Reinforced Anchor Knowledge Graph Generation for News Recommendation Reasoning

Tuo Liu

August 7, 2022

1 Main Contribution

1. The authors propose a brand-new paradigm, AnchorKG, for news recommendation reasoning over a knowledge graph.
2. The authors develop a reinforcement learning-based optimization framework to train the AnchorKG.
3. Experiments on two real-world news datasets demonstrate that AnchorKG not only achieves best accuracy performance but also provides high-quality reasoning knowledge paths.

2 Methodology

2.1 Anchor Graph Generation

The paper propose a new reasoning framework AnchorKG, which is short for Anchor Knowledge Graph generation for recommendation reasoning, and at the core of AnchorKG is the generation of anchor graphs.

The paper use a subgraph generation model π_θ to construct an anchor graph χ_a for each news article n_a. When it comes to the learning part, the paper resorts to reinforcement learning techniques to learn an anchor graph generator π_θ.

The basic idea is, they train an agent that starts from n_a (which can be regarded as a virtual node in the knowledge graph, connecting to all entities mentioned in the article), then iteratively expands the subgraph by examining whether the connecting neighbor nodes should be absorbed into the subgraph, until the subgraph reaches a stopping criterion.

2.1.1 The Markov Decision Process

The authors formulate the anchor graph generation problem as a Markov Decision Process (MDP) $M = \{S, A, R, P\}$, where S is a finite set of states during exploration, A is a finite set of actions, R is the reward function from the environment, and P is the transition probability function.

2.1.2 AnchorKG Policy Network

The paper develop a neural policy network that learns the probability distribution of taking actions a given a state $s_t : \pi_\theta(s_t, a_t) = P(a_t|s_t, \theta)$, where θ is the parameter of the policy network.

2.1.3 Implementation of Rewards

In Section 2.1.1, The rewards are comprised of two components: the terminal reward and the immediate reward. In this part the paper gives a detailed definition for them.

- Terminal Reward: After the generation process finishes, we need to evaluate the quality of the generated anchor graph. Because we expect to use the anchor graph to help both recommendations and reasoning, we rely on the feedback from these two tasks to define the terminal reward:

  - Recommendation Reward: This reward encourages the anchor graph to adjust the document representation, so that related item pairs become closer in the latent space, and unrelated item pairs become less close.
  - Reasoning Reward: The reasoning process is established based on the interactions between two anchor graphs.

- Immediate Reward: Terminal Reward can only be acquired until the agent reaches an ending state. To make the policy network converge better, the paper designs some immediate rewards:
  - Coherence Reward: the reward encourage the agent to select entities which have similar semantic topics (we call it coherence) to the current news article n_a.
  - Hit Reward: the paper hopes the entities mentioned by n_a's related items can be absorbed into the anchor graph. Thus, the hit reward is defined as whether there is at least one document in $\kappa(a)$ which mentions entity e.

2.2 Item-to-item Recommendation

Once the anchor graph χ_a of article n_a is generated, we can use χ_a to enhance the original document representation of n_a.

1. Firstly, the paper encodes an anchor graph into a latent vector.
2. Secondly, the paper uses the cosine similarity between the knowledge-aware representations of two articles to measure the closeness of them.
3. Finally, the paper designs an auxiliary scoring model to predict the label of an item pair with the path score.

2.3 The Optimization Framework

In this section, the paper introduces the multi-task optimization framework to train the AnchorKG.

2.3.1 Critic

The critic network is trained with the Temporal Difference (TD) method, which learns model parameters by bootstrapping from the current estimate of the action-value function.

1. It first calculates a target q_t according to the Bellman equation.
2. Then the critic network is learned by minimizing the TD error.

2.3.2 Actor

The actor aims to maximize the expected payoff of each step. The paper uses the policy gradient method to optimize the actor network.

2.3.3 Warm Start Stage

The papper adopt the idea of behavior cloning to guide the model training from expert demonstrations as a warm start training process.

2.3.4 Negative Sampling

The paper design a simple but effective knowledge-aware negative sampling strategy.

2.3.5 Multi-Task Learning

There are two tasks in our framework, i.e. the anchor graph generation task (generator) and the item2item recommendation task (recommender). At each iteration, the paper first freezes the recommender parameters and optimizes the generator, then freezes the generator parameters and optimizes the recommender. The iteration goes on until the model converges.

3 Experiments

The overall performance of the model is as follow:

Table 1: Overall recommendation performance of different models on two datasets. Results are reported in percentage (%).

	MIND Dataset								Bing News Dataset							
	Top-5				Top-10				Top-5				Top-10			
	NDCG	Prec.	Recall	HR	NDCG	Prec.	Recall	HR	NDCG	Prec.	Recall	HR	NDCG	Prec.	Recall	HR
DKN	3.428	2.217	6.252	8.145	5.258	1.596	10.23	13.41	2.929	1.467	3.466	6.099	4.225	1.001	4.978	9.281
NAML	3.718	2.311	6.879	8.712	5.706	1.734	11.09	14.54	3.318	1.727	4.192	7.357	5.109	1.186	6.063	11.20
S-BERT	3.929	2.426	6.929	9.285	5.997	1.820	11.67	15.30	3.455	1.789	4.263	7.560	5.227	1.224	6.108	11.56
KRED	4.036	2.591	7.157	9.603	6.106	1.869	12.20	16.00	3.617	1.878	4.474	7.936	5.450	1.286	6.813	11.95
PGPR	2.484	1.618	3.894	5.731	4.130	1.229	6.701	9.898	1.752	0.983	1.851	3.306	2.168	0.624	2.630	4.851
ADAC	2.671	1.650	4.110	5.855	4.295	1.276	6.893	10.09	1.777	1.002	1.886	3.371	2.216	0.648	2.693	4.998
AnchorKG	**4.301**	**2.805**	**8.186**	**10.755**	**6.621**	**2.112**	**14.71**	**17.92**	**3.866**	**2.046**	**4.733**	**8.370**	**5.623**	**1.497**	**7.513**	**12.49**

Figure 1: Performance

1. Embedding-based baselines, including DKN, NAML and KRED, are designed for news recommendations with strong text encoders. They are with high recommendation performance but cannot produce explainable reasons for recommendations.

2. ADAC and PGPR are path-finding-based methods for recommendations and are recognized as state-of-the-art recommendation reasoning methods over knowledge graphs. Despite their ability for reasoning, their major weakness lies in the low accuracy of news recommendations, where the textual content plays a key role in matching news relevance.

3. AnchorKG combines the advantage of embedding-based methods and path-find-based methods. It outperforms all the baselines in terms of different metrics, and the conclusions are consistent on two datasets, which demonstrates the effectiveness of AnchorKG.

4 Future Work

The authors plan to conduct experiments with the AnchorKG paradigm on the user-to-item recommendation scenario.

5 My Idea

This paper uses AnchorKG to represent an article, and uses RL method to train the AnchorKG generator. And Once anchor graphs are generated, the knowledge-aware reasoning of any two news articles can be conducted simply based on the interactions of their corresponding anchor graphs. I think it is novel to use a subgraph to represent the item, and in the future we may use this method in other scenarios.