Towards Recommender Systems with Community Detection and Quantum Computing

Riccardo Nembrini
Politecnico di Milano and ContentWise
Italy
riccardo.nembrini@polimi.it

Costantino Carugno
Politecnico di Milano and ContentWise
Italy
costantino.carugno@polimi.it

Maurizio Ferrari Dacrema
Politecnico di Milano
Italy
maurizio.ferrari@polimi.it

Paolo Cremonesi
Politecnico di Milano
Italy
paolo.cremonesi@polimi.it

ABSTRACT

After decades of being mainly confined to theoretical research, Quantum Computing is now becoming a useful tool for solving realistic problems. This work aims to experimentally explore the feasibility of using currently available quantum computers, based on the Quantum Annealing paradigm, to build a recommender system exploiting community detection. Community detection, by partitioning users and items into densely connected clusters, can boost the accuracy of non-personalized recommendation by assuming that users within each community share similar tastes. However, community detection is a computationally expensive process. The recent availability of Quantum Annealers as cloud-based devices, constitutes a new and promising direction to explore community detection, although effectively leveraging this new technology is a long-term path that still requires advancements in both hardware and algorithms. This work aims to begin this path by assessing the quality of community detection formulated as a Quadratic Unconstrained Binary Optimization problem on a real recommendation scenario. Results on several datasets show that the quantum solver is able to detect communities of comparable quality with respect to classical solvers, but with better speedup, and the non-personalized recommendation models built on top of these communities exhibit improved recommendation quality. The takeaway is that quantum computing, although in its early stages of maturity and applicability, shows promise in its ability to support new recommendation models and to bring improved scalability as technology evolves.

CCS CONCEPTS

• **Information systems → Recommender systems**; • **Computer systems organization → Quantum computing**.

KEYWORDS

Community Detection, Recommender Systems, Quantum Computing, Quantum Annealing

ACM Reference Format:
Riccardo Nembrini, Costantino Carugno, Maurizio Ferrari Dacrema, and Paolo Cremonesi. 2022. Towards Recommender Systems with Community Detection and Quantum Computing. In *Sixteenth ACM Conference on Recommender Systems (RecSys '22), September 18–23, 2022, Seattle, WA, USA*. ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3523227.3551478

1 INTRODUCTION

Since the development of the first recommender systems, the size and complexity of the recommendation tasks have grown substantially and now include a multitude of data sources [22, 25, 32, 41]. This has been both an opportunity, opening many directions of research and development of new techniques, but also bringing important scalability constraints for the practical applicability of recommender systems to large scale domains. The growing need for higher computational power has brought the development of new architectures specialized for specific tasks, such as tensor units for linear algebra and GPUs for neural models.

Quantum computing is a new and emerging technology that promises to substantially accelerate several problems, that are difficult to solve with traditional approaches, by leveraging new computational paradigms made possible by quantum-mechanical phenomena. Among the existing paradigms, this paper focuses on Quantum Annealing (QA), which uses a special-purpose device able to sample low-energy solutions for optimization problems. Available quantum annealers have the highest number of qubits compared to other architectures, and are powerful enough to solve small but interesting machine learning and classification problems [27].

The aim of this work is to conduct an exploratory study of this cutting-edge technology that aims to assess whether it can be applied in practice to a relevant problem for the recommender systems field which is known to be computationally difficult, community detection. The goal of effectively leveraging the potential of quantum computing is indeed a long and complex journey which brings together challenges in hardware development but also in understanding how to use it, which tasks fit well the hardware capabilities and how we can integrate it into successful applications. To this end, a further goal of this work is to increase the awareness of

RecSys '22, September 18–23, 2022, Seattle, WA, USA

ACM ISBN 978-1-4503-9278-5/22/09…$15.00
https://doi.org/10.1145/3523227.3551478

the community about this new technology which is more accessible than its reputation might suggest. In this work we present a simple but effective way of using community detection to boost the quality of already available recommendation models. Although the experimental analysis is still limited by the constraints of this innovative technology, the results are promising and open several research questions and possible applications. In particular, this paper addresses the following research questions:

- **RQ1:** Can community detection boost the recommendations quality of non-personalized recommendation models?
- **RQ2:** Can quantum annealing be used to rapidly find useful communities?

2 QUANTUM ANNEALING FOR OPTIMIZATION PROBLEMS

Quantum computing is a new computational paradigm that aims to leverage quantum-mechanical phenomena in order to significantly accelerate the solution of certain problems that are difficult to solve on traditional computers [26]. This paper will focus on Quantum Annealers, which are special-purpose devices, also called Quantum Processing Units (QPU) that can be used to solve optimization problems. In particular, the D-Wave Advantage QPU is accessible on the cloud[1], and has 5640 qubits, where every qubit is connected to 15 others [7].

A quantum annealer operates by representing the optimization problem in terms of the *energy* landscape of a physical system, which will represent the quality of a specific solution: better solutions have lower energy. The device operates by starting from an initial default configuration with an easy-to-obtain quantum state, and then carefully introduces the components of the problem one desires to solve. At the end of this evolution, or *annealing*, the qubits have reached a state that minimizes the overall system's energy and therefore are in an optimal solution. There are several steps to consider when using a QPU:

Problem Formulation. The QPU requires the optimization problem to be formulated as a Quadratic Unconstrained Binary Optimization (QUBO) problem, which is defined as follows:

$$\min_{x \in \{0,1\}^m} y = x^T Q x \quad (1)$$

where y is the energy of the system, $x \in \{0,1\}^m$ is a vector of m binary variables and Q is an $m \times m$ symmetric matrix that defines the function to optimize. Two variables x_i and x_j are said to be *connected* if the corresponding Q_{ij} element in the Q matrix is non-zero. Note that the QUBO formulation does not allow to impose hard constraints, which are instead represented as *penalties*. Several QUBO formulations for important problems exist such as graph partitioning [5, 38], Support Vector Machines [40], Restricted Boltzmann Machines [1, 2], Feature Selection [16] and optimization for resource allocation [10]. QA has also been applied to collaborative filtering hybrid recommender systems [29] and the personalization of the user interface [12]. Note that while QA requires to represent problems as a QUBO, QUBO problems can be solved also with traditional solvers that do not require quantum computing, see Section 4.2.

[1]https://www.dwavesys.com/solutions-and-products/cloud-platform/

Problem Embedding. Once the problem is formulated as a QUBO, it can be programmed on the QPU. This step is called *minor embedding* [11] and requires to bridge the gap that exists between the mathematical formulation of a specific QUBO problem (the Q matrix) and an actual physical device. There are usually two types of factors to account for: the number of qubits (which limits the size of the Q matrix) and the physical connections between them (which limits the non-zero elements in the Q matrix). The original QUBO problem is thus transformed in an equivalent one that accounts for the limited physical connections between qubits. The embedding process often inflates the number of qubits needed (up to a square factor for a fully connected model), and may lower the quality of the solution if the qubits that represent the same variable do not behave coherently.

Submitting the Problem. The QPU is available on the cloud as part of a computing-as-a-service platform. The QUBO problem, with the corresponding embedding, can be sent to the quantum computer via APIs.

Sampling Solutions. As previously mentioned, the device operates by evolving a physical system from an initial default configuration to another that depends on the problem one wishes to solve. This *annealing* phase lasts only a very short amount of time and is usually repeated many times to increase the likelihood of sampling a good solution. It is common to sample a large number of solutions, $10^2 - 10^4$, with each individual annealing lasting between $1 - 100\mu s$.

3 COMMUNITY DETECTION

In the study of real-world networks, a commonly found property is the tendency of certain nodes to form groups (or communities) that are more densely connected internally than with the rest of the network [39]. Community detection is particularly of interest in applied scenarios, as discovering these communities allows to identify nodes that share salient feature, and effectively act as meta-nodes of the graph. Among many applications, community detection has been used, for example, to discover friendship relations in social networks [14], to identify functional groups in metabolic networks [19], to extract research topics from citation networks [34], etc.

In the realm of Recommender Systems, a recent review by Gasparetti et al. [18], has shown how recommender engines based on collaborative filtering can be enhanced by using community detection. Since these systems rely on the feedback from each individual user, in the form of positive or negative interactions expressed by the user on items, the resulting interaction matrix is often sparse, and traditional collaborative filtering-based methods do not always have a way to estimate preference between users or items. Instead, by leveraging the community groups, preferences and interests of the community's users are assumed as approximations of the target user's profile [35]. One of the challenges in performing community detection for Recommender Systems, is that the user-item interaction graph is bipartite – there are two sets of independent nodes that interact only with one another. In this scenario, nodes of the same set (i.e., users, items) have only second-order connections, via a node of the other set, and it is in fact an open research question whether it is better to cluster the users and items separately, or to consider the entire graph as a whole [33, 37].

Another problem that needs to be faced when dealing with communities lies in its definition: a community is a loose concept, bound to the domain of the application of interest, and relative to the topology of the underlined network. This issue has been historically addressed with the introduction of modularity, a function that measures the quality of a partition as compared to a null model – i.e., a random graph – of similar structure [31]. Community detection thus becomes a matter of maximizing modularity by evaluating this function on the different communities that can be formed. However, in general this process can be a daunting task, as it is in fact an NP-Complete problem [8], and one often resorts to efficient approximate methods. Several greedy methods, such as the Louvain [6] and Leiden [36] algorithm, have been employed extensively in practice, although they introduce a trade-off between quality and time feasibility of the detection [23]. On the other hand, Monte-Carlo approaches, such as simulated annealing, can effectively implement an iterative sampling, which allows to analyze graphs up to 10000 nodes [17].

Inspired by simulated annealing, early-available quantum annealing devices were tested on small graphs, and it was shown that they were able to reach comparable quality to classical methods [28]. Recently, the availability of newer quantum annealers via cloud-based services, has allowed to perform modularity maximization on larger graphs, by leveraging a hybrid quantum-classical evaluation library [42, 43]. Although far from conclusive, this recent experiment highlights some prospects in employing a quantum-annealing-based community detection, in order to tackle both the scalability and solution quality problem. Further research will be needed to assess the possible improvements offered by this technology, as quantum annealers improve in qubit number and connectivity.

3.1 Community Detection as a QUBO Problem

The task of community detection is formulated as an optimization problem, with the objective of finding the communities that maximize the modularity. First, recall the mathematical formulation of modularity, as introduced by Newman [30]:

$$Q = \frac{1}{2m} \sum_{i,j} \mathcal{M}_{ij} \delta(c_i, c_j), \qquad \mathcal{M}_{ij} = A_{ij} - P_{ij}, \tag{2}$$

where A_{ij} is the adjacency matrix of the graph, P_{ij} is the real-valued probability matrix generated from the null configuration model, $\mathcal{M}_{ij}$ is the modularity matrix, c_i, c_j are communities that host the nodes i, j, and δ is a Kronecker delta, which evaluates to 1 if both i and j belong to the same community, and to 0 otherwise.

In order to generate the modularity matrix in bipartite networks, two different techniques can be employed: *bipartite modularity* [4], which uses a null model that forbids connection among nodes of the same type, and *weighted projection modularity* [13], which uses the usual modularity formulation evaluated on a new network, generated by connecting two nodes only if, in the original network, both share a connection with the same node of the other set. As described by Negre et al. [28], the obtained modularity matrix is effectively the Q matrix of Eq. (1), up to a normalization factor that accounts for the total number of connections m. The energy function to be minimized is thus simply: $\mathcal{H} = -\frac{1}{m}\boldsymbol{x}^T \mathcal{M} \boldsymbol{x}$.

4 EXPERIMENTAL PIPELINE

Only user-item interaction data is used for the experiments, without any side information. These interactions are randomly split into two sets, training and testing, with respectively 90% and 10% of the interactions. Then, recursive community detection is performed, using the training set's interaction data to build the QUBO problem. Solutions for the problem are searched with classical, quantum or hybrid samplers. Each iteration divides every community found in the previous iteration (or the starting set) into two new communities. The QPU is used only starting from the iteration in which the community have become sufficiently small to fit the hardware. For each community found, a popularity-based recommender is used to make recommendations based solely on the community's user interactions. Every item's score is computed as the number of users that interacted with it. The highest scoring items for each community are recommended to the users of that community. Recommendations are then evaluated on the testing set with both accuracy (Precision, MAP, NDCG) and beyond-accuracy (item coverage) metrics, with a cutoff of 10. We provide the source code to reproduce our experiments in a publicly available repository.[2]

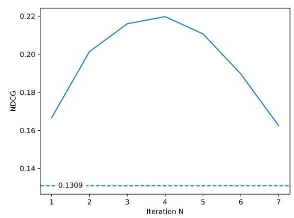
4.1 Datasets

For the experiments in this work, 8 datasets were used: MovieLens100K (943 users, 1682 items) and MovieLens1M (6040 users, 3883 items) [21]; The MovieLens (2113 users, 10109 items) and LastFM (1890 users, 18022 items) datasets from the HetRec 2011 Workshop [9]; FilmTrust (1483 users, 2071 items), another movie rating dataset [20]; Frappe (932 users, 4082 items), a real world mobile app recommendation dataset [3]; CiteULike, a scientific paper recommendation dataset, in its -a (5551 users, 16980 items) and -t (7929 users, 25975 items) versions [24]. Notice that all these datasets have interaction graphs of around 2500 up to 34000 total nodes, which makes them suitable for the current available quantum annealer devices. The cost of accessing quantum computing resources prevented us from using larger datasets.

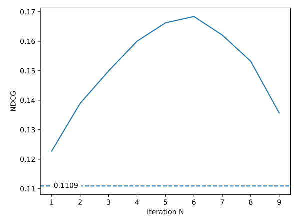
4.2 QUBO Optimizers

In order to search for solutions to a QUBO problem it is possible not only to use a quantum annealer, but also classical and hybrid solutions. In this work, five different optimizers are used: *Steepest Descent (SD)*, a greedy optimizer which iteratively performs the variable flip that causes the highest drop in solution energy; *Simulated Annealing (SA)*, a metaheuristic which stochastically performs local search, accepting worsening solution candidates in order to avoid local minima; *Tabu Search (TS)*, a metaheuristic which performs local search accepting worsening solution candidates, when no improving ones are available, but prevents visiting already explored solutions; *D-Wave Advantage QPU (QA)*, the latest available quantum annealer by D-Wave, accessible on the cloud and programmable as explained in Section 2; *D-Wave Leap Hybrid (HA)*, a quantum-classical hybrid optimizer accessible on the cloud, it performs optimization on the entire problem with classical algorithms, while decomposing it into smaller sub-problems solved with QA.

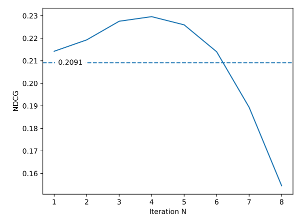
[2] https://github.com/qcpolimi/RecSys22_CommunityDetectionQuantumComputing



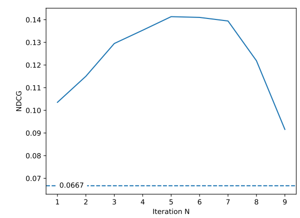
(a) MovieLens100K



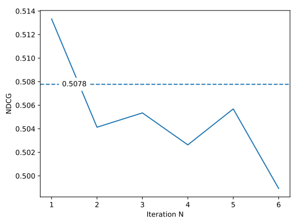
(b) MovieLens1M



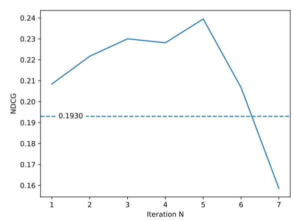
(c) MovieLens Hetrec 2011



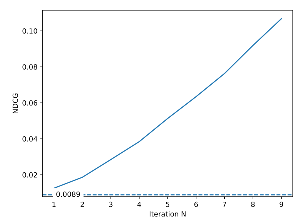
(d) LastFM Hetrec 2011



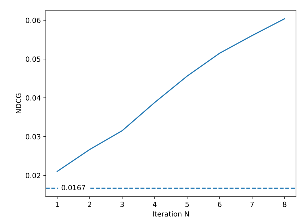
(e) FilmTrust



(f) Frappe



(g) CiteULike-a



(h) CiteULike-t

Figure 1: The figure shows for each dataset how recommendation accuracy (NDCG) at a cutoff of 10 varies when recommending with a community-informed popularity-based algorithm, w.r.t. to the baseline (dashed line), a standard popularity-based algorithm. Communities are found solving the bipartite modularity QUBO with D-Wave Leap Hybrid (or SA for the CiteULike-a and -t datasets).

5 RESULTS AND DISCUSSION

From the results, one thing in particular is made clear with the proposed experiments. As a community-informed recommender system, the popularity-based algorithm benefits from the user's subdivision in communities consistently in all datasets. Figure 1, shows how NDCG varies with the community detection iterations found with the D-Wave Leap Hybrid optimizer on all the datasets - except the CiteULike-a and -t datasets, where Simulated Annealing was used because of resource constraints - w.r.t. the baseline, which is a standard popularity-based recommender using all the users' data. It emerges clearly that community detection is always able to improve the recommendation quality, to varying extent. Most datasets exhibit a common behavior where the recommendation increases up until a certain number of communities and then starts to decrease. In other cases, see for example the CiteULike-a and -t datasets, the quality steadily improves without stabilizing, indicating that further improvements may be attainable by using even more fine-grained communities. Only in one case, FilmTrust, the recommendation quality decreases with the iterations.

Let's now focus on MovieLens1M results, shown in Table 1 for community detection with both modularity formulations solved with D-Wave Leap Hybrid and the D-Wave Advantage QPU from suitable iterations. It can be seen that the item coverage steadily increases with the iterations. This diversification in recommendations improves precision and NDCG up to around 60-70% w.r.t. the baseline, depending on modularity formulation. An interesting observation can be made about iteration 7, where communities found with bipartite modularity fail in further improving the recommendations, while communities found with weighted projection modularity obtain the best overall results on this dataset. This may be due to the smaller size of the problem needed to be solved at the same iteration for weighted projection w.r.t. to bipartite modularity. Looking at the QA optimizer, communities it found led to a comparable accuracy to the ones at the same iteration with HA, using bipartite modularity.

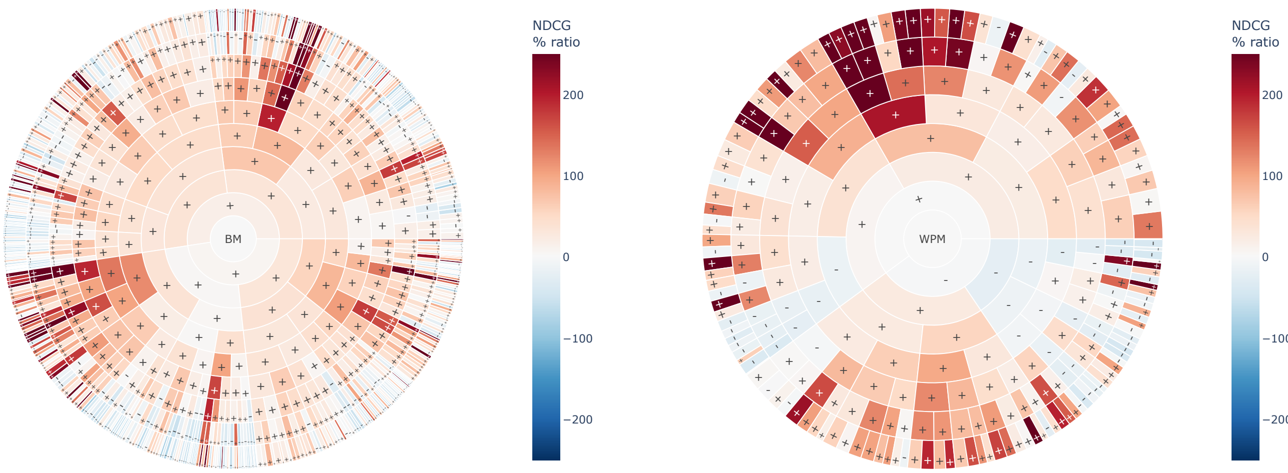
Going even more in-depth into how community detection contributes to popularity-based recommendations, Figure 2 shows how NDCG varies in each community w.r.t. the baseline. In these plots it is clearly visible how some communities are improving the recommendation accuracy, even by a lot, while others do not influence much or result in slightly worse quality. Comparing the two figures it is evident that weighted projection modularity leads to better communities w.r.t. to the bipartite one. This is also valid for MovieLens100K, MovieLens HetRec 2011 and FilmTrust, while it is not for the other datasets, where the two formulations are comparable (CiteULike-a, LastFM HetRec 2011) or worse (CiteULike-t, Frappe). Another very interesting aspect that the figures show is that some communities in intermediate iterations lead to higher accuracy when further divided, while others may lead to comparable or even worse quality. This opens up the possibility of building a recommender system not based on the communities of a specific iteration, but of a combination of communities taken from different iterations. However, this was not in the scope of this paper and is considered as future work.

6 CONCLUSIONS AND FUTURE DIRECTIONS

This work presents an exploratory study that assesses the feasibility of applying quantum annealing for community detection, in order to boost the quality of non-personalized recommendation models.

Table 1: Recommendation results on MovieLens1M of a community-informed popularity-based algorithm with communities found with both the bipartite and weighted projection modularity formulations solved with D-Wave Leap Hybrid and Quantum Annealing from suitable iterations. Accuracy metrics are computed with a cutoff at 10.

	N	C	Bipartite Modularity				Weighted Projection Modularity			
			Precision	MAP	NDCG	I Cov.	Precision	MAP	NDCG	I Cov.
Baseline	-	-	0.1072	0.0586	0.1109	0.0234	0.1072	0.0586	0.1109	0.0234
D-Wave Leap Hybrid	1	2	0.1169	0.0673	0.1228	0.0348	0.1223	0.0734	0.1295	0.0337
	2	4	0.1288	0.0753	0.1389	0.0531	0.1299	0.0775	0.1390	0.0500
	3	8	0.1401	0.0840	0.1499	0.0788	0.1471	0.0885	0.1528	0.0688
	4	16	0.1515	0.0903	0.1600	0.1154	0.1561	0.0948	0.1671	0.0917
	5	32	0.1579	0.0952	0.1662	0.1561	0.1657	0.0998	0.1772	0.1254
	6	64	**0.1603**	**0.0957**	**0.1684**	0.2004	0.1703	**0.1011**	0.1821	0.1476
	7	128	0.1548	0.0895	0.1621	0.2320	**0.1724**	0.1000	**0.1837**	0.1808
	8	256	0.1468	0.0810	0.1532	0.2599	-	-	-	-
	9	512	0.1305	0.0684	0.1357	0.2552	-	-	-	-
Quantum Annealing	7	128	0.1554	0.0895	0.1616	0.2354	-	-	-	-
	8	256	0.1469	0.0810	0.1513	0.2658	-	-	-	-
	9	512	0.1309	0.0688	0.1351	0.2627	-	-	-	-



(a) Bipartite Modularity

(b) Weighted Projection Modularity

Figure 2: The figure summarizes how the recommendation quality within a single community changes as the iterations progress. Communities are found using the D-Wave Leap Hybrid optimizer for both bipartite and weighted projection modularity on the MovieLens1M dataset. The circle at the center represents the recommendation quality measured by using a single community, i.e., the original dataset. Each successive circle around it represents an iteration and it is divided in 2^n parts each representing a community, with n the iteration number. Color represents the *percentual ratio* between the NDCG value of the community divided by the NDCG value obtained when using the full dataset to train the model. Darker red values indicate higher recommendation quality (also denoted by the + annotation), while darker blue indicate lower (also denoted by the − annotation).

The results show that leveraging user communities allows to improve both the recommendation quality and beyond-accuracy and that different communities benefit to different extent, some of them achieving substantial improvements. Furthermore, it is possible to leverage currently available quantum annealing to effectively tackle the community detection problem. Overall, these results open several research questions related to possible applications of quantum annealing to relevant problems in this field. First, studying how this finding generalizes to personalized recommendation models and if different families, i.e., matrix factorization, KNN, neural, may benefit in different ways. Second, exploring different formulations of the community detection problem to assess whether they may provide communities better suited for the recommender system domain. Third, developing more advanced strategies to fine-tune the recommendations for users of specific communities and build hybrid methods that exploit communities, for example to combine community detection with the personalization of the user interface [12, 15]. Finally, as the quantum annealing technology improves and matures, it will be possible to experiment on larger and more complex datasets to the point where the promise of improved scalability may mean community detection techniques will become feasible to apply to a wider extent.

REFERENCES

[1] Steven H. Adachi and Maxwell P. Henderson. 2015. Application of Quantum Annealing to Training of Deep Neural Networks. *CoRR* abs/1510.06356 (2015). arXiv:1510.06356 http://arxiv.org/abs/1510.06356

[2] Mohammad H. Amin, Evgeny Andriyash, Jason Rolfe, Bohdan Kulchytskyy, and Roger Melko. 2018. Quantum Boltzmann Machine. *Phys. Rev. X* 8 (May 2018), 021050. Issue 2. https://doi.org/10.1103/PhysRevX.8.021050

[3] Linas Baltrunas, Karen Church, Alexandros Karatzoglou, and Nuria Oliver. 2015. Frappe: Understanding the Usage and Perception of Mobile App Recommendations In-The-Wild. *CoRR* abs/1505.03014 (2015). arXiv:1505.03014 http://arxiv.org/abs/1505.03014

[4] Michael J. Barber. 2007. Modularity and community detection in bipartite networks. *Phys. Rev. E* 76 (Dec 2007), 066102. Issue 6. https://doi.org/10.1103/PhysRevE.76.066102

[5] Christian Bauckhage, Nico Piatkowski, Rafet Sifa, Dirk Hecker, and Stefan Wrobel. 2019. A QUBO Formulation of the k-Medoids Problem. In *Proceedings of "Lernen, Wissen, Daten, Analysen" (CEUR Workshop Proceedings, Vol. 2454)*. 54–63. http://ceur-ws.org/Vol-2454/paper_39.pdf

[6] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* 2008, 10 (oct 2008), P10008. https://doi.org/10.1088/1742-5468/2008/10/p10008

[7] Kelly Boothby, Paul Bunyk, Jack Raymond, and Aidan Roy. 2020. Next-generation topology of d-wave quantum processors. *arXiv preprint arXiv:2003.00133* (2020). https://doi.org/10.48550/ARXIV.2003.00133

[8] U. Brandes, D. Delling, M. Gaertler, R. Goerke, M. Hoefer, Z. Nikoloski, and D. Wagner. 2006. Maximizing Modularity is hard. https://doi.org/10.48550/ARXIV.PHYSICS/0608255

[9] Iván Cantador, Peter Brusilovsky, and Tsvi Kuflik. 2011. 2nd Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec 2011). In *Proceedings of the 5th ACM conference on Recommender systems* (Chicago, IL, USA) *(RecSys 2011)*. ACM, New York, NY, USA.

[10] Costantino Carugno, Maurizio Ferrari Dacrema, and Paolo Cremonesi. 2022. Evaluating the job shop scheduling problem on a D-wave quantum annealer. *Nature Scientific Reports* 12, 1 (21 Apr 2022), 6539. https://doi.org/10.1038/s41598-022-10169-0

[11] Vicky Choi. 2008. Minor-embedding in adiabatic quantum computation: I. The parameter setting problem. *Quantum Inf. Process.* 7, 5 (2008), 193–209. https://doi.org/10.1007/s11128-008-0082-9

[12] Maurizio Ferrari Dacrema, Nicolò Felicioni, and Paolo Cremonesi. 2021. Optimizing the Selection of Recommendation Carousels with Quantum Computing. In *RecSys '21: Fifteenth ACM Conference on Recommender Systems, Amsterdam, The Netherlands, 27 September 2021 - 1 October 2021*, Humberto Jesús Corona Pampín, Martha A. Larson, Martijn C. Willemsen, Joseph A. Konstan, Julian J. McAuley, Jean Garcia-Gathright, Bouke Huurnink, and Even Oldridge (Eds.). ACM, 691–696. https://doi.org/10.1145/3460231.3478853

[13] Carsten F. Dormann and Rouven Strauss. 2014. A method for detecting modules in quantitative bipartite networks. *Methods in Ecology and Evolution* 5, 1 (2014), 90–98. https://doi.org/10.1111/2041-210X.12139

[14] Hossein Fani and Ebrahim Bagheri. 2017. Community detection in social networks. *Encyclopedia with Semantic Computing and Robotic Intelligence* 01, 01 (2017), 1630001. https://doi.org/10.1142/S2425038416300019

[15] Nicolò Felicioni, Maurizio Ferrari Dacrema, and Paolo Cremonesi. 2021. A Methodology for the Offline Evaluation of Recommender Systems in a User Interface with Multiple Carousels. In *Adjunct Publication of the 29th ACM Conference on User Modeling, Adaptation and Personalization, UMAP 2021, Utrecht, The Netherlands, June 21-25, 2021*, Judith Masthoff, Eelco Herder, Nava Tintarev, and Marko Tkalcic (Eds.). ACM, 10–15. https://doi.org/10.1145/3450614.3461680

[16] Maurizio Ferrari Dacrema, Fabio Moroni, Riccardo Nembrini, Nicola Ferro, Guglielmo Faggioli, and Paolo Cremonesi. 2022. Towards Feature Selection for Ranking and Classification Exploiting Quantum Annealers. In *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*, Enrique Amigó, Pablo Castells, Julio Gonzalo, Ben Carterette, J. Shane Culpepper, and Gabriella Kazai (Eds.). ACM, 2814–2824. https://doi.org/10.1145/3477495.3531755

[17] Santo Fortunato. 2010. Community detection in graphs. *Physics Reports* 486, 3 (2010), 75–174. https://doi.org/10.1016/j.physrep.2009.11.002

[18] Fabio Gasparetti, Giuseppe Sansonetti, and Alessandro Micarelli. 2021. Community detection in social recommender systems: a survey. *Appl. Intell.* 51, 6 (2021), 3975–3995. https://doi.org/10.1007/s10489-020-01962-3

[19] M. Girvan and M. E. J. Newman. 2002. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences* 99, 12 (2002), 7821–7826. https://doi.org/10.1073/pnas.122653799 arXiv:https://www.pnas.org/doi/pdf/10.1073/pnas.122653799

[20] Guibing Guo, Jie Zhang, and Neil Yorke-Smith. 2013. A Novel Bayesian Similarity Measure for Recommender Systems. In *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, August 3-9, 2013*, Francesca Rossi (Ed.). IJCAI/AAAI, 2619–2625. http://www.aaai.org/ocs/index.php/IJCAI/IJCAI13/paper/view/6615

[21] F. Maxwell Harper and Joseph A. Konstan. 2016. The MovieLens Datasets: History and Context. *ACM Trans. Interact. Intell. Syst.* 5, 4 (2016), 19:1–19:19. https://doi.org/10.1145/2827872

[22] Mohsen Jamali and Martin Ester. 2010. A matrix factorization technique with trust propagation for recommendation in social networks. In *Proceedings of the 2010 ACM Conference on Recommender Systems, RecSys 2010, Barcelona, Spain, September 26-30, 2010*, Xavier Amatriain, Marc Torrens, Paul Resnick, and Markus Zanker (Eds.). ACM, 135–142. https://doi.org/10.1145/1864708.1864736

[23] Andrea Lancichinetti and Santo Fortunato. 2009. Community detection algorithms: A comparative analysis. *Phys. Rev. E* 80 (Nov 2009), 056117. Issue 5. https://doi.org/10.1103/PhysRevE.80.056117

[24] Xiaopeng Li and James She. 2017. Collaborative Variational Autoencoder for Recommender Systems. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017*. ACM, 305–314. https://doi.org/10.1145/3097983.3098077

[25] Fernando Benjamín Pérez Maurera, Maurizio Ferrari Dacrema, Lorenzo Saule, Mario Scriminaci, and Paolo Cremonesi. 2020. ContentWise Impressions: An Industrial Dataset with Impressions Included. In *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*, Mathieu d'Aquin, Stefan Dietze, Claudia Hauff, Edward Curry, and Philippe Cudré-Mauroux (Eds.). ACM, 3093–3100. https://doi.org/10.1145/3340531.3412774

[26] Ashley Montanaro. 2016. Quantum algorithms: an overview. *npj Quantum Information* 2, 1 (2016), 1–8. https://doi.org/10.1038/npjqi.2015.23

[27] Rajdeep Kumar Nath, Himanshu Thapliyal, and Travis S. Humble. 2021. A Review of Machine Learning Classification Using Quantum Annealing for Real-World Applications. *SN Comput. Sci.* 2, 5 (2021), 365. https://doi.org/10.1007/s42979-021-00751-0

[28] Christian F. A. Negre, Hayato Ushijima-Mwesigwa, and Susan M. Mniszewski. 2020. Detecting multiple communities using quantum annealing on the D-Wave system. *PLOS ONE* 15, 2 (feb 2020), e0227538. https://doi.org/10.1371/journal.pone.0227538

[29] Riccardo Nembrini, Maurizio Ferrari Dacrema, and Paolo Cremonesi. 2021. Feature Selection for Recommender Systems with Quantum Computing. *Entropy* 23, 8 (2021), 970. https://doi.org/10.3390/e23080970

[30] M. E. J. Newman. 2006. Finding community structure in networks using the eigenvectors of matrices. *Phys. Rev. E* 74 (Sep 2006), 036104. Issue 3. https://doi.org/10.1103/PhysRevE.74.036104

[31] M. E. J. Newman and M. Girvan. 2004. Finding and evaluating community structure in networks. *Phys. Rev. E* 69 (Feb 2004), 026113. Issue 2. https://doi.org/10.1103/PhysRevE.69.026113

[32] Massimo Quadrana, Alexandros Karatzoglou, Balázs Hidasi, and Paolo Cremonesi. 2017. Personalizing Session-based Recommendations with Hierarchical Recurrent Neural Networks. In *Proceedings of the Eleventh ACM Conference on Recommender Systems, RecSys 2017, Como, Italy, August 27-31, 2017*, Paolo Cremonesi, Francesco Ricci, Shlomo Berkovsky, and Alexander Tuzhilin (Eds.). ACM, 130–137. https://doi.org/10.1145/3109859.3109896

[33] P. Krishna Reddy, Masaru Kitsuregawa, P. Sreekanth, and S. Srinivasa Rao. 2002. A Graph Based Approach to Extract a Neighborhood Customer Community for Collaborative Filtering. In *Databases in Networked Information Systems, Second International Workshop, DNIS 2002, Aizu, Japan, December 16-18, 2002, Proceedings (Lecture Notes in Computer Science, Vol. 2544)*, Subhash Bhalla (Ed.). Springer, 188–200. https://doi.org/10.1007/3-540-36233-9_15

[34] S. Redner. 1998. How popular is your paper? An empirical study of the citation distribution. *The European Physical Journal B - Condensed Matter and Complex Systems* 4, 2 (01 Jul 1998), 131–134. https://doi.org/10.1007/s100510050359

[35] Le Hoang Son. 2016. Dealing with the new user cold-start problem in recommender systems: A comparative review. *Information Systems* 58 (2016), 87–104. https://doi.org/10.1016/j.is.2014.10.001

[36] V. A. Traag, L. Waltman, and N. J. van Eck. 2019. From Louvain to Leiden: guaranteeing well-connected communities. *Scientific Reports* 9, 1 (26 Mar 2019), 5233. https://doi.org/10.1038/s41598-019-41695-z

[37] Lyle H. Ungar and Dean P. Foster. 1998. Clustering Methods for Collaborative Filtering. In *AAAI 1998*.

[38] Hayato Ushijima-Mwesigwa, Christian F. A. Negre, and Susan M. Mniszewski. 2017. Graph Partitioning using Quantum Annealing on the D-Wave System. *In Proceedings of the Second International Workshop on Post Moores Era Supercomputing (PMES '17)* abs/1705.03082 (2017). arXiv:1705.03082 http://arxiv.org/abs/1705.03082

[39] Stanley Wasserman and Katherine Faust. 1994. *Social Network Analysis: Methods and Applications*. Cambridge University Press. https://doi.org/10.1017/CBO9780511815478

[40] Dennis Willsch, Madita Willsch, Hans De Raedt, and Kristel Michielsen. 2020. Support vector machines on the D-Wave quantum annealer. *Comput. Phys. Commun.* 248 (2020), 107006. https://doi.org/10.1016/j.cpc.2019.107006

[41] Lei Zheng, Vahid Noroozi, and Philip S. Yu. 2017. Joint Deep Modeling of Users and Items Using Reviews for Recommendation. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, WSDM 2017, Cambridge, United Kingdom, February 6-10, 2017*, Maarten de Rijke, Milad Shokouhi, Andrew Tomkins, and Min Zhang (Eds.). ACM, 425–434. https://doi.org/10.1145/3018661.3018665

[42] Jason Zhu, Kyle Brubaker, and Martin Schuetz. 2021. *Community Detection using Hybrid Quantum Annealing on Amazon Braket – Part 1.* Retrieved July 29, 2022 from https://aws.amazon.com/blogs/quantum-computing/community-detection-in-complex-networks-using-hybrid-quantum-annealing-on-amazon-braket-part-i/

[43] Jason Zhu, Kyle Brubaker, and Martin Schuetz. 2022. *Community Detection using Hybrid Quantum Annealing on Amazon Braket – Part 2.* Retrieved July 29, 2022 from https://aws.amazon.com/blogs/quantum-computing/community-detection-using-hybrid-quantum-annealing-on-amazon-braket-part-2/