Asking Crowdworkers to Write Entailment Examples: The Best of Bad Options

Clara Vania   **Ruijie Chen**   **Samuel R. Bowman**
New York University
{c.vania, rc3959, bowman}@nyu.edu

Abstract

Large-scale natural language inference (NLI) datasets such as SNLI or MNLI have been created by asking crowdworkers to read a *premise* and write three new *hypotheses*, one for each possible semantic relationships (*entailment*, *contradiction*, and *neutral*). While this protocol has been used to create useful benchmark data, it remains unclear whether the writing-based annotation protocol is optimal for any purpose, since it has not been evaluated directly. Furthermore, there is ample evidence that crowdworker writing can introduce artifacts in the data. We investigate two alternative protocols which automatically create candidate (*premise, hypothesis*) pairs for annotators to label. Using these protocols and a writing-based baseline, we collect several new English NLI datasets of over 3k examples each, each using a fixed amount of annotator time, but a varying number of examples to fit that time budget. Our experiments on NLI and transfer learning show negative results: None of the alternative protocols outperforms the baseline in evaluations of generalization within NLI or on transfer to outside target tasks. We conclude that crowdworker writing still the best known option for entailment data, highlighting the need for further data collection work to focus on improving *writing-based* annotation processes.

1 Introduction

Research on natural language understanding has benefited greatly from the availability of large-scale, annotated data, especially for tasks like reading comprehension and natural language inference, which lend themselves to non-expert crowdsourcing. These datasets are useful in three settings: evaluation (Williams et al., 2018; Rajpurkar et al., 2018; Zellers et al., 2019); pretraining (Phang et al., 2018; Conneau et al., 2018; Pruksachatkun et al.,

2020); and as training data for downstream tasks (Trivedi et al., 2019; Portelli et al., 2020).

Natural language inference (NLI), also known as *recognizing textual entailment* (RTE; Dagan et al., 2005) is the problem of determining whether or not a hypothesis semantically entails a premise. The two largest NLI corpora, SNLI (Bowman et al., 2015) and MNLI (Williams et al., 2018) are created by asking crowdworkers to write three labeled *hypothesis* sentences given a *premise* sentence taken from a preexisting text corpus. While these datasets have been widely used as benchmarks for NLU, there have been no studies evaluating writing-based annotation for collecting NLI data. Moreover, there is growing evidence that human writing can introduce *annotation artifacts*, which enable models to perform moderately well just by learning spurious statistical patterns in the data (Gururangan et al., 2018; Tsuchiya, 2018; Poliak et al., 2018a).

This paper explores the possibility of collecting high-quality NLI data without asking crowdworkers to write hypotheses. We introduce two alternative protocols (Figure 1) which substitute crowdworker writing with fully-automated pipelines to generate premise-hypothesis sentence pairs, which annotators then simply label. The first protocol uses a sentence-similarity-based method to pair similar sentences from large unannotated corpora. The second protocol uses parallel sentences and uses machine translation systems to generate sentence pairs. Using the MNLI protocol as our baseline, we collect five datasets using premises taken from Gigaword news text (Parker et al., 2011) and Wikipedia. We then compare models trained using these datasets for their generalization performance within NLI and for transfer learning to other tasks.

We start from the assumption that writing a new hypothesis takes more time and effort than simply labeling a presented hypothesis. As a result, it is plausible that our protocols could offer some value
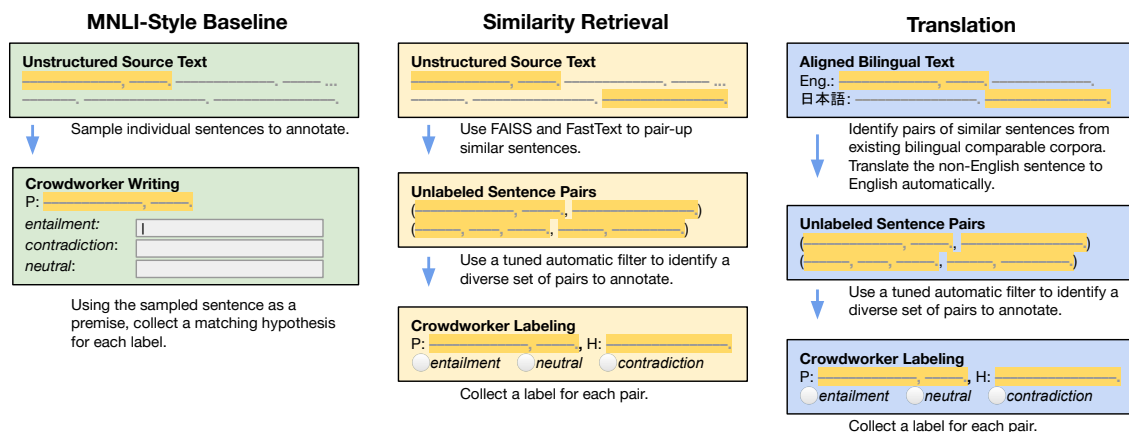
Figure 1: We introduce two new protocols for natural language inference data collection. Both use fully-automated pipelines to generate pairs of semantically-related sentences, which crowdworker annotators then label.

even if the quality of the data they produce is no better than a writing-based baseline. To study the cost trade-off, we collect each dataset under the same fixed annotation budget with a fixed ($\sim$ US $15) hourly wage. Using this constraint, we collect approximately twice as many examples from our new protocols.

Our main results on natural language inference and transfer learning are clearly negative. Human-constructed examples appear to be far superior to automatically-constructed examples in both settings. While crowdworker writing in data collection has known issues, it produces better training data than our automatic methods, or any known comparable methods which intervene the writing-based protocol to help crowdworkers with the writing process (Bowman et al., 2020). This strongly suggests that future work on data quality should focus on improving human-based generation processes.

2 Collecting NLI Data

We compare three protocols for collecting NLI data: (1) a baseline MNLI-style protocol (BASE), (2) a sentence-similarity-based protocol (SIM), and (3) a translation-based protocol (TRANSLATE). To test generalization performance across domains, we collect two datasets for BASE and SIM, using text from Gigaword (news) and Wikipedia (wiki) domains.[1] For TRANSLATE, we collect a dataset from WikiMatrix (Schwenk et al., 2019), a collection of Wikipedia parallel sentences. Table 1 shows examples of sentence pairs collected using

each protocol.

Our new protocols (Figure 1) share a similar automated pipeline. Given an unstructured text, we automatically collect similar sentence pairs which annotators then label. There are two key differences between our new protocols and BASE. First, our automatically paired sentences are *unlabeled*, and thus require a further data labeling process (Section 2.4). Second, our protocols might produce datasets with imbalanced label distributions. This is in contrast to BASE, which ensures each premise will have one hypothesis for each label in the annotation. The following subsections describe each protocol in more detail.

2.1 Baseline (BASE)

Our BASE protocol closely follows that used for MNLI. We randomly sample premise sentences from Gigaword and Wikipedia and ask crowdworkers to write three new hypotheses, one for each relation type.[2]

2.2 Sentence Similarity (SIM)

Our SIM protocol exploits the fact that, in large corpora, it should be easy to find pairs of sentences that describe similar events or situations. For example, in Gigaword, one event might be written differently by different news sources in ways that yield any of our three relationships. We collect similar sentences and automatically match them to form sentence pairs which annotators then label. The whole pipeline consists of three steps:

---

[1]The premise sentences for each protocol can be different although they come from the same source.

[2]Our instructions can be found in the Appendix A, and our FAQs are available at https://sites.google.com/nyu.edu/nlu-mturk-faq/writing-sentences.

Dataset	Label	Premise	Hypothesis
Base-News	E	The city reconsidered that position on Wednesday, saying it was seeking to raise an additional $1.5 million to extend Mardi Gras over two weekends and to pay for overtime on several days.	The city is looking to get more money for Mardi Gras.
Base-Wiki	C	Service books were not included and a note at the end mentions many other books in French, English and Latin which were then considered worthless.	Service books were included.
Sim-News	N	All of them run out like college football players before a big bowl game.	Pray before a college football game.
Sim-Wiki	C	His work was heavily criticised as unscientific by his contemporaries.	His work was recognized and admired by his contemporaries.
Translate-Wiki	E	This was used to indicate a positive response, or truth, or approval of the item in front of it.	This was used to indicate yes, true, or confirmed on items in a list.

Table 1: Examples of sentence pairs chosen randomly from each test set, along with their assigned labels. **E**: *entailment*, **C**: *contradiction*, **N**: *neutral*.

indexing and retrieval, reranking, and crowdworker labeling.

Indexing and Retrieval   Given a raw text, we first split it into sentences.[3] We encode each sentence as a 300-dimensional vector using fastText (Bojanowski et al., 2017) and index them using FAISS (Johnson et al., 2019), an open-source library for large-scale similarity search on vectors.[4] Since Gigaword and Wikipedia consist of billions of sentences, we perform dimensionality reduction using PCA and cluster the search space to allow efficient index and retrieval. We randomly sample query sentences from the text corpus and retrieve the top 1k most similar sentences for each query. This is done by building an index with type `"PCAR64,IVFx,Flat"` in FAISS terms, where `x` varies depending on the corpus size. Details of our indexing and retrieval procedures can be found in Appendix A.1.

Reranking   FastText uses a Continuous Bag-of-Words (CBoW) model to learn word representations. This means given a query, we will sometimes have top matches which are syntactically similar but describe different events or situations. While unrelated sentences can be contradictory or neutral, directly using the top-n sentences from FAISS will give us too few entailment pairs. Furthermore, because we use randomly sampled sentences as queries, there could be no good match at all for a given query.

To collect a set of sentence pairs with a reasonable label distribution, for each query, we retrieve top-K matches and rerank the (query, retrieved sentence) pairs using the following features:

- *FAISS similarity score*: The raw similarity score from FAISS.

- *Word types*: The proportion of word types in the query sentence seen in the retrieved sentence.

- *Noun phrase*: The proportion of noun phrases in the query sentence seen in the retrieved sentence.

- *Subjects*: The proportion of complete subject spans (some sentences with embedded clauses can have more than one subject) in the query sentence seen in the retrieved sentence.

- *Named entity*: The proportion of named entities in the query sentence seen in the retrieved sentence.

- *Time*: A boolean feature which denotes whether two sentences are written in the same month and year (only for Gigaword)

- *Wiki article*: A boolean feature which denotes whether the pairs come from the same article. (only for Wikipedia)

- *Wiki link*: The proportion of hyperlink tokens in the query sentence seen in the retrieved sentence (only for Wikipedia)

The choice of these hand-crafted features will likely impact the distribution of our final dataset, but we

---

[3] We use Spacy's `"en_core_web_lg"` model to segment sentences and extract noun phrase and entities for later use in reranking.

[4] https://github.com/facebookresearch/faiss

don't expect these choices to inject significant label-association artifacts, since our methods play no role in setting labels. We calculate the score for each sentence pair using a weighted sum of these features. We populate pairs from all queries and sort them based on their feature scores. We then select the top $N\%$ pairs as our final pairs.

We use a Bayesian hyperparameter optimization to tune the feature weights, K, and N. In an ideal case, we want our dataset to have a balanced distribution so that all classes will be represented equally. To push for this, we tune these parameters to minimize the Kullback–Leibler (KL) divergence between a uniform distribution across three entailment classes, $P(x)$, and an empirical distribution, $Q(x)$, computed based on the predictions of an NLI model. We run Bayesian optimization for 100 iterations using Optuna (Akiba et al., 2019). For the NLI model, we use a RoBERTa$_{\text{Large}}$ model fine-tuned on a combination of SNLI, MNLI, and ANLI.

2.3 Translation (TRANSLATE)

Multilingual comparable corpora contain *similar* texts in at least two different languages. If they are sentence-aligned, we can automatically translate text from one language to one of the others to yield candidate sentence pairs. Since the alignment behind the corpus can be noisy, the resulting sentence pairs range almost continuously from being parallel to being semantically unrelated, potentially fitting any of the three entailment relationships. In the TRANSLATE protocol, we investigate whether we can use such sentence pairs as entailment data.

We use WikiMatrix (Schwenk et al., 2019), a collection of 135 million Wikipedia parallel sentences, which was constructed by aligning similar sentences in different languages in a joint sentence embedding space (Schwenk, 2018; Artetxe and Schwenk, 2019). It is a mix of translated sentence pairs and comparable sentences written independently about the same information. We collect parallel sentences where one of the sentences is in English, s^E. For the paired non-English languages, we pick 5 languages: German, French, Indonesian, Japanese, and Czech. We then translate the aligned non-English sentence into an English sentence, $s^{\hat{E}}$ using the OPUS-MT (Tiedemann and Thottingal, 2020) machine translation systems, and treat $(s^E, s^{\hat{E}})$ as a sentence pair. The diverse set of languages allows us to collect a more diverse set

	Individual == Gold	No Gold Label
MNLI (Full)	88.7%	1.8%
Base-News	78.7%	13.1%
Base-Wiki	76.4%	10.0%
Sim-News	72.9%	15.8%
Sim-Wiki	74.1%	11.9%
Translate-Wiki	72.8%	14.6%

Table 2: Validation statistics for each protocol, compared to MNLI Full.

of sentence pairs coming from the structural differences across languages. We do not perform any reranking as our predictions using an NLI model on the initially retrieved data (the same one that we used in §2.2) shows a near-balanced distribution.

2.4 Data Labeling

We use Amazon Mechanical Turk to label the automatically-collected sentence pairs (SIM and TRANSLATE). We hire crowdworkers which have completed at least 5000 HITs with at least a 99% acceptance rate. In each task, we present crowdworkers with a sentence pair and ask them to provide a single label (*entailment*, *contradiction*, *neutral* or *"I don't understand"*) for the pair. The latter is used if there are problems with either sentence, e.g., because of errors during preprocessing. We collect one label per sentence pair. We use the same HIT setup for validating our test sets (Section 3).

3 The Resulting Datasets

Using BASE, we collect 3k examples for Base-News and Base-Wiki.[5] For SIM and TRANSLATE, we increase the number of pairs to exhaust the same budget that was used for the corresponding baseline dataset ($1,791 for Base-News and $1,445 for Base-Wiki), allowing us to collect around twice as many examples for each protocol.[6]

For each dataset, we randomly select 250 sentence pairs as the test set and use the rest as the training set. To ensure accurate labeling, we perform an additional round of annotation on the test sets. We ask four crowdworkers to label each pair using the same instructions that we use for data labeling, giving us a total of 5 annotations per example. We assign the majority vote as the gold

---

[5] Our preliminary experiments on subsets of MNLI show that RoBERTa performance starts to stabilize once we use at least 3k training examples.

[6] The resulting datasets are available at https://github.com/nyu-mll/semi-automatic-nli. We provide anonymized worker-ids.

675

		#Pairs	Label Distribution			HL_E		HL_C		HL_N		Word Type Overlap		
			E	C	N	μ	(σ)	μ	(σ)	μ	(σ)	E	C	N
Training	MNLI-3k	2750	33.4	33.9	32.7	9.7	4.4	9.4	4.0	11.0	4.4	25.2	17.3	15.4
	Base-News	2734	33.5	33.4	33.2	12.1	6.0	11.8	5.8	12.4	6.2	23.5	18.4	18.1
	Base-Wiki	2740	33.3	33.7	33.0	11.1	7.7	10.5	4.5	11.6	7.1	31.2	23.4	22.7
	Sim-News	6627	21.8	39.1	39.2	23.2	9.7	22.7	10.0	23.3	9.9	46.6	21.8	23.0
	Sim-Wiki	6174	23.5	40.4	36.1	12.8	6.0	12.7	5.2	13.1	5.3	52.7	31.7	29.7
	Translate-Wiki	6189	34.7	31.4	34.0	18.6	9.6	14.2	7.5	16.0	8.8	41.3	20.0	24.6
Test	MNLI-3k	250	29.2	37.6	33.2	10.6	4.6	9.4	3.7	10.7	4.2	26.3	14.6	15.9
	Base-News	226	38.1	33.2	28.8	12.8	5.7	11.5	5.1	11.6	4.6	22.8	14.4	13.5
	Base-Wiki	234	32.5	32.1	35.5	12.5	8.6	11.7	8.2	11.5	4.8	32.9	24.6	21.1
	Sim-News	219	20.1	44.3	35.6	22.5	11.1	24.9	11.1	23.9	10.9	69.3	20.9	20.6
	Sim-Wiki	229	20.5	45.0	34.5	12.6	7.6	13.7	5.8	12.0	4.5	60.5	32.8	28.7
	Translate-Wiki	222	40.5	29.3	30.2	18.7	8.5	13.0	6.9	14.3	6.7	46.3	15.1	21.1

Table 3: Dataset statistics. **HL** denotes the *average* and *standard deviation* of the hypothesis length of each label.

label.

Table 2 shows the agreement statistics for each protocol. BASE shows a higher agreement than SIM and TRANSLATE, although it is lower than MNLI. Compared to MNLI, all of our datasets show higher number of examples with no gold label (no consensus between annotators). As we strictly follow the MNLI protocol for BASE, this suggests that the different population of crowdworkers is likely responsible for these differences.[7]

3.1 Dataset Statistics

Table 3 shows the statistics of our collected data. As anticipated, datasets collected using SIM and TRANSLATE have slightly unbalanced distributions compared to BASE. In particular, for SIM, we observe that the entailment class has the lowest distribution in the training and test data.

One clear difference between BASE and our new protocols is the hypothesis length. SIM and TRANSLATE tend to create longer hypothesis than BASE. We suspect that this is an artifact of the sentence-similarity method, which prefers *identical* sentences (both syntax and semantics) over semantically *similar* sentences. Across domains, we observe that sentences from news texts are longer than Wikipedia.

Recent work by McCoy et al. (2019) shows that popular NLI models might learn a simple lexical overlap heuristic for predicting entailment labels. While this heuristic is natural for entailment, it can affect the model's generalization especially when it is strongly reflected in the data. We calculate word type overlap by using the intersection of premise

and hypothesis word types, divided by the union of the two sets. The last three columns in Table 3 reports word type overlap in each dataset for each entailment label. We find that word type overlap is a *much* stronger predictor of the label in our new protocols than in BASE. This could be a significant driver of our results and might hurt the generalization performance of models trained using our new protocols' data.

3.2 Annotation Cost

We use the FairWork platform to set payment for each of our HITs (Whiting et al., 2019). Fair-Work surveys workers to estimate the time that each HIT takes and adjusts pay to a target of US $15/hr. Based on its estimation, we pay $0.4 and $0.3 for each written hypothesis of Base-News and Base-Wiki, respectively. For Sim-News, Sim-Wiki, and Translate-Wiki, we pay $0.175, $0.15, $0.15 for each labeled sentence pair, respectively. In total, we spend $1791 for each dataset collected from Gigaword and $1445 for each dataset collected from Wikipedia.

4 Experiments

We aim to test whether our alternative protocols can produce high-quality data that yield models that generalize well within NLI and in transfer learning. For the NLI evaluation, we evaluate each model on nine test sets: (i) the five new individual test sets, each containing ~250 examples; (ii) the MNLI *development* set; and (iii) the three *development* sets of Adversarial NLI (ANLI; Nie et al., 2020), collected from three rounds of annotation (A1, A2, A3). ANLI is collected using an iterative adversarial approach that follows MNLI but encourages

---

[7]MNLI used an organized group of crowdworkers hired through Hybrid (gethybrid.io).

		Test Data									
	Training Data	BN	BW	SN	SW	TW	MNLI	A1	A2	A3	Avg.
CBoW	Base-News	33.4	37.8	32.4	30.1	35.8	35.6	32.8	32.8	**33.4**	34.0
	Base-Wiki	34.1	33.1	37.9	35.4	39.0	35.6	33.1	31.6	33.2	34.8
	Sim-News	35.4	35.9	32.0	32.3	37.8	35.8	33.1	32.8	**33.4**	34.3
	Sim-Wiki	32.3	37.2	52.1	49.1	44.6	36.6	33.1	32.4	32.1	38.8
	Translate-Wiki	37.4	39.3	35.4	35.8	45.5	35.4	33.0	**32.9**	32.8	36.4
RoBERTa	MNLI-3k	79.0	61.3	**76.7**	57.5	58.1	**83.9**	33.4	27.0	28.7	56.2
	Base-News	**79.4**	**76.1**	57.5	61.6	58.1	83.1	**35.8**	29.5	28.0	**56.6**
	Base-Wiki	77.0	74.2	58.5	62.0	61.3	54.0	30.9	31.8	33.1	53.6
	Sim-News	53.3	56.0	65.8	59.8	66.2	79.5	**35.8**	30.2	28.2	52.8
	Sim-Wiki	62.0	62.8	64.8	**64.9**	**69.1**	64.7	32.2	32.0	31.5	53.8
	Translate-Wiki	48.5	54.9	60.7	58.1	67.1	50.9	32.5	32.7	33.2	48.7
	Average per test set	52.0	51.7	52.2	49.7	53.0	54.1	33.2	31.4	31.6	45.4

Table 4: Model performance on individual test sets, as a median over 10 random restarts. **BN**: Base-News, **BW**: Base-Wiki, SN: **Sim-News**, **SW**: Sim-Wiki, **TW**: Translate-Wiki. The last row shows the average performance across models on each test set.

	Test Data									
Training Data	BN	BW	SN	SW	TW	MNLI	A1	A2	A3	Avg.
MNLI-3k	46.5	**50.4**	33.3	38.4	36.2	**52.8**	33.3	33.1	33.0	**39.7**
Base-News	**47.8**	46.6	33.8	33.6	**37.4**	51.5	**32.5**	33.3	33.1	38.8
Base-Wiki	33.2	32.1	**44.3**	**45.0**	29.3	32.8	33.3	**33.3**	33.0	35.1
Sim-News	33.2	35.5	38.8	38.9	29.3	32.8	33.3	**33.3**	33.5	34.3
Sim-Wiki	33.2	30.8	**44.3**	44.6	28.8	32.8	33.3	**33.3**	33.0	34.9
Translate-Wiki	31.4	34.6	34.3	34.5	32.4	33.6	33.3	**33.3**	33.5	33.4
Average per test set	37.5	38.3	38.1	39.2	32.2	39.4	33.2	33.3	33.2	36.0

Table 5: RoBERTa performance on individual test sets for *hypothesis-only* models.

crowdworkers to write sentences that are difficult for a trained NLI model.

We experiment with two sentence encoders: a CBoW baseline initialized with fastText embeddings (Bojanowski et al., 2017), and a more powerful RoBERTa$_{\text{Large}}$ (Liu et al., 2019) model, fine-tuned on individual training sets. We perform a hyperparameter sweep, varying the learning rate $\in \{1e-3, 1e-4, 1e-5\}$ and the dropout rate $\in \{0.1, 0.2\}$. We use batch size of 16 and 4 for CBoW and RoBERTA, respectively. We train each model using the best hyperparameters for 10 epochs, with 10 random restarts. In initial experiments, we find that this setup yields sTable performance given our relatively small datasets, especially when using RoBERTa.[8]

For transfer learning, we test whether each dataset can improve downstream task performance when it is used as intermediate-task data (Phang et al., 2018; Pruksachatkun et al., 2020). As our col-

lected datasets are fairly small (< 10K examples), we use five data-poor downstream target tasks in the SuperGLUE benchmark (Wang et al., 2019a): **COPA** (Roemmele et al., 2011); **WSC** (Levesque et al., 2012); **RTE** (Dagan et al., 2005, et seq), **WiC** (Pilehvar and Camacho-Collados, 2019); and **MultiRC** (Khashabi et al., 2018). We experiment with the BERT$_{\text{Large}}$ (Devlin et al., 2019) and RoBERTa$_{\text{Large}}$ models. We follow Pruksachatkun et al. (2020) for training hyperparameters. We use the Adam optimizer (Kingma and Ba, 2015).

We run experiments using the `jiant` toolkit (Wang et al., 2019b), which is the recommended baseline package for SuperGLUE, and is based on Pytorch (Paszke et al., 2019), HuggingFace Transformers (Wolf et al., 2020), and AllenNLP (Gardner et al., 2017).

4.1 NLI Experiments

Table 4 reports the model performance on individual test sets. We include a baseline training data, a 3k randomly sampled training examples from MNLI (MNLI-3k). We observe that all the

---

[8] This is consistent with the recent findings of Zhang et al. (2020) and Mosbach et al. (2020) regarding fine-tuning BERT-style models on small data.

Intermediate training data	COPA acc.	MultiRC F1$_\alpha$	RTE acc.	WiC acc.	WSC acc.	Avg.
BERT						
None	70.0	70.9	73.3	72.7	62.5	69.9
MNLI-3k	+0.0	-0.1	**+4.0**	-0.8	-2.9	+0.0
Base-News	**+1.0**	-0.5	**+4.3**	-1.7	**+1.0**	**+0.8**
Base-Wiki	**+2.0**	**+0.3**	**+3.2**	-1.2	-1.0	**+0.7**
Sim-News	**+3.0**	-0.3	**+2.2**	-2.3	**+0.0**	**+0.5**
Sim-Wiki	**+7.0**	-0.2	**+4.0**	-2.6	-3.8	**+0.9**
Translate-Wiki	**+4.0**	**+0.1**	**+2.5**	-3.7	0.0	**+0.6**
RoBERTa						
None	88.0	77.0	85.2	71.9	67.3	77.9
MNLI-3k	-4.0	-0.1	+0.7	+0.2	-3.8	-1.5
Base-News	**+1.0**	**+0.4**	**+1.1**	**+0.7**	-1.9	**+0.3**
Base-Wiki	-2.0	-1.2	**+1.1**	+0.5	-1.0	-0.5
Sim-News	-6.0	-3.6	-6.1	-0.1	-3.8	-3.9
Sim-Wiki	-5.0	-1.9	-2.2	-1.2	-16.3	-5.3
Translate-Wiki	-5.0	-2.7	-2.5	-1.8	-6.7	-3.7

Table 6: Results on using each collected dataset as intermediate training data on five SuperGLUE tasks. We report the median performance over 3 random restarts on the intermediate NLI models. *None* denotes experiments without intermediate-task training, i.e., direct fine-tuning on target tasks. The last column shows the average score across the five tasks. We report the difference with respect to *None* using BERT and RoBERTa.

CBoW baselines obtain near chance performance. Using RoBERTa, the top performing models are all trained on datasets collected using BASE: Base-News and MNLI-3k. We find that models trained using Translate-Wiki obtain the worst performance. On average across all training sets, ANLI development sets seem to be the hardest, while MNLI seems to be the easiest.

Unsurprisingly, we do not find a single training set which yields the best model across all test sets. We observe that models trained on Base-News perform the best for Base-News and Base-Wiki test sets. Similarly, Sim-Wiki performs the best on both Sim-Wiki and Sim-News test sets. We find that all models do poorly on all ANLI development sets.

Overall, we find that Base-News outperforms all other datasets. However, it is also better than SIM and TRANSLATE which suggests that our new protocols failed. The lower accuracy for SIM and TRANSLATE on their respective test sets also suggests that they produce datasets with noisier labels.

4.2 Hypothesis-Only Results

Next, we experiment with a hypothesis-only model (Poliak et al., 2018b) to investigate spurious statistical patterns in the hypotheses which might signal the actual labels to the model. Table 5 reports the results for all five datasets and MNLI. On the five new test sets, we observe that MNLI and Base-News are the most solvable by the hypothesis-only models, though their numbers are still much lower than with SNLI with accuracy 69.17.

On average across all test sets, none of the training sets obtain much higher performance than chance. All models achieve chance performance on ANLI. However, all of our training sets are fairly small, and these numbers might not be very informative. This also explains why these numbers are relatively lower than other NLI datasets (Poliak et al., 2018b). Across all training sets, we again see that the MNLI test set is the most solvable by the hypothesis-only models.

Our new protocols show lower performance than the BASE, but that may just be because they are of lower overall quality and not because they are less solvable by the hypothesis-only models. We verify this by looking at their transfer learning performance in the following section.

4.3 Transfer Learning

Table 6 shows our results when using each collected data as intermediate-training data on the five target tasks. We report the median performance of three random restarts on the validation sets. Using BERT, we observe that all our new datasets yield models with better performance than plain BERT or MNLI-3k as intermediate-training data. We see less positive transfer when we use RoBERTa.

If we look at individual target task performance, both Base-News and Base-Wiki data give consistent positive transfer for RTE, a natural language inference task. We also see some positive trans-

	Entailment		Contradiction		Neutral	
M-3k	looked	0.44	no	1.03	also	0.75
	capital	0.43	never	0.95	because	0.71
	population	0.43	any	0.88	better	0.63
B-News	according	0.58	never	1.07	also	0.62
	position	0.45	no	1.02	many	0.52
	set	0.42	any	0.90	most	0.52
B-Wiki	both	0.45	never	1.18	most	0.78
	named	0.38	not	1.01	well	0.64
	early	0.35	any	0.96	many	0.56
S-Giga	summit	0.53	points	0.66	very	0.54
	roads	0.51	we	0.65	research	0.48
	weighted	0.46	–	0.59	weeks	0.48
S-Wiki	division	0.56	census	0.88	through	0.57
	team	0.48	population	0.86	such	0.54
	candidate	0.47	2010	0.82	number	0.49
T-Wiki	;	0.68	brought	0.45	each	0.57
	album	0.58	maintain	0.40	{	0.56
	f	0.55	will	0.39	}	0.56

Table 7: Top three words most associated with each label by PMI. **M**: MNLI, **B**: Base, **S**: Sim, **T**: Translate.

fer for COPA, however since its validation set is very small (100 examples), we can not conclude anything with confidence.

Overall, our BASE shows better transfer learning performance compared to MNLI, suggesting that our setup is sound. However, we also see that our new protocols perform worse than BASE, showing that they produce less useful training data than the strong baseline of crowdworker writing.

5 Dataset Analysis

5.1 Annotation Artifacts

Following Gururangan et al. (2018), we compute the PMI between each hypothesis word and label in the training set to examine whether certain words have high associations with its inference label. For a fair comparison, we only use ~3k training examples from each dataset, and sub-sample data collected using SIM and TRANSLATE.

Table 7 shows the top three most associated words for each label, sorted by their PMI scores. We find that BASE has similar associations to MNLI, especially for the neutral and contradiction labels where we found many negations and adverbs. We observe that both SIM and TRANS-LATE are less susceptible to this artifact. However, this might be a side-effect of high word overlap in the data, which prefers similar words in the premise and hypothesis. This is also a well-known artifact for NLI data (McCoy et al., 2019).

5.2 Qualitative Analysis

Our new protocols use a vector-distance based measurement to find similar sentences, and we find that many of the sentence pairs share similar syntactic structure in their premise and hypothesis, even when both describe different events or entities. We also find that hypothesis in several Sim-News examples differs by only a few words with its premise. For Translate-Wiki, we observe some effects of *translation divergence*, where the translation of the sentence changes semantically because of cross-linguistic distinctions between languages. We provide some examples of these observations in Table 8.

6 Related Work

There is a large body of work on constructing data for natural language inference. The first test suite for entailment problems, FraCas (Consortium et al., 1996), is a very small set created manually by experts to isolate phenomena of interest. The RTE challenge corpora (Dagan et al., 2005, et seq) were built by asking human annotators to judge whether a text entails a hypothesis. The SICK dataset (Marelli et al., 2014) is constructed by mining existing paraphrase sentence pairs from image and video captions, which annotators then label.

Some recent works also use automatic methods for generating sentence pairs for entailment data. Zhang et al. (2017) propose a framework to generate hypotheses based on context from general world knowledge or neural sequence-to-sequence methods. The DNC corpus (Poliak et al., 2018a) is an NLI dataset with ordinal judgments constructed by recasting several NLP datasets to NLI examples and labeling them using custom automatic procedures. QA-NLI (Demszky et al., 2018) is an NLI dataset derived from existing QA datasets. Similar to ours, both DNC and QA-NLI use automatic methods to generate sentence pairs. However, neither of them explicitly evaluates whether machine-generated pairs are better than human-generated pairs.

Bowman et al. (2020) propose four potential modifications to the SNLI/MNLI protocol, all still involving crowdworker writing, and show that none yields improvements in the resulting data. SWAG (Zellers et al., 2018) and HellaSwag (Zellers et al., 2019) construct sentence pairs from specific data sources and use language models to generate challenging negative examples.

Type	Dataset	Premise	Hypothesis	Label
Syntactic structure	Sim-News	**For many people**, choosing wallpaper is one of decorating's more stressful experiences, fraught with anxiety over color, pattern and cost.	**For many people**, anxiety about decorating stems from not understanding the language of furniture, fabrics and decorative styles.	E
	Sim-Wiki	**Its flowers are** pale yellow **to** white **and** spherical.	**Its flowers are** funnel-shaped **and** pink **to** white.	C
	Translate-Wiki	But now, in **the early 1990s**, the Jakarta-Begor railway had turned into a double rail.	However, by **the early 1990s**, McCreery's position within the UDA became less secure.	N
Lexical overlap	Sim-News	**GrandMet owns Burger King, the world's second-biggest hamburger chain, as well as US** frozen foods manufacturer **Pillsbury, which produces the luxury ice-cream Haagen-Daazs.**	**GrandMet owns Burger King, the world's second-biggest hamburger chain, as well as US** food group **Pillsbury, which produces the luxury ice-cream Haagen-Daazs.**	E
Translation divergence	Translate-Wiki	Marcus Claudius then abducted **her** while **she** was on **her** way to school.	Marcus Claudius then kidnapped **him** while **he** was on **his** way to school.	N

Table 8: Dataset observations from our new protocols.

On the topic of cost-effective crowdsourcing, Gao et al. (2015) develop a method to reduce redundant translations when collecting human translated data. When the annotation budget is fixed, Khetan et al. (2018) suggest that it is better to label collect single label per training example as many as possible, rather than collecting less training examples with multiple labels.

7 Conclusion

In this paper, we introduce two data collection protocols which use fully-automatic pipelines to collect hypotheses, replacing crowdworker writing in the MNLI baseline protocol. We find that switching to a writing-free process with the same source data and annotator pool yields poor-quality data. Our main experiments show strong negative results both in NLI generalization and transfer learning, and mixed results on annotation artifacts, suggesting that MNLI-style crowdworker writing examples are broadly better than automatically paired ones. This finding dovetails with that of Bowman et al. (2020), who find that they are unable to improve upon a base MNLI-style prompt when introducing aids meant to improve annotator speed or creativity. Future work along this line might focus on crowdsourcing strategies (beyond the basic HIT design) which encourage crowdworkers to produce high-quality data with reduced artifacts.

While our fully-automatic methods to construct sentence pairs yield negative results, we have not exhausted all possible automatic techniques for collecting similar sentences. However, given that we use state-of-the-art tools including FAISS, RoBERTa, and OPUS, and refine our methods with several rounds of piloting and tuning, we are skeptical that there is low-hanging fruit in the two directions we explored. A more radically different direction might involve generating pairs from scratch, using a large language model like GPT-3 (Brown et al., 2020). However, this would still require training data from crowdworker-written dataset, and might add a major source of potentially difficult-to-diagnose bias.

Finally, despite its known issues, we find that MNLI-style data is still the most effective for both NLI evaluation and transfer learning, and future efforts to create similar data should work from that starting point.

References

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A Next-generation Hyperparameter Optimization Framework. In *Proceedings of the 25rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Mikel Artetxe and Holger Schwenk. 2019. Margin-based parallel corpus mining with multilingual sentence embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3197–3203, Florence, Italy. Association for Computational Linguistics.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

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.

Samuel R. Bowman, Jennimaria Palomaki, Livio Baldini Soares, and Emily Pitler. 2020. Collecting Entailment Data for Pretraining: New Protocols and Negative Results.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared 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 M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam Mc-Candlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485, Brussels, Belgium. Association for Computational Linguistics.

The Fracas Consortium, Robin Cooper, Dick Crouch, Jan Van Eijck, Chris Fox, Josef Van Genabith, Jan Jaspars, Hans Kamp, David Milward, Manfred Pinkal, Massimo Poesio, Steve Pulman, Ted Briscoe, Holger Maier, and Karsten Konrad. 1996. Using the Framework.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. In *Machine Learning Challenges Workshop*, pages 177–190. Springer.

Dorottya Demszky, Kelvin Guu, and Percy Liang. 2018. Transforming Question Answering Datasets Into Natural Language Inference Datasets.

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.

Mingkun Gao, Wei Xu, and Chris Callison-Burch. 2015. Cost optimization in crowdsourcing translation: Low cost translations made even cheaper. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 705–713, Denver, Colorado. Association for Computational Linguistics.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke S. Zettlemoyer. 2017. AllenNLP: A Deep Semantic Natural Language Processing Platform. Unpublished manuscript available on arXiv.

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.

J. Johnson, M. Douze, and H. Jégou. 2019. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*.

Daniel Khashabi, Snigdha Chaturvedi, Michael Roth, Shyam Upadhyay, and Dan Roth. 2018. Looking Beyond the Surface: A Challenge Set for Reading Comprehension over Multiple Sentences. 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 252–262, New Orleans, Louisiana. Association for Computational Linguistics.

Ashish Khetan, Zachary C. Lipton, and Anima Anandkumar. 2018. Learning From Noisy Singly-labeled Data. In *International Conference on Learning Representations*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations,*

ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings.

Hector J. Levesque, Ernest Davis, and Leora Morgenstern. 2012. The Winograd Schema Challenge. In *Proceedings of the Thirteenth International Conference on Principles of Knowledge Representation and Reasoning*, KR'12, pages 552–561. AAAI Press.

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.

Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, and Roberto Zamparelli. 2014. A SICK cure for the evaluation of compositional distributional semantic models. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*, pages 216–223, Reykjavik, Iceland. European Languages Resources Association (ELRA).

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.

Marius Mosbach, Maksym Andriushchenko, and Dietrich Klakow. 2020. On the Stability of Fine-tuning BERT: Misconceptions, Explanations, and Strong Baselines.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, Online. Association for Computational Linguistics.

Robert Parker, David Graff, Junbo Kong, Ke Chen, and Kazuaki Maeda. 2011. English Gigaword Fifth Edition LDC2011T07.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Jason Phang, Thibault Févry, and Samuel R. Bowman. 2018. Sentence Encoders on STILTs: Supplementary Training on Intermediate Labeled-data Tasks.

Mohammad Taher Pilehvar and Jose Camacho-Collados. 2019. WiC: the Word-in-Context Dataset for Evaluating Context-Sensitive Meaning Representations. 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 1267–1273, Minneapolis, Minnesota. Association for Computational Linguistics.

Adam Poliak, Aparajita Haldar, Rachel Rudinger, J. Edward Hu, Ellie Pavlick, Aaron Steven White, and Benjamin Van Durme. 2018a. Collecting diverse natural language inference problems for sentence representation evaluation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 67–81, Brussels, Belgium. Association for Computational Linguistics.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018b. 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.

Beatrice Portelli, Jason Zhao, Tal Schuster, Giuseppe Serra, and Enrico Santus. 2020. Distilling the Evidence to Augment Fact Verification Models. In *Proceedings of the Third Workshop on Fact Extraction and VERification (FEVER)*, pages 47–51, Online. Association for Computational Linguistics.

Yada Pruksachatkun, Jason Phang, Haokun Liu, Phu Mon Htut, Xiaoyi Zhang, Richard Yuanzhe Pang, Clara Vania, Katharina Kann, and Samuel R. Bowman. 2020. Intermediate-task transfer learning with pretrained language models: When and why does it work? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5231–5247, Online. Association for Computational Linguistics.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S Gordon. 2011. Choice of Plausible Alternatives: An evaluation of commonsense causal reasoning. In *2011 AAAI Spring Symposium Series*.

Holger Schwenk. 2018. Filtering and mining parallel data in a joint multilingual space. In *Proceedings of the 56th Annual Meeting of the Association*

for Computational Linguistics (Volume 2: Short Papers), pages 228–234, Melbourne, Australia. Association for Computational Linguistics.

Holger Schwenk, Vishrav Chaudhary, Shuo Sun, Hongyu Gong, and Francisco Guzmán. 2019. WikiMatrix: Mining 135M Parallel Sentences in 1620 Language Pairs from Wikipedia. *CoRR*, abs/1907.05791.

Jörg Tiedemann and Santhosh Thottingal. 2020. OPUS-MT — Building open translation services for the World. In *Proceedings of the 22nd Annual Conferenec of the European Association for Machine Translation (EAMT)*, Lisbon, Portugal.

Harsh Trivedi, Heeyoung Kwon, Tushar Khot, Ashish Sabharwal, and Niranjan Balasubramanian. 2019. Repurposing entailment for multi-hop question answering tasks. 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 2948–2958, Minneapolis, Minnesota. Association for Computational Linguistics.

Masatoshi Tsuchiya. 2018. Performance impact caused by hidden bias of training data for recognizing textual entailment. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, Miyazaki, Japan. European Languages Resources Association (ELRA).

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019a. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems*.

Alex Wang, Ian F. Tenney, Yada Pruksachatkun, Phil Yeres, Jason Phang, Haokun Liu, Phu Mon Htut, Katherin Yu, Jan Hula, Patrick Xia, Raghu Pappagari, Shuning Jin, R. Thomas McCoy, Roma Patel, Yinghui Huang, Edouard Grave, Najoung Kim, Thibault Févry, Berlin Chen, Nikita Nangia, Anhad Mohananey, Katharina Kann, Shikha Bordia, Nicolas Patry, David Benton, Ellie Pavlick, and Samuel R. Bowman. 2019b. `jiant` 1.3: A software toolkit for research on general-purpose text understanding models. http://jiant.info/.

Mark E Whiting, Grant Hugh, and Michael S Bernstein. 2019. Fair Work: Crowd Work Minimum Wage with One Line of Code. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, pages 197–206.

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.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Huggingface's transformers: State-of-the-art natural language processing.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. SWAG: A large-scale adversarial dataset for grounded commonsense inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 93–104, Brussels, Belgium. Association for Computational Linguistics.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy. Association for Computational Linguistics.

Sheng Zhang, Rachel Rudinger, Kevin Duh, and Benjamin Van Durme. 2017. Ordinal common-sense inference. *Transactions of the Association for Computational Linguistics*, 5:379–395.

Tianyi Zhang, Felix Wu, Arzoo Katiyar, Kilian Q. Weinberger, and Yoav Artzi. 2020. Revisiting Few-sample BERT Fine-tuning.

A Appendices

A.1 Indexing and Retrieval

Gigaword The corpus contains texts from seven news sources: afp_eng, apw_eng, cna_eng, ltw_eng, nyt_eng, wpb_eng, and xin_eng. We build one index for each news source with type "`PCAR64,IVFx,Flat`", where `x` defines the number of clusters in the index. This type of index allows faster retrieval, however it requires a training stage to assign a centroid to each cluster. We refer readers to FAISS documentation for more detail explanations.[9]

For each news source, we randomly sample 100 sentences from its monthly articles and use them as seed sentences to train the clusters. We then set the number of clusters `x` to $\frac{N}{100}$ (rounded to the nearest hundred), where N is the number of seed sentences. Table 9 lists the number of seed sentences and clusters used for each news source index.

Source	#seed sentences	x
afp_eng	111,147	1,100
apw_eng	146,119	1,400
cna_eng	125,508	1,200
ltw_eng	90,195	900
nyt_eng	136,827	1,300
wpb_eng	9,144	100
xin_eng	157,760	1,500

Table 9: Number of seed sentences and number of clusters for each news source index.

During retrieval, for each query, we retrieve top 1000 sentences from each index and perform reranking on the combined list, i.e., 7,000 sentence pairs, as described in Section 2.2.

Wikipedia We build one index for the whole Wikipedia corpus. For seed sentences, we use sentences taken from the first paragraph of each article as it usually contains the summary of the article. We set the number of clusters `x` to 15,000.

---

[9]https://github.com/facebookresearch/faiss

A.2 Writing HIT Instructions

Instructions

The New York University Center for Data Science is collecting your answers for use in research on computer understanding of English. Thank you for your help!

This task will involve reading a prompt and writing three sentences that relate to it. The prompt will describe a situation or event. Using only this description and what you know about the world, please:

- Write one sentence that is **definitely correct** about the situation or event in the prompt.
- Write one sentence that **maybe correct** about the situation or event in the prompt.
- Write one sentence that is **definitely incorrect** about the situation or event in the prompt.

These prompts were taken from news and Wikipedia articles. If you recognize an individual, place, or event, please do not use outside knowledge about it to write your sentences. Please write sentences based only on the prompt and general beliefs or assumptions about what happens in the world.

If you have more questions, please consult our FAQ.

Prompt:
"Security and reliability are two important aspects of this service because of the sensitivity and urgency of the data sent over."

Definitely correct
Example: For the prompt *"The cottages near the shoreline, styled like plantation homes with large covered porches, are luxurious within; some come with private hot tubs."*, you could write *"The shoreline has plantation style homes near it, which are luxurious and often have covered porches or hot tubs."*

Maybe correct
Example: For the prompt *"Government Executive magazine annually presents Government Technology Leadership Awards to recognize federal agencies and state governments for their excellent performance with information technology programs."*, you could write *"In addition to their annual Government Technology Leadership Award, Government Executive magazine also presents a cash prize for best dressed agent from a federal agency."*

Definitely incorrect
Example: For the prompt *"Yes, he's still under arrest, which is why USAT's front-page refer headline British Court Frees Chile's Pinochet is a bit off."*, you could write *"The headline `British Court Frees Chile's Pinochet` is correct, since the man is freely roaming the streets."*

Problems (optional) *If something is wrong with the prompt that makes it difficult to understand, let us know here.*

Figure 2: Writing HIT instructions.

A.3 Data Labeling and Validation HIT Instructions

Instructions

The New York University Center for Data Science is collecting your answers for use in research on computer understanding of English. Thank you for your help!

Your job is to figure out, based on a correct prompt (**S1**), if another prompt (**S2**) is also correct:
- Choose **definitely correct** if any event or situation that can be described by S1 would also fit S2.
  Example:
  S1: *"A kitten with spots is playing with yarn."*
  S2: *"A cat is playing"*
- Choose **maybe correct** if S2 could describe an event or situation that fit S1, but could also describe sentences that don't fit S1.
  Example:
  S1: *"A kitten with spots is playing with yarn."*
  S2: *"A kitten is playing with yarn on a sofa"*
- Choose **definitely incorrect** if any event or situation that could possibly described with S1 would not fit S2.
  Example:
  S1: *"A kitten with spots is playing with yarn."*
  S2: *"A puppy is playing with yarn."*
- Choose **I don't understand** if there is something wrong with the prompts that make them hard to understand (more than just a typo).
  Example:
  S1: *"A kitten with spots is playing with yarn."*
  S2: *"Marie talks (Oct. 26 - Nov."*

More questions? Visit our FAQ.

Question 1
S1: ${premise}
S2: ${hypothesis}

   O definitely correct   O maybe correct   O definitely incorrect   O I don't understand

Question 2
S1: ${premise}
S2: ${hypothesis}

   O definitely correct   O maybe correct   O definitely incorrect   O I don't understand

...

Question 5
S1: ${premise}
S2: ${hypothesis}

   O definitely correct   O maybe correct   O definitely incorrect   O I don't understand

---

Figure 3: Data Labeling and Validation HIT instructions. We collect one annotation per example for data labeling and five annotations per example for validation.