Natalia Skachkova, Simon Ostermann, Josef van Genabith, Bernd Kiefer

Do LLMs fail in bridging generation?

Abstract

In this work we investigate whether large language models (LLMs) ‘understand’ bridging relations and can use this knowledge effectively. We present the results obtained from two tasks: generation of texts containing bridging and filling in missing bridging spans. We show that in most of the cases LLMs fail to generate bridging in a reliable way.

1 Introduction

Bridging resolution is the task of linking mentions, which are text spans typically representing entities or events, based on some associative relation, such as *part-whole*, *set-member*, *object-attribute*, etc. (Clark, 1975; Poesio, 2004; Poesio, Vieira, & Teufel, 1997). E.g., in the sentence *“The only indication it is* ***a motel*** *is* ***a sign with a faded picture of a locomotive****.”* the parts in ***bold*** represent the whole (***a motel***) and its part (***a sign ... locomotive***), and are called an antecedent and an anaphor, respectively.

Bridging resolution is a challenging task - the current state-of-the-art end-to-end bridging resolution model by Kobayashi, Hou, and Ng (2023) reaches maximum 26.2 F1 score on the ISNotes dataset (Markert, Hou, & Strube, 2012). One of the reasons for such a poor performance is a lack of training data - manual annotation of bridging is difficult and costly (Poesio et al., 2018). A potentially promising alternative would be to create more data using an LLM. In this paper we investigate how much LLMs ‘know’ about bridging and whether they can apply this knowledge to generate new data. Our contributions are two-fold.

- We prompt the *text-davinci-003* model (OpenAI, 2023) to generate 1,000 texts with bridging, and manually investigate in how many of them the relation holds. We show that the model fails to generate texts with bridging in a reliable way.
- We use 13 LLMs to fill in missing bridging antecedents, anaphors, or both of them, and compare the generated spans with the gold ones using a semantic similarity metric. We provide evidence that LLMs have some knowledge of bridging, but often fail to apply it correctly, or ‘avoid’ using it. We also demonstrate that bridging knowledge contained in LLMs is difficult to extract and quantify.

2 Related Work

Investigation of LLMs’ capabilities of language ‘understanding’, as well as estimation of the amount of knowledge they possess, are active research areas. There is evidence that

LLMs have commonsense knowledge (Bubeck et al., 2023; X. L. Li et al., 2022), can infer latent concepts from textual pre-training (Jin & Rinard, 2024) and capture structural semantics (Cheng et al., 2024). On the other hand, Bian et al. (2023); Z. Li et al. (2024); Zhu et al. (2023) and Saba (2024) show that the reasoning and 'understanding' capabilities of LLMs are often exaggerated. While to the best of our knowledge there are no studies on LLMs and bridging relations, there exist some works focusing on the ability of LLMs to capture related phenomena. Dos Santos and Leal (2024), apply different models to assess the strength of semantic similarity between the word pairs, and come to the conclusion that LLMs' predictions correlate with the scores from human annotators. A similar study is conducted by De Deyne, Liu, and Frermann (2024) who use GPT-4 (OpenAI, 2025) to infer semantic relations for human-produced word associations. They find out that the model is good at identification of broad relations, but struggles with more fine-grained ones. Hu, Mahowald, Lupyan, Ivanova, and Levy (2024) investigate the extend to which LLMs can differentiate between grammatical and ungrammatical sentences. They provide evidence that the models' grammaticality judgments align with human intuitions across a range of linguistic phenomena, including anaphora.

3 Data

For our study, we use the ARRAU 2 RST corpus (Poesio et al., 2018), as it is one of the largest corpora annotated with bridging relations, and is often used for benchmarking bridging resolution systems. ARRAU 2 RST is a subset of the Penn Treebank (Marcus, Santorini, & Marcinkiewicz, 1993) and belongs to the news domain. Dataset statistics and examples of bridging relations can be found in Appendix A.1.

In total, ARRAU 2 RST contains 3,777 bridging pairs. For our experiments, we use the training partition of the dataset and construct the data as follows. We first exclude cases in which the anaphor is part of the antecedent, as we assume that nested spans would be particularly challenging both for the LLM to generate and for us to explain in the prompt. This filtering step yields 2,721 pairs. This subset is used for the first experiment with *text-davinci-003*. Second, for the sake of time and computational efficiency - and to further simplify the task for the models - we limit the number of bridging pairs used in the subsequent experiments. Specifically, we exclude pairs in which the distance between the anaphor and its antecedent exceeds ten whitespace-separated tokens. This results in a set of 554 bridging pairs, which are not necessarily unique. The distribution of bridging relation types in this subset is provided in Appendix A.1. Since each document (or sentence) in ARRAU 2 RST may contain multiple bridging pairs, different pairs may share the same context. To reduce context length, we truncate the text by removing all sentences to the left of the one containing the antecedent and all sentences to the right of the one containing the anaphor. This results in a maximum sequence length of 148 whitespace-separated tokens. Thus, in the second experimental setting, we deliberately focus on bridging spans that occur close to each other in the text. We hypothesize that such spans are significantly easier for LLMs to resolve compared to long-distance and/or nested spans.

Notably, 85 out of the 554 bridging pairs (15.34%) exhibit syntactic head overlap between the antecedent and the anaphor, as in: *“The Labor Department said wage increases in **manufacturing industries** continue to be smaller than those in **other industries** .”*

We do not know whether ARRAU 2 RST was used in the training of any LLMs. Therefore, we cannot rule out the possibility of data leakage.

4 Generating texts with bridging

Experiment. We start with an experiment, where we use *text-davinci-003* to generate short texts with bridging. To do the task, the model receives a definition of bridging, an instruction, three demonstrations and a new bridging pair to construct a text with. The demonstrations, as well as the target bridging pair are chosen randomly from the 2,721 ARRAU 2 RST pairs/texts. To identify both antecedent and anaphor in the text, we ask the model to mark them with the "*" symbol on both sides. The prompt is shown in Example 4.1. During text generation, we filter out all texts not following the specified pattern, namely those that have too many or too few "*" symbols. The generation process is executed until we collect 1,000 well-formed texts. Next, we manually check if a bridging relation holds in each text.

Example 4.1. *"Bridging is a relation of anaphoric references to non-identical associated antecedents. Bridging covers, for example, part-of, subset, set membership, and possession relations. Make a short text in the style of news with the given words keeping the bridging relation between them.*
*Words: * **40 people , or about 15 %** * and * **the personnel** *.*
*Text: Telxon Corp. said its vice president for manufacturing resigned and its Houston work force has been trimmed by * **40 people , or about 15 %** * . The maker of hand-held computers and computer systems said * **the personnel** * changes were needed to improve the efficiency of its manufacturing operation .*
{two more examples}
*Words: * **Federal Reserve banks** * and * **branches** *.*
Text:"

Results. Our analysis shows that only 24.4% of all the texts include correct examples of bridging. Another 22.1% represent cases, where the boundaries of the original bridging pairs need to be modified for the bridging relations to hold. The rest (53.5%) do not contain any bridging relations, despite the fact that the given bridging pairs are present in the generated texts. Example 4.2 is a good illustration of the most common problems that occur when using *text-davinci-003* for the task. First, instead of an associative relation between the spans, we have an explicit one (cf. gold text in the same example). Second, the spans’ boundaries need to be corrected.

Example 4.2. *Gold vs generated texts*
GOLD: Tenders for the bills , available in minimum $ 10,000 denominations , must be received by 1 p.m. EST Monday at the Treasury or at ***Federal Reserve banks*** *or* ***branches*** *.*
*GENERATED: The United States's * **Federal Reserve Banks** * are divided into 12 * **branches** * , each of which holds assets and liabilities of the original Federal Reserve Bank and serves to influence the nation's growth by controlling monetary production and circulation.*

Discussion. Although the model's failure in more than half of the cases may be attributed to factors such as suboptimal prompting, inadequate demonstrations, or the inherent difficulty of the task, we hypothesize that the primary reason is that *text-davinci-003* struggles to genuinely 'understand' bridging. As a result, it cannot reliably use bridging in context, even if it may be capable of explaining how two bridging spans are related. As *text-davinci-003* is currently deprecated, we conduct an experiment using *Falcon-40B* to assess whether this may be the case. We extend the prompt in Example 4.1 with an additional instruction requiring the model to provide an explanation of why a bridging relation holds in the generated text. The results indicate that while the model knows the definition of bridging and can explain the relation between two spans, it still frequently fails to generate text that correctly instantiates this relation. The full prompt, along with representative examples of generated texts and explanations, is provided in Appendix A.2. Although *Falcon-40B* cannot be directly compared to *text-davinci-003*, we hypothesize that the latter would likely exhibit similar behavior.

5 Fill-in-the-gap task

Experiments. To evaluate how well LLMs utilize their knowledge of bridging, we design the following task. For each of the 554 short texts, we successively mask the antecedent, the anaphor, and both spans simultaneously. The LLM is then prompted to process each of the three resulting texts with different types of gaps and to recover the missing spans.

The prompt (see Example 5.1), which is identical across all models, includes four demonstrations. While some LLMs exhibit strong zero- or one-shot capabilities, others may require additional examples to effectively 'understand' the task and produce the desired answer format. Based on our experiments, we found that four demonstrations are optimal for this task. For each masked span or pair, the demonstrations are selected from the remaining 553 gold instances, prioritizing those with the highest semantic similarity to the target spans. To ensure diversity, the spans to be recovered are never identical to those in the demonstrations. Semantic similarity between spans is computed using Sentence-BERT (Reimers & Gurevych, 2019), with similarity scores calculated exclusively on the spans themselves, excluding surrounding context. Notably, the prompt omits both the definition of bridging and any explicit instruction to generate it, as we aim to evaluate how often an LLM can independently infer bridging relations.

Example 5.1. *"You are a helpful AI assistant for filling in the gaps in the text. You are given a text containing [MASK] tokens. Replace each [MASK] token with a suitable word.*
Text with gaps: *She also frequently invites directors , producers , actors , [MASK] and [MASK] [MASK] [MASK] [MASK] for coffee and clips .*
Recovered phrases:
writers
other show business people
Recovered text: *She also frequently invites directors , producers , actors , writers and other show business people for coffee and clips.*
{three more examples}
Text with gaps: *The show , one of five new [MASK] series , is the second casualty of [MASK] [MASK] [MASK] so far this fall .*
Recovered phrases: "

We experiment with publicly available instruct/chat LLMs from different model families and of different sizes, such as *Command* (35B and 104B) (Cohere, 2024), *Falcon* (7B and 40B) (Almazrouei et al., 2023), *Llama3* (8B and 70B) (Grattafiori et al., 2024), *Mistral* (7B and 123B) (Jiang et al., 2023), *Qwen* (7B, 32B and 72B) (Qwen et al., 2025) and *Yi* (9B and 34B) (Young et al., 2025). The full versions are specified in Appendix A.6.

As an exact string match is not suitable for our task, we compare the LLM-generated spans with the original masked spans using a modified version of the BERTScore semantic similarity metric (Zhang, Kishore, Wu, Weinberger, & Artzi, 2020). The rationale for this choice, along with details of the modification, is provided in Appendix A.4. To obtain the lower bounds/baselines, we replace the original bridging pairs with the random, least and most similar pairs (spans) taken from the whole 'pool' of gold bridging pairs in the dataset. Additionally, to assess whether LLMs possess more knowledge about bridging than smaller pre-trained language models, we perform the same recovery task using the encoder-decoder model T5-large (Raffel et al., 2020) and the masked language model DeBERTa-large (He, Liu, Gao, & Chen, 2021).

We formulate the following hypotheses. If an LLM possesses some knowledge about bridging and is able to use it, then 1) the BERT score between the generated spans and the gold ones should be higher than the scores achieved by the baselines and 2) it should be easier for the model to recover one missing span (antecedent or anaphor), than both, i.e. the semantic similarity score should be lower in the latter case.

Importantly, LLMs sometimes produce outputs that do not conform to the format specified in the prompt. For example, a model may generate additional text, return only a single recovered span when two are expected, or omit the recovered spans entirely. When the generated spans cannot be reliably extracted, we insert a dummy span marked as *". . . "* to fill the gap. The number of such invalid outputs produced by each model is reported in Table 5 in Appendix A.5.

Results. Figure 1 presents the BERT scores (F1) between predicted and gold spans for three types of gaps. As expected, larger models typically achieve better results. Interestingly, *Qwen* and *Yi* seem to do the task better than other models of similar sizes, with *Qwen-32B* achieving results comparable to those of larger models. All LLMs beat the *DeBERTa-large*, *Random* and *Least-sim* baselines easily, but only really large ones (70B-123B) surpass *T5-large* and can compete with *Most-sim*, especially in the case where both spans are to be restored. The *paired t-test* shows that while 70B-123B LLMs, as well as *Qwen-32B*, reach significantly higher BERT scores than *Most-sim* when recovering missing antecedents and anaphors, the difference to this baseline when restoring both spans is not statistically significant. Thus, our first hypothesis is only partially supported.

Figure 1 shows stronger evidence for the second hypothesis. Namely, for all the LLMs, except *Mistral-7B*, it is more difficult to restore two spans, rather than one, and the difference between the scores is statistically significant. Also, most LLMs tend to struggle more with recovering antecedents, rather than anaphors, which was also confirmed by the paired t-tests. More details can be found in Table 5 in Appendix A.5.

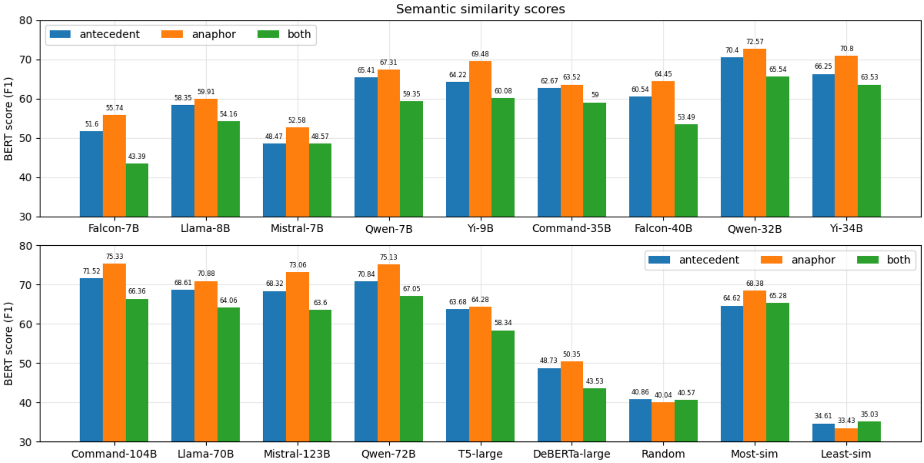

Figure 1: BERT scores between gold and predicted spans

To assess model confidence, we compute average perplexity scores for each LLM across 554 texts, evaluating five span types (predicted, gold, most/least similar, random) and three masked slot types (antecedent, anaphor, both). Detailed results appear in Appendix A.7. Perplexity patterns support prior findings: models are less confident when recovering both slots, with random spans yielding the highest perplexities. Predicted and gold spans are rated more probable than others, with predicted spans generally having lower perplexity. Larger models (except *Llama* and *Qwen*) tend to demonstrate lower perplexities than smaller ones in the same family. However, lower perplexity does not always align with higher semantic similarity - for instance, *Yi-9B* has higher perplexity than *Falcon-7B* but achieves better BERT scores.

Discussion. While Figure 1 provides evidence that LLMs (at least very large ones) may 'know' what bridging is, and can use this knowledge to some extent, it is also important to note that a high similarity score between generated and gold spans does not necessarily guarantee that the bridging relation is preserved or that the generated text is coherent and grammatically correct. Conversely, a low similarity score does not definitively indicate the absence of a bridging relation in the generated pair.

To investigate whether a high BERT score corresponds to correctly generated bridging pairs, we ask two annotators to manually evaluate 100 randomly selected pairs generated by *Qwen-72B* (one of the best-performing models) and 100 randomly selected pairs produced by *Llama3-8B* (lower-scoring). The annotators assess the number of text sequences in which the bridging relation was preserved. Manual inspection reveals that despite relatively high BERT scores, *Qwen-72B* generates bridging only in 35% of cases on average. Another 9% can be classified as bridging, but have incorrect boundaries (see Example A.10 in Appendix A.3). The performance of *Llama3-8B* is notably lower: bridging relations are found in only 16% of generated pairs, with another 7% potentially classifiable as bridging if the span boundaries were predicted correctly. Inter-annotator agreement, measured by Cohen's Kappa (Cohen, 1960), is 0.41 (moderate) and within the typical range for bridging annotation; for example, Poesio and Vieira (1998) report Cohen's Kappa values between 0.31 and 0.59 for the annotation of definite noun phrases as being in bridging relation or not.

Prompt	Antecedent	Anaphor	Both
no bridging	70.84	75.13	67.05
bridging	72.42	76.32	67.53

Table 1: BERT scores (F1) for 3 types of slots.

To some extent, the low proportion of bridging among the generated spans can be explained by the fact that many masked spans do not necessarily require bridging for the text to be coherent and correct (see Example A.11 in Appendix A.3). Since the prompt does not explicitly instruct the model to generate bridging relations, a model may tend to choose 'easier' candidates to fill in the gaps. To verify this assumption, we repeat the fill-in-the-gap experiment with *Qwen-72B*, augmenting the prompt with a definition of bridging and an explicit instruction to generate bridging. As shown in Table 1, this yields slight improvements; however, the differences are statistically insignificant. Consequently, we conclude that an explicit directive to produce bridging relations does not effectively guide *Qwen-72B* toward the desired behavior. Similar experiments with other models are left for future work.

The models' difficulty in recovering bridging relations may also be influenced by characteristics of the dataset, such as the frequent presence of personal names and numerical expressions as markables, which are challenging for models to reproduce accurately. Additionally, bridging markables and relation types are not consistently defined and vary across datasets (Kobayashi & Ng, 2020). Therefore, we hypothesize that our results may have limited generalizability.

The low number of recovered bridging pairs may also reflect the inherent difficulty of the task. It is well-known that annotating bridging relations is challenging (Poesio et al., 2018; Poesio & Vieira, 1998). However, to our knowledge, no prior studies have investigated human performance on tasks involving filling in missing bridging spans or composing texts based on bridging pairs. For a more rigorous evaluation of LLM capabilities, it would be valuable to compare their performance on these tasks with that of human participants.

Finally, employing the same prompt - albeit concise and simple - for all models may be suboptimal and could contribute to less accurate results. As Mizrahi et al. (2024) highlight, model performance can vary significantly across different instruction paraphrases. Therefore, we plan to conduct a multi-prompt evaluation in future work to ensure robustness.

6 Conclusion

In this paper, we investigated to what extent LLMs ‘understand’ bridging and whether we can use this knowledge for data generation. As our analysis covers only a very small portion of the spans generated by LLMs, it is difficult to draw simple and clear conclusions. Based on the experiments’ results, we observe the following trends.

First, bridging remains a highly challenging phenomenon for LLMs, including those with 70B to 123B parameters. Our experiments demonstrate that while such models possess some degree of ‘understanding’ of bridging, they frequently fail to apply this knowledge effectively. Consequently, their use for reliably generating texts with bridging relations is limited.

Second, measuring bridging is inherently difficult. We observed that many masked gaps can be plausibly filled with non-bridging spans, making it challenging to determine whether an LLM fails due to lack of knowledge or simply opts for simpler candidates. The absence of reliable metrics for identifying bridging further complicates evaluation.

Finally, our preliminary findings require validation on additional bridging datasets, preferably focusing on better-defined subsets of bridging relations. Furthermore, multi-prompt evaluations and comparisons with human performance are necessary to support or refute the trends observed in our initial experiments.

Acknowledgments

The research in this paper has been funded by the Horizon Europe project Fluently. Grant ID: 101058680.

References

Almazrouei, E., Alobeidli, H., Alshamsi, A., Cappelli, A., Cojocaru, R., Debbah, M., . . . Penedo, G. (2023). *The Falcon Series of Open Language Models.* Retrieved from https://arxiv.org/abs/2311.16867

Bian, N., Han, X., Sun, L., Lin, H., Lu, Y., & He, B. (2023). ChatGPT Is a Knowledgeable but Inexperienced Solver: An Investigation of Commonsense Problem in Large Language Models. *ArXiv*, *abs/2303.16421*. Retrieved from https://api.semanticscholar.org/CorpusID:257804619

Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E., . . . Zhang, Y. (2023). *Sparks of Artificial General Intelligence: Early experiments with GPT-4.* Retrieved from https://arxiv.org/abs/2303.12712

Cheng, N., Yan, Z., Wang, Z., Li, Z., Yu, J., Zheng, Z., . . . Han, W. (2024). *Potential and Limitations of LLMs in Capturing Structured Semantics: A Case Study on SRL.* Retrieved from https://arxiv.org/abs/2405.06410

Clark, H. H. (1975). Bridging. In *Proceedings of the 1975 workshop on theoretical issues in natural language processing* (p. 169–174). USA: Association for Computational Linguistics. Retrieved from https://doi.org/10.3115/980190.980237 doi: 10.3115/980190.980237

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and psychological measurement*, *20*(1), 37–46.

Cohere. (2024). *Command R: Retrieval-Augmented Generation at Production Scale.* https://cohere.com/blog/command-r. (Accessed: January, 2025)

De Deyne, S., Liu, C., & Frermann, L. (2024, May). Can GPT-4 Recover Latent Semantic Relational Information from Word Associations? A Detailed Analysis of Agreement with Human-annotated Semantic Ontologies. In M. Zock, E. Chersoni, Y.-Y. Hsu, & S. de Deyne (Eds.), *Proceedings of the workshop on cognitive aspects of the lexicon @ lrec-coling 2024* (pp. 68–78). Torino, Italia: ELRA and ICCL. Retrieved from https://aclanthology.org/2024.cogalex-1.8/

Dos Santos, A. F., & Leal, J. P. (2024). Early Findings in Using LLMs to Assess Semantic Relations Strength (Short Paper). In *Slate.* Retrieved from https://api.semanticscholar.org/CorpusID:274024193

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., et al. (2024). *The Llama 3 Herd of Models.* Retrieved from https://arxiv.org/abs/2407.21783

He, P., Liu, X., Gao, J., & Chen, W. (2021). *DeBERTa: Decoding-enhanced BERT with Disentangled Attention.* Retrieved from https://arxiv.org/abs/2006.03654

Hu, J., Mahowald, K., Lupyan, G., Ivanova, A., & Levy, R. (2024, August). Language models align with human judgments on key grammatical constructions. *Proceedings of the National Academy of Sciences*, *121*(36). Retrieved from http://dx.doi.org/10.1073/pnas.2400917121 doi: 10.1073/pnas.2400917121

Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., de las Casas, D., . . . Sayed, W. E. (2023). *Mistral 7B.* Retrieved from https://arxiv.org/abs/2310.06825

Jin, C., & Rinard, M. (2024). *Latent Causal Probing: A Formal Perspective on Probing with Causal Models of Data.* Retrieved from https://arxiv.org/abs/2407.13765

Kobayashi, H., Hou, Y., & Ng, V. (2023, July). PairSpanBERT: An enhanced language model for bridging resolution. In A. Rogers, J. Boyd-Graber, & N. Okazaki (Eds.), *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 6931–6946). Toronto, Canada: Association for Computational Linguistics. Retrieved from https://aclanthology.org/2023.acl-long.383 doi: 10.18653/v1/2023.acl-long.383

Kobayashi, H., & Ng, V. (2020, December). Bridging resolution: A survey of the state of the art. In D. Scott, N. Bel, & C. Zong (Eds.), *Proceedings of the 28th international conference on computational linguistics* (pp. 3708–3721). Barcelona, Spain (Online): International Committee on Computational Linguistics. Retrieved from https://aclanthology.org/2020.coling-main.331/ doi: 10.18653/v1/2020.coling-main.331

Li, X. L., Kuncoro, A., Hoffmann, J., de Masson d'Autume, C., Blunsom, P., & Nematzadeh, A. (2022, December). A Systematic Investigation of Commonsense Knowledge in Large Language Models. In Y. Goldberg, Z. Kozareva, & Y. Zhang (Eds.), *Proceedings of the 2022 conference on empirical methods in natural language processing* (pp. 11838–11855). Abu Dhabi, United Arab Emirates: Association for Computational Linguistics. Retrieved from https://aclanthology.org/2022.emnlp-main.812/ doi: 10.18653/v1/2022.emnlp-main.812

Li, Z., Cao, Y., Xu, X., Jiang, J., Liu, X., Teo, Y. S., … Liu, Y. (2024). LLMs for Relational Reasoning: How Far are We? In *Proceedings of the 1st international workshop on large language models for code* (p. 119–126). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3643795.3648387 doi: 10.1145/3643795.3648387

Marcus, M. P., Santorini, B., & Marcinkiewicz, M. A. (1993). Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, *19*(2), 313–330. Retrieved from https://aclanthology.org/J93-2004/

Markert, K., Hou, Y., & Strube, M. (2012, July). Collective Classification for Fine-grained Information Status. In H. Li, C.-Y. Lin, M. Osborne, G. G. Lee, & J. C. Park (Eds.), *Proceedings of the 50th annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 795–804). Jeju Island, Korea: Association for Computational Linguistics. Retrieved from https://aclanthology.org/P12-1084/

Mizrahi, M., Kaplan, G., Malkin, D., Dror, R., Shahaf, D., & Stanovsky, G. (2024). State of What Art? A Call for Multi-Prompt LLM Evaluation. *Transactions of the Association for Computational Linguistics*, *12*, 933–949. Retrieved from https://aclanthology.org/2024.tacl-1.52/ doi: 10.1162/tacl_a_00681

OpenAI. (2023). *GPT-3.5 text-davinci-003 model.* https://platform.openai.com. (Accessed: October, 2023)

OpenAI. (2025). *GPT-4.* https://platform.openai.com. (Accessed: February, 2025)

Poesio, M. (2004, July). Discourse annotation and semantic annotation in the GNOME corpus. In *Proceedings of the workshop on discourse annotation* (pp. 72–79). Barcelona, Spain: Association for Computational Linguistics. Retrieved from https://aclanthology.org/W04-0210/

Poesio, M., Grishina, Y., Kolhatkar, V., Moosavi, N., Roesiger, I., Roussel, A., . . . Zinsmeister, H. (2018, June). Anaphora resolution with the ARRAU corpus. In M. Poesio, V. Ng, & M. Ogrodniczuk (Eds.), *Proceedings of the first workshop on computational models of reference, anaphora and coreference* (pp. 11–22). New Orleans, Louisiana: Association for Computational Linguistics. Retrieved from https://aclanthology.org/W18-0702 doi: 10.18653/v1/W18-0702

Poesio, M., & Vieira, R. (1998). A corpus-based investigation of definite description use. *Computational Linguistics*, *24*(2), 183–216. Retrieved from https://aclanthology.org/J98-2001/

Poesio, M., Vieira, R., & Teufel, S. (1997). Resolving bridging references in unrestricted text. In *Operational factors in practical, robust anaphora resolution for unrestricted texts.* Retrieved from https://aclanthology.org/W97-1301/

Qwen, Yang, A., Yang, B., Zhang, B., Hui, B., Zheng, B., . . . Qiu, Z. (2025). *Qwen2.5 Technical Report.* Retrieved from https://arxiv.org/abs/2412.15115

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., . . . Liu, P. J. (2020). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research*, *21*(140), 1-67. Retrieved from http://jmlr.org/papers/v21/20-074.html

Reimers, N., & Gurevych, I. (2019, 11). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 conference on empirical methods in natural language processing.* Association for Computational Linguistics. Retrieved from https://arxiv.org/abs/1908.10084

Saba, W. S. (2024). LLMs' Understanding of Natural Language Revealed. *ArXiv*, *abs/2407.19630*. Retrieved from https://api.semanticscholar.org/CorpusID:271533981

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., . . . Rush, A. M. (2020). *HuggingFace's Transformers: State-of-the-art Natural Language Processing.* Retrieved from https://arxiv.org/abs/1910.03771

Young, A., Chen, B., Li, C., Huang, C., Zhang, G., Zhang, G., . . . Dai, Z. (2025). *Yi: Open Foundation Models by 01.AI.* Retrieved from https://arxiv.org/abs/2403.04652

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2020). BERTScore: Evaluating Text Generation with BERT. In *International conference on learning representations.* Retrieved from https://openreview.net/forum?id=SkeHuCVFDr

Zhu, Y., Wang, X., Chen, J., Qiao, S., Ou, Y., Yao, Y., . . . Zhang, N. (2023). LLMs for Knowledge Graph Construction and Reasoning: Recent Capabilities and Future Opportunities. *World Wide Web (WWW)*, *27*, 58. Retrieved from https://api.semanticscholar.org/CorpusID:258833039

A Appendix

A.1 ARRAU 2 RST

All noun phrases in the ARRAU 2 RST dataset are considered markables, which can be referring or non-referring (expletive, quantificational, or predicative), see Table 2 for statistics. Bridging relations are annotated between referring markables and classified into five types, as shown in Table 3. Four of them, namely *possessive*, *subset*, *element* and *other* also have inverse variants. The *undersp-rel* relation is for cases that do not fit into the previous four categories. Examples A.1-A.5 illustrate all five relation types.

documents	413
tokens	228,901
avg. doc length (tok)	554.2
markables	72,013
avg. markables per doc	174.4
non-referring markables	9,552 (13.3%)

Table 2: ARRAU 2 RST corpus statistics (from Poesio et al. (2018)).

poss / poss-inv	87 / 25
subset / subset-inv	1,092 / 368
element / element-inv	1,126 / 152
other / other-inv	332 / 7
undersp-rel	588
total	3,777

Table 3: Distribution of bridging in ARRAU 2 RST (from Poesio et al. (2018)).

Table 4 presents the distribution of bridging relation types among the 554 pairs selected for the fill-in-the-gap task. The distribution broadly reflects that of the full dataset, with the notable exception of a higher proportion of relations labeled as *other*. Interestingly, some pairs are annotated with an *unknown* relation, which is not documented in the dataset paper by Poesio et al. (2018).

element	177 (31.95%)
other	124 (22.38%)
subset	101 (18.23%)
undersp-rel	64 (11.55%)
subset-inv	37 (6.68%)
poss	18 (3.25%)
element-inv	16 (2.89%)
unknown	11 (1.99%)
poss-inv	5 (0.90%)
other-inv	1 (0.18%)

Table 4: Distribution of bridging relations in the ARRAU 2 RST subset for the filling-in-the-gap task.

Example A.1. *'Possessive' relation*
Shearson Lehman Hutton Inc. said it applied to Taiwanese securities officials for ***permission to open brokerage offices in Taipei*** *.* ***Shearson 's application*** *is the first since the Taiwan Securities and Exchange Commission announced June 21 that it would allow foreign brokerage firms to do business in that country .*

Example A.2. *'Subset' relation*
Oil stocks *escaped the brunt of Friday 's selling and* ***several*** *were able to post gains , including Chevron , which rose 5 to 66 3 in Big Board composite trading of 2.4 million shares .*

Example A.3. *'Element' relation*
Elsewhere in ***the oil sector*** *,* ***Exxon*** *rallied 7 to 45 3 ;* ***Amoco*** *rose 1 to 47 ;* ***Texaco*** *was unchanged at 51 3 , and* ***Atlantic Richfield*** *fell 1 5 to 99 1 .*

Example A.4. *'Other' relation*
The precious metals sector *outgained* ***other Dow Jones industry groups*** *by a wide margin for the second consecutive session .*

Example A.5. *'Underspecified' relation*
Taiwan officials are expected to review ***the Shearson application*** *later this year .* ***The new rules*** *will allow investors to buy foreign stocks directly .*

The following characteristics of the ARRAU 2 RST bridging markables are important for our task, as they pose considerable challenges for bridging resolution. First, many of the markables are personal names, e.g., *'Turner Broadcasting System Inc.'*, *'Viacom Pictures'* or *'NBC'*. Some represent an amount of money, like *'53.1 million'* or *'the $200 million portion of the offering'* or other numerical expressions, e.g., *'one brown two-year-old filly'*. Next, the average lengths of an antecedent and an anaphor in the subset used for filling in the gaps task are 4.05 and 3.90 tokens, respectively. However, about 11.01% of the antecedents and 8.84% of the anaphors are longer than 10 tokens. Typically, they contain long relative clauses, e.g., *'Union Carbide, whose third-quarter earnings dropped about 35% from a year earlier and fell short of analysts' expectations'*.

A.2 Generating texts with bridging using reasoning

To support the hypothesis that LLMs may 'know' the definition of bridging, be able to 'explain' the bridging relation between the two spans and still fail to generate a short text where this relation would hold, we present three examples from our experiments with *Falcon-40B*. The prompt we use is based on the prompt for *text-davinci-003* and is given in Example A.6.

Example A.6. *"Bridging is a relation of anaphoric references to non-identical associated antecedents. Bridging covers, for example, part-of, subset, set membership, and possession relations. Make a short text in the style of news with the given words keeping the bridging relation between them. Explain why there is a bridging relation between the given words in the text.*
Words: **** 40 people , or about 15 % **** *and* **** the personnel ****.
Text: Telxon Corp. said its vice president for manufacturing resigned and its Houston work force has been trimmed by **** 40 people , or about 15 % **** *. The maker of hand-held computers and computer systems said* **** the personnel **** *changes were needed to improve the efficiency of its manufacturing operation .*
*Reasoning: There is a bridging relation between * 40 people, or about 15 % * and * the personnel * because * the personnel * introduces a broader group from which the specific subset * 40 people * is later identified.*
{two more examples}

Words: **** Federal Reserve banks **** *and* **** branches *.***
Text:"

As Examples A.7-A.9 illustrate, *Falcon-40B* typically generates a plausible explanation for why the given spans are in a bridging relation. In Example A.7 the generated text aligns with the explanation, although the span boundaries are slightly incorrect. In Example A.8 the span ** wall ** is missing and bridging relation is not preserved. A similar issue occurs in Example A.9 with the span ** thigh **.

Example A.7.
Words: **** leg **** *and* **** tripod ****
*Text: The chair was found to have a loose * leg *, causing the customer to trip when they stood up. As a result, the company issued a recall for all * tripods * on the chair.*
*Reasoning: * Leg * and * tripod * are in a bridging relation because * leg * is a specific example of a type of support, and * tripod * is a particular type of leg structure that has three legs.*

Example A.8.
Words: **** wall **** *and* **** building ****
*Text: Due to the extreme weather conditions, several walls of the old building have collapsed, causing significant damage. Engineers are currently working on plans for repairing and rebuilding the * building *.*
*Reasoning: * Wall * and * building * are in a bridging relation because * wall * is a specific element or part of a building that serves as a support for the structure, while * building * refers to the entire structure.*

Example A.9.
Words: **** thighbone **** *and* **** thigh ****
*Text: Doctors recommend regular exercise to improve mobility and reduce pain in the * thighbone * and the surrounding areas.*
*Reasoning: * Thighbone * and * thigh * are in a bridging relation, because * thighbone * is a part of the body and * thigh * is a part of the leg, and both parts are identified by their locations.*

A.3 Fill-in-the-gap task: Typical mistakes

Examples A.10 and A.11 illustrate two typical types of mistakes made by LLMs when performing the fill-in-the-gap task.

Example A.10. *Wrong span boundaries*
PREDICTED: Once inside , she spends nearly four hours measuring and diagramming each room in the 80-year-old house , gathering enough information to estimate what it would cost to rebuild **** it from scratch **** *. She snaps photos of* **** the original woodwork **** *and the plaster that has fallen away from the walls .*
GOLD: Once inside , she spends nearly four hours measuring and diagramming each room in the 80-year-old house , gathering enough information to estimate what it would

*cost to rebuild * **it** * from scratch . She snaps photos of * **the original woodwork** * and the plaster that has fallen away from the walls .*

Example A.11. *Unrelated spans*
*PREDICTED: Early this century , diamond mining in the magnificent dunes where the Namib Desert meets the Atlantic Ocean was a day at * **the beach** * . Men would crawl in * **on hands** * looking for shiny stones .*
*GOLD: Early this century , diamond mining in the magnificent dunes where the Namib Desert meets the Atlantic Ocean was a day at * **the beach** * . Men would crawl in * **the sand** * looking for shiny stones .*

A.4 BERT score

The original BERT score compares whole sequences and is not designed to compare their parts. It is possible to extract gold spans and compare them with the predicted ones, but in this case the context, i.e. the surrounding text, will be lost. And if we keep the text, then in most of the cases two sequences will be almost identical and this would lead to BERT score $> 90\%$ no matter what the model predicts. To avoid this problem, we modify BERT score as follows. First, we calculate the contextual embeddings of gold and predicted spans within the original text. Then, we provide span indices to the model and calculate the BERT score only between the embeddings of the spans, masking the embeddings of all the other tokens in the sequence.

A.5 Invalid generations and T-test statistics

Table 5 reports the proportion of invalid outputs generated by the LLMs. An output is considered invalid if it fails to follow the format specified in the prompt (Section 5), rendering it impossible to extract the recovered phrases.

The table also reveals whether differences in BERT scores achieved by different models for different types of spans are statistically significant. Insignificant differences (i.e. with *p-value* ≥ 0.05) are given in ***bold***. Given two types of spans, the negative statistic means that the score obtained for the first type is smaller than for the second one, e.g., we see that the BERT scores for the recovered antecedents are smaller than for anaphors across all the models. In most cases, these differences are significant. Next, we compare the scores for antecedents with the scores for both spans. As Table 5 shows, the former are larger than the latter, and the differences are also statistically significant. Given that the scores for antecedents are smaller than for anaphors, we conclude that the differences between the latter and the scores obtained for both spans are significant as well. This supports our hypothesis that for all models it is easier to restore a single bridging span rather than a pair.

Model	# invalid gen.	antec. vs anaphor		antec. vs both	
		statistic	p-value	statistic	p-value
Falcon-7B	172 (10.35%)	-4.25	2.52e-05	9.60	2.83e-20
Llama-8B	10 (0.60%)	-1.35	**0.18**	4.35	1.62e-05
Mistral-7B	185 (11.13%)	-3.89	1.10e-04	-0.11	**0.91**
Qwen-7B	39 (2.35%)	-1.58	**0.12**	5.92	5.66e-09
Yi-9B	45 (2.71%)	-4.33	1.76e-05	3.98	7.75e-05
Command-35B	31 (1.87%)	-0.80	**0.43**	3.83	1.40e-04
Falcon-40B	239 (14.38%)	-3.57	3.90e-04	7.39	5.34e-13
Qwen-32B	54 (3.25%)	-1.75	**0.08**	4.53	7.16e-06
Yi-34B	148 (8.90%)	-3.78	1.70e-04	2.56	0.011
Command-104B	42 (2.53%)	-3.06	2.00e-03	4.58	5.82e-06
Llama-70B	21 (1.26%)	-1.85	**0.065**	4.30	2.04e-05
Mistral-123B	98 (5.90%)	-3.68	2.60e-04	4.09	5.02e-05
Qwen-72B	45 (2.71%)	-3.46	5.90e-04	3.32	9.60e-04
T5-large	1 (0.06%)	n/a	n/a	n/a	n/a

Table 5: Number of invalid spans (out of 1,662) generated by LLMs and statistical significance of differences in BERT scores (F1) for different types of spans.

A.6 Models' versions

To save space and memory we use quantized variants of the models [1] from Hugging Face (Wolf et al., 2020).

- `TechxGenus/c4ai-command-r-v01-GPTQ` (35B)
- `alpindale/c4ai-command-r-plus-GPTQ` (104B)
- `tiiuae/falcon-7b-instruct`
- `tiiuae/falcon-40b-instruct`
- `TechxGenus/Meta-Llama-3-8B-Instruct-GPTQ`
- `TechxGenus/Meta-Llama-3-70B-Instruct-GPTQ`
- `TechxGenus/Mistral-7B-Instruct-v0.3-GPTQ`
- `TechxGenus/Mistral-Large-Instruct-2411-GPTQ` (123B)
- `Qwen/Qwen2.5-7B-Instruct-GPTQ-Int8`
- `Qwen/Qwen2.5-32B-Instruct-GPTQ-Int8`

[1] We did not find working quantized *Falcon* models, therefore we use their standard versions.

- `Qwen/Qwen2.5-72B-Instruct-GPTQ-Int8`
- `LnL-AI/Yi-1.5-9B-Chat-4bit-gptq`
- `zgce/Yi-1.5-34B-Chat-GPTQ-Int8`

A.7 Perplexities

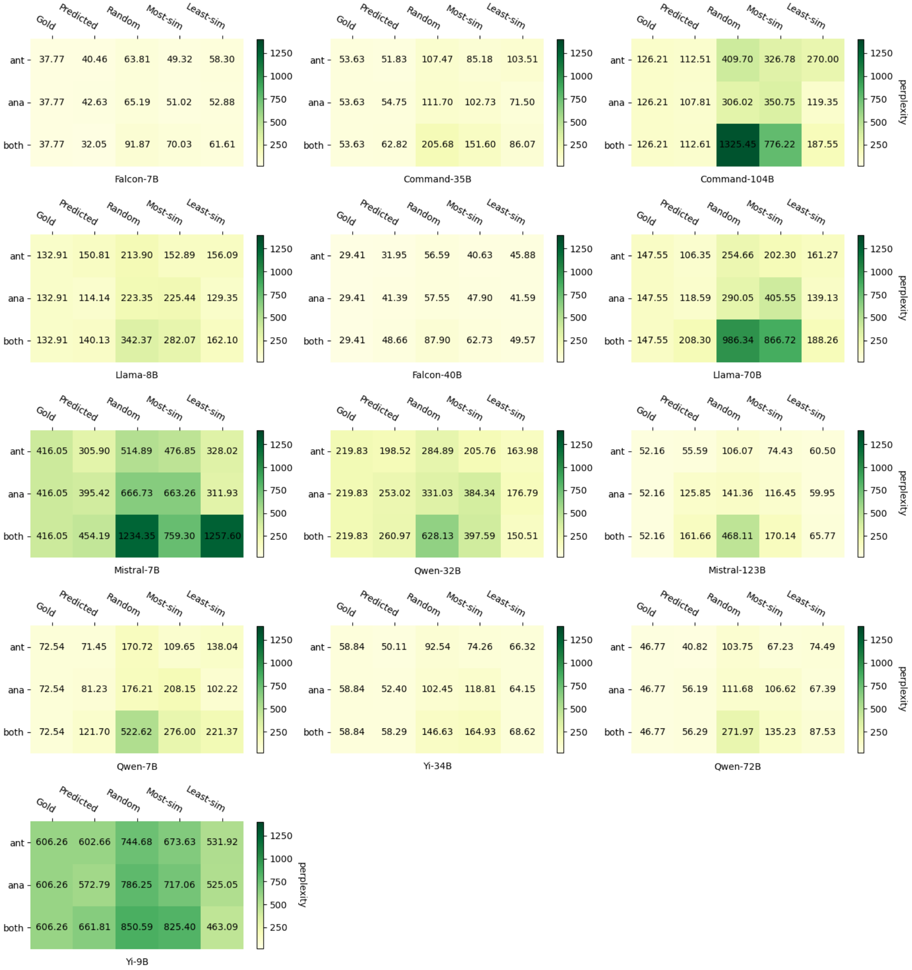

Figure 2: LLMs' perplexities for different types of spans

Correspondence

Natalia Skachkova

Deutsches Forschungszentrum für Künstliche Intelligenz (DFKI)
Saarbrücken, Germany
natalia.skachkova@dfki.de