CADReN: Contextual Anchor-Driven Relational Network for Controllable Cross-Graphs Node Importance Estimation

Anonymous ACL submission

Abstract

Node Importance Estimation (NIE) is crucial for integrating external information into Large Language Models through Retriever-Augmented Generation. Traditional methods, focusing on static, single-graph characteristics, lack adaptability to new graphs and user-specific requirements. **CADReN**, our proposed method, addresses these limitations by introducing a Contextual Anchor (CA) mechanism. This approach enables the network to assess node importance relative to the CA, considering both structural and semantic features within Knowledge Graphs (KGs). Extensive experiments show that CADReN achieves better performance in cross-graph NIE task, with zero-shot prediction ability. CADReN is also proven to match the performance of previous models on single-graph NIE task. Additionally, we introduce and opensource two new datasets, **RIC200** and **WK1K**, specifically designed for cross-graph NIE research, providing a valuable resource for future developments in this domain.

1 Introduction

The advent of Transformer-based Large Language Models (LLMs) (Vaswani et al., 2017; Radford et al., 2018; Brown et al., 2020; OpenAI, 2023; Touvron et al., 2023) has catalyzed the development of AI Agents for advanced analytical and decision-making tasks. Yet, LLMs alone are prone to "hallucination," leading to inaccuracies. The introduction of Retriever-Augmented Generation (RAG) (Lewis et al., 2020) has become essential to enhance LLMs by integrating structured and precise Knowledge Graphs (KGs), thereby mitigating this issue.

KGs provide a structural framework to encapsulate heterogeneous data, allowing for intricate mappings of entity relationships. Their structured nature is conducive to pattern recognition and insight formation. Enhanced by high-performance graph management systems such as Neo4j (Neo4j Company, 2012), KGs have become integral to domains dependent on structural information, including recommendation systems (Le et al., 2023), fraud detection (Chen et al., 2020), and drug discovery (Isert et al., 2023; Atz et al., 2021). Their structured knowledge is essential for augmenting LLMs to improve performance.

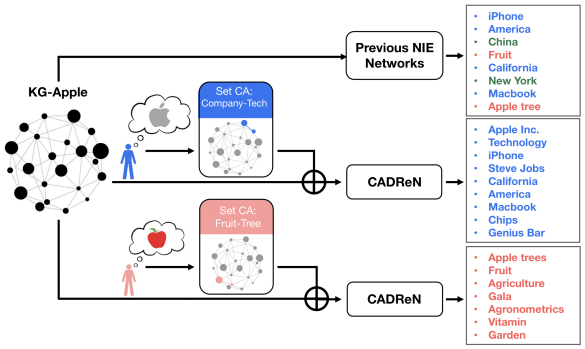

Figure 1: CADReN leverages user-defined Contextual Anchors (CAs) to enhance precision in KG queries. In the figure, *KG-Apple* contains diverse information related to *Apple*. Users applying *Company-Tech* and *Fruit-Tree* CAs receive focused outputs via CADReN, contrasting with the generalized results given by previous NIE networks without CA utilization."

Within the business sphere, leveraging AI to identify new opportunities and predict market disruptions has become a research focus. Integrating KGs with LLMs (Pan et al., 2023) has proven critical, with the effectiveness of KG-enhanced LLMs heavily reliant on the quality of retrieved information. This retrieval, defined as the Node Importance Estimation (NIE) task, is increasingly recognized for its significance.

NIE is a fundamental aspect of Information Retrieval, focusing on evaluating and scoring the relevance of nodes in a Knowledge Graph. This process plays a crucial role in enhancing the effectiveness of RAG by ensuring the most pertinent graph information is prominently featured. Current ap-

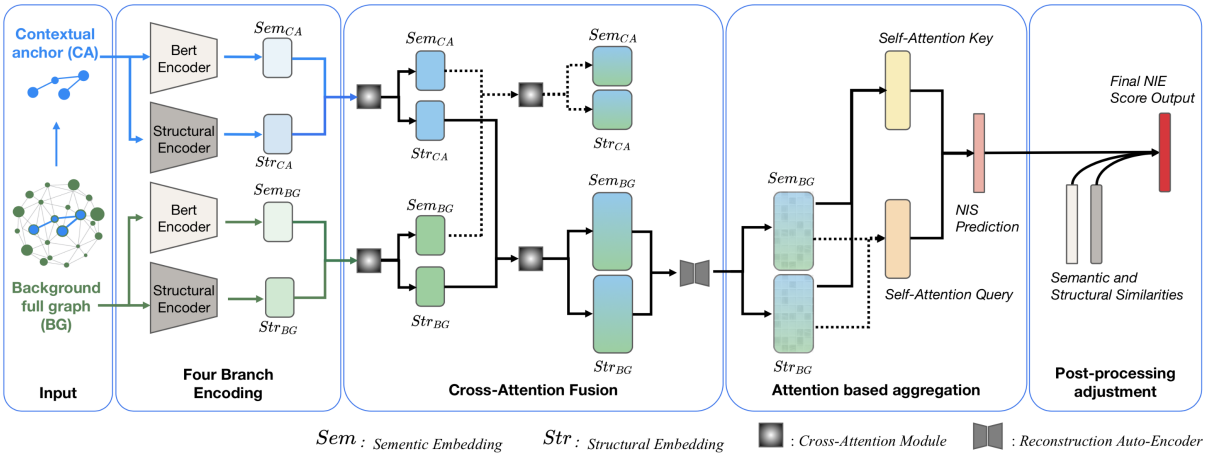

Figure 2: The figure above presents the overall architecture of the CADReN model. The semantic and structural information in CA and BG are encoded in BERT and our proposed structural encoder, respectively. Cross-attention fusion is then applied to the combinations of these embeddings to capture the relational information between CA and BG. The BG embeddings mixed with the information from CA are then used to predict the NIE scores, with the introduction of Reconstruction Auto-encoder, Attention-based Aggregation mechanism and Post-Processing mechanism to improve the quality of the output.

proaches, including Structure-Pattern-Based Methods like PageRank (Page et al., 1999), HITS (Liu et al., 2018), HAR (Li et al., 2012), and Embedding-Based Methods like GNN (Cummings and Nassar, 2020; Tang and Liu, 2023), GENI (Park et al., 2019), and RGTN (Huang et al., 2021), are hindered by two major deficiencies: their focus on static single-graph information and the inability to transfer learning across graphs without retraining. Additionally, their static definition of "importance" often leads to outputs that may not align with the specific interests of users. (see Fig. 1).

Addressing these challenges, we introduce **CADReN** (**C**ontext **A**nchor-**D**riven **Re**lational **N**etwork) for cross-graph NIE tasks. CADReN leverages user input—Contextual Anchors (CA)—to delineate relative node importance within the KG, enabling transferability across graphs and user-driven result customization (detailed in Fig. 2). Extensive experiments showed the effectiveness of our method, especially on multi-graph tests.

The paper proceeds with a review of NIE literature, core concept definitions, CADReN's architecture, experimental datasets and results, culminating in a conclusion.

Our main contributions are:

- A transferable KG modeling method using CA, enabling efficient cross-graph NIE inference without retraining.
- A novel, controllable NIE paradigm with CA as a user-network interface for flexible outcomes.
- The introduction of **RIC200** (**R**elevant **I**nfo in **C**ontext-200) and **WK1K** (**WiK**ipedia-1000) datasets to foster cross-graph NIE research. (Details in section **Dataset**.)

2 Related Works

Node Important Estimation began with an initial focus on structural information, further evolved to embedding-based methods capturing the rich information from KGs, and recently shifted towards more sophisticated paradigms combining these approaches with KGs and LLM.

PageRank (PR) (Page et al., 1999), a seminal NIE technique, initially gauged the importance of web pages effectively. It was refined by Personalized PageRank (PPR) (Wang et al., 2020) and Hub, Authority, and Relevance Score (HAR Score) (Li et al., 2012) to address its limitations. Nevertheless, these approaches, focused on node connectivity, often overlook the nuanced semantics within KGs, resulting in suboptimal performance in complex scenarios, as evidenced by empirical studies (Park et al., 2019; Huang et al., 2021).

2.1 Embedding-Based Approach

The advent of embedding-based frameworks marked a shift towards capturing the intricacies of KGs. Initially, methods like node2vec (Grover and

Leskovec, 2016) still prioritized structural properties. However, the rise of Graph Neural Networks (GNN) (Cummings and Nassar, 2020) signified a methodological leap, leveraging neighborhood aggregation to improve NIE. The continued innovation in network architectures, including Graph Convolution Networks (Kipf and Welling, 2017) and Transformers (Veličković et al., 2017), has seen embeddings become pivotal in KG research. For instance, GENI (Park et al., 2019) and its successor MULTIIMPORT (Park et al., 2020) have pushed the boundaries of latent node importance identification, drawing on GNN and Transformer principles. Yet, despite their efficacy, the application of these models to new KGs often necessitates expensive retraining, limiting their practical deployment.

2.2 Integrating KG to LLMs

Traditional graph-based machine learning methods are facing bottlenecks in handling general knowledge and semantic understanding, necessitating the integration of LLMs with KGs. (Chen et al., 2023). Applications utilizing both, such as SPARQL-enhanced Question Answering (Lehmann et al., 2023) and LARK's KG-based reasoning (Choudhary and Reddy, 2023), have emerged. These integrative approaches generally fall into two streams (Pan et al., 2023): direct knowledge infusion during LLM training, exemplified by ERNIE (Zhang et al., 2019) and K-BERT (Liu et al., 2019), and prompt-based information channeling as seen in ReLMKG (Cao and Liu, 2023) and GreaseLM (Zhang et al., 2022). The latter, accommodating dynamic and real-time knowledge, is particularly apt for the fluid business sector. This highlights NIE's crucial role in extracting relevant information from KGs, especially given the limited context window of LLMs, to ensure that only the most critical and pertinent data is utilized for model inputs.

3 Preliminaries

In this section, we will provide a formal definition of the core concepts, alongside the NIE task.

3.1 Graph

Definition: A graph is a mathematical structure denoted as $G = (V, E)$ consisting of a non-empty set V of vertices (or nodes) and a set of edges E . Vertices represent distinct entities or elements, while the edges delineate the connections or relationships between these vertices.

3.2 Node Importance Estimation task

Definition:The Node Importance Estimation task is centered on assigning an Importance Score to each node within a graph. Specifically, for a given user input q and a KG G, the goal is to identify a function f such that $f(q, G) = I$. Here, I represents a vector wherein the i-th element signifies the Importance Score of the i-th node of G. Previous work learns a function g such that $g(G) = I$, which does not take q as an input.

3.3 CA, BG and GT node subsets/subgraphs

Definition: In the context of a graph, CA (**C**ontextual **A**nchor), BG (**B**ack**G**round), and GT (**G**round **T**ruth) represent three node subsets, satisfying consecutive inclusion: CA $\subset$ GT $\subset$ BG. The CA subset consists of nodes present in the user's input query q. The GT subset comprises nodes designated as "important", which are used as training labels. The BG subset encompasses all the nodes within the graph. CA/GT/BG (sub)graphs are simply the subgraphs containing the CA/GT/BG nodes.

4 Model Architecture

In this section, we outline our model's architecture, detailed in Figure 2. The process begins with separate encoders extracting semantic and structural features from the KG. These features are then fused for both CA and BG graphs, integrating structural and semantic information. A cross-attention mechanism further refines the interaction between CA and BG features. Finally, a classifier predicts the importance of each BG node, with our proposed loss function incorporating Binary Cross-Entropy loss, semantic loss, and structural loss.

4.1 Four Branch Encoding

Our model employs a dual-encoding approach, leveraging both a BERT Encoder (chosen following the setting in (Huang et al., 2021)) for semantic analysis and a naive Structural Encoder for structural insights. This process, termed Four Branch Encoding in Fig. 2, is designed to obtain distinct semantic and structural embeddings for the CA and BG graphs.

4.1.1 Semantic Embedding

Semantic embedding of $node_i$ is derived by encoding the concatenation of $node_i$ and all CA nodes with BERT. Encoding $node_i$ along with the CA nodes is advantageous because the BERT encoding

process encodes information from the CA nodes into the embedding of $node_i$. This facilitates learning of the relative relationships between nodes. In order to get a fix-length embedding for all the nodes, We extract and concatenate the embeddings of the first and last tokens of $node_i$ to form its semantic representation.

4.1.2 Structural Embedding

The structural embeddings encompass 5 key node statistics: [#(child nodes), #(direct child nodes), {max,min,avg} of steps to reach CA nodes]. These features, selected based on business analyst feedback, capture both the structural significance and proximity of $node_i$ to CA nodes. Previous structural encoders like node2vec (Grover and Leskovec, 2016) and GNN (Cummings and Nassar, 2020) facilitate the mapping of structural information onto a higher-dimensional space, thus endowing the model with enhanced representational capabilities. However, integrating relative relationships into these encoders poses notable challenges. In our devised encoder, the relative associations with CA are explicitly taken into account, thereby constituting an initial endeavor towards a CA-aware structural encoder.

4.2 Cross-Attention Fusion

This phase integrates semantic and structural data from both the CA and BG graphs. It employs cross-attention mechanisms, first between semantic and structural embeddings, then between the CA and BG graph embeddings. Each embedding, processed through a Transformer-like encoder, amalgamates information from the other three sources. This fusion not only enhances learning of the "importance" concept but also establishes hidden relationships with CA nodes. The embeddings undergo further refinement via a Reconstruction Auto-Encoder, which aids in model robustness by training a Multi-Layer Perceptron (MLP) to reconstruct randomly dropped node embeddings.

4.3 Attention-based Aggregation

The third segment of our model introduces an Attention-Based Aggregation mechanism. This component is pivotal in predicting the Node Importance Score (NIS) using the embeddings generated in the earlier stages of the model. This mechanism is illustrated in Figure 3.

The core principle underlying this mechanism is the utilization of self-attention. Initially, the

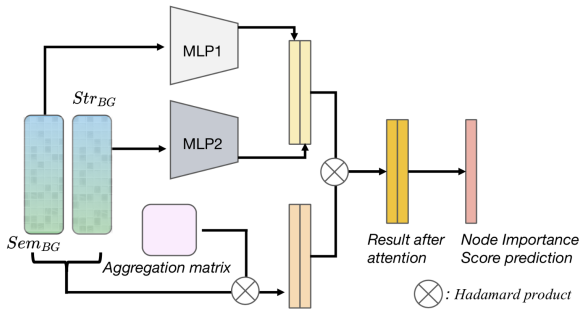

Figure 3: Attention based Aggregation mechanism. The *Aggregation matrix* contains trainable attention parameters, which are used to produce the self-attention Query that guides the prediction of Node Importance Score.

embeddings from the cross-attention module are processed through two MLP encoders. This step generates the Key tensor for self-attention. Concurrently, the embeddings are transformed by an "aggregation matrix" and reshaped into the Query tensor that mirrors the shape of the Key tensor.

The Hadamard product between the Key and Query tensor yields a tensor of shape $[\#node, 2]$. Each row of this tensor encapsulates two NIS, one derived from semantic embeddings and the other from structural embeddings.

To finalize the prediction of NIS, the model aggregates these semantic and structural NIS values. This aggregation is then refined with a softmax function, ensuring a normalized probabilistic output for the NIS.

4.4 Post-processing Adjustment

In the final part, we introduce Post-processing Adjustment to further enhance the model's performance. This is achieved by calculating a weighted summation between the predicted NIS vector, the semantic similarity vector, and the structural similarity vector.

4.4.1 Semantic Similarity Vector

The semantic similarity vector is computed by averaging the cosine similarity between the $node_i$'s semantic embeddings and the embeddings of the CA nodes. The i-th element of the semantic similarity vector, denoted as $\mathcal{S}_{sem,i}$, is calculate as follows:

$$\mathcal{S}_{sem,i} = \frac{\sum_{j=1}^{|CA|} \langle \mathcal{E}_{sem}(node_i) | \mathcal{E}_{sem}(CA_j) \rangle)}{|CA|} \quad (1)$$

where: $\mathcal{E}_{sem}(.)$ represents the semantic embedding obtained via BERT encoder. $\langle .|. \rangle$ denotes the func-

tion of cosine similarity. $|CA|$ denotes the number of nodes in the CA set.

For nodes included in the CA graph, their semantic similarity is assigned a maximum value (1).

4.4.2 Structural Similarity Vector

The structural similarity vector is obtained using a function determined by regression. This function takes the structural features of a node as input and outputs a scalar between 0 and 1 representing the structural similarity between the node and the CA nodes. The $node_i$'s structural similarity $\mathcal{S}_{str,i}$ is defined as:

$$\mathcal{S}_{str,i} = b + \mathcal{R}[\mathcal{E}_{str}(node_i)]^{tr} \quad (2)$$

where: $\mathcal{E}_{str}(.)$ represents the structural embedding of a node. $\mathcal{R}$ and b are the regression parameters and bias respectively.

We perform the regression with 5% randomly sampled data from the training set. The ratio between CA, GT and BG node numbers are kept during the sampling. Once the $\mathcal{R}$ and b are determined, we fix them to calculate the structural similarity of any given node.

4.4.3 Weighted Summation

The final NIS (I_{final}) is obtained as follows:

$$I_{final} = \sigma(\alpha * I_{init} + \beta * \mathcal{S}_{sem} + \gamma * \mathcal{S}_{str}) \quad (3)$$

where: α, β and γ are trainable parameters. $\sigma(.)$ is the sigmoid function.

In this step, we refine the prediction results using the similarity vectors. The similarity vectors provide additional information about the CA nodes, enabling the model to better distinguish nodes with similar initial NIS predictions.

4.4.4 Loss Function

The loss function of our model is defined as follows:

$$\mathcal{L}_{total} = \mathcal{B}(I_{gt}, I_{final}) + \mathcal{L}_{sem} + \mathcal{L}_{str} \quad (4)$$

$$\mathcal{L}_{sem} = \mu * \mathcal{B}(\mathcal{S}_{sem} * I_{gt}, I_{final}) \quad (5)$$

$$\mathcal{L}_{str} = \nu * \mathcal{B}(\mathcal{S}_{str} * I_{gt}, I_{final}) \quad (6)$$

where: $\mathcal{B}(.)$ is the function to calculate Binary Cross Entropy. I_{gt} and I_{final} represent the ground truth and the prediction values of NIS. $\mathcal{L}_{sem}$ and $\mathcal{L}_{str}$ are loss terms weighted on semantic and structural similarities. μ and ν are hyperparameters.

In this loss function, we incorporate two weighted terms to prioritize the losses associated with nodes that are either semantically or structurally important. This setting strengthens the model's robustness against noise from nodes that are semantically unrelated or structurally distant from the CA nodes.

5 Experiments

This section describes our experiments that aim to answer the following research questions:

- Cross-graph Performance: Does CADReN outperform other approaches for cross-graph NIE tasks? Can it do zero-shot inference on different graphs without retraining?
- Single-graph Performance: does our model perform on par with previous works?
- Impact of CA: By introducing CA, does CADReN show better flexibility and controllability in NIE tasks?

5.1 Datasets

Our model is designed for multi-graph scenario, for which there are no datasets readily available. We have created our own datasets, and we plan to opensource RIC200 and WK1K to the community.

For each node inside the graphs of these datasets, it is labeled as one type among {CA, GT, BG}. Nodes are labeled in a way to simulate the real-world application scenario: the CA nodes given by a user reflecting his/her interest, the GT nodes showing the expected responses, and the BG nodes representing the knowledge resource. In other words, the CA and GT nodes are labeled in accordingly, we call them a "pair". It is worth mentioning that, on average, each graph has 5 pairs of (CA, GT). We use different pairs of (CA, GT) to test the model's ability to give flexible outputs.

In order to compare with previous single-graph oriented models, for most of the datasets we used, a single-graph version is constructed, by simply putting all the graphs into one giant graph.

The datasets used are listed in Table 1:

RIC10K: a dataset containing 10k KGs covering the business landscape knowledge of different industries, which are generated based on documents like annual reports and research reports. **RIC200**: a dataset containing 250 KGs selected from RIC10K. **WK1K**: a dataset containing 1000 KGs that are constructed based on Wikipedia data and relevant

Database	#Edges	#BG	#GT	#CA	#Graphs
FB15K-S	592,213	14,591	1,459	150	1
FB15K-M	3006	74	7	5	197
RIC200-S	63,802	36,607	2,004	617	1
RIC200-M	319	183	13	3	250
RIC10K-M	77	43	10	3	10,000
WK300-S	97,654	90,746	1,884	950	1
WK300-M	311	289	6	3	314
WK1K-M	318	295	6	3	1,024

Table 1: Statistics of datasets used in our experiments. All the numbers are averaged numbers. The suffix "-S/-M" represent "**S**ingle/**M**ulti-graph" version.

articles, containing general knowledge across domains. **WK300**: a dataset containing 314 KGs selected from WK1K. **FB15K** (Bollacker et al., 2007): an open dataset containing general information across domains. Following the settings of RGTN, each node in it is accompanied with the descriptions extracted from WikiData [1]. The NIS is represented by the node's pageview number on Wikipedia in the past 30 days. Around top-1% (resp. top-10%) of nodes with the highest pageview numbers are marked as the CA (resp. GT) nodes.

For the two newly proposed datasets, we give the details of their creation process here.

RIC10K: Thousands of open articles are collected from the Internet. Through Named Entity Recognition and Relation Analysis, these articles are turned into 10,000 KGs, grouped by themes. In each KG, we generate some commonly asked questions (queries) with ChatGPT. The nodes mentioned in these queries are labeled as "CA" nodes. Then, a group of consulting experts labeled the nodes highly related to the given query as "GT" nodes. Overall about 7% (resp. 23%) of the nodes are labeled as "CA" (resp. "GT") nodes.

WK1K: 1,000 simulated queries are first generated with ChatGPT. For each query, its relevant articles are obtained via search engines with the query being the search input. The nodes mentioned in the queries are labeled as "CA" nodes, while the top 10% nodes with highest word frequency in the "relevant articles" are marked as the "GT" nodes. Approximately 1% (resp. 2%) of the nodes are labeled as "CA" (resp. "GT") nodes.

During the experiment, when a single-graph based model (GENI, RGTN) is applied on a multi-graph dataset, the model process each graph sequentially. Multi-graph based methods (GPT-3.5, CADReN) are compatible with the single-graph setting, thus can be applied without modification.

[1] https://www.wikidata.org

5.2 Baselines

We compare our work with two previous Transformer-based methods: GENI (Park et al., 2019), RGTN (Huang et al., 2021), as well as a representative of the generative models: GPT-3.5-Turbo (Brown et al., 2020) (referenced as GPT-3.5).

GENI and RGTN adopt Single-Graph Oriented Structure (SGOS), however, real-world KG datasets are composed of multiple KGs. When SGOS models are applied to these datasets, the graphs need to be aggregated into one graph first. In most scenario, this aggregation is not practical because of the size of data. Even in situations when we could aggregate the graphs, our experiments show that such work-around does not give satisfactory results (Table 3). Therefore, our network is deliberately designed to adopt a Multi-Graph Oriented Structure (MGOS). To give a comprehensive comparison, our experiments cover both the single-graph and the multi-graph settings.

CA could be introduced to GPT-3.5 through prompts, while GENI and RGTN can not take CA as input by design. During the experiments of GENI and RGTN, the information from CA was carefully masked to avoid data leakage.

All the baselines were run with the same data under their default settings. The experiments are conducted on NVIDIA GeForce RTX 2080 Ti GPUs. The models are trained until convergence using the Adam Optimizer with a learning rate of 5E-3.

5.3 Metrics

Building upon the study conducted by GENI (Park et al., 2019), we employ the metrics of Normalized Discounted Cumulative Gain (**NDCG**) and Spearman's rank correlation coefficient (**SPM**) to conduct a comprehensive evaluation of the ranking quality and importance correlation. Additionally, we introduce a novel metric called Overlap@k (**OVER**), to assess the recall of important nodes following the ranking of node importance on a dynamic range.

NDCG is a commonly employed metric for evaluating the quality of rankings that takes into account the order of elements. For this specific task, we define the graded relevance values as the ground truth importance values after applying a logarithmic transformation. When presented with a list

of nodes and their corresponding predicted importance scores, as well as their ground truth importance values, we sort the nodes by the predicted importance scores and take the corresponding ground truth importance at the position i as rel_i. $DCG@k$ is defined as:

$$DCG@k = \sum_{i=1}^{k} \frac{rel_i}{log_2(i+1)} \quad (7)$$

The Ideal DCG ($IDCG$) is the DCG of the ground truth list. NormalizedDCG at position k ($NDCG@k$) is calculated by:

$$NDCG@k = \frac{DCG@k}{IDCG@k} \quad (8)$$

SPM, or SPEARMAN, measures the correlation between the predicted NIS list $pred$ and the ground truth list $label$. After converting the raw values $pred$ and $label$ into the ranks R_{pred} and R_{label} , SPM is calculated by:

$$SPM = \frac{cov(R_{pred}, R_{label})}{\sigma_{R_{pred}} \sigma_{R_{label}}} \quad (9)$$

where: $cov()$ is the covariance function. $\sigma_{R_{pred}}$ and $\sigma_{R_{label}}$ are the standard deviations of the ranks.

OVER is the overlap ratio of the top-m important predicted nodes (I_{pred}) and their corresponding labels (I_{gt}). Since we are evaluating a cross-graph task, the m is set dynamically to cope with graphs with different sizes. The $OVER@k$ is attained by:

$$m = k * |GT| \quad (10)$$

$$OVER@k = \frac{|I_{pred,top-m} \cap I_{gt,top-m}|}{m} \quad (11)$$

where: $|GT|$ is the number of nodes in GT set.

5.4 Cross Graph Evaluation

CADReN outperforms other approaches on multi-graph setting due to its MGOS design. The design goal of SGOS models is to learn absolute information about each node in one graph. When they are used to process multiple graphs, information from multiple graphs interfere with each other rather than complement each other. CADReN, on the other hand, with the help of CA, it can learn generalized relative relationship information from multiple graphs, leading to a significantly enhanced performance on multi-graph tasks.

Moreover, CADReN demonstrates its ability of zero-shot inference across graphs. This feature confirms that CADReN learned the transferable relative relations. Results of the experiment are organized in Table 2.

5.5 Single Graph Evaluation

Single-graph NIE has been the center of NIE researches during a long time. In order to better compare with the previous works, CADReN is also tested under single-graph setting with baselines. Experiment results are organized in Table 3. The results show that, even though CADReN is not built upon single-graph scenario, it still matches the performance of previous works, getting the best or second best outcomes in most tests.

5.6 Effectiveness of CA

The introduction of the CA allows users to interact with the NIE network, leading to more accurate and more flexible NIE predictions. To demonstrate this feature, we apply NIE with fixed BG nodes while altering the (CA, GT) pairs. CADReN successfully captures this change and gives prediction accordingly, while previous works can not adapt to the change of context. One qualitative result is shown in Fig. 4. More results in **Appendix A**.

Model	GENI	RGTN	GPT3.5	CADReN	GPT3.5	CADReN
CA	No CA given	No CA given	CA 1: MLCC, Chips		CA 2: IGBT, Thyristor, Mosfet, Power device, Jiejie Microelectronics	
Predictions	SLP board	Electronic Circuits	MLCC	MLCC	IGBT	IGBT
	SLP	Circuit Control	Chip	Chip	Thyristor	Power Devices
	Playstation4	Battery to Motor	Semiconductor	5G	Mosfet	Jiejie Microelectronics
	PTC	Battery to Appliance	Sensor	Electronic Circuit	Power Devices	Thyristor
	PCB/	Motor Control	Battery	Automotive Electronics	Power Semiconductor Modules	Mosfet
	PCB	Photovoltaic	Circuits	Internet of Things	High Frequency High Speed Materials	Mems sensors
	PCB high level hardboard production line	Passive Components	Printed Metal Electrode Paste	Electronic Components	OLED full screen	Servers
	Playstation	Capacitors	Thyristors	Intel	Intelligent Hardware Devices	Circuit Control
	Mosfet	Wind power	Ceramic Dielectric Diaphragms	Low Loss High Frequency High Speed Materials	Photovoltaic	Internet of Things
	MLCC	Automotive Batteries	Integrated Circuits	Medium Loss High Speed Circuit Substrate	Outbreak	Intelligent
	LED	Internet of Things	Otis	Electronic label	Semiconductor	New Energy Vehicles
	Lighting port	Controllers	Rigid Board	Server Materials	High Performance	Battery Management
	OLED full screen	Battery Management	Five Wire Connectors	Automotive Battery	Sensors	UPS
	Mosfet power semiconductor module	Batteries	Capacitors	Circuit Control	Microelectronics	Photovoltaic
	OLED pressure sensitive touch screen	Electric Vehicles	Electronic Components	Printed Metal Electrode Pastes	New Energy Vehicles	Wind power
	OPPO	Integrated Circuits	Optical Devices	Smart Analytics	Chips	Grid
	FPC	IGBT	Camera Module	Servers	Internet of Things	Railroad
	Fitbit	Railroad	Coils	Controllers	Battery	Switches
	AI speaker	New Energy Vehicle	Surface Mount Capacitors	Capacitors	High-end	Electronic Circuits
	EMBB	Switch	Five-wire connectors	Logistics		Motor Controls

Figure 4: Top 20 nodes with highest NIS predicted. Red (resp. orange) nodes are GT nodes corresponding to CA 1 (resp. CA 2) nodes.

5.7 Effectiveness of Structural Information

LLMs are powerful for textual information analysis, it is natural to use LLM for NIE tasks directly. However, due to the lack of structural information and of up-to-date information, GPT-3.5 shows less ideal performance, as shown in Table 4.

Methods	FB15K-M			RIC{200†, 10K‡ }-M			WK1K-M		
	NDCG	SPM	OVER	NDCG	SPM	OVER	NDCG	SPM	OVER
GENI†	0.7761	0.4105	0.5168	0.7825	0.4277	0.4507	0.8136	0.4447	0.7462
RGTN†	0.8563	0.4403	0.5502	0.8228	0.3247	0.4402	0.8412	0.4931	0.7756
CADReN‡	**0.9917**	**0.6294**	**0.8988**	**0.8922**	**0.6232**	**0.8675**	**0.9064**	**0.6390**	**0.8641**
CADReN†,△	0.9617	0.6093	0.8176	0.8633	0.5899	0.8412	0.9007	0.6109	0.8199

Table 2: Evaluation results of different models across datasets under multi-graph NIE task setting. NDCG and SPM are calculated with top 20 nodes, while the k parameter of Overlap is set as 2. The results in the row of CADReN △ is obtained by first training CADReN on RIC10K, then inference on other datasets. Best results are in bold, second best results are underlined.

Methods	FB15K-S			RIC200-S			WK300-S		
	NDCG	SPM	OVER	NDCG	SPM	OVER	NDCG	SPM	OVER
GENI	0.9191	0.7520	0.3901	**0.7095**	0.4231	0.2412	**0.5899**	0.2326	**0.1700**
RGTN	**0.9550**	**0.8007**	**0.4720**	0.6622	0.4387	0.2500	0.5257	**0.2741**	0.1600
CADReN	0.9322	0.7743	0.4172	0.6321	**0.4778**	**0.2612**	0.5311	0.2601	0.1612

Table 3: Evaluation results of different models on single-graph datasets. NDCG and SPM are calculated on the top 100 nodes, while the k parameter of Overlap is set as 2. CADReN achieves similar performance on single-graph NIE compared with previous works even though it is not specifically designed for it. Best results are in bold, and second best results are underlined.

Methods	RIC200-M			WK300-M		
	NDCG	SPM	OVER	NDCG	SPM	OVER
GPT-3.5	0.41	0.51	0.21	0.61	0.55	0.45
CADReN	0.87	0.61	0.85	0.92	0.63	0.87

Table 4: GPT-3.5's ability on NIE task is not satisfactory due to the lack of structural information and of up-to-date information.

	NDCG	SPM	OVER
w/o CA	0.6968	0.3211	0.1275
w/o AA	0.7338	0.5363	0.8095
w/o AE	0.8647	0.6071	0.7979
w/o PP	0.8823	0.6121	0.8207
CADReN	0.9064	0.6390	0.8641

Table 5: Ablation test: each proposed component of CADReN helps to improve the overall performance.

5.8 Ablation Tests

Additional ablation tests are carried out to evaluate the effectiveness of the mechanisms that we proposed: the **C**ontextual **A**nchor (CA), the **A**ttention-bassed **A**ggregation (AA), the **A**uto-**E**ncoder (AE) and the **P**ost-**P**rocessing mechanism (PP). We measure the performance of CADReN on RIC10K with these modules partially disabled. Experiments confirm the effectiveness of these components. Results are organized in Table 5.

6 Conclusion

In conclusion, our method is the first work to emphasize the relative relationship between a Contextual Anchor and other nodes within a Knowledge Graph using a Transformer-based architecture, while utilizing both structural and semantic information, to tackle the cross-graph Node Importance Estimation task. Our approach outperforms existing methods on cross-graph NIE setting and achieves similar performances on single-graph NIE setting. The introduction of CA enables the model to give flexible and accurate predictions.

To further enhance performance, future research could delve into the exploration of novel encoding mechanisms to generate superior embeddings. Specifically, in the case of structural embeddings, there is ample room for improvement. Neural networks, such as Graph Neural Networks, hold promise in providing more detailed structural information. However, a challenge persists in accurately representing the relative distance between the Contextual Anchor and the nodes in background graph. Addressing this issue is of utmost importance for forthcoming researches in this field.

References

Kenneth Atz, Francesca Grisoni, and Gisbert Schneider. 2021. Geometric deep learning on molecular representations. *Nature Machine Intelligence*, 3:1023 – 1032.

Kurt Bollacker, Robert Cook, and Patrick Tufts. 2007. Freebase: A shared database of structured general human knowledge. In *Proceedings of the 22nd National Conference on Artificial Intelligence - Volume 2*, AAAI'07, page 1962–1963. AAAI Press.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Xing Cao and Yun Liu. 2023. Relmkg: reasoning with pre-trained language models and knowledge graphs for complex question answering. *Applied Intelligence*, 53(10):12032–12046.

Cen Chen, Chen Liang, Jianbin Lin, Li Wang, Ziqi Liu, Xinxing Yang, Xiukun Wang, Jun Zhou, Yang Shuang, and Yuan Qi. 2020. Infdetect: a large scale graph-based fraud detection system for e-commerce insurance.

Zhikai Chen, Haitao Mao, Hang Li, Wei Jin, Hongzhi Wen, Xiaochi Wei, Shuaiqiang Wang, Dawei Yin, Wenqi Fan, Hui Liu, and Jiliang Tang. 2023. Exploring the potential of large language models (llms) in learning on graphs.

Nurendra Choudhary and Chandan K. Reddy. 2023. Complex logical reasoning over knowledge graphs using large language models.

Daniel Cummings and Marcel Nassar. 2020. Structured citation trend prediction using graph neural networks. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3897–3901. IEEE.

Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864.

Han Huang, Leilei Sun, Bowen Du, Chuanren Liu, Weifeng Lv, and Hui Xiong. 2021. *Representation Learning on Knowledge Graphs for Node Importance Estimation*, page 646–655. Association for Computing Machinery.

Clemens Isert, Kenneth Atz, and Gisbert Schneider. 2023. Structure-based drug design with geometric deep learning. *Current Opinion in Structural Biology*, 79:102548.

Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *Proceedings of the 5th International Conference on Learning Representations*, ICLR '17, pages 1–14.

Ngoc Luyen Le, Marie-Hé lène Abel, and Philippe Gouspillou. 2023. A personalized recommender system based-on knowledge graph embeddings. In *Lecture Notes on Data Engineering and Communications Technologies*, pages 368–378. Springer Nature Switzerland.

Jens Lehmann, Preetam Gattogi, Dhananjay Bhandiwad, Sébastien Ferré, and Sahar Vahdat. 2023. Language models as controlled natural language semantic parsers for knowledge graph question answering. In *ECAI 2023*, pages 1–9.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.

Xutao Li, Michael K. P. Ng, and Yunming Ye. 2012. Har: Hub, authority and relevance scores in multi-relational data for query search. In *SDM*.

Bin Liu, Shuangyan Jiang, and Quan Zou. 2018. HITS-PR-HHblits: protein remote homology detection by combining PageRank and Hyperlink-Induced Topic Search. *Briefings in Bioinformatics*, 21(1):298–308.

Weijie Liu, Peng Zhou, Zhe Zhao, Zhiruo Wang, Qi Ju, Haotang Deng, and Ping Wang. 2019. K-bert: Enabling language representation with knowledge graph. In *AAAI Conference on Artificial Intelligence*, pages 1–8.

Neo4j Company. 2012. Neo4j - the world's leading graph database.

OpenAI. 2023. Gpt-4 technical report.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. The pagerank citation ranking : Bringing order to the web. In *The Web Conference*.

Shirui Pan, Linhao Luo, Yufei Wang, Chen Chen, Jiapu Wang, and Xindong Wu. 2023. Unifying large language models and knowledge graphs: A roadmap.

Namyong Park, Andrey Kan, Xin Luna Dong, Tong Zhao, and Christos Faloutsos. 2019. Estimating node importance in knowledge graphs using graph neural networks. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery &amp Data Mining*, pages 596–606. ACM.

Namyong Park, Andrey Kan, Xin Luna Dong, Tong Zhao, and Christos Faloutsos. 2020. MultiImport: Inferring node importance in a knowledge graph from multiple input signals. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery &amp Data Mining*, pages 503–512. ACM.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2018. Language models are unsupervised multitask learners.

Huayi Tang and Yong Liu. 2023. Towards understanding the generalization of graph neural networks.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, pages 1–15. Curran Associates, Inc.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2017. Graph attention networks. *6th International Conference on Learning Representations*, pages 1–12.

Hanzhi Wang, Zhewei Wei, Junhao Gan, Sibo Wang, and Zengfeng Huang. 2020. Personalized PageRank to a target node, revisited. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM.

Xikun Zhang, Antoine Bosselut, Michihiro Yasunaga, Hongyu Ren, Percy Liang, Christopher D Manning, and Jure Leskovec. 2022. GreaseLM: Graph REASoning enhanced language models. In *International Conference on Learning Representations*, pages 1–16.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: Enhanced language representation with informative entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, Florence, Italy. Association for Computational Linguistics.

7 Appendix

A More results explained in details

A.1 Results of *Effectiveness of CA* experiment

Here we show the results of different models applied on same BG graphs while altering the CA and GT nodes in figure 5 and figure 6. The nodes marked in red (resp. orange) are the nodes contained in the GT_1 (resp. GT_2) set related to the CA_1 (resp. CA_2) nodes.

A.1.1 Comparison between the gray and blue columns

GENI and RGTN could not take CAs as input, therefore, their predictions are static and not flexible, usually including the generally "popular" nodes (e.g. PlayStation 4) or the acronyms linked to lots of nodes (e.g. DMC and 6F) but are not necessarily related to the user's interest. On the other hand, GPT-3.5 and CADReN could generate predictions according to different CAs. In figure 5, CADReN successfully gives the predictions related to *Titanium* and *Phosphorus chemicals* respectively, and in the example of figure 6, CADReN could distinguish whether the user focuses on *Chips* or *Thyristors*.

A.1.2 Comparison between light blue and dark blue columns

CADReN's predictions are more stable reasonable than the ones given by GPT-3.5. As shown in the figure 5, GPT-3.5 failed to provide a comprehensive prediction likely due to the lack of the niche knowledge of *MDI* or *Titanium dioxide*. As comparison, CADReN gives better prediction covering almost all the GT nodes among top-20 predictions because it can effectively leverage the structural information in KG as from semantic perspective, GPT-3.5 is superior than BERT.

B Prompts used during the experiments of GPT-3.5

"**role**":"system","**content**":"you are an amazing analyst". "**role**":"user","**content**":" Please select top 20 important words based on the key words from a given set of background words. For the important words, please also provide a score (0 to 1). Output should be like word \t score. Thank you.

Key words:

"' CA_1 *AND* CA_2 *AND* CA_3 "'

A set of background words:

"' BG_1, BG_2, BG_3, BG_4, BG_5, BG_6, ... "'

The CA_i and BG_j are filled with actual node entities during the experiments.

Model	GENI	RGTN	GPT3.5	CADReN	GPT3.5	CADReN
CA	No CA given	No CA given	CA 1: MDI, Titanium dioxide, Titanium concentrate, Polyurethane materials		CA 2: Iron Phosphate, Phosphorus Chemicals	
Predictions	Industrial grade monoammonium phosphate	Calcium carbide	MDI	Polyurethane Materials	Lithium iron phosphate	Iron Phosphate
	PVC resin	Iron phosphate	Titanium dioxide	Construction	Iron Phosphate	Phosphorus Chemical
	Chemical raw materials	Tianneng Chemical	Titanium concentrate	Titanium concentrate	Phosphorus Chemical	Phosphorus Ore
	Industrial Silicon	Thermal phosphoric acid	Polyurethane materials	MDI	Phosphorus compounds	Lithium iron phosphate
	New Energy	Soda ash	Construction	Titanium dioxide	Phosphorus Ore	Fluorochemicals
	Iron Phosphate	Nitric acid		Calcium carbide	Ammonium Phosphate	Phosphorus Fertilizer
	Traditional Refineries	Chemical raw materials		Chlorination titanium dioxide		Calcium carbide
	New Energy Vehicle Sales	Yellow phosphorus		Pangang Company		Chemical raw materials
	Sierpong	Phosphorus chemical		Titanium concentrate Pangang		Sanyou Chemical
	Phosphorus trichloride	Phosphate		Polyurethane		Soda ash
	No.2	Coal chemical		Wanhua Chemical		Phosphate
	DMC	Caustic soda		PVC vinyl		Tianneng Chemical
	EO/EG	Methanol		Caustic soda		Basic Chemicals
	C3	Raw material propylene		Chlor-alkali		Coal Chemical
	DMF	Raw material glue		Calcium carbide		Yellow phosphorus
	6F	Phosphorus Fertilizer		Foreign trade		Lithium iron phosphate manufacturers
	Traditional solvents	Fine Chemical		Chlor-alkali chemical		Raw material propylene
	Yuntianhua Company	Sanyou Chemical		Trichloroethylene		Chlor-alkali chemical
	Low carbon	Fluorine chemical		Logistics		Lithium hexa-fluorophosphate
	Zhongtai Chemical	Basic Chemicals		Aluminum oxide		Raw material glue

Figure 5: Results of experiment on BG No. 1608708

Model	GENI	RGTN	GPT3.5	CADReN	GPT3.5	CADReN
CA	No CA given	No CA given	CA 1: MLCC, Chips		CA 2: IGBT, Thyristor, Mosfet, Power device, Jiejie Microelectronics	
Predictions	SLP board	Electronic Circuits	MLCC	MLCC	IGBT	IGBT
	SLP	Circuit Control	Chip	Chip	Thyristor	Power Devices
	Playstation4	Battery to Motor	Semiconductor	5G	Mosfet	Jiejie Microelectronics
	PTC	Battery to Appliance	Sensor	Electronic Circuit	Power Devices	Thyristor
	PCB/	Motor Control	Battery	Automotive Electronics	Power Semiconductor Modules	Mosfet
	PCB	Photovoltaic	Circuits	Internet of Things	High Frequency High Speed Materials	Mems sensors
	PCB high level hardboard production line	Passive Components	Printed Metal Electrode Paste	Electronic Components	OLED full screen	Servers
	Playstation	Capacitors	Thyristors	Intel	Intelligent Hardware Devices	Circuit Control
	Mosfet	Wind power	Ceramic Dielectric Diaphragms	Low Loss High Frequency High Speed Materials	Photovoltaic	Internet of Things
	MLCC	Automotive Batteries	Integrated Circuits	Medium Loss High Speed Circuit Substrate	Outbreak	Intelligent
	LED	Internet of Things	Otis	Electronic label	Semiconductor	New Energy Vehicles
	Lighting port	Controllers	Rigid Board	Server Materials	High Performance	Battery Management
	OLED full screen	Battery Management	Five Wire Connectors	Automotive Battery	Sensors	UPS
	Mosfet power semiconductor module	Batteries	Capacitors	Circuit Control	Microelectronics	Photovoltaic
	OLED pressure sensitive touch screen	Electric Vehicles	Electronic Components	Printed Metal Electrode Pastes	New Energy Vehicles	Wind power
	OPPO	Integrated Circuits	Optical Devices	Smart Analytics	Chips	Grid
	FPC	IGBT	Camera Module	Servers	Internet of Things	Railroad
	Fitbit	Railroad	Coils	Controllers	Battery	Switches
	AI speaker	New Energy Vehicle	Surface Mount Capacitors	Capacitors	High-end	Electronic Circuits
	EMBB	Switch	Five-wire connectors	Logistics		Motor Controls

Figure 6: Results of experiment on BG No. 1610703