Mining Knowledge for Natural Language Inference from Wikipedia Categories

Mingda Chen[3*] **Zewei Chu**[2*] **Karl Stratos**[1] **Kevin Gimpel**[3]
[1]Rutgers University, NJ, USA
[2]University of Chicago, IL, USA
[3]Toyota Technological Institute at Chicago, IL, USA
{mchen,kgimpel}@ttic.edu,zeweichu@gmail.com,stratos@cs.rutgers.edu

Abstract

Accurate lexical entailment (LE) and natural language inference (NLI) often require large quantities of costly annotations. To alleviate the need for labeled data, we introduce WIKINLI: a resource for improving model performance on NLI and LE tasks. It contains 428,899 pairs of phrases constructed from naturally annotated category hierarchies in Wikipedia. We show that we can improve strong baselines such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) by pretraining them on WIKINLI and transferring the models on downstream tasks. We conduct systematic comparisons with phrases extracted from other knowledge bases such as WordNet and Wikidata to find that pretraining on WIKINLI gives the best performance. In addition, we construct WIKINLI in other languages, and show that pretraining on them improves performance on NLI tasks of corresponding languages.[1]

1 Introduction

Natural language inference (NLI) is the task of classifying the relationship, such as entailment or contradiction, between sentences. It has been found useful in downstream tasks, such as summarization (Mehdad et al., 2013) and long-form text generation (Holtzman et al., 2018). NLI involves rich natural language understanding capabilities, many of which relate to world knowledge. To acquire such knowledge, researchers have found benefit from external knowledge bases like WordNet (Fellbaum, 1998), FrameNet (Baker, 2014), Wikidata (Vrandečić and Krötzsch, 2014), and large-scale human-annotated datasets (Bowman et al., 2015; Williams et al., 2018; Nie et al., 2020). Creating these resources generally requires expensive human annotation. In this work, we are interested in automatically generating a large-scale dataset from Wikipedia categories that can improve performance on both NLI and lexical entailment (LE) tasks.

One key component of NLI tasks is recognizing lexical and phrasal hypernym relationships. For example, vehicle is a hypernym of car. In this paper, we take advantage of the naturally-annotated Wikipedia category graph, where we observe that most of the parent-child category pairs are entailment relationships, i.e., a child category entails a parent category. Compared to WordNet and Wikidata, the Wikipedia category graph has more fine-grained connections, which could be helpful for training models. Inspired by this observation, we construct WIKINLI, a dataset for training NLI models constructed automatically from the Wikipedia category graph, by automatic filtering from the Wikipedia category graph. The dataset has 428,899 pairs of phrases and contains three categories that correspond to the entailment and neutral relationships in NLI datasets.

To empirically demonstrate the usefulness of WIKINLI, we pretrain BERT and RoBERTa on WIKINLI, WordNet, and Wikidata, before finetuning on various LE and NLI tasks. Our experimental results show that WIKINLI gives the best performance averaging over 8 tasks for both BERT and RoBERTa.

We perform an in-depth analysis of approaches to handling the Wikipedia category graph and the effects of pretraining with WIKINLI and other data sources under different configurations. We find that WIKINLI brings consistent improvements in a low resource NLI setting where there are limited amounts of training data, and the improvements plateau as the number of training instances increases; more WIKINLI instances for pretraining are beneficial for downstream finetuning tasks

[*]Equal contribution. Listed in alphabetical order.
[1]Code and data are available at https://github.com/ZeweiChu/WikiNLI.

with pretraining on a fourway variant of WIKINLI showing more significant gains for the task requiring higher-level conceptual knowledge; WIKINLI also introduces additional knowledge related to lexical relations benefiting finer-grained LE and NLI tasks.

We also construct WIKINLI in other languages and benchmark several resources on XNLI (Conneau et al., 2018), showing that WIKINLI benefits performance on NLI tasks in the corresponding languages.

2 Related Work

We build on a rich body of literature on leveraging specialized resources (such as knowledge bases) to enhance model performance. These works either (1) pretrain the model on datasets extracted from such resources, or (2) use the resources directly by changing the model itself.

The first approach aims to improve performance at test time by designing useful signals for pretraining, for instance using hyperlinks (Logeswaran et al., 2019; Chen et al., 2019a) or document structure in Wikipedia (Chen et al., 2019b), knowledge bases (Logan et al., 2019), and discourse markers (Nie et al., 2019). Here, we focus on using category hierarchies in Wikipedia. There are some previous works that also use category relations derived from knowledge bases (Shwartz et al., 2016; Riedel et al., 2013), but they are used in a particular form of distant supervision in which they are matched with an additional corpus to create noisy labels. In contrast, we use the category relations directly without requiring such additional steps. Onoe and Durrett (2020) use the direct parent categories of hyperlinks for training entity linking systems.

Within this first approach, there have been many efforts aimed at harvesting inference rules from raw text (Lin and Pantel, 2001; Szpektor et al., 2004; Bhagat et al., 2007; Szpektor and Dagan, 2008; Yates and Etzioni, 2009; Bansal et al., 2014; Berant et al., 2015; Hosseini et al., 2018). Since WIKINLI uses category pairs in which one is a hyponym of the other, it is more closely related to work in extracting hyponym-hypernym pairs from text (Hearst, 1992; Snow et al., 2005, 2006; Pasca and Durme, 2007; McNamee et al., 2008; Le et al., 2019). Pavlick et al. (2015) automatically generate a large-scale phrase pair dataset with several relationships by training classifiers on a relatively small amount of human-annotated data. However,

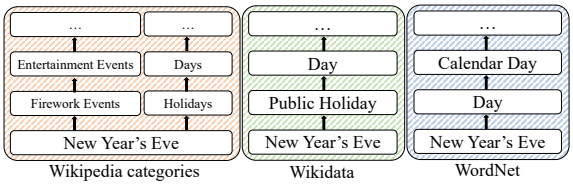

Figure 1: Example hierarchies obtained from Wikipedia categories, Wikidata, and WordNet.

most of this prior work uses raw text or raw text combined with either annotated data or curated resources like WordNet. WIKINLI, on the other hand, seeks a middle road, striving to find large-scale, naturally-annotated data that can improve performance on NLI tasks.

The second approach aims to enable the model to leverage knowledge resources during prediction, for instance by computing attention weights over lexical relations in WordNet (Chen et al., 2018) or linking to reference entities in knowledge bases within the transformer block (Peters et al., 2019). While effective, this approach requires nontrivial and domain-specific modifications of the model itself. In contrast, we develop a simple pretraining method to leverage knowledge bases that can likewise improve the performance of already strong baselines such as BERT without requiring such complex model modifications.

There are some additional related works that focus on the category information of Wikipedia. Ponzetto and Strube (2007) and Nastase and Strube (2008) extract knowledge of entities from the Wikipedia category graphs using predefined rules. Nastase et al. (2010) build a dataset based on Wikipedia article or category titles as well as the relations between categories and pages ("WikiNet"), but they do not empirically validate the usefulness of the dataset. In a similarly non-empirical vein, Zesch and Gurevych (2007) analyze the differences between the graphs from WordNet and the ones from Wikipedia categories. Instead, we address the empirical benefits of leveraging the category information in the modern setting of pretrained text representations.

3 WIKINLI

We now describe how the WIKINLI dataset is constructed from Wikipedia and its principal characteristics. Each Wikipedia article is associated with crowd-sourced categories that correspond to topics or concepts covered by that article. Wikipedia or-

ganizes these categories into a directed graph that models their hierarchical relations. For instance, the category "Days" is a parent node of the category "Holidays" in this graph. The central observation underlying WIKINLI is that this category hierarchy resembles the concept hierarchies and ontologies found in knowledge bases, such as Wikidata and WordNet.

While there are similarities between the three resources, the Wikipedia category hierarchy contains more diverse connections between parent and child concepts. Figure 1 shows an example category "New Year's Eve" and its ancestors under these resources. All resources include a path that corresponds to the generalization of New Year's Eve as a regular day, but Wikipedia additionally includes a path that corresponds to the generalization as a celebration or entertainment event. Thus the Wikipedia hierarchy provides more abstract and fine-grained generalization that can be useful for NLI tasks. In this example, the commonsense knowledge that New Year's Eve implies entertainment is only directly captured by the Wikipedia hierarchy.

WIKINLI is a dataset of category pairs extracted from this Wikipedia hierarchy to be used as an auxiliary task for pretraining NLI models. Specifically, WIKINLI contains three types of category pairs based on their relations in the Wikipedia hierarchy: child-parent ("child"), parent-child ("parent"), and other pairs ("neutral"). The motivation is that child-parent resembles entailment; parent-child resembles reverse entailment; and other pairs resemble a neutral relationship. We find that this simple definition of relations is effective in practice; we also report an exploration with other types of relations such as siblings in experiments.

Table 1 shows examples from WIKINLI that illustrate the diverse set of relations they address. They include conventional knowledge base entries such as "Bone fractures" being a type of "Injuries" and "Chemical accident" being a type of "Pollution". They also include relations that are more fine-grained than those typically found in knowledge bases. For instance, "Pakistan" is a child of "South Asian countries"; in contrast, it is a child of "Country" in Wikidata. WIKINLI also includes a large set of hyponym-hypernym relations in pairs that differ by only one or two words (e.g., "Cantonese music" and "Cantonese culture"); their coverage is extensive and includes relations involving rare words such as "Early Turkish Anatolia" and "Medieval Anatolia".

Category 1	Category 2	Rel.
Injuries	Bone fractures	P
Chemical accident	Pollution	C
Armenian sportspeople	Curaçao male actors	N
Argentine design	Nigerian inventions	N
Cantonese music	Cantonese culture	C
Medieval Anatolia	Early Turkish Anatolia	P
Learned societies	Academic organizations	C
South Asian countries	Pakistan	P

Table 1: Examples from WIKINLI. C = child; P = parent; N = neutral.

More details of constructing WIKINLI are as follows. We use the tables "categorylinks" and "page": these two pages provide category pairs in which one category is the parent of the other. We use all direct category relations. To eliminate trivial pairs, we remove pairs where either is a substring of the other. To construct neutral pairs, we randomly sample two categories where neither category is the ancestor of the other in the category graph. To make neutral pairs more "related" (so that they are harder to discriminate from direct relations), we encode both categories into continuous vectors using ELMo (Peters et al., 2018) (averaging its three layers over all positions) and compute the cosine similarities between pairs.[2] We pick the top-ranked pairs as neutral pairs in WIKINLI. After the above processing, we remove categories longer than 50 characters and those containing certain keywords[3] (see supplementary material for more results and examples on filtering criteria). We ensure the dataset is balanced, and the final dataset has 428,899 unique pairs.

For the following experiments, unless otherwise specified, we only use 100,000 samples from WIKINLI as training data and 5,000 as the development set since we find larger training set does not lead to performance gains (see Sec. 6.3 for more details). WIKINLI is available at https://github.com/ZeweiChu/WikiNLI.

4 Approach

To demonstrate the effectiveness of WIKINLI, we pretrain BERT and RoBERTa on WIKINLI and other resources, and then finetune them on several NLI and LE tasks. We assume that if a pretraining

[2]We choose ELMo over BERT-like models because in our experiments, ELMo is better off-the-shelf without fine-tuning.

[3]all digits, ., !, ?, of, at, in, by, from, to, about, stubs, lists.

dataset	#train	#dev	#test	#train per cat.
Natural Language Inference				
MNLI	3,000	9,815	9,796	1,000
SciTail	3,000	1,304	2,126	1,500
RTE	2,490	277	3,000	1,245
PPDB	13,904	4,633	4,641	1,545
Break	-	-	8,193	-
Lexical Entailment				
K2010	739	82	621	370
B2012	791	87	536	396
T2014	539	59	507	270

Table 2: Dataset statistics.

resource is better aligned with downstream tasks, it will lead to better downstream performance of the models pretrained on it.

4.1 Training

Following Devlin et al. (2019) and Liu et al. (2019), we use the concatenation of two texts as the input to BERT and RoBERTa. Specifically, for a pair of input texts x_1, x_2, the input would be $[\text{CLS}]x_1[\text{SEP}]x_2[\text{SEP}]$. We use the encoded representations at the position of $[\text{CLS}]$ as the input to a two-layer classifier, and finetune the entire model.

We start with a pretrained BERT-large or RoBERTa-large model and further pretrain it on different pretraining resources. After that, we finetune the model on the training sets for the downstream tasks, as we will elaborate on below.

4.2 Evaluation

We use several NLI and LE datasets. Statistics for these datasets are shown in Table 2 and details are provided below.

4.2.1 Natural Language Inference

MNLI. The Multi-Genre Natural Language Inference (MNLI; Williams et al., 2018) dataset is a human-annotated multi-domain NLI dataset. MNLI has three categories: entailment, contradiction, and neutral. Since the training split for this dataset has a large number of instances, models trained on it are capable of picking up information needed regardless of the quality of the pretraining resources we compare, which makes the effects of pretraining resources negligible. To better compare pretraining resources, we simulate a low-resource scenario by randomly sampling 3,000 instances[4] from the original training split as our new training set, but use the standard "matched" development and testing splits.

SciTail. SciTail is created from science questions and the corresponding answer candidates, and premises from relevant web sentences retrieved from a large corpus (Khot et al., 2018). SciTail has two categories: entailment and neutral. Similar to MNLI, we randomly sample 3,000 instances from the training split as our training set.

RTE. We evaluate models on the GLUE (Wang et al., 2019) version of the recognizing textual entailment (RTE) dataset (Dagan et al., 2006; Bar Haim et al., 2006; Giampiccolo et al., 2007; Bentivogli et al., 2009). RTE is a binary task, focusing on identifying if a pair of input sentences has the entailment relation.

PPDB. We use the human-annotated phrase pair dataset from Pavlick et al. (2015), which has 9 text pair relationship labels. The labels are: hyponym, hypernym, synonym, antonym, alternation, other-related, NA, independent, and none. We directly use phrases in PPDB to form input data pairs. We include this dataset for more fine-grained evaluation. Since there is no standard development or testing set for this dataset, we randomly sample 60%/20%/20% as our train/dev/test sets.

Break. Glockner et al. (2018) constructed a challenging NLI dataset called "Break" using external knowledge bases such as WordNet. Since sentence pairs in the dataset only differ by one or two words, similar to a pair of adversarial examples, it has broken many NLI systems.

Due to the fact that Break does not have a training split, we use the aforementioned subsampled MNLI training set as a training set for this dataset. We select the best performing model on the development set of MNLI and evaluate it on Break.

4.2.2 Lexical Entailment

We use the lexical splits for 3 datasets from Levy et al. (2015), including K2010 (Kotlerman et al., 2009), B2012 (Baroni et al., 2012), and T2014 (Turney and Mohammad, 2015). These datasets all similarly formulate lexical entailment as a binary task, and they were constructed from diverse sources, including human annotations, WordNet, and Wikidata.

[4]The number of training instances is chosen based on the number of instances per category, as shown in the last column of Table 2, where we want the number to be close to 1-1.5K.

	Natural Language Inference					Lexical Entailment			
	MNLI	RTE	PPDB	Break	SciTail	K2010	B2012	T2014	avg.
BERT	75.0	69.9	66.7	80.2	92.3	85.2	79.4	63.3	76.5
+WordNet	75.8	71.3	71.1	83.5	90.8	83.5	94.3	71.2	80.2
+Wikidata	75.7	71.3	**75.0**	81.3	91.5	82.3	95.3	70.5	80.4
+WIKINLI	76.4	70.9	70.7	**85.7**	91.8	84.9	**96.1**	71.2	81.0
RoBERTa	82.5	78.8	65.9	81.3	93.6	85.3	65.9	66.8	77.5
+WordNet	83.8	82.2	72.0	82.3	**93.9**	82.5	88.6	70.7	82.0
+Wikidata	84.0	82.3	72.5	83.2	92.9	82.4	94.8	71.0	82.9
+WIKINLI	**84.4**	**83.1**	71.7	83.8	93.0	**85.4**	95.7	**72.9**	**83.8**

Table 3: Test set performance for baselines and models pretrained on various resources. We report accuracy (%) for NLI tasks and F_1 score (%) for LE tasks. The highest results for each model (BERT or RoBERTa) are underlined. The highest numbers in each column are boldfaced.

5 Experiments

5.1 Baselines

We consider three baselines for both BERT and RoBERTa, namely the original model, the model pretrained on WordNet, and the model pretrained on Wikidata.

WordNet. WordNet is a widely-used lexical knowledge base, where words or phrases are connected by several lexical relations. We consider direct hyponym-hypernym relations available from WordNet, resulting in 74,645 pairs.

Wikidata. Wikidata is a database that stores items and relations between these items. Unlike WordNet, Wikidata consists of items beyond word types and commonly seen phrases, offering more diverse domains similar to WIKINLI. The available conceptual relations in Wikidata are: "subclass of" and "instance of". In this work, we consider the "subclass of" relation in Wikidata because (1) it is the most similar relation to category hierarchies from Wikipedia; (2) the relation "instance of" typically involves more detailed information, which is found less useful empirically (see the supplementary material for details). The filtered data has 2,871,194 pairs.

We create training sets from both WordNet and Wikidata following the same procedures used to create WIKINLI. All three datasets are constructed from their corresponding parent-child relationship pairs. Neutral pairs are first randomly sampled from non-ancestor-descendant relationships and top ranked pairs according to cosine similarities of ELMo embeddings are kept. We also ensure these datasets are balanced among the three classes.

5.2 Setup

For all the experiments, we used the Hugging Face implementation (Wolf et al., 2019). When finetuning or pretraining BERT-large models, we mostly follow the hyperparameters suggested by Devlin et al. (2019). Specifically, during pretraining, we use a batch size of 32, a learning rate of 2e-5, a maximum sequence length of 40, and 3 training epochs. During finetuning, we switch to use 8 as batch size due to memory constraints. When funetuning or pretraining RoBERTa-large, we did extra hyperparameter search by adopting some of the hyperparameters recommended from Liu et al. (2019). We use 10% training steps for learning rate warmup, 1e-5 for learning rate, and a maximum sequence length of 40, and train models for 3 epochs.[5]

For both models, we use development sets for model selection during pretraining. During downstream evaluations, we use a maximum sequence length of 128 for datasets involving sentences. We perform early stopping based on task-specific development sets and report the test results for the best models. Due to the variance of performance of 24-layer transformer architectures, we report medians of 5 runs with a fixed set of random seeds for all of our experiments. See the supplementary material for details on the runtime, hyperparameters, etc.

5.3 Results

The results are summarized in Table 3. In general, pretraining on WIKINLI, Wikidata, or WordNet improves the performances on downstream tasks, and pretraining on WIKINLI achieves the best per-

[5]We choose this set of hyperparameters due to computational constraints. Our finetuned RoBERTa achieves 82.3% accuracy on the RTE development set, which is lower than the 86.6% accuracy reported in Liu et al. (2019).

WIKINLI	Wikidata	WordNet
albums	protein	genus
songs	gene	dicot
players	putative	family
male	protein-coding	unit
people	conserved	fish
American	hypothetical	tree
British	languages	bird
writers	disease	person
(band)	RNA	fern

Table 4: 10 words from the top 20 most frequent words in WIKINLI, Wikidata, and WordNet. The full list is in the supplementary material.

	MNLI	RTE	PPDB	Break	avg.
Threeway	**75.6**	**74.4**	**71.2**	85.7	**76.7**
Fourway	**75.6**	74.0	69.8	**86.9**	76.6
Binary (C vs. R)	75.1	72.6	70.5	81.7	75.0
Binary (C/P vs. R)	74.3	72.2	69.8	80.5	74.3

Table 5: Comparing binary, threeway, and fourway classification for pretraining. C = child; P = parent; R = rest. The highest numbers in each column are boldfaced.

formance on average. Especially for Break and MNLI, WIKINLI can lead to much more substantial gains than the other two resources. Although BERT-large + WIKINLI is not better than the baseline BERT-large on RTE, RoBERTa + WIKINLI shows much better performance. Only on PPDB, Wikidata is consistently better than WIKINLI. We note that BERT-large + WIKINLI still shows a sizeable improvement over the BERT-large baseline. More importantly, the improvements to both BERT and RoBERTa brought by WIKINLI show that the benefit of the WIKINLI dataset can generalize to different models. We also note that pretraining on these resources has little benefit for SciTail.

6 Analysis

We perform several kinds of analysis, including using BERT to compare the effects of different settings. Due to the submission constraints of the GLUE leaderboard, we will report dev set results (medians of 5 runs) for the tables in this section, except for Break which is only a test set.

6.1 Lexical Analysis

To qualitatively investigate the differences between WIKINLI, Wikidata, and WordNet, we list the top 20 most frequent words in these three resources in Table 4. Interestingly, WordNet contains mostly abstract words, such as "unit", "family", and "person", while Wikidata contains many domain-specific words, such as "protein" and "gene". In contrast, WIKINLI strikes a middle ground, covering topics broader than those in Wikidata but less generic than those in WordNet.

	MNLI	RTE	PPDB	Break	avg.
Threeway 100k	75.6	74.4	**71.2**	85.7	76.7
Fourway 100k	75.6	74.0	69.8	86.9	76.6
Threeway 400k	**75.7**	**75.5**	70.9	83.0	76.3
Fourway 400k	75.6	75.1	70.8	**89.5**	**77.8**

Table 6: The effect of the number of WIKINLI pretraining instances. The highest numbers in each column are boldfaced.

	MNLI	RTE	PPDB	Break	avg.
① 100k	75.6	74.4	71.2	85.7	76.7
① 50k	74.9	74.7	70.8	76.9	74.3
① 50k + ② 50k	75.0	71.5	70.9	80.2	74.4
① 50k + ③ 50k	75.0	73.6	70.7	81.5	75.3

Table 7: Combining WIKINLI with other datasets for pretraining. ①=WIKINLI;②=WordNet;③=Wikidata.

6.2 Fourway vs. Threeway vs. Binary Pretraining

We investigate the effects of the number of categories for WIKINLI by empirically comparing three settings: fourway, threeway, and binary classification. For fourway classification, we add an extra relation "sibling" in addition to child, parent, and neutral relationships. A sibling pair consists of two categories that share the same parent. We also ensure that neutral pairs are non-siblings, meaning that we separate a category that was considered as part of the neutral relations to provide a more fine-grained pretraining signal.

We construct two versions of WIKINLI with binary class labels. One classifies the child against the rest, including parent, neutral, and sibling ("child vs. rest"). The other classifies child or parent against neutral or sibling ("child/parent vs. rest"). The purpose of these two datasets is to find if a more coarse training signal would reduce the gains from pretraining.

These dataset variations are each balanced among their classes and contain 100,000 training instances and 5,000 development instances.

Table 5 shows results of MNLI, RTE, and PPDB. Overall, fourway and threeway classifications are comparable, although they excel at different tasks. Interestingly, we find that pretraining with child/parent vs. rest is worse than pretraining with child vs. rest. We suspect this is because the child/parent vs. rest task resembles topic clas-

phrase 1	phrase 2	gold	BERT	WIKINLI	WordNet	Wikidata
car	the trunk	hypernym	other-related	hypernym	hypernym	hypernym
return	return home	hypernym	synonym	hypernym	hypernym	hypernym
boys are	the children are	hyponym	synonym	hyponym	hyponym	hyponym
foreign affairs	foreign minister	other-related	hypernym	other-related	hypernym	hypernym
company	debt	other-related	independent	independent	other-related	other-related
europe	japan	alternation	hypernym	alternation	independent	alternation
family	woman	independent	independent	hypernym	independent	other-related

Table 8: Examples from PPDB development set showing the effect of pretraining resources.

	2k	3k	5k	10k	20k
MNLI					
BERT	72.2	74.4	76.6	78.8	80.4
WIKINLI	74.5	75.6	77.3	79.1	80.6
Δ	+2.3	+1.2	+0.7	+0.3	+0.2
PPDB					
BERT	55.5	59.2	59.9	68.1	68.6
WIKINLI	65.0	66.4	67.9	70.2	71.2
Δ	+9.5	+7.2	+8.0	+2.1	+2.6

Table 9: Results for varying numbers of MNLI or PPDB training instances. The rows "Δ" show improvements from WIKINLI. We use all the training instances for PPDB in the "20k" setting.

	antonym	alternation	hyponym	hypernym
w/	34	51	276	346
w/o	1	35	231	248

Table 10: Per category numbers of correctly predicted instances by BERT with or without pretraining on WIKINLI.[6]

sification. The model does not need to determine direction of entailment, but only whether the two phrases are topically related, as neutral pairs are generally either highly unrelated or only vaguely related. The child vs. rest task still requires reasoning about entailment as the models still need to differentiate between child and parent.

In addition, we explore pruning levels in Wikipedia category graphs, and incorporating sentential context, finding that relatively higher levels of knowledge from WIKINLI have more potential of enhancing the performance of NLI systems and sentential context shows promising results on the Break dataset (see supplementary material for more details).

6.3 Larger Training Sets

We train on larger numbers of WIKINLI instances, approximately 400,000, for both threeway and fourway classification. We note that we only pretrain models on WIKINLI for one epoch as it leads to better performance on downstream tasks. The results are in Table 6. We observe that except for PPDB, adding more data generally improves performance. For Break, we observe significant improvements when using fourway WIKINLI for pretraining, whereas threeway WIKINLI seems to hurt the performance.

	R1	R2	R3
BERT	39.8	37.0	41.3
+ WordNet	41.1	38.2	39.9
+ Wikidata	43.2	39.0	41.8
+ WIKINLI	39.6	38.2	39.3
RoBERTa	46.1	39.3	39.4
+ WordNet	53.7	38.7	37.9
+ Wikidata	51.5	39.6	39.8
+ WIKINLI	51.2	38.1	39.4

Table 11: Test results for ANLI.

6.4 Combining Multiple Data Sources

We combine multiple data sources for pretraining. In one setting we combine 50k instances of WIKINLI with 50k instances of WordNet, while in the other setting we combine 50k instances of WIKINLI with 50k instances of Wikidata. Table 7 compares these two settings for pretraining. WIKINLI works the best when pretrained alone.

6.5 Effect of Pretraining Resources

We show several examples of predictions from PPDB in Table 8. In general, we observe that without pretraining, BERT tends to predict symmetric categories, such as synonym, or other-related, instead of predicting entailment-related categories. For example, the phrase pair "car" and "the trunk", "return" and "return home", and "boys are" and "the children are". These are either "hypernym" or "hyponym" relationship, but BERT tends to conflate them with symmetric relationships, such as other-related. To quantify this hypothesis, we compute the numbers of correctly predicted antonym, alternation, hyponym and hypernym and show them in Table 10. It can be seen that with pretraining those numbers increase dramatically, showing the benefit of pretraining on these resources.

[6] We observed similar trends when pretraining on the other resources.

We also observe that the model performance can be affected by the coverage of pretraining resources. In particular, for phrase pair "foreign affairs" and "foreign minister", WIKINLI has a closely related term "foreign affair ministries" and "foreign minister" under the category "international relations", whereas WordNet does not have these two, and Wikidata only has "foreign minister".

As another example, consider the phrase pair "company" and "debt". In WIKINLI, "company" is under the "business" category and debt is under the "finance" category. They are not directly related. In WordNet, due to the polysemy of "company", "company" and "debt" are both hyponyms of "state", and in Wikidata, they are both a subclass of "legal concept".

For the phrase pair "family"/"woman", in WIKINLI, "family" is a parent category of "wives", and in Wikidata, they are related in that "family" is a subclass of "group of humans". In contrast, WordNet does not have such knowledge.

6.6 Finetuning with Different Amounts of Data

We now look into the relationship between the benefit of WIKINLI and the number of training instances from downstream tasks (Table 9). We compare BERT-large to BERT-large pretrained on WIKINLI when finetuning on 2k, 3k, 5k, 10k, and 20k MNLI or PPDB training instances accordingly. In general, the results show that WIKINLI has more significant improvement with less training data, and the gap between BERT-large and WIKINLI narrows as the training data size increases. We hypothesize that the performance gap does not reduce as expected between 3k and 5k or 10k and 20k due in part to the imbalanced number of instances available for the categories. For example, even when using 20k training instances, some of the PPDB categories are still quite rare.

6.7 Evaluating on Adversarial NLI

Adversarial NLI (ANLI; Nie et al., 2020) is collected via an iterative human-and-model-in-the-loop procedure. ANLI has three rounds that progressively increase the difficulty. When finetuning the models for each round, we use the sampled 3k instances from the corresponding training set, perform early stopping on the original development sets, and report results on the original test sets. As shown in Table 11, our pretraining approach has diminishing effect as the round number increases.

	fr	ar	ur	zh	avg.
mBERT	61.5	57.3	49.3	57.9	56.5
mWIKINLI	62.5	56.8	51.5	59.9	57.7
trWIKINLI	63.0	**57.7**	51.3	59.9	58.0
WIKINLI	**63.3**	57.1	**51.8**	**60.0**	**58.1**
Wikidata	63.2	56.9	49.5	59.8	57.4
WordNet	63.1	56.0	50.5	58.6	57.1

Table 12: Test set results for XNLI. mWIKINLI is constructed from Wikipedia in other languages. trWIKINLI is translated from the English WIKINLI. The highest numbers in each column are boldfaced.

This may due to the fact that humans deem the NLI instances that require world knowledge as the hard ones, and therefore when the round number increases, the training set is likely to have more such instances, which makes pretraining on similar resources less helpful. Table 11 also shows that WordNet and Wikidata show stronger performance than WIKINLI. We hypothesize that this is because ANLI has a context length almost 3 times longer than MNLI on average, in which case our phrase-based resources or pretraining approach are not optimal choices. Future research may focus on finding better ways to incorporate sentential context into WIKINLI. For example, we experiment with such a variant of WIKINLI (i.e., WIKISENTNLI) in the supplementary material.

We have similar observations that our phrase-based pretraining has complicated effect (e.g., only part of the implicature results shows improvements) when evaluating these resources on IMPPRES (Jeretic et al., 2020), which focuses on the information implied in the sentential context (please refer to the supplementary materials for more details).

7 Multilingual WIKINLI

Wikipedia has different languages, which naturally motivates us to extend WIKINLI to other languages. We mostly follow the same procedures as English WIKINLI to construct a multilingual version of WIKINLI from Wikipedia in other languages, except that (1) we filter out instances that contain English words for Arabic, Urdu, and Chinese; and (2) we translate the keywords into Chinese when filtering the Chinese WIKINLI. We will refer to this version of WIKINLI as "mWIKINLI". As a baseline, we also consider "trWIKINLI", where we translate the English WIKINLI into other languages using Google Translate. We benchmark these resources on XNLI in four languages: French (fr), Arabic (ar), Urdu (ur), and Chinese (zh). When reporting these results, we pretrain

multilingual BERT (mBERT; Devlin et al., 2019) on the corresponding resources, finetune it on 3000 instances of the training set, perform early stopping on the development set, and test it on the test set. We always use XNLI from the corresponding language. In addition, we pretrain mBERT on English WIKINLI, Wikidata, and WordNet, finetune and evaluate them on other languages using the same language-specific 3000 NLI pairs mentioned earlier. We note that when pretraining on mWIKINLI or trWIKINLI, we use the versions of these datasets with the same languages as the test sets.

Table 12 summarizes the test results on XNLI. In general, pretraining on WIKINLI gives the best results. Phang et al. (2020) also observed that training on English intermediate tasks helps in cross-lingual tasks but in a zero-shot setting. While mWIKINLI is not the best resource, it still gives better results on average than Wikidata, WordNet, and no pretraining at all. The exception is Arabic, where only trWIKINLI performs better than the mBERT baseline. In comparing among different versions of WIKINLI, we find that trWIKINLI performs almost as good as WIKINLI, but for Urdu, trWIKINLI is the worst resource among the three. The variance of trWIKINLI may arise from the variable quality of machine translation across languages.

The accuracy differences between mWIKINLI and WIKINLI could be partly attributed to domain differences across languages. To measure the differences, we compile a list of the top 20 most frequent words in the Chinese mWIKINLI, shown in Table 13. The most frequent words for mWIKINLI in Chinese are mostly related to political concepts, whereas WIKINLI offers a broader range of topics.

Future research will be required to obtain a richer understanding of how training on WIKINLI benefits non-English languages more than training on the language-specific mWIKINLI resources. One possibility is the presence of emergent cross-lingual structure in mBERT (Wu et al., 2019). Nonetheless, we believe mWIKINLI and our training setup offer a useful framework for further research into multilingual learning with pretrained models.

8 Conclusion

We constructed WIKINLI, a large-scale naturally-annotated dataset for improving model performance on NLI and LE tasks. Empirically, we benchmarked WordNet, Wikidata, and WIKINLI using both BERT and RoBERTa by first pretraining these models on those resources, then finetuning on downstream tasks. The results showed that pretraining on WIKINLI gives the largest gains averaging over 8 different datasets. The improvements to both BERT and RoBERTa showed that the benefit of WIKINLI can generalize. We also performed an in-depth analysis on ways of constructing WIKINLI, and a lexical analysis on the differences between the three benchmarked resources. Our experiments on mWIKINLI showed promising results and can benefit the research on multilinguality.

WIKINLI	Chinese mWIKINLI
albums	中国(China)
songs	中华人民共和国(P. R. C.)
players	行政区划(administrative division)
male	人(man)
people	政治(politics)
American	人物(people)
British	各国(countries)
writers	组织(organization)
(band)	各省(provinces)
female	建筑物(building)

Table 13: 10 words from the top 20 most frequent words in WIKINLI, and mWIKINLI in Chinese. Each Chinese word is followed by a translation in parenthesis. The full list is shown in the supplementary material.

Acknowledgments

This research was supported in part by a Bloomberg data science research grant to K. Stratos and K. Gimpel.

References

Collin Baker. 2014. FrameNet: A knowledge base for natural language processing. In *Proceedings of Frame Semantics in NLP: A Workshop in Honor of Chuck Fillmore (1929-2014)*, pages 1–5, Baltimore, MD, USA. Association for Computational Linguistics.

Mohit Bansal, David Burkett, Gerard de Melo, and Dan Klein. 2014. Structured learning for taxonomy induction with belief propagation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1041–1051, Baltimore, Maryland. Association for Computational Linguistics.

Roy Bar Haim, Ido Dagan, Bill Dolan, Lisa Ferro, Danilo Giampiccolo, Bernardo Magnini, and Idan Szpektor. 2006. The second PASCAL recognising textual entailment challenge.

Marco Baroni, Raffaella Bernardi, Ngoc-Quynh Do, and Chung-chieh Shan. 2012. Entailment above the word level in distributional semantics. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 23–32, Avignon, France. Association for Computational Linguistics.

Luisa Bentivogli, Ido Dagan, Hoa Trang Dang, Danilo Giampiccolo, and Bernardo Magnini. 2009. The fifth PASCAL recognizing textual entailment challenge.

Jonathan Berant, Noga Alon, Ido Dagan, and Jacob Goldberger. 2015. Efficient global learning of entailment graphs. *Computational Linguistics*, 41(2):221–263.

Rahul Bhagat, Patrick Pantel, and Eduard Hovy. 2007. LEDIR: An unsupervised algorithm for learning directionality of inference rules. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 161–170, Prague, Czech Republic. 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.

Mingda Chen, Zewei Chu, Yang Chen, Karl Stratos, and Kevin Gimpel. 2019a. EntEval: A holistic evaluation benchmark for entity representations. 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 421–433, Hong Kong, China. Association for Computational Linguistics.

Mingda Chen, Zewei Chu, and Kevin Gimpel. 2019b. Evaluation benchmarks and learning criteria for discourse-aware sentence representations. 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 649–662, Hong Kong, China. Association for Computational Linguistics.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Diana Inkpen, and Si Wei. 2018. Neural natural language inference models enhanced with external knowledge. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2406–2417.

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.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2006. The PASCAL recognising textual entailment challenge. In *Machine learning challenges. evaluating predictive uncertainty, visual object classification, and recognising tectual entailment*, pages 177–190. Springer.

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.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. Language, Speech, and Communication. MIT Press, Cambridge, MA.

Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. 2007. The third PASCAL recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL workshop on textual entailment and paraphrasing*, pages 1–9. Association for Computational Linguistics.

Max Glockner, Vered Shwartz, and Yoav Goldberg. 2018. Breaking NLI systems with sentences that require simple lexical inferences. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 650–655, Melbourne, Australia. Association for Computational Linguistics.

Marti A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *COLING 1992 Volume 2: The 15th International Conference on Computational Linguistics*.

Ari Holtzman, Jan Buys, Maxwell Forbes, Antoine Bosselut, David Golub, and Yejin Choi. 2018. Learning to write with cooperative discriminators. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1638–1649, Melbourne, Australia. Association for Computational Linguistics.

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.

Paloma Jeretic, Alex Warstadt, Suvrat Bhooshan, and Adina Williams. 2020. Are natural language inference models IMPPRESsive? Learning IMPlicature

and PRESupposition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8690–8705, Online. Association for Computational Linguistics.

Tushar Khot, Ashish Sabharwal, and Peter Clark. 2018. Scitail: A textual entailment dataset from science question answering.

Lili Kotlerman, Ido Dagan, Idan Szpektor, and Maayan Zhitomirsky-Geffet. 2009. Directional distributional similarity for lexical expansion. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 69–72, Suntec, Singapore. Association for Computational Linguistics.

Matthew Le, Stephen Roller, Laetitia Papaxanthos, Douwe Kiela, and Maximilian Nickel. 2019. Inferring concept hierarchies from text corpora via hyperbolic embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3231–3241, Florence, Italy. Association for Computational Linguistics.

Omer Levy, Steffen Remus, Chris Biemann, and Ido Dagan. 2015. Do supervised distributional methods really learn lexical inference relations? In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 970–976, Denver, Colorado. Association for Computational Linguistics.

Dekang Lin and Patrick Pantel. 2001. Discovery of inference rules for question-answering. *Nat. Lang. Eng.*, 7(4):343–360.

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 preprint arXiv:1907.11692*.

Robert Logan, Nelson F. Liu, Matthew E. Peters, Matt Gardner, and Sameer Singh. 2019. Barack's wife hillary: Using knowledge graphs for fact-aware language modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5962–5971, Florence, Italy. Association for Computational Linguistics.

Lajanugen Logeswaran, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, Jacob Devlin, and Honglak Lee. 2019. Zero-shot entity linking by reading entity descriptions. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3449–3460, Florence, Italy. Association for Computational Linguistics.

Paul McNamee, Rion Snow, Patrick Schone, and James Mayfield. 2008. Learning named entity hyponyms for question answering. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-II*.

Yashar Mehdad, Giuseppe Carenini, Frank Tompa, and Raymond T. NG. 2013. Abstractive meeting summarization with entailment and fusion. In *Proceedings of the 14th European Workshop on Natural Language Generation*, pages 136–146, Sofia, Bulgaria. Association for Computational Linguistics.

Vivi Nastase and Michael Strube. 2008. Decoding wikipedia categories for knowledge acquisition. In *Proceedings of the 23rd National Conference on Artificial Intelligence - Volume 2*, AAAI'08, page 1219–1224. AAAI Press.

Vivi Nastase, Michael Strube, Benjamin Boerschinger, Caecilia Zirn, and Anas Elghafari. 2010. WikiNet: A very large scale multi-lingual concept network. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Languages Resources Association (ELRA).

Allen Nie, Erin Bennett, and Noah Goodman. 2019. DisSent: Learning sentence representations from explicit discourse relations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4497–4510, Florence, Italy. Association for Computational Linguistics.

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.

Yasumasa Onoe and Greg Durrett. 2020. Fine-grained entity typing for domain independent entity linking. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Marius Pasca and Benjamin Van Durme. 2007. What you seek is what you get: Extraction of class attributes from query logs. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI-07)*, pages 2832–2837.

Ellie Pavlick, Johan Bos, Malvina Nissim, Charley Beller, Benjamin Van Durme, and Chris Callison-Burch. 2015. Adding semantics to data-driven paraphrasing. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1512–1522, Beijing, China. Association for Computational Linguistics.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of NAACL-HLT*, pages 2227–2237.

Matthew E. Peters, Mark Neumann, Robert Logan, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A.

Smith. 2019. Knowledge enhanced contextual word representations. 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 43–54, Hong Kong, China. Association for Computational Linguistics.

Jason Phang, Phu Mon Htut, Yada Pruksachatkun, Haokun Liu, Clara Vania, Katharina Kann, Iacer Calixto, and Samuel R. Bowman. 2020. English intermediate-task training improves zero-shot cross-lingual transfer too.

Simone Paolo Ponzetto and Michael Strube. 2007. Deriving a large scale taxonomy from wikipedia. In *Proceedings of the 22nd National Conference on Artificial Intelligence - Volume 2*, AAAI'07, page 1440–1445. AAAI Press.

Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M. Marlin. 2013. Relation extraction with matrix factorization and universal schemas. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 74–84, Atlanta, Georgia. Association for Computational Linguistics.

Vered Shwartz, Yoav Goldberg, and Ido Dagan. 2016. Improving hypernymy detection with an integrated path-based and distributional method. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2389–2398, Berlin, Germany. Association for Computational Linguistics.

Rion Snow, Daniel Jurafsky, and Andrew Y. Ng. 2005. Learning syntactic patterns for automatic hypernym discovery. In L. K. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 1297–1304. MIT Press.

Rion Snow, Daniel Jurafsky, and Andrew Y. Ng. 2006. Semantic taxonomy induction from heterogenous evidence. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 801–808, Sydney, Australia. 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.

Idan Szpektor, Hristo Tanev, Ido Dagan, and Bonaventura Coppola. 2004. Scaling web-based acquisition of entailment relations. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 41–48, Barcelona, Spain. Association for Computational Linguistics.

Peter D. Turney and Saif M. Mohammad. 2015. Experiments with three approaches to recognizing lexical entailment. *Natural Language Engineering*, 21:437–476.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: A free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations*.

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, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

Shijie Wu, Alexis Conneau, Haoran Li, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Emerging cross-lingual structure in pretrained language models.

Alexander Yates and Oren Etzioni. 2009. Unsupervised methods for determining object and relation synonyms on the web. *J. Artif. Int. Res.*, 34(1):255–296.

Torsten Zesch and Iryna Gurevych. 2007. Analysis of the Wikipedia category graph for NLP applications. In *Proceedings of the Second Workshop on TextGraphs: Graph-Based Algorithms for Natural Language Processing*, pages 1–8, Rochester, NY, USA. Association for Computational Linguistics.