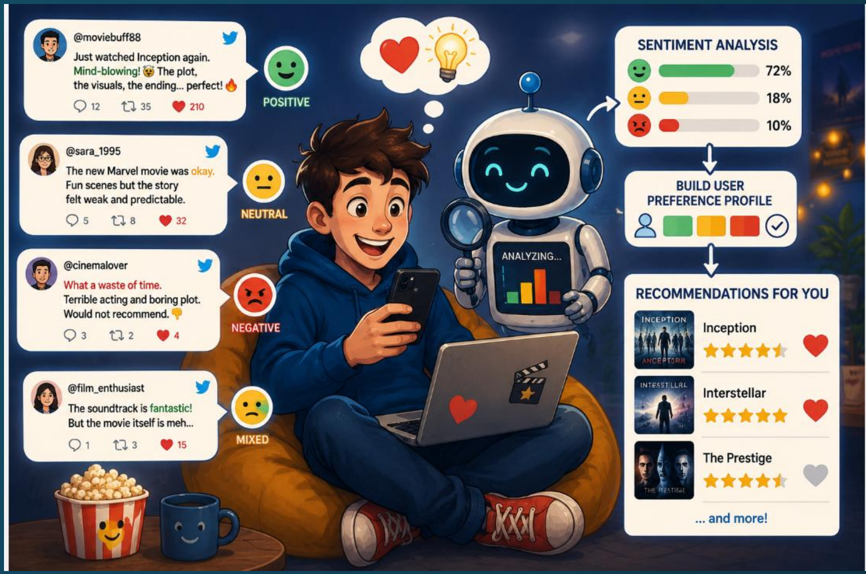
Intelligent techniques for processing large and structured data

Lecture 11

Faculty of Mathematics and Computer Science
Babeș-Bolyai University

Sergiu Limboi, PhD Teaching Assistant



Motto: ““Ratings tell us what users preferred. Sentiment tells us what users felt.”

Sentiment-Aware Recommendation Systems: Designing and Validating Similarity Measures

AGENDA

- Warm-Up
- Motivation
- What is Sentiment Analysis?
- Sentiment Score Computation
- ARP Similarity Measure
- ARP Practical Example
- Validation Framework for Similarity Measures
- Industry Problems
- Key Takeaways

Warm-Up

Faculty of Mathematics and Computer Science

Warm-Up

Go to www.menti.com and enter the code **1232 1100**

or use the QR code

Motivation

Motivation

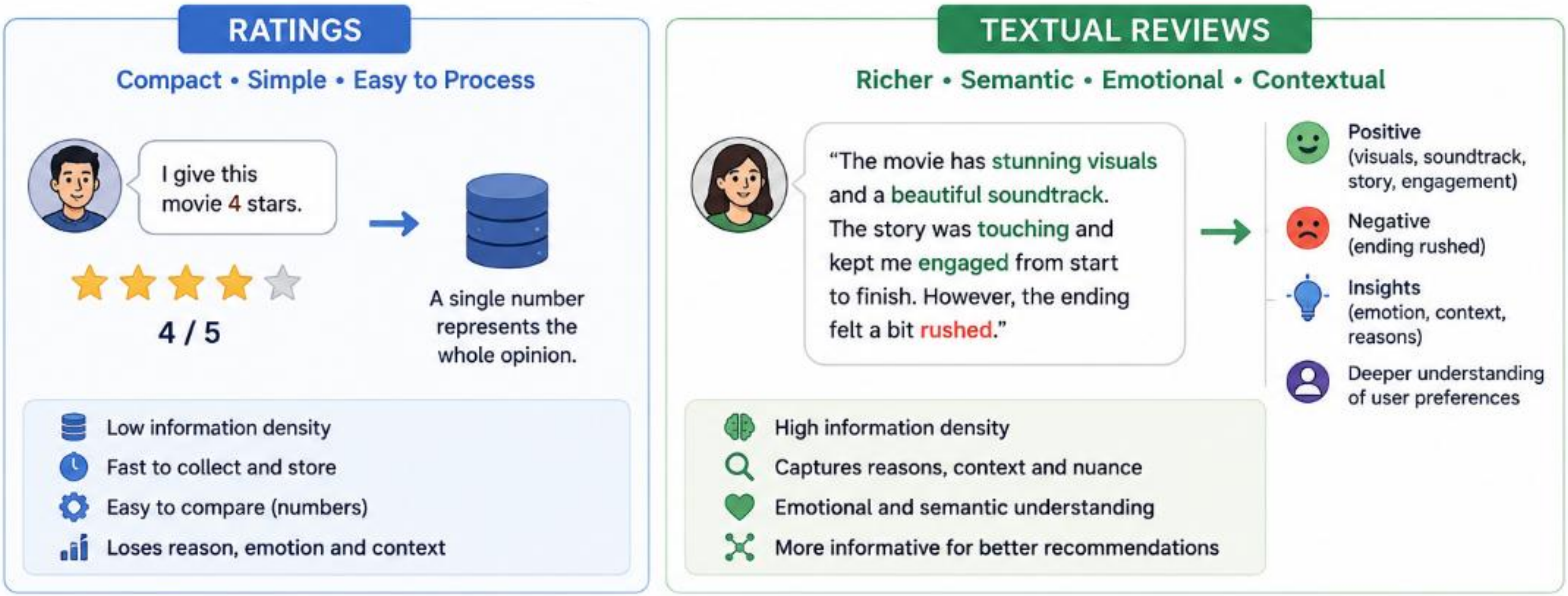
User	Rating	Review
A	5	“Absolutely amazing movie”
B	5	“Good visuals but weak story”
C	5	“Worth watching once”

Traditional systems “see” 5, 5, 5

Semantically, these users are very different.

This is the central motivation of the lecture.

Motivation


RATINGS
Compact • Simple • Easy to Process
I give this movie 4 stars.
4 / 5
A single number represents the whole opinion.
Low information density
Fast to collect and store
Easy to compare (numbers)
Loses reason, emotion and context
TEXTUAL REVIEWS
Richer • Semantic • Emotional • Contextual
"The movie has stunning visuals and a beautiful soundtrack. The story was touching and kept me engaged from start to finish. However, the ending felt a bit rushed."
Positive (visuals, soundtrack, story, engagement)
Negative (ending rushed)
Insights (emotion, context, reasons)
Deeper understanding of user preferences
High information density
Captures reasons, context and nuance
Emotional and semantic understanding
More informative for better recommendations

What is Sentiment Analysis?

Affective Computing

- Affective Computing is the field of AI that studies how systems can recognize, interpret, simulate, and respond to human emotions.

- Emotion sources:
  - facial expressions
  - voice tone
  - gestures
  - physiological signals
  - text and reviews
  - emojis and reactions

- Relevant terms are emotion, affect, and mood.
- It includes activities like Sentiment Analysis, gesture detection, or

affective analysis.

Sentiment Analysis

The Sentiment Analysis domain could be a powerful tool for analyzing text-based information.

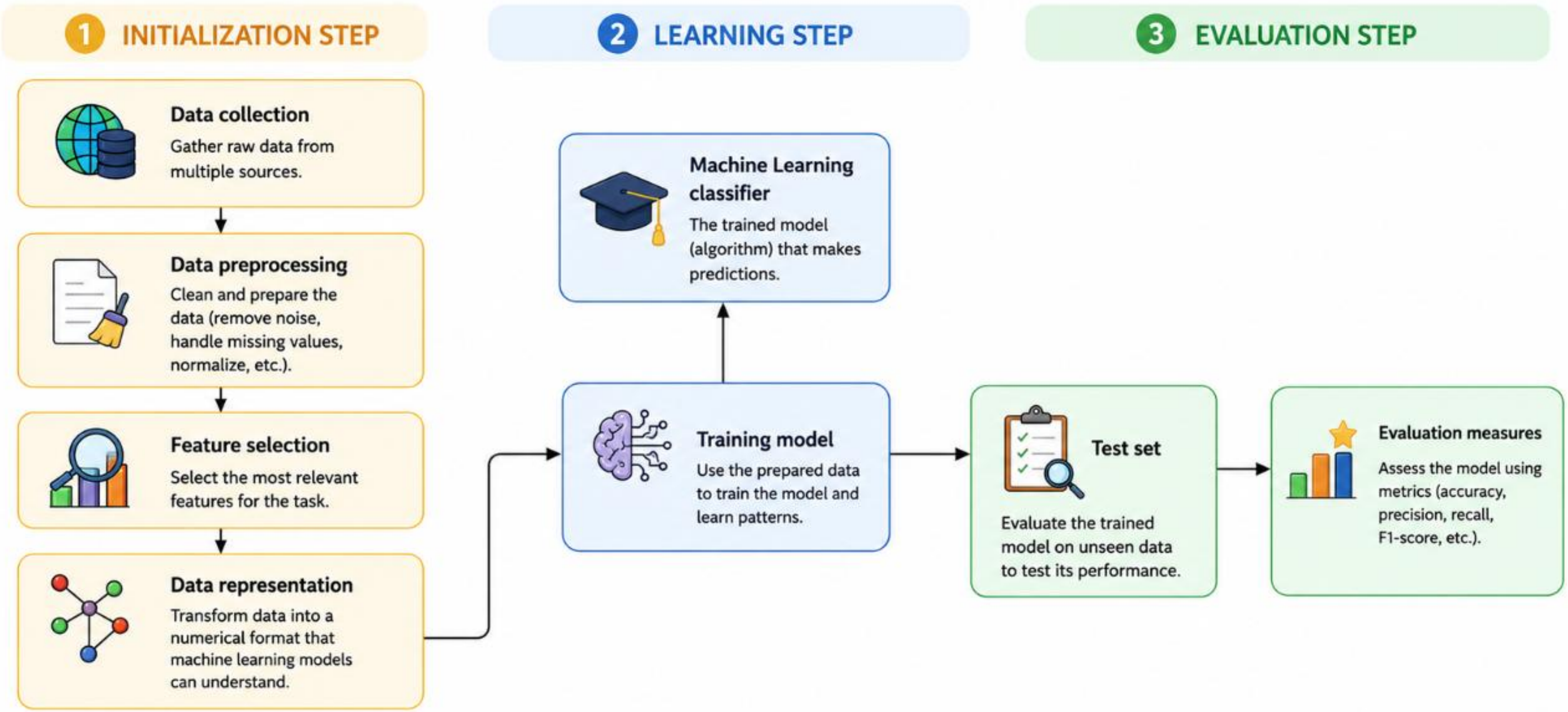
Sentiment Analysis (SA) is a subfield of Affective Computing focused on extracting opinions and emotional polarity from different sources of information.

Sentiment Analysis can be viewed as a classification problem that implies subjectivity classification (classify opinions into subjective and objective ones), polarity classification (classify expressions into negative, positive, and neutral), or opinion spam detection

Sentiment Analysis faces some challenges:

- free writing style;
- sarcasm and irony;
- emojis;
- defining neutral

Phases of text classification problem



Sentiment score computation

Sentiment score computation

- “Amazing food but slow service”

Word	Score
amazing	+0.9
slow	-0.5

→ Final sentiment score: 0.4

- Several sentiment lexicons provide sentiment scores
  - Senti Word Net (used in this lecture)
  - VADER
  - AFINN
  - **NRC** Word-Emotion Association Lexicon

Sentiment score computation

- Senti WordNet tool is used to obtain the polarity score of a word. For each word in the text-based review, the list of synsets (synonyms) is determined.

$$score(word) = \sum_{i=1}^{n} (pos(word_i) - neg(word_i)),$$

$$score(review) = \sum_{i=1}^{m} score(word_i),$$

Sentiment score computation

- A rating function can be defined to map the sentiment score to real one-to-five rating for each text-based review.

$$rating(review) = \begin{cases} 5 & \text{if } score(review) > 5 \\ score & \text{if } score(review) \in [1,5] \\ 1 & \text{if } score(review) \in (-\infty, 1) \end{cases}$$

- Sentiment Analysis can be combined with Recommender Systems for reviews by defining a sentiment score rating approach.

ARP Similarity Measure

Traditional similarity measures

Measures

Cosine
Pearson
Jaccard
Spearman
Etc.

These similarities mostly use:

ratings,
vectors,
overlaps.

But ignore:

textual semantics,
emotional intensity,
review quality.

Similarity limitation example

Two users:

- both give 4/5,
- but one writes:
  - “Beautiful masterpiece”
- another:
  - “Decent but overrated”

Traditional systems→ similar.

Humans→ not really.

ARP Similarity Measure

- The sentiment-based measure ARP is defined based on three factors of similarity: Attractiveness, Relevance, and Popularity

- Let $U=\{user_1, user_2,..., user_n\}$ denote a set of users.

- Let $R=\{review_1, review_2,...,review_m\}$ denote a set of review.

Core idea: Users should be compared not only through ratings, but through emotional behaviour in reviews.

Attractiveness

- How emotionally appealing a review is.
- "Excellent acting and beautiful soundtrack" → high attractiveness
- Strong emotional language increases attractiveness.
- The attractiveness for the j^{th} review in the collection R is defined as follows:

$$attr(review_j) := \frac{1}{k}\sum_{i=1}^{k} score^p(w_j) + \frac{1}{l}\sum_{i=1}^{l} score^n(w_i),$$

where:

- k is the number of words with a positive score.
- l is the number of words with a negative score.
- $score^p(w_i)$, $score^n(w_i)$ is the positive, respectively negative score of a word obtained via Senti Word Net tool.

Popularity

- How much a review deviates from average user behaviour.

- Example: If everyone gives neutral opinions, but one user is extremely positive, that review becomes distinctive.

- The following definitions were formulated for the popularity measures, specifically for the j^{th} review given by the i^{th} user:

$$popularity(review_j^i) := 1 + \left[score(review_j^i) - AM(score(reviews))\right]^2,$$

Where:

- $score(review^i_j)$ is the sentiment score of the j^{th} review calculated as
the sum of sentiment scores for all its words.
- AM(score(reviews)) is the average of the sentiment scores of all
user's reviews.

Popularity

- Popularity of a user:

$$p(user) := \sum_{j=1}^{s} popularity(review_j),$$

where popularity(review$_{j)}$ is the popularity of a review computed previously and s is the number of items reviewed by the user.

- The popularity of user u_1 in relation to user u_2, respectively of user u_2 in relation to user u_1 is determined as follows:

$$p(u_1/u_2) := \sum_{j=1}^{t} popularity(review_j^1),$$

$$p(u_2/u_1) := \sum_{j=1}^{t} popularity(review_j^2),$$

Where:

- $p(u_1/u_2)$ is the popularity of user u_1 in relation to user u_2 computed based on the reviews given by user u_1 for those items that have been evaluated also by user u.
- t is the number of items reviewed by both users u_1 and u_2.

Relevance

- How informative/significant the review is.
- **Example:** “Good.” vs “Excellent cinematography and emotional storytelling.”
- Second review is more relevant.
- Relevance indicates the average deviation of the j^{th} review given by the i^{th} user in terms of attractiveness.

$$relevance(review_j^i) := 1 + \left[attr(review_j^i) - AM(attr(reviews))\right]^2 ,$$

- Where:
  - attr ($review^i_j$) is the attractiveness factor for the j^{th} review given by
the i^{th} user.
  - AM(attr (reviews)) is the average of all attractiveness values for all
user’s text-based reviews.

Relevance

- Relevance of a user: $r(user) := \sum_{j=1}^{s} relevance(review_j),$

- The relevance of user u_1 in relation to user u_2, respectively of user u_2 in relation to user u_1 is determined as follows:

$$r(u_1/u_2) := \sum_{j=1}^{t} relevance(review_j^1),$$

$$r(u_2/u_1) := \sum_{j=1}^{t} relevance(review_j^2),$$

ARP Similarity Measure

- The ARP similarity was defined between user u_1 and u_2 using the following notations:

$$ARP(u_1/u_2) := (p(u_1/u_2), r(u_1/u_2))$$

$$ARP(u_2/u_1) := (p(u_2/u_1), r(u_2/u_1))$$

ARP Similarity Measure

- Finally, the ARP similarity between two users is :

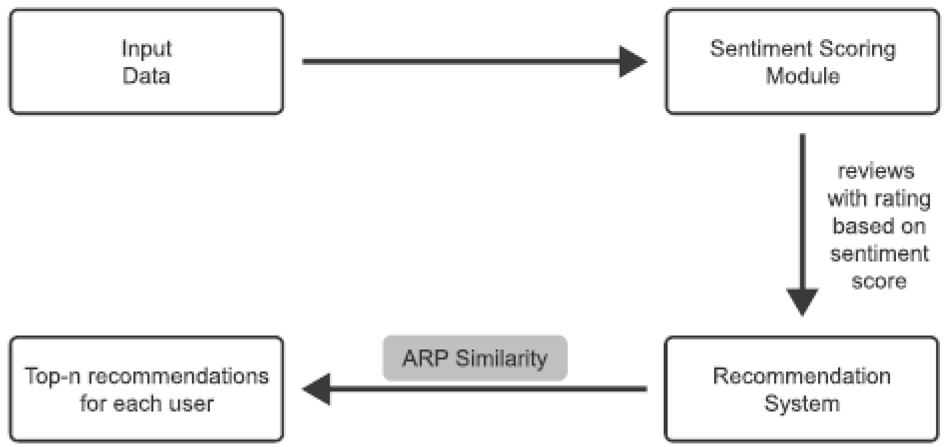
$$ARP(u_1, u_2) := cos((ARP(u_1/u_2), ARP(u_2/u_1))$$

$$= \frac{ARP(u_1/u_2) \cdot ARP(u_2/u_1)}{\|ARP(u_1/u_2)\| \cdot \|ARP(u_2/u_1)\|} =$$

$$\frac{p(u_1/u_2) \cdot p(u_2/u_1) + r(u_1/u_2) \cdot r(u_2/u_1)}{\sqrt{(p(u_1/u_2))^2 + (r(u_1/u_2))^2} \cdot \sqrt{(p(u_2/u_1))^2 + (r(u_2/u_1))^2}}$$

ARP Similarity Measure

- ARP (Attractiveness-Relevance-Popularity) can be integrated into the k-Nearest Neighbors (kNN) collaborative filtering approach for the recommendation process.



ARP Practical Example

ARP Practical Example

User	Movie	Review	Sentiment score	Attractiveness
U1	M1	“Amazing acting and beautiful story”	0.9	0.85
U1	M2	“Good visuals but slow ending”	0.4	0.45
U2	M1	“Great acting and emotional story”	0.8	0.80
U2	M2	“Nice visuals but weak ending”	0.3	0.40

ARP Practical Example

- Step 1 – Compute average sentiment score

$$AM(score) = \frac{0.9 + 0.4 + 0.8 + 0.3}{4} = 0.6$$

- Step 2 – Compute popularity of each review

$$p(review) = 1 + (score(review) - AM(score))^2$$

User	Movie	Score	Popularity
U1	M1	0.9	(1+(0.9-0.6)^2=1.09)
U1	M2	0.4	(1+(0.4-0.6)^2=1.04)
U2	M1	0.8	(1+(0.8-0.6)^2=1.04)
U2	M2	0.3	(1+(0.3-0.6)^2=1.09)

ARP Practical Example

- Step 3 – Compute average attractiveness

$$AM(attr) = \frac{0.85 + 0.45 + 0.80 + 0.40}{4} = 0.625$$

- Step 4 – Compute relevance of each review

$$r(review) = 1 + (attr(review) - AM(attr))^2$$

User	Movie	Attractiveness	Relevance
U1	M1	0.85	(1+(0.85-0.625)^2=1.0506)
U1	M2	0.45	(1+(0.45-0.625)^2=1.0306)
U2	M1	0.80	(1+(0.80-0.625)^2=1.0306)
U2	M2	0.40	(1+(0.40-0.625)^2=1.0506)

ARP Practical Example

- **Step 5 — Compute user relation vectors**
  - Because U1 and U2 reviewed the same two movies, we sum popularity and relevance over the common reviewed items.
  - For U1 in relation to U2:

$$p(U1/U2) = 1.09 + 1.04 = 2.13$$

$$r(U1/U2) = 1.0506 + 1.0306 = 2.0812$$

$$ARP(U1/U2) = (2.13,\ 2.0812)$$

  - For U2 in relation to U1:

$$p(U2/U1) = 1.04 + 1.09 = 2.13$$

$$r(U2/U1) = 1.0306 + 1.0506 = 2.0812$$

$$ARP(U2/U1) = (2.13,\ 2.0812)$$

ARP Practical Example

- Step 6 – Compute final ARP similarity

- Now compute cosine similarity between the two vectors:

$$ARP(U1, U2) = cos((2.13, 2.0812), (2.13, 2.0812))$$

- Because the vectors are identical:

$$ARP(U1, U2) = 1$$

ARP Practical Example

The two users are highly similar because:

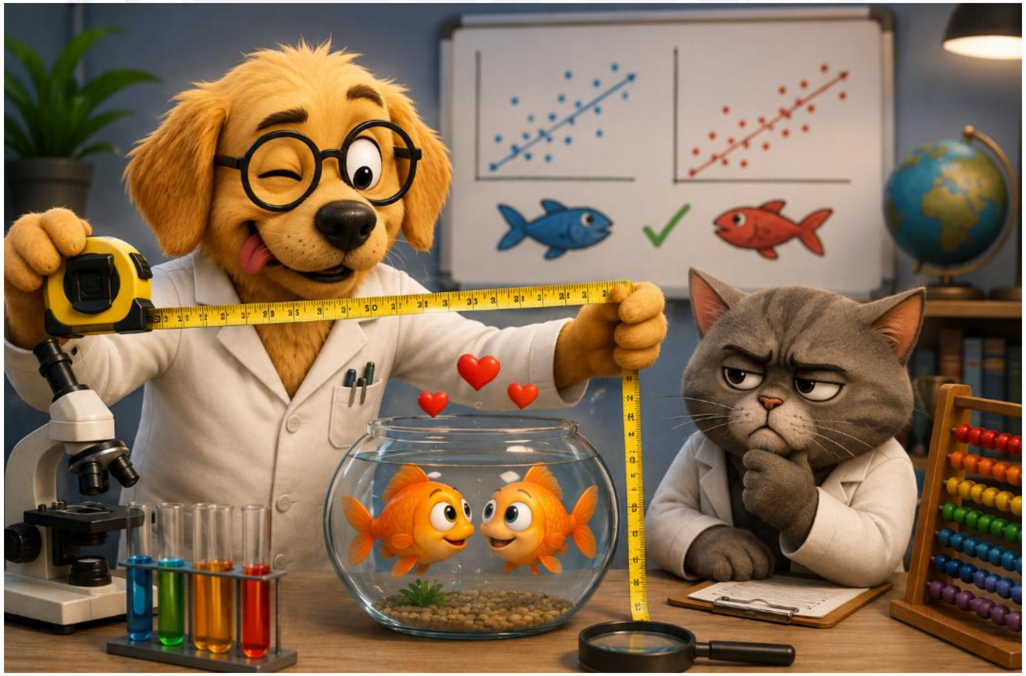
they reviewed the same items,
their sentiment scores are close,
their attractiveness values are close,
their popularity and relevance vectors are almost identical.
So the recommender system can say: U1 and U2 express similar emotional preferences in their reviews.

If both users gave the same ratings but one wrote very emotional reviews and the other wrote neutral reviews, should they still be considered equally similar?

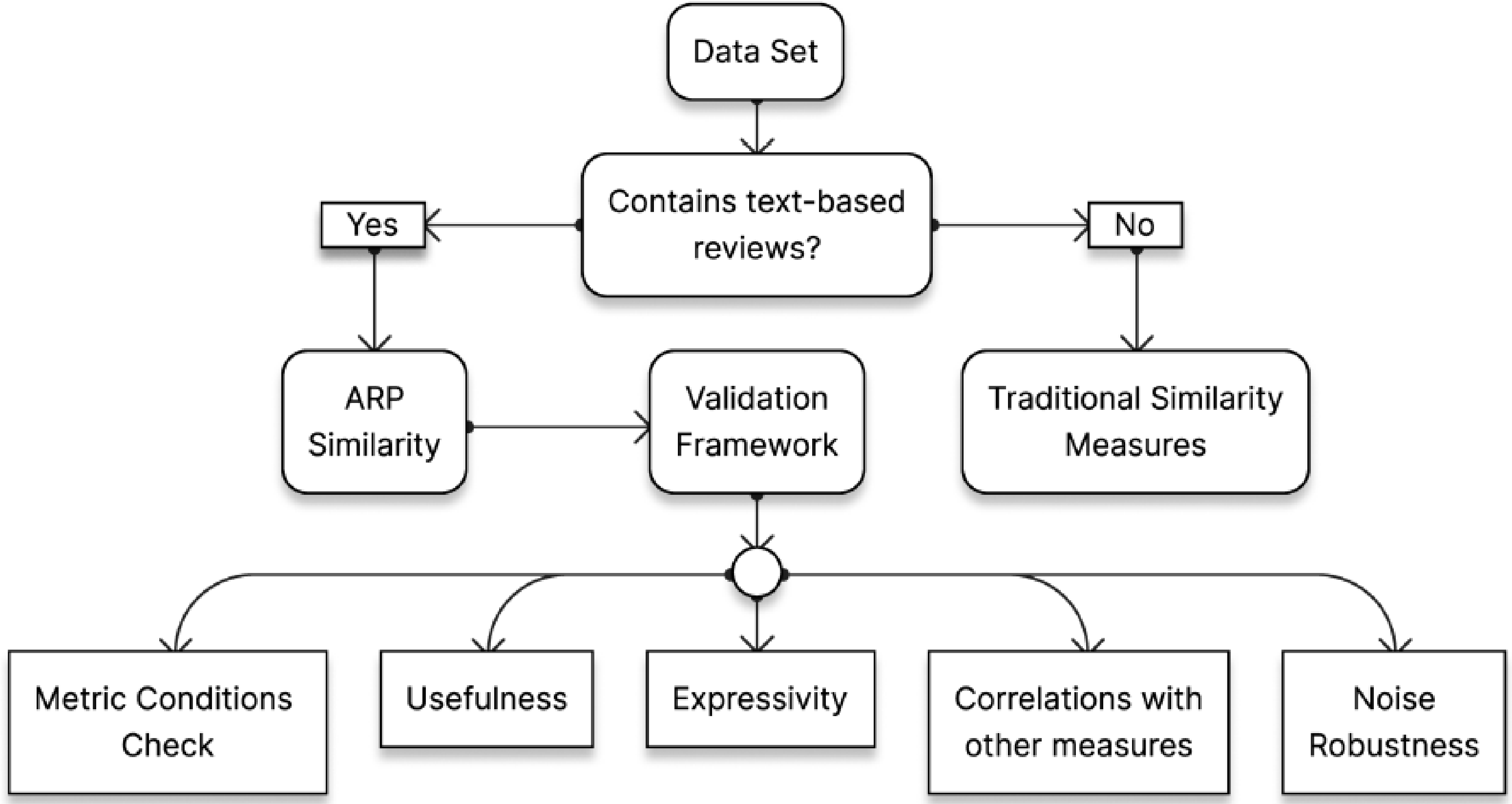
Validation Framework For Similarity Measures

Validation framework for similarity measures

"How do we know a similarity measure is GOOD?"



Validation framework for similarity measures



Metric conditions

- A good similarity should behave logically.

- Metric properties:
  - symmetry,
  - non-negativity,
  - identity,
  - triangle inequality.

- ARP is considered a **semi-metric** because it satisfies some important properties of a mathematical distance/similarity measure, but not all the strict conditions required for a true metric (the heuristic components are not guaranteed to preserve triangle inequality).

Usefulness

Does ARP improve recommendations?

The usefulness component focuses on the applicability of a similarity measure in different contexts.

The measure to be validated should be applicable to a variety of data sets (e.g.: with or without ratings, collections with textual information, etc.) and can be compared to other traditional similarities based on evaluation metrics.

Expressivity

Can ARP capture richer semantic information?

The goal of the expressivity unit is to analyse that the ARP measure is a better fit for data sets with text-based reviews compared to other measures.

Then, a clustering algorithm is applied to the data set. The validation methodology implies that the measure under evaluation should produce a better clustering results compared to the ones produced by other traditional similarity measures.

Correlation Analysis

This component of the validation framework has the goal to compute a correlation coefficient between several similarity measures.

The *Pearson correlation coefficient* is computed to define a relationship (linear or not) between the ARP similarity to be validated and other traditional measures.

Is ARP just cosine similarity? Results: strong correlation with cosine but enriched with sentiment information.

Noise Robustness

Real-world reviews are messy:

- typos,
- slang,
- sarcasm,
- emojis,
- multilingual text.

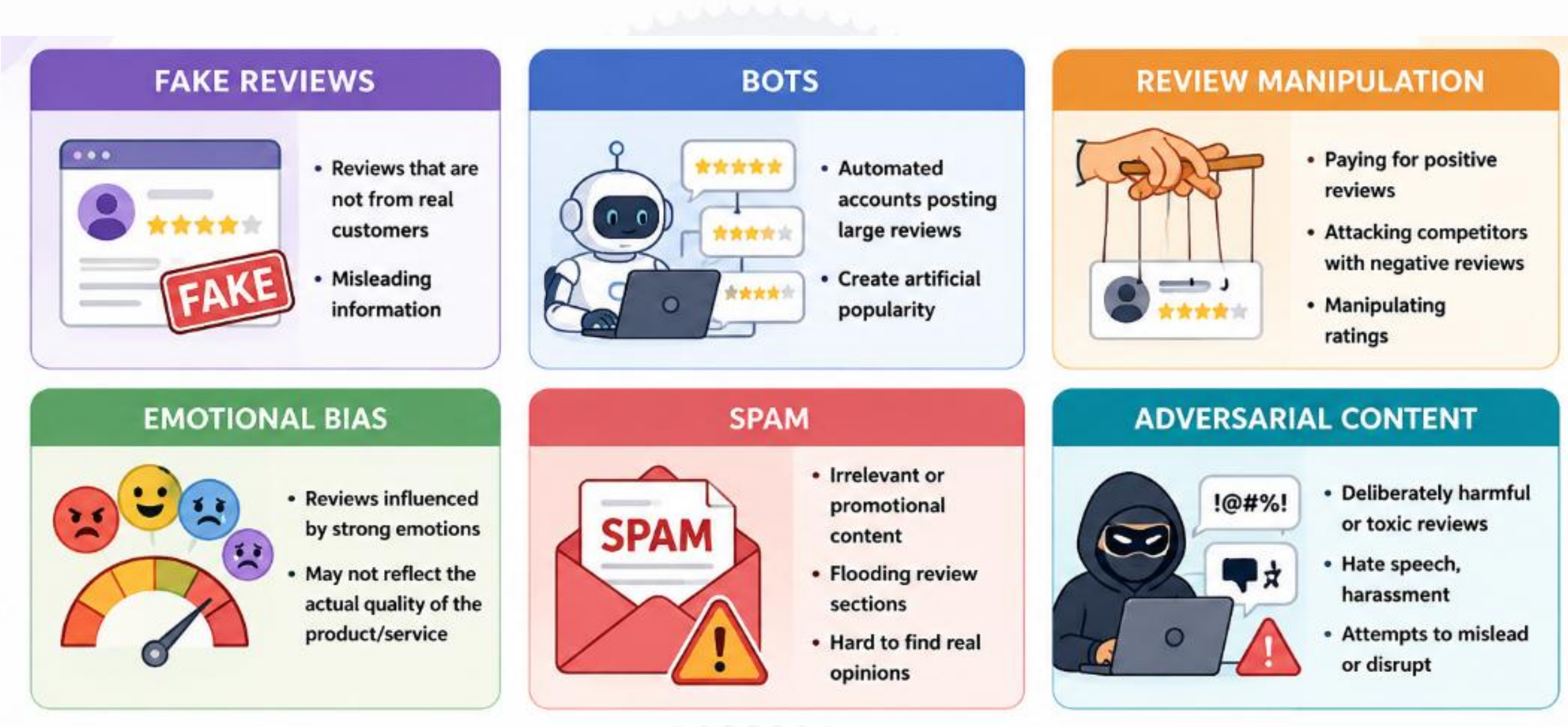
Does the similarity still work?

The measure's behaviour is analysed using different noisy data scenarios. The source of noise comes from two directions: noise from information and classifiers.

Therefore, the robustness of the ARP evaluated similarity is determined by considering different pre-processing steps for the input data and a variety of classifiers.

Industry Problems

Industry Problems



Modern Systems

- Modern systems combine:
  - embeddings,
  - sentiment,
  - behaviour,
  - attention,
  - semantic similarity.

Teamwork Activity

- **Design YOUR OWN similarity measure**
- Choose one domain: gaming, social media, restaurants, music, education platform, etc.
- What features besides ratings would you use?
- **Teams must answer:**
  - **What signals would you collect?**
  - **Which signals are MOST important?**
  - **Which signals are noisy/unreliable?**
  - **How would you combine them?**
  - **How would you validate the similarity?**

Example of similarity:
Similarity=40%sentiment+30%engagement +20%watch time+10%ratings

Key Takeaways

Key Takeaways

Ratings are not enough

Reviews contain semantic information

Similarity measures can be redesigned

Validation matters

Modern recommenders combine multiple AI techniques

Thank you for your attention — questions, thoughts, or challenges?

FACULTY OF MATHEMATICS AND COMPUTER SCIENCE
BABEȘ-BOLYAI UNIVERSITY

1 Mihail Kogălniceanu Street,
Cluj-Napoca, Cluj, România

www.cs.ubbcluj.ro