (e.g., Word2Vec, GloVe), or transformer-based embeddings (BERT) to capture semantic similarity, **not just exact word matches**.

# Example

- Text A: “Artificial intelligence improves education.”
- Text B: “Artificial intelligence transforms education.”

## **1. Vocabulary**

Unique words:

[artificial, intelligence, improves, transforms,  
education]

# Example (Cont'd)

## 2. Create vectors

Word	Text A	Text B
artificial	1	1
intelligence	1	1
improves	1	0
transforms	0	1
education	1	1

Text A vector:  $[1, 1, 1, 0, 1]$

Text B vector:  $[1, 1, 0, 1, 1]$

## Example (Cont'd)

### 3. Compute cosine similarity

$$sim = \frac{A \cdot B}{\|A\| \times \|B\|}$$

$$\mathbf{A} \cdot \mathbf{B} = (1 \times 1) + (1 \times 1) + (1 \times 0) + (0 \times 1) + (1 \times 1) = 3$$

$$\|\mathbf{A}\| = \sqrt{1^2 + 1^2 + 1^2 + 0^2 + 1^2} = \sqrt{4} = 2$$

$$\|\mathbf{B}\| = \sqrt{1^2 + 1^2 + 0^2 + 1^2 + 1^2} = \sqrt{4} = 2$$

$$sim = \frac{3}{2 \times 2} = \frac{3}{4} = \mathbf{0.75}$$

# Example (Cont'd)

## 4. Interpretation

- A cosine similarity of **0.75** indicates a **high degree of similarity between the two sentences** — they share most of their key terms and differ by only one word (“improves” vs. “transforms”).
- If we used a more advanced representation like TF-IDF or word embeddings, the similarity would likely be **even higher**, since those methods capture **semantic similarity between related words** (which cosine similarity doesn't capture by its own).

Aplicație

# Calculul *Cosine Similarity*

**Scop:** Pentru laboratorul de azi veți calcula **cosine similarity** între două texte, ce pot fi **descrieri pentru diverse item-uri** (ex.: filme, cărți, etc.)

1. Pe Moodle veți găsi un **tutorial** pentru calculul *cosine similarity* în Python. Scrieți un script pornind de la acest tutorial. Puteți folosi și alt limbaj de programare.
2. Produsele voastre trebuie **să conțină texte** (ex.: descriere produse, sumar cărți, rezumat al unui film, etc.)
  - *Hint:* puteți folosiți fișierul `tesco_sample.json`, ce conține un dataset cu produse de la supermarket-urile Tesco. Fiecare produs are un câmp `description`, care conține text (și tag-uri HTML). Eliminați tag-urile HTML înainte de a prelucra textul.
3. Opțional: înainte de calcula *cosine similarity*, aplicați un *pipeline* de preprocesare de text (transformare cuvinte, lemma-tizare, eliminare *stopwords*, etc.)
4. După obținerea scorurilor, **evidențiați două produse asemănătoare**. Menționați-le într-un fișier `README.txt`, pe care îl veți urca pe Moodle alături de cod.

# Notare

În cadrul acestui laborator se notează următoarele:

1. **Selectie dataset** – ați selectat un dataset de items ce conțin un câmp cu text – **20p**
2. **Implementare cod** – ați adaptat script-ul din tutorial pe dataset-ul selectat – **30p**
3. **Scoruri similaritate** – ați obținut matricea de scoruri de similaritate – **30p**
4. **Evidențiere rezultate** – ați arătat un exemplu de similaritate între două produse, iar această pereche are sens (menționați în README) – **20p**