ADVERSARIAL MACHINE LEARNING IN RECOMMENDER SYSTEMS (AML-RECSYS)

Yashar Deldjoo Tommaso Di Noia Felice Antonio Merra

Politecnico di Bari

Polytechnic University of Bari , Bari, Italy

ABOUT US

Yashar DELDJOO

@yashardel

SisInf Lab,
Polytechnic **U**niversity of **B**ari,
Italy

Tommaso DI NOIA

@TommasoDiNoia

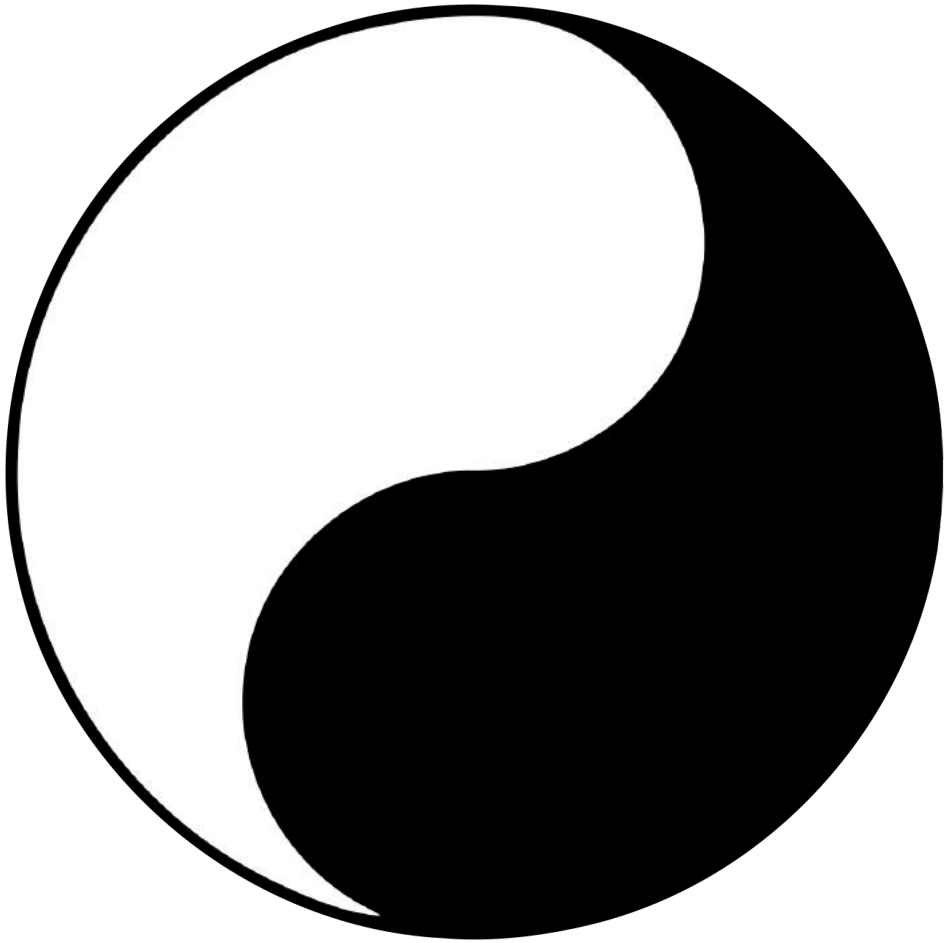
SisInf Lab,
Polytechnic **U**niversity of **B**ari,
Italy

Felice MERRA

@merrafelice

SisInf Lab,
Polytechnic **U**niversity of **B**ari,
Italy

Politecnico di Bari



PLAN FOR TODAY

1. Part I
    1. Background technologies
    2. Main concepts behind adversarial learning
    3. The Attack-Defense game
2. Part II
    1. Foundations to AML for security of ML systems
    2. Adversarial Learning for attack-defense strategies in Recommender Systems
    3. Domains
3. Part III
    1. GAN-based Recommendation Framework (GAN-RF)
    2. Applications
    3. Domains

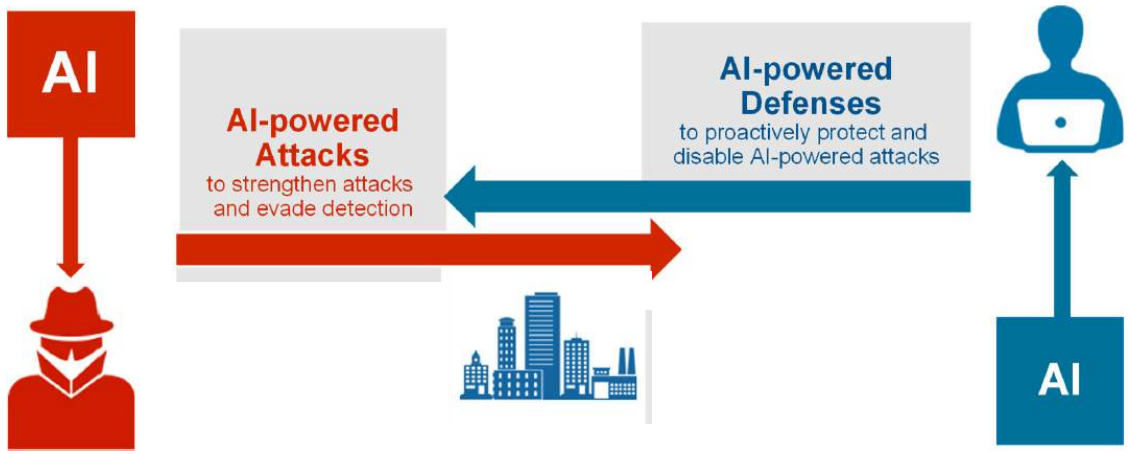
OBJECTIVES

- Bridge the gap between advances made in the field of recommender system and information security
- Understand key concepts in adversarial machine learning
- Adversarial machine learning ≠ generative adversarial network (GANs)
- Present different application of AML for security and defense
- Present several goals of AML for RecSys

Motivating Examples

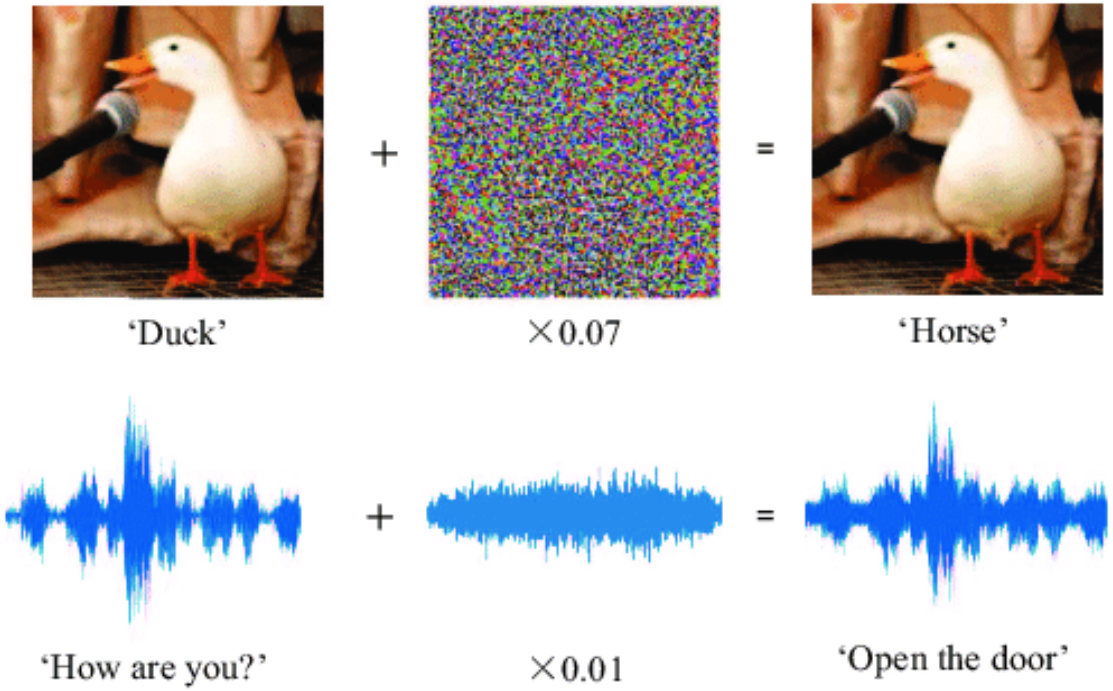
WHY ADVERSARIAL MACHINE LEARNING?

Adversarial Environment



Picture taken from:
https://www.nap.edu/read/25534/chapter/5

WHY ADVERSARIAL MACHINE LEARNING?



Gong, Yuan, and Christian Poellabauer. "Protecting voice-controlled systems using sound source identification based on acoustic cues."
27th International Conference on Computer Communication and Networks (ICCCN). IEEE, 2018.

WHY ADVERSARIAL MACHINE LEARNING?

"panda"

+

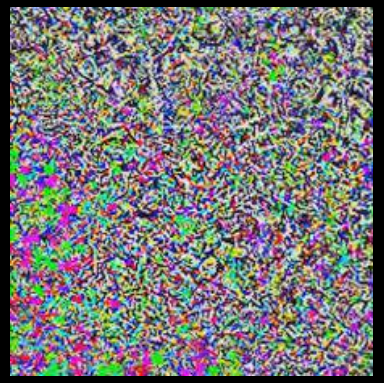
Adversarial Noise

=

"gibbon"

Politecnico di Bari

IS THE PANDA SO IMPORTANT?



Adversarial Noise

What is the origin of such failure?

RecSys + Adversarial Learning

WHY RECSYS + ADVERSARIAL LEARNING?

Evaluation Goals

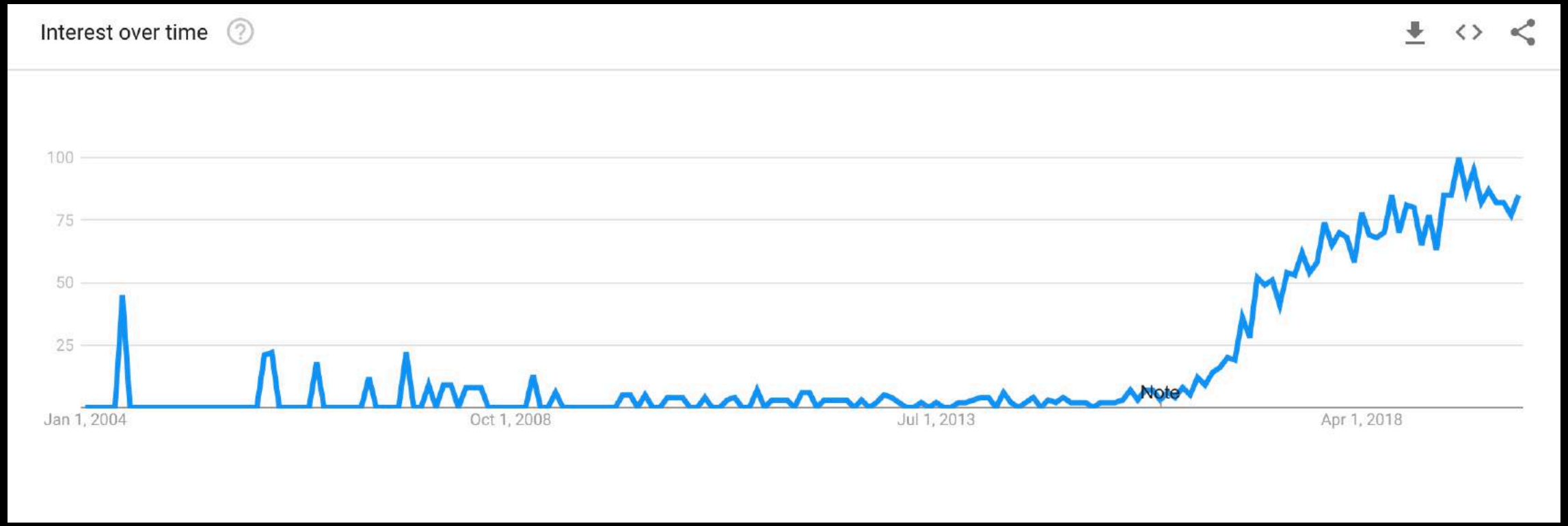
- Accuracy
- Coverage
- Confidence and Trust
- Novelty
- Serendipity
- Diversity
- Security
- Privacy
- Fairness
- Scalability

Increases the ability of learning in adversarial setting

Recomendation Models

- Collaborative Filtering
  - Model-based
  - Memory-based
  - Graph-based
  - Deep
- Content-based Filtering
  - Metadata
  - Multimedia (audio and visual)
  - Knowledge-base
- Context-aware
  - Social
  - Temporal
- Hybrid

ADVERSARIA LEARNING: SEARCH TERM FREQUENCY OVER TIME



[SOURCE GOOGLE TRENDS]

Politecnico di Bari

ADVERSARIA LEARNING: SUGGESTED RELATED TOPICS

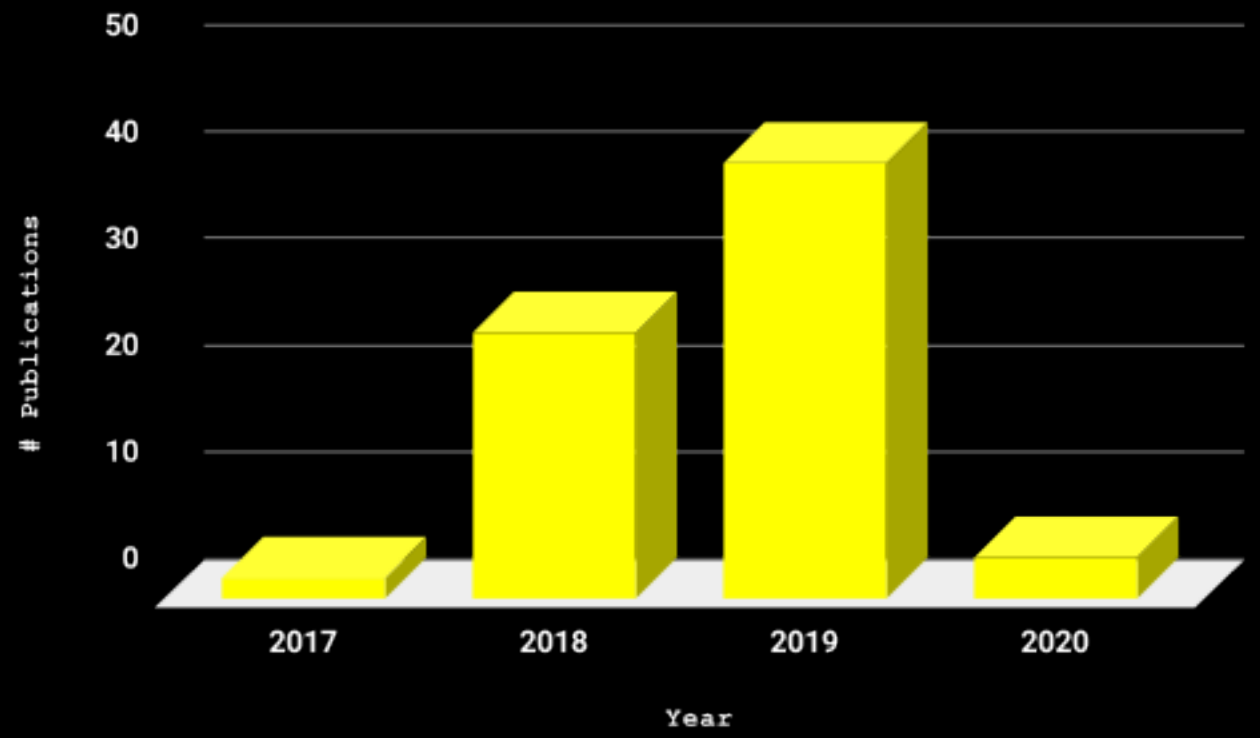
Related topics	Rising
1 Learning - Topic	Breakout
2 Generative adversarial networks - Topic	Breakout
3 Adversarial machine learning - Field of study	Breakout
4 Generative model - Topic	Breakout
5 Deep learning - Topic	Breakout

Related queries	Rising
1 deep learning	Breakout
2 adversarial networks	Breakout
3 adversarial machine learning	Breakout
4 generative adversarial networks	Breakout
5 gan	Breakout

[SOURCE GOOGLE TRENDS]

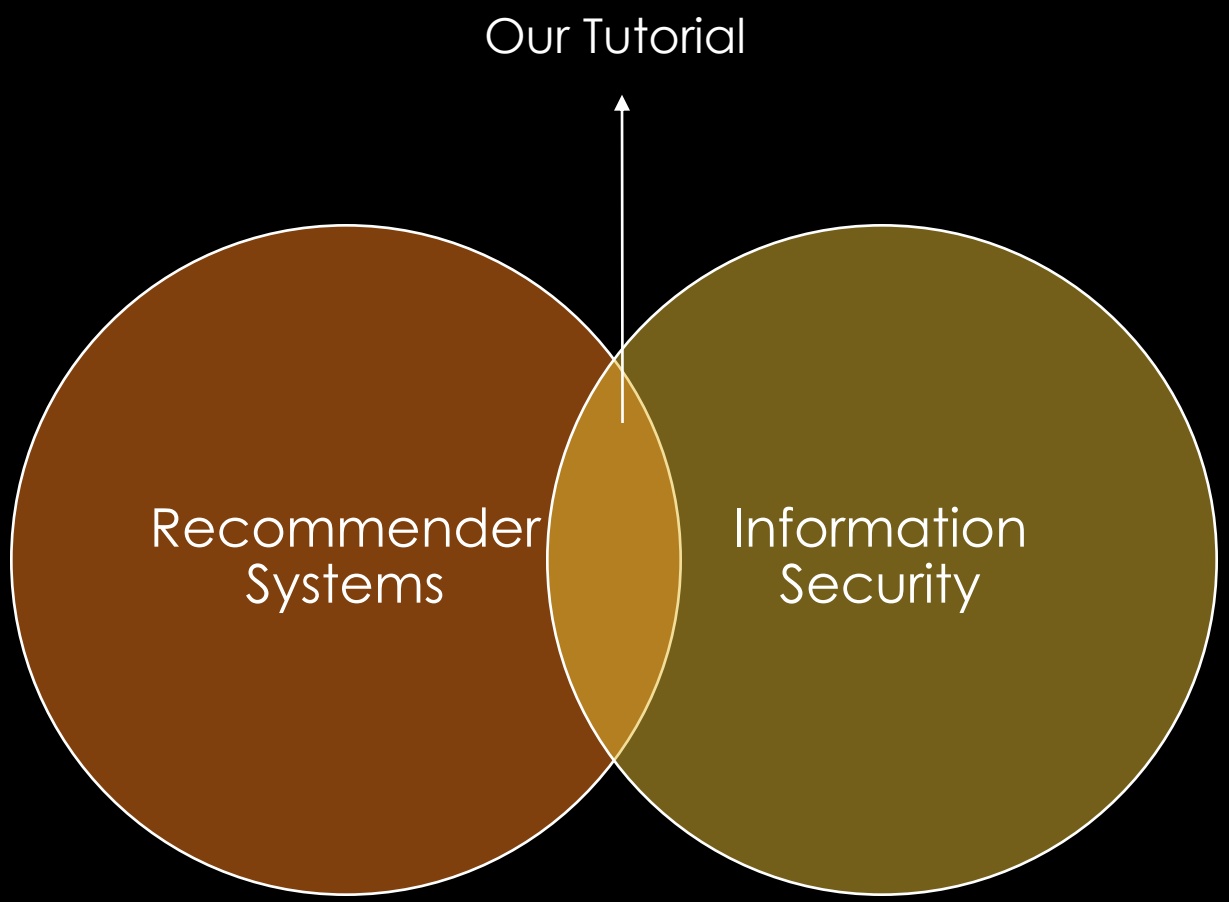
ADVERSARIAL LEARNING FOR RS

- More than **70** papers in the last 2 years
- Top Conferences/Journals involved:
  - WSDM
  - SIGIR
  - RecSys
  - WWW
  - KDD
  - IJCAI
  - ICML
  - CIKM
  - AAAI
  - NIPS



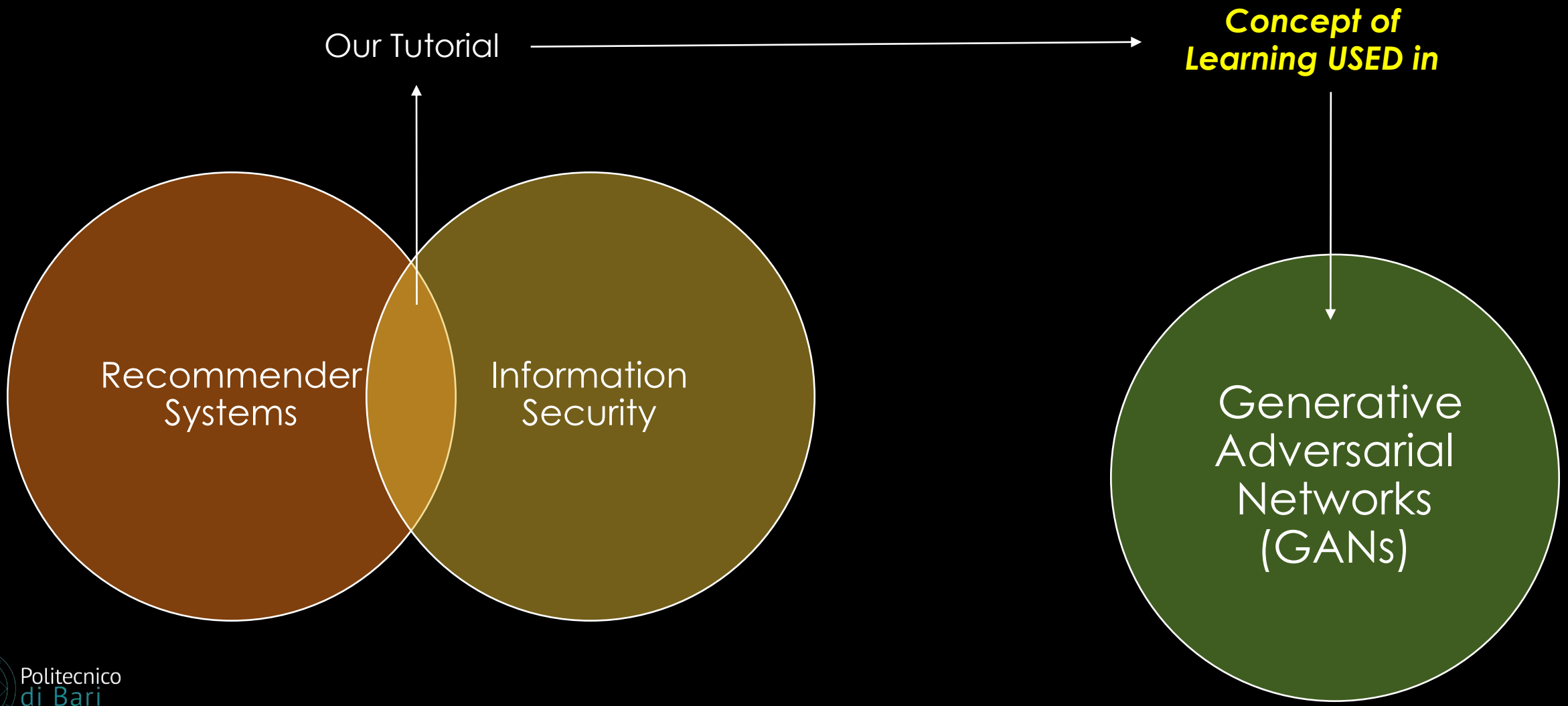
Politecnico di Bari

THE FOCUS OF OUR TUTORIAL



Politecnico di Bari

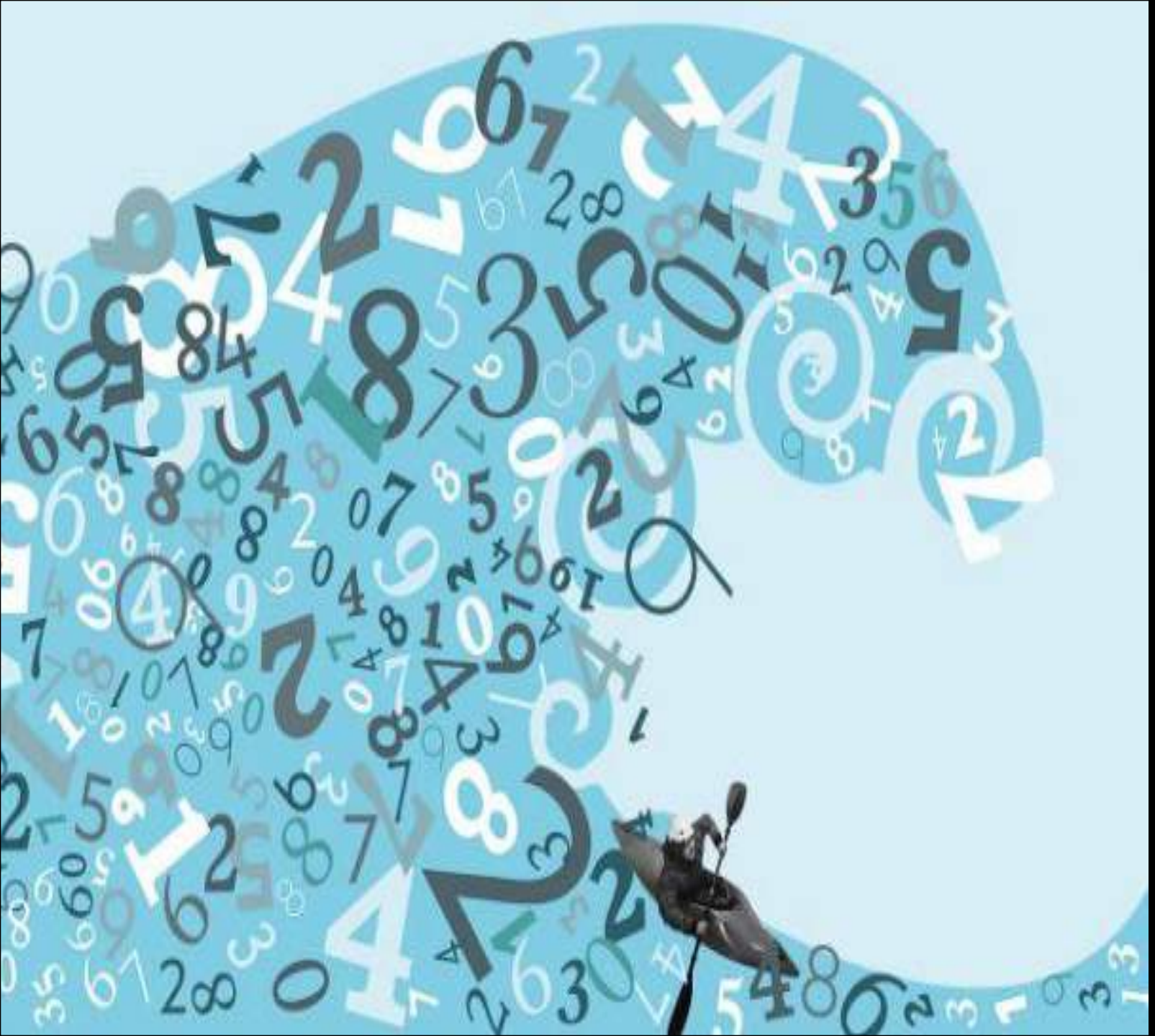
THE FOCUS OF OUR TUTORIAL



Politecnico di Bari

1.1 RECOMMENDER SYSTEMS

Foundations



INFORMATION OVERLOAD

Appeared for the first time in «**Future Shock**» by Alvin Toffler, 1970

Film consigliati per te

Simili ai titoli che hai guardato

I film più popolari Esplora

Film - Drammatico Esplora

INFORMATION OVERLOAD

NETFLIX

SOME DEFINITIONS

- In its most common formulation, **the recommendation problem is reduced to the problem of estimating ratings** for the items that have not been seen by a user.

  [G. Adomavicius and A. Tuzhilin. Toward the Next Generation of Recommender Systems: A survey of the State-of-the-Art and Possible Extension. TKDE, 2005.]

- Recommender Systems (**RS**s) are software tools and techniques providing **suggestions** for items to be of use to a user.

  [F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, editors. Recommender Systems Handbook. Springer, 2015.]

THE PROBLEM

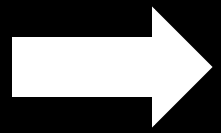
- Estimate a utility function to automatically predict how much a user may like an item which is unknown to them.

Input

Set of users $U = \{u_1, \dots, u_M\}$

Set of items $X = \{x_1, \dots, x_N\}$

Utility function $f: U \times X \rightarrow \mathbb{R}$



Output

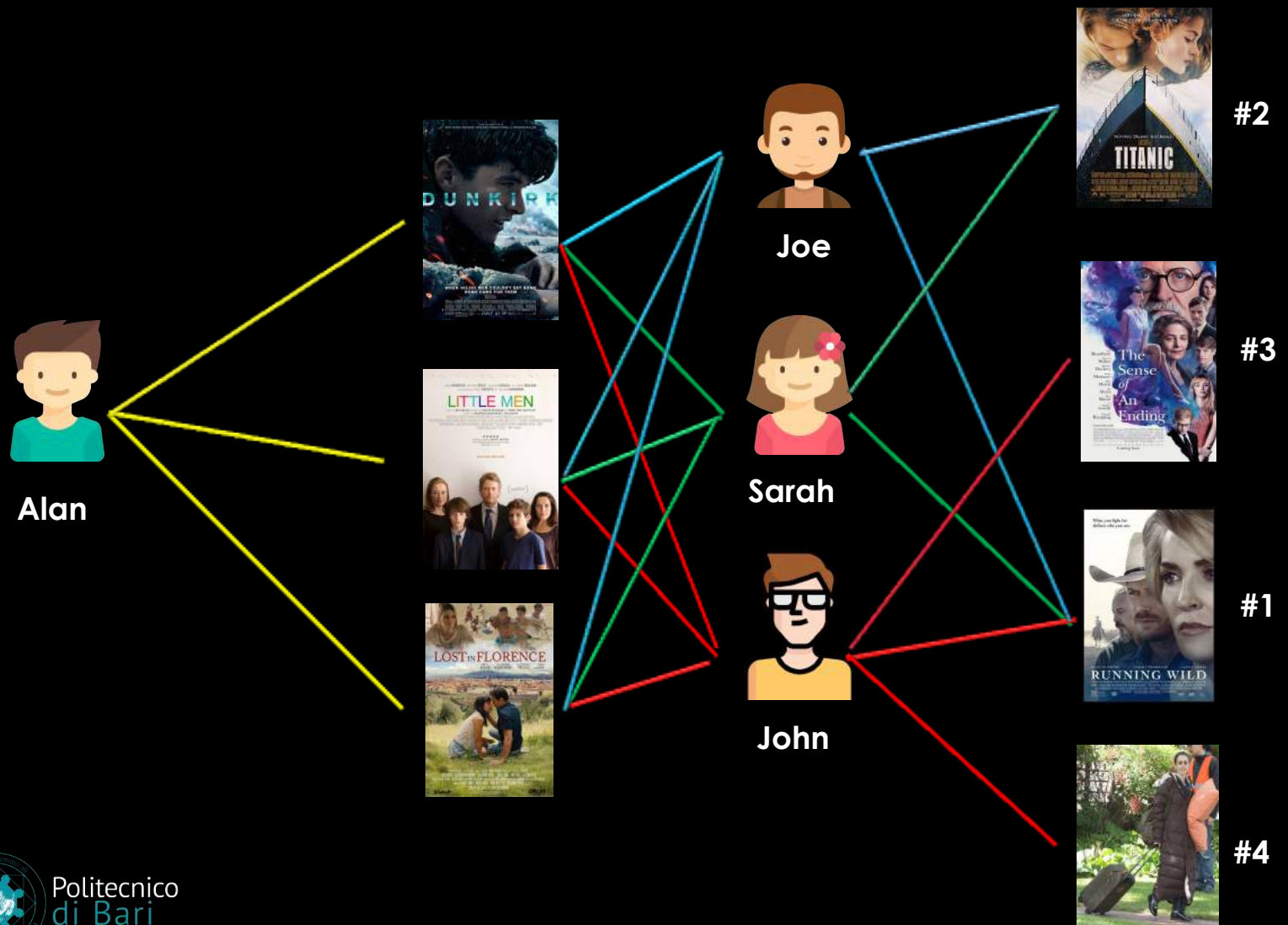
$\forall\, u \in U, x'_u = \arg max_{x \in X}\, f(u, x)$

CLASSICAL COLLABORATIVE NON-NEURAL ERA

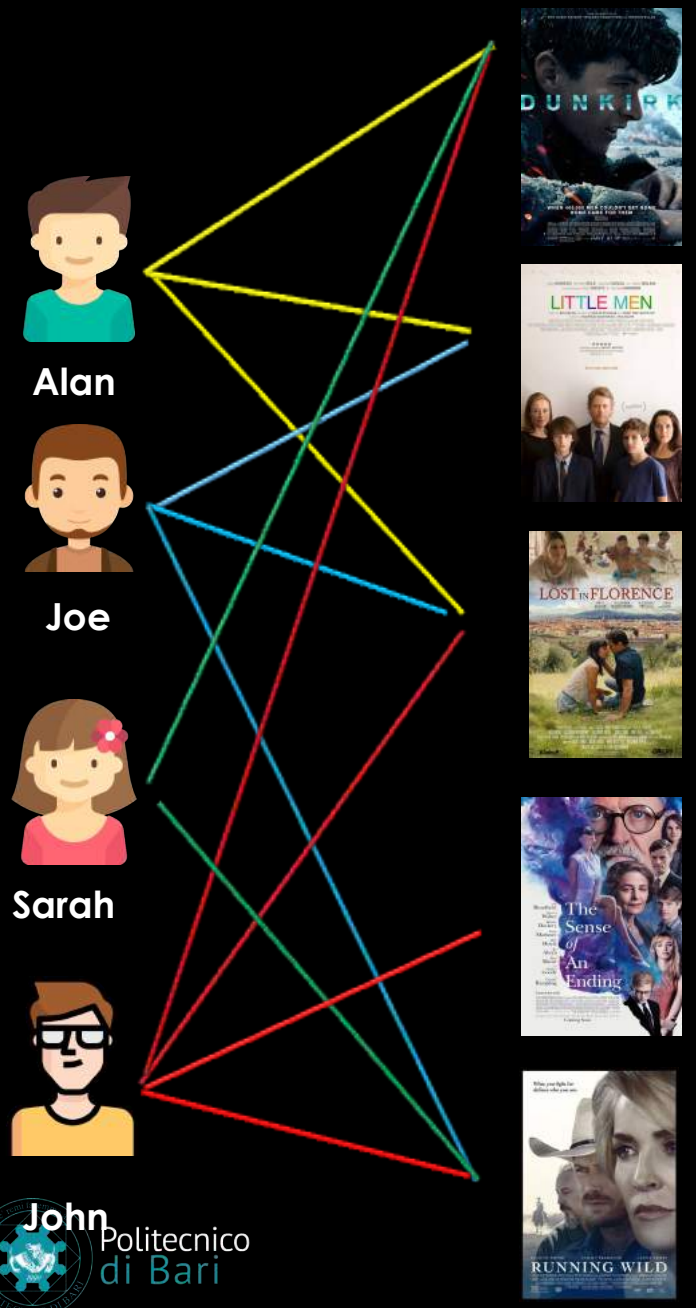
- **Learning paradigm** categorization:
  - **Memory-based CF**: recommendation based on the similarity of users-user, or item-item, interactions. There is no need to train a model.
  - **Model-based CF**: predict users' feedback of unseen items using latent factor model such as matrix-factorization(MF)

- **Model training** categorization:
  - **Point-wise**
  - **Pair-wise**
  - **List-wise**

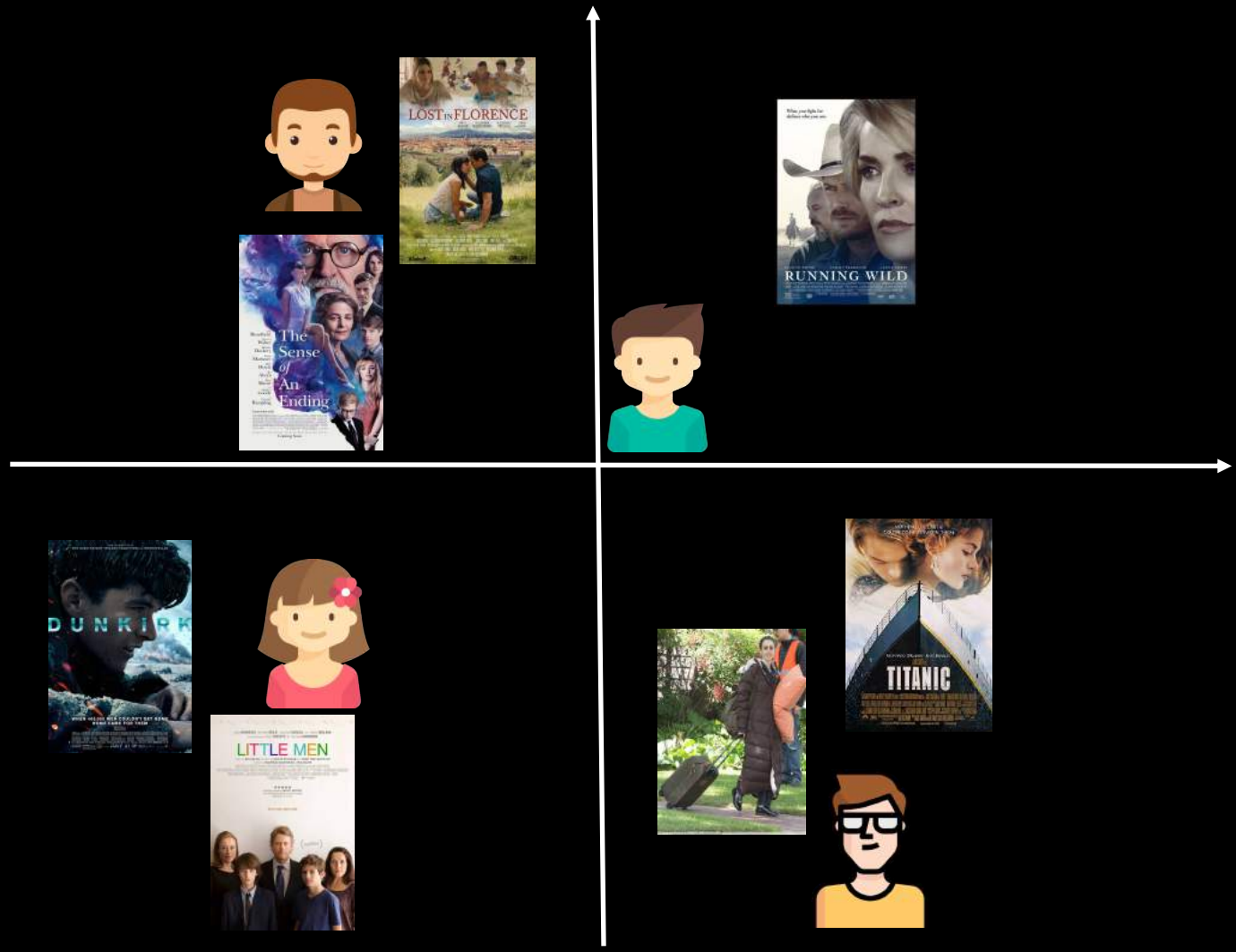
MEMORY-BASED CF



MODEL-BASED CF

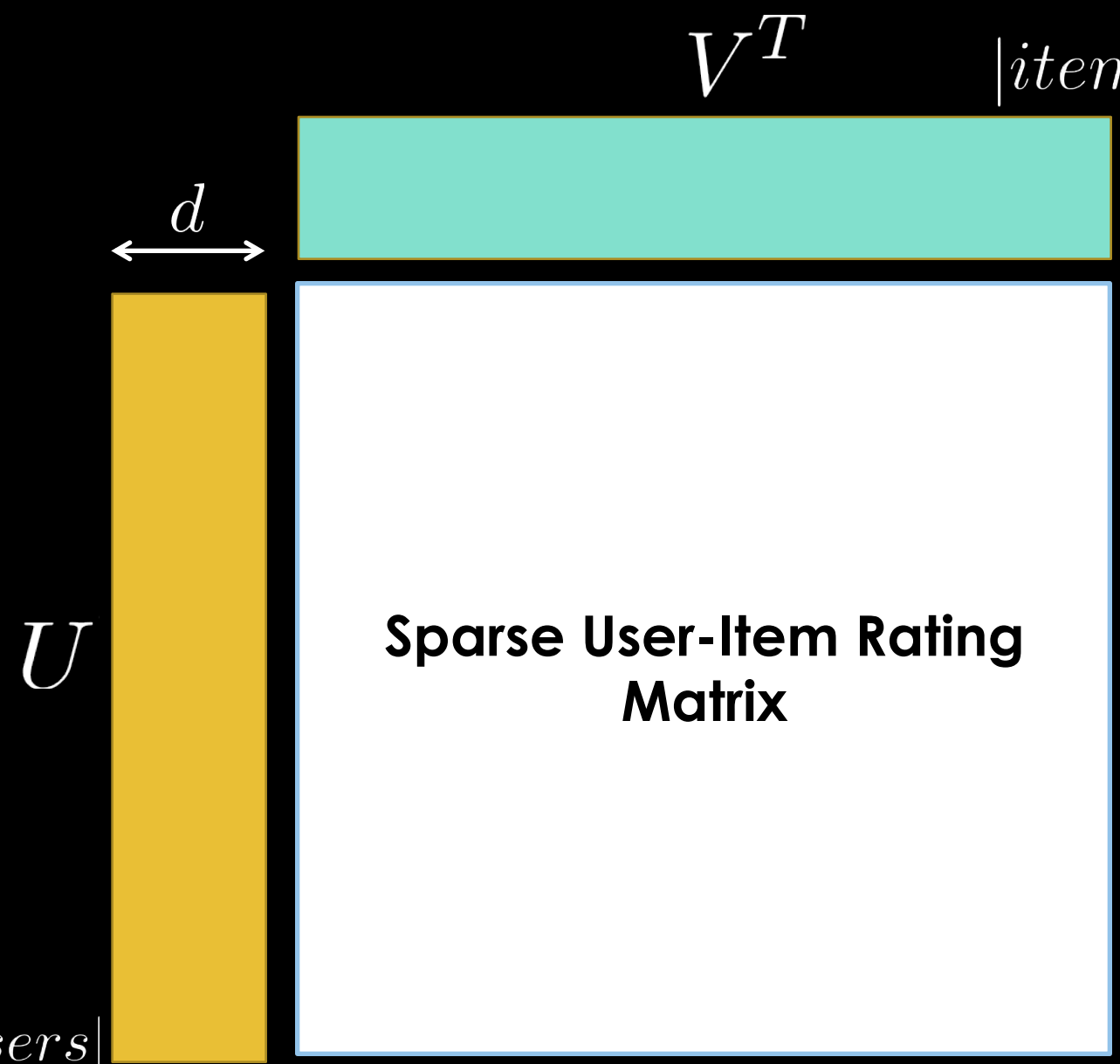


User and items are projected into a same low-d space.



Politecnico di Bari

LATENT FACTOR-BASED CF



$d << |users|, |items|$

$$\hat{R} = U^T V$$

RATING PREDICTION

$$L_{PMF}(U,V) = \sum_{i=1}^{M} \sum_{j \in L_i} (r_{ij} - \mathbf{u}_i^T \mathbf{v}_j)^2 + \lambda(||U||_F^2 + ||V||_F^2)$$

rating prediction loss

regularization

Politecnico di Bari

RANKING

Maximize the probability p that *k* is ranked higher than *j* for all pairs in $\mathcal{D}_s$

$$\prod_{(i,k,j)\in\mathcal{D}_s} p(k >_i j|\Theta), \forall k \in L_{i,+},\ j \in L_{i,-}$$

$\mathcal{D}_s$ contains all$(+,-)$pairs for each user

$$p(k >_i j|\Theta) = \sigma(\hat{x}_{ikj}(\Theta))$$

$$\hat{x}_{ikj}(U,V) = \hat{x}_{ik} - \hat{x}_{ij} = \mathbf{u}_i^T\mathbf{v}_k - \mathbf{u}_i^T\mathbf{v}_j$$

$$L_{BPR}(U,V) = \sum_{(i,k,j)\ \in\mathcal{D}_s} log(1 + exp(-\mathbf{u}_i^T(\mathbf{v}_k - \mathbf{v}_j))) + \lambda(\|U\|_F^2 + \|V\|_F^2)$$

pairwise ranking loss

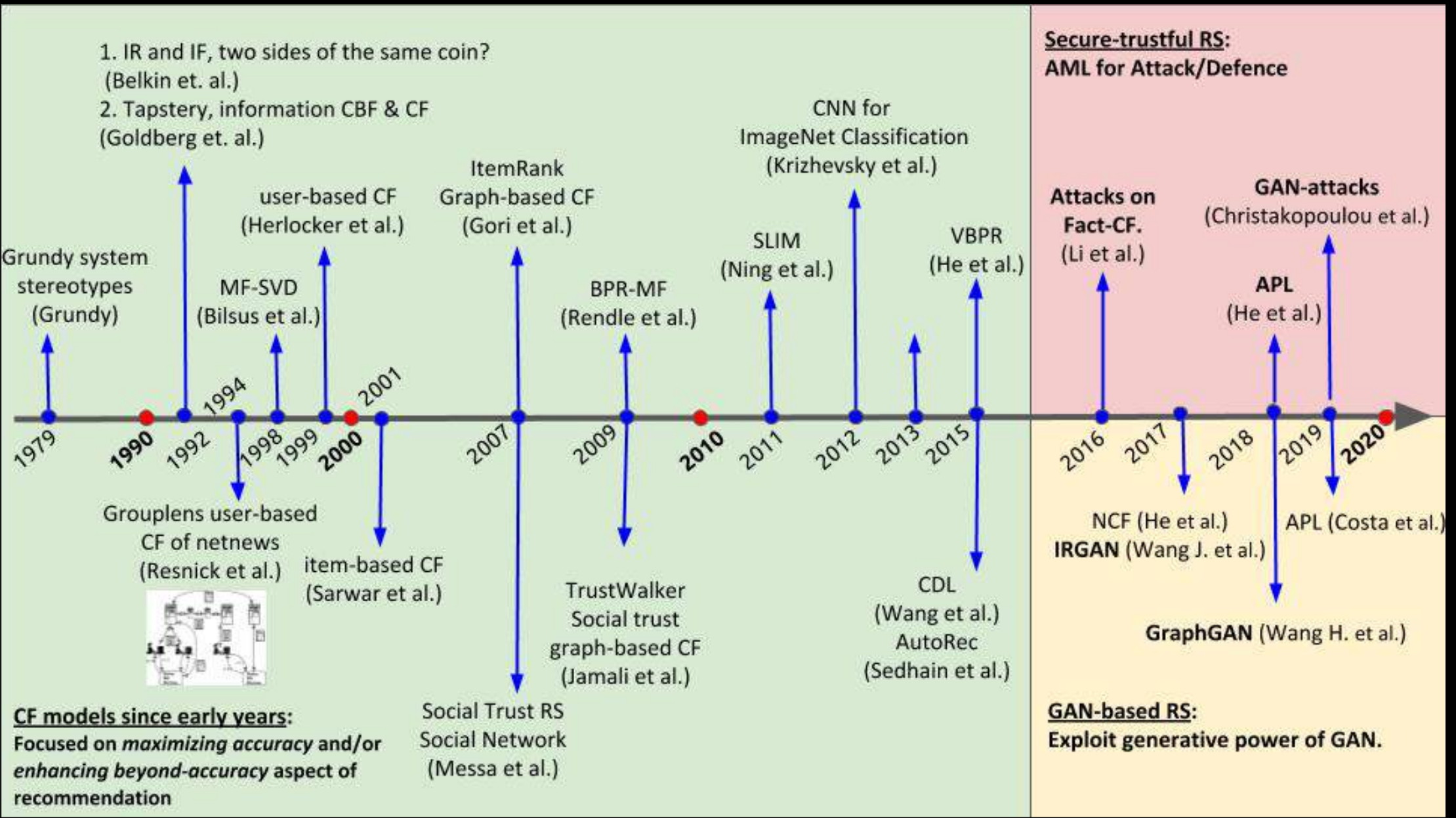
Politecnico di Bari

DOMAIN/TASK-DEPENDENT ERA

- Include sources of **side information**:
  - **User-related** (e.g., demographics, personality traits, social-network information)
  - **Item-related** (e.g., visual features, attributes, description, knowledge graphs)
  - **User-Item** Interplay (e.g., the season of interaction, the number of visits)

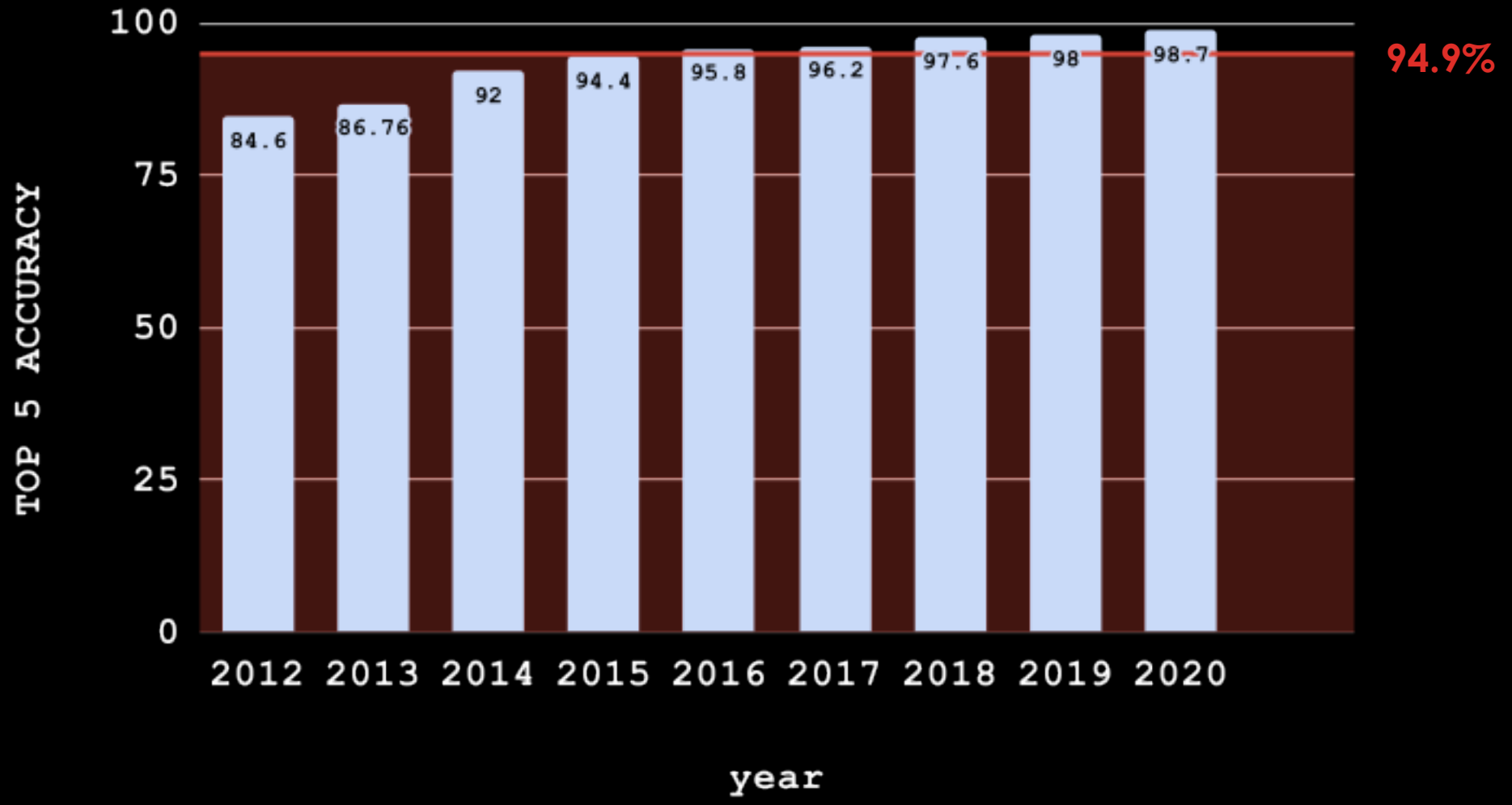
- Based on the unique nature of side information in different domains, different hybrid-CF strategies have been developed

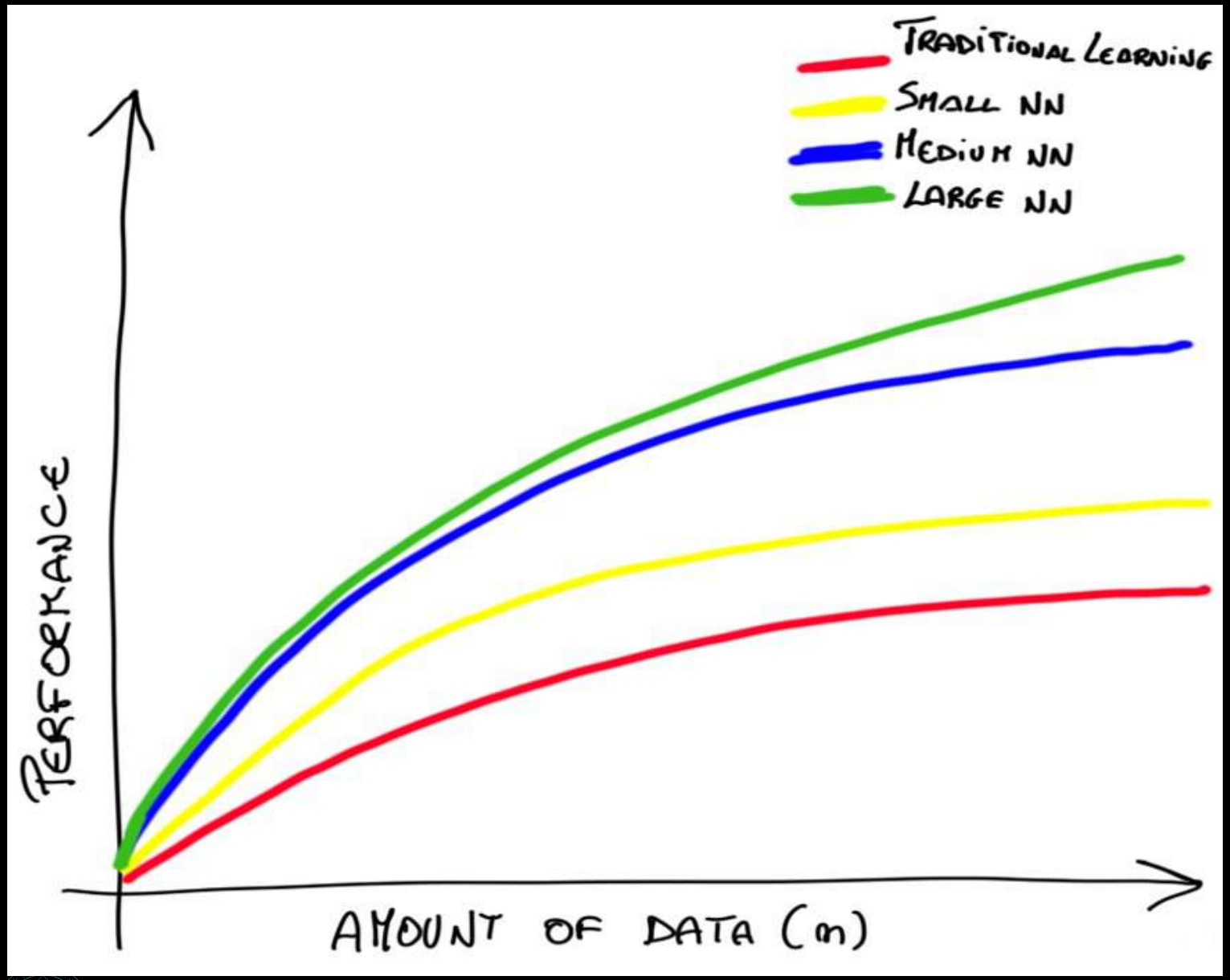
STORY OF RECOMMENDER SYSTEMS



WHY DEEP LEARNING?

ImageNet Visual Recognition





DEEP NEURAL NETWORKS

ARE

UNIVERSAL FUNCTION APPROXIMATORS

DEEP LEARNING IN RS

Discover the **non-linear** and non-trivial user-item relationships	• Capture complex user-item interaction patterns
Extract **latent features**	• Video, images, audio, textual descriptions, etc
Sequence modelling	• RNN and CNN to model user behaviour evolution

DEEP LEARNING IN RS

Discover the **non-linear** and non-trivial user-item relationships	• Capture complex user-item interaction patterns
Extract **latent features**	• Video, images, audio, textual descriptions, etc
Sequence modelling	• RNN and CNN to model user behaviour evolution

DEEP NEURAL CF MODELS

Deep-Learning RS	
CDL [125]	Collaborative Deep Learning (CDL) jointly performs deep representation learning for the item contents and CF for the user feedback.
AutoRec [101]	AutoRec performs performs the recommendation task by exploiting the reconstruction power of auto-encoders.
CDAE [135]	Collaborative Denoising Auto-Encoder (CDAE) generalizes latent factor models by learning full-users preferences thanks to the reconstruction from a sub-set of preferences.
RRN [132]	Recurrent Recommender Networks (RRN) predicts future user preferences by integrating MF with a Long Short-Term Memory (LSTM) to capture dynamics.
NCF [42]	Neural Collaborative Filtering (NCF) learns user-item interaction function from the data by replacing the inner product of MF with a neural architecture.
CVAE [65]	Collaborative Variational Auto-Encoder (CVAE) performs recommendations by learning both deep user-item latent representations from content data and implicit user-item relationships from both content and ratings.

DEEP LEARNING IN RS

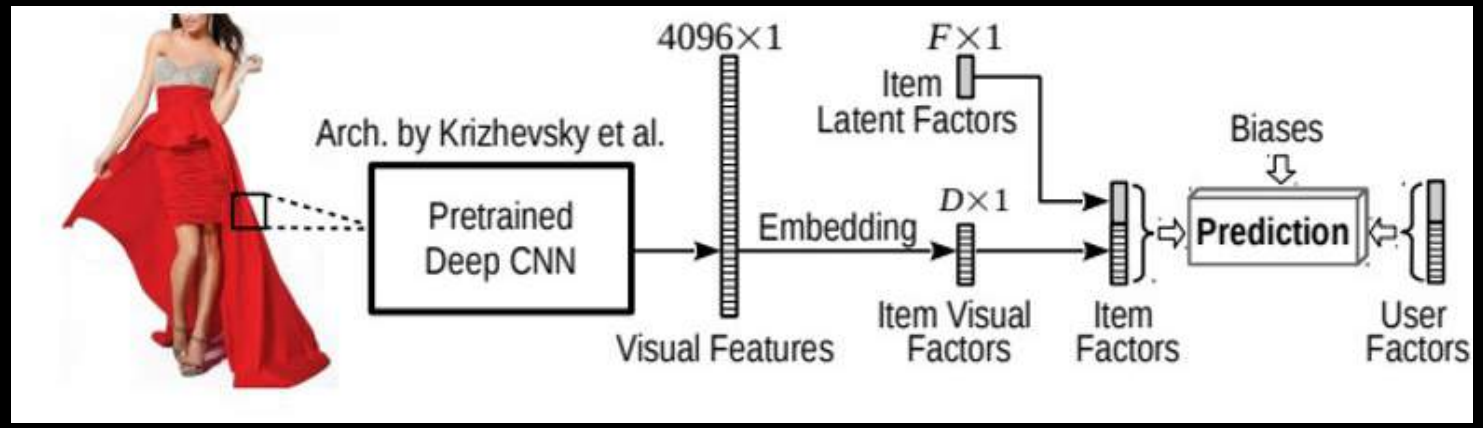
Discover the **non-linear** and non-trivial user-item relationships	• Capture complex user-item interaction patterns
Extract **latent features**	• Video, images, audio, textual descriptions, etc
Sequence modelling	• RNN and CNN to model user behaviour evolution

EXTRACT LATENT FEATURES

VISUAL FEATURE

VBPR

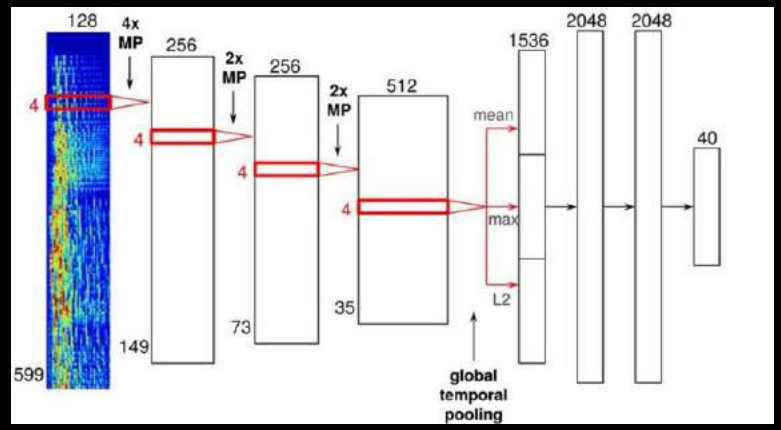
[He and McAuley, 2016]



AUDIO FEATURE

Deep content-based music recommendation

[van den Oord et al., 2016]

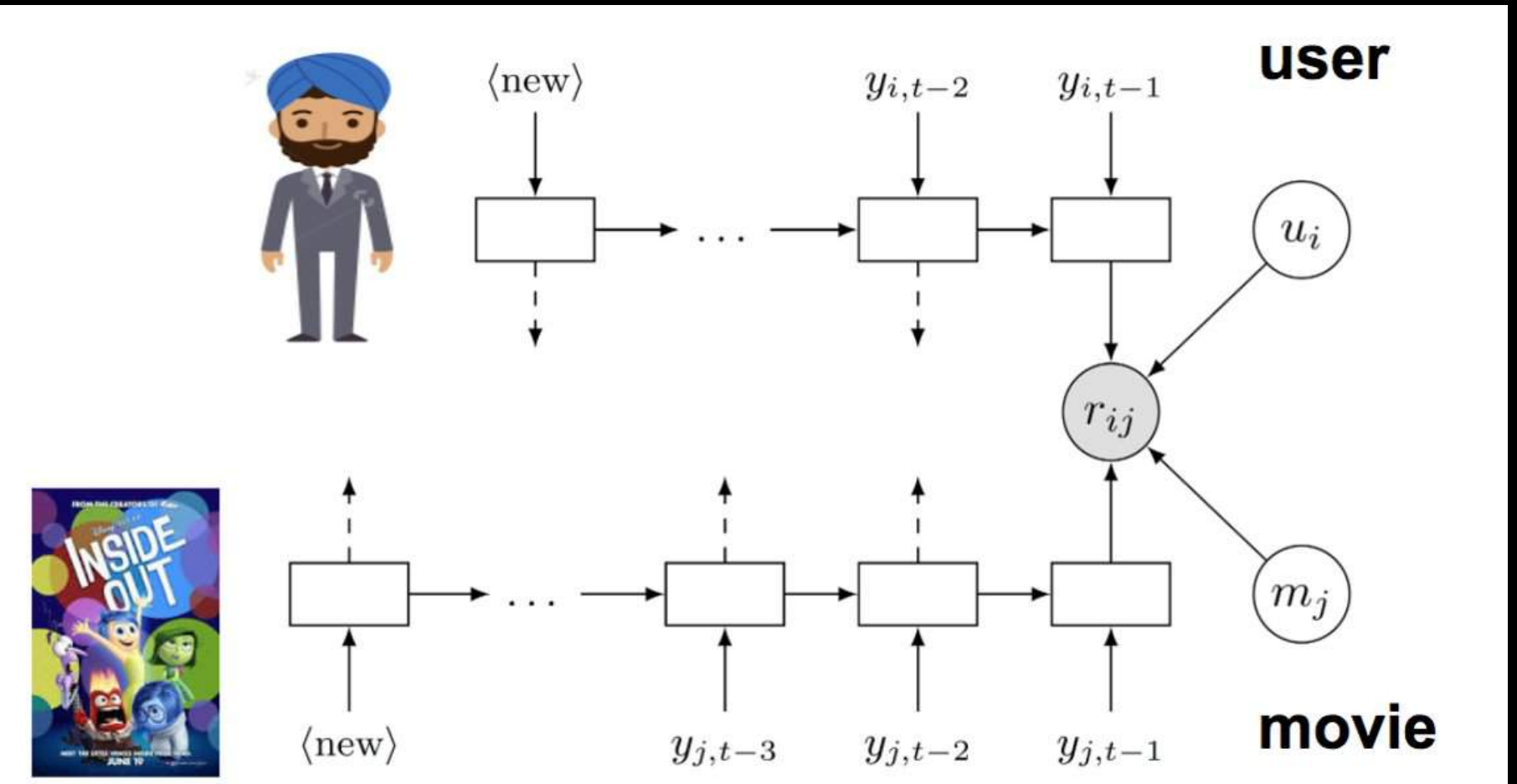


Politecnico
di Bari

DEEP LEARNING IN RS

Discover the **non-linear** and non-trivial user-item relationships	• Capture complex user-item interaction patterns
Extract **latent features**	• Video, images, audio, textual descriptions, etc
Sequence modelling	• RNN and CNN to model user behaviour evolution

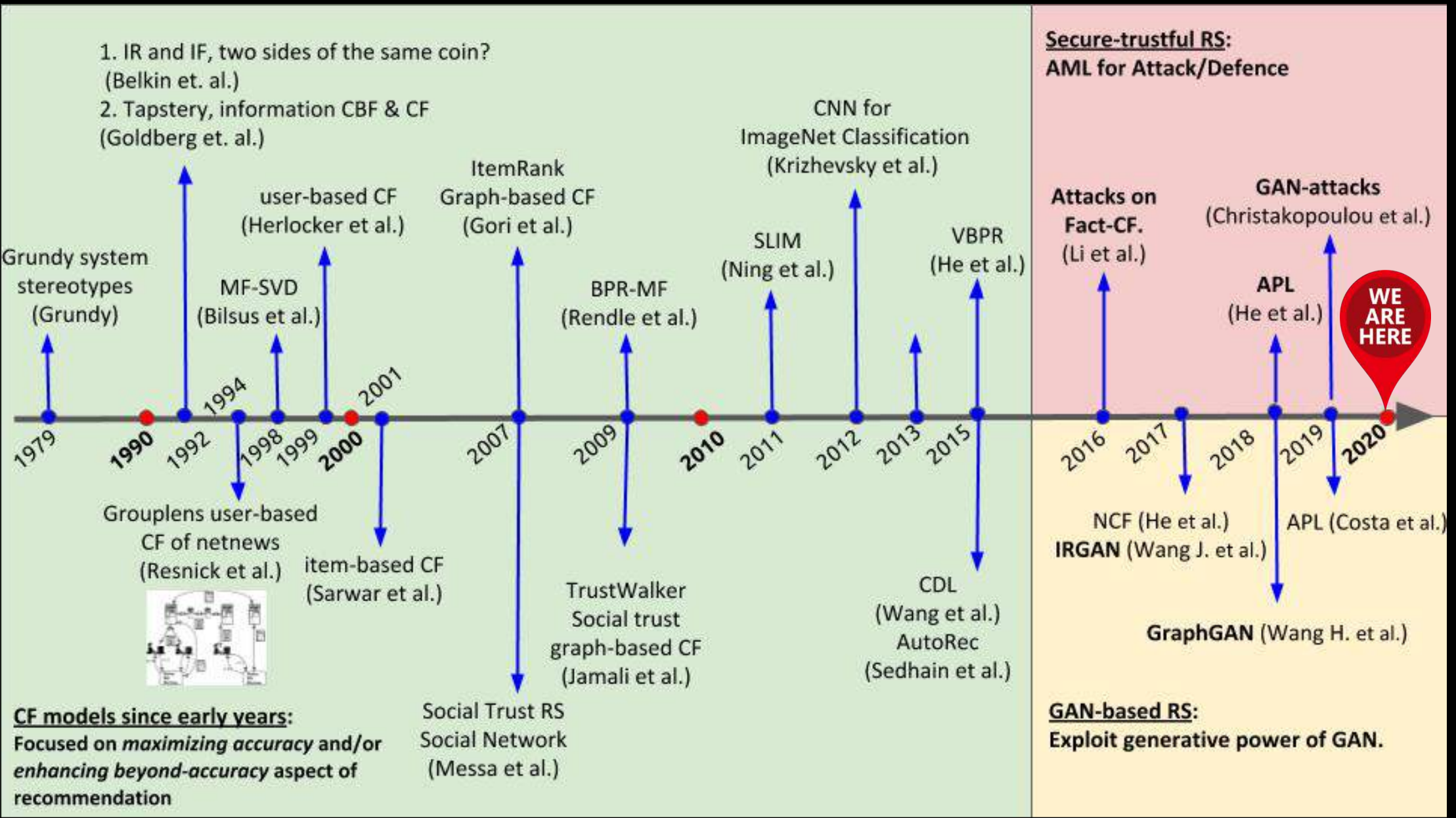
RNN FOR SESSION-BASED RECOMMENDATION



Recurrent Recommender Network

[Chao-Yuan et al., 2017]

STORY OF RECOMMENDER SYSTEMS



1.2 ADVERSARIAL MACHINE LEARNING

Foundations

IT'S ALL ABOUT DATA

«Yet, we found that adversarial examples are relatively robust, and are shared by neural networks with varied number of layers, activations or trained on different subsets of the training data.

That is, if we use one neural net to generate a set of adversarial examples, we find that these examples are still statistically hard for another neural network even when it was trained with different hyperparameters or, most surprisingly, when it was trained on a different set of examples»

[C. SZEGEDY ET AL. INTRIGUING PROPERTIES OF NEURAL NETWORKS, ICLR'14]

ACTUALLY NOT REALLY…

«… adversarial examples can be directly attributed to the presence of *non-robust features*: features (derived from patterns in the data distribution) that are highly predictive, yet brittle and (thus) incomprehensible to humans.»

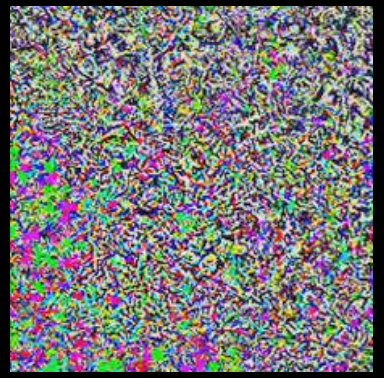
«*Adversarial vulnerability is a direct result of our models' sensitivity to well-generalizing features in the data.*»

«…this perspective establishes adversarial vulnerability as a human-centric phenomenon, since, from the standard supervised learning point of view, non-robust features can be as important as robust ones »

[A. ILYAS ET AL. ADVERSARIAL EXAMPLES ARE NOT BUGS, THEY ARE FEATURES, NIPS'19]

IS THE PANDA SO IMPORTANT?

+

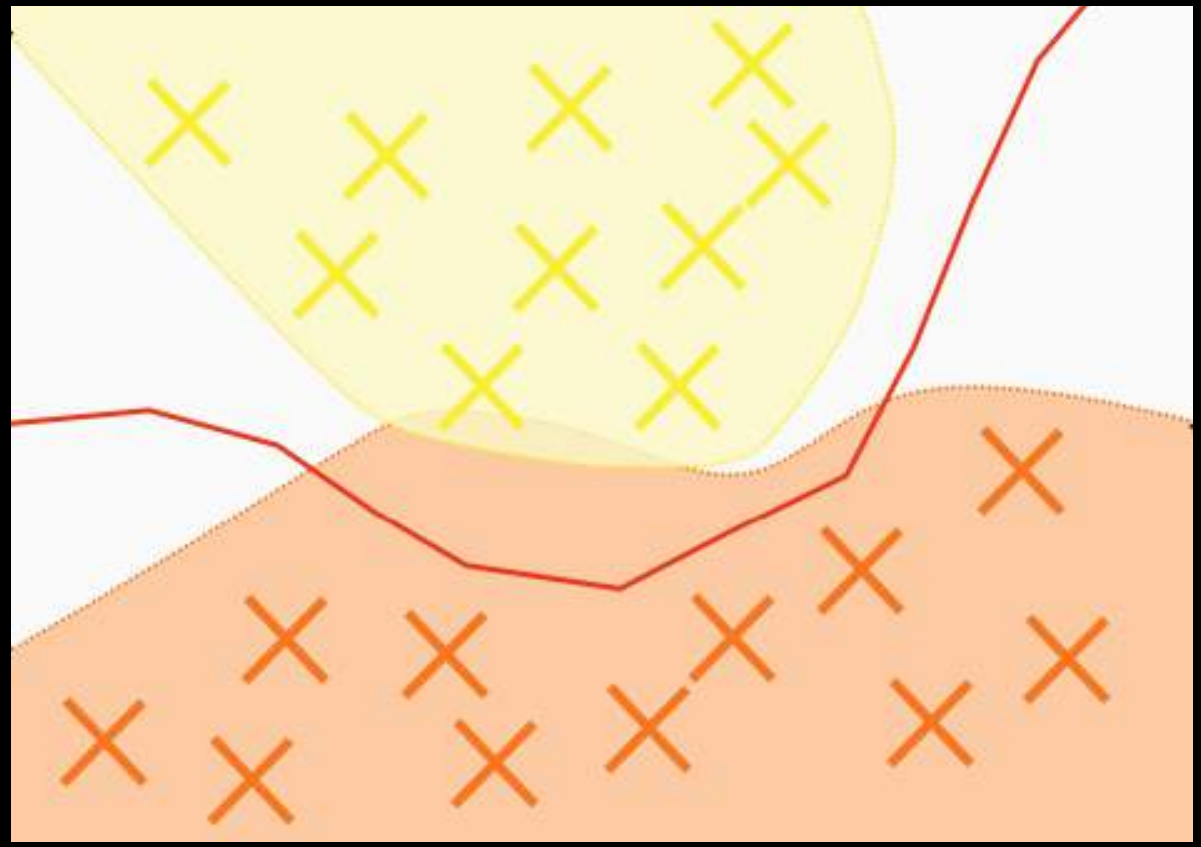


Adversarial Noise

=

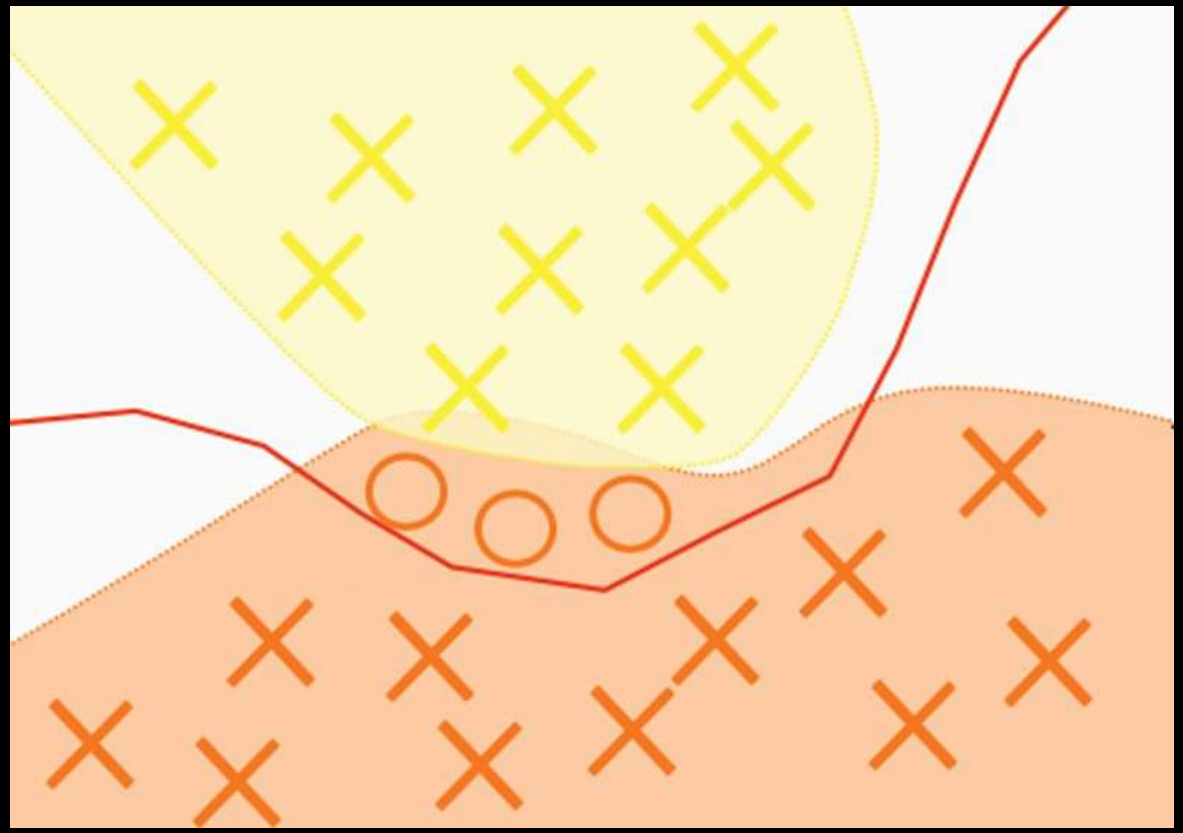
WHAT IS THE ORIGIN OF SUCH FAILURE?

IDEA: DECISION BOUNDARY OF THE MODEL



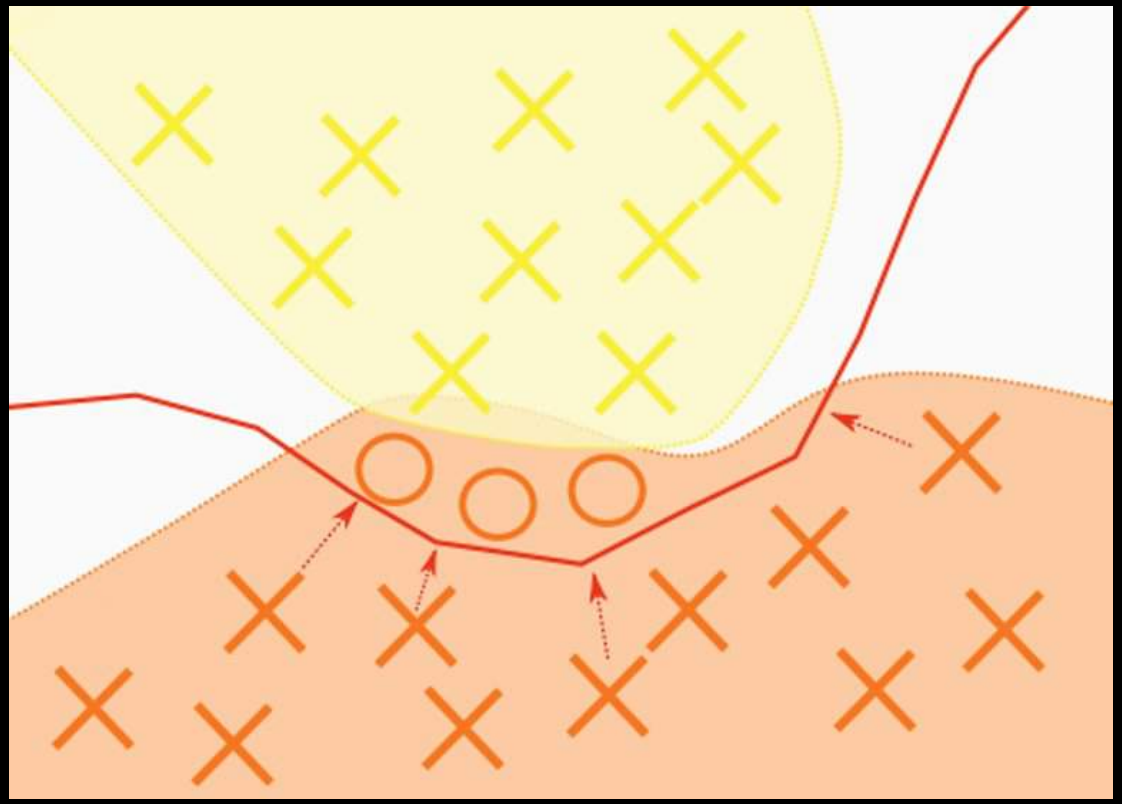
The **learned model** slightly differs from the true data distribution …

IDEA: THE SPACE OF ADVERSARIAL EXAMPLES



…. which makes room for adversarial examples.

ATTACK: USE THE ADVERSARIAL DIRECTIONS



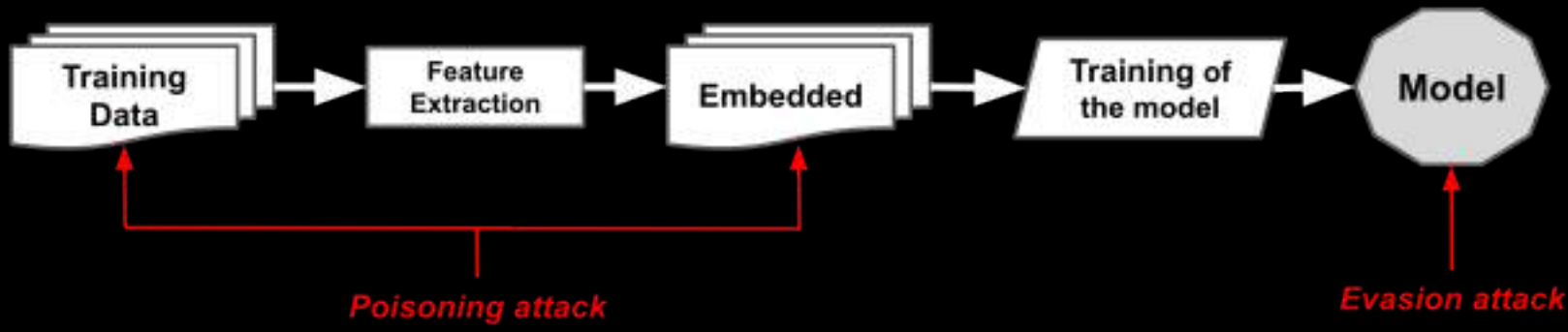
Most attacks try to move inputs across the boundary.

Attacking with a random noise does not work well in real experiment.

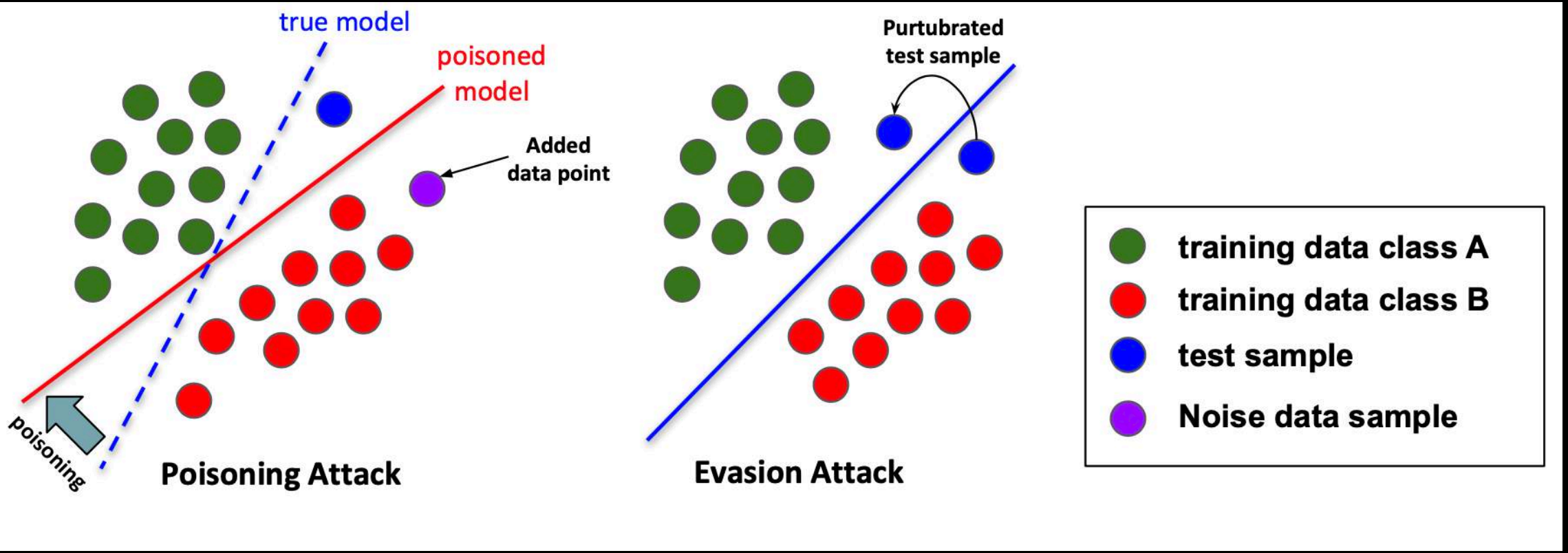
STRATEGIES OF ATTACKS

A crucial distinction between these classes of attacks is

- **Evasion** (decision time) is an attack on the learned model (e.g., a classifier)
- **Poisoning** (training time) is an attack on the algorithm (e.g., BPR) where an adversary introduces malicious samples into the data used for training



EVASION VS. POISONINIG



LEVEL OF KNOWLEDGE OF THE ATTACKER

- **Black-box**: No access to the model only knows the **OUTPUT**
- **White-box**: knowledge of the data, architecture and parameters (**gradient**).
- **Grey-box:** any variation between white- and black-box

Who goes first?

- **Attacker** designs attacks based on the defense strategy
- **Defender** designs defense based on the attacker's knowledge

GOAL OF THE ATTACKER

- **Targeted**
  - Mislead the **classifier** to predict a **target** label
- **Non-targeted**
  - Mislead the **classifier** to predict a **arbitrary** label

****** and in RS? ******

A BIT OF FORMALISM

- Given o model $f(x, \Omega)$ and an input x_0 whose target class is y, find a minimal perturbation Δ_{adv} for x_0 such that the new input $\mathrm{x}' = x_0 + \Delta_{adv}$ is classified as y' with $y \neq y'$

$$\min_{\Delta_{adv}} \|\Delta_{adv}\|$$

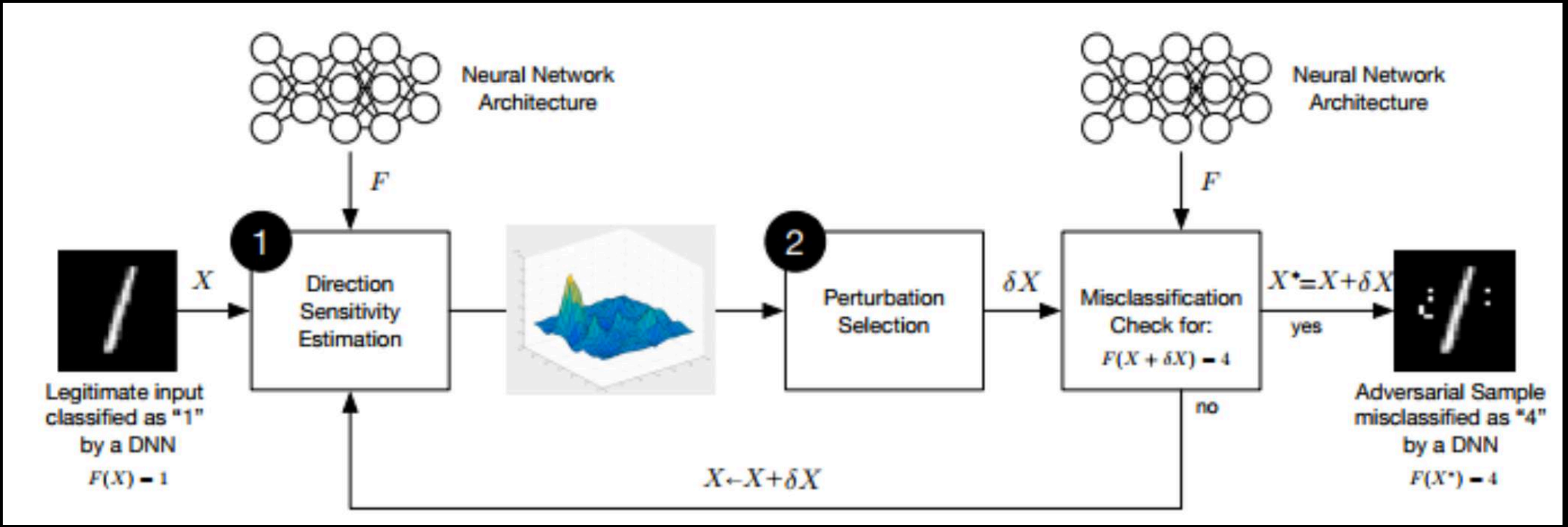
$$s.t.\ f(x_0 + \Delta_{adv}) = y'$$

AS A CONSTRAINT PROBLEM

- Given o model $f(x, \Omega)$ and an input x_0 whose target class is y, find a minimal perturbation Δ_{adv} for x_0 such that the new input $\mathrm{x}' = x_0 + \Delta_{adv}$ is classified as y' with $y \neq y'$

$$\max_{\Delta_{adv}:\|\Delta_{adv}\|\leq\epsilon} J(f(x_0 + \Delta_{adv}, \Omega), \mathrm{y})$$

THE FRAMEWORK



CREDITS: N. Papernot et al. *Distillation as a Defense to Adversarial Perturbations Against Deep Neural Networks*. IEEE Symposium on Security and Privacy, SP 2016

ATTACK METHODS

- **L-BFGS** [Szegedy et al., 2013]
  - Uses Limited-memory BFGS (L-BFGS) algorithm to solve the problem with **linear memory requirement**
- **FSGM -** Fast Gradient Sign Method [Goodfellow et al., ICLR '15]
  - Use the **gradient** of the loss function to work on the costraint problem.
- **Carlini-Wagner** [Carlini and Wagner, 2017a]
  - Refine the L-BFGS attack to **defeat Defensive distillation**
- **JSMA -** Jacobian Saliency Map Attack [Papernot et al., 2015a]
  - Construct an **input-output mapping** (forward derivatives) to find the minimal perturbation
- **DeepFool** [Moosavi-Dezfooli et al., 2015]
  - Perform an iterative attack to find **the closest decision boundary** to a given input

FGSM(INTUITION 1)
[GOODFELLOW ET AL., ICLR'15]

$$x' = x_0 + \Delta_{adv}$$

$$\omega^T \cdot x' = \omega^T \cdot x_0 + \omega^T \cdot \Delta_{adv}$$

GOAL: Maximize the factor $\omega^T \cdot \Delta_{adv}$ under the constraint $\|\Delta_{adv}\|_\infty \leq \epsilon$

IDEA: $\Delta_{adv} = \epsilon \cdot sign(\omega)$

FGSM(INTUITION 2)
[GOODFELLOW ET AL., ICLR'15]

$$\omega^T \cdot \Delta_{adv} = \epsilon \cdot (|\omega_1| + \cdots + |\omega_n|)$$

If we consider $m = avg(\omega_i)$ then the influence of the perturbation $\omega^T \cdot \Delta_{adv}$ grows as $\epsilon \cdot m \cdot n$

RESULT: the perturbation linearly grows with n

FGSM: FAST GRADIENT SIGN METHOD

[GOODFELLOW ET AL., ICLR'15]

Let Ω be the parameters of a ML model, x the input to the model, y the label of x, and $J(\Omega, x, y)$ be the loss function used for the model.

FSGM linearizes J around the fixed value of Ω, obtaining an optimal max-norm constrained adversarial perturbation.

$$\Delta_{avd} = \epsilon \cdot \mathrm{sign}\left(\nabla_{\boldsymbol{x}} J(\boldsymbol{\Omega}, \boldsymbol{x}, y)\right)$$

COUNTERMEASURES

- **Proactive** countermeasures
  - **Adversarial Training** [Goodfellow et al., ICLR '15]
    - Additional training epochs with adversarial exmaples
  - **Defensive Distillation** [Papernot et al., ISS'16]
    - Adapt distillation to increase the robusteness of the network
  - **Robust Optimization** [Madry et al., ICLR'18]
    - design robust DNN to prevent a speciic class of adversarial examples
- **Reactive** countermeasures
  - **Adversarial Detecting**
  - **Input Reconstruction**
  - **Network Verification**

ADVERSARIAL TRAINING
[GOODFELLOW ET AL., ICLR'15]

Including adversarial samples in the **training** of a model makes it **more robust**.
The objective function of the model **adversarially-trained** is:

Adversarial Perturbation

$$J(\boldsymbol{\Omega}, \boldsymbol{x}, y) + \lambda J(\boldsymbol{\Omega}, \boldsymbol{x} + \epsilon \cdot \operatorname{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\Omega}, \boldsymbol{x}, y)))$$

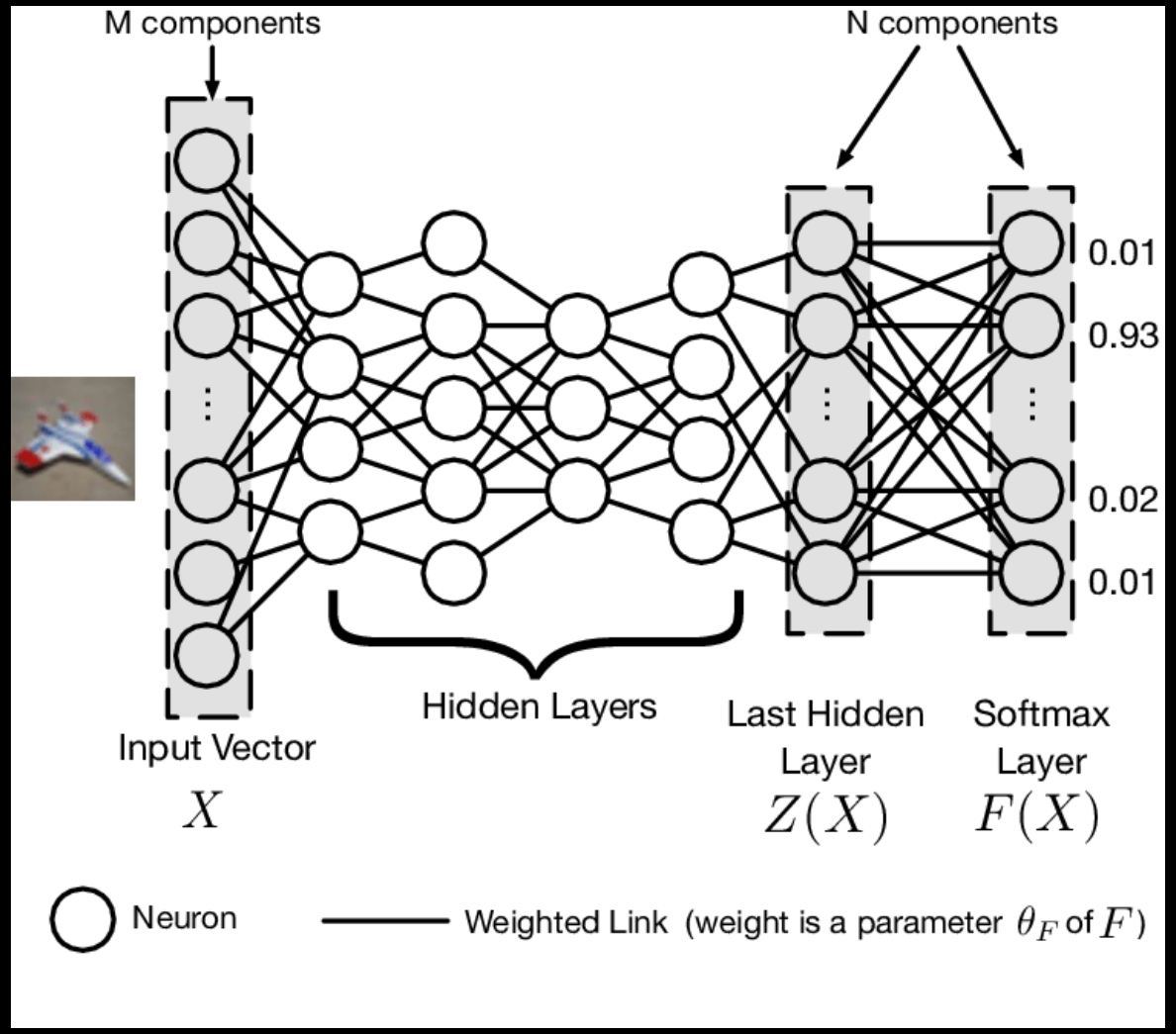
Adversarial Regularization term

Adversarial training provides:

- regularization for DNN
- better generalization performance [Miyato et al., ICLR'17]

DEFENSIVE DISTILLATION

[PAPERNOT ET AL., ISS'16]

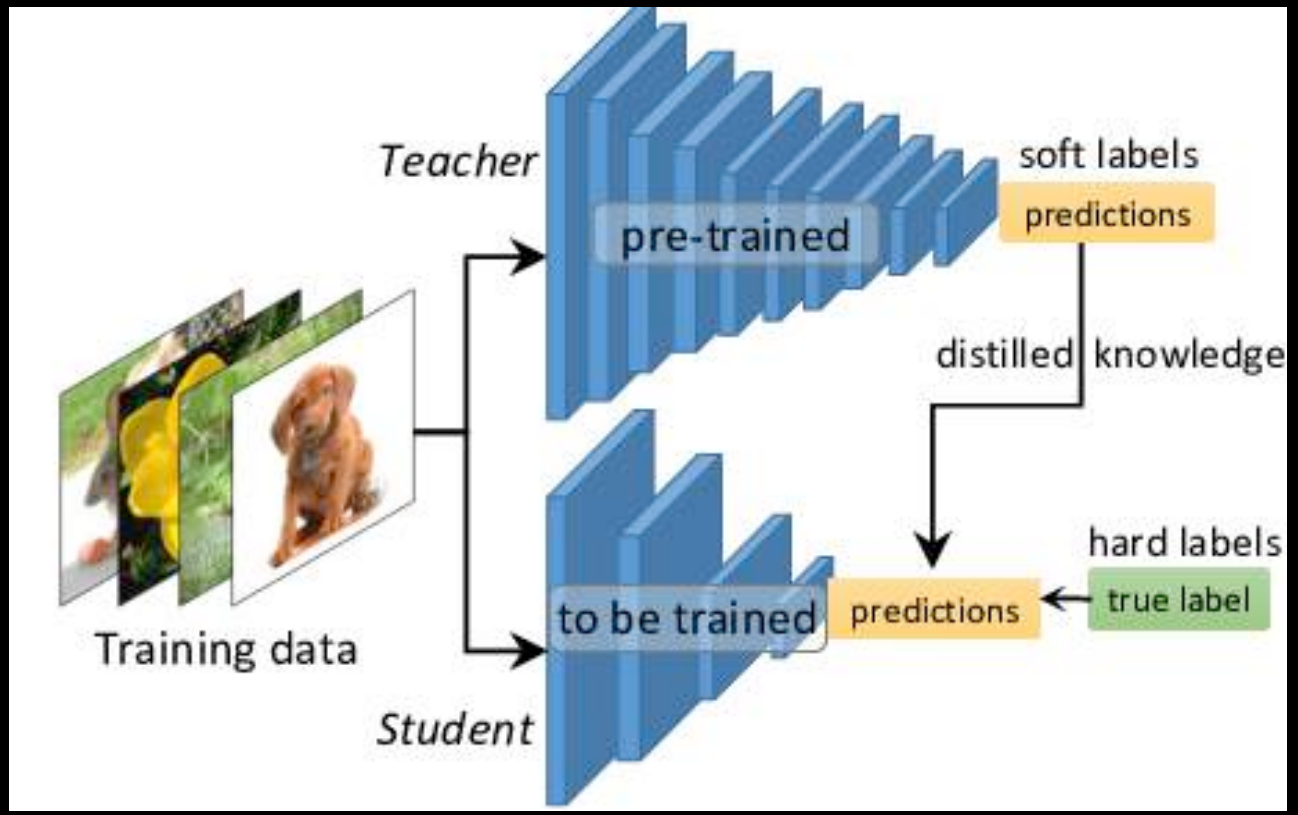


$$F(X) = \left[\frac{e^{z_i(X)/T}}{\sum_{l=0}^{N-1} e^{z_l(X)/T}} \right]_{i \in 0..N-1}$$

Politecnico di Bari

DEFENSIVE DISTILLATION

[PAPERNOT ET AL., ISS'16]

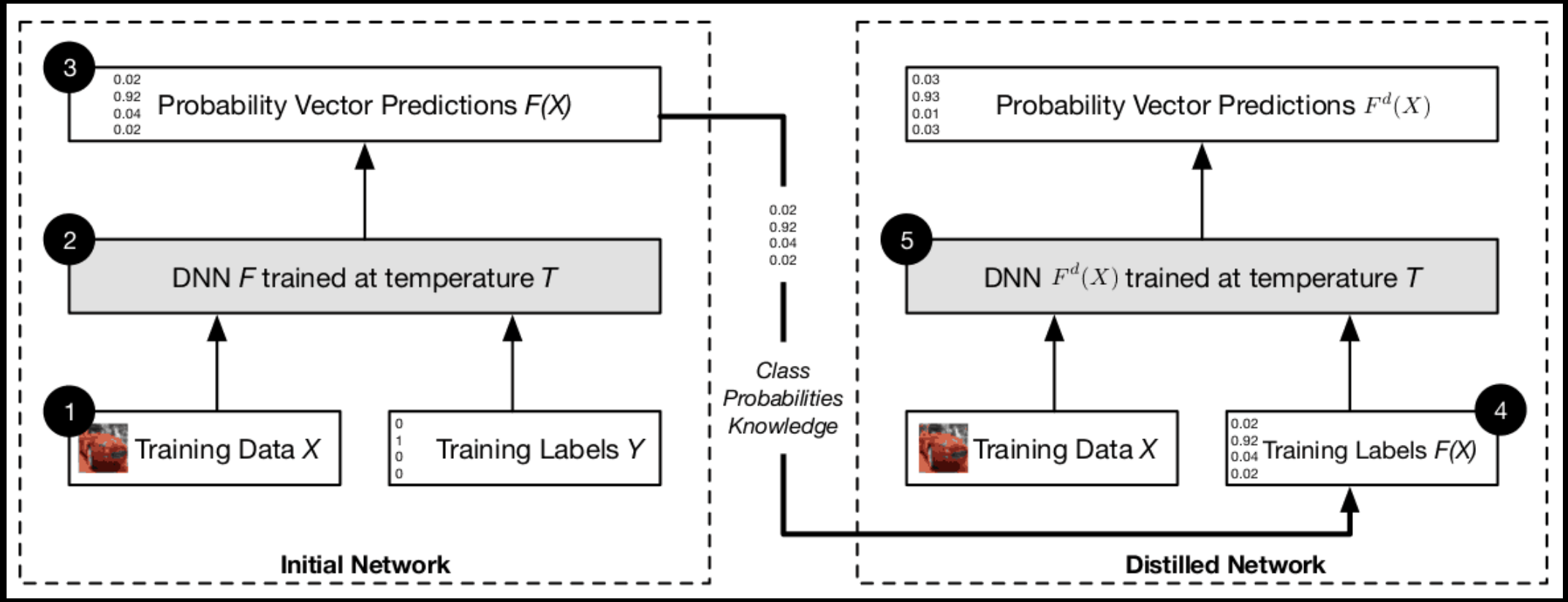


Credits: https://towardsdatascience.com/knowledge-distillation-simplified-dd4973dbc764

Originally proposed for model compression

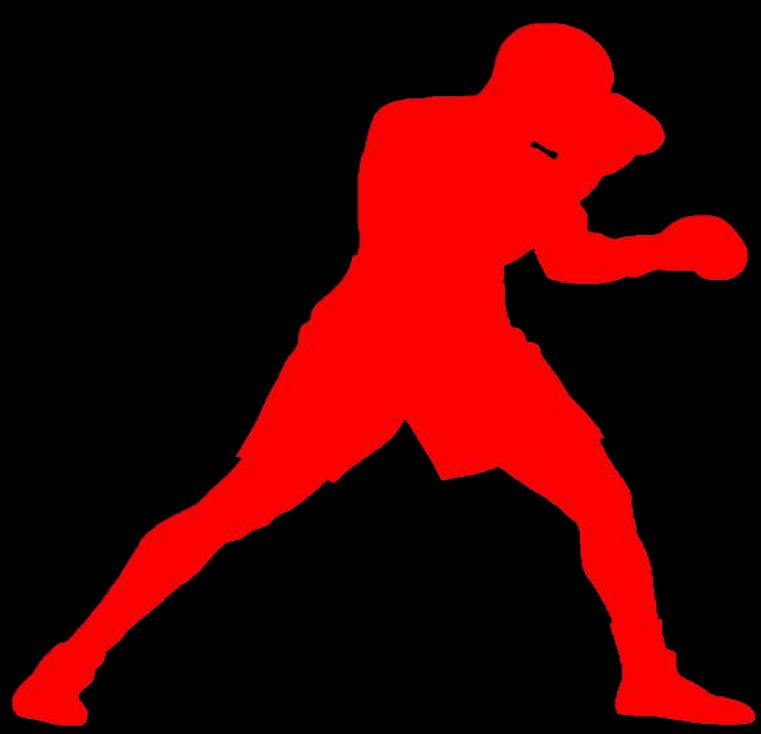
DEFENSIVE DISTILLATION

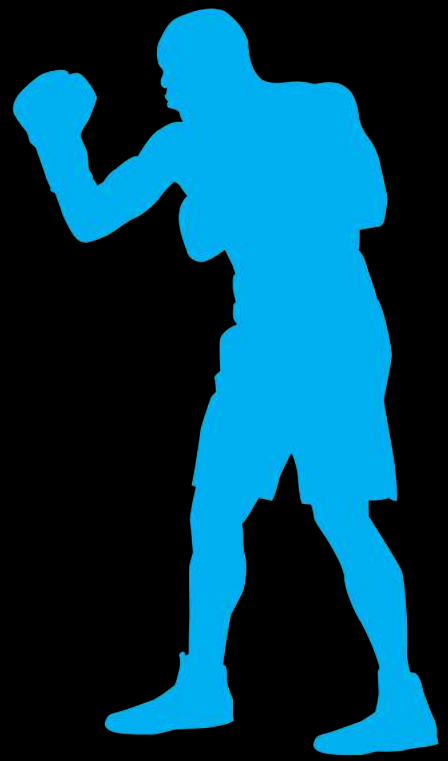
[PAPERNOT ET AL., ISS'16]



Politecnico di Bari

ATTACK-DEFENSE GAME



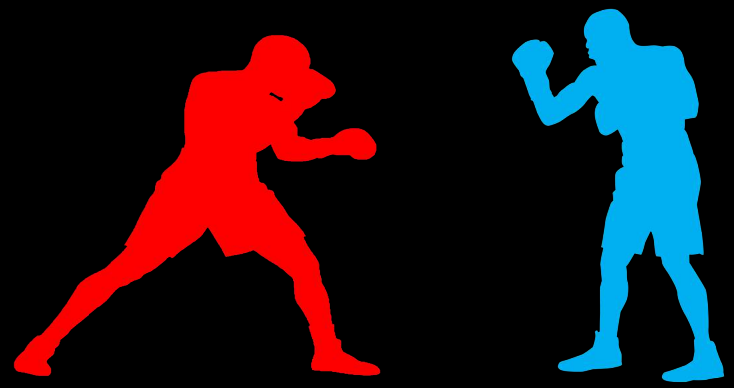


Politecnico
di Bari

ATTACK-DEFENSE GAME

The attack-defence game is a **MINIMAX GAME:**

- in the **literature** for each **ATTACK** there is a **DEFENSE** strategy
- in the **adversarial training** we **MINIMIZE** the Loss and **MAXIMIZE** the perturbation
- **Generative Adversarial Network (GAN)** is trained with an **ATTACKER** that try to alter a **DEFENDER**



Politecnico di Bari

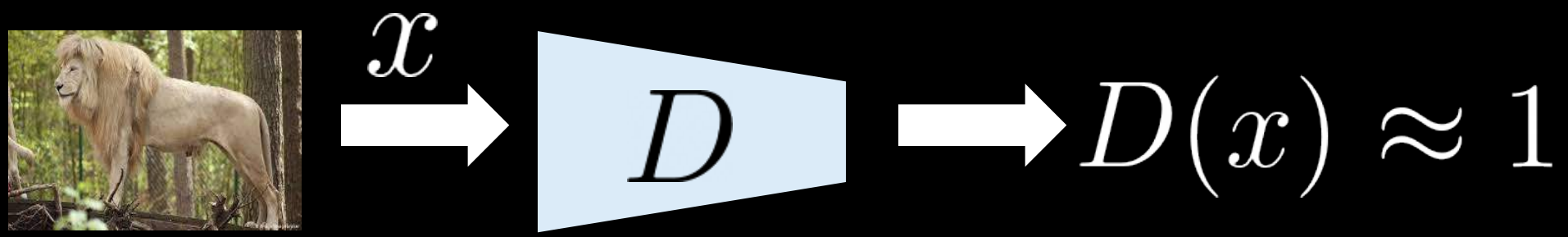
WHY STUDY GAN?

- Generate Examples for Image Datasets
- Generate Photographs of **Faces**
- Generate **Realistic** Photographs
- Generate Cartoon Characters
- **Pose Guided Person** Image Generation
- **Image-to-Image** Translation
- Text-to-Image Translation
- Semantic-Image-to-Photo Translation

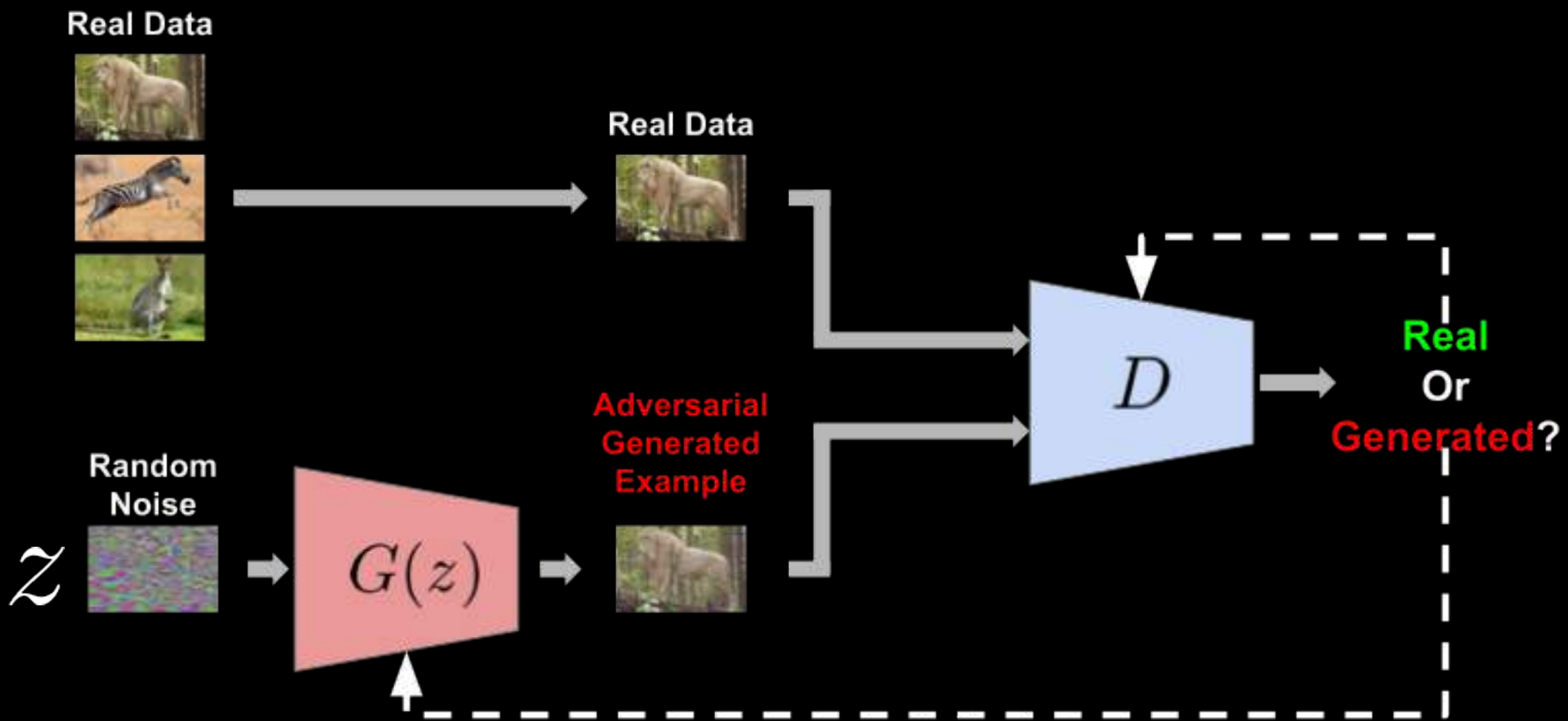
- Photos to Emojis
- Photograph Editing
- High-resolution image synthesis
- Face Aging
- Photo Blending
- Super Resolution
- Photo Inpainting (Style-transfer)
- **Clothing Translation**
- Video-motion Prediction
- 3D Object Generation

HOW DO GANS WORK?

Normal Classifier

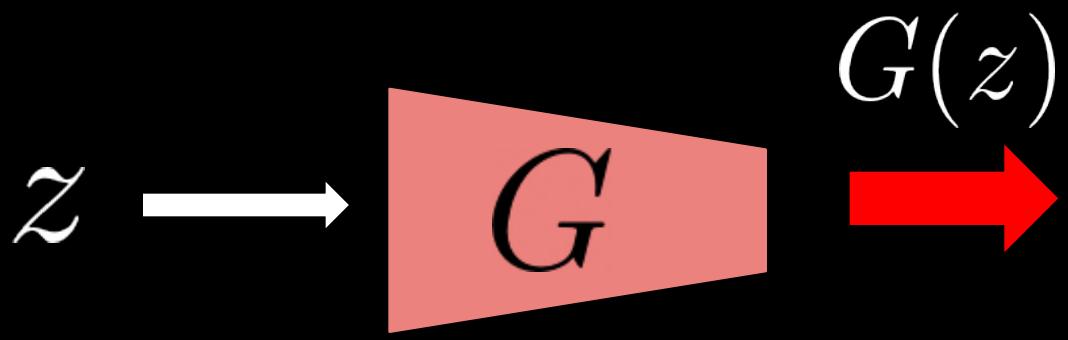


HOW DO GANS WORK?



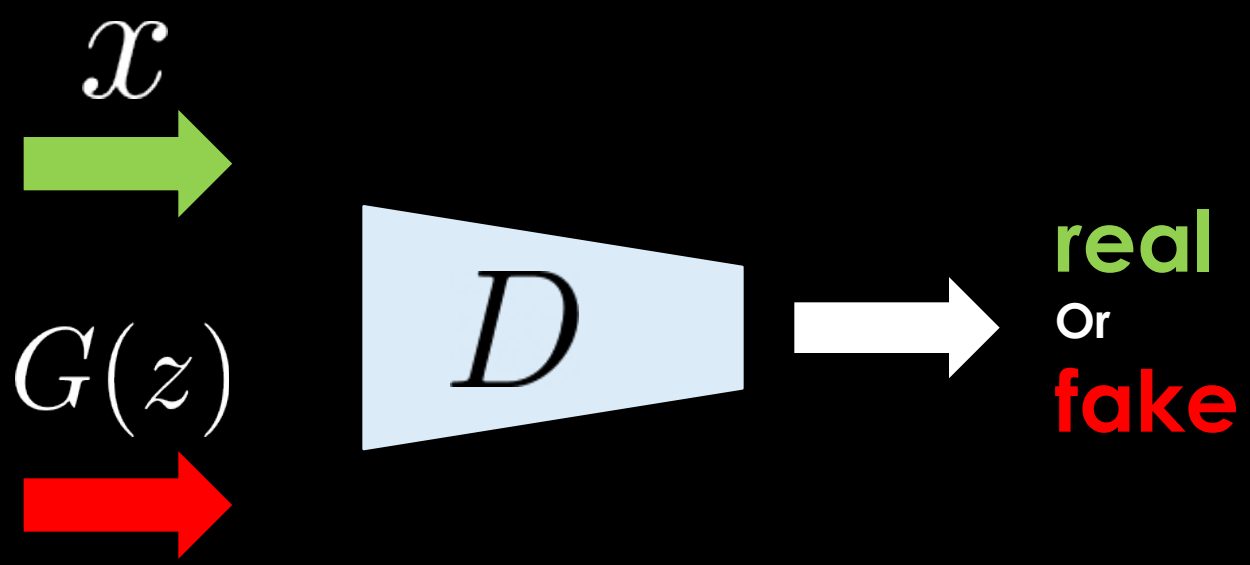
GENERATOR

GOAL? Maps a gaussian random noise z to a **Fake** data to **FOOL** the discriminator

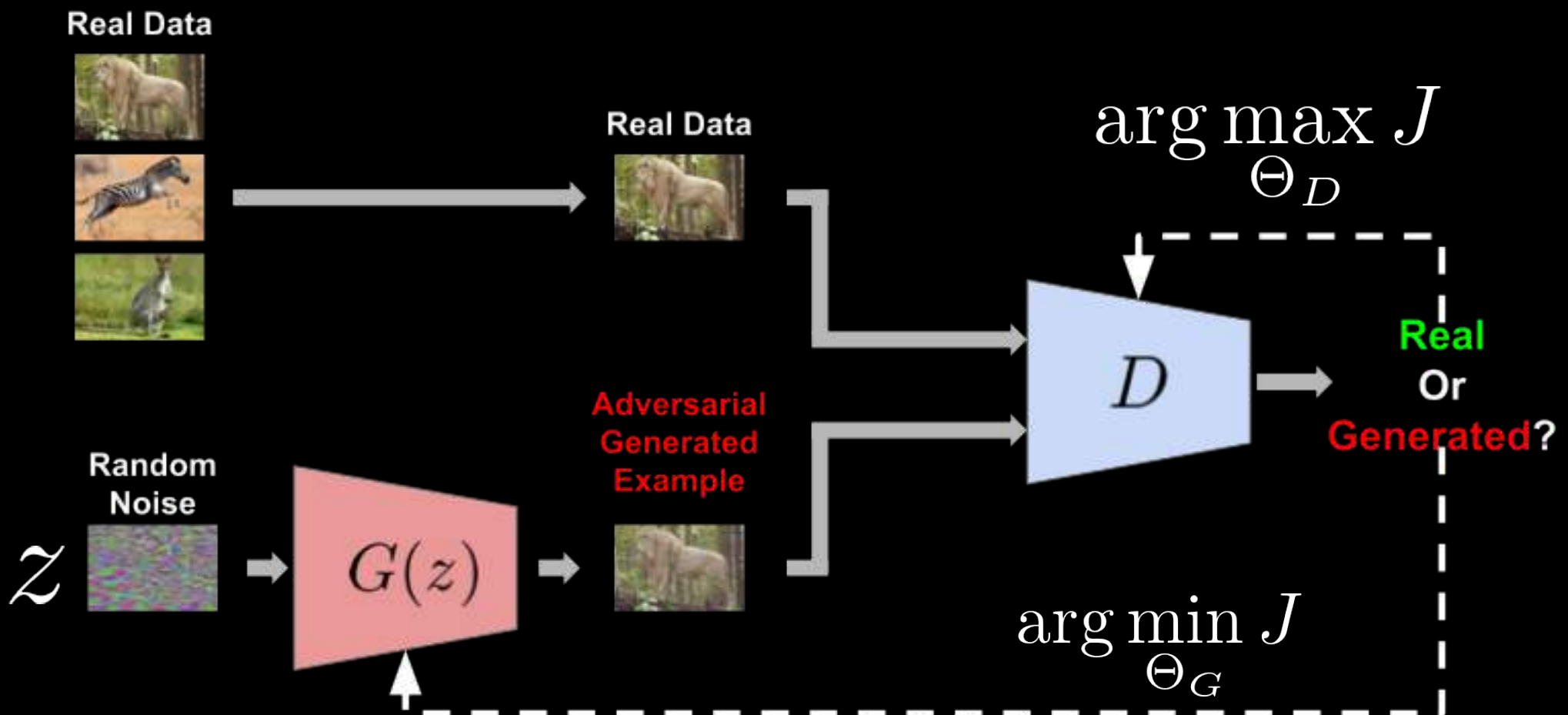


DISCRIMINATOR

GOAL? Output the probability that its input is **real** rather than **fake**



HOW DO GANS WORK?



MINIMAX GAME

$$J = \mathbb{E}_{\boldsymbol{x} \sim \rho_{\text{data}}\,(\boldsymbol{x})} \left[\log D(\boldsymbol{x})\right] + \mathbb{E}_{\boldsymbol{z} \sim p_{\boldsymbol{z}}(z)} \left[\log\left(1 - D\left(G(\boldsymbol{z})\right)\right)\right]$$

DISCRIMINATOR

$$\arg\max_{\Theta_D} J$$

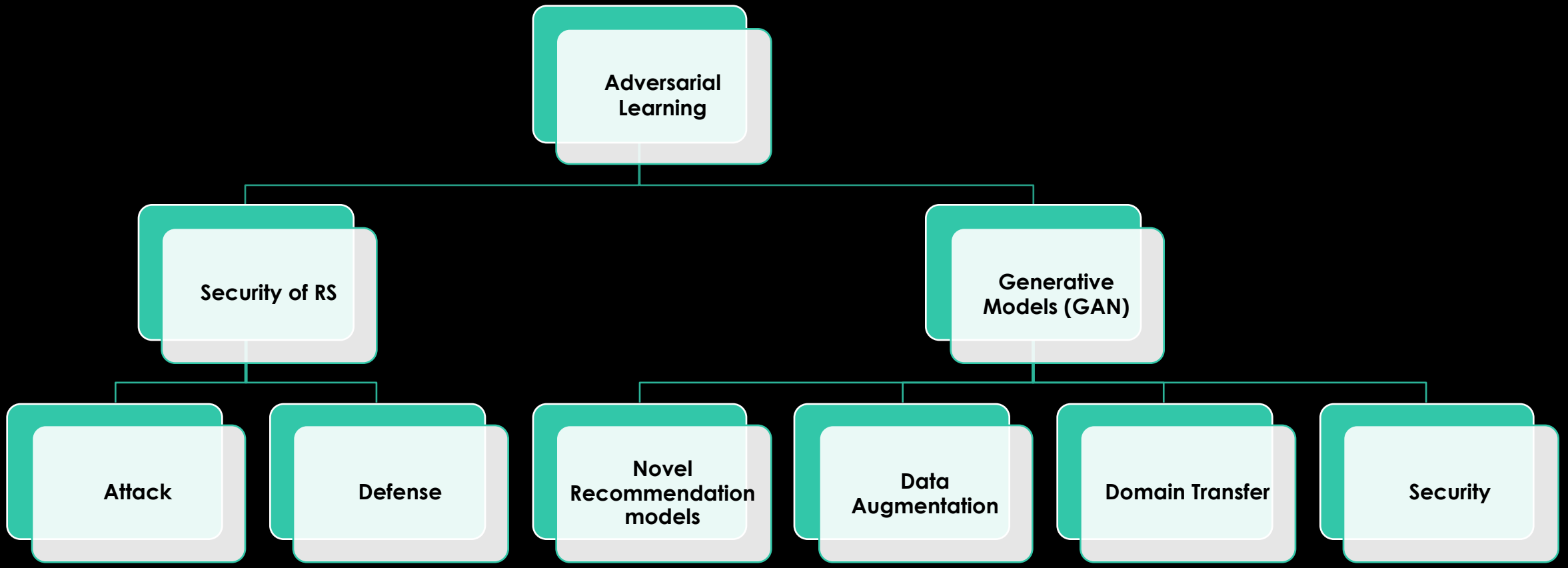
GENERATOR

$$\arg\min_{\Theta_G} J$$

MINIMAX GAME

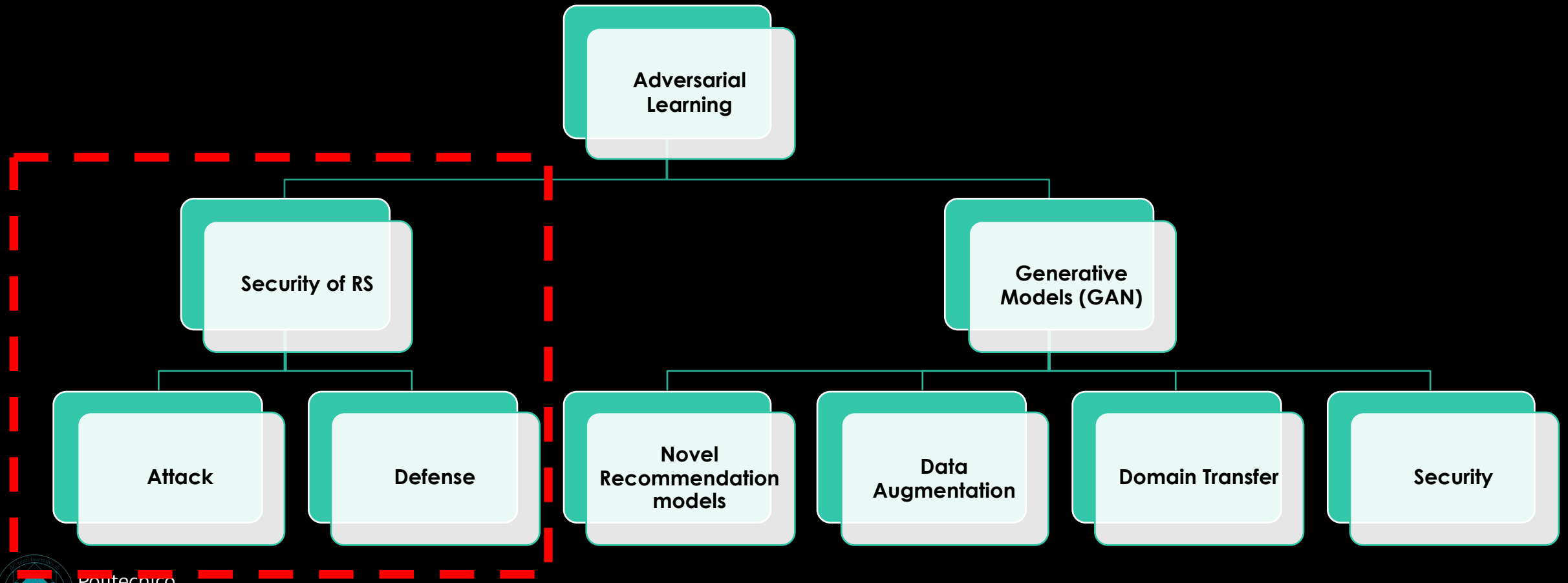
$$\arg\min_{\Theta_G} \max_{\Theta_D} J$$

ADVERSARIAL LEARNING FOR RS

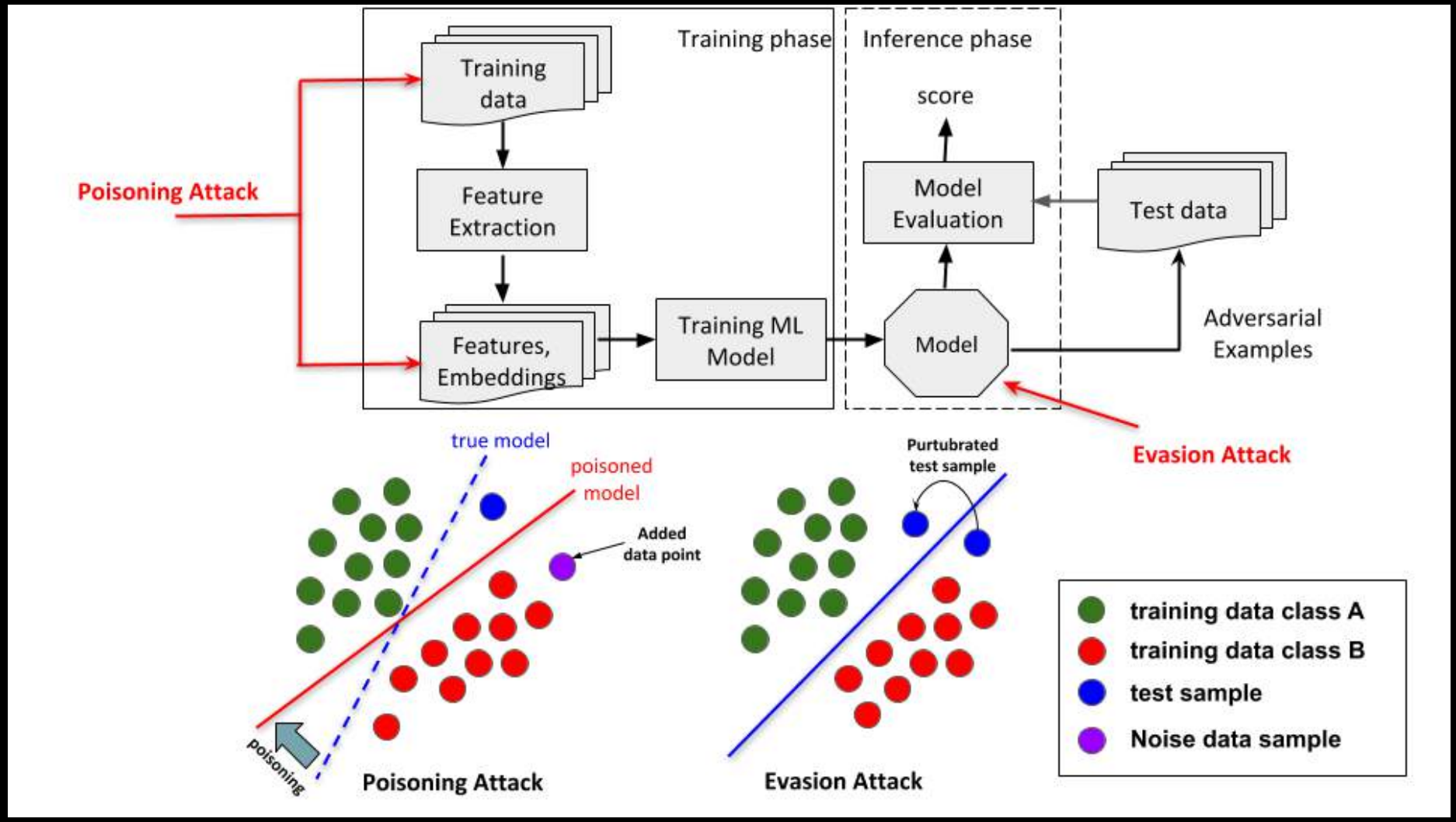


PART 2
ADVERSARIAL LEARNING FOR ATTACK AND DEFENSE RS

ADVERSARIAL LEARNING FOR RS



Attacks Against ML Models



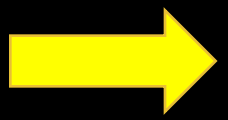
POISION ATTACK APPLICATION IN LITERATURE

- Attack on Binary Classification
  - Label flipping attack
  - Kernel SVM
- Attack on unsupervised learning
  - Clustering
  - Anomaly detection
- Attack on matrix completion
    - Alternating minimization
      - Projected gradianet decent (PGA)
    - Nuclear norm normalization
    - Mimicking user behavior

Vorobeychik, Y., & Kantarcioglu, M. (2018). Adversarial machine learning. Synthesis Lectures on Artificial Intelligence and Machine Learning, 12(3), 1-169.

POISION ATTACK APPLICATION IN LITERATURE

- Attack on Binary Classification
  - Label flipping attack
  - Kernel SVM
- Attack on unsupervised learning
  - Clustering
  - Anomaly detection
- Attack on matrix completion
  - Alternating minimization
    - Projected gradianet decent (PGA)
  - Nuclear norm normalization
  - Mimicking user behavior



Commonly known as "shilling attack" in RecSys.

Vorobeychik, Y., & Kantarcioglu, M. (2018). Adversarial machine learning. Synthesis Lectures on Artificial Intelligence and Machine Learning, 12(3), 1-169.

SHILLING ATTACK ON RS

- Handcrafted excucted by adding fake user profiles
- Typically applied aganist rating-based CF
- Attacker can have knowledge on
  - Training data
  - Recommendation model
- Different attack types are constudtrected based on composition of a user profile

I_S			I_F			$I_\emptyset$			I_T
$i_s^{(1)}$	...	$i_s^{(\alpha)}$	$i_f^{(1)}$	...	$i_f^{(\phi)}$	$i_\emptyset^{(1)}$	...	$i_\emptyset^{(\chi)}$	i_t

	3			5	
		4			
			2		4
	1		5	5	
			4	4	
	2				5

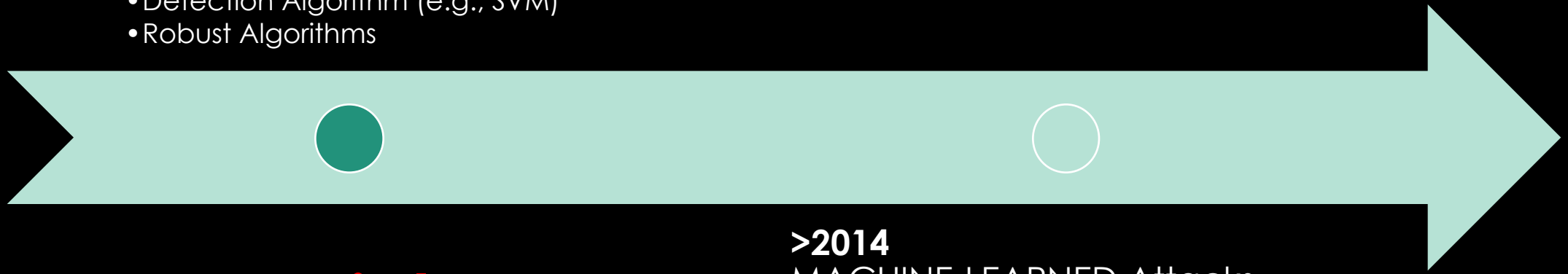
fake user profiles

[Gunes, Ihsan, et al. "Shilling attacks against recommender systems: a comprehensive survey." Artificial Intelligence Review 42.4 (2014): 767-799.]

SECURITY OF RS

Early 2000s
Hand-Engineered Attacks:

- Shilling Profile strategies
- Detection Algorithm (e.g., SVM)
- Robust Algorithms



Tutorial Focuses Here

>2014
MACHINE-LEARNED Attacks

- **Optimized Data Poisoning** attack
- **Adversarial Perturbation on embeddings**
- **Adversarial Regularization**
- **…**

Politecnico di Bari

2.1 ADVERSARIAL RECOMMENDATION FRAMEWORK (ADV-RF)

SECURITY FROM ADVERSARIAL ATTACKS

"*Existing methods that generate adversarial examples for an image classifier are* ***inappropriate***" for recommender systems

*[He, Xiangnan, et al. "**Adversarial personalized ranking for recommendation**." The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. ACM, 2018.]*

SECURITY FROM ADVERSARIAL ATTACKS

Add **adversarial noises** to an input image shall **not change** its **visual content**

"panda"

Adversarial Perturbations

"gibbon"

Politecnico di Bari

SECURITY FROM ADVERSARIAL ATTACKS

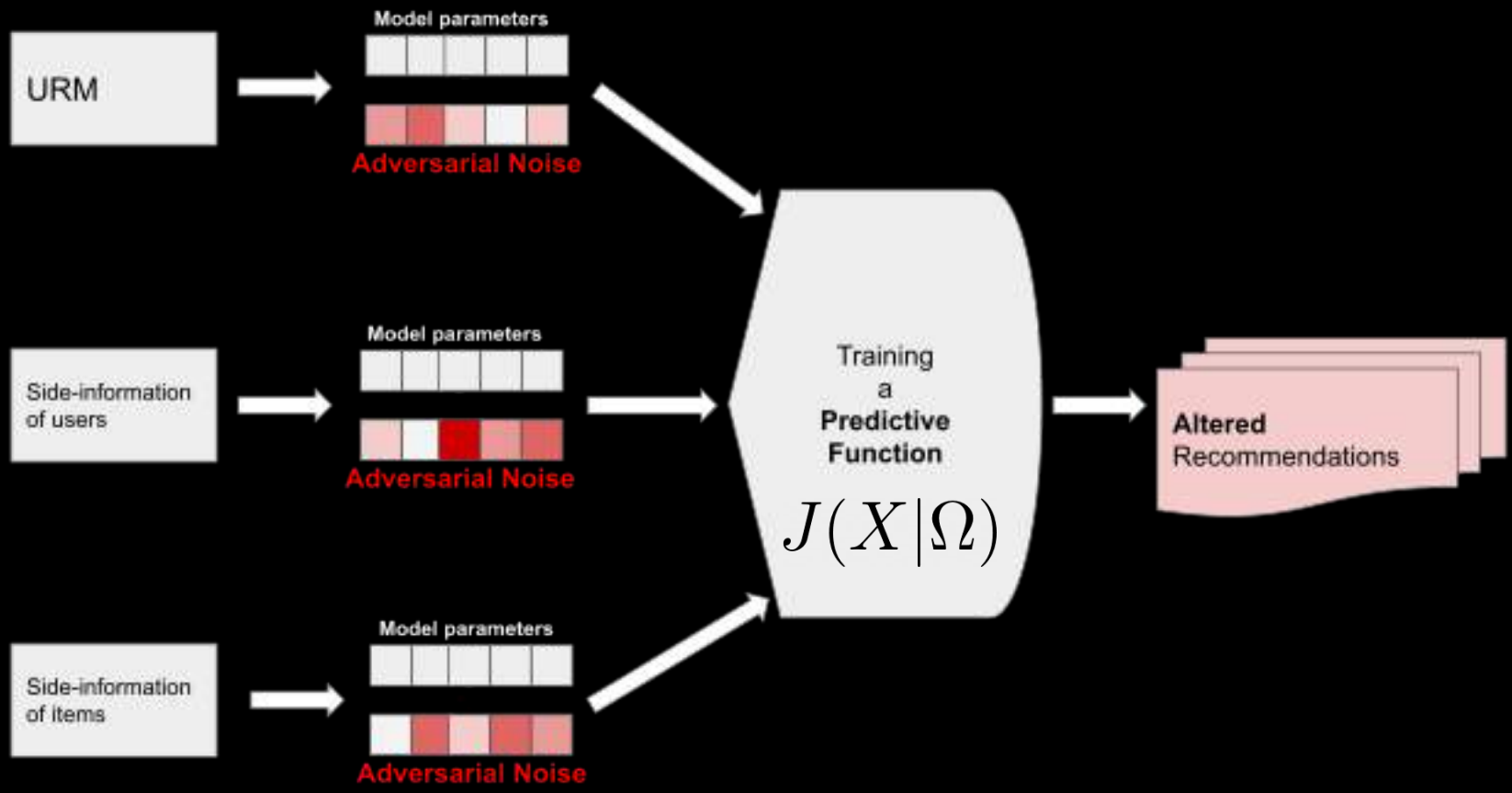
Add **adversarial noises** to a User Profile **changes** its **semantic content**

WHERE DO WE ADD ADVERSARIAL NOISE?

The adversarial perturbations are added to the **embeddings** of Recommender Models

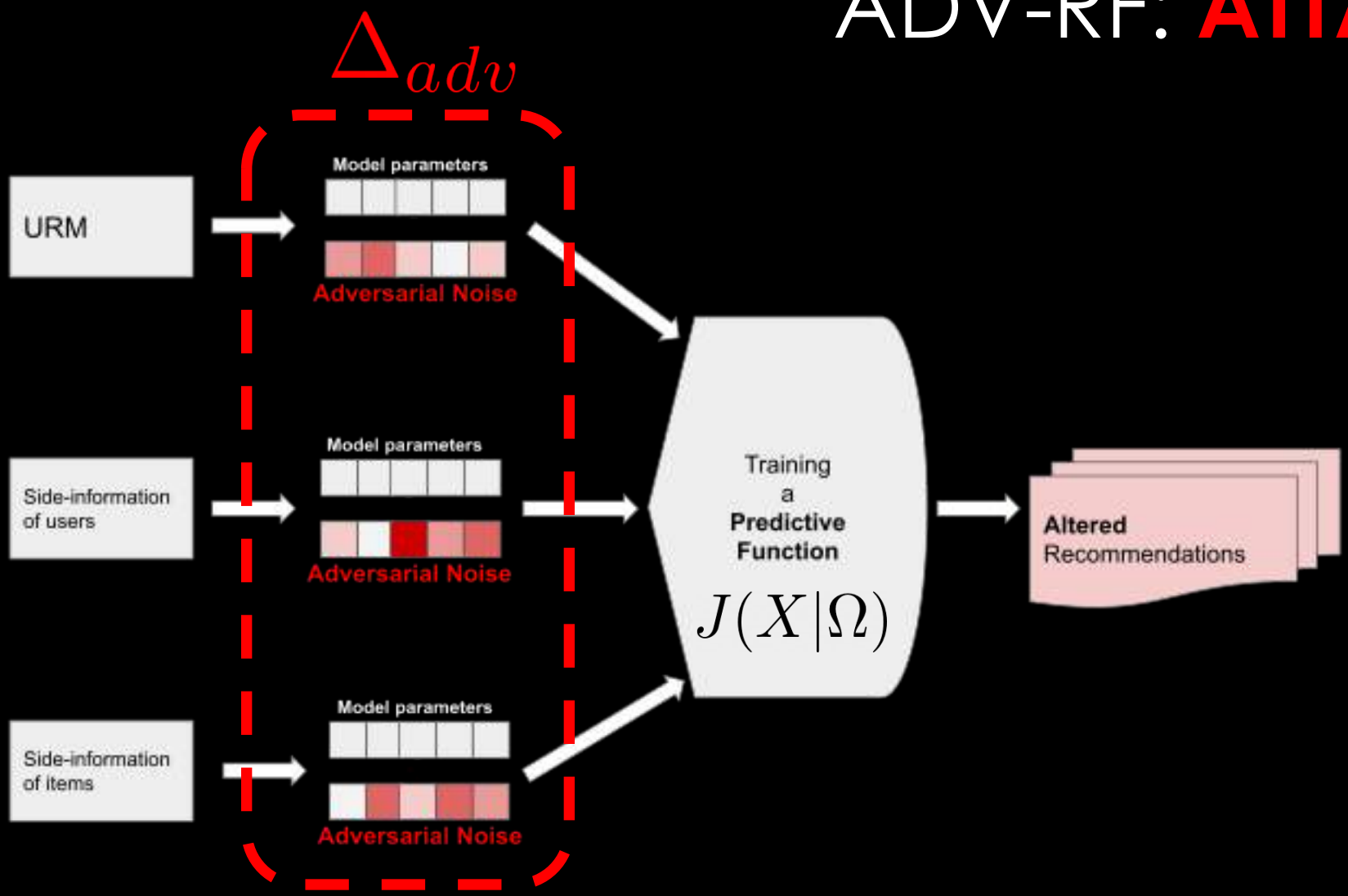
*[He, Xiangnan, et al. "**Adversarial personalized ranking for recommendation**." The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. ACM, 2018.]*

ADVERSARIAL RECOMMENDATION FRAMEWORK (ADV-RF)



Politecnico di Bari

ADV-RF: **ATTACKS**



ADV-RF: **ATTACKS**

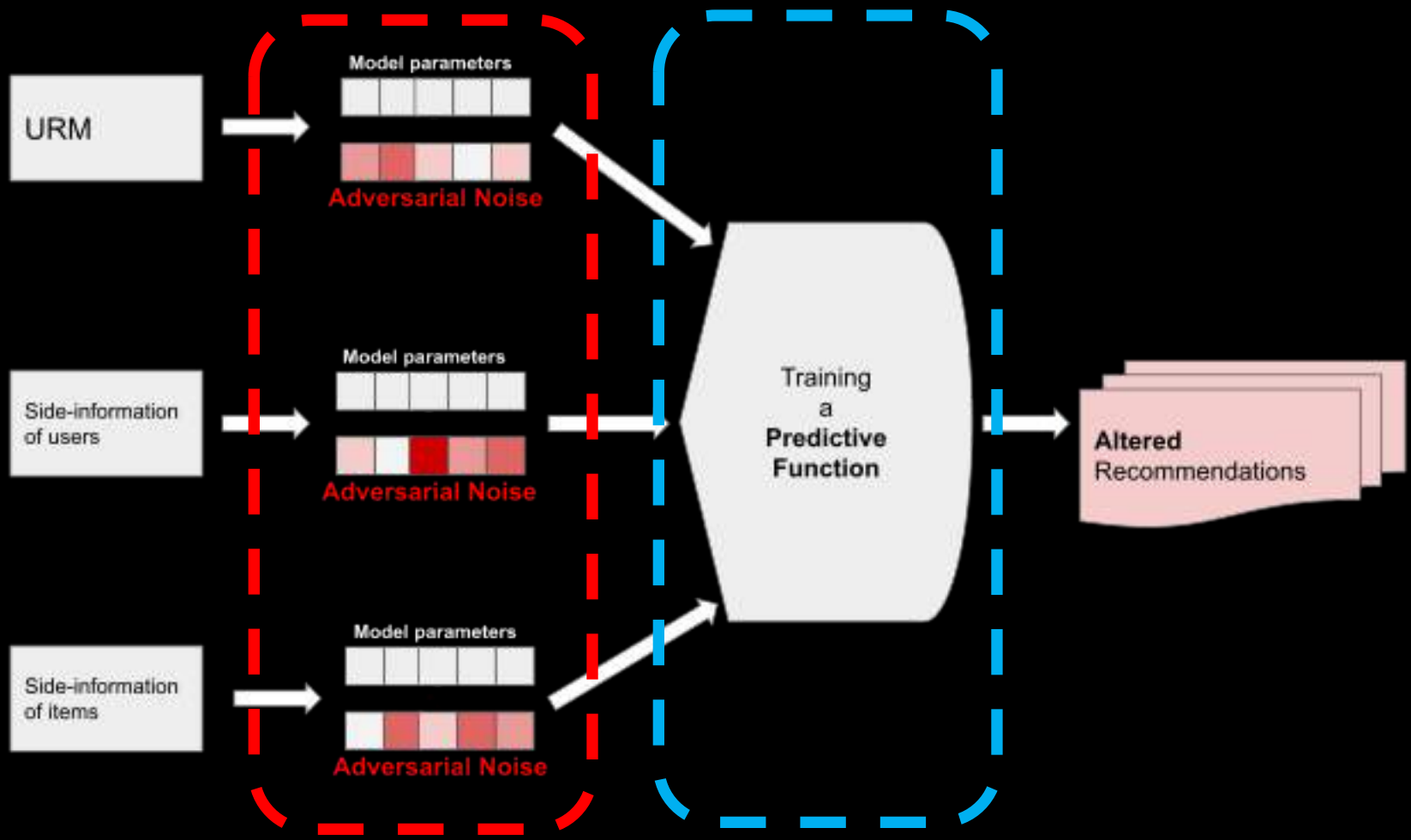
- **ATTACKER Goal? Maximize** the recommender model objective

$$\Delta_{adv} = \arg \max_{\Delta, \|\Delta\| \leq \epsilon} J(X|\Omega + \Delta)$$

- **Attack Method? FGSM**

$$\Delta_{adv} = \epsilon \frac{\Pi}{\|\Pi\|} \quad \text{where} \quad \Pi = \frac{\partial J(X|\Omega + \Delta)}{\partial \Delta}$$

ADV-RF: **DEFENSE**



ADV-RF: **DEFENSE**

- **Defender Goal? Minimize** the attack influence
- **Defence Method? Adversarial (re)training**

$$\arg\min_{\Omega} J(X|\Omega) + \lambda J\left(X|\Omega + \Delta_{adv}\right)$$

$$where \quad \Delta_{adv} = \arg\max_{\Delta, \|\Delta\| \leq \epsilon} J(X|\Omega + \Delta)$$

MINIMAX GAME

The training process is a **MINIMAX GAME**

$$\arg\min_{\Omega} \max_{\Delta, \|\Delta\| \leq \epsilon} J(X|\Omega) + \lambda J(X|\Omega + \Delta)$$

ADV-RF: **ALGORITHM**

Input: Training data $X, \epsilon, \lambda, \alpha$

Pretrain The Recommender Model

While *stopping-criteria* do:

Randomly select an x (or a mini-batch) from X

Construct adversarial perturbations

Update Model Parameters

$$\arg\min_{\Omega} \max_{\Delta, ||\Delta|| \leq \epsilon} \mathcal{L}(X|\Omega) + \lambda \mathcal{L}(X|\Omega + \Delta)$$

Politecnico di Bari

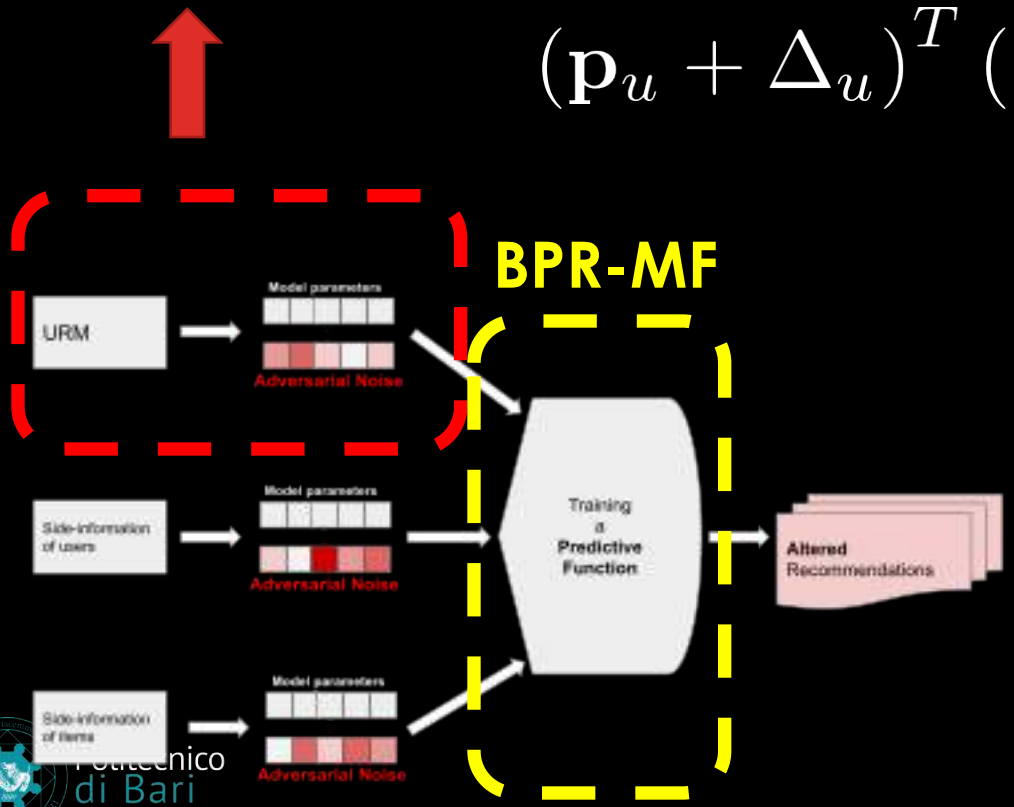
2.2 APPLICATIONS

ADVERSARIAL PERSONALIZED RANKING

[XIANGNAN HE ET AL., SIGIR '18]

Adversarial Perturbation on each **embedding** vector of user and item

$$(\mathbf{p}_u + \Delta_u)^T (\mathbf{q}_i + \Delta_i)$$

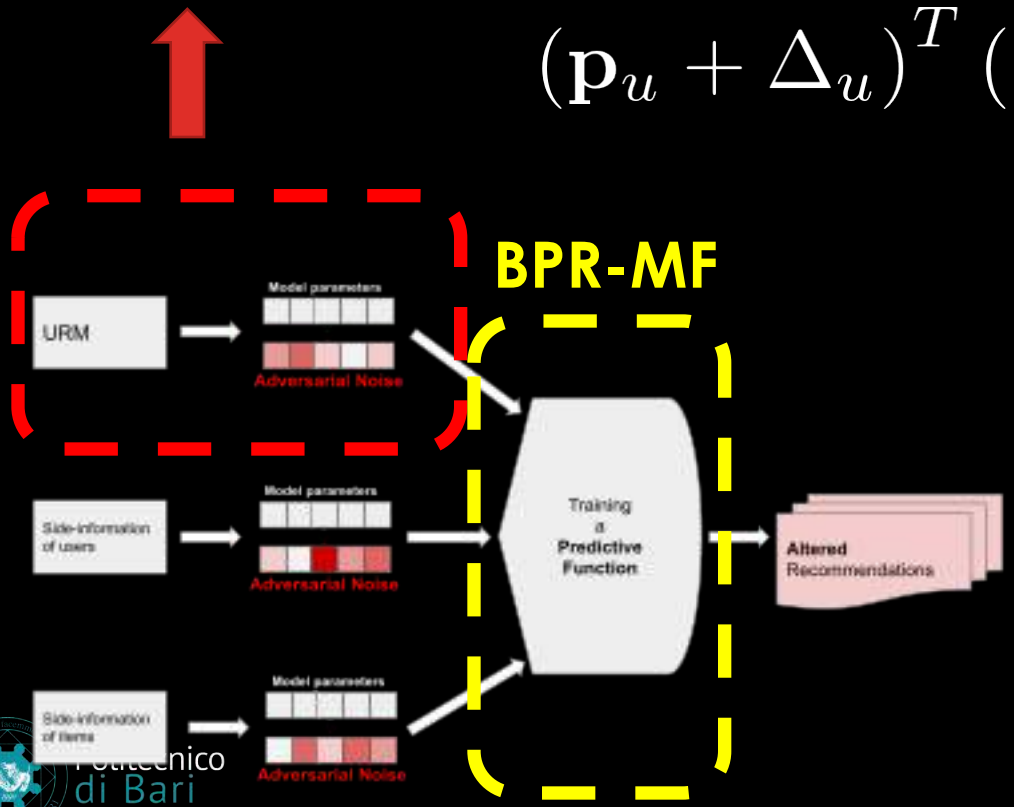


ADVERSARIAL PERSONALIZED RANKING

[XIANGNAN HE ET AL., SIGIR '18]

Adversarial Perturbation on each **embedding** vector of user and item

$$(\mathbf{p}_u + \Delta_u)^T (\mathbf{q}_i + \Delta_i)$$



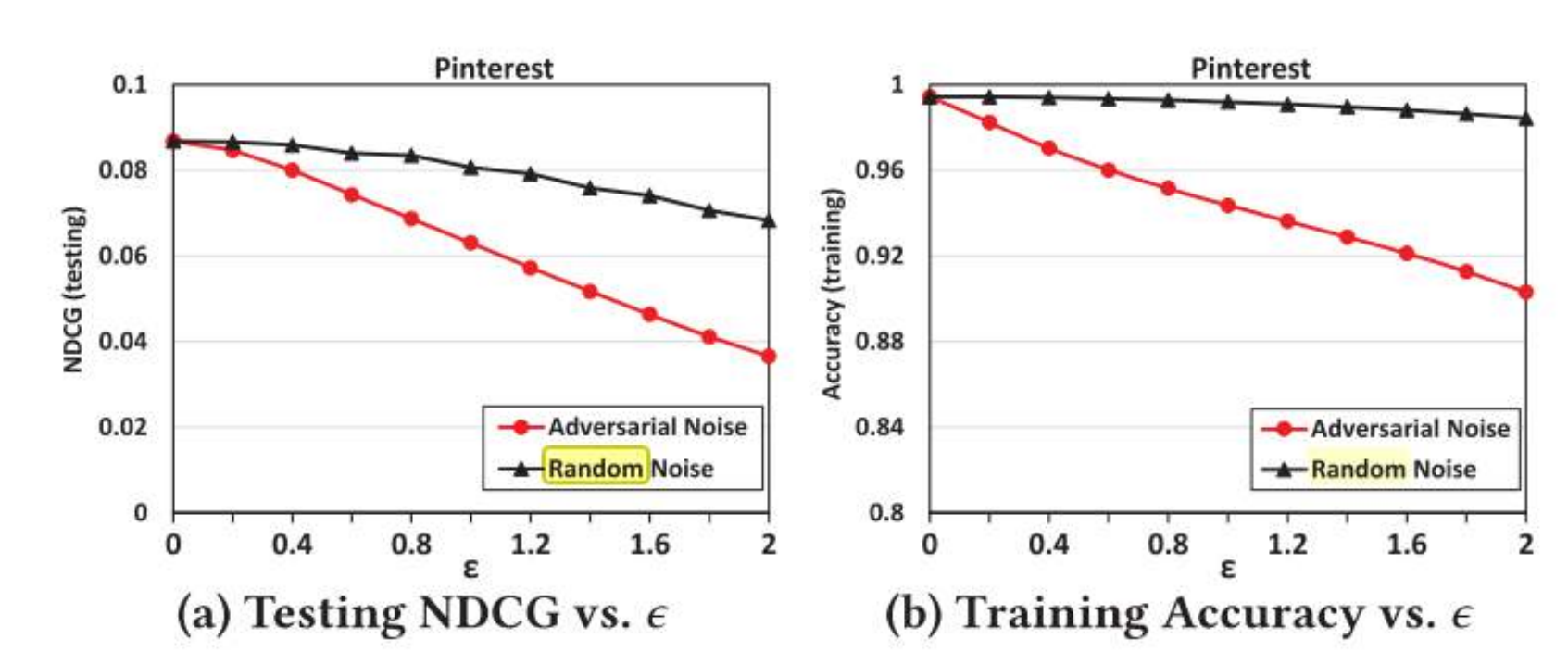
The impact of applying adversarial perturbation

reduction of NDCG@100			
	$\epsilon = 0.5$	$\epsilon = 1$	$\epsilon = 2$
Dataset	BPR-MF	BPR-MF	BPR-MF
Yelp	-22.1%	-42.7%	-63.8%
Pinterest	-9.5%	-25.1%	-55.7%
Gowalla	-26.3%	-53.0%	-78.0%

ADVERSARIAL PERSONALIZED RANKING

[XIANGNAN HE ET AL., SIGIR '18]

Are **Adversarial Perturbation** more **effective** than **random perturbation**?



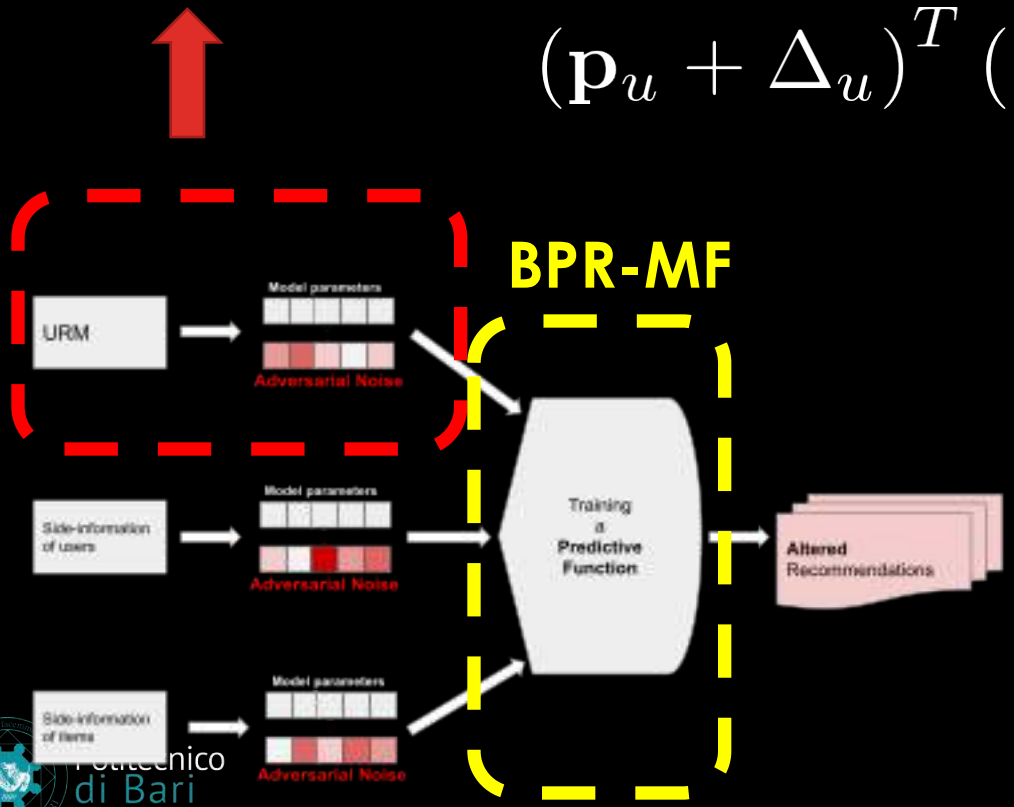
Politecnico di Bari

ADVERSARIAL PERSONALIZED RANKING

[XIANGNAN HE ET AL., SIGIR '18]

Adversarial Perturbation on each **embedding** vector of user and item

$$(\mathbf{p}_u + \Delta_u)^T (\mathbf{q}_i + \Delta_i)$$



Apply Adversarial Training

Adversarial Regularization

$$\arg\min_{\Omega} \max_{\Delta, \|\Delta\| \leq \epsilon} J(X|\Omega) + \lambda J(X|\Omega + \Delta)$$

where $J = J_{BPR}$

Politecnico di Bari

ADVERSARIAL PERSONALIZED RANKING

[XIANGNAN HE ET AL., SIGIR '18]

Do **Adversarial (re)training** improve the **robustness**?

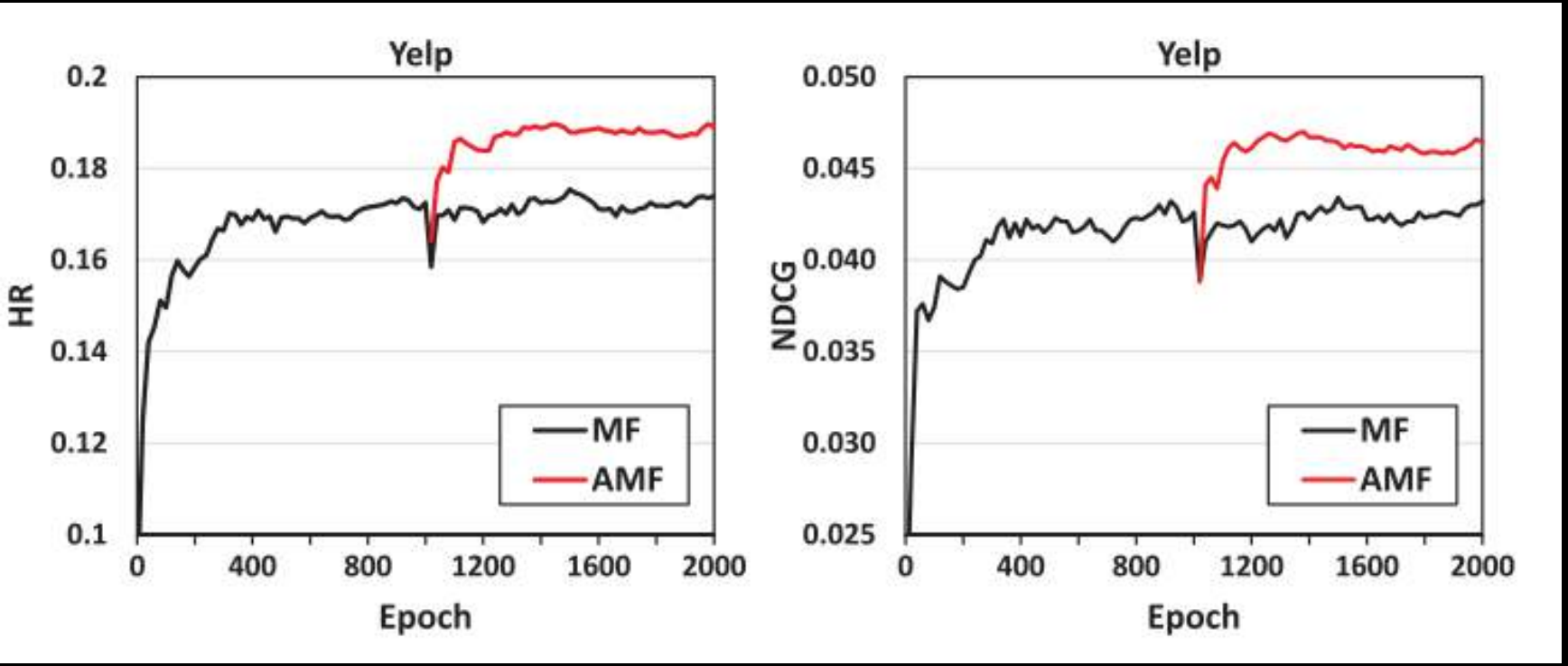
	NDCG@100					
	$\epsilon = 0.5$		$\epsilon = 1$		$\epsilon = 2$	
Dataset	BPR-MF	APR	BPR-MF	APR	BPR-MF	APR
Yelp	-22.1%	-4.7%	-42.7%	-12.5%	-63.8%	-31.0%
Pinterest	-9.5%	-2.6%	-25.1%	-7.2%	-55.7%	-23.4%
Gowalla	-26.3%	-2.9%	-53.0%	-13.2%	-78.0%	-29.2%

ADVERSARIAL PERSONALIZED RANKING

[XIANGNAN HE ET AL., SIGIR '18]

Can **Adversarial (re)training** improve the **recommendation performance**?

+10%



Politecnico di Bari

ADVERSARIAL PERSONALIZED RANKING

[XIANGNAN HE ET AL., SIGIR '18]

Can **Adversarial (re)training** improve the **recommendation performance**?

	Yelp, HR		Yelp, NDCG		Pinterest, HR		Pinterest, NDCG		Gowalla, HR		Gowalla, NDCG		RI
	K=50	K=100	K=50	K=100	K=50	K=100	K=50	K=100	K=50	K=100	K=50	K=100	
ItemPop	0.0405	0.0742	0.0114	0.0169	0.0294	0.0485	0.0085	0.0116	0.1183	0.1560	0.0367	0.0428	+416%
MF-BPR	0.1053	0.1721	0.0312	0.0420	0.2226	0.3403	0.0696	0.0886	0.4061	0.5072	0.1714	0.1878	+11.2%
CDAE [35]	0.1041	0.1733	0.0293	0.0405	0.2254	0.3495	0.0672	0.0873	0.4435	0.5483	0.1837	0.2007	+9.5%
IRGAN [31]	0.1119	0.1765	**0.0361***	**0.0465***	0.2254	0.3363	0.0724	0.0904	0.4157	0.518	0.1853	0.2019	+5.9%
NeuMF [17]	0.1135	0.1817	0.0335	0.0445	0.2342	0.3526	0.0734	0.0925	0.4558	0.5642	0.1962	0.2138	+2.9%
AMF	**0.1176***	**0.1885***	0.0350	**0.0465***	**0.2375***	**0.3595***	**0.0741***	**0.0938***	**0.4693***	**0.5763***	**0.2039***	**0.2212***	-

ADVERSARIAL PERSONALIZED RANKING

[XIANGNAN HE ET AL., SIGIR '18]

Can **Adversarial (re)training** improve the **recommendation performance**?

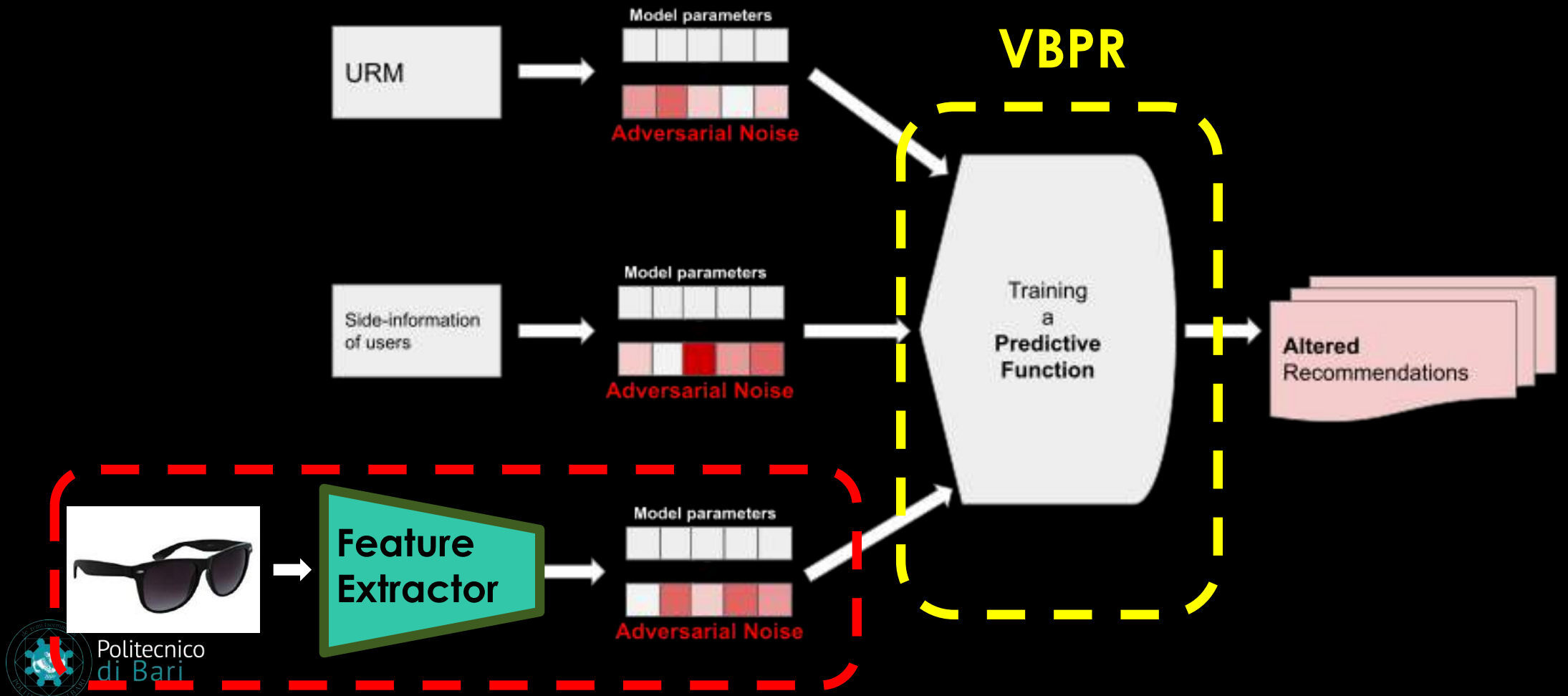
	RI
ItemPop	+416%
MF-BPR	+11.2%
CDAE [35]	+9.5%
IRGAN [31]	+5.9%
NeuMF [17]	+2.9%
AMF	-

Relative Improvement on HR@k and NDCG@k

Politecnico di Bari

ADVERSARIAL MULTIMEDIA RECOMMENDATION

[XIANGNAN HE ET AL., TKDE'19]



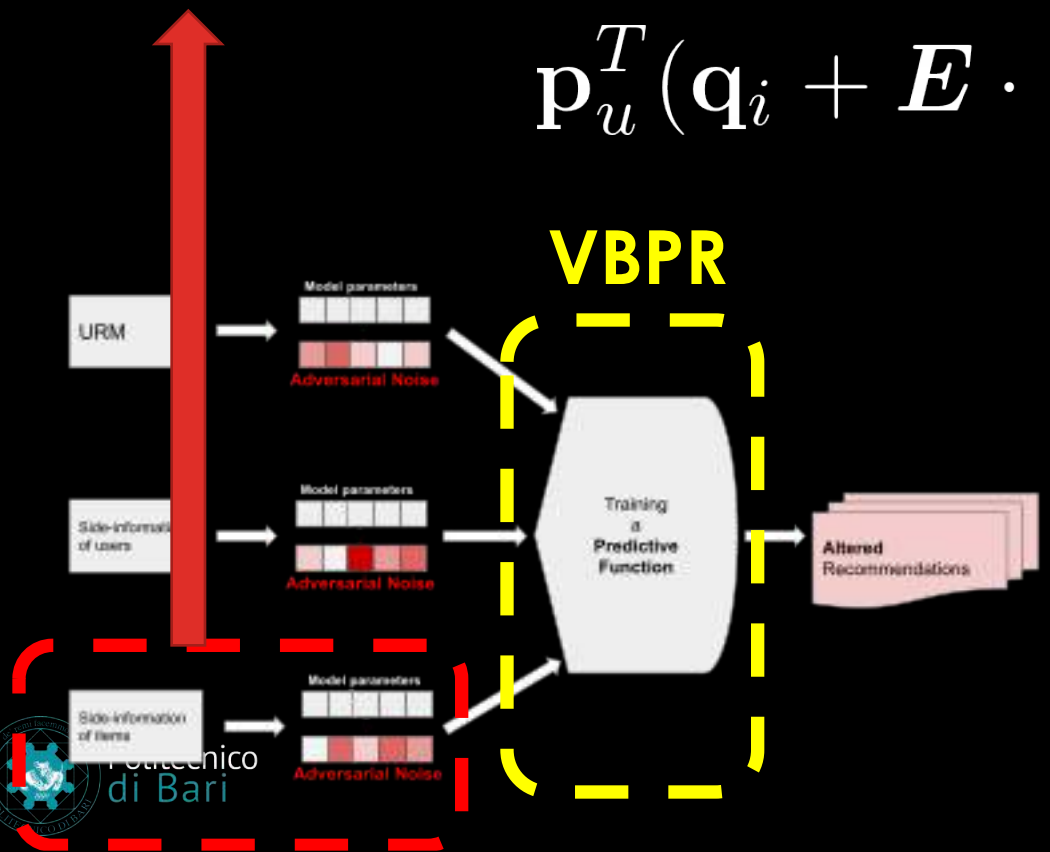
Politecnico di Bari

ADVERSARIAL MULTIMEDIA RECOMMENDATION

[XIANGNAN HE ET AL., TKDE'19]

Adversarial Perturbation on each **CONTENT** embedding.

$$\mathbf{p}_u^T(\mathbf{q}_i + \boldsymbol{E} \cdot (\boldsymbol{c}_i + \boldsymbol{\Delta}_i))$$



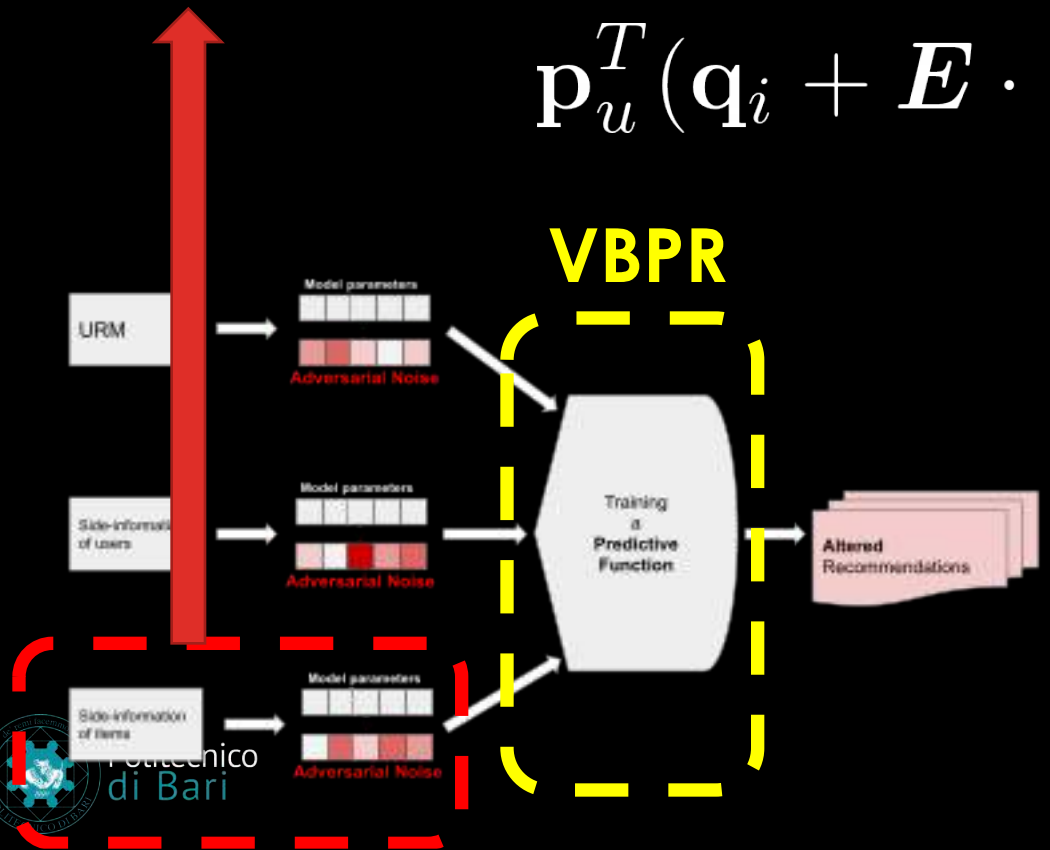
Politecnico di Bari

ADVERSARIAL MULTIMEDIA RECOMMENDATION

[XIANGNAN HE ET AL., TKDE'19]

Adversarial Perturbation on each **CONTENT** embedding.

$$\mathbf{p}_u^T(\mathbf{q}_i + \boldsymbol{E} \cdot (\boldsymbol{c}_i + \boldsymbol{\Delta}_i))$$



The impact of applying adversarial perturbation

reduction of NDCG@10			
	$\epsilon = 0.05$	$\epsilon = 0.1$	$\epsilon = 0.2$
Dataset	**VBPR**	**VBPR**	**VBPR**
Amazon	**-8.7%**	**-30.4%**	**-67.7%**
Pinterest	**-4.2%**	**-11.9%**	**-31.8%**

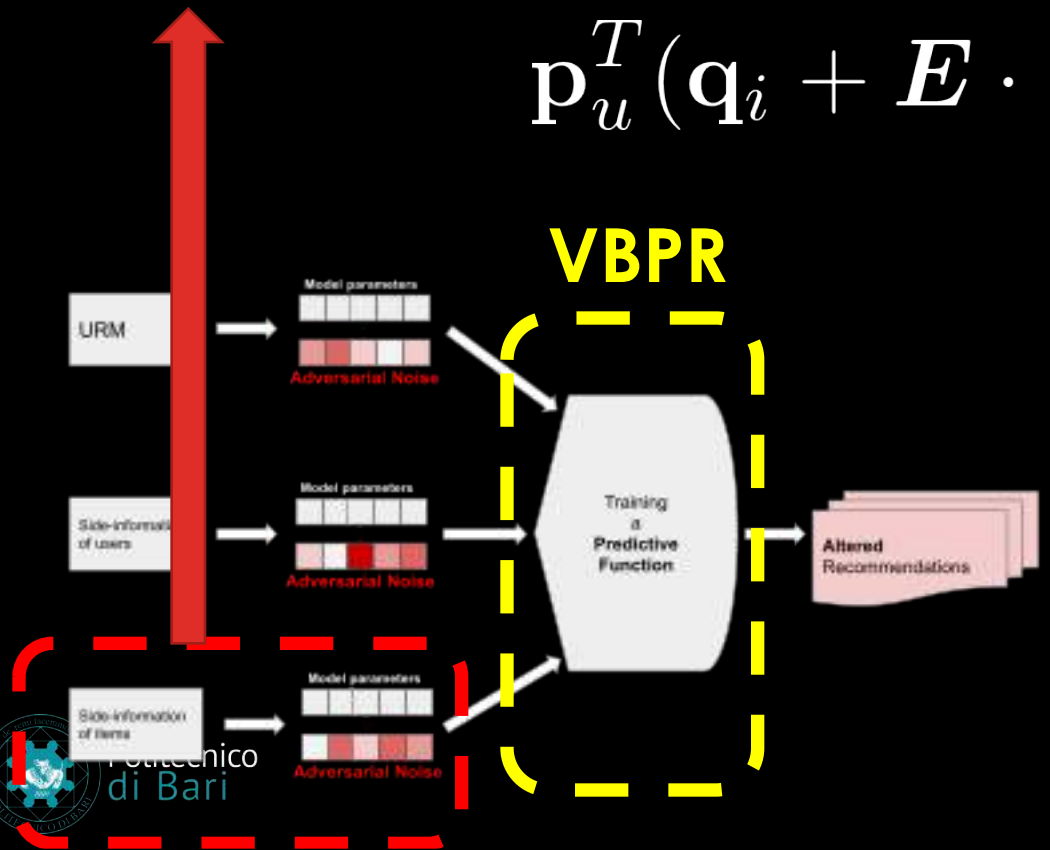
Politecnico di Bari

ADVERSARIAL MULTIMEDIA RECOMMENDATION

[XIANGNAN HE ET AL., TKDE'19]

Adversarial Perturbation on each **CONTENT** embedding.

$$\mathbf{p}_u^T(\mathbf{q}_i + \boldsymbol{E} \cdot (\boldsymbol{c}_i + \boldsymbol{\Delta}_i))$$



Apply Adversarial Training

Adversarial Regularization

$$\arg\min_{\Omega} \max_{\Delta, \|\Delta\| \leq \epsilon} J(X|\Omega) + \lambda J(X|\Omega + \Delta)$$

where $J = J_{VBPR}$

Politecnico di Bari

ADVERSARIAL MULTIMEDIA RECOMMENDATION

[XIANGNAN HE ET AL., TKDE'19]

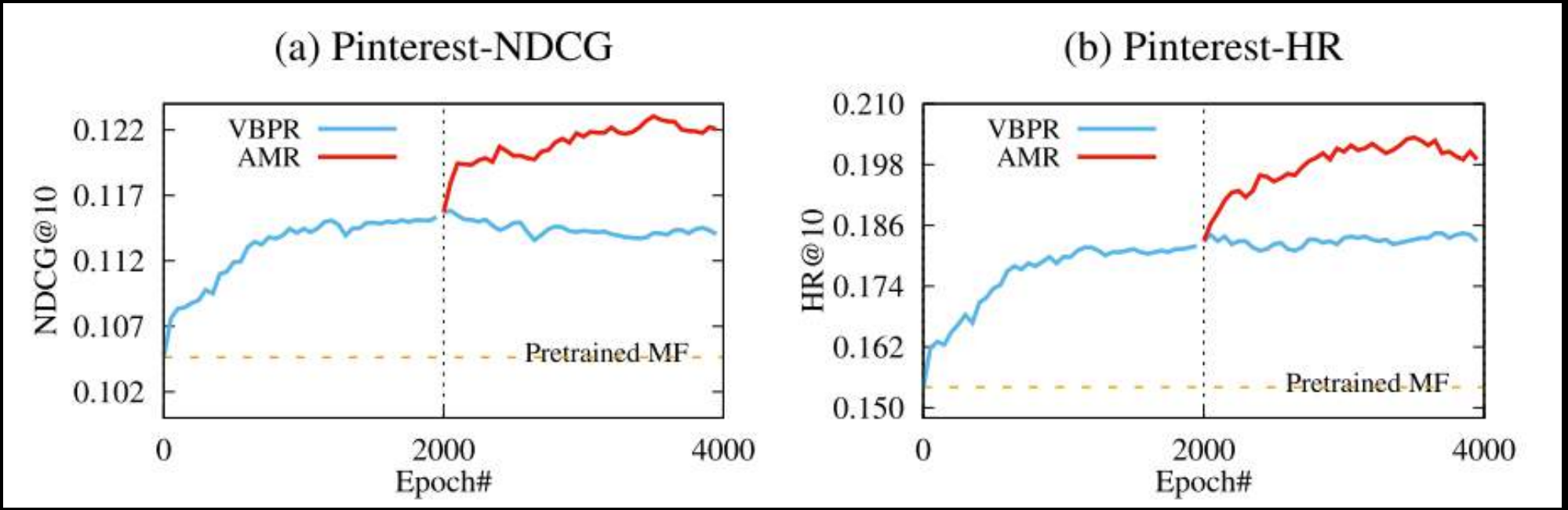
Do **Adversarial (re)training** improve the **robustness**?

	$\epsilon = 0.05$		$\epsilon = 0.1$		$\epsilon = 0.2$	
Dataset	**VBPR**	**AdvReg**	**VBPR**	**AdvReg**	**VBPR**	**AdvReg**
Amazon	-8.7%	-1.4%	-30.4%	-5.3%	-67.7%	-20.2%
Pinterest	-4.2%	-2.6%	-11.9%	-6.2%	-31.8%	-18.4%

ADVERSARIAL MULTIMEDIA RECOMMENDATION

[XIANGNAN HE ET AL., TKDE'19]

Can **Adversarial (re)training** improve the **recommendation performance**?



Politecnico di Bari

ADVERSARIAL MULTIMEDIA RECOMMENDATION

[XIANGNAN HE ET AL., TKDE'19]

Can **Adversarial (re)training** improve the **recommendation performance**?

	RI
Pop	34.27%
MF-eALS	7.98%
MF-BPR	7.91%
DUIF	52.61%
VBPR	5.14%
AMR	-

Relative Improvement on HR@k and NDCG@k

OTHER APPLICATIONS OF **ADV-RF**

[Yuan, F. et al., 2019]

- attack/defense on **Deep Neural Model/ Auto-Encoder**

[Chen and Li, 2019]

- attack/defense on Factorization Machine

[Li R. et al., 2019]

- adv. regularization to improve **Sequential recommendations**

[Park et al., 2019]

- perturb **user-continuous input**

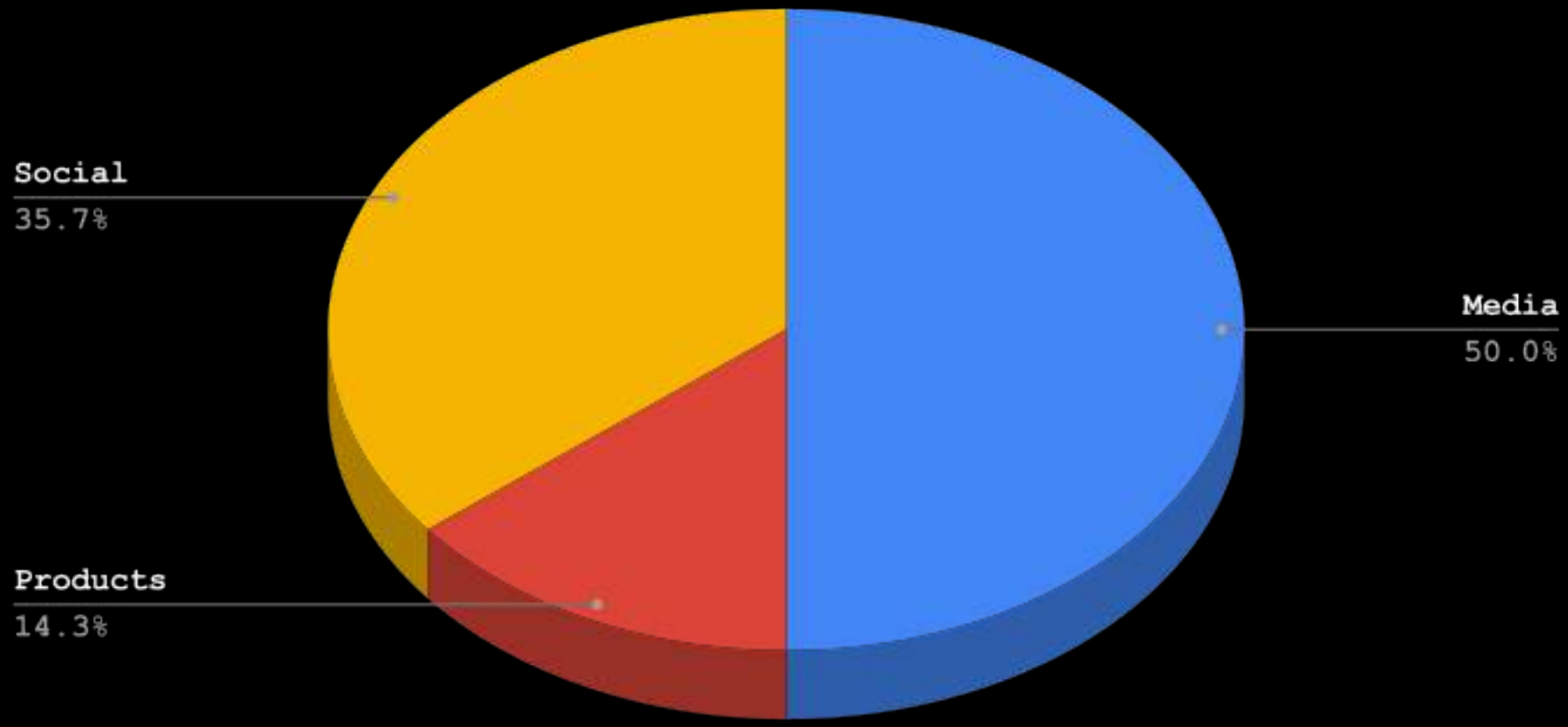
[Du et al., 2019]

- attack with **C&W**, defense with **Defensive Distillation**

For further study check our Survey!

2.3 DOMAIN AND OPEN DIRECTIONS

DOMAINS



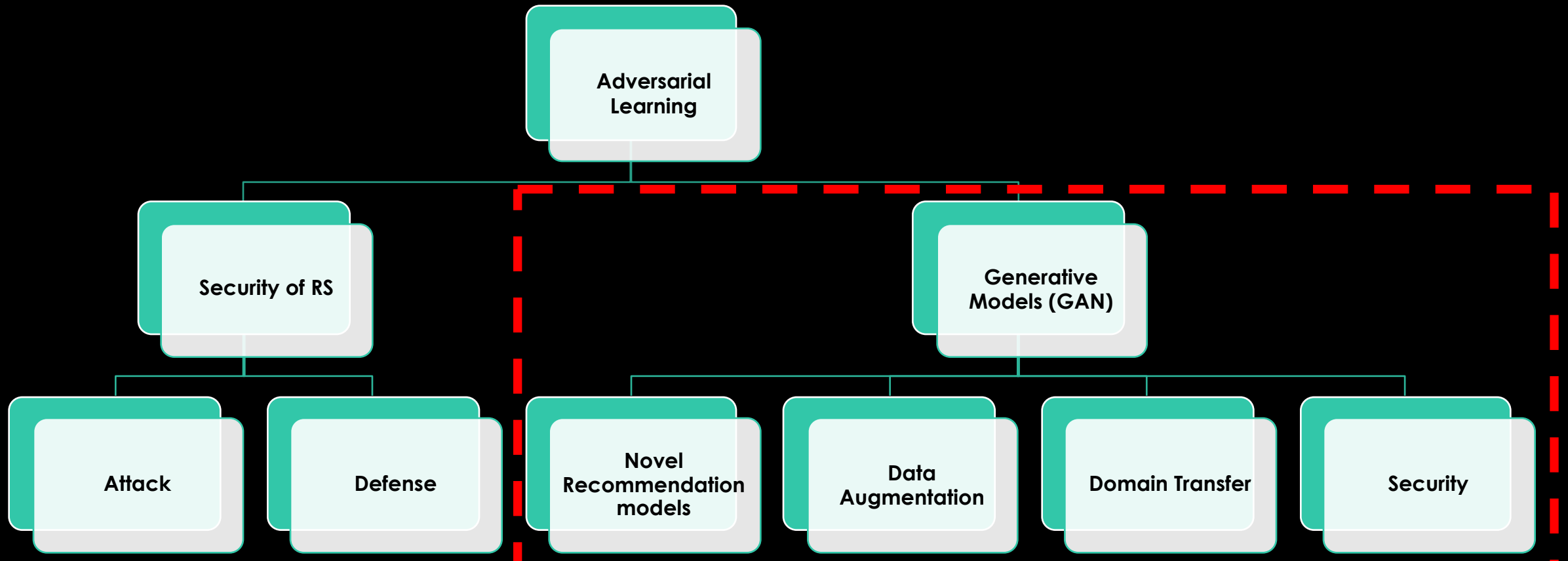
Politecnico
di Bari

OPEN DIRECTIONS

- Other **attacks strategies**
  - Use state-of-the-art adv. Attack strategies
  - Implement perturbation direct on the input:
    - user-rating profile
    - Imitation of implicit feedback
    - images, audio, videos
- Other **defence approaches**
- Verify and Extend the **AVD-RF** on other recomenders
- Othe **domains**

PART 3
ADVERSARIAL LEARNING FOR GAN-BASED RECOMMENDATION

ADVERSARIAL LEARNING FOR RS



3.1 GAN-BASED RECOMMENDATION FRAMEWORK (**GAN-RF**)

PIONEERING WORKS

- **IRGAN** [Wang J. et al., *SIGIR'17*]
  - Solution for CF based on **MF**
  - Combine **generative** and **discriminative** **IR** schools of thinkng under a GAN-framework
- **GraphGAN** [Wang H. et al., *AAAI'18*]
  - Solution for CF based on **graph-representation learning**
  - Combine **generative** and **discriminative** **graph-representation learning** models under a GAN-framework

IRGAN: A MINIMAX GAME FOR UNIFYING GENERATIVE AND DISCRIMINATIVE INFORMATION RETRIEVAL MODELS

[WANG J. ET AL., SIGIR'17]

Generative model

- **Assumption:** Exists a stochastic generative process between i and u

$$u \longrightarrow i$$

- **Model function:** $p_\theta(i|u,r)$ tries to generate **relevant items**, from the candidate pool for the given **user**

IRGAN: A MINIMAX GAME FOR UNIFYING GENERATIVE AND DISCRIMINATIVE INFORMATION RETRIEVAL MODELS

[WANG J. ET AL., SIGIR'17]

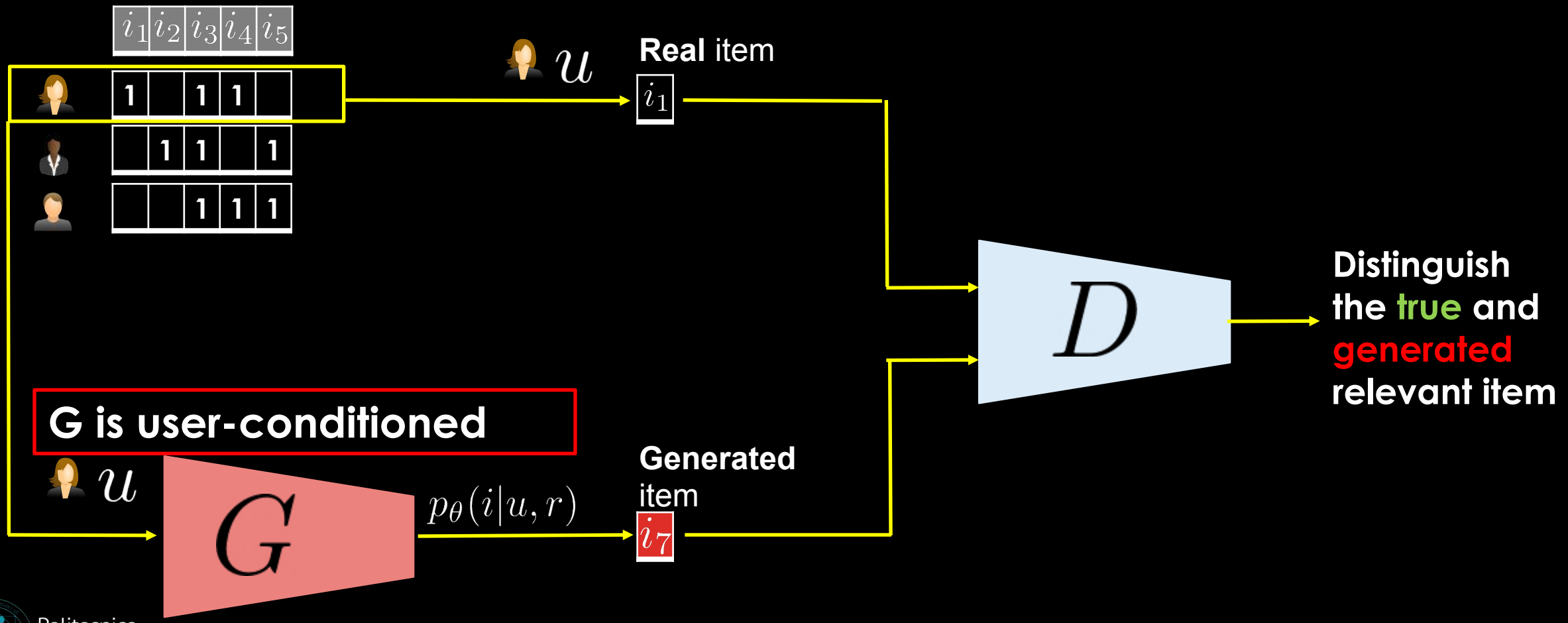
Discriminative model

- **Assumption:** The rating is predicted as a label given i and u

$$u + i \rightarrow r$$

- **Model function:** $p_\phi(u, i)$ tries to discriminate well-matched user-item pairs from ill-matched ones

GAN-RF



Politecnico di Bari

GAN-RF: MODEL FORMULATION

- We model GAN-RF with a **conditional GAN** (CGAN)
- G is user-conditioned (e.g., the user-id) to generate **user-personalized recommendations**

GAN-RF loss function:

the discriminative retrieval model

$$J(\theta, \phi) = \mathbb{E}_{i \sim p_{\text{true}}(i|u,r)}[\log D_\phi(i|u)] + \mathbb{E}_{\hat{i} \sim p_\theta(\hat{i}|u,r)}[\log(1 - D_\phi(\hat{i}|u))]$$

the generative retrieval model

Politecnico di Bari

GAN-RF: MODEL FORMULATION

Optimising **Discriminative Retrieval**

$$\arg\max_{\phi} J(\theta, \phi)$$

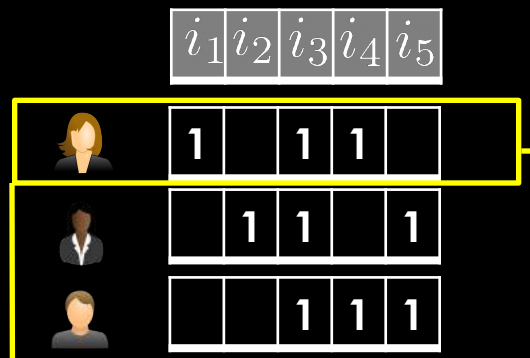
Optimising **Generative Retrieval**

$$\arg\min_{\theta} J(\theta, \phi)$$

The **MINIMAX GAME** to train the **generator** and the **discriminator** is formally defined as:

$$\arg\min_{\theta}\max_{\phi} J(\theta, \phi)$$

GAN-RF



u

Real item

i_1

$\arg\max_{\phi} J(\theta, \phi)$

D

Distinguish the true and generated relevant item

G is user-conditioned

u

G

$p_\theta(i|u, r)$

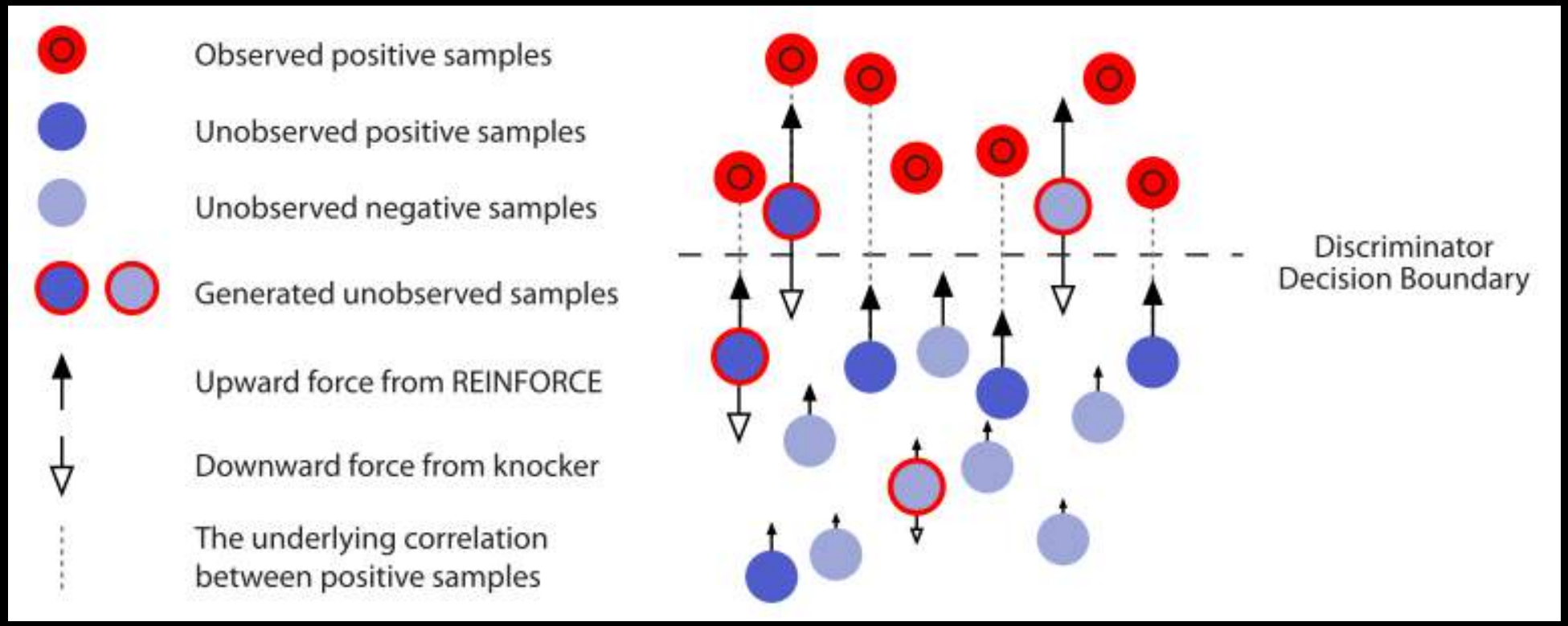
Generated item

i_7

$\arg\min_{\theta} J(\theta, \phi)$

Politecnico di Bari

GAN-RF



Politecnico di Bari

ADV-RF: **ALGORITHM**

Input: Training data **X**
Initialize G and D
Pretrain G and D
While ***stopping-criteria*** **do**:

MINIMAX

For g-steps **do**:
Generate K relevant items for each
Update G parameters
end for
For d-steps **do**:
Combine Real relevant items with Generated Relevant items.
Update D parameters
end for

WE NEED A DIFFERENTIABLE ITEM SAMPLING PROCEDURE

Politecnico di Bari

DIFFERENTIABLE SAMPLING STRATEGY

The generation of recommendation lists is a **discrete item sampling** operation.

The gradients that are derived from $J(\theta, \phi)$ **cannot** be directly used to optimize the **generator** via **gradient descent**

Two main approaches:

1. The **policy gradient based reinforcement learning algorithm** **(REINFORCE)**

   first proposed by [Wang J. et al., *SIGIR2017*]

2. The **Gumbel-Softmax differentiable sampling procedure**

   first proposed by [Yangdong Ye et al., *EXPERT SYST APPL 2019*]

REINFORCE

The **gradient** of the generator is derived as follows:

$$
\begin{aligned}
\nabla_\theta J &= \nabla_\theta \mathbb{E}_{i \sim p_{\text{ true }}(i|u,r)}[\log D_\phi(i|u)] + \mathbb{E}_{\hat{i} \sim p_\theta(\hat{i}|u,r)}[\log(1 - D_\phi(\hat{i}|u)))] \\
&= \nabla_\theta \mathbb{E}_{\hat{i} \sim p_\theta(\hat{i}|u,r)}[\log(1 - D_\phi(\hat{i}|u)))] \\
&= \sum_{m=1}^{M} \nabla_\theta p_\theta(\hat{i}_m|u,r) \log(1 - D_\phi(\hat{i}_m|u)) \\
&= \sum_{m=1}^{M} p_\theta(\hat{i}_m|u,r) \nabla_\theta \log p_\theta(\hat{i}_m|u,r) \log(1 - D_\phi(\hat{i}_m|u))) \\
&= \mathbb{E}_{\hat{i} \sim p_\theta(\hat{i}|u,r)}[\nabla_\theta \log p_\theta(\hat{i}|u,r) \log(1 - D_\phi(\hat{i}|u)))] \\
&\simeq \frac{1}{K} \sum_{k=1}^{K} \nabla_\theta \log p_\theta(\hat{i}_k|u,r) \log(1 - D_\phi(\hat{i}_k|u)))
\end{aligned}
$$

reward for the generative function

$\hat{i}_k$ is the ***k*-th relevant item** sampled from the current version of the generative function

GUMBEL-SOFTMAX

REINFORCE suffers of **instability** in gradient estimation with large set of items

Gumbel-Softmax builds a **differentiable bridge** between **G** and **D**

We obtain an **approximate one-hot representation** of the sampled item:

$$v_k = \frac{\exp\left(\left(\log p_\theta\left(i_k|u,r\right) + g_k\right)/\tau\right)}{\sum_{j=1}^{K}\exp\left(\left(\log p_\theta\left(i_j|u,r\right) + g_j\right)/\tau\right)} \qquad k = 1,\ldots,K$$

τ Temperature parameter

where g_j are i.i.d **sampled noise** from

$$Gumbel(0,1) = -\log(-\log(Uniform(0,1)))$$

Politecnico di Bari

3.2 APPLICATIONS

CATEGORIZATION

1. **Collaborative** Recommendation
    1. Pure Collaborative Filtering
    2. Graph-based Recommendation
    3. Hybrid
2. **Contextual** Recommendation
    1. Temporal-aware
    2. Geographical
3. **Cross-domain** Recommendation
4. **Complementary** Recommendation
5. Other applications

COLLABORATIVE RECOMMENDATION - **PURE**:

CFGAN

[DONG-KYU CHAE, CIKM '18]

PROBLEM

1. Generator's **performance degredation** as training progresses
2. Difficulties in capturing **users's preference** on **0/1-sparse-vector**

PROPOSAL

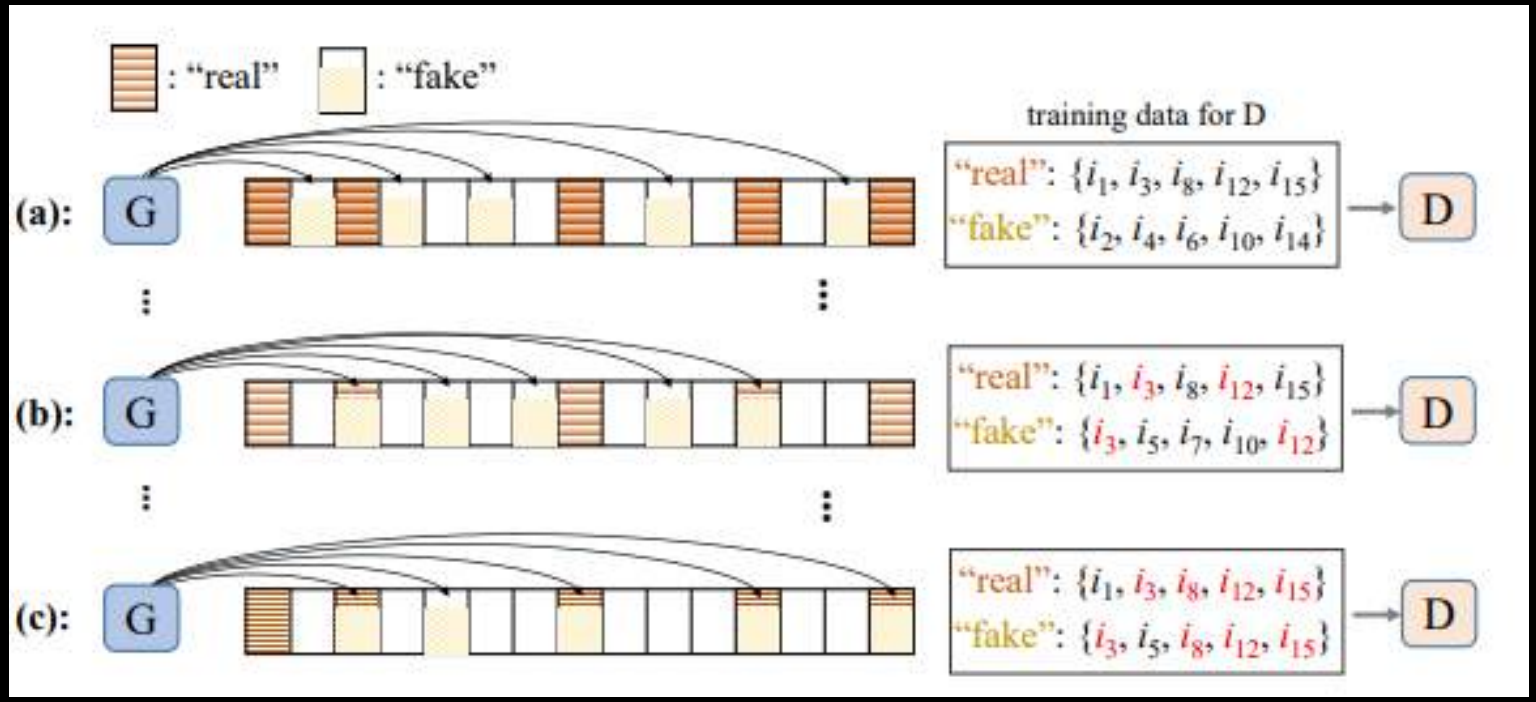
1. **Vector-wise adversarial training** where G generates ***real-valued vectors***
2. Novel **negative-sampling methods to capture the** user's relative preferences

COLLABORATIVE RECOMMENDATION - **PURE**:

CFGAN

[DONG-KYU CHAE, CIKM '18]

Generator's performance degredation because it samples 'real' items with a **contradicting labels**

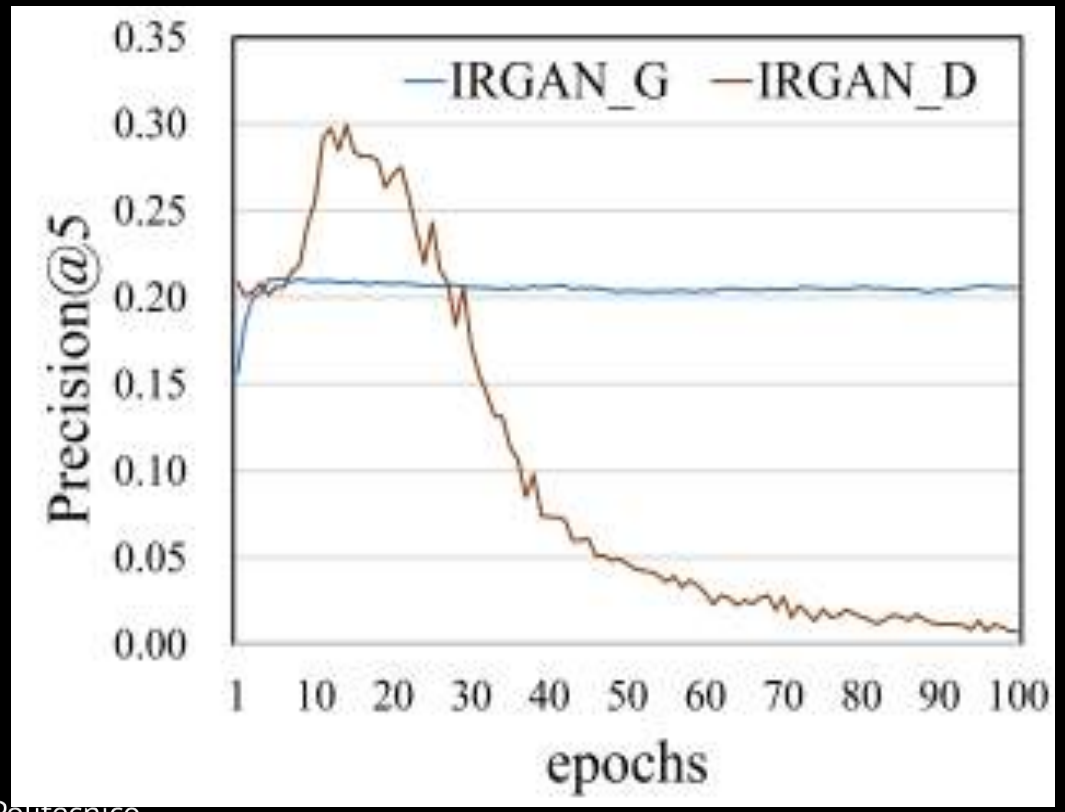


COLLABORATIVE RECOMMENDATION - **PURE**:

CFGAN

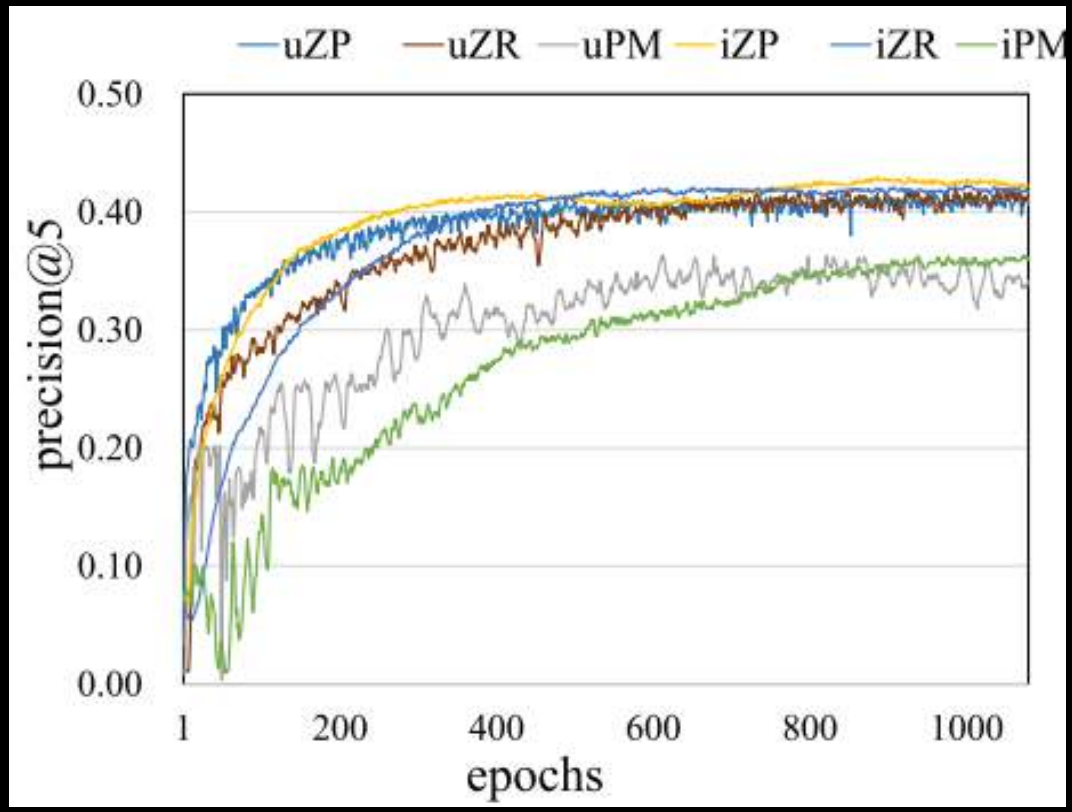
[DONG-KYU CHAE, CIKM '18]

Single-item IRGAN implementation



Results on Movielens 100K

Vector-wise CF-GAN implementation



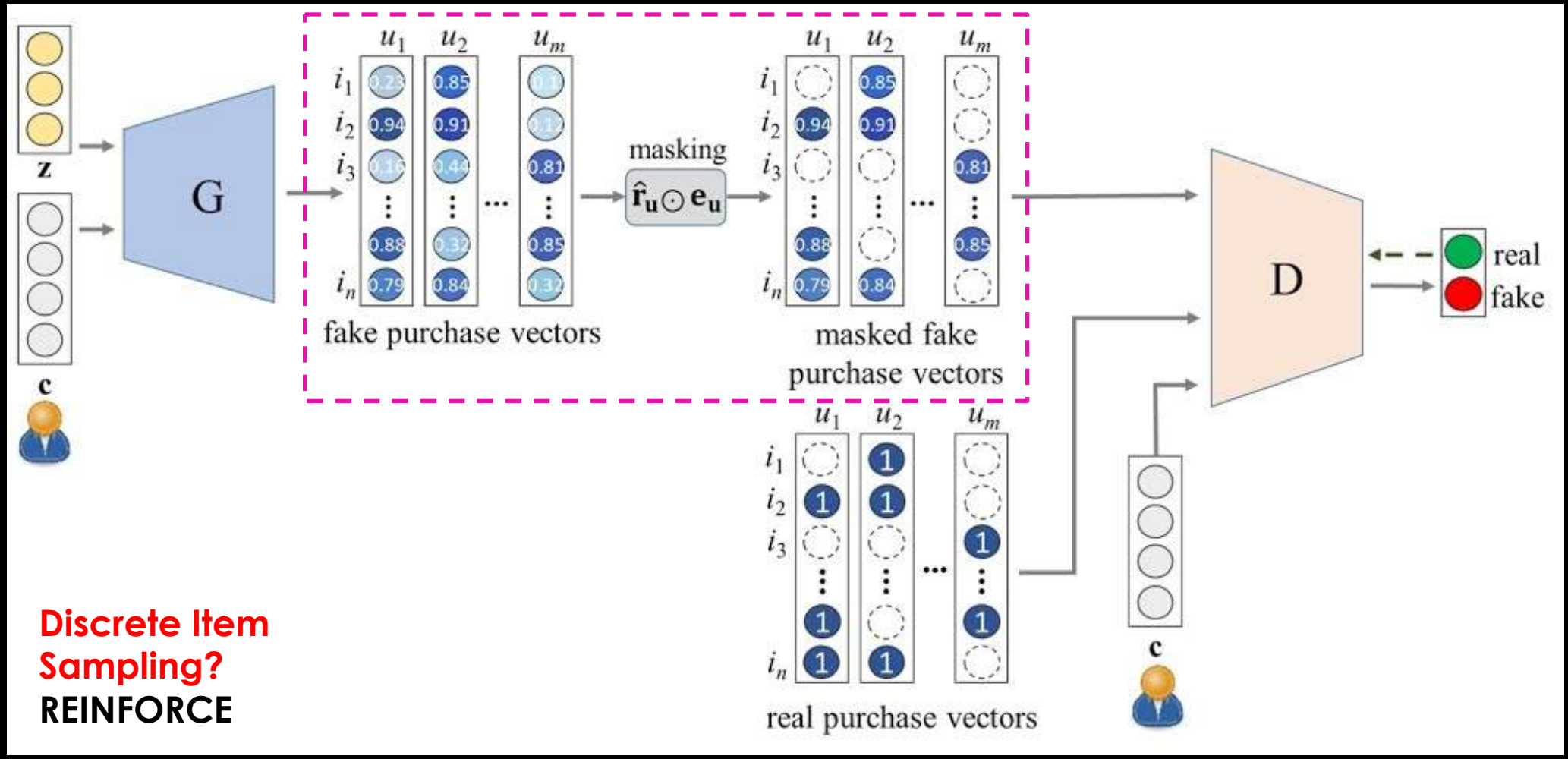
Results on Movielens 100K

Politecnico di Bari

COLLABORATIVE RECOMMENDATION - **PURE**:

CFGAN

[DONG-KYU CHAE, CIKM '18]



Discrete Item Sampling?
REINFORCE

Politecnico di Bari

Code

COLLABORATIVE RECOMMENDATION - **PURE:**

CFGAN

[DONG-KYU CHAE, CIKM '18]

datasets	Movielens 100K								Movielens 1M							
metrics	P@5	P@20	R@5	R@20	G@5	G@20	M@5	M@20	P@5	P@20	R@5	R@20	G@5	G@20	M@5	M@20
ItemPop	.181	.138	.102	.251	.163	.195	.254	.292	.157	.121	.076	.197	.154	.181	.252	.297
BPR	.348	.236	.116	.287	.370	.380	.556	.574	.341	.252	.077	.208	.349	.362	.537	.556
FISM	.426	.285	.140	.353	.462	.429	.674	.685	.420	.302	.107	.270	.443	.399	.637	.651
CDAE	.433	.287	.144	.353	.465	.425	.664	.674	.419	.307	.108	.272	.439	.401	.629	.644
GraphGAN	.212	.151	.102	.260	.183	.249	.282	.312	.178	.194	.070	.179	.205	.184	.281	.316
IRGAN	.312	.221	.107	.275	.342	.368	.536	.523	.263	.214	.072	.166	.264	.246	.301	.338
Ours	**.444**	**.294**	**.152**	**.360**	**.476**	**.433**	**.681**	**.693**	**.432**	**.309**	**.108**	**.272**	**.455**	**.406**	**.647**	**.660**

G@N = normalized Discounted Cumulative Gain

M@N = Mean Reciprocal Rank

+ 2.8% against FISM [Kabbur et al., KDD'13]

Politecnico di Bari

COLLABORATIVE RECOMMENDATION - GRAPH:

GRAPHGAN

[WANG H. ET AL., AAAI'18]

PROBLEM

1. Unify Generative-Discirminative models in graph representation learning
2. Computationally inefficiency for **G** (softmax) gradient update

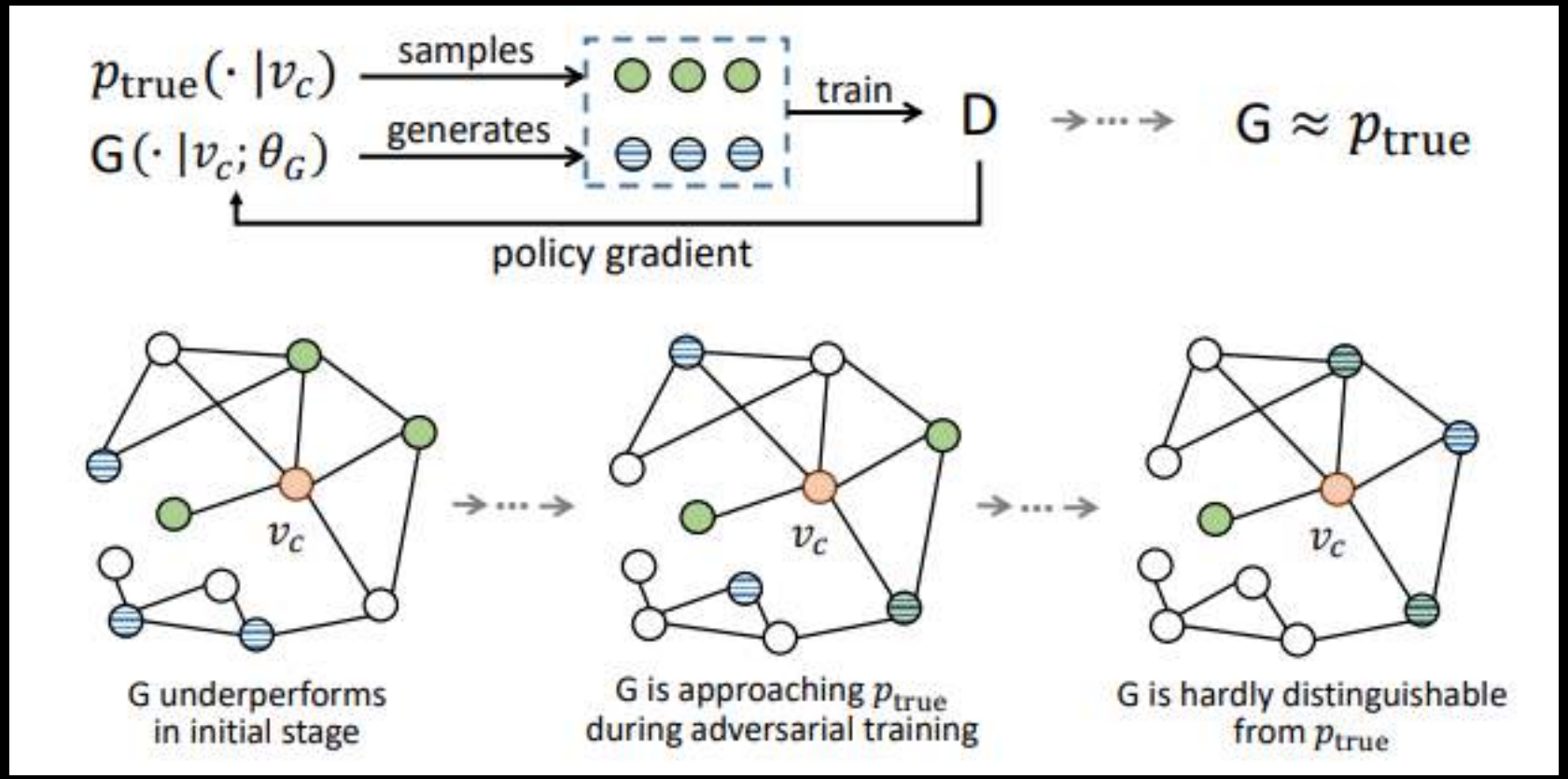
PROPOSAL

1. GAN-based approach (can be used as recommender model is data are represented as **bipartite-graph** and the recommendation problem as a **link prediction task**)
2. **Graph-softmax** to compute connectivity distribution in an **efficient** and g**raph-structural-aware**

COLLABORATIVE RECOMMENDATION - GRAPH:

GRAPHGAN

[WANG H. ET AL., AAAI'18]



Discrete Item Sampling?
REINFORCE

Code

Politecnico di Bari

COLLABORATIVE RECOMMENDATION - HYBRID:

AUGCF

[WANG Q. ET AL., KDD'19]

PROBLEM

1. Data sparsity problem (**cold-users**)
2. Current **hybrid CF** models:
   - apply the expensive process of exploiting the side information to all users (**unnecessary computational resources**)
   - are **not general** (complex to adapt at varying of dataseta/domaina)

PROPOSAL

1. GAN-based framework to train a **generator** to be used as a **data augmenter to generate "real" interaction data for cold-users exploting side information**
2. **End-to-end** training procedure [data augmenter + CF model]

COLLABORATIVE RECOMMENDATION - **HYBRID**:

AUGCF

[WANG Q. ET AL., KDD'19]

Two-phase Procedure for the End-to-End Model

PHASE 1: BUILD the Data Augmenter

Minimax game between **G** and **D.**

The **Generator** aims to create a fake-interacted item by exploting the **side information**

PHASE 2: CF Recommendation

G is fixed and acts as a **data augmentator** with data **indistinguishable** to **D**

D learns to generate users' **relevant items based pm real or sampled items**

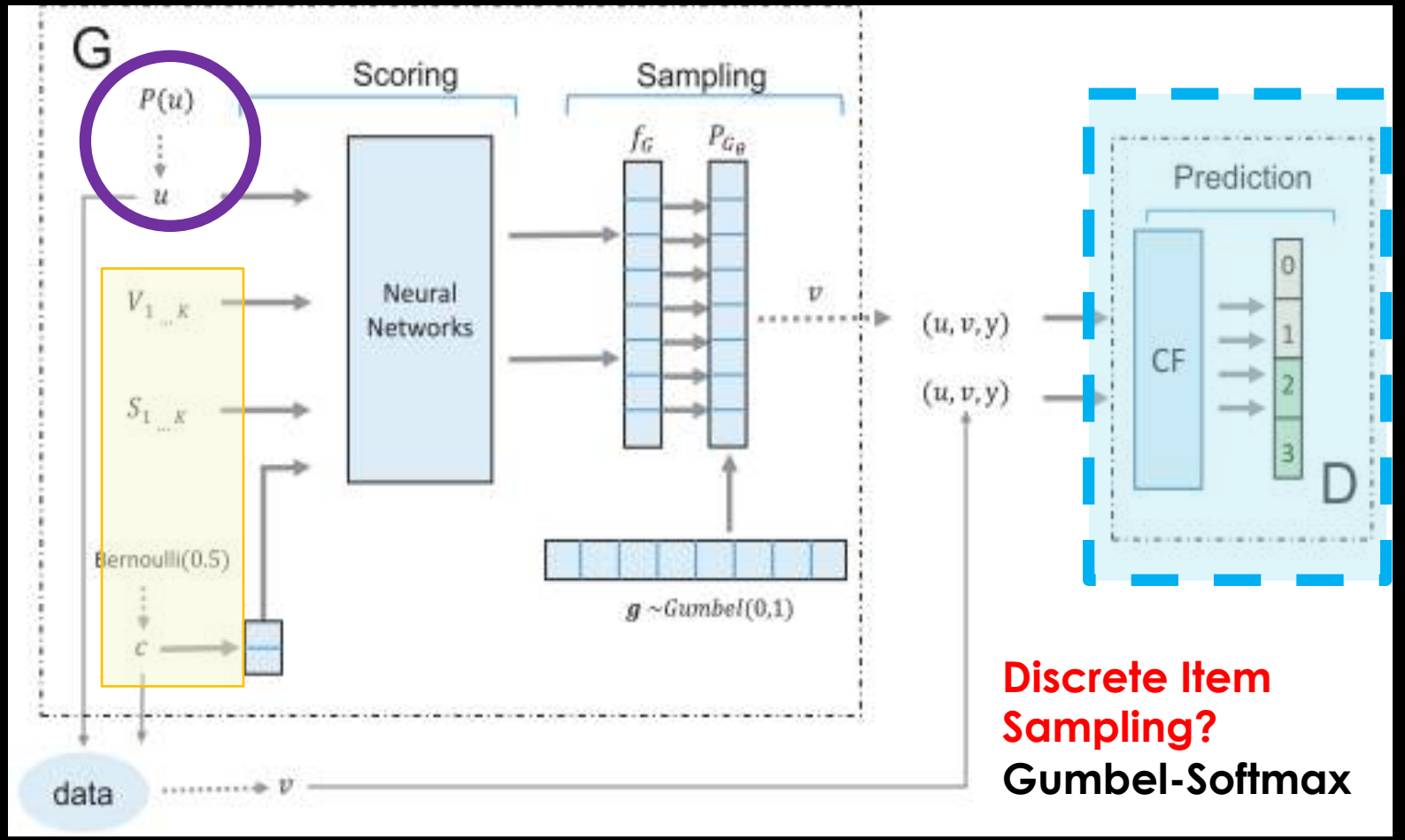
Acts as the RECOMMENDER

COLLABORATIVE RECOMMENDATION - HYBRID:

AUGCF

[WANG Q. ET AL., KDD'19]

less active users have higher probabilities to be sampled



Discrete Item Sampling?

Gumbel-Softmax

Politecnico di Bari

COLLABORATIVE RECOMMENDATION - HYBRID:
AUGCF
[WANG Q. ET AL., KDD'19]

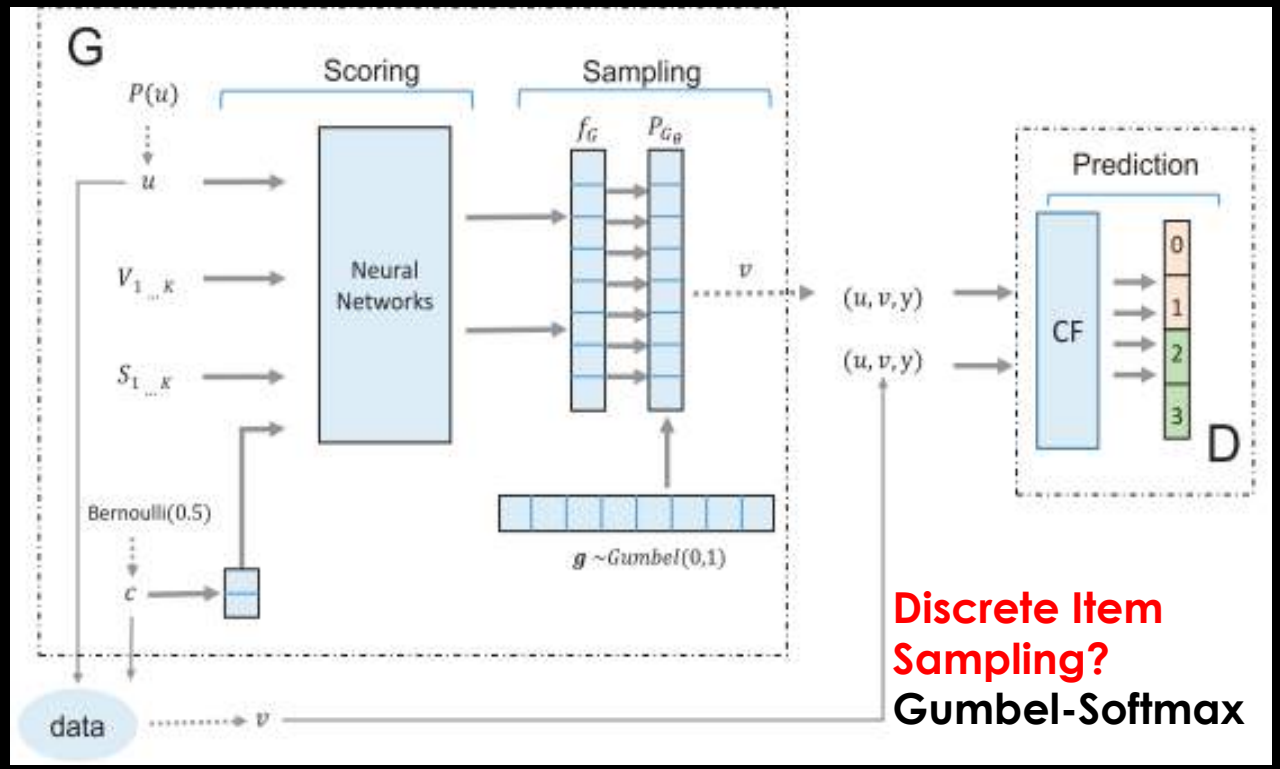
The authors tested the generalizability with 2 hybrid-models:

1. Content-Based AugCF

- Items reviews to model users' behaviors and items properties
- G is implemented with a DeepCoNN (Content-RS)

2. Sparse Feature-Based AugCF

- Sparse features like tags, timestamps
- G is implemented with **Wide & Deep learning framework**



Politecnico di Bari

CONTEXTUAL RECOMMENDATION - **TEMPORAL**:
PLASTIC
[MIN YANG ET AL., IJCAI'19]

PROBLEM

Leverage Long And Short- Term Information in top-*k* recommendation scenario

PROPOSAL

Use adversarial learning to train a generator G to generate **highly rewarded** recommendation list

CONTEXTUAL RECOMMENDATION - **TEMPORAL**:

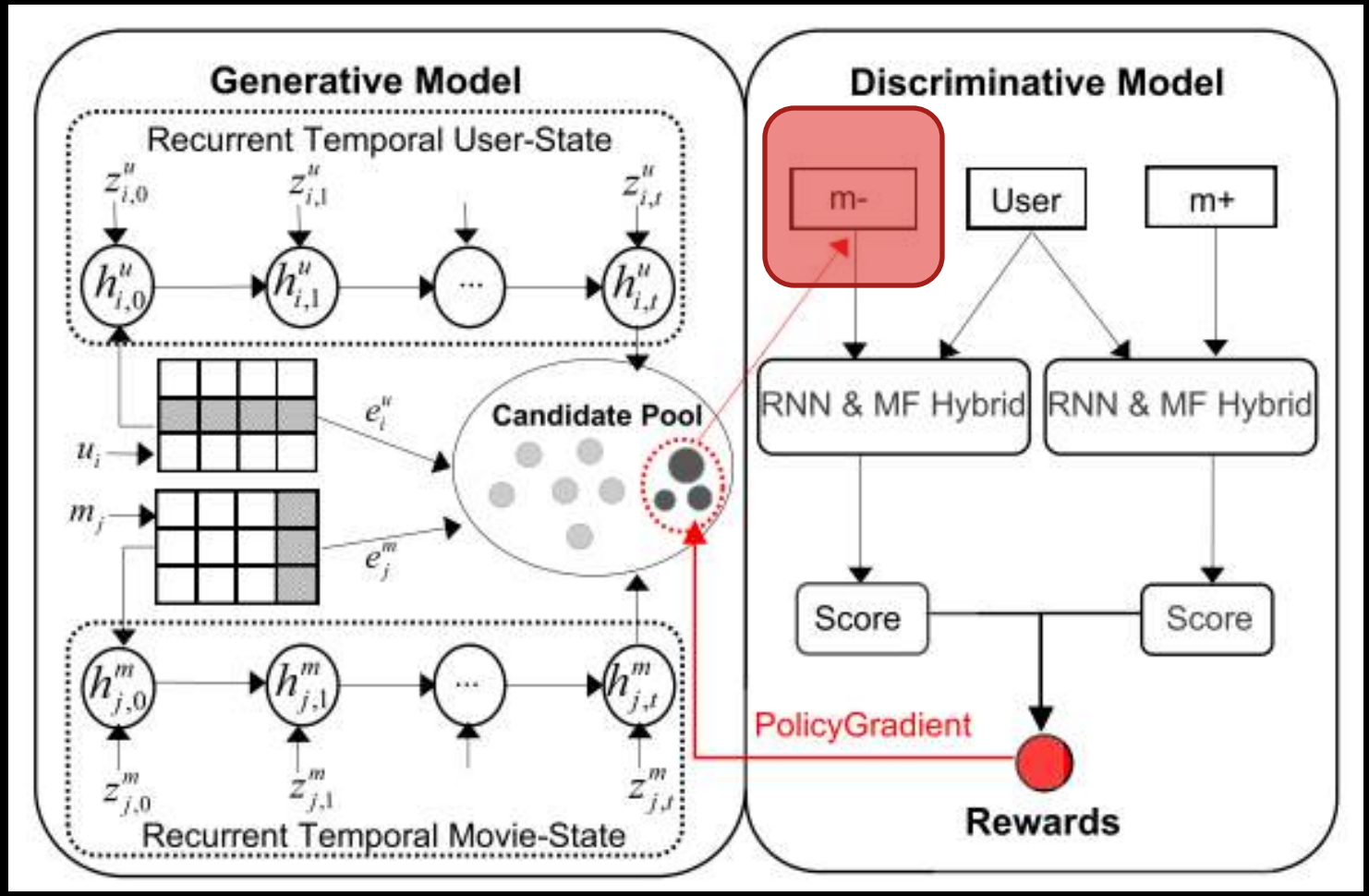
PLASTIC

[MIN YANG ET AL., IJCAI'19]

G takes a user at a specific time as input, **predicts** the recommendation list based on the **historical user-item interactions**

D is **SIAMESE** with two **point-wise** models (**MF + RNN**) that share parameters and are updated by minimizing a **pair-wise** loss. (hinge loss)

Discrete Item Sampling?
REINFORCE



Politecnico di BariCode

CONTEXTUAL RECOMMENDATION - **GEOGRAPHICAL**:

GEO-ALM

[ZHI-JIE WANG ET AL., IJCAI'19]

PROBLEM

1. **Negative samples** for **pair-wise** learning are randomly chosen and are **not informative**
2. How geographical information can be included in **POI recommendation task**

PROPOSAL

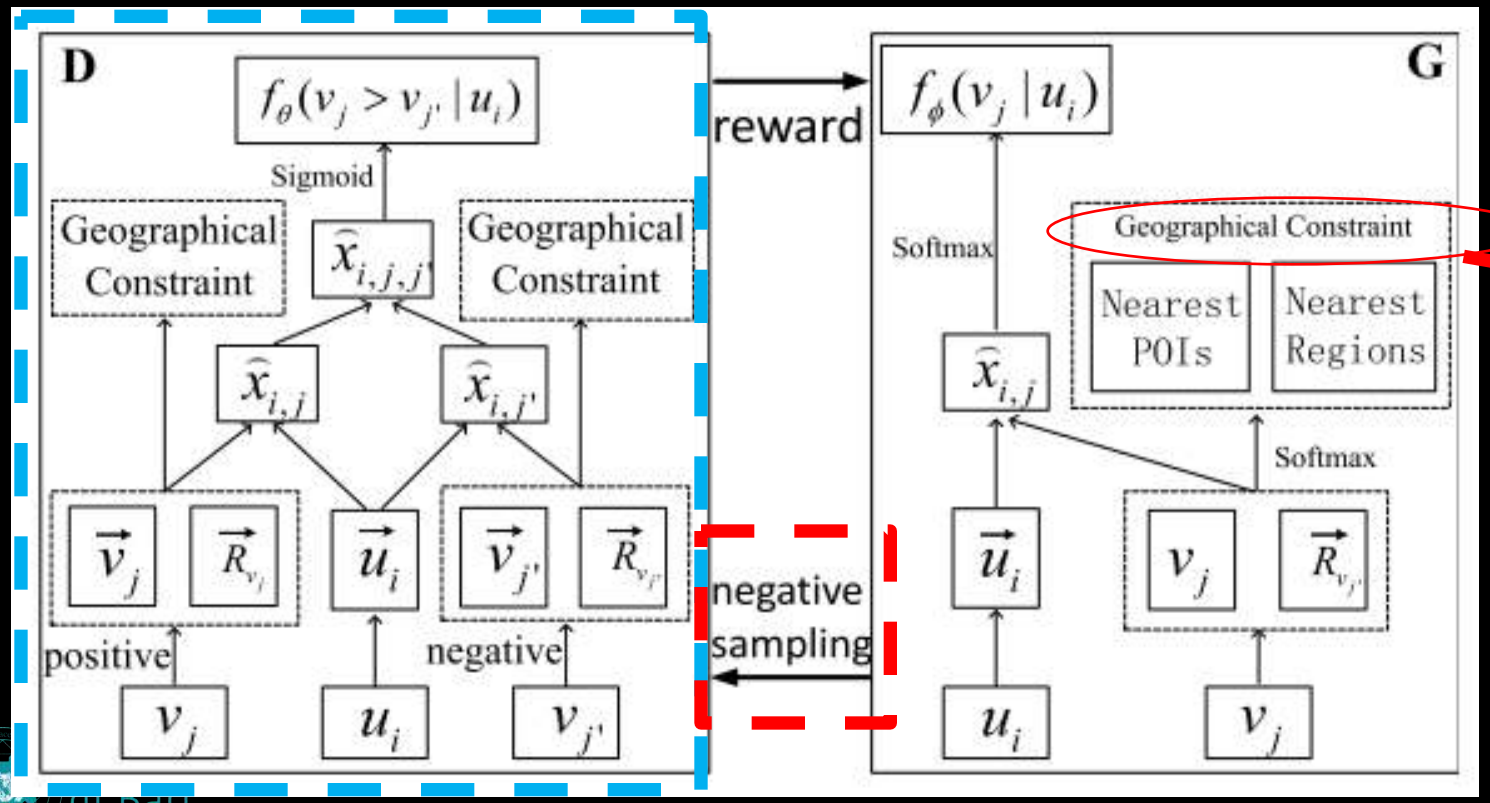
1. Exploit the generator to generate **more critical negative samples**
2. Integrate **POI** feature and **region** feature

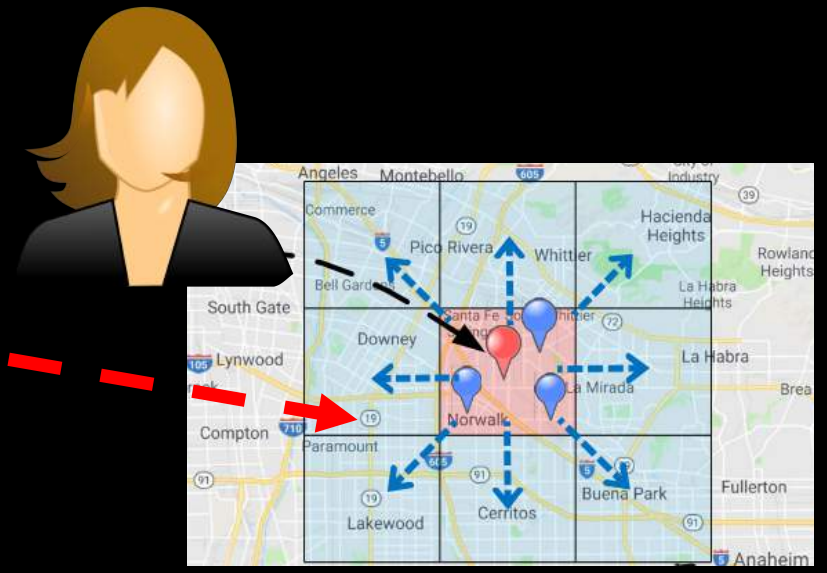
CONTEXTUAL RECOMMENDATION - **GEOGRAPHICAL**:

GEO-ALM

[ZHI-JIE WANG ET AL., IJCAI'19]

Acts as the RECOMMENDER





Discrete Item Sampling?

REINFORCE

CROSS-DOMAIN RECOMMENDATION:

DEEP ADVERSARIAL SOCIAL RECOMMENDATION

[WENQI FAN ET AL., IJCAI'19]

PROBLEM

1. Unify user representation for the **user-item interactions** (item domain) and **user-user connections** (social domain).
2. Negative sampling techniques to provide **informative** guidance during training

PROPOSAL

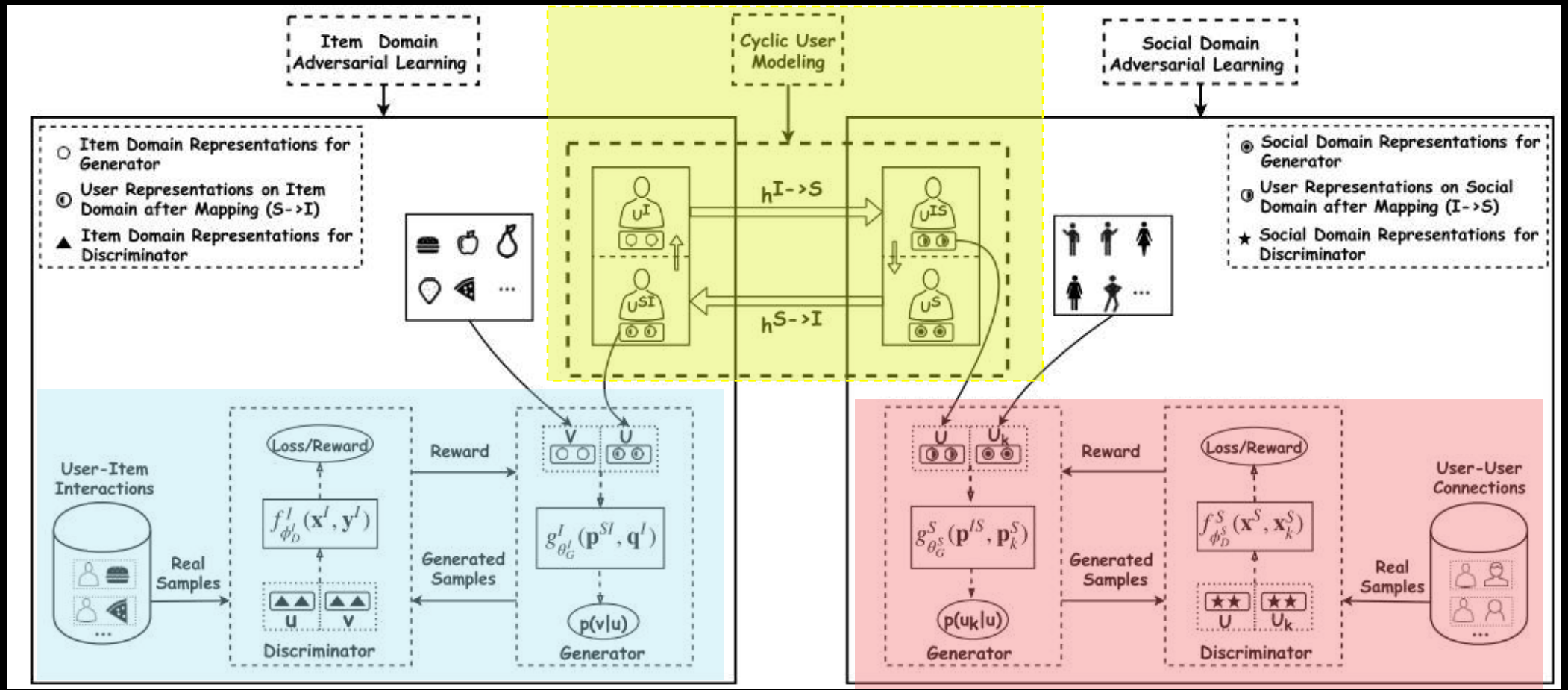
1. Adopts a nonlinear mapping method to **transfer** users' information between **social domain** and **item domain** using **adversarial learning**.
2. Generate **informative** negative samples to guide the training

CROSS-DOMAIN RECOMMENDATION:

DEEP ADVERSARIAL SOCIAL RECOMMENDATION

[WENQI FAN ET AL., IJCAI'19]

The model = **cyclic user modeling** + **item domain** and **social domain** adversarial learning.



Politecnico di Bari

CROSS-DOMAIN RECOMMENDATION:

DEEP ADVERSARIAL SOCIAL RECOMMENDATION

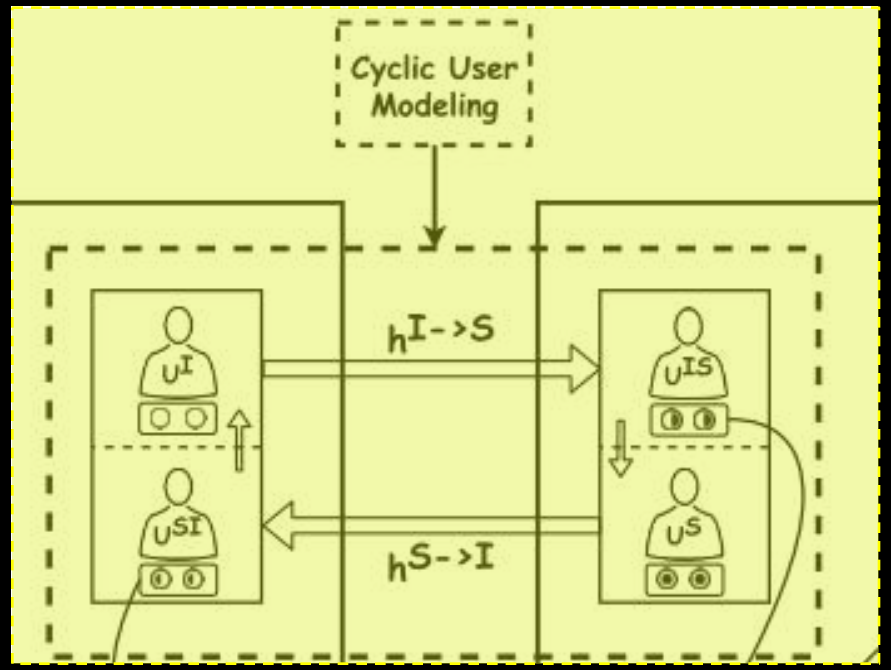
[WENQI FAN ET AL., IJCAI'19]

Cyclic user modeling

- transfer **user's information** from the **social domain** to the **item domain**
- transfer **user's information** from the **item domain** to the **social domain**

Learn an:

intra-domain user representation



Politecnico di Bari

CROSS-DOMAIN RECOMMENDATION:

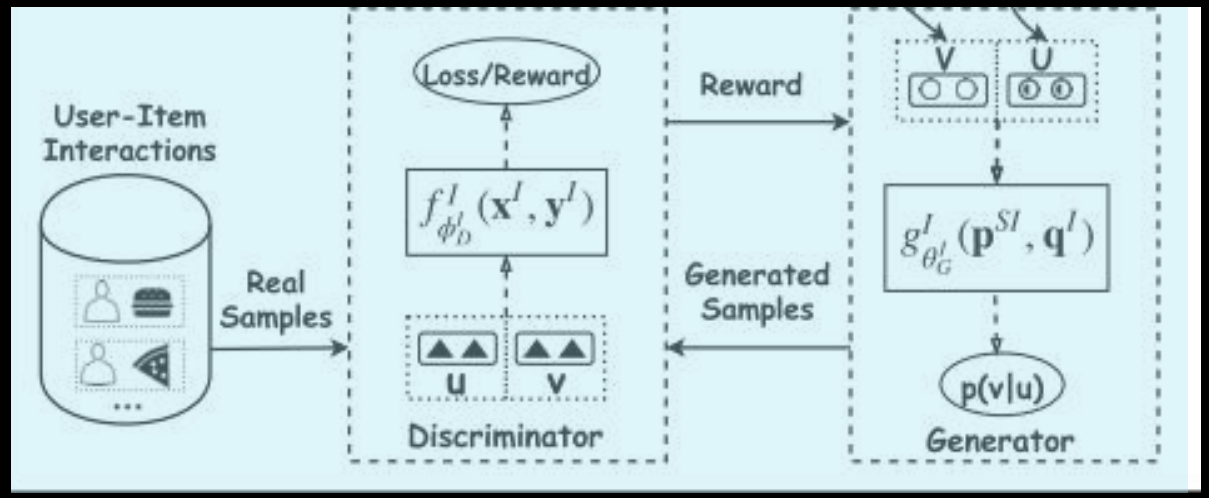
DEEP ADVERSARIAL SOCIAL RECOMMENDATION

[WENQI FAN ET AL., IJCAI'19]

Item domain adversarial learning

- Address the limitation of **negative sampling** for ranking recommendation.

G learns to produce **more informative negative** examples under the influence of **transferred social information**



Discrete Sampling? REINFORCE

Politecnico di Bari

CROSS-DOMAIN RECOMMENDATION:

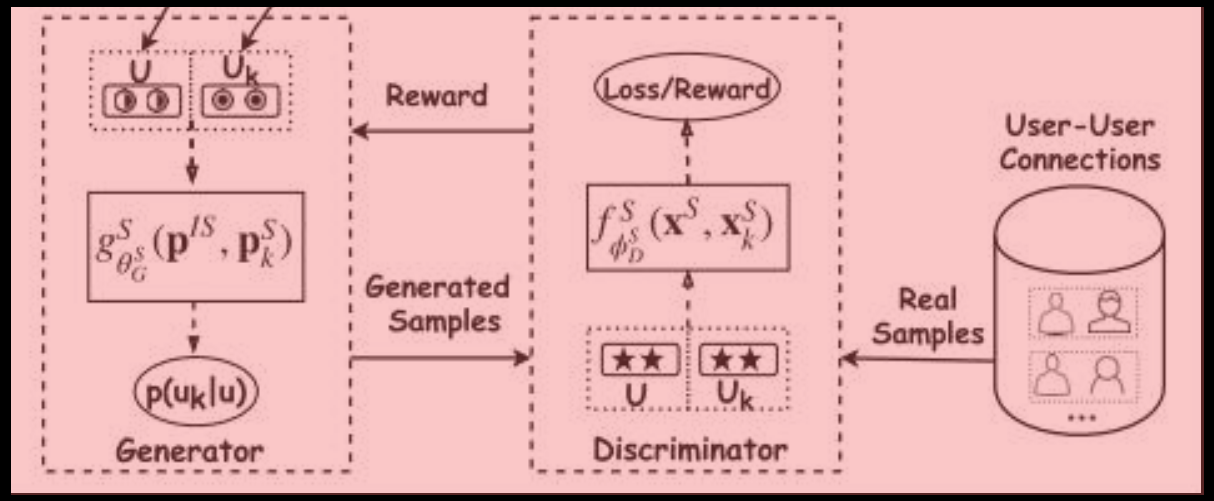
DEEP ADVERSARIAL SOCIAL RECOMMENDATION

[WENQI FAN ET AL., IJCAI'19]

Social domain adversarial learning

- Address the limitation of **negative sampling** for social recommendation.

G learns to produce **more informative negative** (most relevant social-connections) examples under the influence of **transferred item information**



Discrete Sampling? REINFORCE

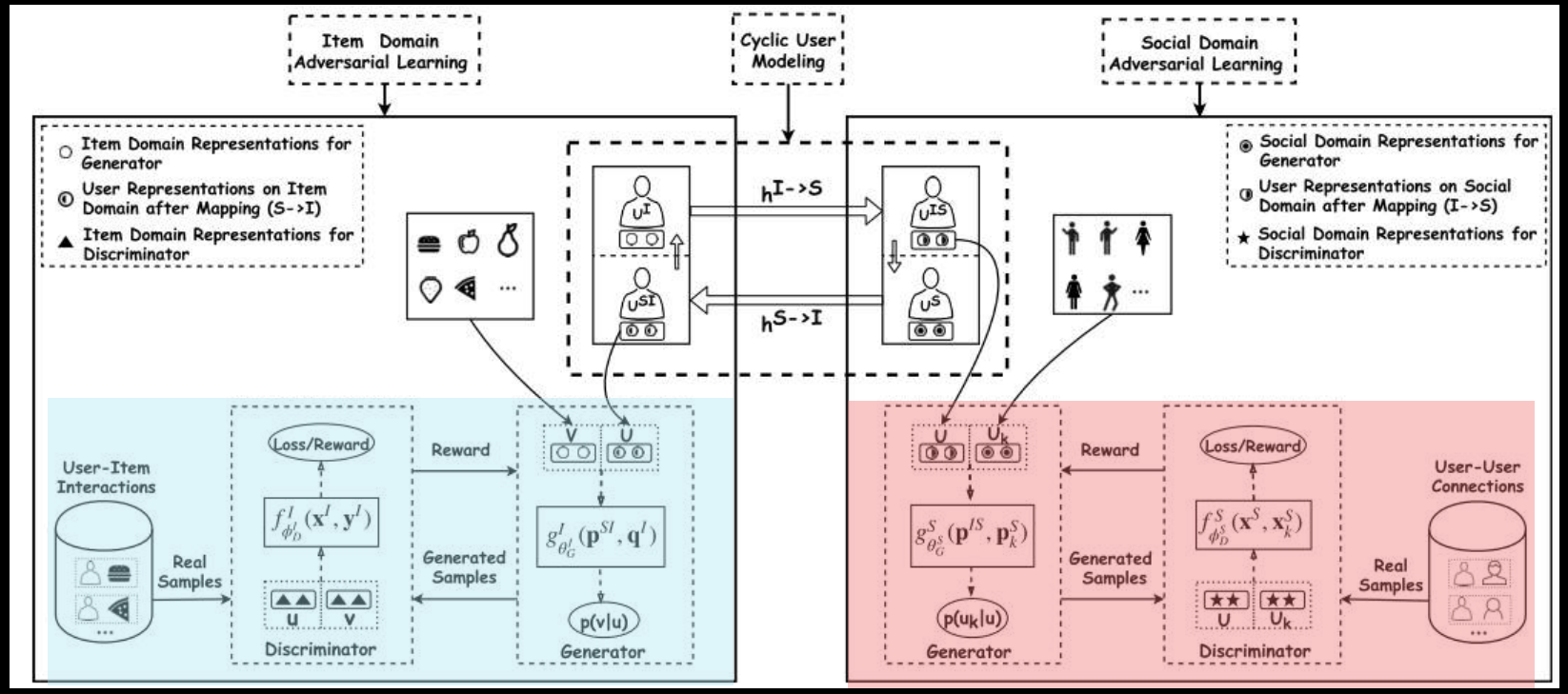
Politecnico di Bari

CROSS-DOMAIN RECOMMENDATION:

DEEP ADVERSARIAL SOCIAL RECOMMENDATION

[WENQI FAN ET AL., IJCAI'19]

When the training is finished, the embedding learned by the **social** and **item** generators are used to perform recommendation



Politecnico di Bari

CROSS-DOMAIN RECOMMENDATION:

DEEP ADVERSARIAL SOCIAL RECOMMENDATION

[WENQI FAN ET AL., IJCAI'19]

Datasets	Metrics	Algorithms						
		BPR	IRGAN	SBPR	SocialMF	DeepSoR	GraphRec	**DASO**
Ciao	Precision@3	0.0154	0.0274	0.0211	0.0260	0.0310	0.0374	**0.0462**
	Precision@5	0.0137	0.0245	0.0204	0.0218	0.0240	0.0326	**0.0451**
	Precision@10	0.0102	0.0239	0.0178	0.0155	0.0201	0.0265	**0.0375**
	NDCG@3	0.0254	0.0337	0.0316	0.0312	0.0380	0.0392	**0.0509**
	NDCG@5	0.0299	0.0350	0.0335	0.0364	0.0356	0.0373	**0.0514**
	NDCG@10	0.0315	0.0376	0.0379	0.0373	0.0396	0.0382	**0.0518**
Epinions	Precision@3	0.0046	0.0138	0.0096	0.0100	0.0105	0.0156	**0.0208**
	Precision@5	0.0042	0.0104	0.0089	0.0090	0.0098	0.0123	**0.0173**
	Precision@10	0.0035	0.0080	0.0066	0.0071	0.0086	0.0102	**0.0140**
	NDCG@3	0.0099	0.0175	0.0136	0.0176	0.0160	0.0183	**0.0226**
	NDCG@5	0.0128	0.0177	0.0152	0.0196	0.0183	0.0182	**0.0217**
	NDCG@10	0.0169	0.0202	0.0198	0.0202	0.0200	0.0217	**0.0234**

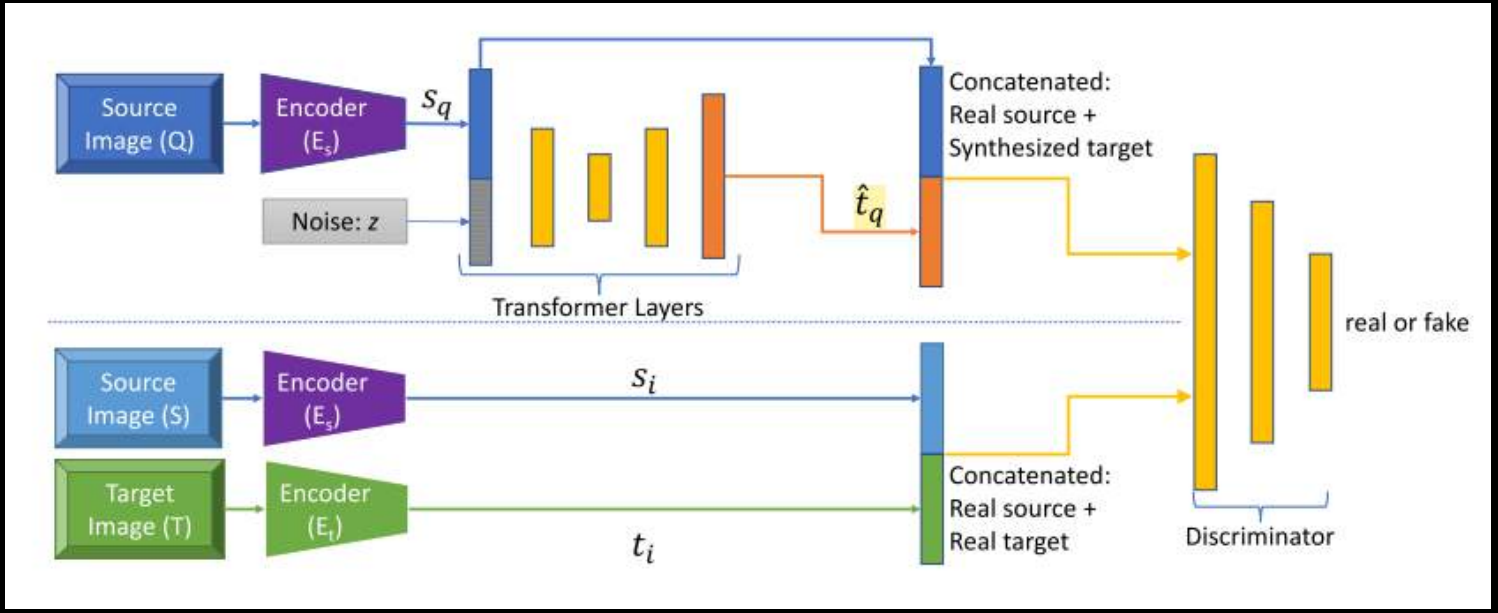
COMPLEMENTARY RECOMMENDATION: CRAFT

[HUYNH, CIPTADI ET AL., ECCV'18]

Generator: Transformer Network

learns to **generate** features of the **complementary items** conditioned on the query item.

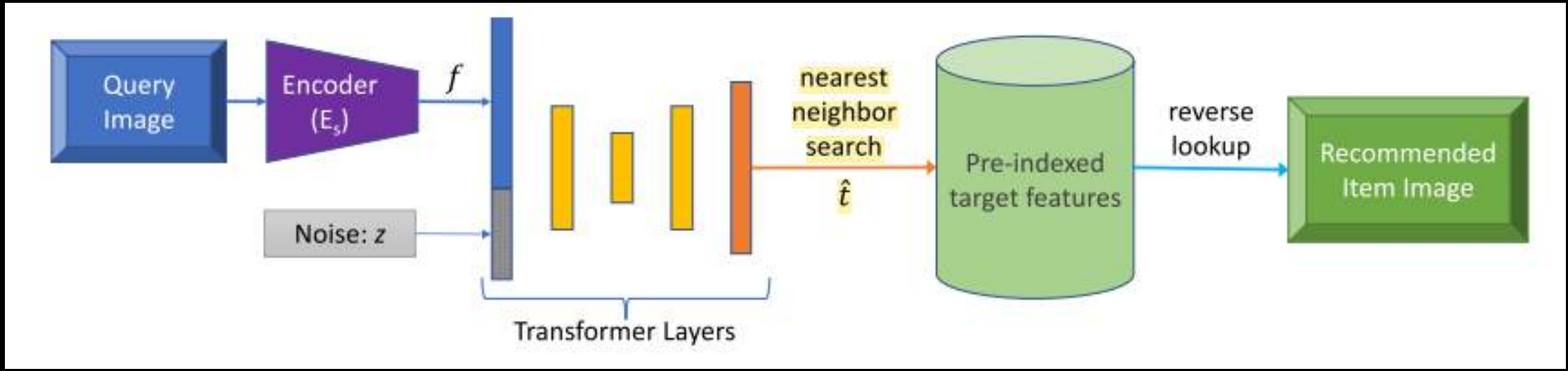
Architecture for the CRAFT framework.



COMPLEMENTARY RECOMMENDATION: CRAFT

[HUYNH, CIPTADI ET AL., ECCV'18]

Generate Recommendations



COMPLEMENTARY RECOMMENDATION: CRAFT

[HUYNH, CIPTADI ET AL., ECCV'18]

Complementary Recommendations

QUERY

CRAFT

Nearest-neighbors

Incompatible

Green box are the accepted recommendation by a fashion specialist

Politecnico di Bari

COMPLEMENTARY RECOMMENDATION:
C+GAN
[KUMAR AND DAS GUPTA, AI FOR FASHION@KDD'19]

GOAL? find best fashion clothes when the user sets a query

$$p(bottom|top)$$

The **Generator** combines several loss components to model distinct characteristics:

1. **MSE** Loss: generated image stays close to the target image
2. **Perceptual** Loss: reconstructed image and the original input match each other under
3. **Adversarial** Loss: encourages the network to prefer solutions that reside on the manifold of natural images

COMPLEMENTARY RECOMMENDATION:

C+GAN

[KUMAR AND DAS GUPTA, AI FOR FASHION@KDD'19]

Politecnico di Bari

OTHER APPLICATIONS: SECURITY

ADVERSARIAL ATTACKS ON AN OBLIVIOUS RECOMMENDER

[CHRISTAKOPOULOU AND BANERJEE, RECSYS'19]

PROBLEM

1. user-rating matrix poisoning attack in an optimized way
2. Unnoticeability of fake profiles

PROPOSAL

1. Zero-order optimization techniques to overcome the challenge that the adversary does not have access to the recommender's gradient
2. Use GANs for generating realistic fake-user samples

OTHER APPLICATIONS: SECURITY

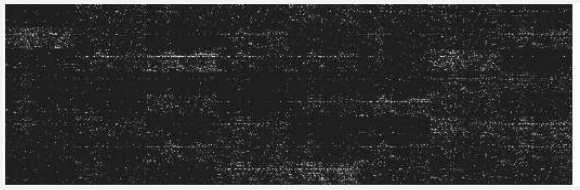
ADVERSARIAL ATTACKS ON AN OBLIVIOUS RECOMMENDER

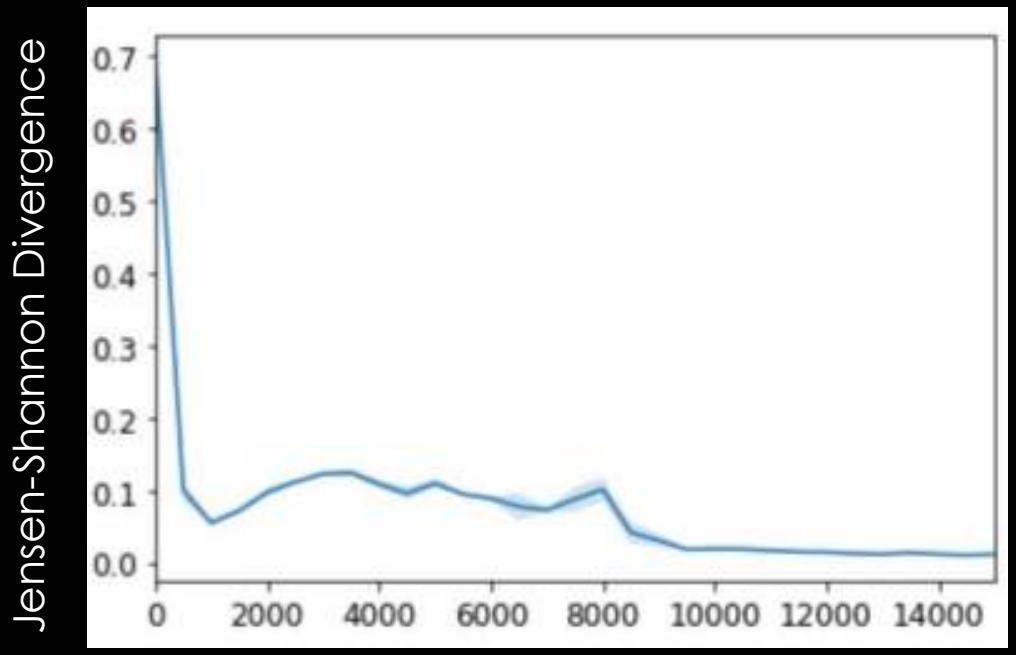
[CHRISTAKOPOULOU AND BANERJEE, RECSYS'19]

METHOD:

Transform each user profile in an image to work with standard GAN for computer vision.

User-to-image transformation





"GANs can produce fake user samples whose distribution is close to the real user distribution."

[CHRISTAKOPOULOU AND BANERJEE, RECSYS'19]

Politecnico di Bari

Name	Year	Generator							Discriminator					
		Linear	MLP	CNN	AE	VAE	RNN-LSTM	RNN-GRU	Linear	MLP	CNN	AE	RNN-LSTM	RNN-GRU
Collaborative Rec.														
IRGAN[126]	2017	✓							✓					
CFGAN[12]	2018		✓							✓				
Chae et a.[13]	2018				✓				✓					
CAAE[14]	2019				✓				✓					
CGAN[115]	2019					✓				✓				
CALF[20]	2019			✓							✓			
PD-GAN[133]	2019	✓							✓					
LambdaGAN[129]	2019	✓							✓					
VAEGAN[140]	2019					✓				✓				
APL[109]	2019	✓							✓					
RsyGAN[138]	2019				✓					✓				
Graph-based Collaborative Rec.														
GraphGAN[124]	2018	✓							✓					
GAN-HBNR [7]	2018				✓							✓		
VCGAN[148]	2018													
Hybrid Collaborative Rec.														
VAE-AR[62]	2017					✓				✓				
DVBPR[53]	2017			✓							✓			
RGD-TR[66]	2018		✓							✓				
aae-RS[137]	2018				✓					✓				
SDNet[18]	2019	✓							✓					
ATR[90]	2019							✓			✓			
AugCF[128]	2019		✓						✓					
RSGAN[139]	2019				✓				✓					
PRGAN[16]	2019	✓								✓				
CGAN[130]	2019	✓							✓					

SUMMARY

Prevalence of

Linear Models

and

Auto-Encoder

SUMMARY

Name	Year	Generator							Discriminator					
		Linear	MLP	CNN	AE	VAE	RNN-LSTM	RNN-GRU	Linear	MLP	CNN	AE	RNN-LSTM	RNN-GRU
Contextual Rec.														
Temporal-aware														
RecGAN[4]	2018							✓						✓
NMRN-GAN[127]	2018		✓										✓	
AAE[118]	2018				✓					✓				
PLASTIC[150]	2018	✓					✓		✓				✓	
LSIC[149]	2019	✓					✓		✓				✓	
GAN-LSTM[17]	2019						✓			✓				
Geographical-aware														
Geo-ALM[69]	2019	✓							✓					
APOIR[151]	2019							✓		✓				
Cross-domain Rec.														
VAE-GAN-CC[77]	2018					✓				✓				
RecSys-DAN[122]	2019			✓						✓				
CnGAN[86]	2019				✓					✓				
DASO[26]	2019		✓							✓				
Complementary Rec.														
CRAFT[48]	2018		✓							✓				
Yang et al.[136]	2018			✓							✓			
MrCGAN[106]	2018			✓							✓			
c$^+$GAN[61]	2019			✓							✓			

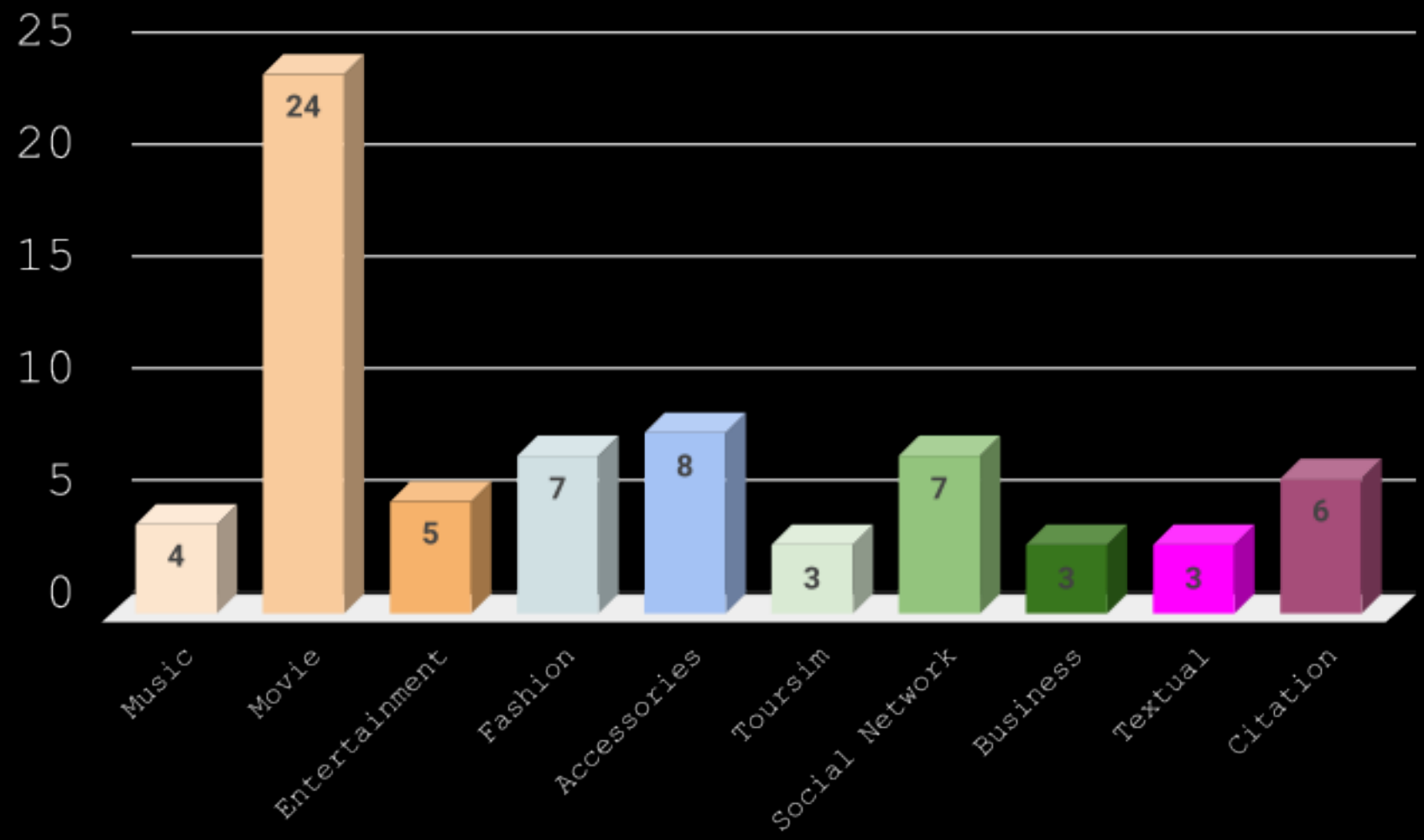
Prevalence **RNN**

Prevalence **CNN**

Politecnico di Bari

3.3 DOMAIN AND OPEN DIRECTIONS

DOMAIN



Politecnico
di Bari

OPEN DIRECTIONS

- Novel solution for the discrete sampling non-differentiability problems
- Integrate user-personalization in complementary recommendation
- Extend GAN-RF to Knowledge-aware recommender systems
- Explore GAN-based attacks to Deep-CF RS
- Verify the effectiveness of GAN-RF with an online evaluation

ADVERSARIAL MACHINE LEARNING IN RECOMMENDER SYSTEMS (AML-RECSYS)

Yashar Deldjoo Tommaso Di Noia Felice Antonio Merra

Politecnico di Bari

Polytechnic University of Bari , Bari, Italy

COMINIG SOON

Adversarial Learning in Recommender Systems: Literature Review and Future Visions

YASHAR DELDJOO, TOMMASO DI NOIA, and FELICE ANTONIO MERRA*, Polytechnic University of Bari

Interested in the survey? Drop us a line!

Corresponding author: felice.merra@poliba.it
yashar.deldjoo@poliba.it
tommaso.dinoia@poliba.it

Politecnico di Bari