ManyEnt: A Dataset for Few-shot Entity Typing

Markus Eberts **Kevin Pech** **Adrian Ulges**
RheinMain University of Applied Sciences, Germany
{markus.eberts, adrian.ulges}@hs-rm.de
kevin.pech@student.hs-rm.de

Abstract

We introduce ManyEnt, a benchmark for entity typing models in few-shot scenarios. ManyEnt offers a rich typeset, with a fine-grain variant featuring 256 entity types and a coarse-grain one with 53 entity types. Both versions have been derived from the Wikidata knowledge graph in a semi-automatic fashion. We also report results for two baselines using BERT, reaching up to 70.68% accuracy (10-way 1-shot).

1 Introduction

Information extraction is targeted at inferring knowledge from unstructured documents, most commonly in the form of entities and relations between them. Like many other NLP tasks, the field has recently experienced a boost from deep neural networks (Wang et al., 2019; Wadden et al., 2019; Li et al., 2019). While research datasets are usually derived from Wikipedia, for industrial applications it is the adaptation to new domains which matters: Here, different relations may be of interest than on the source domain, and existing relations may express themselves differently (think of the "part_of" relation, which differs between the medical domain and mechanical engineering).

As data is often scarce on the target domain, recent work has addressed *few-shot* scenarios: A model is pre-trained on a training set of relations representing the source domain and is then adapted to new relations representing the target domain, usually using only few (i.e., $1-5$) samples. One prominent example is the FewRel benchmark (Han et al., 2018) for relation classification: Given a sentence in which two entity mentions are marked, the tasks is to assign the entity pair to a relation. Models are first trained on a dataset of 80 relation types and are then applied to held-out test relation types.

While FewRel addresses relations, we argue that – since information extraction models targeted at new domains will have to adapt both to new entities and relations – the few-shot scenario is also interesting for entity typing. This way, models pretrained on some entity types (such as *computer*) can be adapted to other types (such as *smartphone*). Therefore, we suggest a novel benchmark "ManyEnt"[1] for few-shot entity typing. The benchmark is based on FewRel and features a similar setting: Given a sentence with a highlighted entity, the entity is to be assigned to an entity type. Here, a rich number of types is beneficial: Only if a breadth of types is known in training transfer can be expected to work well by adapting to "similar" types as the ones from the source domain. While FewRel offers a much richer typeset of relations (80 types) than other datasets, e.g. (Hendrickx et al., 2010), it offers no entity types. Therefore, our first contribution is a semi-automatic annotation of entities on FewRel at two granularities (53 types and 256 types) using the Wikidata knowledge graph. Second, we also report the results of two common baselines based on the well-known BERT model (Devlin et al., 2018), which reach an accuracy of up to 70.68% (10-way 1-shot).

[1] Our dataset (including more detailed statistics) can be found at https://github.com/markus-eberts/many-ent.

Proceedings of the 28th International Conference on Computational Linguistics, pages 5553–5557
Barcelona, Spain (Online), December 8-13, 2020

2 Related Work

Few-shot learning has been applied to numerous NLP tasks such as relation extraction (Han et al., 2018) and text classification (Yu et al., 2018). Here specialized few-shot architectures like meta networks (Munkhdalai and Yu, 2017) or prototypical networks (Snell et al., 2017) were shown to alleviate overfitting and improve generalization compared to naive finetuning baselines.

With respect to entity typing, models and benchmarks addressing rich typesets exist (Ling and Weld, 2012; Choi et al., 2018). We follow a similar approach in label acquisition to Ling and Weld (2012), exploiting the Wikidata hierarchy. Only few previous works, however, are specifically targeted at few-shot scenarios: Hofer et al. (2018) apply a BiLSTM with GloVE embeddings for NER in medical texts. They show that few-shot performance can be improved by a pre-training on related domains. In comparison with ManyEnt, the dataset used by Hofer et al. contains only a small number of entity labels (6) and is very domain specific. Fritzler et al. (2019) train a prototypical network based model on the Ontonotes dataset (18 entity types, about 170k sentences). ManyEnt is smaller regarding sentences, but contains a broader set of entity types. Li et al. (2020) apply a meta-learning approach for NER domain adaption. They use six domains with homogeneous entity types, while our goal is to adapt a model to unknown entity types given only a few labeled samples. Finally, Ma et al. (Ma et al., 2016) present a few-shot and zero-shot model focusing on category representations exploiting similarities between category labels (e.g., *book* vs. *song*). In contrast, our model is based on instance-level prototypes similar to FewRel.

3 The ManyEnt Dataset

The basis for ManyEnt is the FewRel dataset for few-shot relation extraction (Han et al., 2018). FewRel offers a large-scale annotated set of sentences extracted from Wikipedia, annotated with 80 relation types but no entity types. FewRel also contains two annotated entity mentions per sentence, each linked to a Wikidata[2] entity. We denote the set of all these entities with E.

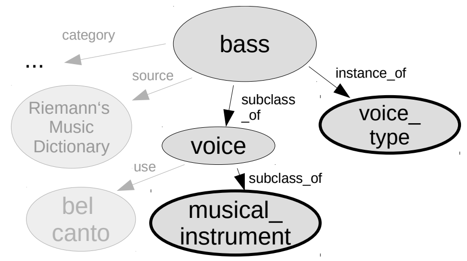
While Wikidata comes with a rich concept hierarchy, we found many of its entity types to be unsuitable: They are either too specific (*high school*), too coarse/uninformative (*physical object*) or unintuitive (*artificial geographic entity*). We would like to select a subset T of "suitable" entity types, following common criteria for concept ontologies (Naphade et al., 2006) such as utility (practical relevance), coverage (semantic breadth), feasibility of detection, and observability in the corpus. Given the subset of suitable entity types T, we map each Wikidata entity from E to a type from T using a semi-automatic procedure outlined in the following section.

3.1 Mapping Entities to Entity Types

For now, we assume the entity typeset T to be given and describe our mapping as a function $map : E \rightarrow T$ from Wikidata entities to entity types. As common for knowledge graphs, Wikidata consists of triples linking concept nodes via relations. Though Wikidata does not feature a separate T-Box and A-Box, we assume that some nodes represent concrete *entities* E (such as *USA*) while others represent *entity types* (such as *country*). We identify six *indicator relations* – such as *instance_of* or *subclass_of* – leading to entity types.

Starting from an entity $e{\in}E$, we follow all edges labeled with any of the six indicator relations in a breadth search until reaching one of our predefined types $t{\in}T$, and set $map(e){=}t$. We maintain nodes to be visited in a queue, and rank the indicator relations such that when expanding a node, certain relations (e.g., subclass_of) are inserted into the queue before others (e.g., part_of). See the example in Figure 1, where we compute an entity type for the voice level "bass": While other relations such as *source* or *use* are ignored, our search follows the relations *subclass_of* and *instance_of*. Since a breadth search is used, we map to the node *voice_type* on Level 1 instead of *musical_instrument* on Level 2. If the search remains unsuccessful, we set $map(e){=}None$.

[2] https://www.wikidata.org/



	#sentences*	#mentions*	#types (ME-53)	#types (ME-256)
training	44,485	89,793	26	80
test	9,536	11,012	27	176

Figure 1: Left: We map the entity "bass"$\in E$ to the entity type *voice_type* $\in T$ using a breadth search following key relations (here, *subclass_of* and *instance_of*). Right: Dataset Statistics (* for ME-256). As ManyEnt addresses a few-shot scenario, we utilize larger types for training and sparse types for testing.

3.2 Identifying Entity Types

Obviously, we face a chicken-egg problem; while the breadth search in $map()$ requires a set T of terminal types, defining appropriate types requires $map()$ to get an impression of their entities. To estimate both T and $map()$, we initialize T by manually selecting 44 coarse preliminary entity types offering high coverage: For a substantial subset of entities $e \in E$, we conduct a manual search of the Wikidata graph for a suitable type. For example, given the entity *iPhone 6S Plus*, we traverse up the concept hierarchy (*iPhone*, *smartphone*, ...) and select the type *device*.

Afterwards, we optimize the set of entity types T by an iterative process alternating between re-estimating the mapping $map()$ automatically and refining the entity types T manually. Thereby, we segment the types T into increasingly finer ones by manually inspect sample type t's entities $map^{-1}(t)$. If we find these entities to be unintuitively diverse, we manually search the Wikidata graph for more fine-grain types and replace t with those types. For example, we found the type *role* to be too diverse and segmented it into subtypes such as *occupation* and *military rank*. We then recompute $map()$ using the procedure in Section 3.1. Note that due to the type refinement some entities may end up with no type. We revisit those *None*-entities and try to identify suitable types for them. Overall, we applied $6-7$ iterations of this process, each time removing about 10 types and adding about 40 finer ones. Finally, we discard 286 entity mentions for which no matching type was found.

3.3 Manual Post-Processing and Coarser Entity Types

We finally apply a manual refinement by a discussion of three experts: 14 non-sensical types were identified based on their instances (such as "abstract object") and discarded. This results in 256 fine-grain entity types, which we refer to as *ManyEnt-256* in the following. To assess the overall data quality, we manually inspect one random mention per type. We manually assign this mention to a type and compare this ground truth to $map(e)$. This resulted in an accuracy of 97.66%.

To also distinguish between different levels of granularity, we aggregate our rich 256-typeset to a medium-granularity typeset. Again, this process was based on a discussion and agreement between the same three experts, who inspected the 256 types and their entities and unified semantically similar types (such as "hill" and "mountain range") to new types. This process resulted in a set of 53 types, which we refer to as *ManyEnt-53*.

Dataset Split The sentences in ManyEnt correspond to FewRel's joined training and validation sets. A challenge is that the distribution of entity types is heavily skewed, which limits the applicability for few-shot scenarios. The most frequent class (human) occurs 24,100 times, while the median number of samples per class is only 86 and the minimum is 11. To include as many test types as possible, we sort the entity types by their frequency and use frequent types for training and infrequent ones for testing. Then each sentence is assigned to either training set or test set based on its two entities' types. If one type belongs to training and one to testing, we assign the sentence to the test set. Table 1 shows the statistics of the resulting splits.

ME-256	1-shot		5-shot	
fine-tuning?	no	yes	no	yes
5-way	53.72	**81.14**	81.08	**90.28**
10-way	43.42	**70.68**	74.14	**85.34**
20-way	34.71	**63.18**	65.37	**77.78**

ME-53	1-shot		5-shot	
fine-tuning?	no	yes	no	yes
5-way	55.46	**79.12**	83.02	**91.88**

Table 1: Accuracy (%) for our two BERT baselines for fine-grain types (left) and coarse types (right). Fine-tuning comes with improvements in accuracy by up to 29% (20-way 1-shot).

4 Experiments

We adapt the commonly used few-shot setting (Han et al., 2018; Chen et al., 2019) for entities. The setting is formulated as a N-way, K-shot problem: Given a set of N entity types (*way*), which the model did not encounter during training, the correct type of a sample must be predicted given only K examples (*shots*) per type. The set of entity type examples is commonly denoted as the *support set* and the samples that the system must classify as the *query set*. We design a prototypical network (Snell et al., 2017) using the pre-trained transformer-type network BERT (Devlin et al., 2018) as the encoder. Prototypical networks aim at creating a concise representation (the *prototype*) of a target class (here, an entity type) from the support set. It is trained on randomly sampled *episodes*, each consisting of K support samples per N entity types and Q query samples per type (see Snell et al. (2017) for more details). We first tokenize an input sentence, obtaining a sequence of n byte-pair encoded tokens. These tokens are then embedded with BERT into a sequence of contextualized embeddings $(\mathbf{e}_1, \mathbf{e}_2, ...\mathbf{e}_n)$. Given an entity span $s := (\mathbf{e}_i, \mathbf{e}_{i+1}, ..., \mathbf{e}_{i+k})$ with length *k*, an entity representation $\mathbf{e}(s)$ is obtained by averaging over s. The prototype representation of an entity type t is acquired by an averaging over the support set $\mathcal{S}_t$ of t:

$$\mathbf{p}(t) = \frac{1}{K} \sum_{s_i \in \mathcal{S}_t} \mathbf{e}(s_i) \tag{1}$$

Each query sample is also encoded (again by averaging over the entity's span) and compared with each prototype by Euclidean distance. A probability distribution over all entity types is obtained using a Softmax function over the negated distances. We then minimize the negative log-likelihood loss of the ground truth entity type. We trained the model for 3 epochs of 4,000 episodes. We use the BERT$_{\text{BASE}}$ (multilingual-cased) model for our experiments. Hyperparameters were tuned on a held-out validation set (10% of training data). We use the Adam optimizer with a learning rate of 3e-5 and a linear increase/decrease schedule and perform early stopping on the validation set.

We evaluate our model on 200 random episodes per setting, each containing 5 query samples per entity type. The accuracy is averaged over all episodes. We also include a second baseline, where BERT is not fine-tuned, to asses the inherent ability of BERT to distinguish entities through its unsupervised pre-training. For the fine-grain dataset, we evaluate on a 5-, 10-, and 20-way setting with 1 or 5 shots. Since ME-53 contains fewer types, only a 5-way setting is evaluated. Table 1 contains the results of our two baseline approaches for the fine-grain (ME-256) and coarse (ME-53) types. While we observe BERT to already perform well without fine-tuning, especially on the 5-shot settings, performance improves in every setting when BERT is fine-tuned instead. Here fine-tuning appears to be particularly important when given only a single example per type (up to 29% accuracy improvement).

5 Conclusions

We have introduced a new benchmark for few-shot entity typing featuring a rich entity typeset of 256 types. Results with two baselines using a BERT-based prototypical network indicate that BERT – particularly when fine-tuned – already provides a solid performance (62-93%). A key challenge to the system appears to be the separation of similar fine-grain types (such as "comic format" and "document"), which poses an interesting challenge for future research.

References

Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Wang, and Jia-Bin Huang. 2019. A Closer Look at Few-shot Classification. In *International Conference on Learning Representations*.

Eunsol Choi, Omer Levy, Yejin Choi, and Luke Zettlemoyer. 2018. Ultra-fine entity typing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 87–96, Melbourne, Australia, July. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR*, abs/1810.04805.

Alexander Fritzler, Varvara Logacheva, and Maksim Kretov. 2019. Few-Shot Classification in Named Entity Recognition Task. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, SAC '19, page 993–1000, New York, NY, USA. Association for Computing Machinery.

Xu Han, Hao Zhu, Pengfei Yu, Ziyun Wang, Yuan Yao, Zhiyuan Liu, and Maosong Sun. 2018. FewRel: A Large-Scale Supervised Few-Shot Relation Classification Dataset with State-of-the-Art Evaluation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4803–4809, Brussels, Belgium, October-November. Association for Computational Linguistics.

Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. 2010. SemEval-2010 Task 8: Multi-Way Classification of Semantic Relations between Pairs of Nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 33–38, Uppsala, Sweden, July. Association for Computational Linguistics.

Maximilian Hofer, Andrey Kormilitzin, Paul Goldberg, and Alejo J. Nevado-Holgado. 2018. Few-shot Learning for Named Entity Recognition in Medical Text. *CoRR*, abs/1811.05468.

Xiaoya Li, Fan Yin, Zijun Sun, Xiayu Li, Arianna Yuan, Duo Chai, Mingxin Zhou, and Jiwei Li. 2019. Entity-Relation Extraction as Multi-Turn Question Answering. In *Proc. of ACL 2019*, pages 1340–1350, Florence, Italy, July. ACL.

Jing Li, Shuo Shang, and Ling Shao. 2020. MetaNER: Named Entity Recognition with Meta-Learning. In *Proceedings of The Web Conference 2020*, WWW '20, page 429–440, New York, NY, USA. Association for Computing Machinery.

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

Yukun Ma, Erik Cambria, and Sa Gao. 2016. Label embedding for zero-shot fine-grained named entity typing. In Nicoletta Calzolari, Yuji Matsumoto, and Rashmi Prasad, editors, *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 171–180. ACL.

Tsendsuren Munkhdalai and Hong Yu. 2017. Meta Networks. *CoRR*, abs/1703.00837.

Milind Naphade, John R. Smith, Jelena Tesic, Shih-Fu Chang, Winston Hsu, Lyndon Kennedy, Alexander Hauptmann, and Jon Curtis. 2006. Large-Scale Concept Ontology for Multimedia. *IEEE MultiMedia*, 13(3):86–91, July.

Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical Networks for Few-shot Learning. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4077–4087. Curran Associates, Inc.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, Relation, and Event Extraction with Contextualized Span Representations. *ArXiv*, abs/1909.03546.

Haoyu Wang, Ming Tan, Mo Yu, Shiyu Chang, Dakuo Wang, Kun Xu, Xiaoxiao Guo, and Saloni Potdar. 2019. Extracting Multiple-Relations in One-Pass with Pre-Trained Transformers. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1371–1377, Florence, Italy, July. Association for Computational Linguistics.

Mo Yu, Xiaoxiao Guo, Jinfeng Yi, Shiyu Chang, Saloni Potdar, Yu Cheng, Gerald Tesauro, Haoyu Wang, and Bowen Zhou. 2018. Diverse Few-Shot Text Classification with Multiple Metrics. 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 1206–1215, New Orleans, Louisiana, June. Association for Computational Linguistics.