Appendix

Textual Entailment for Link Prediction in WordNet

Clément Beliveau[1], Guillermo Echegoyen Blanco[2], and José Manuel Gómez-Pérez[2]

[1] IMT Atlantique, Nantes, France clement.beliveau@imt-atlantique.org
[2] Expert.ai {gblanco, jmgomez}@expert.ai

Abstract. The construction of structured knowledge bases like WordNet where information is stored as nodes interconnected with links is a resource and time-consuming process, requiring large amounts of manual effort from domain experts. Knowledge base completion and population aim to automate the process of mining those links or nodes respectively, reducing the dependency on expert labour. With the advent of language models, new approaches that exploit the inherent knowledge within them have been proposed to address knowledge base completion and population tasks. In parallel, entailment models have shown outstanding performance in various tasks such as relation extraction and classification. However, the application of entailment models to the more challenging task of link prediction between two nodes of a graph knowledge base remains unexplored. In this paper, we propose to solve the link prediction task using entailment as a proxy task. Our model consistently outperforms the state-of-the-art on the WN18RR benchmark by more than 20 points in Hit@1 and Hit@3 with just 10% of the available training examples. Furthermore, we introduce a *two-way* entailment training regime that leverages knowledge about inverse relations to further beat the state-of-the-art, needing as little as 7% of the training data in a few-shot scenario and demonstrating impressive capabilities to predict WordNet links.

1 Acknowledgments

A deep thanks to the team of Expert.ai for their help during this project. José Manuel Gómez and Guillermo E. Blanco had precious advice during the project. BrAin lab of IMT Atlantique provided some GPU NVidia 1060.

2 Hyperparameters

The hyperparameters used in the training of our model are as follows:

- **Per Device Train Batch Size:** 1
- **Gradient Accumulation Steps:** 32
- **Learning Rate:** 4×10^{-6}

- **Warmup Ratio:** 0.1
- **Weight Decay:** 0.06
- **Number of Training Epochs:** 3
- **fp16**

These hyperparameters were selected based on a systematic tuning process to achieve optimal model performance.

2.1 Relation verbalisation

Each relation is associated with a template for the verbalisation. As shown in Sainz et al., 2022, the entailment model leverages the need for experts as the labels verbalised by expert linguists have the same performances as the ones generated by non-experts. Table 1, contains the templates designed for the relations (Template 1) and the ones designed for their inverse (Template 2).

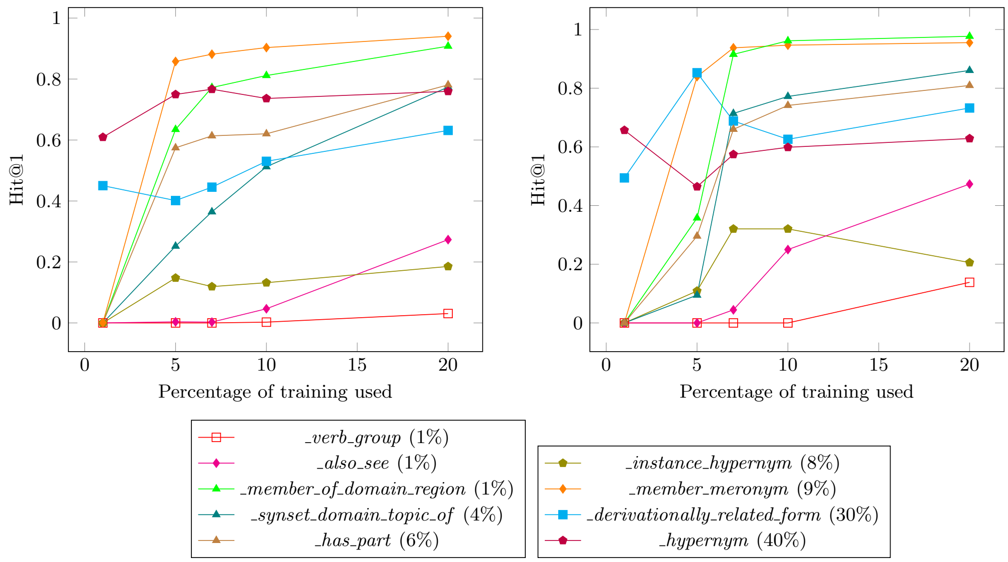
Relation	Template 1	Template 2
_hypernym	{*obj*} specifies {*subj*}	{*subj*} generalize {*obj*}
_derivationally_related_form	{*obj*} derived from {*subj*}	{*subj*} derived from {*obj*}
_instance_hypernym	{*obj*} is a {*subj*}	{*subj*} such as {*obj*}
_also_see	{*obj*} is seen in {*subj*}	{*subj*} has {*obj*}
_member_meronym	{*obj*} is the family of {*subj*}	{*subj*} is a member of {*obj*}
_synset_domain_topic_of	{*obj*} is a topic of {*subj*}	{*subj*} is the context of {*obj*}
_has_part	{*obj*} contains {*subj*}	{*subj*} is a part of {*obj*}
_member_of_domain_region	{*obj*} is the domain region of {*subj*}	{*subj*} belong to the region of {*obj*}
_verb_group	{*obj*} is synonym to {*subj*}	{*subj*} is synonym to {*obj*}
_similar_to	{*obj*} is similar to {*subj*}	{*subj*} similar to {*obj*}

Table 1: Relation templates

3 More Analytics results

3.1 *1-way* Training

Figure 1 above illustrates the performance evolution for each relation. On this figure, we can verify if the model overfits some overrepresented relations and does not learn the underrepresented ones. Indeed, the benchmark is heavily unbalanced, so some relations like `_hypernym` are much more represented than others. This could lead the model to only learn those overrepresented relations. We can see that with a certain amount of data, the performance tends to converge for each relation. Moreover, even if the model is not able to learn some relations



(a) (b)

Fig. 1: Comparison of the performances (Hit@1) of the individual relations between the MNLI and the Naive model in a *1-way* training

like `_verb_group` or `_also_see` which represent 1% of the dataset it is still able to learn relations like `_member_of_domain_region` or `_synset_domain_topic_of` which are also represented in a small fraction of the dataset.

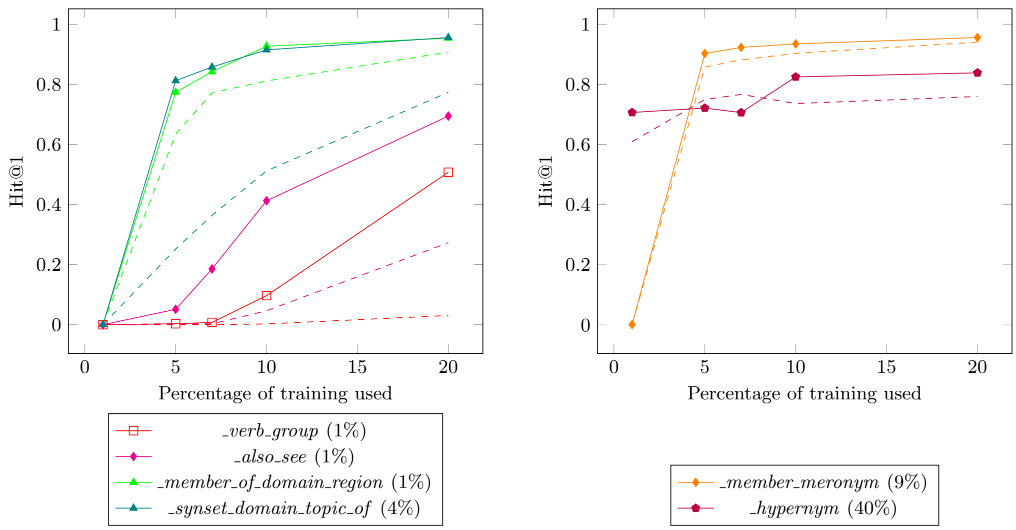
3.2 *2-way* Training

In Figure 2 the increase in performances for the relation `_member_meronym` is not significant, as for `_hypernym` relation where the increase in performances is not as visible as for the underrepresented relations.

3.3 *2-way* Inference

This part focuses on the gain of performance obtained by adding the information of the inverse relation during the inference process. To do so we evaluate a model trained on the *2-way* approach as presented in (**??**) with an inference protocol based on the relation and its inverse relation to infer which is the correct relation as explained in (**??**).

The *2-way* inference does not seem to have an important impact on the inference performance. Even if the results are slightly higher, it is not a huge performance gap. This is probably of entailment of the relation already holding enough information *1-way* inference to be able to correctly discriminate the right



(a) (b)

Fig. 2: Comparison of performances between a *2-way* and a *1-way* (dashed lines) training between under-represented and over-represented relations

Percentage	**Hit**	***2-way* Training**	
		DeBERTta-MNLI	**DeBERTta-Naive**
1%	@1	0.392 ± 0.011	0.463 ± 0.012
	@3	0.885 ± 0.006	0.879 ± 0.008
5%	@1	0.640 ± 0.043	0.622 ± 0.036
	@3	0.983 ± 0.002	0.988 ± 0.001
7%	@1	0.657 ± 0.045	0.615 ± 0.053
	@3	0.986 ± 0.002	0.990 ± 0.001
10%	@1	0.683 ± 0.053	0679 ± 0.063
	@3	0.989 ± 0.001	0.990 ± 0.003
20%	@1	0.804 ± 0.037	0.760 ± 0.037
	@3	0.995 ± 0.001	0.992 ± 0.003
100%**	@1	0.898	0.887
	@3	0.998	0.996

Table 2: Hit@n for DeBERTta-MNLI and DeBERTta-Naive at different training percentages. *2-way* training, ***2-way* inference**

relation from the other ones. Hence adding the inverse probability only adds a small boost in performance.

3.4 ***Unbiased dataset***

As shown in Figure 3 of the paper, the label distribution is highly unbalanced, with two labels representing 70% of the training data. In order to see if our entailment approach is sensitive to the balance of the data, we proceed to a re-balancing of the training data. To do so we force a uniform distribution between the label.

The results presented in Table 3 are presented in section 3 of the appendix. show that the approach is relatively robust to an unbalanced dataset as the gain achieved by balancing the data is a maximum of 2.8%

Percentage	**Hit**	***2-way* Training**	
		DeBERTta-MNLI	**DeBERTta-Naive**
1%	@1	0.420 ± 0.025	0.335 ± 0.041
	@3	0.840 ± 0.030	0.759 ± 0.113
5%	@1	0.632 ± 0.068	0.621 ± 0.047
	@3	0.969 ± 0.015	0.939 ± 0.009
7%	@1	0.675 ± 0.022	0.673 ± 0.048
	@3	0.981 ± 0.002	0.963 ± 0.009
10%	@1	0.675 ± 0.042	0.713 ± 0.028
	@3	0.981 ± 0.002	0.964 ± 0.007

Table 3: Hit@n for DeBERTta-MNLI and DeBERTta-Naive at different training percentages on a **balanced dataset**. *2-way* training, ***1-way* inference**

References

Sainz, O., Gonzalez-Dios, I., Lopez de Lacalle, O., Min, B., & Agirre, E. (2022, July). Textual entailment for event argument extraction: Zero- and few-shot with multi-source learning. In M. Carpuat, M.-C. de Marneffe, & I. V. Meza Ruiz (Eds.), *Findings of the association for computational linguistics: Naacl 2022* (pp. 2439–2455). Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.findings-naacl.187