NATIONAL RESEARCH UNIVERSITY
HIGHER SCHOOL OF ECONOMICS

as a manuscript

Kiselev Dmitrii Andreevich

Graph-based recommender systems using network embeddings

PhD Dissertation Summary
for the purpose of obtaining academic degree
Doctor of Philosophy in Computer Science

Moscow – 2022

The PhD Dissertation was prepared at National Research University Higher School of Economics.

Academic Supervisor: Ilya Makarov, PhD, Senior Research Fellow, Artificial Intelligence Research Institute, Moscow, Russia; Assistant Professor, Faculty of Computer Science, Senior Research Fellow, Laboratory of Algorithms and Technologies for Networks Analysis, National Research University Higher School of Economics.

1 Dissertation topic

Nowadays, recommender systems are one of the key components of different consumer services from e-commerce to social media [2, 10]. It helps to navigate through the large volume of items empowering the user experience. The methods in the fields vary from classic matrix completion techniques to modern ones inspired by NLP sequence models. One of the prominent approaches is to consider the recommender systems as a link prediction problem on bipartite user-item interaction graphs. The graph machine learning and network embedding field recently emerged but still requires adaptation to efficiently solve recommender systems tasks.

2 Object of the research

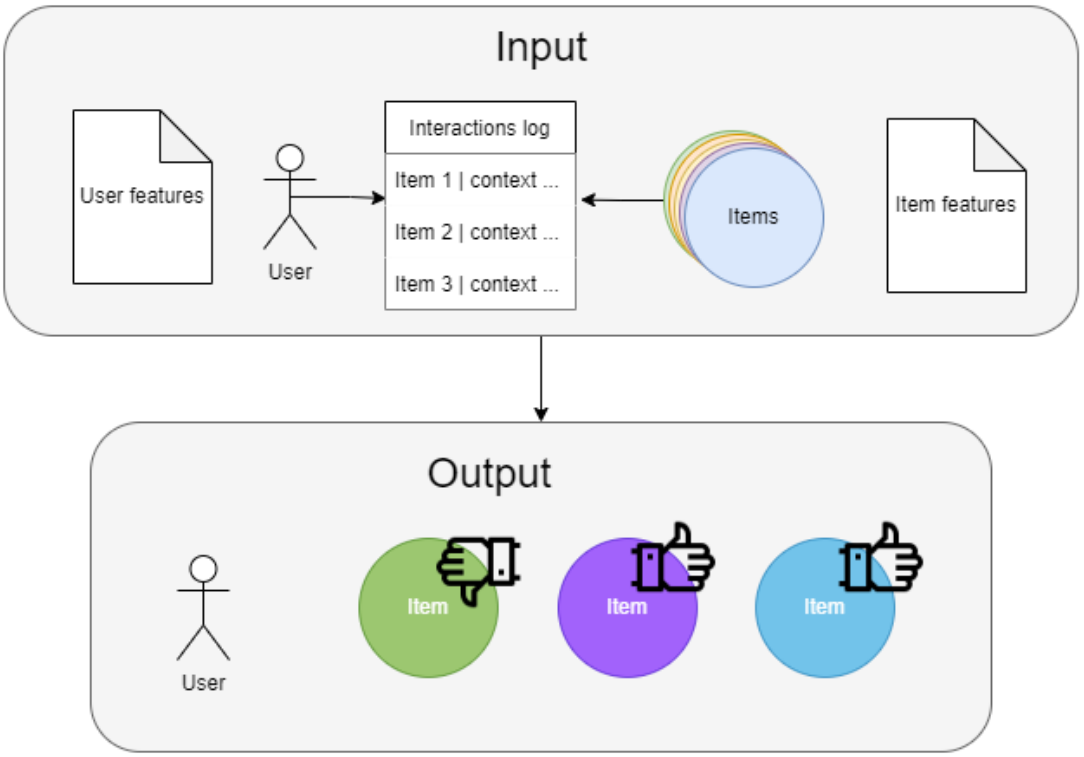

Figure 1: Generic scheme of recommendation models

The object of the dissertation is recommender systems. Figure 1 describes the generic scheme of a recommender system problem. The model receives a set of users, items, their features and interaction log as input. The output of a model

is an item or slate of the most relevant items for a given user that optimize user engagement (e.g. likes, time spent in the app) or business effect (e.g. expected income from item impressions). In section 3.2 we explain classic recommender approaches and corresponding problems in detail.

3 Subject of the research

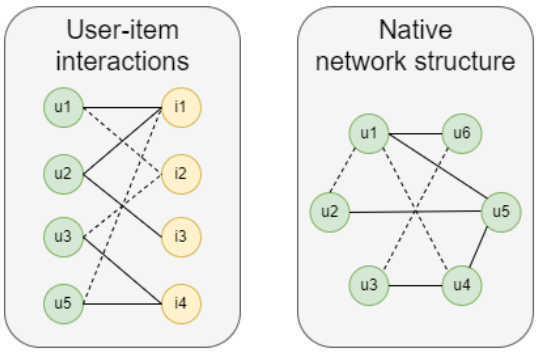

Figure 2: Representations of user-item interactions as graphs for recommender system problem

Recently, a lot of works propose to consider the recommendation problem as a problem on graphs [50]. They replace the user-item interaction matrix as a (bipartite) graph with heterogeneous edges (Figure 2). Such a view allows the fusion of the content and collaborative (interaction) approaches. Moreover, the graph representation of the task allows finding similar users or items just by taking the two-hop neighborhood of the user. So preservation of graph topology can play an important role. By taking the graph into account matrix completion task can be transformed into the future link prediction problem (on Figure 2 our goal is to define whether dotted edges will exist or not). This problem is usually formulated as a well-studied machine learning problem: ranking or binary classification of possible (non-presented) edges in the interaction graph.

Figure 3 presents the link prediction pipeline. Firstly, we take a user-item interaction graph as input. Then, the network structure and other properties are encoded in continuous representation (more in section 3.3). There are three main types of network embedding techniques: matrix factorization, random walk based methods and graph neural networks (GNN). The first group of methods applies dimensionality reduction techniques to some of the graph matrix representations (e.g. adjacency, Laplacian). The second group of methods samples random walks and applies Skip-gram [23] to optimize node co-occurrence in the walks. The

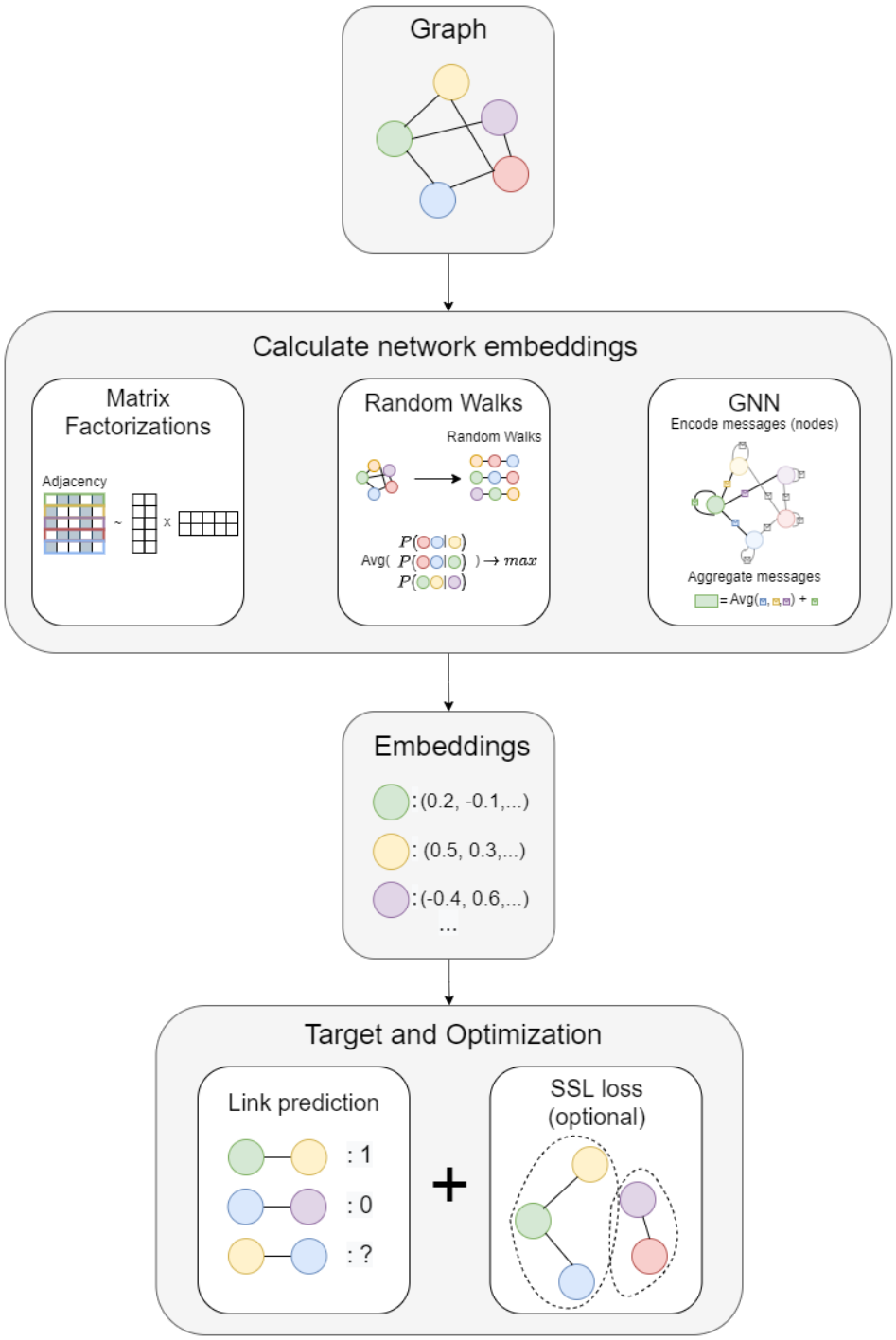

Figure 3: The pipeline of link prediction for recommender systems

last group of methods can be generalized by the Message-Passing framework. It consists of two steps: place non-linear node features transformation on the edge (encode message) and aggregate all messages for a given node including itself. Finally, when network embeddings are obtained, we can solve the link prediction problem with standard table data classification methods. Additionally, we can apply self-supervised learning to preserve specific graph properties or to pretrain end-to-end models.

In the dissertation, we have studied different facets of network embedding techniques, their properties and their effect on downstream link prediction problems. In the section 3.3 objectives and tasks in the research are formulated more precisely.

3.1 PhD Dissertation Relevance

User behavior has changed dramatically over the last 5-10 years. The speed of changes in behavioral trends accelerated. So, it is hard to apply the classic methods to gain impact on client and business value [45, 31]. Therefore, new methods are required to solve the aforementioned behavioral dynamics issues. Graph-based recommender systems allow integration of the best part of different methods to make recommenders more accurate.

3.2 Problems of recommender systems

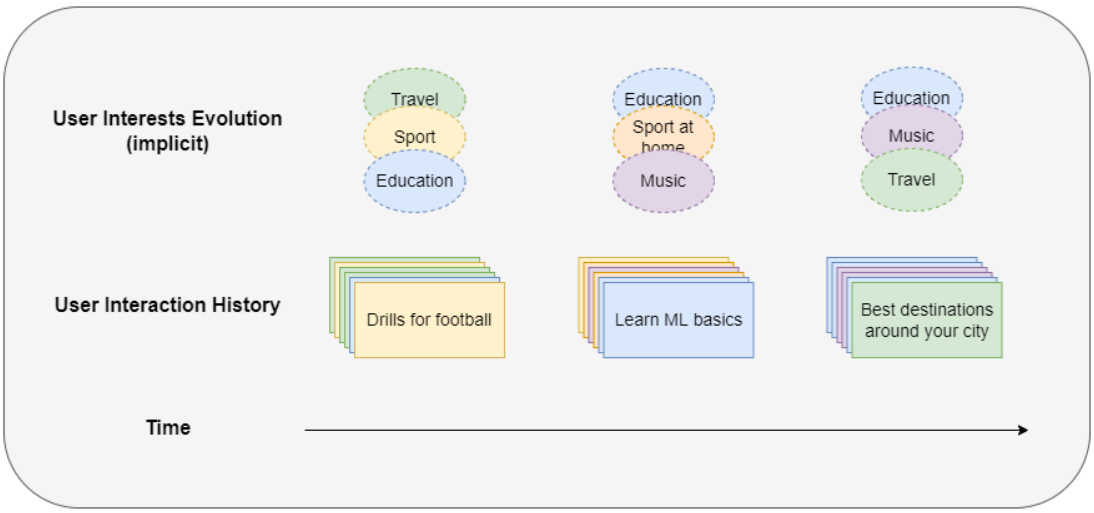

Figure 4: Common theoretical problems of graph-based recommender systems development caused by environment dynamics

Figure 4 describes the problem of recommendation in the dynamic environment. The set of users, items, and their interaction histories are provided as input. We aim to select the items that will engage the user in the future. Let us suppose that we have three time periods in observations: (1) before the COVID-19 pandemic, (2) lockdown, and (3) slight relaxation of restrictions. Implicit user preferences are changing over time due to shifts in the environment and user internal evolution. This example helps us to show the possible problems caused by environmental dynamics. Firstly, we have new content without any previous interactions (cold start) [38]. Moreover, obtained historical data are biased towards previous behavior (feedback loop) [21] and current model performance degrades due to data distribution shifts [5]. Finally, the data has a temporal structure [45].

There are plenty of methods to handle mentioned issues. However, they usually concentrate only on a specific problem and are susceptible to others. The most popular and powerful models for recommender systems are matrix factorization techniques [42]. They are aimed to learn the latent representation of users and items to predict the expected score of its interaction. However, matrix factorization is a static model and is restricted to the already-known set of users and items. Moreover, they can not handle the features of users and items. So, such models are susceptible to the aforementioned issues. The factorization machines [34] generalize the matrix factorization by adding the user and item descriptions. Thus, they can predict interactions between new users and items. However, these predictions do not account for any interactions before the full model update. Other content models [32] usually do not consider user-item interactions at all focusing on the internal content representations of items and users. Thus, factorization machines and content models are not capable to solve the online adaptation issue. Nevertheless, there are a couple of tricks to introduce temporality there. For example, one can find most similar items only to the last watched by user item. More sophisticated models that handle recent changes in user behavior are sequential recommender system models[45]. It looks at the user as on a sequence of viewed items. Such vision allows the application of different generative sequence models from natural language processing like LSTM [55] or Transformers [39]. To embed the item they can use its internal content embedding. So, they are capable of partially solving the cold-start problem. However, they have problems handling new users due to omitting their entity. Thereby, different types of recommender models focus on different issues but do not solve them at once. So, a new view of the problem is required.

3.3 Tasks and objectives of the research

Figure 5 describes the main objectives of the work. They are required for building an efficient graph-based recommender system that is capable to solve issues

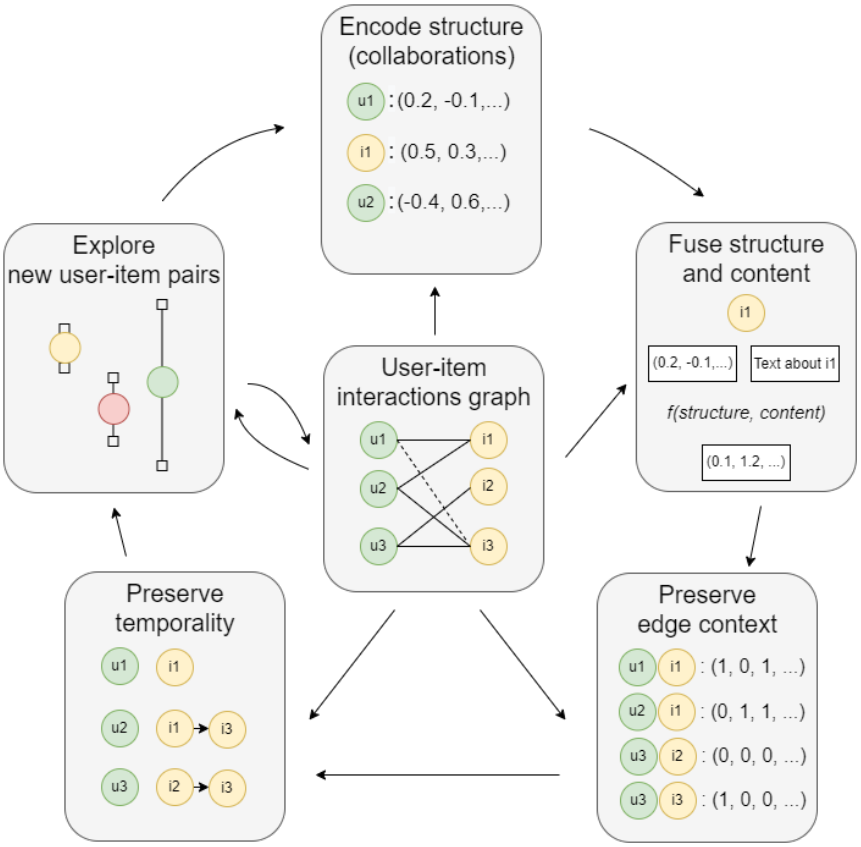

Figure 5: Structure of our research. We increase the setting scale and complexity by developing novel graph-based methods in recommender systems. We start with a static setting of user-item interactions and apply graph embedding techniques to develop a simple collaborative recommender system (a). Then we focus on taking into account node-based (b) and edge-based features (c) to build a hybrid recommender system capable of solving the cold-start problem. Finally, we consider dynamics component (d) and exploration techniques (e) to enable the online-adaptation ability of the underlined models. Each component contributes to the system and empowers the next step, thus providing a strong baseline for graph-based recommendations for solving cold-start, temporality, data distribution shift and feedback loop problems in a novel and efficient way.

caused by environment dynamics: cold start, awareness of temporal structure, data distribution shifts and feedback loops.

As mentioned before, the recommendation problem is equivalent to the link prediction problem which is the binary classification of edge existence. It requires the continuous representation of discrete graph substructures (nodes and edges).

So, **the first objective of the research** is to analyze existent network embedding techniques. However, we can not use only a graph structure due to the cold-start problem. So, **the next two steps** are handling of nodes and edges features and its efficient fusion with collaborative info. **The fourth objective** is to account for the temporal structure of the user-item interaction graph, we aim to handle user interests' evolution and causality in their actions. It is required to add exploration in the models to solve data distribution shifts and feedback loop problems. So, the **the last objective** is to propose graph-based exploration techniques. After this step model can provide an efficient recommendation to the user, receive their feedback and update the user-item interaction graph and node embeddings.

Thus, we postulate the five objectives of the research. We consider these problems independently and provide a chapter for each one. We gradually complicate proposed models to develop a final recommender system that covers all given objectives. One can reformulate the research tasks more precisely as follows:

1. Compare the network embedding techniques to find the best match to specific graph types and attribute representations under a unified framework
2. Propose an efficient strategy to incorporate **node** features with structural information to solve the link prediction (recommendation) problem
3. Propose an efficient strategy to incorporate **edge** features with structural information to solve the link prediction (recommendation) problem
4. Propose the method to efficiently preserve the temporality of networks
5. Propose the method to explore the possible user-item interactions to handle data distribution shifts and feedback loop problems

The main goal of the research is to create new graph methods and show that they are the best fit to solve all the problems caused by environment dynamics and intense changes in user behavior.

4 Key results and conclusions

This section describes the main contributions achieved by the present work, its novelty, theoretical and practical significance, research methodology and the reliability of the results.

Key aspects/ideas to be defended:

1. Detailed taxonomy of the network embedding techniques and their applications [17]
2. Benchmarking of the state-of-the-art graph embedding techniques in the link prediction problem [17]
3. The methodology to automatically extract features in the text-attributed graphs for link prediction and node classification [19]
4. Novel self-supervised joint network node and edge embedding based on consistent Line graph representation [18]
5. Novel temporal network embedding achieving state-of-the-art results in various temporal graph machine learning tasks [20]
6. Standardized temporal network embedding evaluation framework and comparison of state-of-the-art models under common training setting, providing new insights and clarification of real-world performance compared to reported in the original research articles [20]
7. Novel Personalized PageRank [26] based exploration strategy [16]
8. Small-world [49] based exploration strategy [16]
9. The methodology how to apply proposed exploration methods to recommender system with online adaptation in dynamic environments [16]

Scientific novelty. The dissertation concentrates on the recently emerged, poorly studied problem of graph-based recommender systems using automatic feature extraction. The work studies five main components required for successful implementation of graph-based recommender: user-item interaction graph structure encoding, an efficient fusion of structural (collaborative) and node attributes (user and item content), expressive context-aware edge encoding, temporal properties encoding and exploration of poorly known user-item pairs. We study each objective separately in different papers. Each paper conducts a review, comparison and

analysis of the existent methods and provides strategies for optimal applications. For the last three tasks, we proposed novel models and methods that improve the performance of downstream link prediction and recommendation problems.

Theoretical and practical significance. The first two contributions conclude the current progress in the field and propose new directions and views on further development. Also, it provides pieces of advice for the practical use of network embedding techniques. The last three contributions provide novel models that improve performance on practical datasets. Also, the fourth paper presents the standardized evaluation framework that allows the acceleration of research in the temporal graph embedding and graph-based recommender system fields.

The methodology of the research. The study is based on the theory of geometric deep learning, graph neural networks, network science, recommender systems, classic machine learning and statistics.

The reliability of the results. It is provided by complex and exhaustive experiments including calculations of the confidence intervals of the metrics on top with the comparison to the other state-of-the-art methods.

Funding. The research was supported by the Faculty of Computer Science, HSE University; Russian Science Foundation; HSE University Basic Research Program; computational resources of HPC facilities at HSE University.

5 Publications and approbation of research

5.1 First-tier publications

1. Ilya Makarov, Dmitrii Kiselev, Nikita Nikitinsky, and Lovro Subelj. Survey on graph embeddings and their applications to machine learning problems on graphs. *PeerJ Computer Science*, 7:e357, 2021 [17]

2. Ilya Makarov, Mikhail Makarov, and Dmitrii Kiselev. Fusion of text and graph information for machine learning problems on networks. *PeerJ Computer Science*, 7, 2021 [19]

3. Ilya Makarov, Ksenia Korovina, and Dmitrii Kiselev. JONNEE: Joint network nodes and edges embedding. *IEEE Access*, 2021 [18]

4. Ilya Makarov, Andrey Savchenko, Arseny Korovko, Leonid Sherstyuk, Nikita Severin, Dmitrii Kiselev, Aleksandr Mikheev, and Dmitrii Babaev. Temporal network embedding framework with causal anonymous walks representations. *PeerJ Computer Science*, 8:e858, 2022 [20]

5. Dmitrii Kiselev and Ilya Makarov. Exploration in sequential recommender systems via graph representations (on review). *IEEE Access*, 2022 [16]

5.2 Other publications

Dmitrii Kiselev and Ilya Makarov. Prediction of new itinerary markets for airlines via network embedding. In *International Conference on Analysis of Images, Social Networks and Texts*, pages 315–325. Springer, 2019 [15]

5.3 Reports at workshops and conferences

1. Internal seminar of the Sber Recommender Systems Platform, 1 June 2022. Topic: "Exploration in interactive recommender systems via graph representations"

2. 12th International Conference on Network Analysis (NET) 2022. 25 May 2022. Topic: "Predicting Molecule Toxicity via Graph Neural Networks"

3. Journal Club of the Artificial Intelligence Research Institute (AIRI), Moscow, Russia, 27 October 2021. Topic: "Temporal graph embeddings".

4. Internal seminar of the Sber Recommender Systems Platform, 26 October 2021. Topic: "Review of the exploration in recommender systems"

5. Internal seminar of the Sber Recommender Systems Platform, 8 July 2021. Topic: "Graph-based recommender systems"

6. Artificial Intelligence Journey, 20 December 2020. Topic: "Eternal Student: How to check whether adaptive recommender system learns"

7. 2nd International Workshop "New Level of Visualization - New Level of Analytics" (NLVNLA-2020), 24 January 2020. Topic: "Visualizing Structural Data via Network Embeddings"

8. International Conference on Analysis of Images, Social Networks and Texts (AIST-2019), 19 July 2019. Topic: "Prediction of new itinerary markets for airlines via network embedding"

9. 9th International Conference on Network Analysis (NET) 2019. 18 May 2022. Topic: "Prediction of new itinerary markets for airlines via network embedding"

5.4 Research approbation

The model proposed in the [20] was evaluated by Sber AI Lab in the bank downstream task. It was applied to the transaction graph between small and medium enterprises (SMEs) to receive the dynamic SMEs representations. After, these vectors were used to train the LightGBM [11] classifier to define whether the SME would be bankrupted or not after six months. The proposed feature-extraction approach performed competitively with existing bank sequential methods and significantly outperformed other graph-based methods on a subsample of the transactions graph.

Personal contribution to the papers. In [17] supervisor Ilya Makarov and Lovro Subelj stated the problem, revised the manuscript and designed experiments and overviewed the main part of the methods, author of the thesis conducted the experiments, compared models, overviewed the applications and partially overviewed the methods, and derived the open problems and conclusions, Nikita Nikitinskiy also overviewed the applications to NLP problems and revised the paper. In [19] supervisor Ilya Makarov stated the problem, revised the manuscript and designed experiments, the author of the thesis conducted experiments with complex fusion methods like GCN, GraphSAGE, GAT, and GIC, derived the recommendations for practical use, and Mikhail Makarov conducted the rest part of the experiments on the structural methods. In [18] supervisor Ilya Makarov stated the problem, Ksenia Korovina developed the main part of the pipeline, the author of the thesis revised it and enhanced the semi-supervised

training strategy for the JONNEE model, conducted the experiments and compared models. In [20] supervisor Ilya Makarov, Andrey Savchenko and Dmitrii Babaev stated the problem, revised the manuscript and designed experiments, Arseny Korovko and Leonid Sherstyuk implemented the core part of the framework and realized novel model elements, Nikita Severin and author of the thesis rewrote it to production-ready state, implemented state-of-the-art methods within it and conducted more experiments, author of the thesis conducted the final experiments, selected the best variation of model components and tuned it to achieve the state-of-the-art results, revised the paper. Aleksandr Mikheev conducted the experiments on internal Sber data. The paper [16] was fully developed by the author of the thesis with minor revision, consultations and discussion of experiments and model design with Ilya Makarov.

The author of the dissertation is a corresponding author in papers [17, 19, 18].

6 Contents

The section describes the general idea, results and novelty of each chapter in the thesis. The name of the subsection corresponds to the name of the chapter (paper) in the thesis.

Volume and structure of the work. The thesis contains an introductory chapter, concluding chapter and content of five papers. The full volume of the thesis with appendix is **160** pages.

6.1 Survey on graph embeddings and their applications to machine learning problems on graphs

The first chapter of the thesis focus on the study and evaluation of existing approaches to graph embedding procedures. The work overview different types of graphs including user-item interaction graphs, and their application to the recommender systems. The main goal of the paper is to analyze and compare existent models to encode graphs and estimate their efficiency in downstream tasks including link prediction.

Methods to learn graph representations differs. The classic approach is to apply matrix factorization to different matrix views of the graph like graph Laplacian, multi-hop transition matrices and so on [4, 25, 1, 46, 40, 12]. In that way, graph representation learning is similar to recommender systems. Recent methods exploit the fact that the distribution of node occurrence in random walks is proportional to its degree thereby and power law for real-world graphs [29]. The distribution of the word occurrence in the natural language conforms to Zipf law which is also similar to the power law [29]. So, the skip-gram approach can be applied to efficiently learn node embeddings [23]. The methods in these groups usually differ in the random walk sampling strategies that are designed to preserve specific graph properties [29, 7, 36, 30]. Finally, the most modern graph embedding techniques are provided by geometric deep learning. The Graph Convolution Network (GCN) represents a node as a sequence of non-linear node features transformations and aggregations over the neighborhood [14]. Such an approach approximates the spectral decomposition preserving a lot of important graph properties. Graph Attention Networks imply a similar idea but utilize an attention mechanism to learn better aggregations over node neighborhood [43]. Message-passing framework generalizes the Graph Neural Networks scheme decomposing process of node encoding in two main steps: message encoding and neighborhood aggregation [6]. Message encoding is the encoding of node features and placing them on the adjacent edge. The aggregation step applies pooling to calculate the final representation of the node.

Such a framework allows fast batch calculations of Graph Neural Network prediction. GraphSAGE introduces the idea of sampling strategies to node neighborhood to efficiently scale Graph Neural Networks on large graphs [8]. Generative graph models follow the ideas of generative models from computer vision and natural language processing: variational autoencoders, recurrent models and generative adversarial networks. Moreover, such methods are frequently applied with other self-supervised approaches to build more robust and generalizable models.

On the other hand, network embedding techniques can be categorized from the point of different graph types and properties. To handle attributed graphs one can apply methods of three general groups: concatenation of graph matrix representation and feature matrix with further dimensionality reduction, initial graph structure enhancement with edges from content-similarity graph and graph neural networks. Heterogeneous graphs are usually handled by the separation of embedding tasks for different types of nodes and edges. Random walk based methods reformulate the optimization problem to optimize the similarity between nodes of different types. GNNs for heterogeneous graphs apply the attention mechanism in an intra- and inter-group fashion. The methods of dynamic graphs vary due to different representations of such graphs: sequence of snapshots or sequence of individual node and edge events. In the first case, one can separately learn network embedding for each snapshot and infer the dynamics using recurrent models. In the second one, recurrent models are applied personally to each node memory and network aggregation is performed after the memory update. To handle large graphs methods of two types were proposed: different mini-batch sampling techniques to iterate over large graphs and graph coarsening to split it into smaller parts and parallel computations.

The difference between different graph machine learning problems lies in the additional domain-specific targets. For instance, classification models can be enhanced with metric losses to tighten the vectors of nodes from one class. In clustering, it is popular to add modularity-based penalty after k-means or dbscan clustering. Subgraph and graph embeddings add node and edge pooling techniques.

As a result, we have derived several open problems. Firstly, dynamic graph embeddings were poorly studied. Secondly, most of the models focus only on the node representations omitting the edges and their features. Most of the models are susceptible to poor scalability. However, methods for large graphs introduce a lot of biases and have only a little theoretical justification for them. Finally, the most important problem is the absence of generic graph embedding models and strategies to select optimal encoding for a specific graph with specific properties.

Also, we conduct the experiment study of properties for network embedding models of different types. The experiments aim to understand how specific network

properties are aligned with the performance of the state-of-the-art models. Such models are also examined on the different generative graph models to provide a better understanding of network embeddings in a controllable environment.

The novelty of the paper is in the methodology to compare and evaluate graph embedding methods for a specific graph-related machine learning problem. We found that the classic structural embedding methods with the proper objective on specific network properties provide competitive quality with complex deep learning models in the downstream tasks.

As a result, we studied and compared different methods to handle graph structure. This step is essential to build graph-based recommender systems with collaborative knowledge.

The paper was published in the PeerJ Computer Science Journal. It is indexed as Q1 (2020) and Q2 (2021) in Scopus.

6.2 Fusion of text and graph information for machine learning problems on networks

The chapter solves the second task: fusion of content and collaborative information for node classification and link predictions (recommendation) problems.

Previously, the studies in the network embedding fields were focused only on the structural part of the embedding and aims to preserve the structural information better. Due to this problem, the papers on the graph neural networks and other embedding techniques considered only a simple text encoding strategy like Bag-of-Words [9] or Tf-Idf [37]. However, the more important problem raised in the previous chapter is how to properly balance the trade-off between information from the node attributes and the structural network properties. The current chapter is aimed to propose such strategies to efficiently handle text attributes within the framework of graph machine learning problems. So, we study the advanced methods for text encoding: LDA [3], word2vec [24], sent2vec [27], Sentence BERTS [33] and ERNIE [41]. Firstly, we understand whether the text information is enough to solve the node classification and link prediction problems. Next, we analyze how this information can be integrated within the graph embedding framework. Different strategies are established: naive fusion (concatenation) of structural [25, 29, 7] and text embeddings, complex matrix factorization techniques [51, 28] and graph neural networks [13, 8, 44, 22]. Moreover, the graph neural networks were also trained and validated in an end-to-end fashion to provide the best performance on the downstream problems.

The main conclusions of the work are the following. Firstly, we found that complex text encoding strategies allow significantly boost the performance of the structural models in link prediction and node classification problems. Secondly,

we show that SBERT text encoding benefits more the link prediction problem and Sent2Vec the node classification. Finally, we propose the modification of GCN architecture.

The novelty of the study lies in the new view on the graph machine learning problems and accounting for the complex text encoding strategies that allow boosting the performance in practical applications.

As a result, we have shown that for a successful solution of the link prediction (recommendation) problem only structural information is not enough. The model should consider content features too. Moreover, we analyzed different fusion techniques and showed the efficiency of modern text encoding models. Such graphs are one of the most popular in real-world recommender systems, e.g. social networks, media, descriptions of consumer goods on marketplaces and so on.

The paper was published in the PeerJ Computer Science Journal. It is indexed as Q1 (2020) and Q2 (2021) in Scopus.

6.3 JONNEE: Joint network nodes and edges embedding

The current chapter focuses on the problem of edge attribute awareness in graph neural networks. Previously, most papers worked with the node attributes only or with simple edge-weighted graphs. The goal of the model is to provide the framework within which such attributes could be efficiently employed.

The general idea of the model (Figure 6 is to utilize well-studied node embedding techniques to edges. To do it, we apply the VGAE to a Line graph where the original edges play the role of the nodes. Thus, applying similar models to other graph representations we can build fine-grained edge embeddings aware of edge context. However, both models should agree on their representations of nodes and edges. So, we propose a novel self-supervised training procedure. We propose a joint loss function based on the following components

1. Reconstruction loss of autoencoders for both graphs

$$L_G = \|A - \widehat{A}\|_F^2 \sim \frac{1}{|V|^2} \sum_{i,j} (\hat{a}_{ij} - a_{ij})^2 \tag{1}$$

$$L_{G^*} = \|A^* - \widehat{A^*}\|_F^2 \sim \frac{1}{|E|^2} \sum_{i,j} (\hat{a}^*_{ij} - a^*_{ij})^2 \tag{2}$$

2. Joint loss function (f and $f*$ are node embeddings for graph and its Line correspondingly, $N(v)$ is a set of node neighbors)

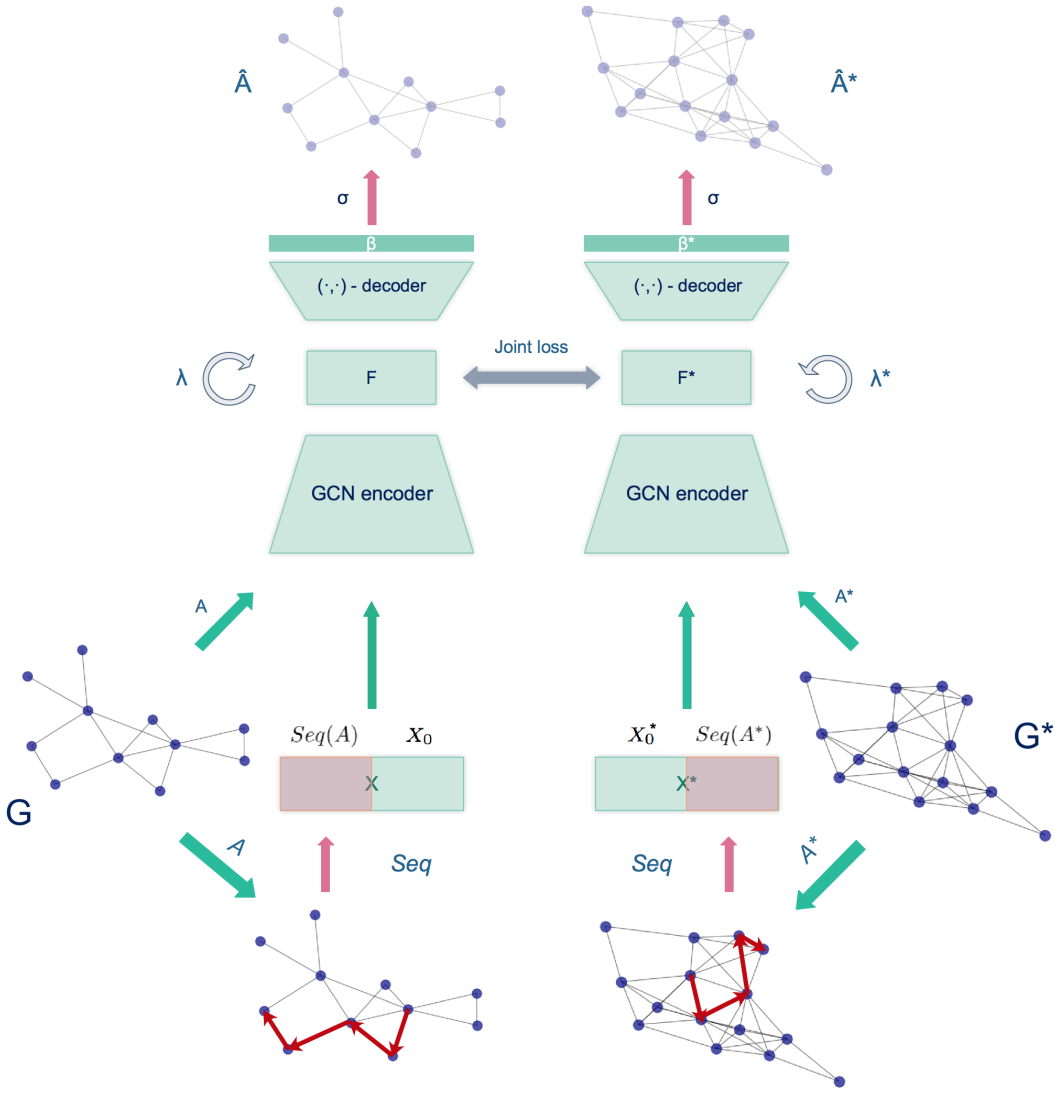

Figure 6: JONNEE architecture

$$L_{G^* \to G} = \sum_{v \in V} \| f(v) - \frac{1}{|N^G(v)|} \sum_{u \in N(v)} f^*((u,v)) \|_2^2 \tag{3}$$

$$L_{G \to G^*} = \sum_{e=(u,v) \in E} \| f^*(e) - \frac{\sum_{t \in N^G(u) \cup N^G(v)} f(t)}{|N^{G^*}(e)|} \|_2^2 \tag{4}$$

3. L_2-regularization for both graphs

The most important part of the loss is the equations 3 and 4. The main idea is to anchor representations of both encoders to each other. Such loss tightens the original graph node embeddings to the mean of corresponding edge vectors in the Line graph and vice-versa. Formulae 1 and 2 are required to train autoencoders themselves. Finally, we apply regularization for embeddings to reduce the model complexity and over-fitting.

Despite the better edge embeddings, the model is more computationally complex due to a large sparse Line graph. To reduce the complexity and fasten convergence we initialize models with random walk based embeddings: node2vec or diff2vec.

The proposed framework performs competitively with the state-of-the-art semi-supervised deep learning models in the link prediction and node classification tasks. The performance of the model is high in both unsupervised and semi-supervised strategies. It provides well-clustered visual representations of the network.

The novelty of the JONNEE is twofold. Firstly, it enhances the classic variational graph autoencoders with the node sequence-encoding strategies. Secondly, it is trained by the novel self-supervised method that tightens node representations with its dual-edge representations of a Line graph.

The JONNEE is an important step to build context-aware recommender systems. Firstly, it improves the edge vectors for the link prediction problem. Secondly, it allows for preserving the decision context of the user. Decision context is essential in a lot of cases, e.g. music recommendations where the user consumes different playlists based on the current situation (work, party).

The paper was published in IEEE Access. It is indexed as Q1 in Scopus and Web of Science.

6.4 Temporal network embedding framework with causal anonymous walks representations

The chapter studies the problem of graph dynamics. One of the main focuses lies in the dynamic heterogeneous interaction networks that are the generalization of the user-item interaction graphs from recommender systems.

In this chapter, we describe a novel network embedding that combines the best elements of the efficient Temporal Graph Network embedding (TGN) ([35]) and Causal Anonymous Walks (CAW) ([48]). We choose the TGN as a backbone because it generalizes most existing temporal graph embedding models via flexible modular architecture. It allows updating node memory in a fast and expressive manner. Also, more modern models like APAN ([47]) or HiLi ([47]) follow a similar paradigm of passing messages through the memory module. The CAW provides an opposite view on the graph evolution problem. It rejects the idea of memory

and anonymizes each node. Instead, it aimed to build such a model, which can implicitly exploit laws of specific graph evolution ignoring node identities. As a result, CAW is unable to create node embedding but can significantly impact model quality by taking into account changes in the graph structure. The fusion of two opposite ideas allows to build more precise network encoding methods. To include CAW features into TGN we concatenate its representation before adding it to the memory and before the final embedding layer.

The second important contribution of the chapter is a standardized framework for the training and validation of temporal network embeddings. It provides standardized wrappers and abstractions to prepare graphs, select batching types and parameters, split data for transductive and inductive (cold-start) subsamples, flexible interfaces for model development and a unified pipeline to train and validate models.

The novelty of the research is twofold. Firstly, we propose a novel model that fixes the problems of the TGN and CAW models by fusing their architectures. Secondly, we propose a unified framework for the evaluation of temporal embedding techniques in downstream graph machine learning tasks. It allows flexible integration of various models and different temporal network data under a unified evaluation framework.

In addition, we prove the effectiveness of the proposed pipeline and its submodules via an extensive ablation study and provide the industrial application of the proposed approach involving the transaction data of a major European bank. We showed that feature enrichment of temporal attention over temporal edge random walks improves quantitative and qualitative results in the real-world application of machine learning tasks on a banking graph.

The experimental study demonstrates the applicability of our method to solve various node/edge prediction tasks on temporal networks and to significantly improve the existing results.

Graph temporality preservation allows the development of recommender systems capable to catch causality and dynamics of user behavior, and being able to adapt to it.

The paper was published in the PeerJ Computer Science Journal. It is indexed as Q1 (2020) and Q2 (2021) in Scopus.

6.5 Exploration in sequential recommender systems via graph representations

The last chapter studies the exploration of poorly known user-item pairs (states) and the corresponding exploration-exploitation trade-off. The underlined problem is to find the new possible interests of the users while their behavior changes

that lead to performance degradation in recommendations and embeddings. The goal of the chapter is to develop novel exploration strategies based on a graph representation of recommender systems problems. To provide exploration we adopt the concept of self-supervised intrinsic motivation from reinforcement learning [52] and bring it to the graph domain.

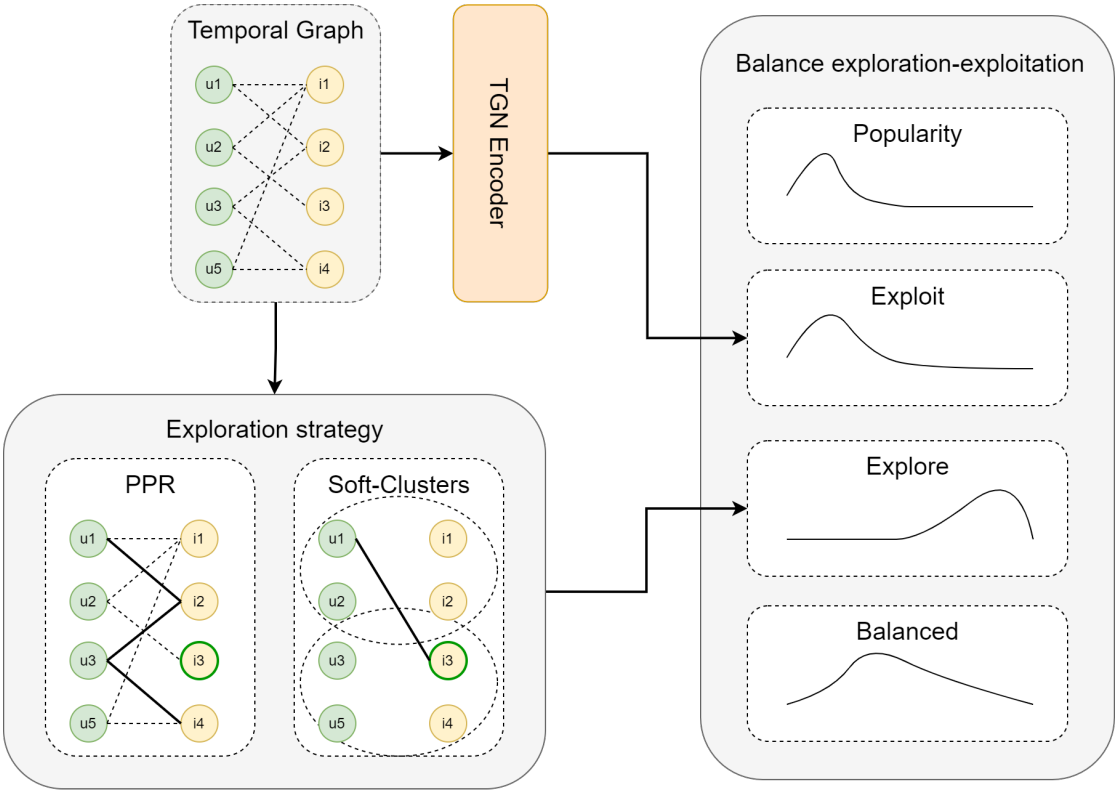

Figure 7: Exploration strategies

The user-item graph is a power-law graph. Thus, new edges in it create between the most popular nodes (Figure 7). So, the exploitation strategy introduces a bias toward the most popular items. In the case of active dynamics in user behavior such nodes become non-relevant anymore. Based on this, we design an exploration strategy based on the personalized PageRank (PPR) ideas [26]. It estimates the local popularity of items near the user. To handle the dynamics, we have modified the PPR to account for only recent temporal information. In our case, random walks are short (length 3) and causal (each next edge occurs before the current). Next, we calculate the number of occurrences of items and add the inverse of it as an exploration bonus to the exploitation scores.

The second exploration method is based on the small-world property. Despite the high density of the user-item interaction graphs, we observe the problem that

the recently added edges are hard to obtain in such graphs. Thus, the probability to aggregate messages from novel nodes is very low. So, we want to add such edges that connect distant nodes to ease the search for the most relevant items. To do it, we perform community detection and connect nodes from distant communities. However, graph clustering is a computationally complex problem. So, we apply the soft-clustering technique. Firstly, it projects graph nodes via GNN in fixed-size space (128). Then, we normalize each embedding with softmax to receive the cluster-assignment vector. Finally, we add an exploration bonus as the cluster-assignment probability of a user divided by the cluster-assignment probability of an item.

The results of the paper show the importance of exploration techniques for the online model adaptation. Models benefit from different types of exploration if the temporal structure is properly presented. The relative performance of exploration methods depends on data properties. For the graphs with few positive edges, it is better to use the personalized PageRank approach. In this case, it samples less diverse nodes and can better estimate the local popularity. On the opposite, graph clustering exploration performs better for the graphs with a large average degree of the node.

The proposed strategies show competitive performance with the other exploration strategies for recommender systems. In the future, we aim to apply the proposed techniques to repeated consumption scenarios. Further, we aim to study proposed exploration strategies in more complex model pipelines like multi-stage recommender systems. Also, it is important to study the proposed techniques for heterogeneous graphs when items and users have different types.

This paper is under review in the IEEE Access Journal. It is rated as Q1 in Web of Science and Scopus.

7 Conclusion

The thesis is based on the published papers [17, 19, 18, 20] and paper [16] which are currently under review in the IEEE Access Journal. The papers [17, 19] analyze the current achievements in the field and conduct an empirical study of the state-of-the-art network embedding techniques and fusion methods, derive practical tips. The papers [18, 20, 16] propose novel models and methods to solve the important graph-based recommender systems issues: context (edge) features awareness, causality, feedback loop and data distribution shift. Together all papers allow solving required tasks to develop the graph-based recommender system and solve its problems caused by environmental dynamics.

The main contributions of this thesis to be defended are the following:

1. We suggested the detailed taxonomy of the network embedding techniques and their applications [17]

2. We conducted benchmarking of the state-of-the-art graph embedding techniques in the link prediction problem [17]

3. We provided the methodology to automatically extract features in the text-attributed graphs for link prediction and node classification [19]

4. We proposed novel self-supervised joint network node and edge embedding based on consistent Line graph representation [18]

5. We suggested novel temporal network embedding achieving state-of-the-art results in various temporal graph machine learning tasks [20]

6. We developed a standardized temporal network embedding evaluation framework and comparison of state-of-the-art models under a common training setting, providing new insights and clarification of real-world performance compared to those reported in the original research articles [20]

7. We provided novel Personalized PageRank [26] based exploration strategy [16]

8. We implemented novel small-world [49] based exploration strategy [16]

9. We described the methodology to apply proposed exploration methods to recommender system with online adaptation in dynamic environments [16]

Directions of future research. Further research problems can be derived from the results of the paper. One of the important findings of the [17] is the lack of meta-strategies to select proper embedding techniques for a specific task in AutoML fashion. The [19] can be extended by applying self-supervised techniques and losses for better joint pretraining of the text and structural encoders. Another prominent way is to reverse the focus from the graph machine learning tasks to the specific problems in information retrieval (e.g. fast first model in multi-stage recommender systems) or Natural Language Processing like in [53]. The main drawback of the JONNEE model [18] is the lack of scalability. To empower work on larger datasets the backbone model in JONNEE can be changed to the fast sampling-based GNNs. Another direction is to analyse the performance of JONNEE on the recommender systems dataset with context information. The paper [20] can be also enhanced by applying scalability techniques. The interactive graph exploration [16] strategies are susceptible to the over-smoothing issue. So, they can be improved by applying differential group normalizations [54] or by pruning a graph on several subgraphs based on the current user context. Another direction for [16] is to consider the heterogeneous edges for different types of user-item interactions like negative or positive feedback, purchase or adding to cart and so on.

References

[1] Sami Abu-El-Haija, Bryan Perozzi, and Rami Al-Rfou. Learning edge representations via low-rank asymmetric projections. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, CIKM '17, page 1787–1796, New York, NY, USA, Nov 2017. Association for Computing Machinery.

[2] Gediminas Adomavicius, Jesse C Bockstedt, Shawn P Curley, and Jingjing Zhang. Effects of online recommendations on consumers' willingness to pay. *Information Systems Research*, 29(1):84–102, 2018.

[3] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3:993–1022, 2003.

[4] Shaosheng Cao, Wei Lu, and Qiongkai Xu. Grarep: Learning graph representations with global structural information. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, CIKM '15, page 891–900, New York, NY, USA, 2015. Association for Computing Machinery.

[5] João Gama, Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. A survey on concept drift adaptation. *ACM computing surveys (CSUR)*, 46(4):1–37, 2014.

[6] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Message passing neural networks. In *Machine learning meets quantum physics*, pages 199–214. Springer, 2020.

[7] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. *arXiv preprint arXiv:1607.00653*, 1607.00653, 2016.

[8] William L Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Proceedings of IC on NIPS'17*, NIPS'17, pages 1025–1035. Curran Associates Inc., 2017.

[9] Zellig Harris. Distributional structure. *Word*, 10(2-3):146–162, 1954.

[10] Dietmar Jannach and Michael Jugovac. Measuring the business value of recommender systems. *ACM Trans. Manage. Inf. Syst.*, 10(4), dec 2019.

[11] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. LightGBM: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30:3146–3154, 2017.

[12] Zekarias T. Kefato, Nasrullah Sheikh, and Alberto Montresor. Which way? direction-aware attributed graph embedding, 2020.

[13] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 1609.02907, 2016.

[14] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 1609.02907, 2017.

[15] Dmitrii Kiselev and Ilya Makarov. Prediction of new itinerary markets for airlines via network embedding. In *International Conference on Analysis of Images, Social Networks and Texts*, pages 315–325. Springer, 2019.

[16] Dmitrii Kiselev and Ilya Makarov. Exploration in sequential recommender systems via graph representations (on review). *IEEE Access*, 2022.

[17] Ilya Makarov, Dmitrii Kiselev, Nikita Nikitinsky, and Lovro Subelj. Survey on graph embeddings and their applications to machine learning problems on graphs. *PeerJ Computer Science*, 7:e357, 2021.

[18] Ilya Makarov, Ksenia Korovina, and Dmitrii Kiselev. JONNEE: Joint network nodes and edges embedding. *IEEE Access*, 2021.

[19] Ilya Makarov, Mikhail Makarov, and Dmitrii Kiselev. Fusion of text and graph information for machine learning problems on networks. *PeerJ Computer Science*, 7, 2021.

[20] Ilya Makarov, Andrey Savchenko, Arseny Korovko, Leonid Sherstyuk, Nikita Severin, Dmitrii Kiselev, Aleksandr Mikheev, and Dmitrii Babaev. Temporal network embedding framework with causal anonymous walks representations. *PeerJ Computer Science*, 8:e858, 2022.

[21] Masoud Mansoury, Himan Abdollahpouri, Mykola Pechenizkiy, Bamshad Mobasher, and Robin Burke. Feedback loop and bias amplification in recommender systems. In *Proceedings of the 29th ACM international conference on information & knowledge management*, pages 2145–2148, 2020.

[22] Costas Mavromatis and George Karypis. Graph infoclust: Leveraging cluster-level node information for unsupervised graph representation learning, 2020.

[23] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013.

[24] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 1301.3781, 2013.

[25] Mingdong Ou, Peng Cui, Jian Pei, Ziwei Zhang, and Wenwu Zhu. Asymmetric transitivity preserving graph embedding. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 1105–1114, New York, NY, USA, Aug 2016. Association for Computing Machinery.

[26] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

[27] Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. Unsupervised learning of sentence embeddings using compositional n-gram features. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 528–540. Association for Computational Linguistics, 2017.

[28] Shirui Pan, Jia Wu, Xingquan Zhu, Chengqi Zhang, and Yang Wang. Tri-party deep network representation. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, IJCAI'16, page 1895–1901. AAAI Press, 2016.

[29] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of ACM SIGKDD IC on KDD'20*, pages 701–710. ACM, 2014.

[30] Bryan Perozzi, Vivek Kulkarni, and Steven Skiena. Walklets: Multiscale graph embeddings for interpretable network classification, 2016.

[31] Idris Rabiu, Naomie Salim, Aminu Da'u, and Akram Osman. Recommender system based on temporal models: A systematic review. *Applied Sciences*, 10(7), 2020.

[32] Shaina Raza and Chen Ding. Progress in context-aware recommender systems—an overview. *Computer Science Review*, 31:84–97, 2019.

[33] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages

3982–3992, Hong Kong, China, November 2019. Association for Computational Linguistics.

[34] Steffen Rendle. Factorization machines. In *2010 IEEE International conference on data mining*, pages 995–1000. IEEE, 2010.

[35] Emanuele Rossi, Ben Chamberlain, Fabrizio Frasca, Davide Eynard, Federico Monti, and Michael Bronstein. Temporal graph networks for deep learning on dynamic graphs. *arXiv preprint arXiv:2006.10637*, 2006.10637, 2020.

[36] Benedek Rozemberczki and Rik Sarkar. Fast sequence-based embedding with diffusion graphs. In *International Workshop on Complex Networks*, pages 99–107. Springer International Publishing, 2018.

[37] Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523, 1988.

[38] Rachna Sethi and Monica Mehrotra. Cold start in recommender systems—a survey from domain perspective. In Jude Hemanth, Robert Bestak, and Joy Iong-Zong Chen, editors, *Intelligent Data Communication Technologies and Internet of Things*, pages 223–232, Singapore, 2021. Springer Singapore.

[39] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*, pages 1441–1450, 2019.

[40] Jiankai Sun, Bortik Bandyopadhyay, Armin Bashizade, Jiongqian Liang, P Sadayappan, and Srinivasan Parthasarathy. Atp: Directed graph embedding with asymmetric transitivity preservation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 265–272. AAAI, 2019.

[41] Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Hao Tian, Hua Wu, and Haifeng Wang. Ernie 2.0: A continual pre-training framework for language understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8968–8975, 2020.

[42] Gábor Takács and Domonkos Tikk. Alternating least squares for personalized ranking. In *Proceedings of the sixth ACM conference on Recommender systems*, pages 83–90, 2012.

[43] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 1710.10903, 2017.

[44] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018.

[45] Shoujin Wang, Liang Hu, Yan Wang, Longbing Cao, Quan Z Sheng, and Mehmet Orgun. Sequential recommender systems: challenges, progress and prospects. *arXiv preprint arXiv:2001.04830*, 2019.

[46] Xiao Wang, Peng Cui, Jing Wang, Jian Pei, Wenwu Zhu, and Shiqiang Yang. Community preserving network embedding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

[47] Xuhong Wang, Ding Lyu, Mengjian Li, Yang Xia, Qi Yang, Xinwen Wang, Xinguang Wang, Ping Cui, Yupu Yang, Bowen Sun, et al. APAN: Asynchronous propagation attention network for real-time temporal graph embedding. *arXiv preprint arXiv:2011.11545*, 2011.11545, 2021.

[48] Yanbang Wang, Yen-Yu Chang, Yunyu Liu, Jure Leskovec, and Pan Li. Inductive representation learning in temporal networks via causal anonymous walks. *arXiv preprint arXiv:2105.02315*, 2101.05974, 2021.

[49] DJ Watts. Strogatz—small world network nature. *Nature*, 393:440–442, 1998.

[50] Shiwen Wu, Fei Sun, Wentao Zhang, Xu Xie, and Bin Cui. Graph neural networks in recommender systems: a survey. *ACM Computing Surveys (CSUR)*, 2020.

[51] Cheng Yang, Zhiyuan Liu, Deli Zhao, Maosong Sun, and Edward Y. Chang. Network representation learning with rich text information. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, page 2111–2117. AAAI Press, 2015.

[52] Tianpei Yang, Hongyao Tang, Chenjia Bai, Jinyi Liu, Jianye Hao, Zhaopeng Meng, and Peng Liu. Exploration in deep reinforcement learning: A comprehensive survey. *arXiv preprint arXiv:2109.06668*, 2021.

[53] Michihiro Yasunaga, Jure Leskovec, and Percy Liang. Linkbert: Pretraining language models with document links. *arXiv preprint arXiv:2203.15827*, 2022.

[54] Kaixiong Zhou, Xiao Huang, Yuening Li, Daochen Zha, Rui Chen, and Xia Hu. Towards deeper graph neural networks with differentiable group normalization. *Advances in Neural Information Processing Systems*, 33:4917–4928, 2020.

[55] Yu Zhu, Hao Li, Yikang Liao, Beidou Wang, Ziyu Guan, Haifeng Liu, and Deng Cai. What to do next: Modeling user behaviors by time-lstm. In *IJCAI*, volume 17, pages 3602–3608, 2017.