Language Models Are Poor Learners of Directional Inference

Tianyi Li◇ **Mohammad Javad Hosseini**◇* **Sabine Weber**◇ **Mark Steedman**◇

◇ School of Informatics, University of Edinburgh

{tianyi.li, javad.hosseini}@ed.ac.uk

s.weber@sms.ed.ac.uk, steedman@inf.ed.ac.uk

Abstract

We examine LMs' competence of directional predicate entailments by supervised fine-tuning with prompts. Our analysis shows that contrary to their apparent success on standard NLI, LMs show limited ability to learn such directional inference; moreover, existing datasets fail to test directionality, and/or are infested by artefacts that can be learnt as proxy for entailments, yielding over-optimistic results. In response, we present BoOQA (Boolean Open QA), a robust multi-lingual evaluation benchmark for directional predicate entailments, extrinsic to existing training sets. On BoOQA, we establish baselines and show evidence of existing LM-prompting models being incompetent directional entailment learners, in contrast to entailment graphs, however limited by sparsity.

1 Introduction

Pre-trained language models have shown impressive performance in natural language understanding (NLU), where prompting methods are widely used for fine-tuning (Raffel et al., 2020; Brown et al., 2020; Schick and Schütze, 2021).

In this paper, we specifically investigate predicate entailment detection, an important sub-task of NLU and specifically, NLI. The task is to predict, given that predicate **p** holds between arguments <**a**, **b**>, whether it can be inferred that predicate **q** also holds between <**a**, **b**>. For instance, "John shopped in IKEA" entails "John went to IKEA", but not "John drove to IKEA".

The primary distinction between predicate entailments and semantic similarity, apart from being focused on predicates, is that the former involves **directional** entailments as well as **symmetric** ones. Directional entailments are those $p \vDash q$ (p entails q) where $q \nvDash p$; conversely, symmetric entailments are those $p \vDash q$ where $q \vDash p$ as well (namely $p \equiv q$).

Directional entailments are important for question answering, since they help filter out the spurious connections from knowledge sources to questions: knowing that John went to IKEA, it is unsafe to infer that he shopped in IKEA, as he may have been there for other reasons. By symmetric similarity (i.e. paraphrase), the two events would be considered related, so a spurious inference chain would emerge; by directional entailments, it would be concluded that while the two events are related, the entailment holds only in the reverse direction, so the spurious connection would be avoided.

Current LM-prompting methods have reported positive results on predicate entailment detection (Schmitt and Schütze, 2021b,a). Since the masked-language-modelling objective naturally enables LMs to separate related and unrelated tokens, they are expected to be good paraphrase detectors; on the other hand, it is less clear whether they also distinguish the directionality of entailments.

To answer this question, we adapt the SOTA LM-prompting model (Schmitt and Schütze, 2021b) as a gauge for the competence of its underlying LMs, in particular RoBERTa (Liu et al., 2019) and BERT (Devlin et al., 2019). We apply it to various subsets of the common benchmark LevyHolt (Levy and Dagan, 2016; Holt, 2019). We find that while it scores highly on the directional subset by Holt (2019), it otherwise shows poor ability in learning the directionality of predicate entailments. We find instead that the LevyHolt directional subset is infested with artefacts, to which the supervised LM-prompting model is overfitting.

These observations show that we need a robust evaluation benchmark for directional predicate entailments, independent of training sets. Inspired by McKenna et al. (2021) and Chen et al. (2017), we present BoOQA, a Boolean Open QA dataset in English and Chinese, which is closer to applications, adversarial to artefacts in supervision, and demands sensitivity to the directionality of entailments.

---

* Now at Google Research.

903

On BoOQA, we re-examine supervised and unsupervised LM methods along with various discrete entailment graphs (EG) (Hosseini et al., 2018, 2021; Chen et al., 2022; Li et al., 2022). We find that the performances of supervised LM-prompting methods are indifferent to directional supervision, and are generally less competitive than suggested on LevyHolt; on the other hand, EGs reach decent precisions with their strongest edges, but are hit by sparsity and noisy unsupervised signals.

Our contributions can be summarized as follows: 1) We show that LevyHolt, the common directional predicate entailments benchmark, is infested by artefacts, allowing supervised methods to perform well by overfitting; 2) We prove that LMs, with supervised fine-tuning, show limited ability to learn directional entailments; 3) We present BoOQA, a robust, extrinsic, multilingual evaluation benchmark for directional predicate entailments, where various baselines are provided and analysed.[1]

2 Background and Setup

Language models have been used under a pretrain-finetune paradigm: the semantics of a token in context are learnt during pre-training and reflected in the dense encodings; when fine-tuning with a task-specific dataset, the model learns which area of its encoding space to look at. Therefore, if a pre-trained LM **cannot** be fine-tuned to solve a task, we cannot reject the null hypothesis that it does not encode the task. In §3, we look into RoBERTa (Liu et al., 2019) and BERT (Devlin et al., 2019) in particular, and examine whether they can be fine-tuned to learn directional predicate entailments.

Model   We adapt the supervised SOTA (Schmitt and Schütze, 2021b), a prompt fine-tuning method, for examining LMs.[2] We call it S&S here and below. S&S fits each premise-hypothesis pair into a few natural language prompts, such as "*John shopped in* IKEA*, which means that John went to* IKEA"; they then convert the task into sentence classification over instantiated prompts. It is a simple SOTA with few additional parameters, and the architecture allows directional judgements. Thus, it is an ideal "gauge" for directional ability of LMs.

Dataset   So far there are two popular predicate entailment datasets: LevyHolt (Levy and Dagan, 2016; Holt, 2019) and Sherliic (Schmitt and Schütze, 2019). We use LevyHolt in our §3 experiments, as it contains data entries ($p \vDash q?$) with their converses ($q \vDash p?$), making the ground truth directionality annotations available. We use the train/dev/test split as in Schmitt and Schütze (2021b).[3] In each data split, We further classify the entries into the following 4 sub-groups, in parentheses are the sizes of each sub-group in each split:

- *DirTrue* (251 / 64 / 892): directional true entailments where the premise entails the hypothesis, but not vice versa; for instance, *Person shopped in Location* $\vDash$ *Person went to Location*;

- *DirFalse* (251 / 64 / 892): directional non-entailments where the hypothesis entails the premise, but not vice versa; for instance, *Person went to Location* $\nvDash$ *Person shopped in Location*;

- *Paraphrases* (615 / 155 / 1939): symmetric paraphrases where the premise and the hypothesis entail each other; for instance, *Person arrived at Location* $\equiv$ *Person got to Location*;

- *Unrelated* (3255 / 831 / 9198): unrelated predicate pairs where the premise and the hypothesis have no entailment relations; for instance, *Person shopped in Location* $\nvDash$ *Person fell ill in Location*.

We define various subsets with pairs of sub-groups, which we introduce and discuss in §3.

Evaluation Metric   In predicate entailment detection, Area-Under-the-Curve with precision $> 50\%$ ($AUC_{50\%}$) has been the metric in use (Hosseini et al., 2018; Schmitt and Schütze, 2021b). It is a solid metric for comparison on the same dataset; however, we are comparing between different subsets, each with a different random baseline precision (i.e. the ratio of true entailments). If we were to set a common precision lower-bound, we would be biased toward those datasets with higher random baseline precisions. To make performance on different datasets comparable, we propose the metric of **Normalized AUC** (AUC_{norm}):

---

[1] Our code and datasets are published at https://github.com/Teddy-Li/LM-DirctionalInference.

[2] There is a follow-up work (Schmitt and Schütze, 2021a) to this, but we found it to have inferior generalisation performance; see Appendix B for details and a brief introduction.

[3] Except when an entry and its converse appear in different splits (e.g. one in train, the other in dev), where we randomly assign both into the same split, so as to avoid information leakage.

904

$$AUC_{norm} = \frac{AUC_\xi - \xi}{1 - \xi} \qquad (1)$$

where ξ is the random baseline precision. Intuitively, AUC_{norm} measures the ratio of area-above-random ($1 - \xi$) that falls below the precision-recall curve ($AUC_\xi - \xi$), see Appendix A for graphic illustration. AUC_{norm} allows us to take into account all gains against the random baseline, and level performance on all datasets to the same scale.

3 Prompt Fine-tuning LM with LevyHolt

In this section, we test for LMs' ability to learn directional entailments with the S&S prompt-based gauge model. We use RoBERTa-base as the primary subject, as it is used by SOTA Schmitt and Schütze (2021b), and is sufficiently lightweight for experiments to run efficiently. In Appendix C, we also report results on RoBERTa-large and BERT models for key experiments, where results are consistent. We use $S\&S_{subset}$ to denote S&S model fine-tuned on each $subset$.

Experiments are graphically summarized in Figure 1. Each edge denotes a LevyHolt subset made of the two sub-groups; the number on each edge is the AUC_{norm} that S&S achieves on each subset. For separating an Entailed sub-group from a Non-entailed one, the original labels are used; for separating two Entailed or two Non-entailed sub-groups, the one with more similar predicates (more paraphrastic) is assigned label "1", the other "0".[4]

Note that we fit S&S to a number of different subsets, so we cannot simply re-use the original hyper-parameters. Instead, to provide a level playing field, we follow Schmitt and Schütze (2021b) in log-uniformly sampling 100 hyper-parameter sets from their specified ranges; for each subset, we choose the one with best dev set result.

If a method is insensitive to directional entailments, then it would treat entailments as similarity between unordered pairs of predicates; it would model *Paraphrases*, *DirTrue* and *DirFalse* similarly, where *DirTrue* and *DirFalse* entries are conceptually somewhat "semi-paraphrastic".

---

[4] We acknowledge that Schmitt and Schütze (2021b) use hand-crafted prompts tuned for entailment detection, so they may be sub-optimal for separating same-label sub-groups; we argue that fixed-prompt LM tuning models are not too sensitive to their specific prompts (Logan IV et al., 2021; Webson and Pavlick, 2022); nonetheless, we also report results from a continuous-prompt model (Schmitt and Schütze, 2021a) in Appendix B, where results are very similar.
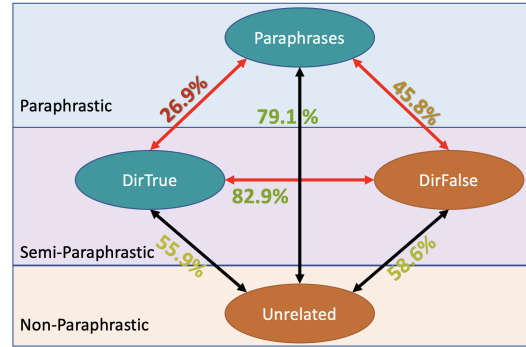

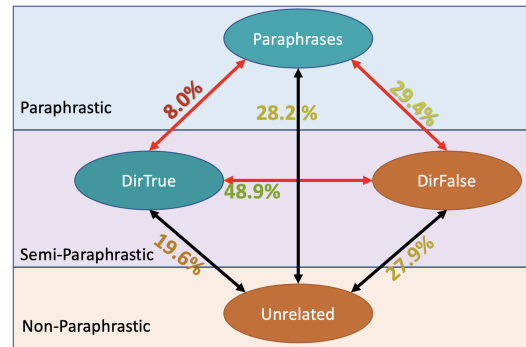Figure 1: S&S models on mesh of pairs of sub-groups, results in AUC_{norm}.

Figure 2: Hypothesis-only artefact baselines on mesh of pairs of sub-groups, results in AUC_{norm}.

If a method is sensitive to directional entailments, it should be able to discriminate between each pair of the four sub-groups. Particularly, it should additionally be able to separate sub-groups in the upper triangle of the mesh, coloured red in Figure 1. Among these three tests of directionality, *DirTrue-DirFalse* is the most challenging: a method with no sensitivity to directionality should be at chance and get 0% AUC_{norm}; this is traditionally called the *directional subset* (Holt, 2019). For the other two subsets, a symmetric measure would do above random by identifying entries in *DirTrue* / *DirFalse* as statistically less similar than *Paraphrases*.

Below we discuss findings around the mesh and the triangle. The S&S model yields 77.7% AUC_{norm} when trained and tested on the **full** Levy-Holt dataset, which we provide for readers' reference.

3.1 The S&S Triangle

The red triangle in Figure 1 presents mixed messages about the directionality of RoBERTa LM: on the most challenging *DirTrue-DirFalse* subset, it achieves an apparently excellent AUC_{norm} of 82.9%; however, on the other subsets, it gets mediocre results at 26.9% and 45.8% respectively.

Train/Dev \ Test	Directional	Symmetric	Full
Directional	**82.9**	9.9	24.7
Symmetric	0.2	**79.1**	61.3
Full	46.4	84.8	**77.7**

Table 1: Generalization performance of RoBERTa-base S&S classifier on the *Directional* and *Symmetric* subsets of LevyHolt. Values are in % of AUC_{norm}.

AUC_{norm} (%)	$S\&S$	$\widehat{S\&S}$
Para-DirTrue	26.9	19.8 (-7.1)
Para-DirFalse	45.8	35.6 (-10.2)

Table 2: Comparison between $S\&S$ with regular and $\widehat{S\&S}$ with symmetric prompts. *Paraphrases-DirTrue* and *Paraphrases-DirFalse* subsets are concerned.

For the *directional* subset (*DirTrue-DirFalse*), the 82.9% AUC_{norm} not only surpasses the 77.7% for Full LevyHolt, but is also on par with the 79.1% for its mirroring *Symmetric* subset (*Paraphrases-Unrelated*), which should be easier by human judgement. Paradoxically for such a challenging subset, the *Directional subset* enjoys the best performance in the mesh.

To understand this result, we did a generalisation experiment between *Directional* and *Symmetric*, the two disjoint, complementary subsets of Levy-Holt. As results in Table 1 show, classifiers from the two subsets do not generalise to each other, and neither does $S\&S_{Directional}$ generalise to full LevyHolt. That is to say, either the two kinds of "entailments", *Directional* and *Symmetric*, are different tasks from the LM's perspective, or the S&S classifier is overfitting to the *directional* subset.

For the *Paraphrases-DirX* subsets, results are far less impressive. To measure how well symmetric baselines do in these subsets, we train two strictly-symmetric S&S models, one on *Paraphrases-DirTrue*, the other on *Paraphrases-DirFalse*. For these strictly-symmetric models we enforce all prompts to be in pairs of reverses (e.g. for the example in §2, we would add "*John went to* IKEA, *which means that John shopped in* IKEA"). That way we guarantee from the input that no directional judgements can be made. We call these symmetric-prompt models $\widehat{S\&S}$. From the results in Table 2, we find that for both *Paraphrases-DirTrue* and *Paraphrases-DirFalse*, there is only a modest difference between the performance of $S\&S$ classifier

and $\widehat{S\&S}$. This suggests that despite the results from the *Directional Subset*, RoBERTa LM shows limited ability to detect directional entailments.

3.2 The Artefacts Triangle

From previous discussions, we notice that the scores for the *directional* subset is anomalously high. Below we reveal that this anomaly is an effect of dataset artefacts, and that the artefacts in question are quite specific to the *directional* subset and generally less prominent in the other subsets. Artefacts aside, the S&S classifiers do not show strong abilities to learn directional entailments.

It is difficult to identify sources of artefacts by manual inspection; on the other hand, Poliak et al. (2018) have shown that hypothesis-only (H-only) models can be a strong proxy for an artefact baseline. Inspired by their findings, we instead use H-only model as a proxy for the aggregated strength of artefacts. For H-only model we use a restricted version of S&S classifier, where we mask all premises with the word "true".[5]

We report the H-only model's results on the same mesh of subsets in Figure 2. For every subset, the H-only model still trails behind the S&S classifiers. These gaps are partly explained by the fact that the H-only model does not capture all existing artefacts, but is merely a proxy to their strengths.

As shown, the *Directional* subset indeed has particularly strong artefacts to exploit, with the H-only model reaching 48.9% AUC_{norm}, far above the other subsets. Between *Paraphrases-DirTrue* and *Paraphrases-DirFalse*, the relative performance of S&S model on them is aligned with their relative strengths of artefacts; this means, for RoBERTa, *Paraphrases* is in fact similarly separable from *DirTrue* and *DirFalse*, in line with expectation.

Also interesting is the comparison between the *directional* and *symmetric* subsets. The two subsets had similar S&S performances; however, there is a difference of over 20% between their hypothesis-only artefact strengths. That means the S&S classifier is after all far better at the *symmetric* subset than the *directional* one.

For a crude ranking, we inspect each subset according to the FullModel-HOnly ratio by AUC_{norm} (lower the stronger artifacts). We find at rock bottom the *Paraphrases-DirFalse* and *DirTrue-DirFalse* subsets with this ratio at 1.55 and

---

[5]We use "*true*" to mask the premise because, when the premise is always true, the correctness of instantiated prompts depends solely on the hypothesis.

1.70 respectively, indicating that their full-model scores are the heaviest over-estimations; next up is the 2.10 for *DirFalse-Unrelated*, all the other subsets have this ratio around 3.

3.3 The Story behind Artefacts - A Summary

From the S&S triangle in Figure 1 and the artefact triangle in Figure 2, we make the findings below.

We find that *DirTrue* and *DirFalse* look similarly separable from *Paraphrases*, as well as from *Unrelated*. Moreover, for *DirTrue* and *DirFalse*, it is consistently harder to separate them from *Paraphrases* than from *Unrelated*. This suggests RoBERTa is modelling both of them as closer to *Paraphrases* than to *Unrelated*. This is typical for symmetric measures, but also possible for directional ones.

However, as we examine the directionality triangles, we find that RoBERTa shows poor directionality in *Paraphrases-DirX* subsets. It also does far worse on the *directional* subset than on *symmetric*, although the AUC_{norm} is inflated by dataset artefacts.

In conclusion, we find evidence that RoBERTa (and BERT, see Appendix C) are poor learners of directional inference. However, AUC_{norm} values are, after all, comparable but not additive, so we still cannot measure the net competence by deducting H-only scores from S&S scores. Thus, in order to draw final conclusions on the directionality of LMs, we present BoOQA, a robust extrinsic evaluation dataset.

4 Boolean Open QA (BoOQA) Dataset

We have shown that the LevyHolt *directional* subset is especially infested with dataset artefacts and is not a robust benchmark for supervised methods. With the directionality triangle, we have shown evidence that LMs have limited ability to learn directional entailments. However, this experiment is tedious and inconclusive: even solving the triangle does not guarantee solving directionality, as artefacts always persist; besides, we are still unable to compare between supervised and unsupervised methods.

To establish an extrinsic, robust and realistic evaluation benchmark for directional predicate entailments, and to encourage future research on the topic, we present the Boolean Open-QA dataset (BoOQA).

BoOQA is an automatically constructed dataset in English and Chinese. It draws inspirations from the *machine reading at scale* task (Chen et al., 2017) as well as the QA evaluation in (McKenna et al., 2021). The task is formalized as: given a proposition and a large pool of context articles, determine whether the queried proposition is true based on the context pool.[6]

4.1 Dataset Elicitation

In principle, we follow McKenna et al. (2021) in building this dataset. We separate large news corpora into context windows of 3-day time spans, select the frequent argument pairs as starring arguments, then select frequent relations about these starring arguments as positive propositions. We generate negatives from positives, by keeping the argument pairs unchanged, and selecting negative predicates from the WordNet hyponyms of positive predicates. For instance, for a positive proposition "John goes to IKEA", we may derive a potential negative proposition "John drives to IKEA".

Negative predicates are hyponyms that are felicitous but wrong in the current context: we check if a hyponym is felicitous by its corpus-wide presence; we check if a hyponym is wrong for the current context by its context-window-wide absence with the corresponding argument pair [7].

WordNet hyponyms are good candidate negative predicates because: 1) they are guaranteed to be similar to their corresponding positives, thus adversarial to symmetric similarity; 2) they are more specific than their positives and less likely to be true, so the risk of getting false negatives is reduced.

We take samples from the positive and negative propositions, and blend them to create the BoOQA dataset. For evaluation, we take these propositions as hypotheses, and retrieve premises from the corresponding context windows[8]. The task is then to judge the truthfulness of the hypotheses given the context windows. For a good entailment measure, positive hypotheses should be entailed by context more confidently than negatives, where context means other mentions of the same starring argu-

---

[6]We consider only propositions with binary relations; unary relations have only one argument for context disambiguation, thus are naturally noisier.

[7]This is based on the assumption that a comprehensive pool of news articles would follow the Gricean cooperative principle of communication (Davis, 2019), that it would include all and only the known facts.

[8]To avoid producing misleadingly optimistic scores, we force all methods to ignore the exact sentences from which the hypotheses (or their corresponding hypernyms) are extracted.

ments within the same context window.

For English, we follow McKenna et al. (2021) in using NewsCrawl (Barrault et al., 2019) as our source corpus; for Chinese, we use the CLUE news corpus (Xu et al., 2020). As we include entailment graphs as unsupervised baselines, we remove from these corpora the articles seen in entailment graph induction corpora; on the other hand, we refrain from deleting overlaps with the much larger LM-pretrain corpora, to avoid biasing our dataset toward non-mainstream coverage.

Given the analogy between our BoOQA dataset and the evaluation in McKenna et al. (2021), we leave most details of dataset construction to Appendix D and refer readers also to their paper. Here we highlight our additional efforts for the quality and robustness of BoOQA.

Quality When generating negative predicates, we look for hyponym replacements for every span in the positive predicates, instead of just the head verbs. For example, for "play game with", we can generate candidate negative "**foul** game with" as well as "play **practice game** with". The former may be absent from the whole corpus, the latter may be present somewhere in the corpus but not the current context window; then the former would be discarded and the latter kept. This allows us to generate negatives for more complex predicates, avoiding biasing toward shorter, simpler predicates; it is especially useful in Chinese, where word boundaries are unmarked and WordNet entries are present across multiple levels of granularity.

When collecting hyponyms of WordNet matches, instead of looking into the first synset of each match, for English we disambiguate matches to their most relevant synsets with a WSD method (Yap et al., 2020). The WSD method takes in a span in sentential context as input, and produces the most likely synset for that span given the context. This way, we find more accurate hyponyms and generate more challenging negatives that are similar to their positives. Unfortunately, no such models or datasets are available for Chinese, so we have to fall back to using the first synset.

Thirdly, we raise the bar for candidate negatives to be considered "wrong in the current context": not only must the predicate itself be absent from the context window (with the current arguments), but all its WordNet synonyms must also be absent. This stricter filter further reduces false negatives.

Fourthly, we expand our source corpora to entire

the NewsCrawl / CLUE corpora instead of their slices, so we avoid biasing toward any particular time spans. With larger corpora, we impose larger thresholds for starring-arguments, so the events in question are thoroughly discussed, fewer positives cannot be inferred from the context, namely fewer false positives[9].

Lastly, we sample data entries in bundles of positives and negatives. Whenever a positive is sampled, its corresponding negatives must also be sampled, and vice versa. This strengthens our dataset in the way that each method needs to assign higher confidence values to positives than their corresponding negatives to get good results.

Robustness While inspecting the QA dataset in McKenna et al. (2021), we find that their positive and negative propositions are separable even without context: the generated negatives simply look less plausible than their corresponding positives. We again use hypothesis-only (H-only) models to quantify this effect. Since we use the context to retrieve premises, the H-only model by definition means "without context". Similarly to Schmitt and Schütze (2021b), we sample train / dev2 sub-splits out of the dev set for fine-tuning.[10]

Ideally, without context, the negatives should be inseparable from the positives, the H-only model should do badly. However, on McKenna dataset, it got 78.3% AUC_{norm}. That means McKenna dataset is also infiltrated by spurious correlations. Although this is not fatal since the dataset is only for evaluation so models are not exposed to such artefacts, it is still desirable to improve robustness by bringing the negatives closer to the positives.

To this end, we also raise the bar for candidate negatives to be considered "felicitous": they have to be not just present in the corpus, but also frequent enough. We set the same frequency thresholds for felicitousness between positives and negatives.

In addition, when sampling from the populations, we create negatives samples such that they have similar frequency distributions as the positives population. We calculate the ratio of predicates in positives population falling into each frequency range, then sample the negatives proportionally.[11]

We apply H-only model to BoOQA and report the results in Table 3. $BoOQA_{EN}$ enjoys a rea-

---

[9]False positives are still true events, but models should not be expected to label them as "positive" given the context.

[10]For details of this H-only model see Appendix E.1.

[11]We sample positives/negatives in bundles, so the positive samples are generally more frequent than the population.

Train / Dev Set	Test Set	AUC_{norm}
$BoOQA_{EN}$	$BoOQA_{EN}$	51.0
$BoOQA_{ZH}$	$BoOQA_{ZH}$	84.2
$McKenna$	$McKenna$	78.3
$LevyHolt_{Full}$ EN	$BoOQA_{EN}$	2.8
$LevyHolt_{Dir}$ EN	$BoOQA_{EN}$	1.1
$LevyHolt_{Sym}$ EN	$BoOQA_{EN}$	0.8
$LevyHolt_{Full}$ ZH	$BoOQA_{ZH}$	4.0
$LevyHolt_{Dir}$ ZH	$BoOQA_{ZH}$	1.2
$LevyHolt_{Sym}$ ZH	$BoOQA_{ZH}$	2.4

Table 3: Robustness measured by H-only model performance, values in %, lower is better. McKenna is the QA dataset in McKenna et al. (2021), $LevyHolt_{Dir}$ and $LevyHolt_{Sym}$ are *directional* and *symmetric* subsets.

sonably significant drop against McKenna dataset for H-only performance, meaning less exploitable spurious correlations. However, we acknowledge that generated negatives are naturally hard to blend with collected positives, and there remain clues in the hypothesis themselves indicating their truthfulness. For $BoOQA_{ZH}$, the H-only result is still high even with our above efforts. Fortunately, for both languages, H-only models trained on LevyHolt subsets cannot identify the truthfulness of BoOQA hypotheses, indicating that BoOQA remains safe as an extrinsic benchmark for evaluation only.

# of	positives	negatives
$BoOQA_{EN}$ dev	21809	36782
$BoOQA_{EN}$ test	21773	36755
$BoOQA_{ZH}$ dev	18855	31117
$BoOQA_{ZH}$ test	19092	31479

Table 4: Number of positive / negative entries in each subset of BoOQA.

Statistics and Discussions We provide dev/test splits in BoOQA, each has circa 20k positives and 40k negatives, as shown in Table 4. Positives and negatives are sampled in bundles where at most 2 negatives correspond to each positive (subject to availability).

Notably, the size of 60K entries is chosen empirically, as we find results to have already stabilized at this size. Though the number of entries is 60K, the number of entailments to be checked is magnitudes larger: in assigning a score to each entry, a method must inspect the relation from each piece of evidence to the proposition at question. Moreover, since the construction process is fully automatic, it is easily scalable and extendable to other corpora.

In Appendix E.2, we also present a human analysis for the felicitousness of BoOQA entries, and show that 85%+ correctness can be expected for both labels and both languages, which is relatively high and even comparable to some crowd-annotated datasets.

On BoOQA, directional entailments are not needed to perform above random, but are needed to reach high precisions. Consider a positive proposition A and its corresponding negative B; a third proposition C is retrieved from context articles. In a simplified sense, a model is correct as long as it scores $C \vDash A$ higher than $C \vDash B$. Now suppose there are two perfect measures of entailments, except that one is directional, the other symmetric. When neither queried propositions entail the context (i.e. $A \nvDash C$ and $B \nvDash C$), the two measures behave similarly; otherwise, when some of these reverse-direction entailments hold, chances are that the symmetric measure would fall into the trap and produce spurious results. Through this property, directionality is involved in BoOQA dataset; we offer a detailed analysis in Appendix E.3.

Moreover, although BoOQA negatives are built off WordNet hyponyms of positives, the directional entailments tested are not confined to hyponymy/hypernymy. This is because BoOQA tests for entailment strengths from context-propositions to the queries, instead of between the positive (hypernym) and the negative (hyponym) queries themselves. This allows us to attend to all kinds of directional entailments as is present in the general news domain, for instance, precondition-consequences.

4.2 Experiments and Baselines

In Table 5, we present baselines on BoOQA test set with 3 types of methods: unsupervised LMs, supervised LM-driven methods and entailment graphs.

Each method retrieves supporting evidence from context, which are the extracted propositions in the same context window involving the same arguments as the queried hypothesis; they then calculate a score from each piece of evidence to the hypothesis, and take the highest score w.r.t any evidence as the final confidence score for a hypothesis.

Unsupervised LMs They compute cosine similarities between BERT-base encodings of the retrieved supporting evidence and the hypothesis, as confidence scores. We also tried BERT-large, RoBERTa as well as alternative retrieval approaches, we leave them to Appendix G, as their dev set results are not

AUC_{norm} (%)	$BoOQA_{En}$	$BoOQA_{Zh}$
$LM_{unsupervised}$	15.9	30.6
$S\&S_{Full}$	25.6	23.1*
$\widehat{S\&S_{Full}}$	26.2	-
$S\&S_{Symmetric}$	25.1	-
$S\&S_{Directional}$	15.1	-
EG BInc	29.8	39.5
EG CNCE	**34.5**	-
EG EGT2	26.8	-

Table 5: Baselines on BoOQA test set in English and Chinese. All methods are taken out-of-the-box. "EG XX" are entailment graphs with various entailment scores as described below.

Figure 3: Precision-recall curves for the representative methods of each class on $BoOQA_{EN}$ test set.

superior to the condition reported.

Supervised LMs They use $S\&S_{subset}$ models with RoBERTa-base to compute entailment scores from retrieved evidence to the hypotheses. We present $S\&S_{full}$ as supervised LM baseline for $BoOQA_{EN}$ and $BoOQA_{ZH}$. On $BoOQA_{EN}$, we also compare $S\&S_{Directional}$, $S\&S_{Symmetric}$ and $\widehat{S\&S_{Full}}$ (trained on full LevyHolt with symmetric prompts, see §3.1). On $BoOQA_{EN}$ we also provide large model results in Appendix G; those are not higher than with base model.

Entailment Graphs They are unsupervised entailment detection methods, where nodes in each graph are predicates, and directed edges between nodes represent entailment relations between predicates. Entailment graphs are induced from textual corpora with directional co-occurrence signals. At test time, entailment scores are collected by looking up the edge weights from each predicate-in-evidence to the predicate-in-hypothesis. For English, we evaluate with BInc (Hosseini et al., 2018), CNCE (Hosseini et al., 2021) and EGT2 (Chen et al., 2022); for Chinese, we evaluate with BInc (Li et al., 2022).

On $BoOQA_{EN}$, we re-examine the $S\&S$ gauge models in comparison to various unsupervised methods. $LM_{unsupervised}$ does poorly, yet $S\&S_{Directional}$ does even worse. This verifies that its judgements are polluted by dataset artefacts. Moreover, it does not matter whether $S\&S$ is exposed to only $symmetric$ entries or uses purely symmetric prompts, the performance is almost identical. This means RoBERTa is not learning from directional supervision (in LevyHolt) the directional ability to make progress on BoOQA. These results echo our findings in §3. Though being supervised and only with domain transfer, the $S\&S$ models fail to outperform fully unsupervised
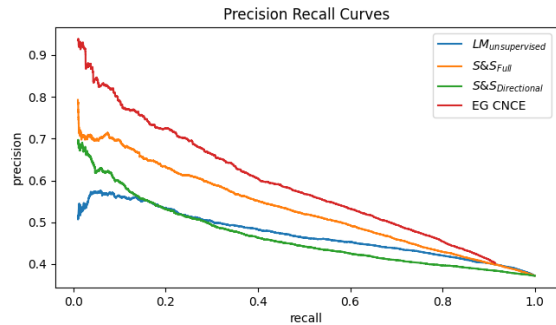
entailment graphs on BoOQA.

Entailment graph methods generally have decent performance. EG CNCE enjoys a considerable advantage, particularly at the higher-precision end. While it may seem surprising for an unsupervised method to prevail in high-precisions range, this is expected given that in BoOQA, directionality is required for methods to reach high precisions. On the other hand, these discrete entailment graphs are hampered by sparsity; although a score is retrieved for most queries with the help of fuzzy matching[12], the high precisions are short-lived measured by recall, dropping rapidly to the moderate range of sub-60%. This suggests generalising unsupervised directional entailment graphs to continuous encoding spaces could be a promising direction to follow.

5 Related Work

Related work to our paper broadly fall into three categories: Language-Model prompting, entailment graphs and natural language inference.

LM-prompting methods convert input in target tasks into natural language sentences (prompts), thereby converting target tasks into either language modelling or sentence classification. Petroni et al. (2019) probed LM for factual knowledge, where Jiang et al. (2020) proposed automatic prompt discovery; on the other hand, GPT-3 (Brown et al., 2020) showed large LMs can do few-shot / zero-shot transfer with prompts, and Schick and Schütze (2021) showed a similar effect with smaller LMs and natural language prompts. Liu et al. (2021) provided a comprehensive review of LM-prompting, which we refer readers to for more details.

Entailment graphs (EGs) are graphs of predicates where directed edges represent entailment

---

[12]Ignoring semantic role labels for arguments in the entailment graph representations of predicates; see Appendix G for more details and a comparison of Pr-Rec curves.

relations between pairs of predicates. Weights on edges are entailment scores, induced from large textual corpora via unsupervised learning, based on directional co-occurrence signals; examples include BInc (Szpektor and Dagan, 2008; Hosseini et al., 2018), link prediction (Hosseini et al., 2019) and the LM-driven CNCE (Hosseini et al., 2021). Being fully unsupervised, EGs are sheltered from artefacts in training data, but are at the same time subject to noisy co-occurrence signals and sparsity common to discrete language resources.

Notably, while CNCE EG (2021) uses LM, it involves contextualized encodings for premises, and uncontextualized encodings for hypotheses, trained from scratch with millions of unsupervised data entries. The architecture and abundant supervision enables CNCE to encode directional entailments. EGT2 (Chen et al., 2022) is another LM-driven EG contemporary to our paper, with intermediate fine-tuning on MNLI (Williams et al., 2018) and PPDB (Ganitkevitch et al., 2013) annotations. They report good resuts on the *directional* subset of LevyHolt[13], but perform mediocrely on BoOQA. We speculate that this could be due to instabilities of the small LevyHolt *directional* test set.

Natural Language Inference is concerned with the general task of "Does sentence X entail sentence Y?". Classic NLI datasets such as SNLI (Bowman et al., 2015) or MNLI (Williams et al., 2018) are populated with entries involving hypernyms of nouns, logic reasoning like $A \wedge B \to B$ and "A under the condition of B" $\to A$, whereas predicate entailments are rarely involved. Predicate entailment datasets like LevyHolt (Levy and Dagan, 2016; Holt, 2019), SherlIic (Schmitt and Schütze, 2019) or our BoOQA dataset, are a subclass of NLI datasets focused on inference with predicates, where the arguments and adjuncts are carried over from premise to hypothesis. In particular, LevyHolt and our BoOQA dataset both feature the property of directionality, where the entailment is valid in one direction, but invalid in the other.

General NLI datasets are prune to simple dataset artefacts such as length, lexical overlaps or hypothesis-only models (Gururangan et al., 2018; Poliak et al., 2018; Kalouli et al., 2021; McCoy et al., 2019). These echo our observations on predicate entailments, and underscore the importance of a robust and extrinsic evaluation.

---

[13]Their reported directional subset is different from the original one in Holt (2019), see Appendix H for details.

6   Conclusion

In this paper, we show that existing LM prompting methods show limited ability to learn directional entailments, and instead overfit to dataset artefacts in the LevyHolt *directional* subset. We present BoOQA, a robust and extrinsic multilingual evaluation benchmark on directional predicate entailments as a Boolean Open QA dataset. On BoOQA, we show that LM prompting methods do not learn to pay attention to directionality from directional supervision, while the entailment graph methods remain limited by sparsity. We point to the generalisation of entailment graphs to continuous spaces and directionality-oriented intermediate pre-training of LMs as directions of future work.

Limitations

The remaining limitations of our work are two-fold. Firstly, our experiments are done primarily with RoBERTa and BERT models. This choice is taken following the positive results reported in Schmitt and Schütze (2021b,a). We argue that the consistency among these models indicates a broader generalisability of our conclusions, however we are unfortunately unable to enumerate over the wide variety of language models to prove this point. Particularly, due to limits to our computational resources, we are unable to experiment with extremely large language models such as GPT-3. We release the code and data for the experiments as described in §3 and §4, and encourage the community to try out any language models they might be interested in.

The second aspect of limitation is, despite our efforts to improve the robustness of our BoOQA dataset, there remain correlations between just the surface forms of the propositions and their truthfulness, which could be exploitable under a supervised setting. We argue that one cannot eliminate such correlations without heavily manipulating which propositions are of interest to the dataset, which is against our aim of preserving the natural distribution of propositions in large-scale news corpora; we also argue that this is harmless as long as BoOQA remains an extrinsic benchmark, namely, the training set of any supervised method should not have the same kinds of correlations. We have checked that the LevyHolt dataset and its various subsets satisfy this criterion; we advise that readers attempting this dataset with their own methods do the same check to ensure comparable results.

Acknowledgements

References

Loïc Barrault, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, Christof Monz, Mathias Müller, Santanu Pal, Matt Post, and Marcos Zampieri. 2019. Findings of the 2019 Conference on Machine Translation (WMT19). In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 1–61, Florence, Italy. Association for Computational Linguistics.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

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

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading Wikipedia to Answer Open-Domain Questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879, Vancouver, Canada. Association for Computational Linguistics.

Zhibin Chen, Yansong Feng, and Dongyan Zhao. 2022. Entailment Graph Learning with Textual Entailment and Soft Transitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5899–5910, Dublin, Ireland. Association for Computational Linguistics.

Wayne Davis. 2019. Implicature. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*, fall 2019 edition. Metaphysics Research Lab, Stanford University.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. 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 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. PPDB: The Paraphrase Database. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 758–764, Atlanta, Georgia. Association for Computational Linguistics.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. 2018. Annotation Artifacts in Natural Language Inference Data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112, New Orleans, Louisiana. Association for Computational Linguistics.

Xavier Holt. 2019. Probabilistic Models of Relational Implication. *arXiv:1907.12048 [cs, stat]*. ArXiv: 1907.12048.

Mohammad Javad Hosseini, Nathanael Chambers, Siva Reddy, Xavier R. Holt, Shay B. Cohen, Mark Johnson, and Mark Steedman. 2018. Learning Typed Entailment Graphs with Global Soft Constraints. *Transactions of the Association for Computational Linguistics*, 6:703–717.

Mohammad Javad Hosseini, Shay B. Cohen, Mark Johnson, and Mark Steedman. 2019. Duality of Link Prediction and Entailment Graph Induction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4736–4746, Florence, Italy. Association for Computational Linguistics.

Mohammad Javad Hosseini, Shay B. Cohen, Mark Johnson, and Mark Steedman. 2021. Open-Domain Contextual Link Prediction and its Complementarity with Entailment Graphs. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2790–2802, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020. How Can We Know What Language Models Know? *Transactions of the Association for Computational Linguistics*, 8:423–438. Place: Cambridge, MA Publisher: MIT Press.

Aikaterini-Lida Kalouli, Livy Real, Annebeth Buis, Martha Palmer, and Valeria de Paiva. 2021. Annotation Difficulties in Natural Language Inference. In *Anais do Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana (STIL)*, pages 247–254. SBC. ISSN: 0000-0000.

Keshav Kolluru, Vaibhav Adlakha, Samarth Aggarwal, Mausam, and Soumen Chakrabarti. 2020. OpenIE6: Iterative Grid Labeling and Coordination Analysis for Open Information Extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3748–3761, Online. Association for Computational Linguistics.

Omer Levy and Ido Dagan. 2016. Annotating Relation Inference in Context via Question Answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 249–255, Berlin, Germany. Association for Computational Linguistics.

Tianyi Li, Sabine Weber, Mohammad Javad Hosseini, Liane Guillou, and Mark Steedman. 2022. Cross-lingual Inference with A Chinese Entailment Graph. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1214–1233, Dublin, Ireland. Association for Computational Linguistics.

Xiao Ling and Daniel S. Weld. 2012. Fine-grained entity recognition. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, AAAI'12, pages 94–100, Toronto, Ontario, Canada. AAAI Press.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. *arXiv:2107.13586 [cs]*. ArXiv: 2107.13586.

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:1907.11692 [cs]*. ArXiv: 1907.11692.

Robert L. Logan IV, Ivana Balažević, Eric Wallace, Fabio Petroni, Sameer Singh, and Sebastian Riedel. 2021. Cutting Down on Prompts and Parameters: Simple Few-Shot Learning with Language Models. Technical Report arXiv:2106.13353, arXiv. ArXiv:2106.13353 [cs] type: article.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the Wrong Reasons: Diagnosing Syntactic Heuristics in Natural Language Inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.

Nick McKenna, Liane Guillou, Mohammad Javad Hosseini, Sander Bijl de Vroe, Mark Johnson, and Mark Steedman. 2021. Multivalent Entailment Graphs for Question Answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10758–10768, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018. Hypothesis Only Baselines in Natural Language Inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 180–191, New Orleans, Louisiana. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Siva Reddy, Mirella Lapata, and Mark Steedman. 2014. Large-scale Semantic Parsing without Question-Answer Pairs. *Transactions of the Association for Computational Linguistics*, 2:377–392.

Timo Schick and Hinrich Schütze. 2021. It's Not Just Size That Matters: Small Language Models Are Also Few-Shot Learners. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2339–2352, Online. Association for Computational Linguistics.

Martin Schmitt and Hinrich Schütze. 2019. SherLIiC: A Typed Event-Focused Lexical Inference Benchmark for Evaluating Natural Language Inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 902–914, Florence, Italy. Association for Computational Linguistics.

Martin Schmitt and Hinrich Schütze. 2021a. Continuous Entailment Patterns for Lexical Inference in Context. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6952–6959, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Martin Schmitt and Hinrich Schütze. 2021b. Language Models for Lexical Inference in Context. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics:*

Main Volume, pages 1267–1280, Online. Association for Computational Linguistics.

Idan Szpektor and Ido Dagan. 2008. Learning Entailment Rules for Unary Templates. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 849–856, Manchester, UK. Coling 2008 Organizing Committee.

Albert Webson and Ellie Pavlick. 2022. Do Prompt-Based Models Really Understand the Meaning of their Prompts? Technical Report arXiv:2109.01247, arXiv. ArXiv:2109.01247 [cs] type: article.

Julie Weeds and David Weir. 2003. A General Framework for Distributional Similarity. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 81–88.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Liang Xu, Hai Hu, Xuanwei Zhang, Lu Li, Chenjie Cao, Yudong Li, Yechen Xu, Kai Sun, Dian Yu, Cong Yu, Yin Tian, Qianqian Dong, Weitang Liu, Bo Shi, Yiming Cui, Junyi Li, Jun Zeng, Rongzhao Wang, Weijian Xie, Yanting Li, Yina Patterson, Zuoyu Tian, Yiwen Zhang, He Zhou, Shaoweihua Liu, Zhe Zhao, Qipeng Zhao, Cong Yue, Xinrui Zhang, Zhengliang Yang, Kyle Richardson, and Zhenzhong Lan. 2020. CLUE: A Chinese Language Understanding Evaluation Benchmark. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4762–4772, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Boon Peng Yap, Andrew Koh, and Eng Siong Chng. 2020. Adapting BERT for Word Sense Disambiguation with Gloss Selection Objective and Example Sentences. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 41–46, Online. Association for Computational Linguistics.

A Graphic Illustration for AUC_{norm}

See Figure 4 for a graphic illustration of AUC_{norm}, and a comparison between different metrics.

B Details and Comparison of S&S Models

B.1 Between Discrete / Continuous Prompts

The S&S models (Schmitt and Schütze, 2021b,a) are LM-prompting models based on RoBERTa. Of which, S&S-discrete (Schmitt and Schütze, 2021b) falls into the category of "Fixed-prompt LM Tuning" according to the taxonomy in Liu
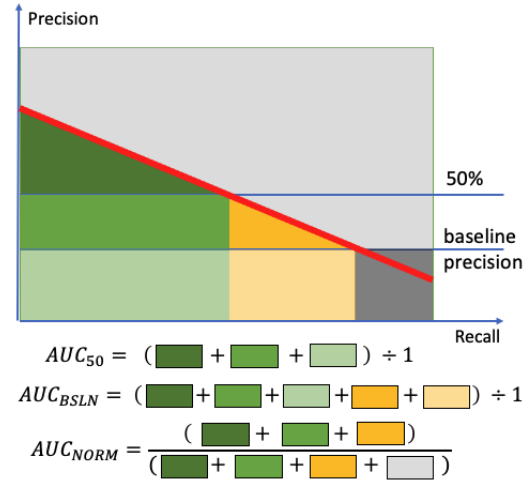

Figure 4: Diagram illustration of AUC_{norm} in comparison to other AUC metrics. Red and bolded is a simplified precision-recall curve, each AUC metric is represented as ratios of area of the respective coloured blocks. The total area of all blocks equals to 1. Note that in practice the precision of any method rarely comes below random (as that would mean the least confident entries carry some kind of reverse judgements), and in any case not beyond the scale reasonable for random fluctuations. That is, real pr-rec curves do not cut deeply into the dark grey area like the over-simplified demo.

et al. (2021); whereas its follow-up work S&S-continuous (Schmitt and Schütze, 2021a) falls into the category of "Prompt+LM Tuning".

S&S-discrete uses human-engineered natural language prompts. Given a premise-hypothesis pair, it instantiates each prompt with the pair, feeds them into the LM, thus converts the task into a regular binary classification task from the LM representations of the instantiated prompts.

Notably, we discovered a bug in the implementation of Schmitt and Schütze (2021b): due to missing negations, for experiments on LevyHolt dataset, the manual prompt (d) and (e) listed in Schmitt and Schütze (2021b) were effectively converses of prompt (c) and (a) respectively. After fixing the bug, we found the model performance with RoBERTa-base, measured by $AUC_{50\%}$, as reported in their original paper, rises from 76.9% to 80.7%; correspondingly, the AUC_{norm} value rises from 75.3% to 77.7%.

S&S-continuous is in essence similar to S&S-discrete, except that it replaces the embeddings of human-engineered natural language prompt tokens with sequences of trainable continuous embeddings. S&S-continuous reported SOTA results, outperforming S&S-discrete; however, we stick

Model	LevyHolt	SherlIic
Trained with LevyHolt		
S&S-discrete	80.7 (77.7)	69.5 (59.5)
S&S-continuous	79.2 (76.2)	65.1 (54.3)
Trained with SherlIic		
S&S-discrete	35.5 (47.7)	63.8 (51.5)
S&S-continuous	25.1 (32.8)	66.5 (57.9)

Table 6: Generalization experiment between Levy-Holt and SherLIic datasets, examined models are S&S-discrete and S&S-continuous. Performance is measured in $AUC_{50\%}$, where the AUC_{norm} values are appended in brackets. Note that reported values are from our reproduction with our 100 sets of hyper-parameter configs, so the exact values slightly differ from the originals.

with the S&S-discrete model for our experiments for the following reasons:

- After fixing the bug in S&S-discrete implementation, its genuine performance is not worse than that reported for S&S-continuous;

- We did a generalization experiment between the two predicate entailment datasets, Levy-Holt (Levy and Dagan, 2016) and SherlIic (Schmitt and Schütze, 2019), where we found that S&S-discrete is better at generalizing from one dataset to the other, as shown in Table 6; that suggests the S&S-discrete is less prone to overfitting;

- For the same LM backbone, S&S-discrete has a smaller computational footprint.

B.2 Subsets with Same-label Sub-groups

AUC_{norm} (%)	Discrete	Continuous
Para-DirTrue	26.9	25.5
DirFalse-Unrl	58.6	62.2

Table 7: Performance of $S\&S$ model on *Paraphrases-DirTrue* and *DirFalse-Unrelated* subsets. *Discrete* and *Continuous* are S&S-discrete and S&S-continuous.

For the subsets with same-label sub-groups, namely, the subsets whose two sub-groups are either both true entailments or both non-entailments, we assign positive labels to the sub-group with higher level of relevance, and negative labels to the other. As discussed in §3, we acknowledge that Schmitt and Schütze (2021b) uses hand-crafted prompts which are designed for labelling the truthfulness of a candidate entailment, so they may not be ideal for separating same-label sub-groups.

Test Train/Dev	Directional	Symmetric	Full
Directional	71.1	2.6	11.6
Symmetric	2.1	90.1	70.9
Full	52.4	90.7	83.7

Table 8: Generalization performance of **RoBERTa-large** S&S classifier on the *DirTrue-DirFalse* (Directional) and *Paraphrases-Unrelated* (Symmetric) subsets of LevyHolt. Values are in % of AUC_{norm}. Rows are train/dev subsets, columns are test subsets.

AUC_{norm}	$S\&S$	$\widehat{S\&S}$
Para-DirTrue	43.4	33.2 (-10.2%)
Para-DirFalse	59.4	49.9 (-9.5%)

Table 9: Comparison between **RoBERTa-large** $S\&S$ classifiers (regular) and $\widehat{S\&S}$ (symmetric prompts). *Paraphrases-DirTrue* and *Paraphrases-DirFalse* subsets are concerned.

While we argue that fixed-prompt LM tuning models are not too sensitive to their specific prompts (Logan IV et al., 2021; Webson and Pavlick, 2022), we also supplement the experiment with continuous prompt models.

The prompts in S&S-continuous are sequences of trainable embeddings, which can be tuned for any purpose, not just to separate true and false entailments; moreover, as Table 6 shows, S&S-continuous and S&S-discrete have similar performance when tested on the same dataset as train set. So we use S&S-continuous as a control model, to understand how much the S&S-discrete is affected by prompt suitability.

In Table 7, we report results on the two same-label subsets with S&S-continuous in comparison to S&S-discrete. We find the results from the two models to be very similar. This means, as aligned with our argument above, the prompt-suitability issue is not jeopardising our experiment.

C LevyHolt Experiments on More LMs

In this section, we report more results from the LevyHolt experiments. In particular, we replicate the generalisation experiment and the symmetric-prompt experiment in §3.1 with RoBERTa-large, BERT-base and BERT-large models.

Directional-Symmetric generalization results for the three LMs are summarized in Table 8, 10 and 12 respectively, mirroring Table 1. The conclusions are consistent, where S&S classifiers trained on the

Test Train/Dev	Directional	Symmetric	Full
Directional	75.7	4.4	15.4
Symmetric	0.3	67.5	53.0
Full	30.1	69.5	62.4

Table 10: Generalization performance of **BERT-base** S&S classifier on the *DirTrue-DirFalse* (Directional) and *Paraphrases-Unrelated* (Symmetric) subsets of LevyHolt. Values are in % of AUC_{norm}. Rows are train/dev subsets, columns are test subsets.

AUC_{norm}	S&S	$\widehat{S\&S}$
Para-DirTrue	5.0	3.4 (-1.6%)
Para-DirFalse	17.8	11.7 (-6.1%)

Table 11: Comparison between **BERT-base** $S\&S$ classifiers (regular) and $\widehat{S\&S}$ (symmetric prompts). *Paraphrases-DirTrue* and *Paraphrases-DirFalse* subsets are concerned.

Test Train/Dev	Directional	Symmetric	Full
Directional	71.7	2.7	9.1
Symmetric	0.0	77.1	58.2
Full	49.4	80.9	75.4

Table 12: Generalization performance of **BERT-large** S&S classifier on the *DirTrue-DirFalse* (Directional) and *Paraphrases-Unrelated* (Symmetric) subsets of LevyHolt. Values are in % of AUC_{norm}. Rows are train/dev subsets, columns are test subsets.

AUC_{norm}	S&S	$\widehat{S\&S}$
Para-DirTrue	10.8	20.2 (+9.4%)
Para-DirFalse	54.9	37.4 (-17.5%)

Table 13: Comparison between **BERT-large** $S\&S$ classifiers (regular) and $\widehat{S\&S}$ (symmetric prompts). *Paraphrases-DirTrue* and *Paraphrases-DirFalse* subsets are concerned.

two subsets are still not generalising to each other.

The comparison with symmetric-prompt baselines for the above three LMs are in Table 9, 11 and 13. There again, scaling up the language model or switching from RoBERTa to BERT cannot lift the regular-prompt S&S model further away from the symmetric-prompt one. Particularly in Table 13, $\widehat{S\&S}_{Para-DirTrue}$ actually outperforms $S\&S_{Para-DirTrue}$, highlighting that performances of LMs are unstable on smaller fine-tune datasets, even with 100 sets of hyper-parameters for choice.

These observations echo those listed in §3, and are verify that our conclusions in §3 are generalisable at least to RoBERTa-large and BERT language models. We further argue that they are very likely generalisable to other language models following the same training scheme as well.

D   Detailed Guide for BoOQA Construction

This section is of interest to readers who wish to dig into the details of how BoOQA is built, or to build their own flavour of BoOQA dataset.

The BoOQA dataset is constructed off large-scale general-domain news corpora. For English, we use the NewsCrawl corpus, which contains 7.8M news articles published between 2007 and 2017; for Chinese, we use the CLUE corpus, which contains 2.4M news articles, all published in 2016. With corpora at such scale, they naturally have

overlaps with the corpora used for building models. Our introduced baselines include LM-driven methods and Entailment-Graph-based methods. The entailment graphs are induced from smaller corpora, which we can afford to exclude from NewsCrawl or CLUE; on the other hand, the LMs are pre-trained on corpora of much larger scale, where it is impractical to remove all overlaps between the corpora, so we refrain from such attempts.

We partition each corpus into context windows by disjoint 3-day time spans. We extract open relation triples (subject, predicate, object) from the articles with parsers (Reddy et al., 2014; Li et al., 2022). The use of automatic parsing-based relation extraction enables us to collect data entries at large scales. The reason for using these parsers instead of neural taggers such as OpenIE (Kolluru et al., 2020), is that they have wider coverage and are capable of handling discontinuous predicates, whereas OpenIE only extracts contiguous spans.

From each context window, we look for frequent argument-pairs mentioned in at least 15 different articles with at least 15 different predicates. These argument-pairs star in thoroughly-discussed events in this context window, so that valid propositions about these argument-pairs can be expected to be inferable from existing mentions.

On the other hand, in order to battle the noise introduced by the automatic parsers, and include only the propositions with felicitous predicates in our dataset, we introduce thresholds for predicates

to be taken into consideration. Each predicate has to be present with at least 30 different argument pairs for it to be considered felicitous; these can be anywhere in the corpus.

We select propositions between felicitous predicates and starring argument-pairs as the population of positives.[14] Then, we generate adversarial negative propositions from positive propositions.

Negative propositions need to be related to their corresponding positive propositions, but they also need to be false for the current context (i.e. with the current arguments, within the current context window). In practice, we consider a proposition to be false for the current context when it (and all its synonyms) is absent from the articles in the current context window with the current arguments. This is based on the assumption that a comprehensive pool of news articles would follow the Gricean cooperative principle of communication (Davis, 2019), that it would include all and only the known facts.

We generate negatives from WordNet hyponyms: hyponyms are naturally related to their corresponding hypernyms; moreover, they are more specific, so they are less likely to be true. The effect of this property is, when a candidate negative proposition generated from WordNet hyponyms is absent from the current context, it has an even smaller chance of actually being correct for the context. Namely, we are in smaller danger of false negatives.

When selecting negatives from WordNet hyponyms, we iterate over all spans-of-tokens in the seed positive proposition. We check if each span matches any entries in the WordNet; when a match is found, we scan through all its hyponyms, and replace the original span in the proposition with each hyponym, in order to create candidate negatives. For instance, for a positive proposition "A play game with B", we can replace the span "play" with its hyponym "foul" to create a candidate "A foul game with B"; we can also replace the span "game" with its hyponym "practice game" to create a candidate "A play practice game with B".

Each candidate negative has to satisfy two criteria to be really considered a negative: it also has to be felicitous, and it has to be absent from the current context. For felicitousness, we set the same threshold as for positives, that is, mentioned with at least 30 different argument pairs anywhere in the corpus. For absence from the current context, we check that not only the predicate in the candidate

proposition itself is absent from the current context window, but all its WordNet synonyms must be absent from the current context window as well.

Additionally, for $BoOQA_{EN}$, where we are able to make use of word-sense disambiguation tools (Yap et al., 2020), for each WordNet match, we select the synset that best suits the context; for $BoOQA_{ZH}$, we select the first synset, since it is the most common synset and has the best chance of being suitable for any given context.

With the positives selected from felicitous predicates between starring arguments, and the negatives generated from the WordNet hyponyms of positives, we have two populations of propositions. We finally create balanced samples from these populations as the resulting BoOQA dataset. We sample no more than two negatives for each positive; we dictate that positives and negatives must be sampled in bundles: no positives can be sampled without their corresponding negatives, and vice versa; we also sample from each context window proportional to its number of articles.

Further, we bring the frequency distributions of positives and negatives closer to each other with bucket sampling. We collect the ratio of predicates in the population of positive propositions falling into each of a pre-defined set of frequency buckets[15], and sample the negatives in each frequency bucket according to a quota proportional to the ratio of the positive population this bucket represents.

E Analysis and Discussions for BoOQA

E.1 Details of H-only Model in BoOQA Robustness Test

For testing the robustness of our BoOQA dataset, we re-use the H-only model in §3.2, with (XLM-)RoBERTa-base LMs. Again we use the word "true" or its Chinese equivalent "正确" to mask the premise slot in the prompts.

Unlike the LevyHolt H-only setting, in BoOQA, the negative hypotheses are not only not entailed by (any) premise, but also false statements in ground truth; on the other hand, the hypotheses in negative entries of LevyHolt can have any truthfulness in reality. For this reason, if we directly use the hypothesis propositions as training data, the model would be learning not just the artefacts, but also identifying the truthfulness of each statement using

---

[14]Population in the sense of statistical population.

[15]The set of bucket frequency boundaries is: (60, 100, 300, 500, 700, 1000, 1500, 2000, 2500, 3000, 4000, 5000, 6000, 8000, 10000, 15000, 20000, 30000, 50000, 100000)

Felicitous Propositions
Pos: Mark Zuckerburg says in Facebook
Neg: Mark Zuckerburg responds in Facebook
Infelicitous Propositions
Pos: shown at time for time
Neg: Police recites suspect

Table 14: Examples of felicitous / infelicitous propositions; "pos" denotes positives, "neg" negatives.

its memory from pre-training. To get more precise readings of artefact strengths, we mask the arguments in the propositions with their entity types in FIGER (Ling and Weld, 2012) ontology. This way, we balance between not letting the model know the truthfulness of each proposition and letting the model understand the context of each proposition.

BoOQA comes with only dev / test set. In order to get a train set, we follow Schmitt and Schütze (2021b) in sub-splitting the dev set into train / dev2 sets; further, we enforce that these train / dev2 sets are of similar sizes to those in LevyHolt.

In BoOQA dataset, dev / test sets are sampled from different time spans, and only used for testing. So it is harmless that a hypothesis appears in both dev and test sets, because their truthfulness could be different depending on their different contexts (premises). However, for H-only model, since we no longer use the premises, we do need to eliminate the overlap between train / dev / test splits. We check for overlaps of hypothesis propositions between the data splits and assign them randomly to one set, and remove them from the other. Since the same hypothesis can bear different truthfulness values in different context windows, we consider two propositions as overlaps as long as the propositions themselves are identical, regardless of their assigned truthfulness value.

E.2 Case Study

We conduct a case study of dataset quality. Since we expect the premises of the entailment relations to come from large pools of news articles, there are no straight-forward ways of evaluating the label correctness for each context; however, it is possible to evaluate the felicitousness of the hypotheses.

Similarly to above, we again mask the actual arguments with their FIGER (2012) types, so human judgements for felicitousness don't get polluted by prior expectations of correctness.

We draw a sample of 100 positive and 100 negatives from both $BoOQA_{EN}$ and $BoOQA_{ZH}$, and

manually label their felicitousness. We find the ratio of felicitous samples to be consistent: for English, 86/100 of positive hypotheses are indeed felicitous, 87/100 of negative hypotheses are indeed felicitous; for Chinese, the numbers are 87 for positives and 85 for negatives.

We list some examples of felicitous and infelicitous entries in Table 14. We find that the major source of infelicitousness for positive propositions is propagated error from relation extraction[16]; for negative propositions, the major source include the retrieved hyponym being semantically reasonable but pragmatically inappropriate, and the retrieved hyponym corresponding to an incorrect word sense for the respective positive.

E.3 How Directionality Is Involved in BoOQA

In this section, we explain how BoOQA tests for directionality. Suppose in BoOQA we have a positive proposition A and its corresponding negative proposition B, which has WordNet hyponyms replaced into one of its spans. At evaluation, we retrieve from the context window a third proposition C, which we know is true and is under the same context as A and B. We have two measures of entailments, one is *directional*, the other is *symmetric*; both are perfect in their own right.

A perfect *directional* measure is contemplated as a measure that would assign score 1 to all true entailments and 0 to all non-entailments; a perfect *symmetric* measure would assign score 1 to all paraphrastic pairs, 0.5 to all semi-paraphrastic pairs (directional true entailments or directional non-entailments), and 0 to all unrelated pairs. This is an oversimplified hypothetical situation, where we ignore the non-categorical feature of predicate entailments; we note that this is only for the benefit of theoretical discussions.

There are 8 possible scenarios according to the genuine correctness of $C \vDash A$, $A \vDash C$, $B \vDash C$. We only care about these three edge out of the 6, because: 1) whether A and B entail each other is unimportant for determining whether A or B are true given C; 2) it must be the case the $C \nvDash B$ for the entry to be valid, otherwise the false proposition B would be indorsed by the context.

The goal of a method is to rank the positive higher than the negative in each scenario; when both methods rank the positive and the negative

---

[16]since we use automatic open relation extraction methods to replace human-annotated predicates, in order for scaling up the number of entailments inspected in our dataset.
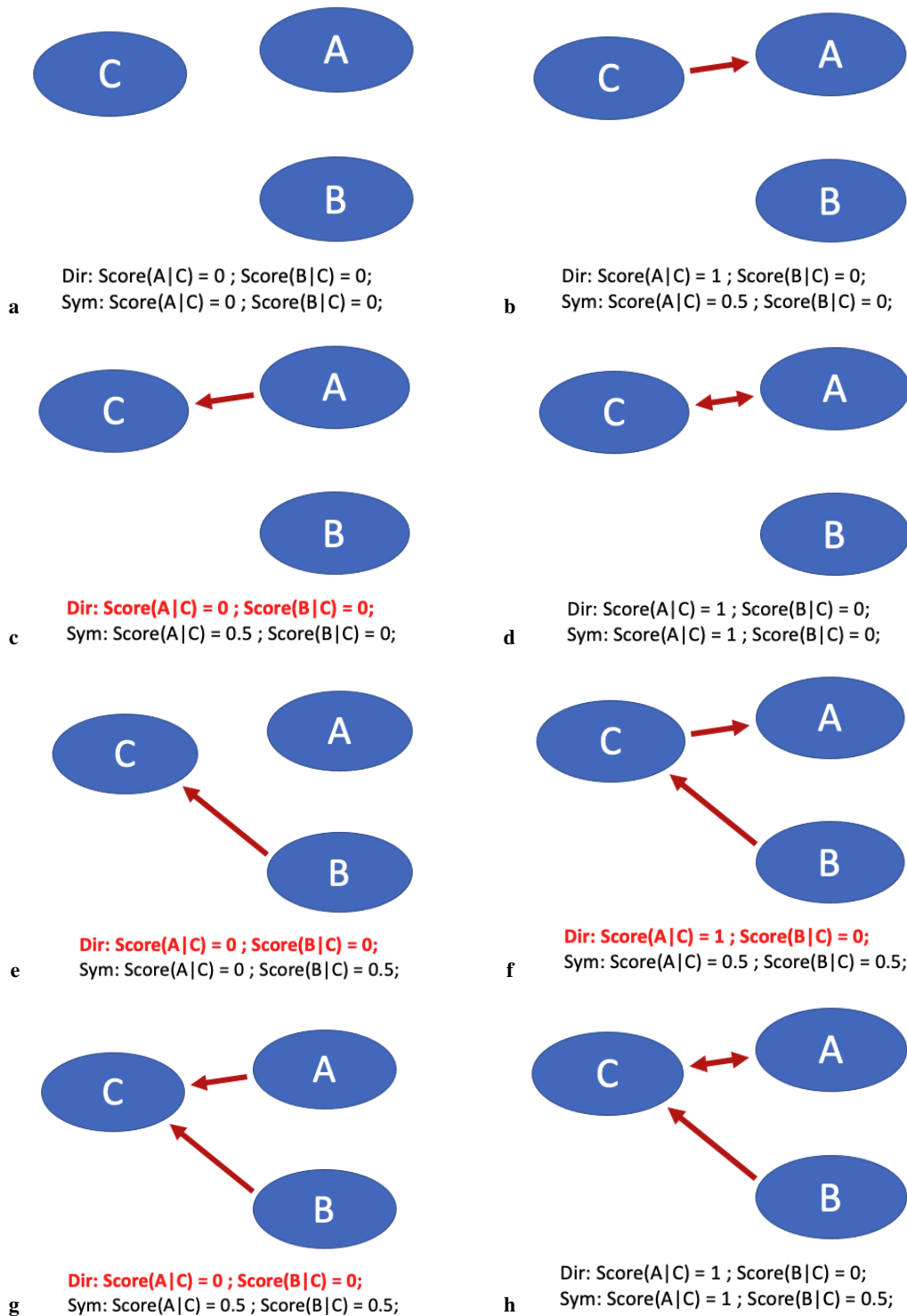
Figure 5: Diagram illustration of each of the 8 possible scenarios of ground truth entailment relations between (A, C) and (B, C). When two methods show disparity in performance, the winning method is coloured red.

equally, the one that assigns lower confidence scores to them is considered the winner. This is because, for each proposition, there are multiple context propositions to refer to. Therefore, when evidence from the current context is insufficient for a judgement, being conservative harms less.

We graphically analyse the 8 possible scenarios in Figure 5. In 4 of the 8 scenarios (scenario a, b, d, h), a perfect symmetric measure would do equally well as a perfect directional measure; however, in the other 4 scenarios (scenario c, e, f, g), a perfect directional measure would prevail.

This analysis illustrates how directional entailments are involved in our BoOQA evaluation. It

also illustrates how we manage to test for a broad set of directional entailments by building a dataset with only hyponyms / hypernyms. The relation between A and B is hyponym / hypernym, however the entailments that we really test for are between the third proposition C and each of A / B, which can be all kinds of entailments.

F Computational Infrastructure and Model Footprint

For our experiments, we use RTX 2080Ti GPUs with 11GB graphic memories. We set the computational limit for each job to be 168 GPU hours (equivalent to 7 days on 1 GPU). Our experiments are not CPU-intensive, but we note that for evaluation of entailment graphs on BoOQA, $\sim$ 120GB of memory is required for speedy processing with the context windows pre-loaded into memory.

S&S fine-tuning on full LevyHolt dataset and the *symmetric* subset typically take around 40 minutes to finish on these GPU; we run the fine-tuning with 100 different hyper-parameters for each experiment, so the total duration is approximately 60 GPU hours; S&S fine-tuning on smaller subset typically take around 15 minutes for each run, and around 25 GPU hours for each experiment.

For evaluation on the BoOQA dataset, unsupervised LMs take $\sim$ 10 GPU hours; S&S with base models take $\sim$ 40 GPU hours, S&S with large models take $\sim$ 100 GPU hours. Evaluation of entailment graphs does not require GPU, which typically take 40-50 CPU hours with $\sim$120GB memory.

The $S\&S$ models have a similar # of parameters to their underlying LMs, with only a linear layer added upon them; the rest of methods are either out-of-the-box LMs or discrete language resources.

G Details of BoOQA Baselines

In this section, we discuss some implementation details of BoOQA baselines, report AUC_{norm} values for baselines on BoOQA dev sets in Table 15, and discuss the presentation choices in Table 5; we also present more baselines on the BoOQA test set.

For each method, we allow up to 3200 pieces of context evidence for each hypothesis. The only exception is $S\&S_{full}$ for $BoOQA_{ZH}$, where the speed is so slow that we had to cap this threshold to 90, hence the asterisk in Table 5.

LM$_{unsupervised}$ Firstly is the context retrieval approaches for unsupervised LM methods. We experimented with three variants:

Dev	$BoOQA_{En}$	$BoOQA_{Zh}$
LM-base		
$BERT_{tfidf}$	0.00	12.3
$BERT_{sent}$	0.3	9.8
$BERT_{rel}$	15.3	31.2
$RoBERTa_{rel}$	5.0	9.1
$S\&S_{Full}$	25.7	21.5
LM-large		
$BERT_{rel}$	16.9	26.9
$S\&S_{Full}$	24.9	-
Entailment Graphs		
EG BInc	29.9	39.6
EG CNCE	33.6	-
EG EGT2	25.9	-

Table 15: Dev set results on English and Chinese BoOQA datasets for various baselines, values in % of AUC_{norm}. All methods are out-of-the-box with no tuning; for Chinese we do not train $S\&S$ model with XLM-RoBERTa-large due to computational limits.

Test	$BoOQA_{En}$	$BoOQA_{Zh}$
LM-base		
$BERT_{tfidf}$	0.01	11.5
$BERT_{sent}$	0.08	10.8
$BERT_{rel}$	15.9	30.6
$RoBERTa_{rel}$	4.8	8.5
$S\&S_{Full}$	25.6	23.1
LM-large		
$BERT_{rel}$	17.7	26.0
$S\&S_{Full}$	24.8	-

Table 16: More test set baseline results on English and Chinese BoOQA dataset, values in % of AUC_{norm}. All methods are out-of-the-box with no tuning.

- $BERT_{tfidf}$: for each hypothesis, retrieves the top 5 most relevant articles by TF-IDF measure following Chen et al. (2017);

- $BERT_{sent}$: for each hypothesis, retrieves the *host-sentences* of all relations involving the same arguments as the queried hypothesis;

- $BERT_{rel}$: for each hypothesis, retrieves the *relations* involving the same arguments as the queried hypothesis (in short sentences);

Among them, $BERT_{rel}$ outperforms the other two, likely because it uses more focused context input of concatenated relations rather than the more noisy raw news sentences.

Between BERT and RoBERTa LMs, Although RoBERTa is shown (Schmitt and Schütze, 2021b) to be good for prompting with LevyHolt, BERT has
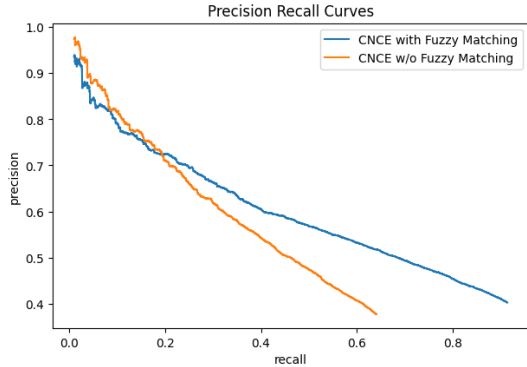
Figure 6: Precision-recall curves of CNCE entailment graphs on BoOQA dev set, w. / w.o. fuzzy matching.

a clear advantage for unsupervised cosine similarity. Between BERT-base and BERT-large, the more computationally expensive BERT-large model does not show a clear advantage, so we stick to the base version. Therefore, on test set we report $BERT_{rel}$ results with BERT-base as $LM_{unsupervised}$ in Table 5, and leave the rest to Table 16.

S&S Models For $S\&S$ methods we additionally report baselines trained on full LevyHolt with RoBERTa-large. Interestingly $S\&S_{Full}large$ does not outperform $S\&S_{Full}base$ on BoOQA, therefore in Table 5, we stick with RoBERTa-base.

EG Here for EG methods, we include a discussion on fuzzy matching and sparsity. Entailment graphs represent predicates in the form like "(go.1, go.to.2)", where the subject and object semantic roles are indicated by the two parts separated by a comma. In practice, multiple semantic role assignments can be given to one predicate in natural language form; so with fuzzy matching, we ground the natural language predicates in propositions to entailment graph nodes regardless of the semantic role assignments they may be associated to.

The fuzzy matching also brings the entailment graph methods to the same page as LM-driven methods: looking only at the natural language forms. As shown in Figure 6, without fuzzy matching the recall hit a wall at 64.1%, where sparsity is at full display. With fuzzy matching, much more entries are assigned some score, bringing the right boundary of recall to 91.4%; however, on one hand this harms the precision for top-notch edges, on the other hand the precision still drops rapidly with increasing recall, meaning that beyond the most confident entailment edges, the quality of entailment graph edges deteriorates quickly. That is to say, sparsity still hampers the performance of en-

Method	$AUC_{50\%}$
BInc (Szpektor and Dagan, 2008)	54.7
Weeds (Weeds and Weir, 2003)	55.2
CNCE (Hosseini et al., 2021)	55.9
EGT2 (Chen et al., 2022)	63.3

Table 17: Performance of various entailment graphs on the LevyHolt *directional* subset, values in %.

tailment graphs.

H   The Directional Subset with Chen et al. (2022)

Contemporary to us, Chen et al. (2022) presented a new entailment score for building entailment graphs, where they first fine-tune a DeBERTa model on MNLI, then re-fine-tune it with pairs of short propositions composed of (subject, predicate, object) triples, using PPDB annotations for supervision, in order for domain transfer. They reported results of their EGT2 graph on a version of *directional* subset of LevyHolt. However, we have found their *directional* subset to be different from that first presented by Holt (2019).

We re-evaluate the EGT2 graph on the Holt (2019) original version of *directional* subset and list the results in Table 17, which are presented in $AUC_{50\%}$. Consistent with the results reported in Chen et al. (2022), the EGT2 graph still outperforms the rest of the graphs, albeit not by as much. This introduces a slight inconsistency between results on the LevyHolt *directional* subset and the BoOQA dataset; we trust the larger BoOQA dataset and speculatively argue that the instability of the small LevyHolt *directional* test set and some transferrable spurious correlations in MNLI related to the behavior of human annotators may be to blame.

One last remark is, there is a slight difference in the calculation of AUC between Hosseini et al. (2018)/Chen et al. (2022), and Schmitt and Schütze (2021b), in terms of left boundaries of recall, where the calculations by Schmitt and Schütze (2021b) yield higher numbers. Since the calculation by Schmitt and Schütze (2021b) is compatible with all methods, we use the this calculation throughout.