MuCAL: Contrastive Alignment for Preference-Driven KG-to-Text Generation

Yifei Song and **Claire Gardent**
CNRS/LORIA and Université de Lorraine
{yifei.song, claire.gardent}@loria.fr

Abstract

We propose **MuCAL** (*Multilingual Contrastive Alignment Learning*) to tackle the challenge of Knowledge Graphs (KG)-to-Text generation using preference learning, where reliable preference data is scarce. MuCAL is a multilingual KG/Text alignment model achieving robust cross-modal retrieval across multiple languages and difficulty levels. Building on MuCAL, we automatically create preference data by ranking candidate texts from three LLMs (Qwen2.5 , DeepSeek-v3, Llama-3). We then apply Direct Preference Optimisation (DPO) on these preference data, bypassing typical reward modelling steps to directly align generation outputs with graph semantics. Extensive experiments on KG-to-English Text generation show two main advantages: (1) Our KG/Text alignment model provides a better signal for DPO than similar existing metrics, and (2) significantly better generalisation on out-of-domain datasets compared to standard instruction tuning. Our results highlight MuCAL's effectiveness in supporting preference learning for KG-to-English Text generation and lay the foundation for future multilingual extensions. Code and data are available at https://github.com/MeloS7/MuCAL_DPO/tree/main.

1 Introduction

Knowledge graphs (KG) and their verbalization into natural language play a pivotal role in bridging symbolic AI with human-centric applications (Schneider et al., 2022). While KG-to-Text generation has seen advancements through encoder-decoder architectures (Clive et al., 2022; Castro Ferreira et al., 2020) and large language model (LLM) fine-tuning (Warczyński et al., 2024; Cripwell et al., 2023a), critical challenges persist: (1) **Generalization to unseen graphs**. Fine-tuned models struggle to generate accurate text when faced with graphs containing properties or entities unseen at training time (Nikiforovskaya and Gardent, 2024); (2)

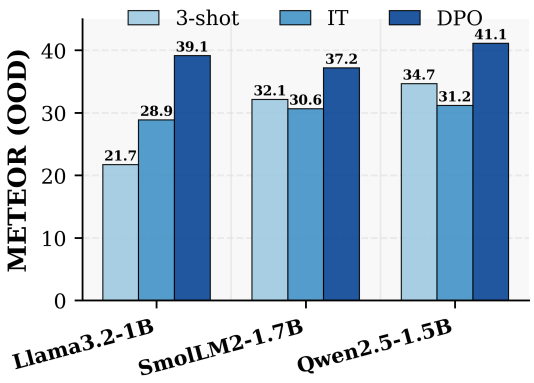

Figure 1: **Comparing instruction tuning (IT), 3-shot prompting and DPO-training.** For all three LLMs (Llama3.2-1B-Instruct, SmolLM2-1.7B-Instruct, Qwen2.5-1.5B-Instruct), DPO outperforms IT and 3-shot prompting on out of domain data (GOLD-OOD-472)

Creating **high-quality aligned (graph, text) data** is labor-expensive (Gardent et al., 2017; Shimorina et al., 2019)[1] and eliciting preference data for fine-grained optimization is even more costly; (3) **Existing reference-less metrics** often capture limited aspects of factual correctness and their ability to capture human preferences remains underexplored (Deutsch et al., 2022).

To tackle these challenges, we propose a *preference-driven* framework that targets *out-of-domain* data. Our key insight is that *preference learning* (Im and Li, 2024) – optimizing model outputs based on pairwise or ranked comparisons – can significantly improve a system's ability to handle unseen data by focusing on *factual correctness* rather than relying solely on fixed reference texts. However, to leverage preference learning effectively, we need a way to *construct preference data* without human labeling. We address this need by introducing **MuCAL**, a multilingual cross-modal

[1]It is also costly. The creation of the WebNLG 2017 data cost 7K euros.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 14239–14282
November 4-9, 2025

KG/Text alignment model trained via contrastive learning to map graphs and texts into a shared semantic space. MuCAL not only provides robust KG/Text representations but also serves as the backbone for our *automated preference construction* pipeline, where we generate multiple candidate texts (via diverse LLMs) and rank them based on MuCAL's KG/Text similarity scores.

Building on these automatically constructed preference pairs, we then apply **Direct Preference Optimization (DPO)** (Rafailov et al., 2024) to fine-tune an LLM, directly aligning outputs with the underlying graph semantics. By integrating MuCAL-based ranking with DPO, the model better captures factual correctness and improves out-of-domain generalization, all without requiring manually curated preference annotations.

Our contributions can be summarised as follows.

- **MuCAL for Cross-Modal Alignment.** We provide a novel multilingual KG/Text encoder that robustly aligns Knowledge Graphs with text across six natural languages, enabling automated preference scoring.
- **Automated Preference Data Construction.** We introduce an LLM-driven pipeline that generates preference triplets in the form of (graph, *chosen text*, *rejected text*) from unlabeled KGs and diverse model outputs, bypassing the need for expensive human-annotated preference data.
- **Preference-Driven Optimization.** We demonstrate how DPO, fueled by MuCAL-based preference data, significantly improves factual accuracy and out-of-domain KG-to-English Text generation over standard instruction tuning and few-shot prompting (cf. Figure 1).

2 Related Work

We review relevant literature from KG-to-Text Generation, Cross-Modal Alignment, and Preference Learning for Natural Language Generation (NLG).

KG-to-Text Generation The task of generating natural language text from knowledge graphs has been extensively studied. Some work fine-tuned encoder-decoder architectures, such as T5 (Raffel et al., 2023) and BART (Lewis et al., 2019) on the WebNLG dataset of aligned KG/Text pairs, achieving state-of-the-art performance on various KG-to-Text generation benchmarks (Cripwell et al. 2023b; Castro Ferreira et al. 2020). However, their generalization capability across diverse datasets and languages remains limited, primarily due to reliance on domain-specific fine-tuning and the difficulty of obtaining large-scale, multilingual aligned datasets (Nikiforovskaya and Gardent, 2024). In contrast, we investigate how a KG-to-English text model fine-tuned on silver data can be optimised using preference learning.

Cross-Modal Alignment Researchers have explored various methods for cross-modal alignment, with contrastive learning emerging as a prominent technique (Zhang et al., 2021). In particular, in-batch contrastive learning has been effectively leveraged to align features from different modalities by bringing related pairs closer in the embedding space and pushing unrelated pairs apart (Tang et al., 2022). Previous work (Scao and Gardent, 2023) has shown its feasibility for KG-to-Text alignment, but only for English. We extend on this work by training on multilingual data, improving KG/Text retrieval and successfully employing our model to create preference data for DPO.

Preference Learning for NLG Preference learning has gained traction in NLG, particularly through methods like Reinforcement Learning from Human Feedback (RLHF) (Li et al., 2023). In various domains (including machine translation and summarization), RLHF has been shown to improving text quality by aligning model outputs with human preferences (Lai et al., 2023). However, the application of preference learning in KG-to-Text generation has not been explored. A significant challenge is the scarcity of reliable preference data, which is crucial for training models to generate text that faithfully represents the underlying graph structures. We address this challenge by introducing a framework that automatically constructs high-quality preference pairs, thereby facilitating the application of preference learning techniques to KG-to-Text generation.

3 Multilingual Cross-Modal Alignment Learning

Following previous work on text/text (Reimers and Gurevych, 2019) and text/image (Gandelsman et al., 2024) alignment, we train bi- and cross- encoders that map graphs and texts to a shared semantic space. Briefly, a bi-encoder is a siamese

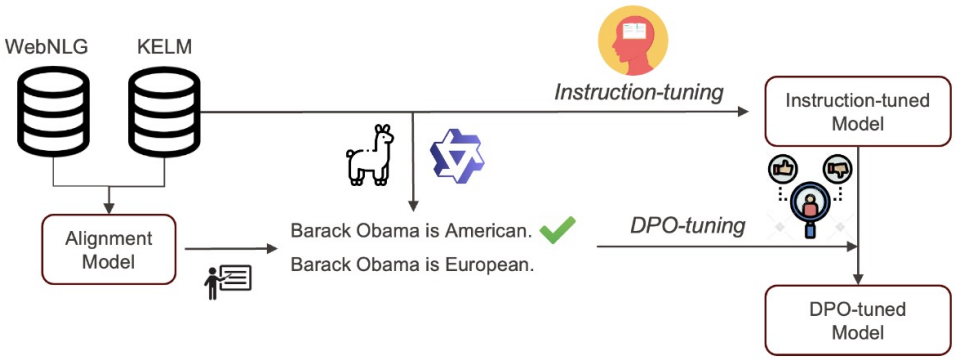

Figure 2: **MuCAL-powered DPO training pipeline. Step 1: Multilingual augmentation.** WebNLG and KELM texts are automatically translated into five additional languages, yielding a six-language silver corpus. **Step 2: MuCAL training.** We train a bi-encoder on this corpus with in-batch contrastive loss, producing the *multilingual alignment model*. **Step 3: Reference LLM.** A base LLM (e.g. QWEN2.5-1.5B-INSTRUCT) is instruction tuned on KELM-Q1 to obtain the reference policy π_{ref}. **Step 4: Preference construction & DPO.** Diverse LLMs generate candidate texts for each graph; MuCAL ranks them and extracts (G, t_C, t_R) triplets, which we use to DPO-tune the training policy π_θ, yielding the final DPO-MUCAL model.

network (two encoders with tied weights) which takes graph and text as separate input and applies mean-pooling to create fixed sized embeddings for each input. Similarity is the cosine score on these embeddings. By contrast, a cross-encoder attends to both items simultaneously and feed into a dense layer followed by a sigmoid activation function. This produces a matching score between 0 and 1, representing the degree of alignment between the text and the graph. See Appendix D for a more detailed explanation of the difference between a cross- and a bi-encoder.

3.1 Data

We construct our training and test sets using data from the WebNLG (Ferreira et al., 2020) and KELM (Agarwal et al., 2021) corpora.

Source Datasets. The WebNLG dataset consists of manually aligned Knowledge Graphs and their corresponding English textual descriptions. We use both its training (WEBNLG-TRAIN, 14,878 pairs) and its test set (WEBNLG-TEST, 1,779 pairs). The KELM dataset is a large-scale silver dataset of KG/English text pairs extracted from Wikidata and Wikipedia using distant supervision and text generation. We employ two subsets: (1) KELM-Q1, a filtered version containing 18,723 pairs based on semantic matching, and (2) KELM-TEST, a manually validated subset with 3,437 pairs (Nikiforovskaya and Gardent, 2024). The creation of KELM-Q1 is explained in detail in Appendix C.

Training Data. We merge WebNLG-Train and KELM-Q1 to form our English training set (EN-TRAIN). We then machine-translate the English texts into five target languages: Arabic, Chinese, French, Russian, and Spanish, creating the MULTI-TRAIN-SILVER set. Appendix C explains which Machine Translation (MT) models was selected for each language and why.

Test Data. We create three datasets to test retrieval in different scenarios.

MULTI-TEST-1K. A balanced subset of 1K KG/English text pairs sampled from the WebNLG and KELM test sets, ensuring diversity in graph properties and sizes (1–5 triples). All texts are translated into five languages. Additional details on its construction are given in Appendix F. We use this dataset to test the ability of our models to retrieve the correct item from a set of diverse KG/-Text pairs with little overlap in terms of properties and entities.

MULTI-WEBNLG-TEST. This dataset contains the 1,779 graphs of the WebNLG test set for English with, for each graph, one of the WebNLG English verbalisations together with its translations into our 5 target languages. This dataset is more challenging as the WebNLG graphs often share multiple properties or entities making it more difficult to identify the correct KG/Text at retrieval time.

MULTI-TEST-1K-CORR. An extension of Multi-Test-1K designed to evaluate robustness to KG/Text misalignments. Each original text is

paired with its matching graph and five corrupted graphs generated through operations such as *removing*, *adding*, or *swapping* triples, as well as *replacing* predicates or entities. As detailed in Appendix H, we use various heuristics to maintain a high level of similarity between correct and corrupted graphs. This dataset is the most challenging as it requires retrieval to distinguish between closely related graphs.

Table 4 summarises the dataset's statistics.

3.2 Contrastive Learning

We train our encoder models using contrastive learning with in-batch negatives. For each graph in a batch, we have six aligned texts in different languages, forming multiple positive pairs. We generate all possible KG/Text pairs within the batch, treating mismatched pairs as negatives.

Loss Function. Our contrastive loss function is inspired by the Soft Nearest Neighbor (SNN) Loss (Frosst et al., 2019), which accounts for multiple positive and negative samples jointly. This approach accommodates the multilingual setting, where aligned texts in different languages may vary in quality and similarity. The modified contrastive loss is defined as

$$l = -\sum_{i \in I} \log \left(\frac{\exp\left(\sum_{lg \in L} \text{sim}(\text{t}_i^{lg}, \text{g}_i)/\tau\right)}{\sum_{j \in I} \exp\left(\sum_{lg \in L} \left(\text{sim}(\text{t}_i^{lg}, \text{g}_j)/\tau\right)\right)} \right) \tag{1}$$

with I the set of training instances in the batch, L the set of target languages and t_i^{lg} the text of instance i in language lg. For the Bi-Encoder, the similarity function $\text{sim}(\cdot, \cdot)$ is the cosine similarity between graph and text embeddings. For the Cross-Encoder, it is the alignment score output by the model. The loss also includes a temperature τ, which controls the sharpness of the distribution. We discuss its choice in Appendix E.2.

Hard Negatives For the bi-encoder, we also experiment with two types of hard negatives. Confounders are constructed from the correct graph by corrupting a triple inside that graph either by swapping arguments (inverting subject and object in a triple) or by substituting a property for a property different from, but related to that property. See Section H for more details.

In practice, we pair each text with one correct graph, several in-batch negatives, and a set of hard negatives. This combined approach ensures that the model is exposed to both easy and challenging negative samples during training, thereby promoting greater robustness and discriminative capability.

Note that for the cross-encoder, we do not add hard negatives as the self-attention mechanism, which is applied to the concatenation of a graph and a text, has a time- and space-complexity that is quadratic in the length of the input, which makes computation over large batches untractable.

3.3 Encoder Model, Variants and Baselines

We compare two encoders for multilingual graph–text alignment: a **bi-** (BE-MPNet) and a **cross-encoder** (CE-MPNet), both are initialised from the multilingual MPNet[2] text encoder. Implementation and architecture details are deferred to Appendix D.

Model Variants By default the batch size is 32 (BE-MPNet). We also investigate batch size 8 (BE-MPNet (bs8)) and 16 (BE-MPNet (bs16)). For the bi-encoder, we experiment with 1 (BE-MPNet-Hard1), 2 (BE-MPNet-Hard2) and 4 (BE-MPNet-Hard4) hard negatives for each graph in a batch.

Baselines. We compare our models to five baselines: MPNET, the multilingual sentence embedding model we use to initialise our graph-text alignment model; BGE-M3, the current multilingual SOTA embedding model for text[3]; ERE-DAT, a SOTA KG/English text alignment model on retrieval[4]; FACTSPOTTER, the SOTA model on KG/English text alignment under human evaluation[5]; and CLS-MPNET, the MPNet model trained as a binary classifier on the same Multi-Train-Silver data we use to train our alignement model (See Appendix G for details). This latter baseline is to assess the impact of contrastive learning compared to standard teacher training[6].

3.4 Evaluation

We evaluate our cross-modal alignment models through retrieval (Given a KG/text, how well can the model identify the matching text/graph?) and using Mean Reciprocal Rank (MRR) and Recall at

[2]https://huggingface.co/sentence-transformers/paraphrase-multilingual-mpnet-base-v2

[3]https://huggingface.co/BAAI/bge-m3

[4]https://huggingface.co/teven/bi_all_bs192_hardneg_finetuned_WebNLG2017

[5]https://huggingface.co/Inria-CEDAR/FactSpotter-DeBERTaV3-Base

[6]We also experimented with using BGE-M3 to initialise our alignment models, but the results were not significantly different while the training times were much longer as BGE-M3 is much larger (567M parameters) than MPNet (278M).

one (R@1), two metrics which measure a model's ability to correctly rank a set of candidates.

3.5 Results

Retrieval results for all models, languages, test sets and retrieval tasks (mono- and multi-lingual, Graph-to-Text and Text-to-Graph) are reported in Sections L, M and O. Here we focus on English only as this is the language we consider for DPO training. Table 1 summarizes the results.

Improvement over the baselines. As mentioned in Section 3.1, the graphs in Multi-Test-1K have limited property and entity overlap, which facilitates retrieval. On this dataset, the SOTA text (BGE-M3) and KG/English Text (EREDAT) encoders perform competitively. However, on more challenging test sets (Multi-WebNLG-Test, which contains graphs with higher overlap in terms of properties and entities, and Multi-Test-1K-Corr, which includes deliberately corrupted graphs), our models bring significant improvements over the four baselines. This demonstrates the effectiveness of our approach over simply using textual embeddings (BGE-M3) or a non-contrastive learning method (CLS-MPNet). It also shows that, although multilingual, our models outperform a state-of-the-art monolingual KG/Text encoder optimised for English (EREDAT).

Differences between testsets. All models follow a similar trend: results are best on Multi-Test-1K, second best on Multi-WebNLG-Test and third best on Multi-Test-1K-Corr, confirming the increasing complexity of the three datasets.

In-Batch vs. Hard Negatives. Integrating hard negatives into the training data drastically improves results on the hardest test set while maintaining reasonable performance on the other two, the latter likely resulting from the fact that a model trained to detect mismatches due to corrupted graphs underperforms on a test set which does not contain any.

Number of Hard Negatives. Using 2 hard negatives (rather than 1 or 4) yields the best results. Empirically, we also found that negative graphs including swapped arguments (inverting subject and object in a triple) or substituted properties (substituting a property in a triple for a property different but related to the initial property) brings best results (See Sec. H for a more detailed exploration of this point). We hypothesize that the other types of corruption (adding or removing a triple, substituting an entity) need not be added as they might already be present in the in-batch negatives: a batch may contain a super- or a sub-graph of another graph present in the batch; similarly it might contain two graphs that only differ on one entity.

Batch Size. As already shown in e.g., (Qu et al., 2021), batch size has an important impact with all our best bi-encoder models having the maximum batch size (32) we could explore given our computational resources.

Impact of Multilingual Training. The comparison with EREDAT (a monolingual KG/English text alignment model) suggests that training on multilingual data helps improve results, as shown in Fig. 3. To disentangle the effect of data size from language coverage, we introduce the *All-6-Small* setting, which matches the training size of the English-only subset but distributes it across all six languages. *All-6-Small* still significantly outperforms English-only, showing that the gains come from multilingual coverage rather than simply larger data volume. (Details in App. P)

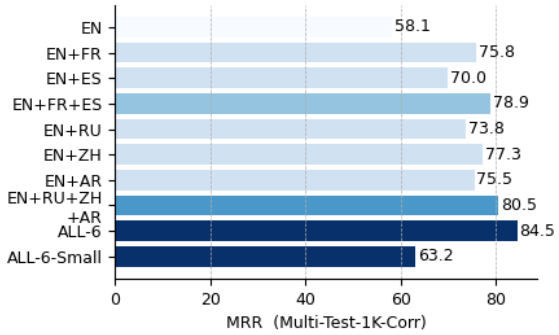

Figure 3: **Effect of language coverage on English retrieval quality.** Mean reciprocal rank (MRR) on MULTI-TEST-1K-CORR for BE-MPNET-HARD2 trained with progressively larger language subsets (one bar per subset; darker shade = more languages). *All-6* denotes the full six-language mix: English (EN), Arabic (AR), Chinese (ZH), French (FR), Spanish (ES), and Russian (RU). *All-6-Small* matches the training size of the English-only setting, but covers all six languages, thereby controlling for data volume.

Model Selection for our DPO experiment. To create preference data, we need a model able to assess the quality of both good and bad texts with respect to a graph where a bad text might diverge

Model Variants	Multi-Test-1K				Multi-WebNLG-Test				Multi-Test-1K-Corr	
	G2T		T2G		G2T		T2G		T2G	
	R@1	MRR	R@1	MRR	R@1	MRR	R@1	MRR	R@1	MRR
Models Selected for Preference Learning										
BE-MPNet-Hard2	95.60	97.30	96.10	97.40	80.33	86.92	81.62	87.65	**73.50**	**84.55**
CE-MPNet (bs4)	96.40	97.51	96.60	97.53	**85.39**	**90.52**	**86.23**	**91.20**	24.10	55.30
Baselines										
MPNet	83.20	88.98	83.20	89.16	43.28	57.17	39.91	54.67	25.00	50.25
CLS-MPNet	91.10	94.00	91.60	94.32	65.99	76.61	62.06	74.97	29.10	57.05
BGE-M3	92.90	96.09	96.00	97.77	70.49	80.55	80.04	87.69	45.90	68.53
EREDAT	95.20	97.10	96.50	98.01	76.67	84.65	82.91	89.46	41.00	66.54
FactSpotter	71.10	80.74	67.70	80.52	38.90	55.46	37.27	56.90	32.70	55.77
Batch Size Variants										
BE-MPNet (bs8)	95.70	97.53	96.10	97.79	79.60	86.61	81.06	88.04	41.90	65.66
BE-MPNet (bs16)	**96.60**	**98.14**	**97.60**	**98.69**	82.18	88.37	83.08	89.50	43.40	67.38
BE-MPNet (bs32)	96.10	97.66	**97.60**	**98.69**	83.53	89.34	84.94	90.68	46.40	69.53
Hard Negative Variants										
BE-MPNet-Hard1	95.00	96.99	96.70	97.90	79.26	86.33	81.84	88.05	69.90	82.75
BE-MPNet-Hard4	94.90	96.85	94.20	96.11	78.70	85.61	78.81	85.77	69.60	81.76

Table 1: **Retrieval Results on monolingual Data (English Texts)**. BE, CE: Bi- and Cross-Encoder, G2T, T2G: Graph-to-Text and Text-to-Graph Retrieval, R@1: Recall@1, MRR: Mean Reciprocal Rank. Unless specified otherwise (bs8, bs16, bs32), the batch size is 4 for the cross-encoder and 32 for the bi-encoders. HardX indicates the number (X) of hard negatives per graph.

from the graph by adding, omitting or substituting content. The above results suggest that hard negatives are necessary for substitution cases but that the corresponding models slightly underperform on the other cases. We therefore consider and compare two models for the creation of preference data: HARD-MUCAL, our bi-encoder variant (BE-MPNet-Hard2) which achieves remarkable performance (Acc=73.50, MRR=84.55) on the hardest test set (Multi-Test-1K-Corr); and CE-MUCAL, a cross-encoder variant (CE-MPNet (bs4)) which excels on the challenging Multi-WebNLG-Test (MRR=91.20), demonstrating strong disambiguation capability for closely related graphs.

4 Direct Preference Optimization for KG-to-Text Generation

We next explore how to leverage our cross-modal alignment models to create preference data as a signal for Direct Preference Optimization (DPO). Unlike RLHF approaches that require explicit reward modeling or policy gradients, DPO directly optimizes the model on *preference pairs* of generated texts, encouraging the model to prefer higher-scoring outputs while remaining close to a reference policy.

4.1 Creating Preference Data

To train KG-to-Text models using DPO, we need preference data of the form (graph, good text, bad text). We create this data by (i) generating several texts from a graph using multiple LLMs; (ii) scoring each KG/Text pair using our alignment models (HARD-MUCAL and CE-MUCAL); and (iii) forming the required triples based on the resulting KG/Text scores.

We compute KG/Text scores using both our models and three other KG/Text scoring metrics to compare the ability of all 5 approaches to support the creation of preference data. Figure 4 illustrates the overall pipeline, from LLM candidate generation to score-based ranking and final pairwise selection.

Generating candidate texts for a graph. To create the candidate texts, we input the graphs from KELM-Q1 to six instruction-tuned LLMs using few-shot prompting. (Prompt details are reported in Appendix J.) For each graph g in KELM-Q1, this process yields a set of candidate texts $\mathcal{T}_G = \{t_1, t_2, ..., t_7\}$ where each t_i is generated by one of the six LLMs used or from the KELM-Q1 dataset.

Scoring Candidate Texts. We score each KG/-Text pair using our two best KG/Text alignment models (cf. Section 3.5) and three existing KG/Text similarity metrics: EREDAT (Le Scao and Gardent, 2023), FACTSPOTTER (Zhang et al., 2023), and DATA QUEST-EVAL (Rebuffel et al., 2021a).

Given a chosen scoring function Score(G, t), we rank the texts in $\mathcal{T}_G$ from highest to lowest. Let

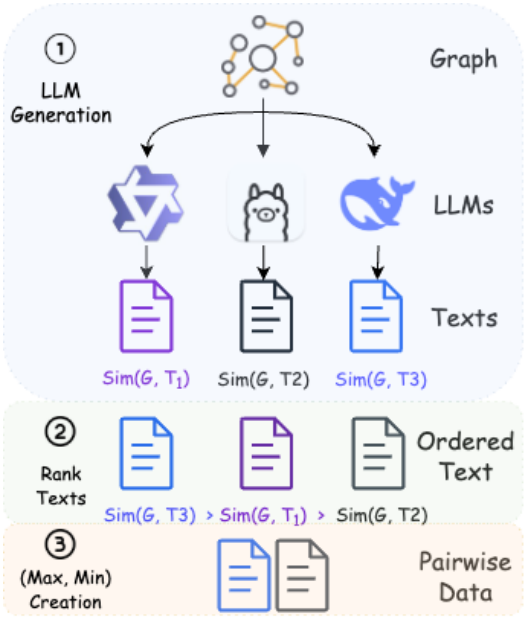

Figure 4: **Pipeline for constructing pairwise preference data from a given input graph.** First, LLMs generate the verbalisations based on the graph. Next, we compute the similarity between the graph and each generated text and rank the texts accordingly. Finally, we select the texts with the highest and lowest similarity scores to create preference pairs.

t_{chosen} (t_C) denote the *top-ranked* candidate and t_{rejected} (t_R) the *bottom-ranked* candidate. We then form $(G, t_{\text{C}}, t_{\text{R}})$ triplets for DPO training. This top-vs-bottom selection *maximizes the scoring gap* between preferred and dispreferred texts, making the preference signals more pronounced.

As we use 5 models to compute KG/Text similarity scores, we create 5 preference datasets, one for each model. This permits comparing our two alignment models to existing KG/Text reference-less metrics (i.e., EREDAT, FactSpotter and Data Quest-Eval) on the downstream task of creating preference data for DPO leaning, in effect allowing for a downstream, task driven, evaluation.

4.2 DPO Training

To evaluate how well MuCAL preferences drive generation, we apply DPO training to three instruction-tuned LLMs —LLAMA3-1B-INSTRUCT, SMOLLM2-1.7B-INSTRUCT, and QWEN2.5-1.5B-INSTRUCT. Each LLM is first fine-tuned on KELM-Q1 with the standard language-model objective and teacher forcing, yielding the reference policy π_{ref}.

Using the preference pairs (G, t_C, t_R) constructed in Section 4.1, we further train the policy π_θ with the DPO loss

$$\mathcal{L}_{\text{DPO}} = -\mathbb{E}_{(t_{\text{C}}, t_{\text{R}}) \sim \mathcal{D}_{\text{pref}}} \log \sigma \left(\beta \, \Delta_\theta(G, t_C, t_R) \right)$$

$$\Delta_\theta(G, t_C, t_R) = \log \frac{\pi_\theta(t_C|G)}{\pi_{\text{ref}}(t_C|G)} - \log \frac{\pi_\theta(t_R|G)}{\pi_{\text{ref}}(t_R|G)}$$

Notation. G is the input graph; (t_C, t_R) are the chosen and rejected texts; $\mathcal{D}_{\text{pref}}$ is the MuCAL-derived preference set; π_{ref} is the instruction-tuned reference model; π_θ is the trainable policy; $\beta = 0.1$ scales the KL term; and σ is the sigmoid function.

4.3 KG-to-Text Evaluation

Models. We compare DPO models trained on the preference data created using our 2 KG/Text alignment models with: Zero- and 3-shot prompting; Instruction-tuning on KELM-Q1 and 3 DPO models trained on preference data created using alternative KG/Text similarity metrics. For each approach (DPO, fine-tuning and prompting), we compare 3 LLMs (Llama3.2-1B-Instruct, SmoILM2-1.7B-Instruct, Qwen2.5-1.5B-Instruct).

Test sets. We evaluate on three test sets. KELM-TEST (IN-DOMAIN) is the manually validated test set from KELM (cf. Section 3.1). Since our DPO models are first fine-tuned on Kelm-Q1, this dataset serves as our primary *in-domain* benchmark. In contrast, the LLMs baselines may have had limited prior exposure to it as it is not publicly available. WEBNLG-TEST (PUBLIC) is the publicly released WebNLG test set (Gardent et al., 2017), which many LLMs might have partially seen during pretraining[7]. Finally, GOLD-OOD-472 (OUT-OF-DOMAIN) is a new dataset we created to assess performance on new or infrequent properties. We curated a set of 472 *unseen* Wikidata graphs, each containing 3–10 triples and covering domains not covered by WebNLG (e.g., from domains like Airlines, Chemicals, and Conferences). For each graph, we then generated English reference texts using an LLM-based approach and having these generated texts manually edited by native speakers. This dataset is neither included in the training nor available on the internet, thus representing a genuine *out-of-domain* challenge.

Evaluation Metrics. We use 8 reference-based metrics such as SacreBLEU (Post, 2018), TER (Snover et al., 2006), METEOR (Banerjee and Lavie, 2005), ChrF++ (Popović, 2017),

[7]Details of WebNLG-Test are also provided in Section 3.1

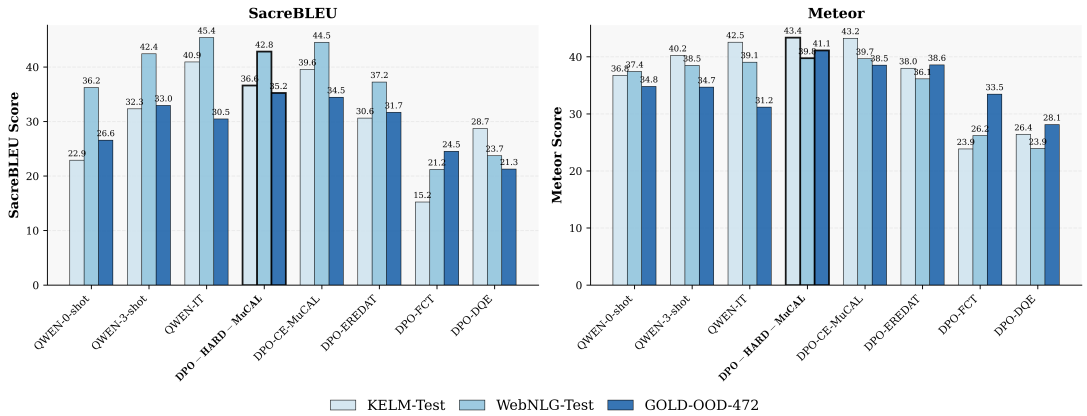

Figure 5: **SacreBLEU and Meteor Scores for KG-to-Text Generation.** KELM-Test is *in-domain*, WebNLG-Test is OOD but seen as it is a widely used public benchmark available on the internet, and GOLD-OOD-472 is both unseen and OOD, containing *unseen* graphs from Wikidata (out-of-domain). We compare various QWEN baselines (*0-shot*, *3-shot*, *IT*) against DPO models with different preference signals (DPO-{HARD-MuCAL, CE-MuCAL, EREDAT, FCT, DQE}). Our DPO-HARD-MuCAL approach (highlighted in black) demonstrates notably stronger generalization on *unseen* data than instruction-tuned QWEN and other DPO variants. *Abbreviations:* QWEN denotes QWEN2.5-1.5B-Instruct; IT = Instruction Tuning; DPO-X = DPO trained with preference signal X.

BERTScore (Zhang et al., 2020), BLEURT (Sellam et al., 2020), PARENT (Dhingra et al., 2019), SEScore2 (Xu et al., 2023). We also evaluate factual alignment with the input graph using reference-less metrics like EREDAT (Scao and Gardent, 2023), FactSpotter (Zhang et al., 2023), and Data QuestEval (Rebuffel et al., 2021b). This combination of in-domain, public-domain, and out-of-domain test sets, along with both reference-based and reference-less metrics, offers a comprehensive view of the quality of DPO-tuned models' generations.

Qualitative Analysis. To further assess the generation capabilities of our DPO-tuned models, we conducted a small-scale qualitative analysis on the most challenging test set, GOLD-OOD-472. We compared the outputs of the best-performing DPO-tuned model against those of the instruction-tuned baseline. Specifically, we ranked all examples by the absolute difference in their METEOR scores between the two models. We then selected the top three cases where the DPO model substantially outperformed the instruction-tuned model, as well as the top three cases in the opposite direction. This procedure highlights instances where one model is most confidently favored over the other according to METEOR, providing insight into their respective strengths and weaknesses.

4.4 Results and Discussion

For lack of space, we focus here on two metrics: **SacreBLEU**, a widely adopted reference-based metric that gauges surface-level similarity to gold texts, and **METEOR**, which considers stemming and synonyms and therefore handles paraphrasing better. Full results for additional metrics are reported in Section Q.

Global results for all three approaches and each of the three LLMs are shown in Figure 1. Since QWEN2.5-1.5B-INSTRUCT gains the largest out-of-domain improvement from DPO among the three LLMs, all ablations in this section are performed on Qwen. Figure 5 presents the Qwen-based results.

In addition to quantitative metrics, we present some examples comparing DPO-tuned and instruction-tuned outputs in Appendix R.

DPO generalises better than LLMs and Fine-Tuned Models. A salient trend emerges on unseen data (*GOLD-OOD-472*): Although instruction tuning (Qwen-IT) excels at matching the style of known domains, its performance degrades substantially on unseen data. In contrast, our DPO trained model (DPO-HARD-MUCAL) maintains robust scores on both SacreBLEU and METEOR, indicating improved generalization to out-of-domain data.

DPO paraphrases the reference. On the *in-domain* or *seen* testsets (KELM-Test, WebNLG-

Public), the instruction-tuned model (Qwen-IT) achieves higher SacreBLEU than both LLMs and DPO models indicating stronger adaptation to the target domain. However, on this same testsets, *DPO*-based models yield better METEOR scores than both the fine-tuned and the LLM models. This suggests that DPO learning supports the generation of texts that paraphrase rather then memorize the reference which again suggests a better ability to generalise for DPO models.

Our KG/Text similarity models provide a better signal for DPO than existing metrics. DPO models trained on preference data generated using EREDAT, FactSpotter and Data-QuestEval underperform our models on all test sets and for both evaluation metrics (SacreBLEU and METEOR). This indicates that our models are better at ranking texts according to their semantic similarity with a graph thereby demonstrating improvement over these existing KG/Text similarity metrics.

The bi-encoder is more effective at ranking KG/-Text pairs than the cross encoder. While our two DPO models perform on par on in-domain and seen data, the bi-encoder based model (DPO-HARD-MuCAL) outperforms the cross-encoder (DPO-CE-MuCal) on unseen, OOD data (GOLD-OOD-472) highlighting the importance of including hard negatives in the training data of the KG/-Text alignment model.

5 Conclusion

We proposed novel models for KG/Text semantic alignment and studied both their retrieval accuracy and their ability to support the creation of preference data for DPO training. For retrieval, we show that our models outperform all baselines on three benchmarks of increasing difficulty. For DPO, we demonstrate that our models outperform existing KG/Text similarity metrics in constructing robust preference data. Experimental results also show that our DPO-tuned models generalize better to out-of-domain data compared to instruction-tuned baselines. Finally, our LLM-based preference construction framework can produce such datasets without relying on manually aligned KG/Text pairs, allowing for seamless extension to multilingual scenarios. In future work, we plan to extend our approach to multilingual, DPO-trained KG-to-Text generation and to explore alternative preference learning algorithms for KG-to-Text generation, aiming to further enhance both fidelity and adaptability across diverse domains and languages.

Limitations

While **MuCAL** is designed for multilingual KG/-Text alignment, our current DPO experiments focus solely on English KG-to-Text generation. Extending the pipeline to other languages remains a key direction for future work.

Our approach to creating preference datasets relies on high-quality texts from large LLMs (e.g., DeepSeek-v3 with 671B parameters), which can be costly and potentially restrictive.

Even though MuCAL is trained on (machine-translated) silver data and shows promising retrieval performance across languages, its efficacy for guiding *non-English* DPO training remains to be shown.

Acknowledgements

We thank the anonymous reviewers for their feedback. This work received government funding managed by the French National Research Agency under France 2030, reference number “ANR-23-IACL-0004.” (AI Chair Gardent: "Semantically Consistent LLM Based Text Generation"). Experiments presented in this paper were carried out using the Grid’5000 testbed, supported by a scientific in terest group hosted by Inria and including CNRS, RENATER and several Universities as well as other organizations (see https://www.grid5000.fr). This work was also granted access to the HPC resources of IDRIS under the allocation AD011016561 made by GENCI.

References

Oshin Agarwal, Heming Ge, Siamak Shakeri, and Rami Al-Rfou. 2021. Knowledge graph based synthetic corpus generation for knowledge-enhanced language model pre-training. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3554–3565, Online. Association for Computational Linguistics.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Thiago Castro Ferreira, Claire Gardent, Nikolai Ilinykh, Chris van der Lee, Simon Mille, Diego Moussallem, and Anastasia Shimorina, editors. 2020. *Proceedings of the 3rd International Workshop on Natural Language Generation from the Semantic Web (WebNLG+)*. Association for Computational Linguistics, Dublin, Ireland (Virtual).

Jordan Clive, Kris Cao, and Marek Rei. 2022. Control prefixes for parameter-efficient text generation. In *Proceedings of the 2nd Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pages 363–382, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Liam Cripwell, Anya Belz, Claire Gardent, Albert Gatt, Claudia Borg, Marthese Borg, John Judge, Michela Lorandi, Anna Nikiforovskaya, William Soto-Martinez, and Craig Thomson. 2023a. The 2023 WebNLG shared task on low resource languages. overview and evaluation results (WebNLG 2023). In *Proceedings of the Workshop on Multimodal, Multilingual Natural Language Generation and Multilingual WebNLG Challenge (MM-NLG 2023)*, pages 55–66, Prague, Czech Republic. Association for Computational Linguistics.

Liam Cripwell, Anya Belz, Claire Gardent, Albert Gatt, Claudia Borg, Marthese Borg, John Judge, Michela Lorandi, Anna Nikiforovskaya, William Soto-Martinez, and Craig Thomson. 2023b. The 2023 WebNLG Shared Task on Low Resource Languages Overview and Evaluation Results (WebNLG 2023). In *Proceedings of the Workshop on Multimodal, Multilingual Natural Language Generation and Multilingual WebNLG Challenge (MM-NLG 2023)*, Prague, Czech Republic.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jiawei Wang, Jin Chen, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Litong Wang, Liyue Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J. Chen, R. L. Jin, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runxin Xu, Ruoyu Zhang, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaoqing Wu, Shengfeng Ye, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Shuting Pan, T. Wang, Tao Yun, Tian Pei, Tianyu Sun, W. L. Xiao, Wangding Zeng, Wanjia Zhao, Wei An, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, X. Q. Li, Xiangyue Jin, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaojin Shen, Xiaokang Chen, Xiaokang Zhang, Xiaosha Chen, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingkai Yu, Xinnan Song, Xinxia Shan, Xinyi Zhou, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, Y. K. Li, Y. Q. Wang, Y. X. Wei, Y. X. Zhu, Yang Zhang, Yanhong Xu, Yanhong Xu, Yanping Huang, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Li, Yaohui Wang, Yi Yu, Yi Zheng, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Ying Tang, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yu Wu, Yuan Ou, Yuchen Zhu, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yukun Zha, Yunfan Xiong, Yunxian Ma, Yuting Yan, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Z. F. Wu, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhen Huang, Zhen Zhang, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhipeng Xu, Zhiyu Wu, Zhongyu Zhang, Zhuoshu Li, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Ziyi Gao, and Zizheng Pan. 2025. Deepseek-v3 technical report. *Preprint*, arXiv:2412.19437.

Daniel Deutsch, Rotem Dror, and Dan Roth. 2022. On the limitations of reference-free evaluations of generated text. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10960–10977, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Bhuwan Dhingra, Manaal Faruqui, Ankur Parikh, Ming-Wei Chang, Dipanjan Das, and William W. Cohen. 2019. Handling divergent reference texts when evaluating table-to-text generation. *Preprint*, arXiv:1906.01081.

Thiago Castro Ferreira, Claire Gardent, Nikolai Ilinykh, Chris van Der Lee, Simon Mille, Diego Moussallem, and Anastasia Shimorina. 2020. The 2020 Bilingual, Bi-Directional WebNLG+ Shared Task Overview and Evaluation Results (WebNLG+ 2020). In *Proceedings of the 3rd International Workshop on Natural Language Generation from the Semantic Web (WebNLG+)*, Dublin/Virtual, Ireland.

Nicholas Frosst, Nicolas Papernot, and Geoffrey Hinton. 2019. Analyzing and improving representations with the soft nearest neighbor loss. *Preprint*, arXiv:1902.01889.

Yossi Gandelsman, Alexei A. Efros, and Jacob Steinhardt. 2024. Interpreting clip's image representation via text-based decomposition. *Preprint*, arXiv:2310.05916.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. The WebNLG challenge: Generating text from RDF data. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 124–133, Santiago de Compostela, Spain. Association for Computational Linguistics.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov,

Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Shawn Im and Yixuan Li. 2024. On the generalization of preference learning with dpo. *Preprint*, arXiv:2408.03459.

Viet Dac Lai, Chien Van Nguyen, Nghia Trung Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan A. Rossi, and Thien Huu Nguyen. 2023. Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback. *Preprint*, arXiv:2307.16039.

Teven Le Scao and Claire Gardent. 2023. Joint representations of text and knowledge graphs for retrieval and evaluation. In *Findings of the Association for Computational Linguistics: IJCNLP-AACL 2023 (Findings)*, pages 110–122, Nusa Dua, Bali. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *Preprint*, arXiv:1910.13461.

Zihao Li, Zhuoran Yang, and Mengdi Wang. 2023. Reinforcement learning with human feedback: Learning dynamic choices via pessimism. *Preprint*, arXiv:2305.18438.

Anna Nikiforovskaya and Claire Gardent. 2024. Evaluating RDF-to-text generation models for English and Russian on out of domain data. In *Proceedings of the 17th International Natural Language Generation Conference*, pages 134–144, Tokyo, Japan. Association for Computational Linguistics.

Maja Popović. 2017. chrF++: words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation*, pages 612–618, Copenhagen, Denmark. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting bleu scores. *Preprint*, arXiv:1804.08771.

Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. 2021. RocketQA: An optimized training approach to dense passage retrieval for open-domain question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5835–5847, Online. Association for Computational Linguistics.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Preprint*, arXiv:2305.18290.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2023. Exploring the limits of transfer learning with a unified text-to-text transformer. *Preprint*, arXiv:1910.10683.

Clement Rebuffel, Thomas Scialom, Laure Soulier, Benjamin Piwowarski, Sylvain Lamprier, Jacopo Staiano, Geoffrey Scoutheeten, and Patrick Gallinari. 2021a. Data-QuestEval: A referenceless metric for data-to-text semantic evaluation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8029–8036, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Clément Rebuffel, Thomas Scialom, Laure Soulier, Benjamin Piwowarski, Sylvain Lamprier, Jacopo Staiano, Geoffrey Scoutheeten, and Patrick Gallinari. 2021b. Data-questeval: A referenceless metric for data-to-text semantic evaluation. *Preprint*, arXiv:2104.07555.

Nils Reimers and Iryna Gurevych. 2019. Sentence-{BERT}: Sentence Embeddings using {S}iamese {BERT}-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Teven Le Scao and Claire Gardent. 2023. Joint representations of text and knowledge graphs for retrieval and evaluation. *Preprint*, arXiv:2302.14785.

Phillip Schneider, Tim Schopf, Juraj Vladika, Mikhail Galkin, Elena Simperl, and Florian Matthes. 2022. A decade of knowledge graphs in natural language processing: A survey. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 601–614, Online only. Association for Computational Linguistics.

Thibault Sellam, Dipanjan Das, and Ankur P. Parikh. 2020. Bleurt: Learning robust metrics for text generation. *Preprint*, arXiv:2004.04696.

Anastasia Shimorina, Elena Khasanova, and Claire Gardent. 2019. Creating a corpus for Russian data-to-text generation using neural machine translation and post-editing. In *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing*, pages 44–49, Florence, Italy. Association for Computational Linguistics.

Matthew Snover, Bonnie Dorr, Rich Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pages 223–231, Cambridge, Massachusetts, USA. Association for Machine Translation in the Americas.

Yixin Tang, Hua Cheng, Yiquan Fang, and Yiming Pan. 2022. In-batch negatives' enhanced self-supervised learning. In *2022 IEEE 34th International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 161–166.

Jędrzej Warczyński, Mateusz Lango, and Ondrej Dusek. 2024. Leveraging large language models for building interpretable rule-based data-to-text systems. In *Proceedings of the 17th International Natural Language Generation Conference*, pages 622–630, Tokyo, Japan. Association for Computational Linguistics.

Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, Bin Zhang, Xiong Wang, Yunfei Chu, and Junyang Lin. 2025. Qwen2.5-omni technical report. *Preprint*, arXiv:2503.20215.

Wenda Xu, Xian Qian, Mingxuan Wang, Lei Li, and William Yang Wang. 2023. SESCORE2: Learning text generation evaluation via synthesizing realistic mistakes. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5166–5183, Toronto, Canada. Association for Computational Linguistics.

Han Zhang, Jing Yu Koh, Jason Baldridge, Honglak Lee, and Yinfei Yang. 2021. Cross-modal contrastive learning for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 833–842.

Kun Zhang, Oana Balalau, and Ioana Manolescu. 2023. FactSpotter: Evaluating the factual faithfulness of graph-to-text generation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10025–10042, Singapore. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. *Preprint*, arXiv:1904.09675.

A Model Size and Hardware

Table 2 summarises the parameter counts, GPU configurations, and wall-clock training times for every model used in our study.

Model	Params	Hardware	Duration
MPNet	278M	*Pre-trained*	*Pre-trained*
BGE-M3	567M	*Pre-trained*	*Pre-trained*
HARD-MuCAL	278M	1×A40 (46GB)	5.0 h
CE-MuCAL	278M	1×A100 (40GB)	39.0 h
QWEN-IT	1.5B	1×L40S (48GB)	0.75 h
DPO-MuCAL	1.5B	1×A40 (46GB)	1.37 h

Table 2: Model Specifications and Training Infrastructure

B Datasets Description

We summarise the datasets, their descriptions, and statistics in Table 4.

C Creating the Multilingual Silver Data

Kelm-Q1. (Agarwal et al., 2021) constructed a corpus of 15M (graph, text) pairs from Wikidata and Wikipedia using distant supervision, fine-tuning and generation. Since the dataset is automatically generated, the alignment between graph and text is imperfect. We therefore filter the dataset in several steps as follows. Using the cross-modal RDF-Text encoder from (Scao and Gardent, 2023), we first filter out all (graph, text) pairs with cosine similarity lower than 0.9. We balance the dataset across graph sizes and Wikidata properties to prevent skew towards frequent properties. This reduces the dataset to 80K (graph, text) pairs.

To further improve data quality, we apply the Data QuestEval (DQE) metric (Rebuffel et al., 2021a), designed for data-to-text tasks and leveraging question generation and question answering models to evaluate the alignment between graphs and texts, and retain only the top-scoring quartile (Q1) of the filtered KELM data based on DQE scores. This final filtering step yields 18,723 high-quality (graph, text) pairs.

Multilingual Silver Data. To create the multilingual silver data, we compared the performance of 6 MT models on the Test-1K (see Appendix F for its creation details) and selected the best one for each language. Specifically, we translate the Test-1K into target languages using the MT models, and then evaluate the translations' quality using cross-language semantic similarity metrics. For example, we evaluate our French translations with the gold English text by computing $Sim(T_{EN}, T_{FR})$.

Model	Arabic	Chinese	French	Russian	Spanish
Helsinki-NLP			✓		✓
M2M100-418M	✓				
NLLB-200-600M					
mBART-large-50					
M2M100-1.2B		✓		✓	
NLLB-200-3.3B					

Table 3: Selected Translation Models for Each Target Language. Models are selected based on their results on the average normalised scores over 6 metrics. Cf. Table 9.

Table 3 shows the best Machine Translation (MT) model for each language and Table 9 shows the scores of all 6 MT models for all six metrics and five languages when evaluated on the Test-1K (graph, English text) test set using multilingual text similarity metrics.

D Encoder Architectures

Bi-Encoder. As depicted in Figure 6, the Bi-Encoder treats the text and graph as separate modalities. Knowledge Graphs are sets of (subject, predicate, object) triples. We linearize each graph into a sequence using the format: `[S]` $\texttt{subject}_1$ `[P]` $\texttt{predicate}_1$ `[O]` $\texttt{object}_1$ `[S]` $\texttt{subject}_2$ ... `[O]` $\texttt{object}_n$, where n is the number of triples and `[S]`, `[P]`, `[O]` are special tokens indicating the triple elements. The text and the linearized graph are independently encoded using the same multilingual embedding model.

Cross-Encoder. In contrast, the Cross-Encoder (Figure 6(right)) jointly encodes the text and graph by concatenating them with a special separator token `[SEP]`. The concatenated sequence is input to the multilingual embedding model, and the output is passed through a dense layer followed by a sigmoid activation function. This produces a matching score between 0 and 1, representing the degree of alignment between the text and the graph.

E Experimental Setup

E.1 Early Stopping

We employ an early stopping strategy for both MuCAL and DPO training to prevent overfitting while ensuring stable performance.

For MuCAL, we monitor the validation Mean Reciprocal Rank (MRR) in both the Graph-to-Text (G2T) and Text-to-Graph (T2G) directions, stop-

Dataset	Description	# KG/Text Pairs
Source Datasets (Texts are in English)		
WebNLG-Train	Gold	14,878
KELM-Q1	Silver	18,723
WebNLG-Test	Gold	1,779
KELM-Test	Gold	3,437
Training Sets		
EN-Train	KELM-Q1 + WebNLG-Train	33,601
Multi-Train-Silver	EN-Train + Translations	201,606
Test Sets		
Multi-Test-1K	1K (KELM-Test + WebNLG-Test) + Translations	6,000
Multi-WebNLG-Test	WebNLG-Test + Translations	10,674
Multi-Test-1K-Corr	Multi-Test-1K + Corrupted Graphs	10,800

Table 4: **Datasets of KG/Text pairs.** The sets with name "Multi-" include the machine translated texts in five target languages: Arabic, Chinese, French, Russian and Spanish. In all other datasets, the texts are in English. The knowledge graphs are from DBPedia (WebNLG) and Wikidata (KELM).

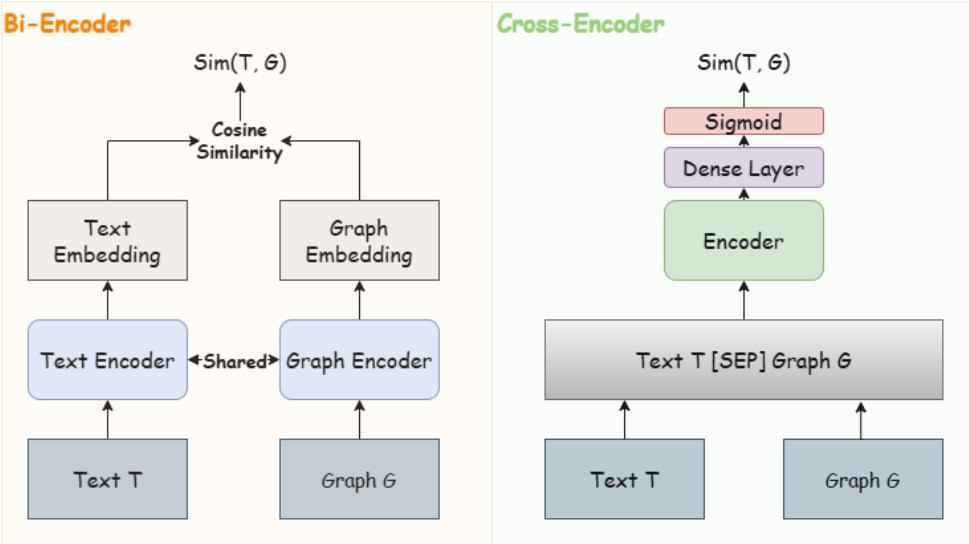

Figure 6: Overview of the Bi-Encoder architecture for multilingual KG/Text alignment. The input graph $\mathcal{G}$ is first linearized into a sequence such as `[S]` $\texttt{subject}_1$ `[P]` $\texttt{predicate}_1$ `[O]` $\texttt{object}_1$ `[S]` $\texttt{subject}_2$ `...` `[O]` $\texttt{object}_N$. The text and graph embeddings are produced by two parameter-shared encoders and updated jointly via backpropagation. The final similarity score $\text{Sim}(T, \mathcal{G})$ is constrained to the interval $[0, 1]$.

ping training when the performance ceases to improve.

For DPO training, we track the validation loss throughout the process and terminate training if no improvement is observed for five consecutive epochs. This conservative approach helps maintain the policy's alignment with the reference distribution π_{ref}, while preventing unnecessary updates that could lead to overfitting.

E.2 Temperature for MuCAL Training

The loss function 1 for MuCAL training incorporates a temperature parameter τ, which modulates the sharpness of the similarity distribution within a batch. Lower values of τ lead to sharper distributions that focus more on the most similar instances, while higher values result in smoother, more uniform distributions.

Notably, for memory-intensive training setups such as cross-encoder architectures, a smaller τ is particularly beneficial when using lower-precision computations (e.g., bfloat16), as it helps maintain numerical stability and mitigates precision-related artefacts. In our experiments, we set $\tau = 0.2$.

F English Test-1K Creation

To evaluate the robustness and retrieval performance of our multilingual KG/text representation models, we constructed the Test-1K dataset—a balanced and diverse subset of (graph, text) pairs. This appendix details the steps taken to create Test-1K from the WebNLG and KELM test data.

F.1 Data Source and Objectives

We utilized the WebNLG and KELM test datasets as our data sources, comprising a total of 5,216 (graph, text) pairs with 470 unique properties from both DBpedia and Wikidata. Our primary objectives were:

- **Maximizing Property Coverage:** Ensure the test set includes as many unique properties as possible to enhance diversity.
- **Balancing Graph Sizes:** Maintain an equal distribution of graph sizes ranging from 1 to 5 triples.

An overview of the test data comparison is presented in Table 5.

Test Data	Graph Size	Entry Num.	Prop. Num.	Prop. Cov.
WebNLG+KELM test	1-7	5216	470	-
Test-1K	1-5	1000	420	89.4%

Table 5: **Test data comparison.**

Entry Size	From KELM	From WebNLG	Total
1	25	175	200
2	106	94	200
3	101	99	200
4	97	103	200
5	98	102	200
Sum	427	573	1000

Table 6: **Test-1K source track.**

F.2 Step 1: Selecting Rare Properties

To ensure diversity and maximize property coverage, we prioritized graphs containing rare properties:

1. Subdivision by Graph Size: We divided the test data into five subsets based on the number of triples in each graph, corresponding to graph sizes from 1 to 5.
2. Frequency Analysis: Within each subset, we calculated the frequency of each property across all test data and within the subset.
3. Identification of Rare Properties: We identified properties with the minimum frequency within each subset. For instance, if the minimum frequency in a subset was 1, we selected properties occurring exactly once.
4. Intersection Selection: We cross-referenced these rare properties with those in the entire test data to select graphs containing them.

This process yielded a preliminary subset of 169 graphs covering 266 unique properties. Tables 7 and 8 present the property distribution before and after rare property selection.

Graph size	Non-Selected Num.	Property Num.	Seleted Num.
1	2462	225	0
2	648	213	0
3	672	229	0
4	563	293	0
5	451	298	0
Total:	4473	-	0

Table 7: **Initial test data before rare property selection.** "Non-Selected graphs" are graphs not yet selected in the source data; "Selected graphs" are graphs selected for the new test set and initially zero.

Graph size	Non-Selected Num.	Property Num.	Seleted Num.
1	2367	0	54
2	608	0	40
3	635	0	16
4	505	0	28
5	358	0	31
Total:	4473	0	169

Table 8: **Test data after rare property selection.** "Selected graphs" are graphs containing rare properties.

F.3 Step 2: Random Complementation

To achieve a balanced dataset with 200 entries for each graph size, we complemented the selected entries with additional samples:

1. **Random Selection:** For each graph size subset, we randomly selected additional non-redundant entries from the remaining unselected data until each subset contained 200 entries.

2. **Maximizing Property Coverage:** We prioritized entries that introduced new properties to maximize overall property coverage.

3. **Iteration for Optimization:** This process was repeated 100 times, and the iteration yielding the highest property coverage was selected as the final Test-1K dataset.

F.4 Final Composition of Test-1K

The final Test-1K dataset consists of:

- **Total Entries:** 1,000 (graph, text) pairs.
- **Balanced Graph Sizes:** 200 entries for each graph size from 1 to 5 triples.
- **Property Coverage:** 420 unique properties, achieving 89.4% coverage relative to the original datasets.

G Non-Contrastive Baseline Details

This section supplements Sec. 3.3 by providing full implementation details for our **non-contrastive binary classification baseline** (CLS-MPNet). First, we recap the in-batch negative sampling to create a balanced training set. Then, we describe the two classifier architectures we explore: bi-encoder and cross-encoder. Finally, we specify the binary cross-entropy loss used for training. These design choices mirror those of our contrastive model wherever possible to ensure a fair ablation.

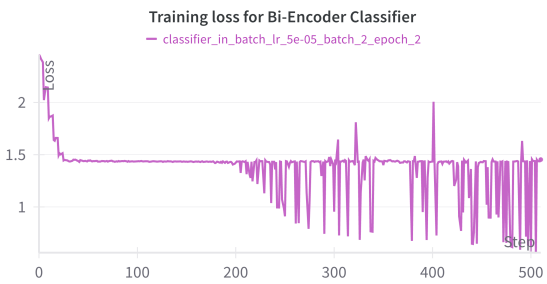

Figure 7: Training loss for Bi-Encoder Classifier.

G.1 In-Batch Negative Sampling

For strict comparability with contrastive training, we adopt an analogous in-batch negative-sampling strategy. Each mini-batch contains two aligned examples (G_i, T_i^l) with $i \in \{1, 2\}$ and target language l. Negatives are generated on-the-fly by cross-pairing graphs and texts, (G_i, T_j^l) with $j \neq i$. Consequently, every batch supplies $2 \times |l|$ positives and an equal number of negatives, ensuring a balanced signal at no extra preprocessing cost.

G.2 Classifier Architectures

Bi-Encoder. Our first instantiation is a lightweight bi-encoder. Instead of cosine similarity, we concatenate graph and text embeddings together with two interaction features—their L_1 distance and Hadamard product—and feed the resulting vector[8]

$$E_{\text{comb}} = \left[E_T \parallel E_G \parallel \|E_T - E_G\|_1 \parallel E_T \odot E_G\right]$$

to a single linear layer followed by a sigmoid activation.Although computationally efficient, the bi-encoder fails to capture subtle misalignments; the loss curve in Figure 7 plateaus early, indicating under-fitting to in-batch negatives. We attribute this to (i) information compression in the joint embedding and (ii) the additional complexity introduced by multilingual data.

Cross-Encoder. We therefore train a cross-encoder (Figure 6, right), which jointly encodes the concatenated graph–text sequence and predicts alignment with a classification head. Training proceeds smoothly, and early stopping on validation loss (Figure 8) selects the best checkpoint at roughly 9k steps. Despite showing an advantage to its backbone model (MPNet), the cross-encoder classifier still lags behind our contrastive model on all test sets (Table 1).

[8] $\parallel$ denotes concatenation.

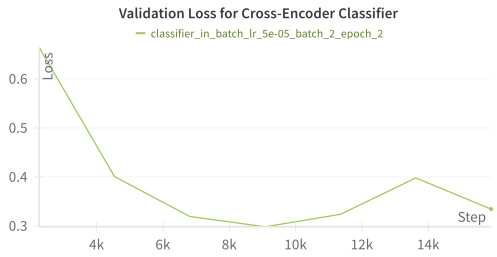

Figure 8: Validation loss for Cross-Encoder Classifier.

G.3 Loss Function

The non-contrastive model is trained with the standard *binary cross-entropy* loss, implemented as `torch.nn.BCEWithLogitsLoss`. Given a predicted logit $z \in \mathbb{R}$ and a gold label $y \in \{0, 1\}$, the loss for a single example is

$$\mathcal{L}_{\text{BCE}}(z, y) = -\big[y \log \sigma(z) + (1 - y) \log\big(1 - \sigma(z)\big)\big], \tag{2}$$

where $\sigma(\cdot)$ is the sigmoid function. Using the *logit* formulation avoids numerical underflow when $|z| \gg 0$ and therefore yields more stable gradients than applying a standalone sigmoid followed by log.

We keep the positive-to-negative ratio balanced within each batch (cf. §G*In-Batch Negative Sampling*), so the default loss weights $\alpha = \beta = 1$ suffice. At inference time, a pair (G, T) is classified as *aligned* when $\sigma(z) \geq 0.5$.

H Graph Corruption Details

This section presents all information related to graph, including (i) Corruption type definitions; (ii) Hard negative selection for alignment model training. It also shows some important experiments and findings, such as how to construct the hardest negative for the alignment model and which types of hard negatives make the model the most vulnerable.

H.1 Type Definitions of Graph Corruption

We provide additional information on how we generate corrupted graphs for **Multi-Test-1K-Corr** in this part. Each text in the test set is paired with one correct graph and five corrupted graphs. We use various heuristics to maintain as high a similarity between the correct and the corrupted graphs. Our corruption methods and the associated heuristics (H) are as follows:

- **Removed.** We remove one triple at random from a graph with at least two triples. For single-triple graphs (i.e., 1-triple), this corruption is skipped. For example:
  - Original graph: (A, _predicate1_, B), (B, _predicate2_, C)
  - After removal: (B, _predicate2_, C)

  H: Since the dataset is aligned at the KG/Text level, we assume that every triple expresses a fact reflected in the text. Therefore, removing any triple is expected to degrade the semantic alignment.

- **Added.** We add one new triple to a given graph, prioritizing triples that share at least one entity with the existing graph. For instance:
  - Original graph: (A, _predicate1_, B)
  - Added triple: (B, _predicate3_, D)
  - Corrupted graph: (A, _predicate1_, B), (B, _predicate3_, D)

  H: We prioritized triples that exist in the test set and share at least one entity with the original graph, thereby maintaining local connectivity. For instance, in the above example, the added triple (B, predicate3, D) shares the entity B with the original graph (A, predicate1, B).

- **Replace_Pred.** We pick a random triple and replace its predicate with a predicate not present in the graph.
  - Original triple: (A, _bornIn_, B)
  - Replaced triple: (A, _livesIn_, B)

  H: We first constructed a property space from all test set properties. For each property, we identified semantically similar alternatives using SBERT and filtered out those with a similarity score above 0.4, to avoid trivial substitutions which would result in an acceptable graph for the initial text. We then selected the most similar property below this threshold.

- **Replace_Entity.** We select a random triple (s, p, o) and replace its object with a new entity $o' \neq o$:
  - Original triple: (A, _locatedIn_, CityX)
  - Replaced triple: (A, _locatedIn_, CityZ)

  H: To keep the corruption local, we first gather the set $\mathcal{N}(s) = \{e \mid (s, p', e) \in$

G or $(e, p', s) \in G\}$ of entities already connected to the subject s. If $\mathcal{N}(s) \setminus \{o\}$ is non-empty, we sample $o' \sim \mathcal{N}(s) \setminus \{o\}$; otherwise we draw o' uniformly from the global entity pool $\mathcal{E} \setminus V(G)$. This strategy preserves graph connectivity while introducing a fact that is plausible yet incompatible with the original text.

- **Swapped.** We swap the subject and object of a random triple, provided they are distinct entities. To account for the inherent symmetry in certain predicates, we exclude triples involving predicates identified as Symmetrical Relationships (e.g., "taxon synonym", "partner in business or sport", etc.) from the swapping process.
  - Original triple: (A, _parentOf_, B)
  - Swapped triple: (B, _parentOf_, A)

Corruption Distribution. Across the entire dataset, each corruption type is applied with equal probability, yielding five corrupted graphs per graph. In total, we generate 5,800 corrupted graphs for 1,000 unique graphs in *Multi-Test-1K* where we skip generating *Removed* corrupted graphs for 200 single-triple graphs.

H.2 Hard Negatives Construction for Model Training

To identify the ability to distinguish aligned graphs from corrupted ones, we first conduct a robustness evaluation on the Multi-Test-1K-Corr dataset. This analysis reveals model-specific vulnerabilities that directly inform our strategy for constructing challenging hard negatives.

H.2.1 Robustness Evaluation

We assess model robustness by measuring their ability to rank correct graphs higher than corrupted counterparts in the retrieval task. For each language l, we proceed as follows:

1. **Compute Similarity Scores**: For each text instance $t_i^{(l)}$, we compute similarity scores between the text and:
   - The *correct graph* g_i
   - Five types of *corrupted graphs* $\mathcal{G}_i^{\text{corr}} = \{g_i^{(1)}, \ldots, g_i^{(K)}\}$
2. **Aggregate Similarity Scores**: Collect all similarity scores between text and graphs across all instances.
3. **Rank All Instances**: Rank all graph instances (both correct and corrupted) based on their similarity scores in descending order.
4. **Analyze Top-N Instances**: Consider the top N ranked instances and calculate the proportion of each graph type within these top N instances. For each graph type, compute:

$$\text{Proportion}_{\text{type}} = \frac{N_{\text{type}}^{\text{top}}}{N}, \tag{3}$$

where:
   - type contains *Good* and the corruptions.
   - $N_{\text{type}}^{\text{top}}$ is the number of instances of the graph type within the top N.
   - N is the total number of correct graphs.

In our evaluation, N is set to the total number of correct graphs, reflecting the ideal scenario where all correct graphs are ranked above corrupted ones.

Robustness Evaluation Results. We present full results in Section N.

Findings. Both bi-encoder and cross-encoder models exhibit pronounced vulnerability to *Swapped* and *Replaced_Pred* corruptions in all languages. This indicates that:

- Models struggle to detect inverted subject-object relationships, particularly for asymmetric predicates.
- Predicate substitution with semantically similar alternatives creates highly confusing negatives.

H.2.2 Targeted Negative Construction

Motivated by these findings, we prioritize *Swapped* and *Replaced_Pred* corruptions for hard negative generation, as they maximally exploit model weaknesses observed in the robustness analysis. The details of these two graph corruptions are discussed in Section H.1.

I Mono- & Multi-lingual Retrieval Setup

We assess the models' retrieval performance in both directions: **Text-to-Graph Retrieval** (Given a text, retrieve its corresponding graph from a set of candidate graphs) and **Graph-to-Text Retrieval** (Given a graph, retrieve its corresponding text from a set of candidate texts). For each retrieval direction, we consider both **monolingual** and **multilingual** settings.

Monolingual Retrieval seeks to retrieve the matching text/graph from a monolingual corpus. Let $\mathcal{G} = g_1, g_2, \ldots, g_N$ be the set of all graphs. $\mathcal{L}$ be the set of target languages. For each graph g_i, let $\mathcal{T}_i = \{t_i^{(l)} \mid l \in \mathcal{L}\}$ be the set of texts corresponding to g_i in each language l. Then in the monolingual setting for language l:

- **Text-to-Graph**: For each text $t_i^{(l)}$, we compute similarity scores $s_{ij} = \text{sim}(t_i^{(l)}, g_j)$ for all $g_j \in \mathcal{G}$, rank the graphs based on s_{ij} and retrieve the top ranking graph.
- **Graph-to-Text**: For each graph g_i, we compute similarity scores $s_{ij} = \text{sim}(g_i, t_j^{(l)})$ for all $t_j^{(l)}$, rank the texts based on s_{ij} and retrieve the top ranking text.

Multilingual Retrieval considers all target languages simultaneously rather than treating each language independently. As same as the monolingual retrieval, we have the retrieval tasks in two directions as follows:

- **Text-to-Graph** :For each text $t_i^{(l)}$ in language $l \in \mathcal{L}$, compute similarity scores $s_{ij} = \text{sim}(t_i^{(l)}, g_j)$ for all $g_j \in \mathcal{G}$.

  Aggregate the scores for each graph g_j across all languages:

$$S_{ij} = \sum_{l \in \mathcal{L}} s_{ij}^{(l)}, \tag{4}$$

  where $s_{ij}^{(l)}$ is the similarity between $t_i^{(l)}$ and g_j.

  Rank the graphs based on the aggregated scores S_{ij} and record the rank rank_i of the correct graph g_i.
- **Graph-to-Text**: For each graph g_i, compute similarity scores $s_{ij}^{(l)} = \text{sim}(g_i, t_j^{(l)})$ for all texts $t_j^{(l)}$ in all languages $l \in \mathcal{L}$.

  Aggregate the scores for each set of texts $\mathcal{T}_j$ corresponding to graph g_j:

$$S_{ij} = \sum_{l \in \mathcal{L}} s_{ij}^{(l)}, \tag{5}$$

  where $s_{ij}^{(l)}$ is the similarity between g_i and $t_j^{(l)}$.

  Rank the sets of texts $\mathcal{T}_j$ based on the aggregated scores S_{ij} and record the rank rank_i of the correct text set $\mathcal{T}_i$.

J Prompting Details

J.1 Instruction-Tuned LLMs

To obtain lexically diverse candidates while avoiding very low-quality generations, we query three well-established open-source model families, restricting the size to ≥ 7 B parameters:

- **Qwen2.5**—7B, 14B, and 32B-Int8 instruction variants (Xu et al., 2025);
- **DeepSeek**—v3 and `r1-distill-qwen-7B` (DeepSeek-AI et al., 2025);
- **Llama-3**—8B-INSTRUCT (Grattafiori et al., 2024).

J.2 Few-Shot Demonstration Selection

We employ two sampling strategies:

Random sampling (Qwen / DeepSeek). Because input graphs vary in size, we randomly draw three (graph, text) pairs from the manually validated KELM-Test set (Nikiforovskaya and Gardent, 2024), constraining their graph lengths to 1, 3, and 5 triples. This exposes the model to small, medium, and larger structures within a single prompt.

Heuristic Sampling (LLaMA-3). We prompt Llama-3-8B-Instruct using a *property-based* few-shot strategy that maximizes overlap between the input graph and the few-shot examples. Specifically, we maintain a pool of KG/Text pairs (from KELM-Test) sorted by the number of properties shared with the target graph. We select examples from top-ranked candidates such that their properties match or closely resemble those in the input graph. If certain properties are missing, we retrieve the most similar ones via a k-nearest neighbors (k-NN) search in a multilingual embedding space (e.g., LaBSE). This ensures that each few-shot example is closely aligned with the structural and semantic content of the target graph, thus providing more relevant demonstrations.

J.3 Prompt Templates

```
The following is a graph represented as
    a set of triples. Each triple
    provides a fact in the form '[S]
    subject [P] predicate [O] object'.
    Please convert this graph into
    fluent and natural language text.
    The output should be a concise and
    coherent description, consisting of
    one or a few sentences. Ensure that:
1. All facts from the graph are included
     in the description.
2. The text is fluent, natural, and easy
     to understand.
3. There is no repetition or missing
    details.

Graph:
<Graph1>
Text:
<Text1>

Graph:
<Graph2>
Text:
<Text2>

Graph:
<Graph3>
Text:
<Text3>

Graph:
<Graph to verbalise>
Text:
```

J.4 Prompting Efficiency and Cost

Our preference pipeline relies on advanced LLMs, but we access them through official APIs rather than hosting models locally. This choice greatly reduces inference time and eliminates the need for expensive hardware. As described above, our 3-shot prompting template is lightweight, producing a single verbalization per graph. For instance, generating verbalizations for the entire KELM-Q1 dataset (18,723 graphs) with the DeepSeek-V3 API incurred a cost of under $10, demonstrating that the pipeline is both cost-effective and scalable in practice.

K MT Model Evaluation Results on Test-1K

Lang	Model	NMT_Direct	NMT_Pivot	NMT_Cross	SBert	BERTScore	Cometkiwi	Norm_Row_Avg
EN-ZH	**Helsinki-NLP**	0.634	0.754	0.590	0.8528	0.8984	0.618	0.420
	M2M100-418M	0.719	0.820	0.682	0.8896	0.9039	0.634	0.799
	NLLB-200-600M	0.583	0.712	0.493	0.7876	0.8886	0.591	0.000
	MBart-large-50	0.695	0.784	0.625	0.8705	0.8946	0.687	0.660
	M2M100-1.2B	0.749	0.844	0.695	0.8975	0.9067	0.679	**0.986**
	NLLB-200-3.3B	0.624	0.738	0.525	0.8023	0.8906	0.665	0.270
EN-AR	**Helsinki-NLP**	0.658	0.732	0.583	0.7820	0.8821	0.610	0.211
	M2M100-418M	0.742	0.822	0.698	0.8613	0.8995	0.598	**0.874**
	NLLB-200-600M	0.712	0.808	0.623	0.8012	0.8886	0.651	0.539
	MBart-large-50	0.691	0.786	0.622	0.8354	0.8989	0.630	0.645
	M2M100-1.2B	0.607	0.688	0.569	0.8494	0.8992	0.495	0.305
	NLLB-200-3.3B	0.754	0.846	0.665	0.8127	0.8911	0.710	0.775
EN-FR	**Helsinki-NLP**	0.885	0.963	0.891	0.9714	0.9370	0.715	**0.996**
	M2M100-418M	0.848	0.943	0.875	0.9653	0.9372	0.638	0.679
	NLLB-200-600M	0.805	0.887	0.822	0.9501	0.9291	0.612	0.055
	MBart-large-50	0.833	0.915	0.848	0.9654	0.9348	0.667	0.532
	M2M100-1.2B	0.860	0.951	0.884	0.9645	0.9363	0.661	0.756
	NLLB-200-3.3B	0.807	0.879	0.819	0.9473	0.9284	0.631	0.035
EN-RU	**Helsinki-NLP**	0.618	0.748	0.649	0.8559	0.9074	0.582	0.000
	M2M100-418M	0.712	0.863	0.773	0.8978	0.9190	0.631	0.735
	NLLB-200-600M	0.670	0.798	0.696	0.8657	0.9094	0.601	0.259
	MBart-large-50	0.712	0.844	0.738	0.9087	0.9198	0.661	0.771
	M2M100-1.2B	0.756	0.892	0.802	0.9093	0.9216	0.677	**1.000**
	NLLB-200-3.3B	0.694	0.805	0.710	0.8712	0.9102	0.648	0.421
EN-ES	**Helsinki-NLP**	0.884	1.016	0.906	0.9725	0.9434	0.767	**1.000**
	M2M100-418M	0.838	0.956	0.881	0.9659	0.9422	0.682	0.539
	NLLB-200-600M	0.815	0.925	0.851	0.9530	0.9362	0.662	0.094
	MBart-large-50	0.794	0.912	0.837	0.9548	0.9398	0.671	0.113
	M2M100-1.2B	0.853	0.972	0.894	0.9683	0.9422	0.707	0.684
	NLLB-200-3.3B	0.851	0.957	0.882	0.9625	0.9389	0.723	0.527

Table 9: Results on the 1K English test set for translation models across five target languages (Arabic, Chinese, French, Russian, Spanish). 'Norm_Row_Avg' is the row-wise normalized average over the other metrics. We use multilingual evaluation metrics to compare the generated text to the English reference.

L Multilingual Retrieval Task Results

L.1 Multilingual T2G&G2T on Multi-Test-1K

Model	Multi T2G				Multi G2T			
	Recall@1	Recall@3	Recall@10	MRR	Recall@1	Recall@3	Recall@10	MRR
BI-ENCODER								
Baselines								
MPNet	82.60	92.80	99.60	88.52	80.50	92.80	100.00	87.18
CLS-MPNet	88.10	97.10	98.90	92.73	90.60	97.10	99.70	94.09
BGE-M3	94.50	99.30	100.00	96.96	90.80	98.30	100.00	94.68
Batch Size Variants								
BE-MPNet (bs8; ep2)	96.30	99.60	99.90	97.91	93.00	99.40	99.90	96.06
BE-MPNet (bs16; ep2)	97.60	99.70	100.00	98.65	95.20	99.60	100.00	97.36
BE-MPNet (bs32; ep2)	97.40	99.60	100.00	98.53	94.80	99.50	99.90	97.10
Base Model Variants								
BE-MPNet	97.70	99.80	99.90	98.75	95.20	99.50	99.90	97.30
BE-BGE-M3	97.40	99.80	99.90	98.58	95.50	99.50	100.00	97.44
Hard Negative Variants								
BE-MPNet-Hard1	96.60	98.80	99.40	97.83	93.60	99.20	99.50	96.24
BE-MPNet-Hard2	95.30	98.60	99.10	96.96	94.60	98.80	99.50	96.72
BE-MPNet-Hard4	93.90	97.80	99.00	95.94	93.10	98.30	99.60	95.79
BE-BGE-M3-Hard1	97.20	99.30	99.70	98.29	94.60	99.10	99.70	96.94
BE-BGE-M3-Hard2	95.50	97.90	98.60	96.79	94.30	98.00	98.60	96.26
CROSS-ENCODER								
Batch Size Variants								
CE-MPNet (bs2; ep2)	92.50	98.40	99.50	95.50	90.70	98.60	99.80	94.66
CE-MPNet (bs3; ep2)	94.80	99.20	99.60	97.03	93.20	98.30	99.80	95.97
CE-MPNet (bs4; ep2)	97.10	99.60	99.90	98.39	96.20	99.00	99.80	97.75
Best CE Model								
CE-MPNet	98.60	99.80	99.90	99.23	97.20	99.60	99.90	98.36

Table 10: **Evaluation of models on multilingual tasks (Multi-Test-1K).** We report Recall@k and MRR for both text-to-graph (T2G) and graph-to-text (G2T) retrieval tasks. "BE" and "CE" denote bi-encoder and cross-encoder models, respectively. `MPNet` and `BGE-M3` are pre-trained embedding models used as baselines and as backbones for other variants; `CLS-MPNet` is a binary classification baseline. **Abbreviations**: "ep" = epochs, "bs" = batch size, "HardX" = X hard negatives per graph. Unless otherwise specified, bi-encoders use batch size 32 and cross-encoders use batch size 4; they both use training epoch number 10.

L.2 Multilingual T2G&G2T on Multi-WebNLG-Test

Model	Multi T2G				Multi G2T			
	Recall@1	Recall@3	Recall@10	MRR	Recall@1	Recall@3	Recall@10	MRR
BI-ENCODER								
Baselines								
MPNet	38.56	60.48	82.97	53.07	39.52	60.54	83.75	53.67
CLS-MPNet	62.06	85.67	95.78	74.97	65.99	84.37	95.39	76.61
BGE-M3	77.46	92.30	98.99	85.55	64.70	85.10	96.57	76.10
Batch Size Variants								
BE-MPNet (bs8; ep2)	79.82	93.25	99.16	87.21	74.26	89.49	98.20	82.87
BE-MPNet (bs16; ep2)	81.84	94.88	99.21	88.69	76.50	91.40	98.71	84.75
BE-MPNet (bs32; ep2)	83.87	95.39	99.10	89.90	78.02	91.23	98.54	85.58
Base Model Variants								
BE-MPNet	84.71	96.29	99.49	90.59	78.36	92.19	98.82	85.99
BE-BGE-M3	87.07	97.25	99.78	92.25	79.88	94.72	99.55	87.66
Hard Negative Variants								
BE-MPNet-Hard1	81.17	92.69	98.03	87.57	75.49	89.77	97.86	83.57
BE-MPNet-Hard2	79.93	92.92	97.25	86.71	76.56	91.46	98.03	84.64
BE-MPNet-Hard4	78.41	91.46	96.85	85.43	74.14	89.60	97.13	82.50
BE-BGE-M3-Hard1	86.68	97.64	99.55	92.15	80.21	95.05	99.33	87.89
BE-BGE-M3-Hard2	85.95	95.78	98.26	91.01	79.82	93.65	98.82	87.17
CROSS-ENCODER								
Batch Size Variants								
CE-MPNet (bs2; ep2)	67.90	89.54	98.15	79.43	66.78	88.14	97.64	78.44
CE-MPNet (bs3; ep2)	72.57	93.54	99.04	83.31	69.08	90.39	98.20	80.37
CE-MPNet (bs4; ep2)	81.34	95.45	99.21	88.62	78.64	93.20	98.82	86.50
Best CE Model								
CE-MPNet	88.42	98.31	99.78	93.37	85.39	95.45	97.36	90.52

Table 11: **Evaluation of models on multilingual tasks (Multi-WebNLG-Test).** We report Recall@k and MRR for both text-to-graph (T2G) and graph-to-text (G2T) retrieval tasks. "BE" and "CE" denote bi-encoder and cross-encoder models, respectively. `MPNet` and `BGE-M3` are pre-trained embedding models used as baselines and as backbones for other variants; `CLS-MPNet` is a binary classification baseline. **Abbreviations**: "ep" = epochs, "bs" = batch size, "HardX" = X hard negatives per graph. Unless otherwise specified, bi-encoders use batch size 32 and cross-encoders use batch size 4; they both use training epoch number 10.

M Monolingual Retrieval Task Results

M.1 Monolingual T2G on Multi-Test-1K

M.1.1 Mono-T2G Retrieval Results for English and Chinese on Multi-Test-1K

Model	EN				ZH			
	Recall@1	Recall@3	Recall@10	MRR	Recall@1	Recall@3	Recall@10	MRR
BI-ENCODER								
Baselines								
MPNet	83.20	93.90	99.50	89.16	78.70	91.00	99.10	85.74
CLS-MPNet	91.60	96.50	98.80	94.32	72.60	84.70	93.70	79.91
BGE-M3	96.00	99.60	100.00	97.77	90.90	98.00	100.00	94.58
EREDAT	96.50	99.60	99.90	98.01	-	-	-	-
FactSpotter	67.70	94.40	96.90	80.52	-	-	-	-
Batch Size Variants								
BE-MPNet (bs8; ep2)	96.10	99.80	99.90	97.79	87.90	95.20	97.90	91.90
BE-MPNet (bs16; ep2)	97.60	99.80	100	98.69	91.90	97.40	99.60	94.92
BE-MPNet (bs32; ep2)	97.60	99.80	100	98.69	93.50	98.30	99.90	96.00
Base Model Variants								
BE-MPNet	97.90	99.90	99.90	98.87	93.80	98.40	99.90	96.28
BE-BGE-M3	97.40	99.80	99.90	98.59	92.40	97.50	99.30	95.10
Hard Negative Variants								
BE-MPNet-Hard1	96.70	98.90	99.40	97.90	92.50	97.00	99.20	95.05
BE-MPNet-Hard2	96.10	98.70	99.30	97.40	92.30	97.00	98.90	94.83
BE-MPNet-Hard4	94.20	97.80	99.10	96.11	88.20	95.10	98.30	92.04
BE-BGE-M3-Hard1	97.20	99.30	99.80	98.29	91.70	97.10	99.00	94.58
BE-BGE-M3-Hard2	96.20	98.00	98.40	97.19	90.00	95.00	97.70	92.95
CROSS-ENCODER								
Batch Size Variants								
CE-MPNet (bs2; ep2)	93.50	97.60	98.50	95.60	76.00	89.30	95.40	83.43
CE-MPNet (bs3; ep2)	93.30	98.40	98.90	95.84	84.80	95.50	98.80	90.41
CE-MPNet (bs4; ep2)	95.60	98.70	99.10	97.14	90.40	98.70	99.80	94.42
Best CE Model								
CE-MPNet	96.60	98.60	98.60	97.53	92.60	98.70	99.50	95.60

Table 12: **Mono-T2G Retrieval Results for English and Chinese on Multi-Test-1K.** We report Recall@k and MRR for the text-to-graph (T2G) retrieval task. "BE" and "CE" denote bi-encoder and cross-encoder models, respectively. `MPNet` and `BGE-M3` are pre-trained embedding models used as baselines and as backbones for other variants; `CLS-MPNet` is a binary classification baseline. **Abbreviations**: "ep" = epochs, "bs" = batch size, "HardX" = X hard negatives per graph. Unless otherwise specified, bi-encoders use batch size 32 and cross-encoders use batch size 4; they both use training epoch number 10.

M.1.2 Mono-T2G Retrieval Results for French and Arabic on Multi-Test-1K

Model	FR				AR			
	Recall@1	Recall@3	Recall@10	MRR	Recall@1	Recall@3	Recall@10	MRR
BI-ENCODER								
Baselines								
CLS-MPNet	84.70	93.20	97.80	89.49	64.80	78.70	91.10	73.43
MPNet	82.50	93.30	99.50	88.59	73.20	86.90	98.10	81.62
BGE-M3	94.70	99.10	100.00	96.99	88.90	96.80	99.40	93.13
Batch Size Variants								
BE-MPNet (bs8; ep2)	94.10	98.80	99.70	96.43	84.20	93.10	96.70	89.05
BE-MPNet (bs16; ep2)	96.10	99.00	99.90	97.71	87.20	96.30	98.90	91.99
BE-MPNet (bs32; ep2)	96.80	99.50	99.90	98.12	89.80	97.00	98.90	93.54
Base Model Variants								
BE-MPNet	97.30	99.60	99.90	98.43	90.20	97.60	99.40	93.98
BE-BGE-M3	94.40	98.80	99.50	96.66	87.00	95.10	98.00	91.44
Hard Negative Variants								
BE-MPNet-Hard1	96.20	98.70	99.30	97.52	89.20	95.90	98.30	92.77
BE-MPNet-Hard2	94.70	98.10	99.20	96.50	88.10	95.70	98.30	92.10
BE-MPNet-Hard4	93.30	97.10	98.70	95.43	85.00	93.00	97.40	89.60
BE-BGE-M3-Hard1	94.90	98.70	99.40	96.85	88.20	95.10	97.50	91.93
BE-BGE-M3-Hard2	93.40	96.50	98.00	95.13	86.60	93.10	96.30	90.30
CROSS-ENCODER								
Batch Size Variants								
CE-MPNet (bs2; ep2)	89.40	96.00	97.20	92.82	69.90	84.40	91.00	77.87
CE-MPNet (bs3; ep2)	90.80	97.10	98.50	94.15	82.60	94.10	96.90	88.50
CE-MPNet (bs4; ep2)	94.00	98.50	99.00	96.26	88.40	96.60	98.50	92.60
Best CE Model								
CE-MPNet	95.50	97.60	97.90	96.57	89.60	97.80	99.00	93.61

Table 13: **Mono-T2G Retrieval Results for French and Arabic on Multi-Test-1K.** We report Recall@k and MRR for the text-to-graph (T2G) retrieval task. "BE" and "CE" denote bi-encoder and cross-encoder models, respectively. `MPNet` and `BGE-M3` are pre-trained embedding models used as baselines and as backbones for other variants; `CLS-MPNet` is a binary classification baseline. **Abbreviations**: "ep" = epochs, "bs" = batch size, "HardX" = X hard negatives per graph. Unless otherwise specified, bi-encoders use batch size 32 and cross-encoders use batch size 4; they both use training epoch number 10.

M.1.3 Mono-T2G Retrieval Results for Spanish and Russian on Multi-Test-1K

Model	ES				RU			
	Recall@1	Recall@3	Recall@10	MRR	Recall@1	Recall@3	Recall@10	MRR
BI-ENCODER								
Baselines								
CLS-MPNet	86.80	94.90	97.60	91.04	75.80	87.80	93.80	82.60
MPNet	82.70	92.90	99.60	88.64	78.50	90.70	99.20	85.66
BGE-M3	95.10	99.20	100.00	97.18	90.70	97.70	100.00	94.40
Batch Size Variants								
BE-MPNet (bs8; ep2)	94.90	98.70	99.40	96.86	89.00	95.10	98.40	92.46
BE-MPNet (bs16; ep2)	96.30	98.90	99.70	97.71	92.10	97.20	99.60	94.85
BE-MPNet (bs16; ep2)	97.00	99.20	99.80	98.17	92.40	97.90	99.60	95.20
Base Model Variants								
BE-MPNet	97.40	99.20	99.70	98.37	92.80	98.10	99.70	95.59
BE-BGE-M3	95.80	99.00	99.60	97.37	91.40	97.50	98.90	94.48
Hard Negative Variants								
BE-MPNet-Hard1	95.70	98.40	99.30	97.22	91.70	96.60	98.80	94.36
BE-MPNet-Hard2	94.80	97.90	99.30	96.50	91.20	96.60	98.90	94.07
BE-MPNet-Hard4	93.30	97.20	98.90	95.49	88.40	95.10	97.70	91.86
BE-BGE-M3-Hard1	95.90	98.90	99.50	97.43	91.60	97.40	98.60	94.60
BE-BGE-M3-Hard2	93.70	97.30	98.00	95.50	89.10	94.80	96.90	92.27
CROSS-ENCODER								
Batch Size Variants								
CE-MPNet (bs2; ep2)	89.90	96.60	98.10	93.30	78.30	89.80	95.60	84.79
CE-MPNet (bs3; ep2)	90.80	97.30	98.50	94.04	86.50	95.10	98.20	91.11
CE-MPNet (bs4; ep2)	94.50	98.90	99.20	96.71	91.70	97.80	99.50	94.92
Best CE Model								
CE-MPNet	94.40	97.30	97.60	95.84	91.30	98.00	98.70	94.63

Table 14: **Mono-T2G Retrieval Results for Spanish and Russian on Multi-Test-1K.** We report Recall@k and MRR for the text-to-graph (T2G) retrieval task. "BE" and "CE" denote bi-encoder and cross-encoder models, respectively. `MPNet` and `BGE-M3` are pre-trained embedding models used as baselines and as backbones for other variants; `CLS-MPNet` is a binary classification baseline. **Abbreviations**: "ep" = epochs, "bs" = batch size, "HardX" = X hard negatives per graph. Unless otherwise specified, bi-encoders use batch size 32 and cross-encoders use batch size 4; they both use training epoch number 10.

M.2 Monolingual T2G on Multi-WebNLG-Test

M.2.1 Mono-T2G Retrieval Results for English and Chinese on Multi-WebNLG-Test

Model	EN				ZH			
	Recall@1	Recall@3	Recall@10	MRR	Recall@1	Recall@3	Recall@10	MRR
BI-ENCODER								
Baselines								
CLS-MPNet	71.95	89.54	96.68	81.58	48.57	69.59	85.83	61.56
MPNet	39.91	62.28	85.16	54.67	34.63	55.20	79.48	48.84
BGE-M3	80.04	94.66	99.44	87.69	63.58	83.47	95.50	75.06
EREDAT	82.91	95.05	99.66	89.46	-	-	-	-
FactSpotter	37.27	70.55	95.11	56.90	-	-	-	-
Batch Size Variants								
BE-MPNet (bs8; ep2)	81.06	93.87	99.16	88.04	68.35	85.10	95.39	78.06
BE-MPNet (bs16; ep2)	83.08	95.62	99.10	89.50	71.28	88.25	96.91	80.66
BE-MPNet (bs32; ep2)	84.94	95.78	99.16	90.68	74.09	89.49	97.30	82.79
Base Model Variants								
BE-MPNet	86.17	96.18	99.66	91.40	74.09	90.22	97.70	82.86
BE-BGE-M3	88.48	97.81	99.66	93.22	78.36	91.91	98.26	85.82
Hard Negative Variants								
BE-MPNet-Hard1	81.84	93.31	98.09	88.05	71.73	86.96	96.07	80.47
BE-MPNet-Hard2	81.62	93.03	97.19	87.65	71.33	87.75	95.67	80.31
BE-MPNet-Hard4	78.81	91.96	97.02	85.77	67.06	85.27	94.66	77.25
BE-BGE-M3-Hard1	88.98	97.75	99.49	93.43	78.08	92.41	98.37	85.77
BE-BGE-M3-Hard2	88.48	96.01	98.37	92.49	77.74	90.50	97.02	84.84
CROSS-ENCODER								
Batch Size Variants								
CE-MPNet (bs2; ep2)	70.83	90.11	96.68	80.96	43.51	69.42	89.09	59.23
CE-MPNet (bs3; ep2)	70.66	91.40	97.64	81.60	53.01	79.26	94.38	67.82
CE-MPNet (bs4; ep2)	77.91	93.76	97.81	86.03	60.48	83.98	97.13	73.80
Best CE Model								
CE-MPNet	86.23	96.23	97.53	91.20	69.25	90.78	98.88	80.76

Table 15: **Mono-T2G Retrieval Results for English and Chinese on Multi-WebNLG-Test.** We report Recall@k and MRR for the text-to-graph (T2G) retrieval task. "BE" and "CE" denote bi-encoder and cross-encoder models, respectively. `MPNet` and `BGE-M3` are pre-trained embedding models used as baselines and as backbones for other variants; `CLS-MPNet` is a binary classification baseline. **Abbreviations**: "ep" = epochs, "bs" = batch size, "HardX" = X hard negatives per graph. Unless otherwise specified, bi-encoders use batch size 32 and cross-encoders use batch size 4; they both use training epoch number 10.

M.2.2 Mono-T2G Retrieval Results for French and Arabic on Multi-WebNLG-Test

Model	FR				AR			
	Recall@1	Recall@3	Recall@10	MRR	Recall@1	Recall@3	Recall@10	MRR
BI-ENCODER								
Baselines								
CLS-MPNet	63.35	82.86	92.75	74.33	39.63	60.09	77.57	52.71
MPNet	39.12	61.10	83.59	53.68	31.25	50.87	76.45	45.45
BGE-M3	75.66	91.29	99.21	84.32	62.73	81.84	94.49	73.92
Batch Size Variants								
BE-MPNet (bs8; ep2)	76.50	91.62	98.37	84.65	61.44	79.43	91.79	72.18
BE-MPNet (bs16; ep2)	79.93	93.54	98.65	87.13	64.70	82.97	93.99	75.29
BE-MPNet (bs32; ep2)	82.46	93.37	98.93	88.66	68.02	84.20	95.11	77.49
Base Model Variants								
BE-MPNet	83.75	94.77	99.21	89.64	67.40	84.60	95.62	77.51
BE-BGE-M3	84.20	95.84	99.27	90.30	71.16	86.62	95.67	80.08
Hard Negative Variants								
BE-MPNet-Hard1	80.33	91.85	97.92	86.78	65.65	82.97	94.49	75.76
BE-MPNet-Hard2	79.37	91.51	97.13	86.00	64.81	82.46	93.54	75.11
BE-MPNet-Hard4	76.90	90.78	96.51	84.34	60.93	80.04	92.30	72.11
BE-BGE-M3-Hard1	84.94	95.67	99.04	90.55	70.26	87.07	96.07	79.69
BE-BGE-M3-Hard2	84.20	93.87	97.98	89.57	69.31	85.05	94.38	78.46
CROSS-ENCODER								
Batch Size Variants								
CE-MPNet (bs2; ep2)	64.31	86.85	96.35	76.43	39.91	64.76	84.43	55.05
CE-MPNet (bs3; ep2)	67.12	89.77	97.36	79.08	51.10	76.00	92.02	65.38
CE-MPNet (bs4; ep2)	73.97	91.74	97.30	83.30	57.11	80.61	94.55	70.59
Best CE Model								
CE-MPNet	83.81	95.22	97.02	89.55	65.32	87.75	96.80	77.19

Table 16: **Mono-T2G Retrieval Results for French and Arabic on Multi-WebNLG-Test.** We report Recall@k and MRR for the text-to-graph (T2G) retrieval task. "BE" and "CE" denote bi-encoder and cross-encoder models, respectively. `MPNet` and `BGE-M3` are pre-trained embedding models used as baselines and as backbones for other variants; `CLS-MPNet` is a binary classification baseline. **Abbreviations**: "ep" = epochs, "bs" = batch size, "HardX" = X hard negatives per graph. Unless otherwise specified, bi-encoders use batch size 32 and cross-encoders use batch size 4; they both use training epoch number 10.

M.2.3 Mono-T2G Retrieval Results for Spanish and Russian on Multi-WebNLG-Test

Model	ES				RU			
	Recall@1	Recall@3	Recall@10	MRR	Recall@1	Recall@3	Recall@10	MRR
BI-ENCODER								
Baselines								
CLS-MPNet	65.21	84.32	94.04	75.90	49.97	69.31	84.20	62.06
MPNet	39.74	61.38	83.42	54.20	33.95	55.42	79.60	48.61
BGE-M3	76.50	92.41	99.21	85.08	67.34	85.83	96.29	77.72
Batch Size Variants								
BE-MPNet (bs8; ep2)	79.15	93.37	98.65	86.58	68.35	83.75	94.04	77.57
BE-MPNet (bs16; ep2)	81.56	94.32	98.65	88.35	69.98	86.79	95.50	79.32
BE-MPNet (bs32; ep2)	83.47	94.88	98.82	89.41	73.07	87.13	96.46	81.32
Base Model Variants								
BE-MPNet	85.33	95.62	99.21	90.66	72.63	87.97	97.41	81.37
BE-BGE-M3	85.10	96.07	99.33	90.94	76.39	90.05	97.53	84.05
Hard Negative Variants								
BE-MPNet-Hard1	80.10	92.19	97.81	86.74	70.04	84.77	95.39	78.80
BE-MPNet-Hard2	80.21	92.02	96.85	86.56	69.48	84.88	95.39	78.46
BE-MPNet-Hard4	77.40	90.61	96.96	84.63	66.55	82.74	94.49	76.22
BE-BGE-M3-Hard1	86.12	96.46	99.49	91.51	75.60	89.99	97.70	83.60
BE-BGE-M3-Hard2	85.22	94.32	98.03	90.17	74.59	87.63	96.01	82.18
CROSS-ENCODER								
Batch Size Variants								
CE-MPNet (bs2; ep2)	63.58	86.73	96.18	76.00	48.06	73.30	89.99	62.84
CE-MPNet (bs3; ep2)	66.10	89.04	97.53	78.42	56.27	81.45	94.10	70.20
CE-MPNet (bs4; ep2)	74.70	92.52	97.86	84.00	63.13	85.50	96.40	75.30
Best CE Model								
CE-MPNet	82.63	94.83	97.30	88.84	71.73	90.95	97.86	81.87

Table 17: **Mono-T2G Retrieval Results for Spanish and Russian on Multi-WebNLG-Test.** We report Recall@k and MRR for the text-to-graph (T2G) retrieval task. "BE" and "CE" denote bi-encoder and cross-encoder models, respectively. `MPNet` and `BGE-M3` are pre-trained embedding models used as baselines and as backbones for other variants; `CLS-MPNet` is a binary classification baseline. **Abbreviations**: "ep" = epochs, "bs" = batch size, "HardX" = X hard negatives per graph. Unless otherwise specified, bi-encoders use batch size 32 and cross-encoders use batch size 4; they both use training epoch number 10.

M.3 Monolingual G2T on Multi-Test-1K

M.3.1 Mono-G2T Retrieval Results for English and Chinese on Multi-Test-1K

Model	EN				ZH			
	Recall@1	Recall@3	Recall@10	MRR	Recall@1	Recall@3	Recall@10	MRR
BI-ENCODER								
Baselines								
CLS-MPNet	91.20	96.60	98.70	94.05	77.30	88.50	95.60	83.91
MPNet	83.20	93.50	99.80	88.98	75.10	88.50	98.80	83.06
BGE-M3	92.90	99.50	100.00	96.09	84.20	95.00	99.80	89.36
EREDAT	95.20	98.90	99.80	97.10	-	-	-	-
FactSpotter	71.10	87.80	99.20	80.74	-	-	-	-
Batch Size Variants								
be_ep2_bs8	95.70	99.30	99.90	97.53	87.30	95.00	98.10	91.44
be_ep2_bs16	96.60	99.80	100.00	98.14	89.90	97.00	99.60	93.64
be_ep10_bs32	96.10	99.10	100.00	97.66	89.70	96.10	99.10	93.29
Base Model Variants								
BE-MPNet	96.70	99.90	99.90	98.25	92.30	98.00	99.80	95.28
BE-BGE-M3	97.20	99.70	100.00	98.49	91.50	97.30	99.00	94.52
Hard Negative Variants								
BE-MPNet-Hard1	95.00	99.40	99.50	96.99	89.80	97.00	99.30	93.59
BE-MPNet-Hard2	95.60	99.10	99.50	97.30	90.70	97.50	99.10	94.20
BE-MPNet-Hard4	94.90	98.60	99.50	96.85	88.10	95.80	99.00	92.40
BE-BGE-M3-Hard1	97.40	99.40	99.70	98.45	91.20	97.40	98.80	94.40
BE-BGE-M3-Hard2	96.30	98.60	98.90	97.48	89.30	96.20	97.80	92.88
CROSS-ENCODER								
Batch Size Variants								
CE-MPNet (bs2; ep2)	91.30	98.00	98.00	94.75	70.70	87.10	97.00	80.14
CE-MPNet (bs3; ep2)	92.80	98.70	99.40	95.81	75.60	92.40	98.90	84.49
CE-MPNet (bs4; ep2)	95.70	98.60	99.20	97.26	89.20	97.60	99.50	93.50
Best CE Model								
CE-MPNet	96.40	98.60	98.70	97.51	90.60	98.00	99.50	94.41

Table 18: **Mono-G2T Retrieval Results for English and Chinese on Multi-Test-1K.** We report Recall@k and MRR for the graph-to-text (G2T) retrieval task. "BE" and "CE" denote bi-encoder and cross-encoder models, respectively. `MPNet` and `BGE-M3` are pre-trained embedding models used as baselines and as backbones for other variants; `CLS-MPNet` is a binary classification baseline. **Abbreviations**: "ep" = epochs, "bs" = batch size, "HardX" = X hard negatives per graph. Unless otherwise specified, bi-encoders use batch size 32 and cross-encoders use batch size 4; they both use training epoch number 10.

M.3.2 Mono-G2T Retrieval Results for French and Arabic on Multi-Test-1K

Model	FR				AR			
	Recall@1	Recall@3	Recall@10	MRR	Recall@1	Recall@3	Recall@10	MRR
BI-ENCODER								
Baselines								
CLS-MPNet	86.50	94.00	98.00	90.78	63.90	78.60	88.90	72.76
MPNet	82.10	91.90	99.80	87.93	70.70	85.90	97.50	79.80
BGE-M3	91.80	98.00	100.00	95.08	82.20	92.80	99.30	88.16
Batch Size Variants								
be_ep2_bs8	93.30	98.50	99.70	96.01	81.80	92.30	96.40	87.50
be_ep2_bs16	95.10	99.10	99.70	97.12	86.80	95.40	98.60	91.38
be_ep2_bs32	96.00	99.10	99.80	97.63	88.10	95.90	99.10	92.28
Base Model Variants								
BE-MPNet	96.20	99.50	99.90	97.81	90.00	96.50	99.30	93.59
BE-BGE-M3	95.10	99.00	99.60	97.04	88.20	95.00	98.10	91.91
Hard Negative Variants								
BE-MPNet-Hard1	94.00	98.90	99.40	96.33	89.60	97.00	98.80	93.25
BE-MPNet-Hard2	94.60	98.30	99.40	96.56	88.10	96.20	98.60	92.24
BE-MPNet-Hard4	94.20	97.90	99.40	96.21	84.10	94.60	97.90	89.70
BE-BGE-M3-Hard1	94.80	98.50	99.40	96.83	88.20	95.60	97.70	92.08
BE-BGE-M3-Hard2	93.00	97.20	98.30	95.21	87.60	93.90	96.30	91.16
CROSS-ENCODER								
Batch Size Variants								
CE-MPNet (bs2; ep2)	86.10	96.10	98.40	91.18	65.80	83.10	93.40	75.76
CE-MPNet (bs3; ep2)	86.00	97.90	99.30	91.79	72.20	88.30	96.10	81.11
CE-MPNet (bs4; ep2)	93.70	98.00	99.20	95.99	82.70	93.10	98.00	88.47
Best CE Model								
CE-MPNet	95.20	98.00	98.60	96.66	84.80	93.80	98.60	89.75

Table 19: **Mono-G2T Retrieval Results for French and Arabic on Multi-Test-1K.** We report Recall@k and MRR for the graph-to-text (G2T) retrieval task. "BE" and "CE" denote bi-encoder and cross-encoder models, respectively. `MPNet` and `BGE-M3` are pre-trained embedding models used as baselines and as backbones for other variants; `CLS-MPNet` is a binary classification baseline. **Abbreviations**: "ep" = epochs, "bs" = batch size, "HardX" = X hard negatives per graph. Unless otherwise specified, bi-encoders use batch size 32 and cross-encoders use batch size 4; they both use training epoch number 10.

M.3.3 Mono-G2T Retrieval Results for Spanish and Russian on Multi-Test-1K

Model	ES				RU			
	Recall@1	Recall@3	Recall@10	MRR	Recall@1	Recall@3	Recall@10	MRR
BI-ENCODER								
Baselines								
CLS-MPNet	88.20	94.20	97.20	91.67	78.50	89.40	94.70	84.61
MPNet	82.10	93.20	99.70	88.01	77.40	89.60	98.70	84.43
BGE-M3	92.30	98.50	99.90	95.42	85.90	95.20	99.70	91.08
Batch Size Variants								
be_ep2_bs8	93.70	98.20	99.40	96.11	88.80	95.70	98.60	92.33
be_ep2_bs16	95.20	98.80	99.70	97.07	89.70	96.80	99.10	93.41
be_ep2_bs32	95.70	99.30	99.80	97.49	90.50	97.50	99.60	94.11
Base Model Variants								
BE-MPNet	95.60	99.20	99.60	97.41	91.70	97.30	99.50	94.82
BE-BGE-M3	95.40	98.80	99.60	97.21	90.70	96.90	98.70	94.04
Hard Negative Variants								
BE-MPNet-Hard1	94.00	98.90	99.40	96.33	89.60	97.00	98.80	93.25
BE-MPNet-Hard2	94.40	98.50	99.20	96.44	90.30	96.90	99.20	93.85
BE-MPNet-Hard4	93.30	97.90	99.50	95.77	88.70	95.80	98.80	92.66
BE-BGE-M3-Hard1	95.10	99.20	99.40	97.13	90.50	96.90	98.70	93.91
BE-BGE-M3-Hard2	94.00	97.60	98.60	95.93	89.20	95.00	97.50	92.47
CROSS-ENCODER								
Batch Size Variants								
CE-MPNet (bs2; ep2)	86.60	96.50	98.80	91.69	77.50	89.50	95.90	84.38
CE-MPNet (bs3; ep2)	88.10	96.80	98.70	92.54	82.00	92.80	97.90	88.08
CE-MPNet (bs4; ep2)	93.20	98.10	98.80	95.63	89.90	97.60	99.60	93.80
Best CE Model								
CE-MPNet	94.80	97.30	97.50	96.06	91.30	97.60	98.60	94.45

Table 20: **Mono-G2T Retrieval Results for Spanish and Russian on Multi-Test-1K.** We report Recall@k and MRR for the graph-to-text (G2T) retrieval task. "BE" and "CE" denote bi-encoder and cross-encoder models, respectively. `MPNet` and `BGE-M3` are pre-trained embedding models used as baselines and as backbones for other variants; `CLS-MPNet` is a binary classification baseline. **Abbreviations**: "ep" = epochs, "bs" = batch size, "HardX" = X hard negatives per graph. Unless otherwise specified, bi-encoders use batch size 32 and cross-encoders use batch size 4; they both use training epoch number 10.

M.4 Monolingual G2T on Multi-WebNLG-Test

M.4.1 Mono-G2T Retrieval Results for English and Chinese on Multi-WebNLG-Test

Model	EN				ZH			
	Recall@1	Recall@3	Recall@10	MRR	Recall@1	Recall@3	Recall@10	MRR
BI-ENCODER								
Baselines								
CLS-MPNet	72.29	88.81	96.29	81.34	53.06	71.89	86.79	64.88
MPNet	43.28	64.87	85.27	57.17	31.37	51.26	75.32	45.43
BGE-M3	70.49	88.53	98.15	80.55	46.49	68.07	88.42	60.36
EREDAT	76.67	91.06	98.43	84.65	-	-	-	-
FactSpotter	38.90	64.70	90.33	55.46	-	-	-	-
Batch Size Variants								
BE-MPNet (bs8; ep2)	79.60	92.36	98.37	86.61	65.09	83.42	93.76	75.46
BE-MPNet (bs16; ep2)	82.18	92.97	98.71	88.37	67.90	85.33	96.18	77.76
BE-MPNet (bs32; ep2)	83.53	93.76	98.88	89.34	69.59	86.23	96.29	79.22
Base Model Variants								
BE-MPNet	83.31	94.32	99.21	89.31	71.22	87.68	96.74	80.45
BE-BGE-M3	85.72	96.74	99.61	91.48	73.86	90.33	97.53	82.95
Hard Negative Variants								
BE-MPNet-Hard1	79.26	92.13	97.86	86.33	68.35	84.26	95.28	77.75
BE-MPNet-Hard2	80.33	91.91	97.81	86.92	70.15	86.68	96.35	79.44
BE-MPNet-Hard4	78.70	91.34	97.70	85.61	65.21	83.47	95.11	75.68
BE-BGE-M3-Hard1	86.40	96.91	99.44	91.70	73.92	90.44	97.41	82.94
BE-BGE-M3-Hard2	86.12	95.62	98.76	91.17	74.48	88.70	96.80	82.54
CROSS-ENCODER								
Batch Size Variants								
CE-MPNet (bs2; ep2)	70.21	89.32	96.57	80.39	47.61	71.44	90.56	62.05
CE-MPNet (bs3; ep2)	69.76	90.50	97.75	80.75	49.63	76.34	92.75	64.86
CE-MPNet (bs4; ep2)	76.90	92.69	97.64	85.31	58.74	83.70	95.45	72.31
Best CE Model								
CE-MPNet	85.39	95.45	97.36	90.52	67.34	87.46	97.02	78.47

Table 21: **Mono-G2T Retrieval Results for English and Chinese on Multi-WebNLG-Test.** We report Recall@k and MRR for the graph-to-text (G2T) retrieval task. "BE" and "CE" denote bi-encoder and cross-encoder models, respectively. `MPNet` and `BGE-M3` are pre-trained embedding models used as baselines and as backbones for other variants; `CLS-MPNet` is a binary classification baseline. **Abbreviations**: "ep" = epochs, "bs" = batch size, "HardX" = X hard negatives per graph. Unless otherwise specified, bi-encoders use batch size 32 and cross-encoders use batch size 4; they both use training epoch number 10.

M.4.2 Mono-G2T Retrieval Results for French and Arabic on Multi-WebNLG-Test

Model	FR				AR			
	Recall@1	Recall@3	Recall@10	MRR	Recall@1	Recall@3	Recall@10	MRR
BI-ENCODER								
Baselines								
CLS-MPNet	66.44	84.09	93.87	76.46	45.08	63.74	79.20	56.76
MPNet	41.60	61.44	82.80	54.90	27.15	44.35	69.70	40.58
BGE-M3	67.00	84.99	97.02	77.43	42.78	64.08	84.88	56.64
Batch Size Variants								
BE-MPNet (bs8; ep2)	75.55	90.61	97.02	83.76	61.44	78.92	90.05	71.75
BE-MPNet (bs16; ep2)	78.25	91.29	97.64	85.52	64.08	81.34	93.14	74.27
BE-MPNet (bs32; ep2)	79.93	92.07	98.15	86.69	65.21	82.91	93.76	75.43
Base Model Variants								
BE-MPNet	80.10	92.36	98.59	86.95	64.81	82.69	94.44	75.31
BE-BGE-M3	82.12	94.83	99.10	88.89	68.52	85.22	94.66	78.04
Hard Negative Variants								
BE-MPNet-Hard1	77.57	90.95	97.58	84.92	64.08	81.39	92.92	74.28
BE-MPNet-Hard2	78.25	91.01	97.70	85.35	63.80	82.29	93.87	74.56
BE-MPNet-Hard4	74.20	89.49	96.74	82.60	58.12	77.07	92.02	69.67
BE-BGE-M3-Hard1	82.46	94.49	98.82	88.89	67.90	84.94	94.21	77.54
BE-BGE-M3-Hard2	81.45	93.48	97.92	87.86	66.39	83.59	93.65	76.29
CROSS-ENCODER								
Batch Size Variants								
CE-MPNet (bs2; ep2)	64.31	83.98	95.33	75.38	43.51	65.99	85.89	57.55
CE-MPNet (bs3; ep2)	63.52	86.57	96.57	76.04	46.88	69.65	89.04	61.15
CE-MPNet (bs4; ep2)	70.43	90.33	96.91	81.05	52.56	76.50	92.02	66.49
Best CE Model								
CE-MPNet	80.66	94.10	97.02	87.57	59.36	82.52	93.25	71.99

Table 22: **Mono-G2T Retrieval Results for French and Arabic on Multi-WebNLG-Test.** We report Recall@k and MRR for the graph-to-text (G2T) retrieval task. "BE" and "CE" denote bi-encoder and cross-encoder models, respectively. `MPNet` and `BGE-M3` are pre-trained embedding models used as baselines and as backbones for other variants; `CLS-MPNet` is a binary classification baseline. **Abbreviations**: "ep" = epochs, "bs" = batch size, "HardX" = X hard negatives per graph. Unless otherwise specified, bi-encoders use batch size 32 and cross-encoders use batch size 4; they both use training epoch number 10.

M.4.3 Mono-G2T Retrieval Results for Spanish and Russian on Multi-WebNLG-Test

Model	ES				RU			
	Recall@1	Recall@3	Recall@10	MRR	Recall@1	Recall@3	Recall@10	MRR
BI-ENCODER								
Baselines								
CLS-MPNet	66.84	84.60	93.70	76.85	54.41	72.29	84.77	65.19
MPNet	41.60	61.27	83.47	55.14	34.35	53.34	76.05	47.79
BGE-M3	67.96	86.00	96.68	78.25	53.68	73.86	90.50	66.23
Batch Size Variants								
BE-MPNet (bs8; ep2)	76.73	90.11	97.70	84.25	65.65	82.97	92.80	75.62
BE-MPNet (bs16; ep2)	78.92	92.07	98.48	86.11	69.03	84.04	94.66	78.05
BE-MPNet (bs32; ep2)	80.44	92.47	98.37	87.19	70.94	84.77	95.00	79.35
Base Model Variants								
BE-MPNet	80.83	92.69	98.59	87.38	70.38	85.95	95.73	79.30
BE-BGE-M3	82.63	95.45	99.33	89.29	72.46	88.70	96.35	81.47
Hard Negative Variants								
BE-MPNet-Hard1	77.12	90.67	97.58	84.74	67.90	83.59	94.60	77.27
BE-MPNet-Hard2	78.70	91.57	97.58	85.80	68.41	84.99	95.22	77.96
BE-MPNet-Hard4	75.32	89.21	96.80	83.26	66.16	82.80	93.99	75.95
BE-BGE-M3-Hard1	82.52	95.28	99.44	89.17	72.63	88.08	96.29	81.34
BE-BGE-M3-Hard2	82.57	94.27	98.37	88.81	72.74	86.85	95.67	81.09
CROSS-ENCODER								
Batch Size Variants								
CE-MPNet (bs2; ep2)	63.97	85.33	95.95	75.81	50.42	72.74	89.32	63.90
CE-MPNet (bs3; ep2)	63.46	86.45	97.13	75.94	53.68	77.12	92.02	67.31
CE-MPNet (bs4; ep2)	72.29	91.40	96.80	82.20	60.65	83.36	94.94	73.27
Best CE Model								
CE-MPNet	80.66	93.82	96.96	87.47	67.79	89.04	96.80	79.03

Table 23: **Mono-G2T Retrieval Results for Spanish and Russian on Multi-WebNLG-Test.** We report Recall@k and MRR for the graph-to-text (G2T) retrieval task. "BE" and "CE" denote bi-encoder and cross-encoder models, respectively. `MPNet` and `BGE-M3` are pre-trained embedding models used as baselines and as backbones for other variants; `CLS-MPNet` is a binary classification baseline. **Abbreviations**: "ep" = epochs, "bs" = batch size, "HardX" = X hard negatives per graph. Unless otherwise specified, bi-encoders use batch size 32 and cross-encoders use batch size 4; they both use training epoch number 10.

N Robustness to Corrupted Data Evaluation Results

N.1 English

Model	Correct	Error Types				
	Good	Removed	Added	ReplP	ReplE	Swapped
BE-MPNet (bs8; ep2)	39.70	7.30	8.50	18.20	0.40	25.90
BE-MPNet (bs16; ep2)	41.00	6.20	6.20	17.40	0.10	29.10
BE-MPNet (bs32; ep2)	40.90	6.00	5.50	17.10	0.00	30.50
BE-MPNet (bs32; ep10)	40.90	7.00	4.10	16.70	0.00	31.30
CE-MPNet (bs2; ep2)	33.20	9.80	8.60	13.40	0.30	34.70
CE-MPNet (bs3; ep2)	30.90	9.70	8.40	19.40	1.10	30.50
CE-MPNet (bs4; ep2)	35.80	6.40	9.30	12.60	0.50	35.40
CE-MPNet (bs4; ep10)	32.40	6.50	6.70	23.60	0.40	30.40

Table 24: **Top-N Type Distribution for English on Multi-Test-1K-Corr.** The table reports the distribution of correct matches ("Good") and various error types ("Removed", "Added", "Replace_Pred", "Replace_Entity", and "Swapped") for the top 1000 retrieval results, where "Good" indicates correctly matched graphs. The results illustrate the robustness of models against corrupted graphs. **BE-MPNet (bs32; ep10)** refers to the best MPNet-based alignment model without hard negatives.

N.2 French

Model	Correct	Error Types				
	Good	Removed	Added	ReplP	ReplE	Swapped
BE-MPNet (bs8; ep2)	36.60	6.70	9.30	19.30	0.50	27.60
BE-MPNet (bs16; ep2)	37.30	6.00	7.10	17.80	0.10	31.70
BE-MPNet (bs32; ep2)	38.30	5.90	6.30	17.90	0.00	31.60
BE-MPNet (bs32; ep10)	38.70	6.70	4.00	17.60	0.30	32.70
CE-MPNet (bs2; ep2)	32.60	9.50	7.40	16.90	1.10	32.50
CE-MPNet (bs3; ep2)	27.90	10.00	9.90	21.30	1.40	29.50
CE-MPNet (bs4; ep2)	23.50	9.50	15.20	23.90	4.30	23.60
CE-MPNet (bs4; ep10)	33.90	7.20	8.00	16.60	0.50	33.80

Table 25: **Top-N Type Distribution for French on Multi-Test-1K-Corr.** The table reports the distribution of correct matches ("Good") and various error types ("Removed", "Added", "Replace_Pred", "Replace_Entity", and "Swapped") for the top 1000 retrieval results, where "Good" indicates correctly matched graphs. The results illustrate the robustness of models against corrupted graphs. **BE-MPNet (bs32; ep10)** refers to the best MPNet-based alignment model without hard negatives.

N.3 Spanish

Model	Correct	Error Types				
	Good	Removed	Added	ReplP	ReplE	Swapped
BE-MPNet (bs8; ep2)	37.60	7.40	10.20	19.00	0.30	25.50
BE-MPNet (bs16; ep2)	39.40	6.70	7.20	18.90	0.00	27.80
BE-MPNet (bs32; ep2)	39.60	6.50	6.20	18.00	0.00	29.70
BE-MPNet (bs32; ep10)	38.20	6.30	4.80	18.40	0.20	32.10
CE-MPNet (bs2; ep2)	31.20	10.50	7.60	17.20	0.80	32.70
CE-MPNet (bs3; ep2)	27.30	11.40	9.90	21.30	1.00	29.10
CE-MPNet (bs4; ep2)	22.80	10.20	15.30	24.90	3.60	23.20
CE-MPNet (bs4; ep10)	35.60	6.80	7.80	14.90	0.30	34.60

Table 26: **Top-N Type Distribution for Spanish on Multi-Test-1K-Corr.** The table reports the distribution of correct matches ("Good") and various error types ("Removed", "Added", "Replace_Pred", "Replace_Entity", and "Swapped") for the top 1000 retrieval results, where "Good" indicates correctly matched graphs. The results illustrate the robustness of models against corrupted graphs. **BE-MPNet (bs32; ep10)** refers to the best MPNet-based alignment model without hard negatives.

N.4 Russian

Model	Correct	Error Types				
	Good	Removed	Added	ReplP	ReplE	Swapped
BE-MPNet (bs8; ep2)	33.00	10.00	9.70	19.30	0.90	27.10
BE-MPNet (bs16; ep2)	33.30	8.70	6.90	18.50	0.20	32.40
BE-MPNet (bs32; ep2)	34.00	8.30	6.80	20.20	0.10	30.60
BE-MPNet (bs32; ep10)	35.80	7.80	5.70	19.10	0.10	31.50
CE-MPNet (bs2; ep2)	29.60	13.70	9.20	15.80	1.50	30.20
CE-MPNet (bs3; ep2)	27.10	10.40	11.30	20.30	2.70	28.20
CE-MPNet (bs4; ep2)	23.60	8.80	18.40	19.40	5.40	24.40
CE-MPNet (bs4; ep10)	32.50	8.40	8.70	17.00	0.70	32.70

Table 27: **Top-N Type Distribution for Russian on Multi-Test-1K-Corr.** The table reports the distribution of correct matches ("Good") and various error types ("Removed", "Added", "Replace_Pred", "Replace_Entity", and "Swapped") for the top 1000 retrieval results, where "Good" indicates correctly matched graphs. The results illustrate the robustness of models against corrupted graphs. **BE-MPNet (bs32; ep10)** refers to the best MPNet-based alignment model without hard negatives.

N.5 Arabic

Model	Correct	Error Types				
	Good	Removed	Added	ReplP	ReplE	Swapped
BE-MPNet (bs8; ep2)	32.80	8.90	9.10	20.20	1.10	27.90
BE-MPNet (bs16; ep2)	32.80	7.60	7.90	19.60	0.30	31.80
BE-MPNet (bs32; ep2)	33.00	7.20	7.20	21.50	0.50	30.60
BE-MPNet (bs32; ep10)	34.20	7.30	5.70	19.00	0.90	32.90
CE-MPNet (bs2; ep2)	29.50	11.90	10.90	16.40	2.00	29.30
CE-MPNet (bs3; ep2)	27.70	9.90	11.60	20.10	2.90	27.80
CE-MPNet (bs4; ep2)	24.50	8.20	17.50	21.40	4.20	24.20
CE-MPNet (bs4; ep2)	32.90	7.70	9.90	16.80	1.10	31.60

Table 28: **Top-N Type Distribution for Arabic on Multi-Test-1K-Corr.** The table reports the distribution of correct matches ("Good") and various error types ("Removed", "Added", "Replace_Pred", "Replace_Entity", and "Swapped") for the top 1000 retrieval results, where "Good" indicates correctly matched graphs. The results illustrate the robustness of models against corrupted graphs. **BE-MPNet (bs32; ep10)** refers to the best MPNet-based alignment model without hard negatives.

N.6 Chinese

Model	Correct	Error Types				
	Good	Removed	Added	ReplP	ReplE	Swapped
BE-MPNet (bs8; ep2)	32.20	9.10	10.30	18.70	0.60	29.10
BE-MPNet (bs16; ep2)	33.50	8.60	7.70	18.00	0.40	31.80
BE-MPNet (bs32; ep2)	33.70	7.90	7.80	19.30	0.30	31.00
BE-MPNet (bs32; ep10)	32.90	8.50	6.90	20.10	0.50	31.10
CE-MPNet (bs2; ep2)	28.90	13.20	9.60	16.20	2.20	29.90
CE-MPNet (bs3; ep2)	26.90	11.70	10.50	19.80	3.00	28.10
CE-MPNet (bs4; ep2)	23.70	8.40	16.40	20.80	5.50	25.20
CE-MPNet (bs4; ep10)	25.80	8.20	11.00	26.70	1.80	26.50

Table 29: **Top-N Type Distribution for Chinese on Multi-Test-1K-Corr.** The table reports the distribution of correct matches ("Good") and various error types ("Removed", "Added", "Replace_Pred", "Replace_Entity", and "Swapped") for the top 1000 retrieval results, where "Good" indicates correctly matched graphs. The results illustrate the robustness of models against corrupted graphs. **BE-MPNet (bs32; ep10)** refers to the best MPNet-based alignment model without hard negatives.

O Multi-Test-1K-Corr Retrieval Result

Model	Overall		EN		AR		ES		FR		RU		ZH	
	MRR	Acc.	MRR	Acc.	MRR	Acc.	MRR	Acc.	MRR	Acc.	MRR	Acc.	MRR	Acc.
BI-ENCODER														
Baselines														
MPNet	46.96	22.20	50.25	25.00	42.21	19.10	48.45	22.70	48.51	22.80	46.23	21.90	46.11	21.70
CLS_MPNet	47.61	22.68	57.05	29.10	36.91	15.90	52.52	25.40	50.86	23.80	45.65	22.10	42.63	19.80
BGE-M3	66.27	44.50	68.53	45.90	63.48	42.70	69.06	47.50	68.47	46.40	64.94	43.30	63.11	41.20
EREDAT	-	-	66.54	41.00	-	-	-	-	-	-	-	-	-	-
FactSpotter	-	-	55.77	32.70	-	-	-	-	-	-	-	-	-	-
Batch Size Variants														
BE-MPNet (bs8; ep2)	60.09	36.70	65.66	41.90	54.57	32.70	64.23	40.90	62.81	38.40	57.40	33.80	55.84	32.50
BE-MPNet (bs16; ep2)	61.72	37.27	67.38	43.40	56.79	33.30	64.49	39.80	64.27	39.50	57.89	32.40	59.52	35.20
BE-MPNet (bs32; ep2)	63.75	39.57	69.53	46.40	58.57	34.70	66.64	42.50	65.27	40.20	60.93	36.50	61.54	37.10
Base Model Variants														
BE-MPNet	64.44	40.22	69.79	47.30	58.74	34.00	67.33	43.10	66.48	42.40	61.31	36.00	63.00	38.50
BE-BGE-M3	63.15	40.22	68.71	46.40	57.52	34.20	66.50	43.90	65.75	43.70	59.87	36.50	60.56	36.60
Hard Negative Variants														
BE-MPNet-Hard1	76.06	60.33	82.75	69.90	70.35	53.40	78.69	63.40	79.22	64.00	72.00	54.70	73.34	56.60
BE-MPNet-Hard2 (ALL-6)	**78.67**	**64.92**	**84.55**	**73.50**	**72.12**	**56.30**	**81.92**	**69.40**	**81.57**	**68.90**	75.06	59.60	**76.80**	**61.80**
BE-MPNet-Hard4	76.19	62.37	81.76	69.60	69.32	53.50	79.42	66.10	79.57	66.60	73.59	59.30	73.48	59.10
BE-BGE-M3-Hard1	75.74	59.43	82.63	68.90	68.26	49.50	78.83	63.10	78.69	63.60	73.24	56.40	72.77	55.10
BE-BGE-M3-Hard2	77.69	64.00	83.24	70.70	72.01	57.40	80.87	68.00	79.61	66.10	**75.48**	**62.00**	74.90	59.80
Languages Variants With 2 Hard Negatives														
BE-MPNet-Hard2-EN	49.39	26.40	58.10	34.00	42.66	22.70	54.17	29.70	52.77	28.00	45.36	23.00	43.27	21.00
BE-MPNet-Hard2-EN-FR	64.55	45.35	75.84	58.30	52.30	33.00	71.55	52.80	72.79	54.10	58.59	37.90	56.24	36.00
BE-MPNet-Hard2-EN-ES	59.81	39.57	69.97	49.60	48.81	29.90	67.45	46.80	65.68	44.40	55.43	34.70	51.52	32.00
BE-MPNet-Hard2-EN-FR-ES	68.86	51.33	78.88	63.70	57.23	38.40	75.86	59.00	75.49	59.00	64.80	46.60	60.93	41.30
BE-MPNet-Hard2-EN-RU	65.92	45.80	73.75	54.30	57.93	37.90	69.84	49.70	68.82	47.90	64.61	44.50	61.00	40.50
BE-MPNet-Hard2-EN-ZH	69.20	50.98	77.30	61.60	60.53	41.00	70.79	51.80	72.69	54.90	65.27	46.90	68.62	49.70
BE-MPNet-Hard2-EN-AR	66.08	46.33	75.50	58.30	62.38	42.40	67.08	47.10	68.41	48.40	62.72	42.40	60.41	39.40
BE-MPNet-Hard2-EN-RU-ZH-AR	74.17	57.92	80.48	66.20	68.01	50.80	76.00	59.60	76.33	60.00	71.76	55.70	72.43	55.20
BE-MPNet-Hard2-ALL-6-Small	60.58	32.95	63.20	34.20	57.12	30.30	61.96	33.20	62.31	34.40	59.03	32.30	59.90	33.30
CROSS-ENCODER														
Batch Size Variants														
CE-MPNet (bs2; ep2)	48.90	22.73	56.98	28.30	42.07	19.90	52.58	24.10	51.36	22.40	46.11	21.50	44.30	20.20
CE-MPNet (bs3; ep2)	50.35	22.80	55.45	26.70	48.19	22.30	50.90	22.40	51.08	22.00	47.66	20.50	48.83	22.90
CE-MPNet (bs4; ep2)	44.41	17.03	47.63	17.90	42.04	16.70	44.99	16.10	45.52	16.60	44.32	18.50	41.93	16.40
Best CE Model														
CE-MPNet	49.64	19.43	55.30	24.10	43.97	13.80	52.24	21.80	52.43	22.40	46.78	17.00	47.12	17.50

Table 30: **Performance of various models in terms of Mean Reciprocal Rank (MRR) and Accuracy (Acc.) across different languages on the complex retrieval task.** Overall results represent aggregated metrics across all languages. "BE" and "CE" denote bi-encoder and cross-encoder models, respectively. `MPNet` and `BGE-M3` are pre-trained embedding models used as baselines and as backbones for other variants; `CLS-MPNet` is a binary classification baseline. **Abbreviations**: "ep" = epochs, "bs" = batch size, "HardX" = X hard negatives per graph. Unless otherwise specified, bi-encoders use batch size 32 and cross-encoders use batch size 4; they both use training epoch number 10.

P Impact of Multilingual Training on English Alignment

P.1 All-6-Small Set Creation

To disentangle the effect of data size from language coverage, we construct the *All-6-Small* training set, which matches the overall size of the English-only subset while distributing it across six languages. The subset is extracted from *EN-Train* using the same procedure as for the English *Test-1K* (see App. F), i.e., maximizing property coverage while maintaining balance across graph sizes. Specifically, we select 5,600 (Graph, English Text) pairs from *EN-Train*, covering 74.76% of the properties in the original set. This subset is then translated into the other five target languages following the same procedure as for *Multi-Train-Silver*, resulting in *All-6-Small* with 33,600 (Graph, Text) pairs across six languages (5,600 per language). Table 31 compares *All-6-Small* with other training sets, and Table 32 details the composition of the *All-6-Small* subset by graph size and source.

Training Sets	Description	Languages	#(KG, Text) Pairs	#Property
EN-Train	KELM-Q1 + WebNLG-Train	En	33,601	4,188
Multi-Train-Silver (All-6)	EN-Train + Translation	En, Fr, Es, Ru, Zh, Ar	201,606	4,188
EN-Train-Subset	Subset of EN-Train	En	5,600	3,131
All-6-Small	EN-Train-Subset + Translation	En, Fr, Es, Ru, Zh, Ar	33,600	3,131

Table 31: Statistics of training datasets. *All-6-Small* matches the size of *EN-Train* but covers six languages.

Graph Size	KELM-Q1	WebNLG-Train	Total
1	589	531	1,120
2	1,117	3	1,120
3	1,027	93	1,120
4	611	509	1,120
5	648	472	1,120
Total	3,992	1,608	5,600

Table 32: Composition of the *All-6-Small* subset by graph size and source dataset.

P.2 Full Results on English Retrieval Tasks

Table 33 reports the full results of alignment models trained on different training settings. We observe a consistent improvement in English retrieval performance as additional languages are added to the training data. Notably, the *All-6-Small* variant—which controls for training size by matching the English-only subset—achieves substantially better results than the English-only model, and even surpasses or closely matches the full *All-6* model on the two test sets without graph corruptions (*Multi-Test-1K* and *Multi-WebNLG-Test*). This confirms that the benefit arises from multilingual coverage rather than data volume. On the other hand, *All-6-Small* underperforms on *Multi-Test-1K-Corr*. A likely explanation is that the reduced training subset leads to weaker property and entity coverage, resulting in corrupted graphs that diverge more significantly from the original ones, thus hurting robustness under corruption.

Model Variants	Multi-Test-1K				Multi-WebNLG-Test				Multi-Test-1K-Corr	
	G2T		T2G		G2T		T2G		T2G	
	R@1	MRR	R@1	MRR	R@1	MRR	R@1	MRR	R@1	MRR
Languages Variants										
BE-MPNet-Hard2-EN	90.70	94.33	91.60	94.89	66.55	77.15	67.90	78.78	34.00	58.10
BE-MPNet-Hard2-EN-FR	94.40	96.32	94.50	96.18	74.76	82.48	76.17	83.58	58.30	75.84
BE-MPNet-Hard2-EN-ES	91.70	95.24	93.90	96.25	69.93	79.65	75.38	83.80	49.60	69.97
BE-MPNet-Hard2-EN-FR-ES	94.70	96.67	94.60	96.39	76.50	84.46	77.85	85.34	63.70	78.88
BE-MPNet-Hard2-EN-RU	93.70	96.06	94.60	96.53	74.48	82.73	76.11	84.13	54.30	73.75
BE-MPNet-Hard2-EN-ZH	92.90	95.13	93.30	95.23	74.20	81.84	76.17	83.20	61.60	77.33
BE-MPNet-Hard2-EN-AR	93.90	95.91	94.10	96.19	75.89	83.19	77.23	84.21	58.30	75.50
BE-MPNet-Hard2-EN-RU-ZH-AR	94.30	96.42	96.30	97.48	76.90	84.58	79.88	86.72	66.20	80.48
BE-MPNet-Hard2-All-6	95.60	97.30	96.10	97.40	**80.33**	**86.92**	**81.62**	87.65	**73.50**	**84.55**
BE-MPNet-Hard2-All-6-Small	**96.30**	**97.96**	**97.40**	**98.59**	78.98	86.34	81.17	**88.23**	34.20	63.18

Table 33: **Model Performance Comparison on Test Sets for monolingual tasks (English)**. BE: Bi-Encoder, CE: Cross-Encoder, G2T: Graph-to-Text Retrieval, T2G: Text-to-Graph Retrieval, Recall@1 (R@1), Mean Reciprocal Rank (MRR). The batch size for all BE models is 32 unless it is explicitly stated. *All-6* denotes the full six-language mix: English (EN), Arabic (AR), Chinese (ZH), French (FR), Spanish (ES), and Russian (RU). *All-6-Small* matches the size of the English-only set but covers six languages.

Q DPO Evaluation Result

Q.1 KELM Test

Model	BLEU	Meteor	ChrF	TER	BertScore	Bleurt	EREDAT	Facts	Parent	Quest-Eval	Sescore2
Prompting Baselines											
QWEN-0-shot	22.89	36.75	16.84	91.40	93.64	75.22	84.69	67.27	34.25	69.10	-6.33
QWEN-3-shot	32.33	40.23	65.89	54.49	95.04	78.95	88.72	82.22	45.09	71.85	-3.24
Instrution Tuning											
QWEN-IT	**40.91**	42.55	69.77	**42.99**	**95.77**	**81.67**	90.40	56.93	50.72	71.64	**-1.69**
QWEN DPO Variants											
DPO-HARD-MuCAL	36.60	**43.37**	69.83	54.35	95.14	78.27	92.38	91.70	52.18	71.65	-2.70
DPO-CE-MuCAL	39.56	43.24	**69.87**	46.00	95.53	79.26	91.22	90.09	**53.03**	72.19	-2.10
DPO-EREDAT	30.62	37.99	59.76	100.06	93.92	76.20	**92.59**	91.89	49.51	71.72	-2.70
DPO-FactSpotter	15.23	23.85	11.30	772.00	90.10	64.45	80.06	**96.71**	36.01	68.69	-12.90
DPO-DQE	28.71	26.41	37.60	351.48	92.65	78.52	88.57	94.05	55.42	**74.58**	-7.16

Table 34: Performance of different models on the KELM Test set across reference-based and graph-based evaluation metrics. Abbreviations: QWEN refers to QWEN2.5-1.5B-Instruct, IT stands for Instruction Tuning, and DPO-X denotes DPO training applied on preference data where the preference signals are provided by X.

Q.2 WebNLG Test

Model	BLEU	Meteor	ChrF	TER	BertScore	Bleurt	EREDAT	Facts	Parent	Quest-Eval	Sescore2
Prompting Baselines											
QWEN-0-shot	36.23	37.41	57.23	61.98	94.09	75.40	84.05	90.18	49.69	69.87	-6.58
QWEN-3-shot	42.43	38.48	63.44	51.87	94.80	77.56	85.59	91.55	55.52	71.63	-4.69
Instrution Tuning											
QWEN-IT	**45.39**	39.06	64.63	**47.22**	**95.25**	**78.28**	86.37	90.68	59.17	71.84	**-3.62**
QWEN DPO Variants											
DPO-HARD-MuCAL	42.82	**39.76**	**65.56**	51.35	94.85	77.91	89.56	95.80	**61.53**	74.61	-4.12
DPO-CE-MuCAL	44.51	39.67	65.15	48.32	95.04	78.12	88.50	93.40	59.23	73.43	-3.95
DPO-EREDAT	37.23	36.13	31.38	60.49	93.72	76.97	**90.38**	97.32	59.36	74.81	-6.21
DPO-FactSpotter	21.19	26.19	12.26	96.70	90.20	68.41	78.97	**97.69**	44.28	70.56	-13.75
DPO-DQE	23.72	23.92	35.08	88.21	89.98	76.37	82.38	95.56	59.90	**75.79**	-12.67

Table 35: Performance of different models on the WebNLG Test set across reference-based and graph-based evaluation metrics. Abbreviations: QWEN refers to QWEN2.5-1.5B-Instruct, IT stands for Instruction Tuning, and DPO-X denotes DPO training applied on preference data where the preference signals are provided by X.

Q.3 GOLD-OOD-472

Model	BLEU	Meteor	ChrF	TER	BertScore	Bleurt	EREDAT	Facts	Parent	Quest-Eval	Sescore2
Prompting Baselines											
QWEN-0-shot	26.57	34.81	31.52	96.69	93.37	72.81	79.36	83.97	41.67	59.88	-9.39
QWEN-3-shot	32.95	34.67	57.39	61.30	94.35	75.49	80.73	85.11	48.98	61.81	-6.97
Instrution Tuning											
QWEN-IT	30.46	31.17	53.65	60.53	93.88	74.63	79.81	81.01	47.08	61.95	-6.32
QWEN DPO Variants											
DPO-HARD-MuCAL	**35.23**	**41.12**	**63.40**	**59.02**	**94.88**	**77.34**	84.92	93.86	**54.93**	64.85	**-5.55**
DPO-CE-MuCAL	34.45	38.52	60.21	59.35	94.71	76.25	83.07	89.18	53.82	63.22	-5.63
DPO-EREDAT	31.67	38.58	59.51	93.57	93.98	75.30	**85.35**	94.99	54.04	**65.23**	-7.23
DPO-FactSpotter	24.52	33.46	17.37	307.95	91.58	69.74	78.89	**96.86**	45.08	62.14	-11.29
DPO-DQE	21.27	28.12	40.04	306.22	90.17	74.90	79.81	92.81	53.71	65.02	-13.80

Table 36: Performance of different models on the GOLD-OOD-472 set across reference-based and graph-based evaluation metrics. Abbreviations: QWEN refers to QWEN2.5-1.5B-Instruct, IT stands for Instruction Tuning, and DPO-X denotes DPO training applied on preference data where the preference signals are provided by X.

R Qualitative Analysis Results

R.1 QWEN-IT better than DPO-HARD-MuCAL by METEOR

	Graph	QWEN-IT (Text / METEOR)	DPO-HARD-MuCAL (Text / METEOR)	Gap	Human Observation
Ex1	[2009 UK Championship | sport | snooker]	The 2009 UK Championship was a snooker tournament. (95.79)	The 2009 UK Championship took place in snooker. (34.65)	+61.14	Both are factually correct, but IT output is more fluent and natural, while DPO sounds awkward.
Ex2	[1997 British Open | tournament type | ranking tournament]	The 1997 British Open was a ranking tournament. (100.00)	The 1997 British Open has the tournament type of ranking tournament. (45.19)	+54.81	Both outputs are correct, but IT is more concise and idiomatic; DPO is grammatical yet less natural.
Ex3	[Sweden | Happy Planet Index score | 50.5]	Sweden has a happy planet index score of 50.5. (100.00)	The happy planet index score for Sweden is 50.5. (46.87)	+53.13	Both are factually correct; IT output is simpler and reads more naturally, whereas DPO is verbose.

Table 37: **Qualitative examples where IT outperforms DPO (IT > DPO).** We show model outputs with their METEOR scores, the score gap (IT–DPO), and human observations.

R.2 DPO-HARD-MuCAL better than QWEN-IT by METEOR

	Graph	QWEN-IT (Text / METEOR)	DPO-HARD-MuCAL (Text / METEOR)	Gap	Human Observation
Ex1	[calcitriol | has use | medication; calcitriol | ATC code | D05AX03; calcitriol | canonical SMILES | C[C@H](CCCC(C)(C)O)C1CCC2C1(CCCC2=CC=C3CC(CC(C3=C)O)O)C; calcitriol | InChI | InChI=1S/C27H44O3/c1-18(8-6-14-26(3,4)30)23-12-13-24-20(9-7-15-27(23,24)5)10-11-21-16-22(28)17-25(29)19(21)2/h10-11,18,22-25,28-30H,2,6-9,12-17H2,1,3-5H3/b20-10+,21-11-/t18-,22-,23-,24+,25+,27-/m1/s1; calcitriol | chemical formula | $C_{27}H_{44}O_3$]	Calcitriol (D05AX03) has a molecular formula of $C_{27}H_{44}O_3$. It is used as a medication. (3.15)	Calcitriol has the use of medication and its ATC code is D05AX03. Its canonical SMILES is CC(CCCC(C)(C)O)C1CCC2C1(CCCC2=CC=C3CC(CC(C3=C)O)O)C and its InChI is InChI=1S/C27H44O3/c1-18(8-6-14-26(3,4)30)23-12-13-24-20(9-7-15-27(23,24)5)10-11-21-16-22(28)17-25(29)19(21)2/h10-11,18,22-25,28-30H,2,6-9,12-17H2,1,3-5H3/b20-10+,21-11-/t18-,22-,23-,24+,25+,27-/m1/s1 and its chemical formula is $C_{27}H_{44}O_3$. (59.40)	−56.25	**DPO** is markedly better by explicitly covering ATC code, SMILES, and InChI, whereas **IT** omits most identifiers.
Ex2	[Antigua and Barbuda | official language | English]	English is an official language of Antigua and Barbuda. (46.53)	The official language of Antigua and Barbuda is English. (100.00)	−53.47	Both are correct; **DPO** is preferable due to precise wording *the official language* while **IT** uses *an official language*, introducing ambiguity.
Ex3	[valsartan | has use | medication; valsartan | canonical SMILES | CCCCC(=O)N(CC1=CC=C(C=C1)C2=CC=CC=C2C3=NNN=N3)C(C(C)C)C(=O)O; valsartan | InChI | InChI=1S/C24H29N5O3/c1-4-5-10-21(30)29(22(16(2)3)24(31)32)15-17-11-13-18(14-12-17)19-8-6-7-9-20(19)23-25-27-28-26-23/h6-9,11-14,16,22H,4-5,10,15H2,1-3H3,(H,31,32)(H,25,26,27,28)/t22-/m0/s1; valsartan | chemical formula | $C_{24}H_{29}N_5O_3$]	The molecular formula for valsartan is $C_{24}H_{29}N_5O_3$. It is used as an anti-hypertensive drug. Its SMILES code is: CC(C1=C(NCCC[C@H](https://www.notion.so/CN)=O)C2=C1NC3=C(NC=CC=C3)C=C1)C(O)OC(C)OC. (12.36)	The valsartan has a use for medication and its canonical SMILES is CCCCC(=O)N(CC1=CC=C(C=C1)C2=CC=CC=C2C3=NNN=N3)C(C(C)C)C(=O)O. Its InChI is InChI=1S/C24H29N5O3/c1-4-5-10-21(30)29(22(16(2)3)24(31)32)15-17-11-13-18(14-12-17)19-8-6-7-9-20(19)23-25-27-28-26-23/h6-9,11-14,16,22H,4-5,10,15H2,1-3H3,(H,31,32)(H,25,26,27,28). Its chemical formula is $C_{24}H_{29}N_5O_3$. (57.62)	−45.26	**IT** includes extraneous/incorrect SMILES content; **DPO** is more accurate and complete with correct canonical SMILES and InChI. Overall, **DPO** is preferred.

Table 38: **Qualitative examples where DPO outperforms IT (DPO > IT).** Colored highlights mark key facts generated by DPO but omitted or weaker in IT: ATC code (blue), SMILES (red, monospaced) and InChI (green, monospaced). We show model outputs with their METEOR scores, the score gap (IT–DPO; negative means DPO higher), and human observations.