Master Internship 2020

# Rivalry-Aware Recommender Systems: An Optimal Transport Approach

Supervisors: Michèle Sebag, Philippe Caillou, Gwendoline de Bie
Michele.Sebag@lri.fr, Philippe.Caillou@lri.fr, Gwendoline.de.Bie@ens.fr
Lab: TAU – CNRS – INRIA – U. Paris-Saclay

Abstract :
At the core of e-commerce are recommender systems: when facing a virtually infinite catalog, some help is required for the user to find out the items s/he might be interested in.
Recommender systems basically proceed by recommending items similar to items previously chosen, or items chosen by similar users [1]. This is done by exploiting classical collaborative filtering data: description of users $X = \{x_i, i = 1 \ldots m\}$, items $Y = \{y_j, j = 1 \ldots n\}$ and $(m, n)$ scoring matrix $M$, with $M(x_i, y_j) = 1$ iff $x_i$ selected $y_j$.
In most contexts, it does not harm to recommend the same item to many people (how many blockbusters did you see this summer ?); in contexts related to human resources however, e.g. job market or matrimonial market, one might want to avoid recommending all users the same job ad, or the same potential significant other.

The goal of the internship is to explore and design such a *rivalry-aware* recommender system (RARs), borrowing optimal transport principles:
Optimal transport (OT), assuming some cost matrix $C$ (with $C(x, y)$ denoting the cost of transporting $x$ onto $y$), aims to build a (probabilistic) mapping from a set of $x_i$ onto a set of $y_j$ (with $\gamma(x_i, y_j)$ denoting the probability of mapping $x_i$ onto $y_j$) in such a way that it incurs a minimal cost (i.e. it minimizes $\sum_{i,j} \gamma(x_i, y_j)C(x_i, y_j)$) [2,3].

The use of OT is motivated by the fact that rivalry-aware recommendation takes into account the whole population of users (as opposed to, finding the best recommendations for each user standalone): the goal is to allocate the $X$ population to the $Y$ population in a "globally optimal" manner, in the sense that each user will be recommended relevant iterms while ensuring that the same item is recommended to few users.

The student will i) propose performance indicators relevant to RARs; ii) design and implement an appropriate optimization process (in terms of scalability and accuracy); iii) if time permits, conduct experiments on real-world data; the internship will be conducted in the context of a hiring agency and data are available.

## Proposed Approach

Assume the standard recommendation problem defined from the data be solved with $H$ a recommender score, defined up to a monotonous transformation, where the top-k items recommended to user $x$ are those with highest value of $H(x, y)$. Let $H_Z$ denote the probability distribution defined from $H$, with:

$$H_Z(x, y) = \frac{H(x, y)}{\sum_{x'} H(x', y) \times \sum_{y'} H(x, y')}$$

and $H_Z(x, y) = 0$ in case $\sum_{x'} H(x', y) = 0$ or $\sum_{y'} H(x, y') = 0$. Let $U$ denote the uniform probability defined on $X \times Y$ with $U(x, y) = \frac{1}{n \times m}$.

Intuitively, the sought solution $S$ should achieve some trade-off between $H_Z$ (reflecting the data, where by construction $x_i$ selects at most a single $y_j$ and $y_j$ is selected by at most a single $x_i$) and the uninformed $U$.

This naive formalization faces limitations due to poor data and due to numerical issues. On the data side, the matching matrix $M$ is usually far from reflecting an optimal allocation: in a job market for instance, many people remain unemployed while many jobs remain unfilled. On the numerical side, regularizing the sought learning criterion using distance between distributions $S$ and $U$ raises numerical issues (e.g. a Kullback-Leibler distance $KL(S, U)$ is ill-defined unless $S(x, y) > 0$ for all $x$ and $y$) and/or scalability issues (e.g. using Maximum Mean Discrepancy [4] is in $(mn)^2$).

The limitation due to poor data can be handled by considering instead of $M$ a pseudo scoring matrix $M'$ with:

$$M' = M + \alpha R \odot U$$

with $\odot$ the elementwise matrix multiplication and $R(x, y) = 1$ iff $x$ did not select any $y'$ and $y$ was not selected by any $x'$. Intuitively, $M'$ considers that vacant $y_i$ are selected with a small probability $\alpha$ by any $x_j$ who did not select any item.

The limitation wrt regularization, that actually aims at avoiding rivalry, can be handled by bounding $\langle S(x, \cdot), S(x', \cdot)\rangle$ (the alignment between $x$ and $x'$ preferences).

The internship requires excellent theoretical and algorithmic skills (programming environment in Python).

## Références


[1] *Recommender Systems*, Aggarwal, Springer 2016.

[2] *Computational Optimal Transport*, Peyré & Cuturi, Foundations and Trends in Machine Learning 11(5-6), 2019

[3] *CO-Optimal Transport*, Redko et al., ArXiv 2020

[4] *A Kernel Two-Sample Test*, Arthur, Gretton et al., JMLR 2012.