NeuroComparatives: Neuro-Symbolic Distillation of Comparative Knowledge

Anonymous ACL submission

Abstract

Comparative knowledge (e.g., steel is stronger and heavier than styrofoam) is an essential component of our world knowledge, yet understudied in prior literature. In this paper, we harvest the dramatic improvements in knowledge capabilities of language models into a large-scale comparative knowledge base. While the ease of acquisition of such comparative knowledge is much higher from extreme-scale models like GPT-4, compared to their considerably smaller and weaker counterparts such as GPT-2, not even the most powerful models are exempt from making errors. We thus ask: to what extent are models at different scales able to generate valid and diverse comparative knowledge?

We introduce NeuroComparatives, a novel framework for comparative knowledge distillation overgenerated from language models such as GPT-variants and LLaMA, followed by stringent filtering of the generated knowledge. Our framework acquires comparative knowledge between everyday objects, producing a corpus of up to 8.8M comparisons over 1.74M entity pairs—10X larger and 30% more diverse than existing resources. Moreover, human evaluations show that NeuroComparatives outperform existing resources (up to 32% absolute improvement). Our acquired NeuroComparatives leads to performance improvements on three downstream tasks. We find that neuro-symbolic manipulation of smaller models offers complementary benefits to the currently dominant practice of prompting extreme-scale language models for knowledge distillation.

1 Introduction

In their book *"Surfaces and Essences"* on *concepts* and *analogies*, Hofstadter and Sander (2013) elucidate how concept learning requires *comparing* a pair of concepts, and parsing out their similarities and dissimilarities. Indeed, comparative knowledge is an essential component of our world knowledge (Ilievski et al., 2021; Davis, 2023), underpinning some of the classical commonsense reasoning problems. For example, the problem *"The large ball crashed right through the table because it was made of [steel/styrofoam]. What was made of [steel/styrofoam]?"* in Winograd Schema Challenge (Levesque et al., 2011) requires comparing the relative strength between steel and styrofoam. Yet, compared to *general* knowledge acquisition, there has been relatively little research focus on *comparative* knowledge acquisition, possibly due to the longstanding challenges of high-quality knowledge acquisition itself, let alone comparative knowledge. The few resources for comparative knowledge, all derived from web mining, however limited in size and diversity (§5), have nonetheless been useful for challenging multimodal reasoning tasks (Wang et al., 2018), highlighting the value of comparative knowledge.

In this paper, we draw inspirations from such literature about concept learning and inquire two related questions on *comparative knowledge*: (1) how well do models at different scales fare at the task of producing large-scale, high-quality comparative knowledge about a broad range of concepts? and (2) what are the implications for downstream tasks? Compared to prior resources of commonsense knowledge acquired either via crowdsourcing (Speer et al., 2017; Sap et al., 2019) or via information extraction (e.g., WebChild (Tandon et al., 2017) and ASER (Zhang et al., 2020)), our attempt to (re-)focus on the task of comparative knowledge acquisition takes the perspective of *"language models as knowledge bases"* (West et al., 2021; AlKhamissi et al., 2022), motivated by the dramatic improvements in the capabilities of extreme-scale neural language models.

We build on such research, but in addition to few-shot inference with extreme-scale model APIs such as GPT-4, also rely on customized inference with smaller language models (See et al., 2019; Sheng

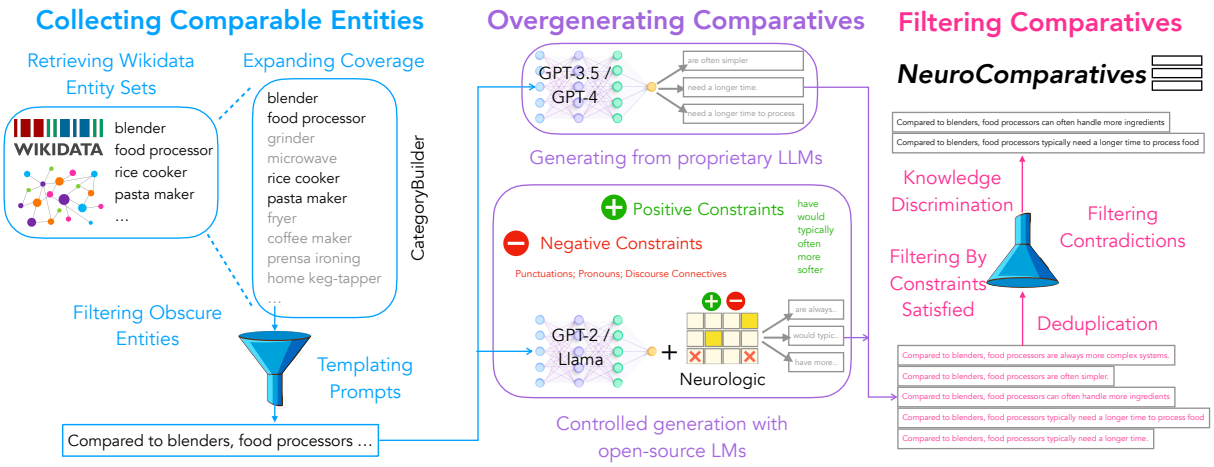

Figure 1: Our neuro-symbolic framework to distill **NeuroComparatives**. (1) We seed entity pairs for comparison from Wikidata, and expand the set with CategoryBuilder to construct templated prompts for a language model. (2) Next, we use these prompts to overgenerate comparatives from different language models to ensure our generations contain valid comparisons between a given pair of entities. (3) Finally, we discard contradictory and otherwise lower quality generations via various clustering and filtering techniques. Our resultant corpus NeuroComparatives contains 8.8 million comparisons over 1.74 million entity pairs.

et al., 2020; Liu et al., 2021). We ask a seemingly implausible question: can considerably smaller and weaker language models such as GPT-2 (Radford et al., 2019), complement the capabilities of their large-scale counterparts in the acquisition of *comparative knowledge* between a pair of concepts? To this end, we follow an overgenerate-and-filter mechanism (Langkilde and Knight, 1998; Walker et al., 2001) to create a large-scale, high-quality resource: **NeuroComparatives**, a corpus with up to 8.8 million comparisons over 1.74 million pairs of entities. Our framework is illustrated in Figure 1.

Compared to the only other large-scale commonsense KG containing comparative knowledge (Tandon et al., 2017, WebChild), NeuroComparatives is up to 10x larger, 30% more diverse, and has a 19% higher human acceptance rate. Additionally, we show that a knowledge discriminator model can further improve the the human acceptance rate of our knowledge to 90%, representing a 32% absolute gain compared to WebChild while still being over 2X larger in scale. Our analyses also show that NeuroComparatives are, on aggregate, more diverse than WebChild comparatives and more effective on three different downstream benchmarks. Overall, our findings motivate customizable neuro-symbolic manipulation of smaller scale models as a cost-effective complement to the dominant practice of performing simple inferences under extreme-scale yet closed language models. We will release our code and data.[1]

2 Distilling NeuroComparatives

Our framework for distilling comparatives from an autoregressive LM comprises three stages, illustrated in Figure 1. First, we collect comparable entities to construct prompts for eliciting comparative knowledge statements (§2.1). Next, we employ LMs to overgenerate (potential) comparatives for every pair of selected entities (§2.2). Finally, we filter the generations (§2.3) to obtain a large-scale, high-quality collection of comparative statements, which we call **NeuroComparatives (NCs)** (§3).

2.1 Collecting Comparable Entities

One unique challenge in probing language models for knowledge acquisition, as opposed to extracting pre-existing knowledge from web text, is knowing *exactly what* to probe LMs for, i.e., concept pairs for comparison. For practical applications, the comparatives are more likely to be useful when they are about entities sharing some common properties, e.g., "red wine" and "white wine" (Fig. 4), rather than unrelated ones e.g., "cucumber" and "car". Identifying a vast array of diverse yet relevant concepts for comparison is complex; thus, we developed a systematic process below: retrieval (§2.1.1), expansion (§2.1.2), and filtering (§2.1.3).

[1] http://anonymous/

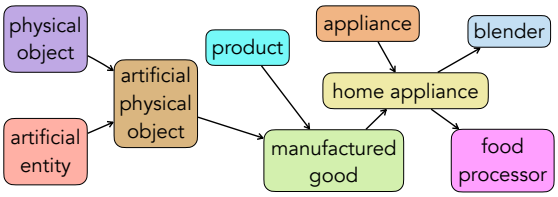

Figure 2: Wikidata hierarchical class structure for retrieved entities 'blender' and 'food processor'.

2.1.1 Retrieving Seed Entity Sets

We start our entity collection using two broad Wikidata (Vrandečić and Krötzsch, 2014) classes, as our seed classes: *physical object* and *artificial physical object*. Each seed class contains entities and subclasses, which themselves may contain additional entities. Figure 2 illustrates an example Wikidata class structure for "blender" and "food processor", with "physical object" as the root. Using a breadth-first traversal of Wikidata, we retrieve all classes up to two levels[2] below the root class. Overall, we retrieved 1.5K classes with 23K entities from Wikidata. While Wikidata provides a good starting point, we find that many of its classes are incomplete, a common challenge with any taxonomic resource. Thus, we next expand our entity sets to increase the coverage.

2.1.2 Expanding the Coverage of Entity Sets

We expand our entity collection using Category-Builder (Mahabal et al., 2018), a system for lexical entity set expansion. We append each retrieved entity set from Wikidata with the top $n = 100$ related terms identified by Category Builder using the hyperparameter $\rho = 3.0$ for context weighting. This results in a total of 40K entities corresponding to 1.5K Wikidata classes. Noting the presence of obscure entities, e.g., "home keg tapper" and "prensa ironing" in the "home appliance" class (Fig. 1), we then moved to eliminate these obscure entities.

2.1.3 Filtering Obscure Candidate Entities

Obscure entities would occur infrequently, thus we discard entities which occured less than $n = 100$ times in the language model's training corpus.[3] We additionally discard all classes with less than 2 entities after this filtering step. These filtering steps are applied twice: first on the original Wikidata entities, and then again after we expand the entity sets with Category Builder. This results in 568 classes with a total of 15,476 entities.

2.1.4 Templating Comparative Prompts

We generate comparison candidates by pairing entities within each class. For each such pair, $(\texttt{entity}_1, \texttt{entity}_2)$, we use the following template[4] to form the prompt for generation:

$$\text{Compared to } \texttt{entity}_1, \texttt{entity}_2 \ldots \quad (1)$$

As a final step, we further filter 30% of the created prompts using GPT-2 XL perplexity to remove potentially disfluent or nonsensical prompts. This results in a total of 1,741,962 prompts.

2.2 Overgenerating Comparatives

Since there's no supervision available, we follow an inference-only process for generating the initial set of comparative statements. We employ two approaches: constrained decoding with open-source LMs (§2.2.1) and few-shot prompting of proprietary LLMs (§2.2.2). We hypothesize that both approaches may offer complementary benefits.

2.2.1 Generation with open-source LMs

We experiment with GPT-2 XL and Llama-2 7B, two open-source models where inference can be customized to generate comparatives. We use a customized controlled decoding algorithm, NeuroLogic (Lu et al., 2021) to guide the generation using the prompts constructed above (§2.1).

Formulating the Constraint Sets We classify our constraints into three types: *positive*, *negative*, and *comparative adjectives*. Positive constraints ensure tokens appear in the output; we include auxiliary verbs (e.g., 'have', 'are', 'would') and adverbs of frequency (e.g. 'typically', 'often') (Appendix A for details). Negative constraints ensure tokens do not appear in the output to reduce hallucinations; we include certain punctuation characters, pronouns, discourse connectives, and relative clauses(Table 9 in Appendix A for details).

Constrained decoding with NeuroLogic We adapted NeuroLogic constrained decoding for fluent text generation under specific lexical constraints (NeuroLogic accepts *clauses* $\{C_i \mid i \in 1, \cdots m\}$ as constraints; see details in App. A.1).

To encourage diversity in the generated comparatives, the generator must use different comparative

[2]We use a maximum search depth of two based on the observation that descending lower in Wikidata results in entities that are too specific or obscure for generating comparatives.

[3]The only corpus we have access to is the open-source counterpart of GPT-2's training corpus, OpenWebText.

[4]We experimented with other templates but found that this one was most consistent at generating valid comparisons

Prompt	WebChild Assertions	Completions in NeuroComparatives (Ours)
Compared to helicopters, planes . . .	. . . were cooler ✓✓✗	. . . are more stable in flight ✓✓✓
	. . . are noisier ✓✓✗	. . . typically have higher operating costs ✓✓✗
	. . . are better ✓✓✗	. . . can often carry more cargo ✓✓✓
Compared to floppy disks, hard drives . . .	. . . are better ✓✓✓	. . . are generally considered more reliable ✓✓✓
Compared to cars, motorcycles . . .	. . . are cheaper ✓✗✓	. . . generally have fewer moving parts ✓✓✓
	. . . are smaller ✓✓✗	. . . generally have lower fuel consumption ✓✓✓
	. . . are cooler ✗✗✗	. . . tend to have shorter range ✓✓✓
Compared to blenders, food processors . . .	. . . are larger ✓✓✓	. . . can often be more expensive ✓✓✓
	. . . work better ✓✓✓	. . . can often handle more ingredients ✓✓✓

Table 1: Generations from NeuroComparatives and WebChild assertions for the same entity pair. Each example was annotated by three human workers: ✓indicates acceptance and ✗ rejection. In contrast to WebChild assertions, NeuroComparatives can be more specific to the entity pairs under consideration, diverse and less subjective.

adjectives, without explicit enumeration. Hence, we dynamically promote top-k comparative adjectives with the highest probabilities at each decoding time step (we use k=5). We customize this as a special type of positive constraint (§2.1.4).

We additionally modify NeuroLogic to handle ordered constraint satisfaction, for fine-grained control. For each clause C_i, we assign one or more order indices $m_i \in \{1, ..., m\}$ which correspond to the positional order in which clause C_i can appear in the generation. Specifying more than one order index allows a clause to appear in multiple different positions. Ordered constraint satisfaction provides more fine-grained control for generating valid comparatives, as illustrated in Figure 4.

2.2.2 Generation with Propriety LLMs

Many recent LLMs are proprietary, disabling our ability to customize the decoding process due inability to modify the decoding process. Hence, we leverage their in-context learning abilities to overgenerate completions for each entity pair. Specifically, we use six in-context examples of comparatives followed by our templated comparative prompt (details in App. C).

2.2.3 Scale of Overgeneration

Given its open-source availability and relatively low computational cost, we overgenerate the most comparatives using GPT-2-XL as our base LM. In total, we perform 30 passes of NeuroLogic with GPT-2 over the 1.74 million entity pairs from §2.1.4, where each iteration uses a different combination of the positive constraints, while adhering to the same negative and comparative adjective constraints. Each pass produces 10 generations, resulting in 300 candidate comparatives for each entity pair. This process produces a total of 522 million overgenerations across the 1.74 million entity pairs. We similarly use Llama-2-7b with NeuroLogic to generate comparatives for 50K entity pairs, which produces 15 million overgenerations.

In addition, we utilize three propriety LLMs (InstructGPT, ChatGPT, and GPT-4) to overgenerate comparatives for a smaller set of 2.3k entity pairs due to the much higher inference cost. For each entity pair, we return 128 completions, which is the maximum allowed via the inference API. This produces a total of 300k overgenerated comparatives.

2.3 Filtering Overgenerated Comparatives

Despite using a combination of effective language generation, we achieve quality control through aggressive filtering of the overgenerated comparatives. This last filtering step consists of deduplication (§2.3.1), filtration by constraint satisfaction (§2.3.2), filtration of contradictory knowledge (§2.3.3), and additional filtering via a knowledge discriminator model (§2.3.4).

2.3.1 Deduplication

To address LM's tendency to generate redundant comparisons, we deduplicate our generations. We use agglomerative clustering of all generated comparatives using the inner product of their sentence T5 embeddings (Ni et al., 2021) as the distance. For each cluster, we retain only the generation with the best decoding score, S[5] (App. A.1). Approximately 17% of the original generations remain.

[5]For proprietary LLMs, we use length-penalized perplexity from GPT-2 XL in lieu of the decoding score, S.

2.3.2 Filtration by Constraint Satisfaction

After deduplication, we group the remaining generations by how they satisfied the positive constraints to encourage greater diversity in our knowledge base. Specifically, we group generations by the generated auxiliary verb, adverb of frequency, and comparative adjective and select only the generation with the best S^5. This further reduces the total number of generations to approximately 9% of the overgenerated comparatives.

2.3.3 Filtration by Contradiction

The tendency of language models to hallucinate information (Ji et al., 2022) sometimes results in unreliable generations which contradict each other. Using a RoBERTa contradiction classifier (Liu et al., 2019; Wang et al., 2022), we discard comparatives that contradict others more often than not. To increase the precision of the pre-trained classifier, we set a high threshold probability for classifying contradiction and entailment (0.99 and 0.85, resp.). Approximately 5% of the overgenerated comparatives remain after this stage of filtering, from which we select only the $k = 5$ best-scoring generations by their S^5 for each entity pair.

2.3.4 Discriminative Filtering

To further increase the quality of our retained knowledge, we build a final discriminator, adjustable to the desired balance between knowledge quantity and quality. We train a knowledge discriminator using crowdsourced annotations of valid and invalid knowledge generations, following prior work (West et al., 2021). For a random 10K sample of our generated comparatives for unique entity pairs, we crowdsource the validity of each comparison (using the same setup described subsequently in §3.1). We train a classifier to discriminate between aggregated "Accept" and "Reject" crowdsourced labels (see Appendix B for additional details) and vary the threshold for the "Accept" class to filter at different levels of knowledge precision.

3 NeuroComparatives

Our large-scale generation effort produced 8.8m comparatives before discriminative filtering (details in App. A), which we refer to as NC-XL. Specifically, NC-XL includes 8.7M comparatives generated from GPT-2 XL, 78K from Llama-2-7b, 16.3K from InstructGPT, 10.7K from ChatGPT, and 6.6K from GPT-4. We further apply our knowledge discriminator model on NC-XL to create subsets containing only the top-50% and top-20% of comparatives, which we refer to as NC-L and NC-S, respectively. While NC-XL contains the greatest breadth of comparative knowledge, the more stringent filtering applied to produce NC-L and NC-S results in the highest knowledge quality, without hurting diversity.

The largest existing resource of comparative commonsense knowledge is WebChild (Tandon et al., 2017), collected via information extraction methods. While WebChild contains over 18M general assertions covering 2M concepts and activities, we focus on its comparative knowledge, which spans 813K assertions over 335K entity pairs. Compared to WebChild, our NC-XL corpus is 10x larger. Table 1 provides examples of NC-XL in contrast to WebChild assertions across four pairs of entities. For ease of comparison, we convert the WebChild assertions from OpenIE triplets to a natural language format, similar to ours.

The first set of examples for the entity pair (helicopters, planes) illustrates the more detailed, domain-specific properties, such as "operating costs", "more cargo", and "stable in flight" in NC-XL. In contrast, WebChild assertions are more generic (e.g., "cooler", "better") and not specific to the domain of flight. This example also highlights how NC-XL assertions are more informative and interesting to humans, as evidenced by their lower rate of rejection shown in Table 1.

We also compare our generated knowledge to ATOMIC (Sap et al., 2019) and ConceptNet (Speer et al., 2017). Although neither explicitly contains comparative knowledge, they do contain relations from which comparatives can be inferred. We use the `AtLocation` and `MadeUpOf` relations in ATOMIC, as well as the `AtLocation`, `PartOf`, and `MadeOf` relations in ConceptNet, to infer size comparisons over entities. We convert these entries to our NCs format for evaluation; for e.g., the ATOMIC triple (`human body`, `MadeUpOf`, `brain`) results in the comparative: "Compared to brains, human bodies are larger.". We use human (§3.1) and automatic (§3.2) evaluation to compare the quality and diversity of NeuroComparatives with existing comparative knowledge resources.

3.1 Human Evaluation of Validity

We task 3 workers from Amazon Mechanical Turk with classifying each comparative into one of six categories: 'True', 'False', 'Too subjective to judge', 'Too vague to judge', 'Too unfamiliar to

judge' and 'Invalid'.[6] We discard examples where there was no majority consensus among the 3 workers, and those marked as 'Too unfamiliar to judge' by a majority vote. Examples marked as 'True' are considered valid, and all others, invalid. Appendix D.2 details our annotation process (Fig. 6).

	Source	Size	Accept. ↑	SB2 ↓	SB3 ↓
Existing KBs	✍ ConceptNet	34,355	91.8%	1.00	1.00
	✍ ATOMIC	23,566	89.6%	1.00	1.00
	WebChild	812,862	58.1%	0.77	0.71
Few-shot LLMs	InstructGPT	-	72.7%	0.91	0.89
	ChatGPT	-	86.2%	0.90	0.88
	GPT-4	-	89.4%	0.87	0.84
NCs (GPT-2)	NC-XL	8,709,810	76.9%	0.64	0.58
	NC-L	4,354,905	84.4%	0.64	0.58
	NC-S	1,741,962	90.1%	0.65	0.59
NCs (Other LLMs)	NCs (LLaMA-7b)	77,798	90.0%	0.64	0.59
	NCs (InstructGPT)	16,300	83.8%	0.60	0.56
	NCs (ChatGPT)	10,756	88.6%	0.48	0.44
	NCs (GPT-4)	6,630	89.0%	0.52	0.48

Table 2: Size, human acceptance rates (on a 0.5k subset) and diversity measures, Self-BLEU-2 (SB2) and Self-BLEU-3 (SB3) for different comparatives. ✍ indicates human-authored comparatives. NCs generated from GPT-2 achieve a better trade-off between acceptance and diversity than generations from few-shot LLMs (no filtering); our filtered NCs from LLMs fare even better. Discriminative filtering of NeuroComparatives improves acceptance without hurting diversity.

We evaluate 500 randomly sampled comparatives from NC-XL, WebChild, ConceptNet, and ATOMIC. For evaluation of comparative knowledge extracted from GPT-3, ChatGPT, and GPT-4 without our framework (i.e., without any filtering), we obtain a sample of 500 completions to the same prompts used to generate the sampled NeuroComparatives (see Appendix C for additional discussion on the impact of filtering on ChatGPT and GPT-4). Human acceptance results are shown in Table 2 along with the size (total num. of comparatives) of different sources of comparative knowledge. While human-authored comparatives in ConceptNet and ATOMIC have the highest acceptance, these sources are the smallest in size, involved expensive human efforts and cannot be arbitrarily scaled. Among generated comparatives, NC-XL achieves nearly a 20% absolute improvement in human acceptance relative to WebChild, while containing over 10x more comparative knowledge. On average, few-shot prompting without filtering achieves lower acceptance than our NCs with the same LLMs, highlighting the benefits of our approach. NCs generated by LLaMA-7b achieved a similar human acceptance rate as that generated by GPT-2, suggesting that high-quality knowledge may be acquired from smaller-scale LMs.

[6]This is an absolute evaluation scheme; relative comparisons of pairs of comparatives are somewhat unfair since the comparisons might be along different dimensions.

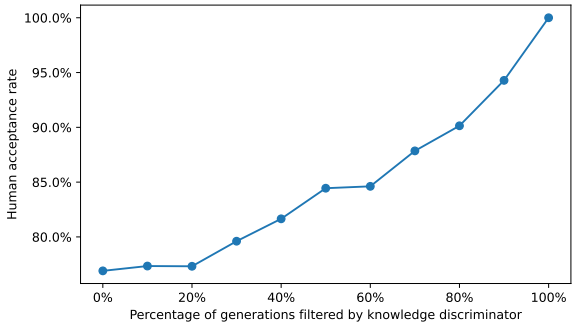

Figure 3: As our knowledge discriminator gets stricter, human acceptance of NeuroComparatives increases.

Filtering Contradictions improves NeuroComparatives We conduct an ablation to study the impact of filtering contradictions on generating NeuroComparatives (§2.3.3). We obtain a sample of comparatives from GPT-2-XL without the contradiction filter. The overall acceptance rate of these comparatives is 69.1%, which is an absolute decrease of 7.8% compared to NC-XL, confirming the importance of contradiction filtration.

Impact of Discriminative Filtering Figure 3 and Table 2 show that NC-L, which includes only the top-50% of our generated knowledge, achieves an acceptance rate of 84.4%, a 7.5% absolute increase relative to NC-XL. The top-20% filtering used to produce NC-S improves the acceptance rate even further to 90.1%, which exceeds the acceptance of few-shot prompting with both ChatGPT and GPT-4 (without our filtering) and is on par with human-authored sources. At this level of filtering, NC-S is still over 2x larger than WebChild while having an approximately 32% absolute gain in human acceptance rate.

3.2 NeuroComparatives' Diversity

We report Self-BLEU (Zhu et al., 2018), using both bigrams (SB2) and trigrams (SB3), to compare the diversity of comparatives from different sources in Table 2. We randomly sample 500 entity pairs from WebChild containing at least 5 comparisons. For few-shot prompting with GPT-3, ChatGPT, and

GPT-4, we use the same prompts (see §3.1) to obtain the top-5 generations for each entity pair. For each source, we compute Self-BLEU among the 5 candidates. Since the comparatives from ConceptNet and ATOMIC are limited to a single relation (size), measure of diversity doesn't apply. NC-XL and NC-L exhibit the greatest diversity, with a 31% reduction in Self-BLEU-3 relative to few-shot prompting with GPT-4. Crucially, knowledge discrimination does not impact the diversity of NC-S.

While NC-XL by design contain 5 comparisons for each pair of entities, the amount of comparative knowledge per entity pair in WebChild is heavily skewed: approximately 80% of the entity pairs have only 1 comparison and over 10k assertions comparing the entities "man" and "woman." WebChild is also heavily skewed towards a small number of frequently-occurring relations (e.g., "better"). We illustrate the skew in Figure 5 in Appendix D.1.

To further quantify the diversity of comparatives in NC-XL and WebChild, we consider identical relations from each source by considering unique comparative adjective phrases. We build a frequency-based probability distribution for relations in each source, and use it to compute their respective entropies— higher entropy indicates greater diversity of comparative relations. Our results show that the entropy of NC-XL is 7.9, which is 30% higher (indicating greater diversity) than the WebChild entropy of 6.1.

4 Downstream Task Performance

While NeuroComparatives are demonstratably diverse (§3.2), the impact on downstream task can be measured via their coverage on existing benchmarks. Here, we demonstrate NCs' wide knowledge coverage without compromising knowledge quality on three benchmarks.

To this end, we use two existing benchmarks: Elephant (Elazar et al., 2019) and Verb Physics Reannotated (Elazar et al., 2019, VPR). Elephant contains 486 comparisons of sizes of various transportation vehicles and animals. VPR reannotates VerbPhysics Forbes and Choi (2017) by filtering out the examples where the objects were not comparable, or the annotators disagreed among themselves. This re-annotation results in 2964 examples of object relations (1927 unique pairs of objects) along the dimensions of speed, length and mass.

The unstructured knowledge in NeuroComparatives facilitates easy integration with LLMs through natural language, therefore enabling the use of comparative reasoning abilities of LLMs. As there are no benchmarks for systematically evaluating the comparative reasoning capabilities, we constructed a new comparative reasoning QA benchmark, **ComparativeQA** to investigate this question. Specifically, we identified all comparatives that were unanimously assigned a label of 'True' by annotators in our human evaluation of validity experiments (§3.1). We then rephrased these comparatives as questions in order to construct a QA benchmark (see App. E.2). This produced a total of 656 comparative reasoning questions.

	Elephant		VPR	
NCs Source	**#Overlap**	**Acc. ↑**	**#Overlap**	**Acc. ↑**
NC-S (GPT-2 XL)	205	66.7%	869	66.7%
NCs (LLaMA-2-7b)	132	90.2%	760	71.2%
NCs (InstructGPT)	250	76.8%	1106	76.2%
NCs (ChatGPT)	189	97.9%	1090	88.2%
NCs (GPT-4)	226	**98.7%**	1351	**93.3%**

Table 3: Coverage (overlap) and accuracy of NCs on 486 examples in the Elephant dataset as well as 2964 examples in the VPR dataset.

Coverage on Elephant We evaluated five different versions of NeuroComparatives that differed by model scale: NC-S generated from GPT-2 XL, NCs from LLaMA-2-7b, from GPT-3, from ChatGPT, and from GPT-4; all variants went through discriminative filtering. Because NeuroComparatives are generated without restrictions on the attribute used for comparison, only a subset of these comparisons are along the size dimension corresponding to the 486 comparisons in Elephant. We identify this subset via a simple string matching approach and filter out the remaining generations for this evaluation.

As shown in Table 3, GPT-4 NCs achieved nearly perfect exact match accuracy—98.7% express the correct size relationship according to the Elephant annotations, and higher overlap with the Elephant dataset. We observe a very clear trend that more capable models generate NCs with higher quality.

Coverage on VPR We used our framework to generate NCs with the unique object pairs in VPR and additionally filtered them to contain comparisons only along speed, length, and mass using few-shot classification with Falcon-40b-instruct. We then used a BART-large model trained on MNLI to determine if a pair of NCs and VPR entail each other. Entailment means that our generated knowl-

LLaMA-2-7b Finetuning Source	Acc. ↑
None (5-shot)	60.7%
NC-S (GPT-2)	**91.0%**
NC-S (GPT-2 + ChatGPT + GPT-4)	**92.7%**
Reversed-NC-S	14.6%
WebChild	79.0%

Table 4: LLaMA-2-7b performance on comparative reasoning QA. fine-tuned on 50k sampled comparatives from various sources. All results are 5-shot.

edge agrees with VPR used as ground truth (see Appendix E.1 for details).

In Table 3, we can see that GPT-4 NCs again achieved the best overall accuracy of 93.3% as well as the highest coverage, highlighting the value of large-scale models. On top of being accurate, our NCs generates much more diverse knowledge beyond these three evaluated attributes, not directly reflected in the results.

Results: ComparativeQA We used a LLaMA-2-7b model for QA on our constructed ComparativeQA benchmark. As shown in Table 4, vanilla LLaMA-2-7b achieves only 60.7% accuracy on this dataset, using a prompt with five in-context examples. After finetuning LLaMA-2-7b for 3 epochs on a separate dataset of 50k QA comparatives sampled from NC-S from GPT-2, its accuracy on the test dataset increases to 91%. This is despite the lack of overlap between the entity pairs in the 50k NCs sampled for training and the test set. On additionally including 17k NC-S from ChatGPT and GPT-4 for finetuning, we see a further increase in validation accuracy to 92.7%.

As a control experiment, we finetuned on directionally reversed comparatives on the sampled NC-S. Finetuning on this set of incorrect comparatives degrades the accuracy of LLaMA-2-7b on the test set to 14.6%, highlighting the quality of our NCs. In addition, we also finetuned LLaMA-2-7b using a random sample of 50k comparatives from WebChild, and this results in an accuracy of 79%.

5 Related Work and Discussion

Comparative Knowledge: Despite the significance of comparative knowledge (Hofstadter and Sander, 2013), resources for the same are few and far between. Those that do exist have been collected almost exclusively via IE and data mining methods (Jindal and Liu, 2006; Cao et al., 2010; Jain and Pantel, 2011; Jang et al., 2012; Tandon et al., 2014, 2017; Elazar et al., 2019, i.a.), raising questions about coverage and diversity. Our approach is designed to address such issues via distilling knowledge from models at different scales. Most modern knowledge resources such as ATOMIC (Sap et al., 2019), VerbPhysics (Forbes and Choi, 2017) and DoQ (Elazar et al., 2019) may contain implicit comparisons via relationships between and properties of entities. A few knowledge resources involve explicit comparisons, but only along a few specific dimensions like physical properties (e.g. size (Bagherinezhad et al., 2016) or number (Narisawa et al., 2013)). Recently, Yu et al. (2023) collect data from structured and unstructured sources on specific real-world entities. Shivade et al. (2015) compare gradable lexical items, primarily adjectives and adverbs. In contrast, our NCs involve *explicit* comparisons between *nominal everyday concepts*, *without restrictions on the comparison dimensions*.

LM Knowledge Distillation: The ascendance of LLMs has begun to replace expensive, manually constructed knowledge bases due to their coverage benefits (AlKhamissi et al., 2022). LLMs have been used to create resources of factual knowledge (Petroni et al., 2019), structured knowledge graphs (Hao et al., 2022), instructions for further training (Wang et al., 2023) and training data for different tasks (Liu et al., 2022; Chakrabarty et al., 2022, *i.a.*). Our overgenerate and filter framework is inspired by West et al. (2021) who distill GPT-3 into a commonsense KG with a supervised critic. Our NCs focuses on *comparative knowledge* distilled from (among others) GPT-2 and LLaMA-2 with neuro-symbolic constrained decoding; Allaway et al. (2022) and Bhagavatula et al. (2022) use a related method to distill generics knowledge (Hampton, 2012) from GPT-2.

We demonstrate distillation of high-quality comparative knowledge from LMs at different scales and produce NeuroComparatives: the largest comparative knowledge corpus to date. NC-XL is 10x larger, 30% more diverse, and 19% more human acceptable than existing sources; with knowledge discrimination, we additionally achieve over 90% human acceptance. Our work highlights the value of comparative knowledge and exploits both neuro-symbolic manipulation of small-scale models and extreme-scale models for knowledge distillation.

Limitations

NeuroComparatives is a collection of fully generated data with limited manual verification. Caution must be exercised around training and deploying models on such data, due to reasons outlined below.

While our work centers around distilling knowledge from language models, it is well known that language models generate misinformation as well as toxic content. The scale of generations in our paper makes it challenging to manually analyze each generation. We expect that our filtering stage (§2.3) and knowledge discriminator (§2.3.4) are able to filter out many contradictory statements, but the veracity of the filtered remainder is challenging to determine. It is conceivable that there remain some fallacies in the data. As our comparisons are designed to be restricted to be between physical objects (as our root seed entities), we avoid comparisons between animate entities and any toxic content that might be associated with such comparisons.

NeuroComparatives is a resource in English only. Further, we restricted our entities to be objects in the real world which are nouns. However, there could be many potentially useful comparisons among verbs and adjectives. Due to limited resources, we leave the investigation of those to future work.

References

Badr AlKhamissi, Millicent Li, Asli Celikyilmaz, Mona Diab, and Marjan Ghazvininejad. 2022. A review on language models as knowledge bases.

Emily Allaway, Jena D Hwang, Chandra Bhagavatula, Kathleen McKeown, Doug Downey, and Yejin Choi. 2022. Penguins don't fly: Reasoning about generics through instantiations and exceptions. *arXiv preprint arXiv:2205.11658*.

Hessam Bagherinezhad, Hannaneh Hajishirzi, Yejin Choi, and Ali Farhadi. 2016. Are elephants bigger than butterflies? reasoning about sizes of objects. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, page 3449–3456. AAAI Press.

Chandra Bhagavatula, Jena D. Hwang, Doug Downey, Ronan Le Bras, Ximing Lu, Keisuke Sakaguchi, Swabha Swayamdipta, Peter West, and Yejin Choi. 2022. I2d2: Inductive knowledge distillation with neurologic and self-imitation.

Yanan Cao, Cungen Cao, Liangjun Zang, Shi Wang, and Dongsheng Wang. 2010. Extracting comparative commonsense from the web. In *Intelligent Information Processing*.

Tuhin Chakrabarty, Arkadiy Saakyan, Debanjan Ghosh, and Smaranda Muresan. 2022. FLUTE: Figurative language understanding through textual explanations. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7139–7159, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Ernest Davis. 2023. Benchmarks for automated commonsense reasoning: A survey.

Yanai Elazar, Abhijit Mahabal, Deepak Ramachandran, Tania Bedrax-Weiss, and Dan Roth. 2019. How Large Are Lions? Inducing Distributions over Quantitative Attributes. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3973–3983.

Maxwell Forbes and Yejin Choi. 2017. Verb physics: Relative physical knowledge of actions and objects. In *Annual Meeting of the Association for Computational Linguistics*.

James A Hampton. 2012. Generics as reflecting conceptual knowledge. *Recherches linguistiques de Vincennes*, pages 9–24.

Shibo Hao, Bowen Tan, Kaiwen Tang, Hengzhe Zhang, Eric P Xing, and Zhiting Hu. 2022. Bertnet: Harvesting knowledge graphs from pretrained language models. *arXiv preprint arXiv:2206.14268*.

Douglas Hofstadter and Emmanuel Sander. 2013. *Surfaces and Essence: : Analogy as the Fuel and Fire of Thinking*. Basic Books.

Filip Ilievski, Alessandro Oltramari, Kaixin Ma, Bin Zhang, Deborah L McGuinness, and Pedro Szekely. 2021. Dimensions of commonsense knowledge. *Knowledge-Based Systems*, 229:107347.

Alpa Jain and Patrick Pantel. 2011. How do they compare? automatic identification of comparable entities on the web. *2011 IEEE International Conference on Information Reuse & Integration*, pages 228–233.

Myungha Jang, Jin-Woo Park, and Seung won Hwang. 2012. Predictive mining of comparable entities from the web. *Proceedings of the AAAI Conference on Artificial Intelligence*.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yejin Bang, Andrea Madotto, and Pascale Fung. 2022. Survey of hallucination in natural language generation. *ACM Computing Surveys*.

Nitin Jindal and B. Liu. 2006. Mining comparative sentences and relations. In *AAAI Conference on Artificial Intelligence*.

Irene Langkilde and Kevin Knight. 1998. Generation that exploits corpus-based statistical knowledge. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, pages 704–710, Montreal, Quebec, Canada. Association for Computational Linguistics.

Hector J Levesque, Ernest Davis, and Leora Morgenstern. 2011. The Winograd schema challenge. In *AAAI*, volume 46, page 47.

Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A. Smith, and Yejin Choi. 2021. DExperts: Decoding-time controlled text generation with experts and anti-experts. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6691–6706, Online. Association for Computational Linguistics.

Alisa Liu, Swabha Swayamdipta, Noah A. Smith, and Yejin Choi. 2022. WANLI: Worker and AI collaboration for natural language inference dataset creation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6826–6847, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692.

Ximing Lu, Peter West, Rowan Zellers, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Neurologic decoding:(un) supervised neural text generation with predicate logic constraints. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4288–4299.

Abhijit Mahabal, Dan Roth, and Sid Mittal. 2018. Robust handling of polysemy via sparse representations. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 265–275.

Katsuma Narisawa, Yotaro Watanabe, Junta Mizuno, Naoaki Okazaki, and Kentaro Inui. 2013. Is a 204 cm man tall or small ? acquisition of numerical common sense from the web. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 382–391, Sofia, Bulgaria. Association for Computational Linguistics.

Jianmo Ni, Gustavo Hern'andez 'Abrego, Noah Constant, Ji Ma, Keith B. Hall, Daniel Matthew Cer, and Yinfei Yang. 2021. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. In *Findings*.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? 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 2463–2473, Hong Kong, China. Association for Computational Linguistics.

A. Radford, Jeffrey Wu, R. Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A Smith, and Yejin Choi. 2019. Atomic: An atlas of machine commonsense for if-then reasoning. *Proceedings of the AAAI conference on artificial intelligence*, 33(01):3027–3035.

Abigail See, Stephen Roller, Douwe Kiela, and Jason Weston. 2019. What makes a good conversation? how controllable attributes affect human judgments. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1702–1723, Minneapolis, Minnesota. Association for Computational Linguistics.

Emily Sheng, Kai-Wei Chang, Prem Natarajan, and Nanyun Peng. 2020. Towards Controllable Biases in Language Generation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3239–3254, Online. Association for Computational Linguistics.

Chaitanya Shivade, Marie-Catherine de Marneffe, Eric Fosler-Lussier, and Albert M. Lai. 2015. Corpus-based discovery of semantic intensity scales. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–493, Denver, Colorado. Association for Computational Linguistics.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Thirty-first AAAI conference on artificial intelligence*.

Niket Tandon, Gerard de Melo, and Gerhard Weikum. 2014. Acquiring comparative commonsense knowledge from the web. In *AAAI Conference on Artificial Intelligence*.

Niket Tandon, Gerard De Melo, and Gerhard Weikum. 2017. Webchild 2.0: Fine-grained commonsense knowledge distillation. In *Proceedings of ACL 2017, System Demonstrations*, pages 115–120.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

Marilyn A. Walker, Owen Rambow, and Monica Rogati. 2001. SPoT: A trainable sentence planner. In *Second Meeting of the North American Chapter of the Association for Computational Linguistics*.

Peng Wang, Qi Wu, Chunhua Shen, Anthony Dick, and Anton van den Hengel. 2018. FVQA: Fact-Based Visual Question Answering. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(10):2413–2427.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. Self-instruct: Aligning language models with self-generated instructions. In *Proc. of ACL (to appear)*.

Peter West, Chandra Bhagavatula, Jack Hessel, Jena D Hwang, Liwei Jiang, Ronan Le Bras, Ximing Lu, Sean Welleck, and Yejin Choi. 2021. Symbolic knowledge distillation: from general language models to commonsense models. *arXiv preprint arXiv:2110.07178*.

Mengxia Yu, Zhihan Zhang, Wenhao Yu, and Meng Jiang. 2023. Pre-training language models for comparative reasoning.

Hongming Zhang, Xin Liu, Haojie Pan, Yangqiu Song, and Cane Wing-Ki Leung. 2020. Aser: A large-scale eventuality knowledge graph. In *Proceedings of the web conference 2020*, pages 201–211.

Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. 2018. Texygen: A benchmarking platform for text generation models.

A Additional Details on the Generation of NeuroComparatives

A.1 NeuroLogic Background

NeuroLogic accepts a series of constraints $D(\mathbf{a}, \mathbf{y})$ which are true iff '$\mathbf{a}$' appears in the generated sequence '$\mathbf{y}$'. where each constraint is a set of *clauses* $\{C_i \mid i \in 1, \cdots m\}$ consisting of one or more predicates in Conjunctive Normal Form (CNF):

$$\underbrace{(D_1 \vee D_2 \cdots \vee D_i)}_{C_1} \wedge \cdots \wedge \underbrace{(D_k \vee D_{k+1} \cdots \vee D_n)}_{C_m}. \quad (2)$$

Each constraint D might be positive or negative; $D(\mathbf{a}_i, \mathbf{y})$ is satisfied (i.e., evaluates as true) if $\mathbf{a}_i$ is present or absent, respectively, in $\mathbf{y}$

NeuroLogic employs a beam search approximation of an objective function which maximizes the probability of the generated sequence while penalizing deviations from m clauses:

$$\hat{\mathbf{y}}_t = \arg\max_{\mathbf{y} \in \mathcal{Y}} p_\theta(\mathbf{y}_t | \mathbf{y}_{<t}) - \lambda \sum_{j=1}^{m} (1 - C_j) \quad (3)$$

where $\lambda \gg 0$ penalizes deviations from the constraints.

Candidates are scored at each t per their (partial) satisfaction of the constraints:

$$f(\mathbf{y}_{\leq t}) = \log p_\theta(\mathbf{y}_{\leq t} | \mathbf{x}) + \lambda \max_{D(\mathbf{a}, \mathbf{y}_{\leq t})} \frac{|\hat{\mathbf{a}}|}{|\mathbf{a}|} \quad (4)$$

where $\hat{\mathbf{a}}$ represents a subsequence of $\mathbf{a}$ in the current generation. This has the effect of preferring candidates which at least partially satisfy multi-token constraints; for example, a generated sequence $\mathbf{y}_{\leq t}$ = "Compared to train tickets, airline tickets are generally more" would be rewarded for partially satisfying the constraint $\mathbf{a}$ = "more expensive" via its subsequence $\hat{\mathbf{a}}$ = "more".

Unlike the top-k selection strategy used in traditional beam search, NeuroLogic performs pruning, grouping, and selection steps to identify the best candidates which satisfy the given constraints. Specifically, candidates which irreversibly violate one or more constraints are pruned, and the remaining candidates are grouped according to their number of satisfied clauses in order to encourage diversity. The best candidate within each group is then selected according to the scoring function in Equation 4.

Each pass of NeuroLogic returns multiple generations, which are scored according to the sum of their length-penalized log probabilities:

$$\frac{1}{N^\alpha} \sum_{t=1}^{N} \log p_\theta(\mathbf{y}_t | \mathbf{y}_{<t})$$

where N denotes the length of the generated sequence $\mathbf{y}$ and α is a length penalty to encourage shorter generations (we use $\alpha = 0.1$). We refer to this score as as the S.

A.2 Constraint Sets for NeuroLogic

Table 5 provides the positive constraints used in NeuroLogic decoding. The table lists tokens used for two different positive constraint sets. For each of the 30 pairwise combinations of these auxiliary verbs and adverbs, we generate a completion of the prompt where the corresponding auxiliary verb and adverb is required to be present in the generation.

Auxiliary verbs	Adverbs of frequency
have	typically
need	often
may	always
are	generally
would	normally

Table 5: Positive constraint sets.

Prompt	Aux. Verb	Adverb
Compared to cherries, peaches ...	have	typically
Compared to cherries, peaches ...	have	often
Compared to cherries, peaches ...	have	always
⋮	⋮	⋮
Compared to cherries, peaches ...	would	normally

Table 6: Example of the prompt and 30 combinations of positive constraints for the entity pair (`cherries`, `peaches`).

An illustration of the prompt and the positive constraint combinations used to generate comparisons for an entity pair is provided in Table 6.

Table 5 provides the negative constraints used in NeuroLogic decoding.

We use GPT-2 XL as our language model, which has 1,542M parameters. For decoding with NeuroLogic, we use a beam size of 15, length penalty of 0.1, and an n-gram size of 3 for preventing repetitions. We use $\beta = 1.25$ as the reward factor for in-progress constraint satisfaction and set the constraint satisfaction tolerance to 3, which means that only candidates which have a number of satisfied constraints within 3 of the maximum are kept at each step. The hyperparameters are manually curated. Please refer to Lu et al. (2021) for details on these hyperparameters.

Our experiments were conducted on a cluster with Nvidia RTX A6000 GPUs. We distributed the generation across 64 GPUs, with each GPU running 4 decoding iterations in parallel. The total compute time to generate our knowledge base in this environment was approximately 5 weeks.

A.3 Ordered Generation

We additionally modify NeuroLogic to handle ordered constraint satisfaction, for fine-grained control. For each clause C_i, we assign one or more order indices $m_i \in \{1, ..., m\}$ which correspond to the positional order in which clause C_i can appear in the generation. Specifying more than one order index allows a clause to appear in multiple different positions. Ordered constraint satisfaction provides more fine-grained control for generating valid comparatives, as illustrated in Figure 4.

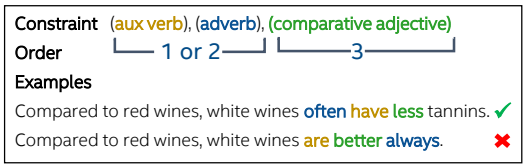

Figure 4: Examples of generated comparatives which satisfy and violate our constraint ordering.

B Details of knowledge discriminator model

We use 80% of the labeled data for training the knowledge discriminator and 20% for validation. We trained the knowledge discriminator on a Ubuntu 18.04 system with a single Nvidia RTX 3090 GPU. Specifically, we finetune RoBERTa-large previously trained on MNLI[7] using a learning rate of 5e-6, a batch size of 32, and a dropout probability of 0.1. Hyperparameters are manually curated. We train the model for a maximum of 50 epochs and monitor precision at recall $= 80\%$ on the validation set, terminating training if this metric fails to improve for 5 consecutive epochs. The total training time of the model was 13 minutes. Precision and recall on the validation set were 0.589 and 0.642, respectively.

C Details of experiments with InstructGPT, ChatGPT, and GPT-4

To compare our knowledge generations to InstructGPT, ChatGPT, and GPT-4, we use a prompt which instructs each model to complete a statement comparing two entities. The instruction is followed by five hand-crafted examples and the prefix that we want the model to complete in order to form a comparative knowledge statement. An example of the full prompt used to generate a comparative knowledge statement for the entity pair (computer keyboards, game controllers) is provided below.

Complete a statement which compares two entities.
Compared to blueberries, pineapples are heavier.
Compared to chairs, sofas are larger.
Compared to salad, pizza is less healthy.
Compared to a knife, a machete is more dangerious.
Compared to a bicycle, a skateboard is slower.
Compared to computer keyboards, game controllers

We use OpenAI's API with the text-davinci-001

[7] https://huggingface.co/roberta-large-mnli

model for InstructGPT, gpt-3.5-turbo for ChatGPT, and gpt-4 for GPT-4 results. We use the default parameter settings for each model and evaluate human acceptance using the first generation returned for each prompt. For diversity evaluations, we utilize the first 5 generations returned by the API for each prompt.

D More on Evaluation

D.1 More on diversity of NeuroComparatives

Figure 5 depicts the top-20 most frequent relations in each source, showing that the WebChild relations are more skewed, with its most-frequent relation ("better") representing over 12% of all relations. In contrast, the most frequent relation in NC-XL ("more expensive") represents only 4% of all relations. The most-frequent relations in NeuroComparatives are also generally more descriptive and less subjective than those in WebChild.

D.2 Crowdsourced evaluation details

Our crowdsourced evaluations utilized Amazon Mechanical Turk workers who were required to have completed at least 5,000 HITs, have a lifetime task acceptance rate $\geq 95\%$, and have achieved the 'Masters' qualification. A reward of $0.07 was paid to the workers for each submitted label.

To ensure that all sources of knowledge were evaluated in the same form, we transformed triples in WebChild into a comparative knowledge statement format. Specifically, we pluralized the head and tail entities of each triple using the `inflect` Python package and then formed a comparative knowledge statement using the following template: "Compared to {tail}, {head} {relation}".

We provided the following set of instructions and examples to the workers.

D.3 Instructions

In this task, you will be given a sentence which compares two entities.

- Determine whether the comparison is true or false (or indicate that you cannot determine its truthfulness) by selecting one of the 6 options.
- If the sentence is incoherent or not a valid comparison, select "Invalid". Please be forgiving of spelling or grammatical errors and avoid labeling it as invalid if the sentence only has minor grammatical mistakes.
- If the comparison is too vague or requires additional information to determine its truthfulness, select "Too vague to judge".
- If the comparison is overly subjective or expresses a personal opinion which is not commonly held by most people, select "Too subjective to judge".
- If the terms are too obscure or you do not know the truth of the comparison, it is okay to select "Too unfamiliar to judge". If you can answer (e.g., based on likelihood), please provide a response.
- If a comparison in unjudgeable due to more than one of the above reasons, select the option corresponding to the primary reason it cannot be judged.

D.4 Examples

True: "Compared to homes, office buildings are more expensive to build."

False: "Compared to doctorates, master's degrees are more difficult to obtain."

Invalid: "Compared to toothbrushes, utility knives may be less efficient at cleaning always on."
Explanation: It is unclear what being "less efficient at cleaning always on" means.

Too vague to judge: "Compared to text messages, video chats generally have higher levels."
Explanation: Higher levels of what? The comparison lacks details needed to determine its truthfulness.

Too subjective to judge: "Compared to french toast, pancakes are better."
Explanation: Although this comparison may be true for many people, it is a subjective opinion which varies substantially from person-to-person.

True: "Compared to frozen foods, fresh foods are healthier."
Explanation: While this comparison could also be considered an opinion, it is one which is widely held by most people and therefore should be labeled as True.

Too unfamiliar to judge: "Compared to gyroscopes, microelectromechanical systems may often

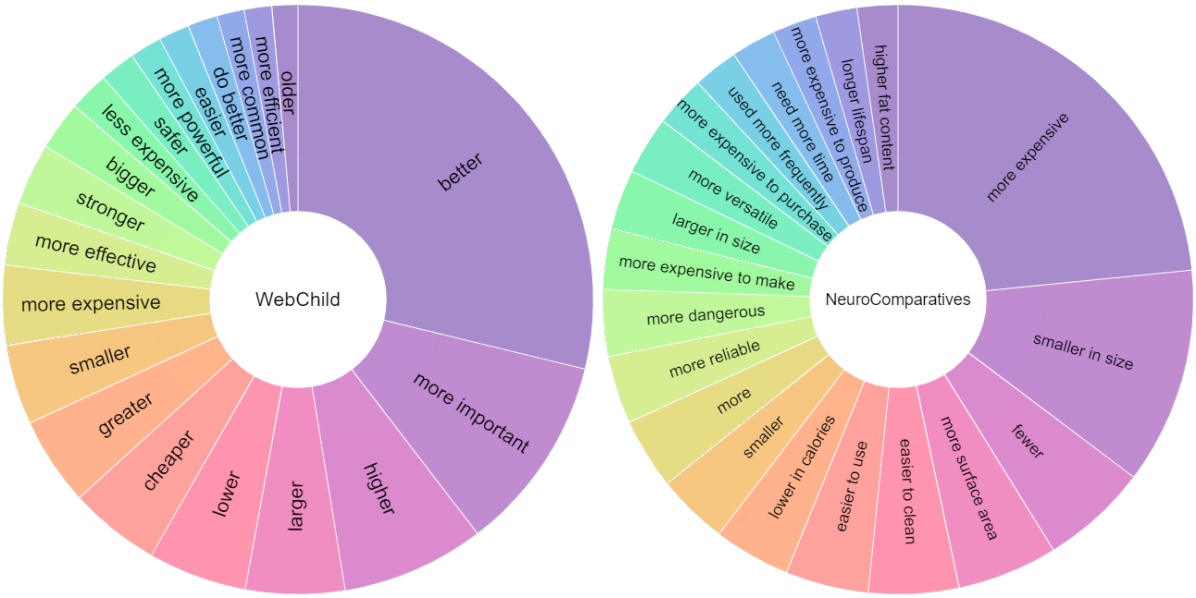

Figure 5: The distribution of the top-20 relations in WebChild is more skewed than NC-XL.

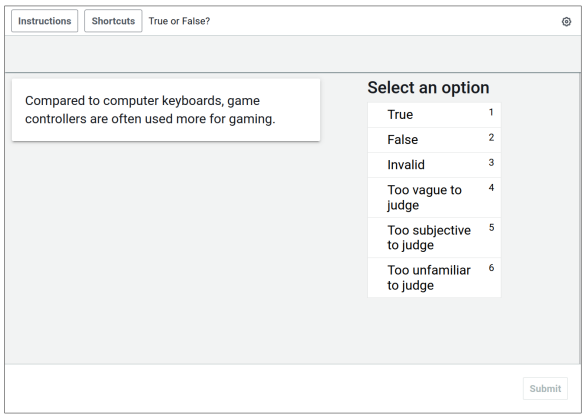

Figure 6: Validity labeling interface for crowdsourced workers

provide better performance."
Explanation: I am too unfamiliar with "gyroscopes" and "microelectromechanical systems" to judge this comparison.

E Case Study

E.1 Verb Physics

Filtering Here is the prompt used for filtering:

> *Solve a textual classification task by having a Thought, then Finish with your answer. Thought can reason about the current situation. Finish[answer] returns the answer and finishes the task. There are 4 classes you need to decide among speed, length, mass, and others. Don't answer with anything else. Here are some examples:*
> *{FEW-SHOT EXAMPLES}*
> *(END OF EXAMPLES)*
> *Sentence:*

Some examples are manually labeled and annotated and put in the place of {FEW-SHOT EXAMPLES} above(refer to Tab. 7 for a complete list of few-shot examples). We make sure those examples are not in the final evaluation set.

Classification We then used a BART-large model trained on MNLI[8] to determine if a pair of NeuroComparatives and VPR entail each other. We perform this entailment classification twice—first using NeuroComparatives as the premise and VPR examples as hypothesis, and second with the premise and hypothesis reversed. To better illustrate how the NLI is done, here is a quick example of the entailment classification: (ENT1, ENT2, length, 1) is one example in VPR where length means this is a comparison of length and 1 means ENT1 is longer than ENT2. Then we set the NeuroComparatives generated with this entity pair as the premise, and ['ENT1 is longer than ENT2', 'ENT1 is shorter than ENT2'] as the hypothesis. Then entailment classification is run with each hypothesis and get respective entailment probability. The hypothesis with higher entailment probability out of the two is chosen as the predicted entailment. If this predicted entailment complies with the original VPR example, we say our NeuroComparatives and the example in VPR entail each other, otherwise not. E.g., if 'ENT1 is longer than ENT2' has higher entailment probability, that means we

[8] https://huggingface.co/facebook/bart-large-mnli

our the example in the VPR is an entailment to our NeuroComparatives. This same process is done a second time but with premise being the example in VPR and hypothesis being our VPR (we just reverse the entities in our NeuroComparatives to get the counterfactual hypothesis). We count our NeuroComparatives as conformed / agreed by VPR if either of these two evaluations result in an entailment. Overall, we find that NeuroComparatives have a 66.7% entailment rate in this study, which is consistent with our previous analysis on the accuracy of matching NeuroComparatives to the Elephant dataset (§4).

E.2 Comparative Reasoning QA

Here is an example of the QA transformation from comparative data:

> ***Comparative:***
> *Compared to graham crackers, kiwifruits can be considered healthier",*
> ***QA Format:***
> *Question: Which of the following can be considered healthier?*
> *A. graham crackers*
> *B. kiwifruits*
> *Answer: B*

F Terms and Licenses

We use all of the datasets and models the way it is intended to be used if specified. For the ComparativeQA dataset we release, we intend its use for research only.

Few-shot Examples for Dimension Classification
Sentence: Compared to boats, trains are typically a lot slower. Thought: Let's think step by step. we can analyze the words and their context to determine the appropriate class. In this case, the sentence compares the speed of boats and trains, implying that the class we should assign is "speed." Action: Finish[speed] Sentence: Compared to cars, horses are often considered less practical. Thought: Let's think step by step. "Less practical" specifically refers to the practicality of horses in comparison to cars. We can conclude that the appropriate class for this sentence is "practicality," as it discusses the comparative practicality of horses compared to cars. However, practicality is not one of the 4 classes (speed, length, mass, and others) we are supposed to choose from. Therefore, we should assign the class "others" to this sentence. Action: Finish[others] Sentence: Compared to chins, heads often have more muscle mass. Thought: Let's think step by step. "More muscle mass" specifically refers to the muscle mass of heads in comparison to chins. Considering all these factors, we can conclude that the appropriate class for this sentence is "mass," as it discusses the comparative muscle mass of heads compared to chins. Action: Finish[mass] Sentence: Compared to fingers, arms are always longer. Thought: Let's think step by step. "Longer" specifically refers to the length of arms in comparison to fingers. Considering all these factors, we can conclude that the appropriate class for this sentence is "length," as it discusses the comparative length of arms compared to fingers. Action: Finish[length] Sentence: Compared to dresses, purses can typically hold more goods. Thought: Let's think step by step. "smaller" specifically refers to the size or dimensions of purses in comparison to dresses. Considering all these factors, we can conclude that the appropriate class for this sentence is "size" or "dimensions," as it discusses the comparative size of purses compared to dresses. Action: Finish[length]

Table 7: Few-shot examples used to classify NeuroComparatives into 4 dimensions: speed, length, mass and others.

Comparative Adjectives
littler, denser, sweeter, dumber, itchier, rawer, skinnier, righter, bloodier, harder
wider, creepier, cheaper, sorrier, sillier, hairier, odder, worthier, idler, cooler
higher, sourer, softener, unhappier, sadder, stingier, hotter, busier, slimmer, narrower
subtler, sharper, shorter, sparser, lesser, needier, drier, greasier, pricklier, neater
lighter, cuter, shyer, sweatier, floppier, shadier, fitter, lazier, crazier, muddier
purer, sooner, nearer, fresher, further, louder, chubbier, whiter, crueler, thirstier
slighter, flakier, clumsier, greener, rougher, fatter, prettier, calmer, damper, politer
fiercer, messier, darker, poorer, lovelier, lower, handier, steeper, deadlier, jointer
greedier, cleverer, steadier, headier, blunter, blander, outer, younger, dirtier, wiser
direr, graver, greater, riper, milder, noisier, likelier, meaner, sneakier, unlikelier
tougher, upper, angrier, stronger, shinier, stricter, smoother, fuzzier, tenther, sorer
classier, fairer, gentler, brighter, trickier, grainier, looser, harsher, extremer, grander
juicier, guiltier, colder, ruder, tighter, sunnier, newer, stickier, wealthier, crankier
quicker, dustier, trendier, cleaner, rosier, richer, braver, prouder, shaggier, earlier
larger, lengthier, windier, fonder, sleepier, heartier, bluer, filthier, worser, taller
worse, spicier, heavier, quirkier, stockier, scarier, creamier, roomier, smarter, curlier
clearer, goofier, hardier, breezier, grosser, laster, firmer, mushier, quieter, chewier
plainer, jumpier, lonelier, madder, touchier, readier, smokier, mightier, bitterer, sexier
unhealthier, snowier, wilder, norther, closer, later, saner, crispier, flatter, nastier
deeper, briefer, finer, smaller, cozier, hungrier, curvier, tastier, bigger, happier
smellier, faster, simpler, easter, tinier, kinder, fainter, thinner, blacker, bolder
funnier, holier, weightier, poppier, sturdier, nobler, livelier, hipper, duller, fuller
slower, cloudier, rustier, rarer, wetter, coarser, better, leaner, firer, crunchier
gloomier, speedier, abler, riskier, warmer, blanker, soggier, nicer, keener, moister
shallower, yellower, stranger, weirder, stiffer, stupider, lousier, humbler, friendlier
stealthier, straighter, softer, bossier, icier, fancier, broader, uglier, nexter, loftier, naughtier
scarcer, worldlier, tanner, luckier, sincerer, bulkier, oilier, easier, warier, healthier
earthier, wobblier, less, more, choppier, swifter, longer, saltier, truer, weaker
older, fussier, steepler, fewer, safer, slimier, fattier, chillier, thicker, nimbler

Table 8: Full list of comparative adjectives (290 words).

Punctuation & Nonsensical Characters (separated by tab)
;). – … -- .[— !! — .) .. ********** * ". í ($);]"
… % – %,]: $ >> + -, & !" $, .— -- .--][…. .'
]]]- ″ ⍰ ´ ` … .). !? ° '. …… …" ."" ……]. …. …" .]
.」 ., ⍰ (£ ½ %) %- %). " —" # ?' %), .⍰ (. ?! ê .-
-$](?). '." +, \'], ° .; - ½ ,' ~]," ?" ….. ', ?!" × \
~~ (< p _____]; _____ ____ Â ㅋㅋ ---------- _____ 」
【 ¨ !!! © c (" ___ り の か ん 。 ま し た な う い
っ で す ⍰⍰ !!!! a н ō o e _. | ・ м у д "- _____
Ã]) – ā ˜ ~ ī ???? _____ __ …… • · %; т
⍰⍰⍰ …… ……. _____ _____
是 子 不 à á ….. ㅋ ь и к 『 中 = ;) ,. ′ (>)" __
___ .") _ _ ; ?". 』 「 ** !!!!! ---------- ét ö ??? ';
_____ в :) ******** « …… …… …… …… ……. â
_ 一 、 .'" 】 大 "… !?" š 人 的 — .", ."[ ][/ 上 » л
+ _____ o⍰ ы я],[
][¢ 三 ~ 「 ña ū ' _(! = << £ | ?, ø ?」]. 光 •
]' "— :// ," — ら ~~~~ ,'" .").]). ° .(-. ----------
----------)) ú '- ((ür *** ?? 方 神 ' き て る あ と ⍰⍰⍰⍰
> ./ é ,[ís è)] %: %" :- → ée "," " ":{" / "}," ,"),
", - ." ? !) (

Pronouns
I / think / you / You / He / he / he. / They they / they. / she / she. / She / my / my. / We / we /

Discourse Connectives & Relative Clause
without / without. between / between. / much / much. / either / either. / neither / neither. / and / and. / when when. / while / while. / although / although. / am / am. / no / no. / nor / nor. not / not. / as / as. / because / because. / since / since. / although / although. / finally finally. / however / however. / therefore / therefore. / because / because. / consequently / consequently. / furthermore / furthermore. nonetheless / nonetheless. / moreover / moreover. / alternatively / alternatively. / henceforward / henceforward. / nevertheless / nevertheless. / whereas whereas. / meanwhile / meanwhile. / this / this. / there / there. / here / here. / same / same. few / few. / similar / similar. / the following / the following. / by now / by now. / into / into. / than / than. / and

Table 9: Full list of negative constraint sets separated by "/".