Bachelor projects 2022

Data Intensive systems group

In addition to the projects described in this document, we can also provide several projects in **machine learning algorithms, data mining, and data management**. Feel free to contact any of us (Ira Assent, Panos Karras, Davide Mottin, Cigdem Aslay) for more information.

1. Evolutionary games and spatial upstream reciprocity

Advisor: *P. Karras (piekarras@gmail.com), A. Pavlogiannis (pavlogiannis@cs.au.dk)*

Game theory studies how selfish individuals shape their behavior according to the behavior of their neighbors. One apparent paradox to selfishness is *cooperation: why do we spend time/effort/resources in cooperating with others?* Evolutionary game theory takes a Darwinian viewpoint to behavior, and looks for simple mechanisms that make cooperation evolutionary advantageous. One such mechanism is *upstream reciprocity: when you receive help, you feel good and might help others*. Upstream reciprocity alone does not promote cooperation, but it does so when the population interacts on a simple 1-dimensional structure.

The project aims to extend this study of upstream reciprocity to more realistic, 2-dimensional population structures, such as 2-dimensional grids. It consists of 2 stages:

1. Implement the model of evolutionary games with upstream reciprocity on graphs: The model takes as input a 2-dimensional graph (e.g., a grid, or more generally a planar graph), and some parameters of the game, and simulates the evolution of the game on the graph.
2. Use the model to phrase and answer questions in upstream reciprocity, such as “does upstream reciprocity favor cooperation on 2-dimensional structures”, and “for what parameters of the evolutionary game it does so?”

The project requires interest in discrete math/graph theory, probability, programming and simulation.

Related work:
https://royalsocietypublishing.org/doi/10.1098/rspb.2006.0125

2. Structural social balance under controversy

Advisor: *P. Karras (piekarras@gmail.com), A. Pavlogiannis (pavlogiannis@cs.au.dk)*

"The enemy of my enemy is my friend". This social axiom has been shaping interpersonal relations since the dawn of society. Structural balance theory models relationships as a signed graph: nodes correspond to entities (people, nations etc), and a signed edge between two nodes captures their relationship ("+" means "friends", "-" means "enemies"). A triangle is imbalanced when the product of its signs is “-". Imbalance creates social unrest that eventually gets resolved by one sign (randomly) changing (e.g., two enemies make amends under the uniting influence of a common friend). What is the state of the final population?

The project will study a similar setting, in the presence of a controversial subject X. Signs between nodes still capture interpersonal relations (friends/enemies). Every individual also has an opinion on X: “+” denotes support, “-“ denotes resentment. This time, triangles are formed between two nodes and X. If the nodes disagree on X, they will either become enemies or one will convince the other to change stance on X.

The project aims to develop and study this stochastic model of social balance. It consists of 2 stages:

1. Implement this model on graphs. Given an input graph and other parameters of the process, the task is to simulate the evolution of the graph until balance is reached.
2. Use the model to study basic questions of social balance, such as the number of friendships made and broken in this process.

The project requires interest in discrete math/graph theory, probability, programming and simulation.

Related work:
https://en.wikipedia.org/wiki/Signed_graph
http://physics.bu.edu/~redner/pubs/pdf/dresden.pdf

3. Similarity Search with Dynamic Time Warping

Advisor: *P. Karras (piekarras@gmail.com), Ira Assent (ira@cs.au.dk)*

Similarity search in time-series (or data-series) databases finds application in finance, health care, energy installation monitoring, and bioinformatics. In its offline formulation, the problem is to extract records from a repository, which are the most similar to a given query record. The way similarity is defined plays a cardinal role. The Euclidean distance metric, which is most popular, cannot capture similarity in the presence of shifts, skews, and discontinuities. For that purpose, one can use other distance measures, such as Dynamic Time Warping (DTW), which allows the flexibility to capture such features. However, it remains a challenge to devise index that enables efficient DTW- based similarity search over large time series databases. The curse of dimensionality, which arises in such problems, becomes even more debilitating in the case of DTW measures. Past approaches to indexing DTW lack the versatility to offer multiple levels of resolution that can speed up search and retrieve results more efficiently.

In this thesis project, you will study previous works in the area, using existing code base, and develop a novel solution for DTW-based data-series similarity search using adaptations of state-of-the-art methodologies. The conducted work will have components of research, data preprocessing, implementation, and experimentation.

Related work:
Ira Assent, Marc Wichterich, Ralph Krieger, Hardy Kremer, Thomas Seidl: Anticipatory DTW for Efficient Similarity Search in Time Series Databases. PVLDB 2(1): 826-837 (2009)
Shrikant Kashyap, Panagiotis Karras: Scalable kNN search on vertically stored time series. KDD 2011: 1334-1342

4. ReliK: Reliable Knowledge Graph Embeddings

Advisor: *Davide Mottin* (*davide@cs.au.dk*)

Graph Embeddings and graph representation learning [1] are methods that transform a graph (i.e., a network of nodes connected through edges) into a set of points of a multi-dimensional space. Recently, the best methods rely on neural architectures, such as neural networks to exploit the power of non-linear transformations. Unfortunately, these methods are not easily explainable and the connection between the embeddings and the graph are lost, with the consequence that practitioners in different areas from biology to journalists, cannot understand the reason for a specific result. For instance, if we embed a graph of news, it might happen that an article concerning Covid is associated with no-vax protests, but the reason is obscure.

This projects aims at providing and testing a reliability measure over graph embeddings, in particular knowledge graph embeddings [2] that are specifically designed to embed graphs in which nodes are entities, such as Moderna and Covid, and edges are connections, such as Moderna *has_vaccine_agains* Covid.

Tasks:

- Reading and understanding the main concepts of Knowledge graph embeddings
- Getting familiar with a library for knowledge graph embeddings (pykeen, ampligraph, …)
- Perform preliminary experiments testing which structures are preserved in a knowledge graph embedding (e.g., edges, degree, communities, patterns, rules, …)
- Test different reliability measures.
- Plot and report the results.

Related work:

[1] Hamilton W. Graph representation learning book https://www.cs.mcgill.ca/~wlh/grl_book/
[2] Knowledge graph embeddings: a Tutorial https://kge-tutorial-ecai2020.github.io/

5. BeCom: A community benchmark for community detection

Advisors: *Davide Mottin* (*davide@cs.au.dk*) *and Ira Assent* (ira@cs.au.dk)

> *"We cannot live only for ourselves. A thousand fibers connect us with our fellow men."*
> – Herman Melville

Community detection [1] is the task of finding nodes in a graph (i.e., a network of nodes connected through edges) that participate in similar activities. In a social network, a community is a group of people that share similar interests. While most of the methods detect communities that are non-overlapping, and therefore, no person should belong to more than one community, in reality communities overlap. A few methods have been proposed to study the overlapping community detection problem, from more traditional approaches [2] to the recent ones that use special neural networks for graphs called graph neural networks [3]. Although overlapping community detection is important, there is still no comprehensive benchmark of datasets and methods to test.

This project aims at constructing a benchmark for overlapping community detection, by collecting data from available resources (e.g., Twitter, Wikipedia) and testing some of the latest methods. The results can be published as a new resource and benefit thousands of researchers around the world.

Tasks:

- Reading and understanding the problem of community detection and the main algorithmic ideas
- Familiarize with the current datasets and the limitations
- Retrieve data and ground truth communities from known sources
- Construct a comprehensive benchmark of datasets
- Report the main characteristics of the datasets and report results of the main algorithms
- [Optional] Present a resource paper at a top conference.

[1] Su, X., Xue, S., Liu, F., Wu, J., Yang, J., Zhou, C., Hu, W., Paris, C., Nepal, S., Jin, D. and Sheng, Q.Z., 2021. A Comprehensive Survey on Community Detection with Deep Learning. *arXiv preprint arXiv:2105.12584*.
[2] Yang, J. and Leskovec, J., 2013, February. Overlapping community detection at scale: a nonnegative matrix factorization approach. In *Proceedings of the sixth ACM international conference on Web search and data mining* (pp. 587-596).
[3] Shchur O, Günnemann S. Overlapping community detection with graph neural networks. arXiv preprint arXiv:1909.12201. 2019 Sep 26.

6. SAGA: Scalable Algorithms for Graph Alignment

Advisor: *Davide Mottin* (*davide@cs.au.dk*) *and Panos Karras* (piekarras@gmail.com)

Graph alignment is the problem of finding correspondences among nodes in two (or more) graphs. The problem has been studied extensively and multiple solutions have been proposed to tackle the problem (for instance [1,2,3]). In our recent work we showed how these algorithms do not scale up to graphs with a large number of nodes (> 10^4 nodes) and therefore cannot align big social or biological graphs.

In this project, we will explore how to create scalable algorithms that maintain the same quality guarantees. In particular, we will start from one algorithm (potentially [3]) and try to increase its efficiency. If successful, the project will pave the way to more practical algorithms for graph alignment.

Tasks:

- Reading, understanding the relevant literature in graph alignment
- Replicate some small test with existing algorithms and our library for graph alignment https://github.com/constantinosskitsas/Framework_GraphAlignment
- Implement some GPU optimization (parallelism) or existing approximate algorithms to render algorithms more scalable.
- Test the results on large datasets of increasing size

Related work:

[1] Heimann, M., Shen, H., Safavi, T. and Koutra, D., 2018, October. Regal: Representation learning-based graph alignment. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management (pp. 117-126). ACM.

[2] Nassar, H., Veldt, N., Mohammadi, S., Grama, A. and Gleich, D.F., 2018, April. Low rank spectral network alignment. In Proceedings of the 2018 World Wide Web Conference (pp. 619-628). International World Wide Web Conferences Steering Committee.
[3] Hermanns, J., Tsitsulin, A., Munkhoeva, M., Bronstein, A., Mottin, D. and Karras, P., 2021. GRASP: Graph Alignment through Spectral Signatures. arXiv preprint arXiv:2106.05729.

7. Studying the fairness of ML models

Advisor: *Ira Assent (ira@cs.au.dk)*

In this project, we study a novel approach for evaluating the fairness of Machine Learning models. As Machine Learning is increasingly used to support decision-making that affects people’s lives, e.g. when granting access to bank loans or in deciding recidivism cases, issues of fairness are of increasing concern. Specifically, the question is whether ML reproduce or even introduce discriminative patterns into decision-making, typically indirectly via biased data sources.
Building on an existing code base in our research group (python), publicly available data sets with sensitive attributes, as well as a draft report, the project should investigate a recent proposal by two students to check the fairness of models using a sampling-based approach. The goal is to devise a proper experimental evaluation that validates the claims in the report, and that further clarifies the methodology and its underlying reasoning, and possibly even expands it further.

Tasks

- Create an overview over existing fairness in machine learning models
- Review claims and approach in draft report
- Familiarize with existing code base, re-run experiments
- Identify and pre-process suitable data sources (setting and discussing requirements)
- Devise new experiments (using tools, data), evaluate, and derive conclusions
- Describe and analyze methodology in accordance with experimental findings

Related work:
Surveys with overview over fairness in machine learning:
[1] https://arxiv.org/pdf/2010.04053.pdf
[2] https://arxiv.org/pdf/1908.09635.pdf

8. Algorithms for Approximating Betweenness Centrality on Large Graphs

Advisor: *Cigdem Aslay* (cigdem@cs.au.dk)

Centrality measures are fundamental concepts in graph analysis that help to quantify the importance of nodes in a variety of applications including social, biological and communication networks. Betweenness centrality is a widely used centrality measure that defines the importance of a node proportionally to the fraction of all-pairs shortest paths passing through that node. Computation of betweenness centrality is very costly for large graphs, therefore, many approximate methods based on sampling have been proposed.

The goal of the project is to provide a theoretical and experimental comparison of the existing algorithms approximating betweenness centrality on static and dynamic graphs. The final report should include

- literature review on exact and approximation algorithms,
- a theory section covering the basic analysis of different approximate methods,
- a summary of the implemented algorithms,
- an experimental evaluation of the implemented algorithms comparing their accuracy and efficiency.

Related work:

[1] Brandes, Ulrik. "A faster algorithm for betweenness centrality." Journal of mathematical sociology 25.2 (2001): 163-177.

[2] Yoshida, Yuichi. "Almost linear-time algorithms for adaptive betweenness centrality using hypergraph sketches." Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. 2014.

[3] Riondato, Matteo, and Eli Upfal. "ABRA: Approximating betweenness centrality in static and dynamic graphs with rademacher averages." ACM Transactions on Knowledge Discovery from Data (TKDD) 12.5 (2018): 1-38.

9. Efficient and Scalable Influence Maximization in Online Social Networks

Advisor: *Cigdem Aslay* (*cigdem@cs.au.dk*)

Given a social network G and a positive integer k, the influence maximization problem asks for k nodes (in G) whose adoptions of a certain idea or product can trigger the largest expected number of follow-up adoptions by the remaining nodes. This problem has been extensively studied in the literature, and the state-of-the-art algorithms run in $O((k + \ell)(n + m) \log n/\varepsilon\ 2)$ expected time and return a $(1 - 1/e - \varepsilon)$-approximate solution with at least $1 - 1/n^{\ell}$ probability. Although these algorithms provide the same worst-case guarantee, their empirical efficiencies vary greatly.

The goal of the project is to provide a theoretical and experimental comparison of the existing influence maximization algorithms. The final report should include

- literature review on exact and approximation algorithms,
- a theory section covering the basic analysis of different methods,
- a summary of the implemented algorithms,
- an experimental evaluation of the implemented algorithms comparing their accuracy and efficiency.

Related work:

[1] Tang Y, Xiao X, Shi Y. Influence maximization: Near-optimal time complexity meets practical efficiency. InProceedings of the 2014 ACM SIGMOD International Conference on Management of data 2014 Jun 18 (pp. 75-86).

[2] Tang Y, Shi Y, Xiao X. Influence maximization in near-linear time: A martingale approach. InProceedings of the 2015 ACM SIGMOD International Conference on Management of Data 2015 May 27 (pp. 1539-1554).

[3] Tang J, Tang X, Xiao X, Yuan J. Online processing algorithms for influence maximization. InProceedings of the 2018 ACM SIGMOD International Conference on Management of Data 2018 May 27 (pp. 991-1005).

10. Creating and personalizing knowledge graphs

Advisor: *Ira Assent (ira@cs.au.dk)*

This is a rather open-ended project, in which the goal is to analyze and evaluate empirically techniques for creating, maintaining, expanding knowledge graphs. Knowledge graphs are increasingly common, and have found use in a number of applications [1]. Given the dynamic development of methods and tools, we would like to establish an overview, in particular for the medical domain [2,3].

The project can be adapted to the interests of the students within this general setup. In particular, recent research articles on knowledge graph analysis provide a theoretical foundation [4]; the empirical evaluation can either focus on an in-depth analysis of a single technique against requirements or on a comparative evaluation of a number of methods. Projects include, among others, machine learning methods to complete knowledge graphs and predict new connections, explanations of machine learning methods, graph summarization, modern queries on knowledge graphs.

Related work:

[1] Industry-scale knowledge graphs https://queue.acm.org/detail.cfm?id=3332266

[2] Tutorial with overview over recent research: https://kgtutorial.github.io/

[3] Blog article discussion use of tool Grakn.AI which can serve as inspiration: https://blog.grakn.ai/text-mined-knowledge-graphs-beyond-text-mining-1ff207a7d850

[4] Song, Q., Wu, Y., Lin, P., Dong, L. X., & Sun, H. (2018). Mining summaries for knowledge graph search. *IEEE Transactions on Knowledge and Data Engineering*, *30*(10), 1887-1900.

Tasks

- Select methods and tools that are adequate for managing and analyzing medical records
- Identify and pre-process suitable data sources (setting and discussing requirements, basic coding)
- Create knowledge graphs (using tools, data)
- Evaluate the performance (coding test scripts, running experiments)
- (optional) Discuss and relate to recent research articles