A Contextualized BERT model for Knowledge Graph Completion

Haji Gul
School of Digital Science,
Universiti Brunei Darussalam
23h1710@ubd.edu.bn

Abul Ghani Haji Naim
School of Digital Science,
Universiti Brunei Darussalam
ghani.naim@ubd.edu.bn

Ajaz Ahmad Bhat
School of Digital Science,
Universiti Brunei Darussalam
ajaz.bhat@ubd.edu.bn

Abstract

Knowledge graphs (KGs) are valuable for representing structured, interconnected information across domains, enabling tasks like semantic search, recommendation systems and inference. A pertinent challenge with KGs, however, is that many entities (i.e., heads, tails) or relationships are unknown. Knowledge Graph Completion (KGC) addresses this by predicting these missing nodes or links, enhancing the graph's informational depth and utility. Traditional methods like TransE and ComplEx predict tail entities but struggle with unseen entities. Textual-based models leverage additional semantics but come with high computational costs, semantic inconsistencies, and data imbalance issues. Recent LLM-based models show improvement but overlook contextual information and rely heavily on entity descriptions. In this study, we introduce a contextualized BERT model for KGC that overcomes these limitations by utilizing the contextual information from neighbouring entities and relationships to predict tail entities. Our model eliminates the need for entity descriptions and negative triplet sampling, reducing computational demands while improving performance. Our model outperforms state-of-the-art methods on standard datasets, improving Hit@1 by 5.3% and 4.88% on FB15k-237 and WN18RR respectively, setting a new benchmark in KGC.

1 Introduction

A knowledge graph (KG) is a structured representation of entities (as nodes) and relationships (as links) that supports search, recommendation and other downstream reasoning tasks. However, KGs are often incomplete, with many entities (heads/tails) or relationships missing, limiting their utility in real-world applications [1]. Consequently, Knowledge Graph Completion (KGC)—predicting a missing tail entity $(h, r, ?)$, head entity $(?, r, t)$, or relationship $(h, ?, t)$ in a triplet—has become a critical research objective, with numerous methodologies proposed to tackle this issue.

Embedding-based methods, for instance, learn vector embeddings for entities and relationships from training data, but these methods struggle to generalize to *unseen* entities or relationships, impairing performance in tail prediction during testing [2]. Recently, large language model (LLM)-based approaches for KGC have shown potential in overcoming this limitation by leveraging LLMs trained on extensive datasets to capture complex semantic relationships and generalize better to unseen entities [3, 10, 5, 10]. Despite these strengths, LLM-based models are computationally demanding, often overlook relation context, and depend heavily on entity descriptions and negative sampling.

Submitted to 38th Conference on Neural Information Processing Systems (NeurIPS 2024). Do not distribute.

More recent LLM prompting-based approaches encode KGs into prompts[10], but injecting all relevant facts from a KG into prompts is labor-intensive, and generic LLMs often struggle with domain-specific KGs. Additionally, textual information-based methods like NN-KGC and Sim-KGC utilize neighborhood information for KGC, but they often require entity descriptions, which may not be available in many datasets, and add computational overhead [6, 7]. To address these limitations, we propose a Context-Aware BERT for Knowledge Graph Completion (CAB-KGC) that extracts contextual information associated with the operational relationship, its neighboring entities, and relationships associated with the head entity. This context is then integrated with the BERT model to enhance the prediction of tail entities. To summarise, this study makes the following contributions to the KG domain:

- We introduce the CAB-KGC approach to address the KGC problem, leveraging graph features of head entity context and relationship context and the BERT model. The CAB-KGC approach outperforms SOTA KGC methods.
- CAB-KGC eliminates reliance on entity descriptions, focusing solely on head and relationship contexts for improved predictions available in all KGs.
- CAB-KGC does not require negative sample training, enhancing training speed and resilience against negative sample selection.
- Extensive experiments on various benchmark datasets demonstrate that CAB-KGC reliably excels in tail entity prediction.

2 Methodology

Problem Formulation (see Table 1 for notations): Consider a knowledge graph $G(E, R)$ as a collection of triplets (h, r, t), where $h \epsilon E$ is the head entity, $t \epsilon E$ is the tail entity, and $r \epsilon R$ represents the relationship between them, our CAB-KGC model predicts a missing tail t (represented by ?) given an incomplete triple $(h\ ,\ r\ ,\ ?)$.

Table 1: Mathematical Notations and Symbols

Notation	Description	Notation	Description
e	entity or node	r	relationship
h	head entity node	t	tail entity node
E	Entities Set	R	Relationships Set
H_c	Head (h) or Entity context	R_c	Relationship context
$R(h), E(h)$	relation and entities associated to head h	N_T	Total number of triplets
$p_\theta(t_i \mid h_i, r_i)$	Tail t_i probability given head h_i and relationship r_i	rank$_i$	Rank of the true tail entity t_i in the prediction
$\downarrow$	Results: Lower is better	$\uparrow$	Results: Higher is better

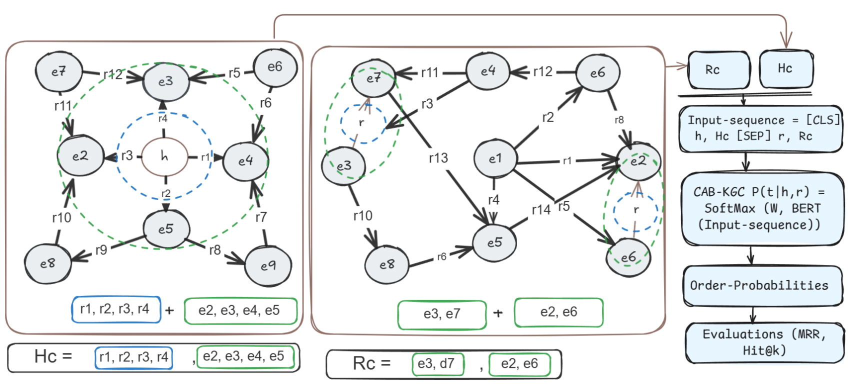

Figure 1: A concise view of the CAB-KGC Method. Box on the left shows head context H_c calculation; the middle one shows relationship context R_c calculation. H_c and R_c are then fed into the model pipeline shown on the right side.

Figure 1 provides an overview of the CAB-KGC model. It predicts the tail entity t given a head h and a relationship r, in the following steps:

1. **Extract Head Context** H_c: To extract the contextual information for the head i.e. H_c, we first identify the relationships r that are associated with the head entity h, i.e., $R(h)$. If k

relationships are associated with the head h from the set R of all relationships r_i in the graph G, then:

$$\mathcal{R}(h) = A_{i=1}^{k}\left(\{r_i \mid (h, r_i, e_j) \in T,\, e_j \in E\}\right) \tag{1}$$

Next, we find the entities e that are neighbours (have a direct connection) with the head entity h, i.e., $E(h)$ using the identified relationships $R(h)$.These neighbour entities can be mathematically expressed as:

$$\mathcal{E}(h) = A_{i=1}^{m}\left(\{e_i \mid (h, r_j, e_i) \in T,\, r_j \in R\}\right) \tag{2}$$

The head context H_c is then calculated as the union of the connected relationships $R(h)$ and the neighbour entities $E(h)$, as shown below in Equation 3.

$$H_c = (\mathcal{R}(h)) \cup (\mathcal{E}(h)) \tag{3}$$

2. **Extract Relationship Context** R_c: To acquire the relationship context R_c, we identify all the entities associated with the operational relationship r in the knowledge Graph G. R_c is given as:

$$R_c = A_{i,j=1}^{l}\left(\{e_i, e_j \mid (e_i, r, e_j) \in T\}\right) \tag{4}$$

3. **Prepare Input Sequence for BERT Classifier**: The contextual information extracted in the above steps forms the input to BERT. Specifically, the input sequence contains h, H_c from Equation 3, r, and R_c from Equation 4, as shown below:

$$\text{Input Sequence} = [CLS]\; h,\; H_c\; [SEP]\; r,\; R_c \tag{5}$$

where [CLS] is BERT's classifier token and [SEP] is the separator token.

4. **Predict and train with BERT Classifier**: A classification layer is added on top of the BERT model, which aims to classify the tail entity $(h\ ,\ r\ ,\ ?)$. Once the BERT classifier receives the input, it processes it through various transformer layers, provides a contextualized representation of each token and uses that to classify the input. The classifier model predicts the tail entity by employing a softmax function over the output embedding to calculate the probability for all the available tail entities. The input-output description of the model is given as:

$$P(t \mid h, r) = \text{softmax}(W \cdot\ \text{BERT(Input Sequence)}) \tag{6}$$

Where W is a learned weight matrix. Putting the above equations together, the CAB-KGC model can be expressed as:

$$\text{CAB-KGC}(t \mid h, r) = \text{softmax}(W \cdot\ \text{BERT}(h, H_c, r, R_c)) \tag{7}$$

The CAB-KGC model is trained using cross-entropy loss, which compares the probability distribution of the predicted label with the true label for the tail entity. The cross-entropy loss is given by:

$$L = -\sum_{i=1}^{N} y_i \log P(t_i \mid h, r) g \tag{8}$$

In this equation the one-hot encoded true label for the tail object t_i is indicated as y_i. The predicted probability for the true tail entity could be denoted as $P(t_i \mid h, r)$, where h is the head and r is the relation.

2.1 Experiments Setup

Datasets: We assessed the proposed CAB-KGC model on various commonly used KG datasets. These datasets are briefly explained here:

- FB15k-237 [23] is an updated version subset of the FB15k dataset, where the inverse triplets have been removed to increase the difficulty of the KGC. It has 14541 unique entities and 237 relationships.
- WN18RR [24] is the subset of WN18, where the reverse triplets are removed, making it more complex for the models to incorporate the problem of KGC.

Hyperparameters: The experiments used a batch size of 16 and a learning rate of 5e-5, Adam as the optimizer and cross-entropy as the loss function. The experiments were accomplished on an NVIDIA GeForce RTX 3090 GPU with 24 GB of memory. Training for the CAB-KGC model was halted once evaluation metrics stabilized to the third decimal place.

Evaluation: Various standard evaluation metrics in KGC, as given in Equation 9, such as MRR, and Hit@k, are utilized to assess the performance of the proposed method and other state-of-the-art approaches.

$$\text{MRR} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{\text{rank}_i} \quad ; \quad \text{Hits@k} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{1}(\text{rank}_i \leq k) \tag{9}$$

where rank_i is the correct entity rank position in the descending order sorted list of predicted scores for the i-th triplet. The function $\mathbf{1}(\text{rank}_i \leq k)$ is a ranking function that outputs one if the true entity is ranked within the top k predictions and 0 otherwise.

2.2 Results

Our CAB-KGC approach shows superior results on the FB15k-237 dataset. CAB-KGC's significant performance is its Hits@1 score of 0.322, which improves SOTA by almost 5.3%, showing a superior ability to rank accurate entities in the first place. It obtains a Hits@3 score of 0.399 and improves by 0.5%, notably above other models, indicating that CAB-KGC reliably predicts relevant entities within the top 3 ranks. The CAB-KGC method performed well on the WN18RR dataset, getting an MRR of 0.685, which is an improvement of 1.2% over SOTA models and a Hits@1 of 0.637, an improvement of 4.88%, outperforming state-of-the-art methods. All the results are reported in Table 2. Note that we have excluded results from KICGPT[10] during comparison for reasons: (a) the model is prompt-based and not trainable, (b) its performance is highly dependent on the underlying LLM's knowledge base and (c) large KGs cannot be injected as prompts to this model. Furthermore, injecting all relevant facts from different KGs into prompts is labor-intensive, and underlying LLM will oftern struggle with domain-specific KGs when it does not contain enough relevant knowledge.

Table 2: Comparison of the proposed and baseline methods on the datasets FB15k-237 and WN18RR. The optimal outcome for each metric is highlighted in bold, while the second-best result is underlined. The circle symbol ○ denotes that the results have been extracted from the study conducted by Wei et al. [8], while the symbol □ indicates that the results have been extracted from the study conducted by Yao et al. in [3].

Dataset	**FB15k-237**			**WN18RR**		
Methods	**MRR ↑**	**Hits@1 ↑**	**Hits@3 ↑**	**MRR ↑**	**Hits@1 ↑**	**Hits@3 ↑**
Embedding-Based Methods						
RESCAL [11] ○	0.356	0.266	0.390	0.467	0.439	0.478
TransE [12] ○	0.279	0.198	0.376	0.243	0.043	0.441
DistMult [13] ○	0.241	0.155	0.263	0.430	0.390	0.440
ComplEx [14] ○	0.247	0.158	0.275	0.440	0.410	0.460
RotatE [15] ○	0.338	0.241	0.375	0.476	0.428	0.492
TuckER [16] ○	**0.358**	0.266	0.394	0.470	0.443	0.482
CompGCN [17] ○	0.355	0.264	0.390	0.479	0.443	0.494
HittER [18] ○	0.344	0.246	0.380	0.496	0.449	0.514
HAKE [19] ○	0.346	0.250	0.381	0.497	0.452	0.516
Text-and Description-Based Methods						
Pretrain-KGE [5] ○	0.332	-	-	0.235	-	-
StAR [20] ○	0.263	0.171	0.287	0.364	0.222	0.436
MEM-KGC (w/o EP) [21] ○	0.339	0.249	0.372	0.533	0.473	0.570
MEM-KGC (w/ EP) [21] ○	0.346	0.253	0.381	0.557	0.475	0.604
SimKGC [6] ○	0.333	0.246	0.363	0.671	0.585	**0.731**
NNKGC [22] ○	0.338	0.252	0.365	0.674	0.596	0.722
LLM-Based Methods						
ChatGPTzero-shot [5] □	-	0.237	-	-	0.190	-
ChatGPTone-shot [5] □	-	0.267	-	-	0.212	-
KICGPT [5] □	0.410	0.321	0.430	0.564	0.478	0.612
Proposed						
Proposed CAB-KGC	0.350	**0.322**	0.399	**0.685**	**0.637**	0.687

3 Conclusion

The proposed CAB-KGC approach exploits the contexual information to outperform existing methods in MRR and Hit@k measures, with improvements of 5.3% and 4.88% over FB15k-237 and WN18RR, respectively.

References

[1] Ding, J. and Jia, W. (2018) The research advances of knowledge graph completion algorithm. *Information and Communication Technology* **1**:56–62.

[2] Xie, R., Liu, Z., Jia, J., Luan, H., and Sun, M. (2016) Representation learning of knowledge graphs with entity descriptions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 30, No. 1.

[3] Yao, L., Mao, C., and Luo, Y. (2019) KG-BERT: BERT for Knowledge Graph Completion. *arXiv preprint* arXiv:1909.03193. Available at: https://arxiv.org/abs/1909.03193.

[4] Wei, Y., Huang, Q., Zhang, Y., and Kwok, J. (2023) KICGPT: Large Language Model with Knowledge in Context for Knowledge Graph Completion. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 8667–8683. Association for Computational Linguistics. Available at: http://dx.doi.org/10.18653/v1/2023.findings-emnlp.580.

[5] Zhang, Z., Liu, X., Zhang, Y., Su, Q., Sun, X., and He, B. (2020) Pretrain-KGE: Learning Knowledge Representation from Pretrained Language Models. In T. Cohn, Y. He, and Y. Liu (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 259–266. Association for Computational Linguistics. Available at: https://aclanthology.org/2020.findings-emnlp.25.

[6] Wang, L., Zhao, W., Wei, Z., and Liu, J. (2022) SimKGC: Simple Contrastive Knowledge Graph Completion with Pre-trained Language Models. *arXiv preprint* arXiv:2203.02167. Available at: https://arxiv.org/abs/2203.02167.

[7] Li, D., Tan, Z., Chen, T., and Liu, H. (2024) Contextualization Distillation from Large Language Model for Knowledge Graph Completion. *arXiv preprint* arXiv:2402.01729. Available at: https://arxiv.org/abs/2402.01729.

[8] Wei, Y., Huang, Q., Kwok, J.T., and Zhang, Y. (2024) Kicgpt: Large language model with knowledge in context for knowledge graph completion. *arXiv preprint* arXiv:2402.02389.

[9] Anonymous Authors. (2025) CAB-KGC: Context-Aware BERT for Knowledge Graph Completion. *Submitted to ICLR 2025 for consideration*

[10] Wei, Y., Huang, Q., Zhang, Y., and Kwok, J. (2023) KICGPT: Large language model with knowledge in context for knowledge graph completion. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 8667–8683. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.580.

[11] Nickel, M., Tresp, V., and Kriegel, H.-P. (2011) A three-way model for collective learning on multi-relational data. In *Proceedings of the 28th International Conference on Machine Learning*, pp. 809–816. Omnipress. Madison, WI, USA.

[12] Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., and Yakhnenko, O. (2013) Translating embeddings for modeling multi-relational data. In C.J. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger (eds.), *Advances in Neural Information Processing Systems*, Vol. 26. Curran Associates, Inc.

[13] Yang, B., Yih, W.-t., He, X., Gao, J., and Deng, L. (2014) Embedding entities and relations for learning and inference in knowledge bases. In *International Conference on Learning Representations*. Available at: https://api.semanticscholar.org/CorpusID:2768038.

[14] Trouillon, T., Welbl, J., Riedel, S., Gaussier, É., and Bouchard, G. (2016) Complex embeddings for simple link prediction. *arXiv preprint* arXiv:1606.06357. Available at: https://arxiv.org/abs/1606.06357.

[15] Sun, Z., Deng, Z.-H., Nie, J.-Y., and Tang, J. (2019) RotatE: Knowledge graph embedding by relational rotation in complex space. *arXiv preprint* arXiv:1902.10197. Available at: https://arxiv.org/abs/1902.10197.

[16] Wang, Y., Broscheit, S., and Gemulla, R. (2019) A relational Tucker decomposition for multi-relational link prediction. *arXiv preprint* arXiv:1902.00898. Available at: https://arxiv.org/abs/1902.00898.

[17] Vashishth, S., Sanyal, S., Nitin, V., and Talukdar, P. (2020) Composition-based multi-relational graph convolutional networks. *arXiv preprint* arXiv:1911.03082. Available at: https://arxiv.org/abs/1911.03082.

[18] Chen, S., Liu, X., Gao, J., Jiao, J., Zhang, R., and Ji, Y. (2020) Hitter: Hierarchical transformers for knowledge graph embeddings. *arXiv preprint* arXiv:2008.12813.

[19] Zhang, Z., Cai, J., Zhang, Y., and Wang, J. (2022) Learning hierarchy-aware knowledge graph embeddings for link prediction. *arXiv preprint* arXiv:1911.09419. Available at: https://arxiv.org/abs/1911.09419.

[20] Wang, B., Shen, T., Long, G., Zhou, T., Wang, Y., and Chang, Y. (2021) Structure-augmented text representation learning for efficient knowledge graph completion. In *Proceedings of the Web Conference 2021*, pp. 1737–1748.

[21] Choi, B., Jang, D., and Ko, Y. (2021) MEM-KGC: Masked entity model for knowledge graph completion with pre-trained language model. *IEEE Access* **9**, 132025-132032. Available at: https://api.semanticscholar.org/CorpusID:238243180.

[22] Li, I., and Yang, B. (2023) NNKGC: Improving knowledge graph completion with node neighborhoods. *arXiv preprint* arXiv:2302.06132. Available at: https://arxiv.org/abs/2302.06132.

[23] Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008) Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, pp. 1247–1250.

[24] Miller, G. A. (1995) WordNet: a lexical database for English. *Communications of the ACM* **38**(11):39–41.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.

- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not

including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [No]

    Justification: In our approach, we build upon existing work by utilizing the BERT model, incorporating the contextual information of both the head and tail entities. This allows us to leverage richer contextual cues from both ends of the relationship, enhancing the model's ability to understand and capture the nuances in the data.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.