Generic Overgeneralization in Pre-trained Language Models

Sello Ralethe and **Jan Buys**
Department of Computer Science, University of Cape Town, South Africa
rltsel002@myuct.ac.za, jbuys@cs.uct.ac.za

Abstract

Generic statements such as "ducks lay eggs" make claims about *kinds*, e.g., ducks as a category. The generic overgeneralization effect refers to the inclination to accept false universal generalizations such as "all ducks lay eggs" or "all lions have manes" as true. In this paper, we investigate the generic overgeneralization effect in pre-trained language models experimentally. We show that pre-trained language models suffer from overgeneralization and tend to treat quantified generic statements such as "all ducks lay eggs" as if they were true generics. Furthermore, we demonstrate how knowledge embedding methods can lessen this effect by injecting factual knowledge about *kinds* into pre-trained language models. To this end, we source factual knowledge about two types of generics, minority characteristic generics and majority characteristic generics, and inject this knowledge using a knowledge embedding model. Our results show that knowledge injection reduces, but does not eliminate, generic overgeneralization, and that majority characteristic generics of kinds are more susceptible to overgeneralization bias. We release the dataset and code[1].

	PLM		**PLM+KEPLER**	
#	BERT	RoBERTa	BERT	RoBERTa
1	All	Some	Mountain	Male
2	Most	Most	Young	Mountain
3	Some	All	Male	Sea
4	Every	Many	Most	Some
5	Many	Even	Some	Most

Table 1: The top 5 words predicted by BERT and RoBERTa for filling the mask in the generic "[MASK] lions have manes". The outputs are shown before and after knowledge injection with KEPLER.

1 Introduction

Generics are sentences such as "tigers have stripes" that express generalizations about kinds, although they are not universal or without exceptions. For example, there are albino tigers that do not have stripes. Even though there are exceptions, generics are regarded as true. However, universally quantified statements such as "all ducks lay eggs" should be perceived as false as they can easily be invalidated because the quantifier *all* does not allow exceptions; it is only mature female ducks that are capable of laying eggs.

Empirical data from linguistics studies show that children and adults often tend to treat quantified statements such as "all tigers have stripes" as if they were generics (Khemlani et al., 2007; Hollander et al., 2002). Leslie et al. (2011) term this phenomena the *generic overgeneralization* (GOG) effect and allot it to a cognitive tendency that causes people to overgeneralize from the truth of a generic ("lions have manes") to the truth of a corresponding universal statement ("all lions have manes").

Pre-trained language models (PLMs) such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019b) learn useful language representation from large-scale unstructured text and are able to store information about the world in their parameters (Clark et al., 2019; Liu et al., 2019a; Rogers et al., 2020). In this paper, we investigate the GOG effect in PLMs, and ask if embedding factual knowledge about kinds during pre-training can reduce this effect.

First, we investigate if PLMs can improve the classification accuracy of distinguishing between generic and non-generic statements. We construct datasets of minority and majority characteristic generics of *kinds*, and use these to train and evaluate a baseline model and the PLMs. Results show that the pre-trained language models outperform the baseline on the classification task. We also demonstrate that the fine-tuned language models

[1] https://github.com/sello-ralethe/GOG-in-PLMs

Proceedings of the 29th International Conference on Computational Linguistics, pages 3187–3196
October 12–17, 2022.

classify quantified statements such as “all ducks lay eggs” as generics, indicating that the models do overgeneralize and thus exhibit the GOG effect.

Next, we define a masked word prediction task to test if the PLMs exhibit the GOG effect by observing if the PLMs predict quantifiers to fill the masked words. For example, we ask PLMs to fill the masked position in a statement such as “`[MASK]` lions have manes” (Table 1). The low performance of the PLMs on this task, evaluated using mean reciprocal rank and precision at 5, confirms the presence of the GOG effect.

Given these observations, we ask whether injecting factual knowledge about kinds during language model pre-training reduces the GOG effect. We source factual knowledge about kinds from ASCENT KB (Nguyen et al., 2021), a knowledge base which contains facet-enriched assertions together with their associated context. The knowledge from ASCENT KB is injected into the PLMs using KEPLER (Wang et al., 2021b), a model for embedding knowledge into PLMs using entity descriptions and entity relations data. Experimental results suggest that the injected knowledge lessens the GOG effect, but does not eliminate it completely.

Our contributions are: (i) we introduce new datasets for evaluating generic overgeneralization in PLMs, (ii) we demonstrate that PLMs exhibit the GOG effect, (iii) we show that embedding factual knowledge can reduce the GOG effect, and (iv) our results suggest that majority characteristic generics are more susceptible to overgeneralization bias. To the best of our knowledge, we present the first work investigating generic overgeneralization in PLMs.

2 Background

2.1 Genericity and the Generic Overgeneralization Effect

Generics express generalizations about kinds, and lack explicit quantifiers such as *all*, *some* and *most*. Unlike quantified statements, generics do not communicate information about how many members of the kind have the property in question.

Similarly, there is no direct relation between the prevalence of a property among members of a kind and the acceptability of the relevant generic. For example, the generic statement “ducks lay eggs” is accepted even though only mature fertile females lay eggs, but the generic statement “ducks are female” is rejected (Leslie et al., 2011).

If people believe that the statement “ducks lay eggs” is true, they will tend to accept a quantified statement such as “all ducks lay eggs”, because resorting to a default operation saves cognitive effort. This phenomena is called the *generic overgeneralization* (GOG) effect and is defined as “overgeneralizing from the truth of a generic to the truth of the corresponding universal statement” (Leslie et al., 2011, p. 17).

Quantifiers have been shown to influence the GOG effect, but the question of which types of quantified statements are susceptible to overgeneralization has not been resolved yet (Karczewski et al., 2020).

In this paper we focus on minority characteristic generics and majority characteristic generics. Minority characteristic generics include generics such as “lions have manes”, which are only true about a minority of the kind and usually refer to gender-related properties. Conversely, majority characteristic generics include generics such as “tigers have stripes”, which refer to properties that are directly related to the nature of the kind and are prevalent, though not universal, among members of the kind (Prasada et al., 2013). Majority characteristic generics do not need to express exceptionless universal generalizations, since some tigers (e.g., albino tigers) may fail to possess the property.

2.2 Genericity in NLP

Genericity is a key component in the study of human cognition because it demonstrates our inclination to organize our experience of the world into categories, kinds or classes (Lazaridou-Chatzigoga, 2019). The importance of generic language has been recognized in the artificial intelligence and natural language processing community for tasks that involve knowledge acquisition, ontology development, and semantic inference (Monahan et al., 2015; Zhou et al., 2015).

Reiter and Frank (2010) developed a corpus-based supervised learning approach for identifying generic noun phrases in context, using linguistically-motivated features in a Bayesian network classifier. Their experiments were restricted to generic noun phrases, as at the time there were no corpora available that contain annotations for genericity at the sentence level. Friedrich and Pinkal (2015) presented a discourse-sensitive genericity labeler, using Conditional Random Fields as a sequence labeler. Their experiments showed that context information improves accuracy, and their model outperforms the approach proposed by Reiter and Frank (2010).

Govindarajan et al. (2019) proposed a semantic framework for modeling linguistic expressions of generalization, suggesting that such expressions should be captured in a continuous multi-label system, rather than a multi-class system. This was accomplished by decomposing categories such as episodic, habitual, and generic into simple referential properties of predicates and their arguments. The framework was used to construct a dataset covering the full Universal Dependencies English Web Treebank. Furthermore, Govindarajan et al. (2019) presented models for predicting expressions of linguistic generalization, which combine hand-engineered type- and token-level features with static and contextual learned representations.

In summary, although multiple prior works cover genericity in NLP, the generic overgeneralization effect has not yet been investigated specifically.

2.3 Knowledge Enhanced PLMs

Incorporating commonsense knowledge is necessary and beneficial for language inference (LoBue and Yates, 2011; Bowman et al., 2015; Rashkin et al., 2018b), reading comprehension (Mihaylov and Frank, 2018; Rashkin et al., 2018a), and generation based question answering (Chen et al., 2020). Recent research has shown that PLMs do not sufficiently capture factual commonsense world knowledge from the text used in their pre-training (Wang et al., 2021a; Yu et al., 2022; Gong et al., 2020). To address this problem, knowledge embedding methods have been proposed with the aim of encoding the relational facts in knowledge graphs through entity embeddings (Liu et al., 2020; Tang et al., 2020; Dai et al., 2020).

In this paper, we implement KEPLER (Wang et al., 2021b) to inject factual knowledge into PLMs (BERT and RoBERTa) with the aim of reducing the GOG effect. KEPLER jointly optimizes parameters with knowledge embedding and masked language modelling objectives to blend factual knowledge with language representations. The texts and entities are encoded into a unified semantic space using a single PLM encoder. For the knowledge embedding objective, entity descriptions are encoded as entity embeddings and are trained similarly to other knowledge embedding methods such as AutoETER (Niu et al., 2020), which is a knowledge graph embedding framework with automated entity type representation. The masked language modelling objective is implemented using existing approaches such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019b).

3 The GOG Effect Evaluation Task

We introduce the GOG effect evaluation task, describe how the datasets were created, and discuss how the factual knowledge about kinds that can be used for knowledge injection was extracted.

3.1 Task Definition

We introduce two tasks: a classification task, and a masked language modelling task. The classification task evaluates whether a model can classify a statement as generic or non-generic: PLMs can be fine-tuned on this task using the training data provided. However, this task does not evaluate the GOG effect directly.

Secondly, we propose a masked language modelling task to evaluate if PLMs exhibit the GOG effect. We mask the position preceding a generic statement and use the PLM to predict the token. For example, given the generic "lions have manes", we ask a PLM to fill the blank in "`[MASK]` lions have manes". We use mean reciprocal rank (MRR) on the predicted word distribution to evaluate the PLM's ability to fill the masked position; precision at 5 (P@5) measures the relevance of top 5 predicted words. We evaluate whether one of the following universal quantifiers are predicted: *all*, *every*, *most*, *some*, *few* and *many*. Here, we do not consider the truthfulness of the quantified generic statement; that is, although the resulting quantified statement might not be factual, our aim is to evaluate if the PLMs would give a high probability to quantifier tokens when asked to predict the mask token. The higher the rank of the quantifiers, the stronger the PLM exhibits the GOG effect; a very low rank or low precision would indicate that the PLM does not exhibit the GOG effect.

3.2 Task Data

The proposed tasks focus on minority and majority characteristic of kind generics. These are generics that are only true about a minority or majority of a kind. We created a list of animals, which includes reptiles, fish, birds, mammals and amphibians, and used it to sample generic statements from GenericsKB (Bhakthavatsalam et al., 2020). GenericsKB is a large repository of 3.4M standalone generics har-

	Animal Types	**ASCENT KB Triplets**	**Generic Statements**	**Minority Generics**	**Majority Generics**
Amphibians	18	6 783	1 856	256	414
Reptiles	31	18 349	4 266	598	862
Fish	60	38 633	5002	471	788
Birds	76	63 967	10 604	1 051	1 811
Mammals	263	275 101	38 640	3 508	4 875
Total	448	402 833	60 368	5 884	8 750

Table 2: Datasets statistics. The minority and majority characteristic generic statements are used for evaluation and excluded from the other generic statements used for training.

vested from a webcrawl of 1.7B sentences. GenericsKB was constructed by first using a set of rules to identify candidate standalone generic sentences, and then applying a crowdsource-trained BERT classifier to assign a confidence to each generic sentence (Bhakthavatsalam et al., 2020).

Our list contains animals that are present in both GenericsKB and in ASCENT KB; the latter is used to sample factual assertions for each animal (see §3.3). For each animal, we sampled generic sentences with an associated confidence score greater than 0.5. Table 2 shows the statistics of the datasets we constructed. In total, we have three disjoint generic statements datasets.

To construct the minority characteristic generics dataset, we created a list of identifiers to sample generic statements about each animal from GenericsKB: *female*, *male*, *infant*, *young*, *adult*, *mature*, and *old*. Furthermore, we sampled generic statements from GenericsKB that have existential quantifiers *some* and *few*. In the final dataset, we removed the identifier words and the quantifiers from all generic sentences. For example, a generic statement such as "male lions have manes" is stored in the minority characteristic generics dataset as "lions have manes".

We similarly constructed the majority characteristic generics dataset by sampling quantified generic statements from GenericsKB that had these quantifiers: *all*, *many*, *every*, and *most*. Quantifiers were also removed from the generic statements before adding them to the dataset. For example, a generic statement such as "all zebras have different stripes" is stored in the majority characteristic generics dataset as "zebras have different stripes".

The third dataset consists of other types of generic statements about the animals in our list. We use this dataset to train models for the classification task and to further pretrain the language models, while the gold minority and majority characteristic generics datasets were used for evaluation and knowledge probing.

Additionally, we created a dataset of non-generic statements for training the classifiers by sampling sentences with a confidence score of les that 0.3 from GenericsKB. This dataset contains statements such as "a pitbull mauled a child" and "the snake laid some eggs". These statements are not generics because they apply only to a specific individual member of a kind.

3.3 Commonsense Knowledge Data

In order to perform PLM knowledge injection using KEPLER we need factual knowledge in the form of <*Subject, Predicate, Object*> triplets, and textual description data for each subject in a given triplet. We use the list of animals to sample SPO triplets from ASCENT KB (Nguyen et al., 2021) as a source of factual knowledge. For each animal in our list, we webcrawled A-Z-Animals.com[2] and scrapped textual data that include description like classification and evolution, anatomy and appearance, distribution and habitat, behaviour and lifestyle, reproduction and life cycles, and diet and prey. This information is then used as entity descriptions for each subject in the SPO triplets. For each SPO triplet, we align the textual description data with each subject in the triplet. For example, given a triplet <*elephant, uses, its trunk*>, we align the textual data about Elephants crawled from A-Z-Animals.com. Table 2 shows the number of SPO triplets extracted from ASCENT KB for each of the different types of animals in our list.

The factual data includes general information about the kinds, and makes exceptions to generic

[2] https://a-z-animals.com/animals/

statements salient. For example, existence of albino tigers implies that not all tigers have stripes. Therefore we hypothesize that the factual knowledge could be used to reduce the GOG effect in language models with respect to universal majority and minority characteristic generics. For majority characteristic generics, the factual data includes information about differences between sub-kinds of a given kind, such as the color of fur and type of food. For minor characteristic, it contains knowledge that emphasizes gender differences such as the different sizes or different roles of males and females.

4 Task Results

4.1 Generics Classification Task

First we train and evaluate models to classify statements as generics or not. As a baseline, we train a bi-directional LSTM on the generic and non-generic statements training dataset (which excludes the minority and majority characteristic generics). We use the base versions of BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019b) as PLMs and fine-tune them on the same classification task.

For the classification task, we had **60k** generic statements, and sampled **50k** non-generic sentences from GenericsKB, which resulted in 110k training sentences for the classification task. From these, we used 20% of the sentences for testing and then used the held out minority and majority characteristic generics data for evaluation.

We report the F1-score and accuracy of the generics classification in Table 3, evaluated using the minority generics dataset and majority generics dataset. The results show that, as expected, the PLMs outperformed our baseline on the classification task.

4.2 The GOG Effect Evaluation Task

In this evaluation, we use the generics training data (not include minority and majority characteristic generics) to further pretrain the PLMs. The masked token prediction task is used to determine if the PLMs exhibit the GOG effect. We implement KEPLER (Wang et al., 2021b) to inject factual knowledge into the PLMs, using the data extracted from ASCENT KB (§3.3). The presence of quantifiers in the highest-ranked words that the PLMs predict in the masked positions would indicate that the PLMs exhibit the GOG effect. For each minority or majority characteristic generic statement in our datasets, we evaluate whether one of the following universal quantifiers are predicted: *all*, *every*, *most*, *some*, *few* and *many*. We conduct separate experiments for minority characteristic generics and majority characteristic generics in order to demonstrate which type of generics is more susceptible to overgeneralization bias.

The results in Tables 4 and 5 show that the MRR and P@5 scores are considerably lower after knowledge injection for both BERT and RoBERTa. This means that knowledge injection with KEPLER decreases the likelihood of predicting quantifiers, and therefore reduces the GOG effect. The scores of majority characteristic generics are higher than those of minority characteristic generics, indicating that majority characteristic generics are more susceptible to overgeneralization effect.

However the injection of factual knowledge results in a bigger reducing in the GOG effect for majority characteristics, such that the overgeneralization scores on minority and majority characteristics are very similar after knowledge injection. We postulate that this could be due to the quantity of factual knowledge that made exceptions more salient for majority characteristic generics. That is, a higher number of triples sampled from ASCENT KB contain factual knowledge about majority characteristic of kinds than factual knowledge about minority characteristic of kinds. Thus, more knowledge about majority characteristic of kinds was injected in the PLMs compared to knowledge about minority characteristic of kinds.

As a qualitative example, we asked BERT and RoBERTa to fill in the mask in the statement "`[MASK]` lions have manes" before and after knowledge injection (Table 1). Without knowledge injection, both models exhibits a preference for universal quantifiers, although RoBERTa ranks the conditional quantifier *some* highest, which suggests less overgeneralization than BERT which ranks *all* at the top. After knowledge injection the top three words are no longer quantifiers, which shows that overgeneralization is reduced. However, the presence of *most* among the top 5 predictions indicates that the GOG effect has not been eliminated completely in either model.

5 Probing the Injected Knowledge

The results reported in the previous section warrant us to probe the injected knowledge in order to determine if the PLMs "understand" the injected

	Minority Generics		Majority Generics	
Model	**F1**	**Accuracy**	**F1**	**Accuracy**
Bi-LSTM	0.80	0.83	0.82	0.86
BERT	0.88	0.90	0.89	0.91
RoBERTa	0.90	0.93	0.92	0.95

Table 3: Results of the generics classification task, comparing the PLMs (BERT and RoBERTa) against a Bi-LSTM baseline on minority and majority characteristic generics datasets.

	PLM		PLM+KEPLER	
Model	**MRR**	**P@5**	**MRR**	**P@5**
BERT	0.326	0.305	0.137	0.106
RoBERTa	0.329	0.307	0.135	0.108

Table 4: Mean Reciprocal Rank (MRR) and Precision at 5 (P@5) of universal quantifiers on the GOG effect evaluation task for minority characteristic generics. We report the scores before and after injecting factual knowledge into the PLMs with KEPLER. Lower scores indicate less overgeneralization.

	PLM		PLM+KEPLER	
Model	**MRR**	**P@5**	**MRR**	**P@5**
BERT	0.337	0.318	0.138	0.109
RoBERTa	0.428	0.411	0.152	0.117

Table 5: Mean Reciprocal Rank (MRR) and Precision at 5 (P@5) of universal quantifiers on the GOG effect evaluation task for majority characteristic generics. We report the scores before and after injecting factual knowledge into the PLMs with KEPLER. Lower scores indicate less overgeneralization.

factual knowledge. For example, do the PLMs understand that it is only mature, male lions that can have manes? Furthermore, do PLMs, with factual knowledge, correctly predict relevant tokens that could make quantified generic statements true?

5.1 Quantified Statement Classification

We fine-tune the knowledge-enhanced PLMs on the generics classification task (§4.1) and test if *quantified statements* are classified as generics. We quantify the minority characteristic generics with the quantifiers *many* and *most*, and the majority characteristic generics with *few* and *some*. This allows us to falsify the generics in both datasets. For example, the statement "most lions have manes" is not a true generic statement because only a minority of lions have manes. Similarly, "few tigers have stripes" is also not a true generic statement because most tigers do have stripes. Although the classifiers were trained to classify if a given statement is a generic, we aim to evaluate if the PLMs can use the injected knowledge to resolve that the falsified generics are wrongly quantified statements and should be classified as non-generic. This is because the injected knowledge has factual information that should contradict the falsified generics.

Table 6 reports the accuracy of zero-shot classification of universally quantified statements as non-generics, before and after knowledge injection. Knowledge injection almost doubles classification accuracy, but all the models still overwhelmingly predict that the statements are true generics. Knowledge injection leads to a bigger (absolute) improvement in accuracy for majority characteristic generics than for minority characteristic generics.

Based on this result, we postulate that the PLMs do not understand that the quantifier *all* in a noun phrase such as *all lions* implies *male* + *female* lions. This is made evident by the presence of quantifiers when asking to fill in the blank for a generic sentence such as "`[MASK]` lions have manes". We sampled factual knowledge from ASCENT KB that emphasizes minority and majority characteristic of kinds. This includes assertions such as "male lions have manes"; therefore, natural language inference should lead to the conclusion that "all lions have manes" cannot be a true generic statement because the factual knowledge emphasized gender differences, thus making exceptions more salient.

5.2 GOG Effect Evaluation Probing

We extend the evaluation of the effect of PLM knowledge injection on the GOG effect (§4.2) to probe whether knowledge injection enables the model to distinguish between which quantifiers make a minority or majority characteristic generics true and which quantifiers make them false. For ex-

	Quantified Minority Generics		Quantified Majority Generics	
Model	**PLM**	**PLM+KEPLER**	**PLM**	**PLM+KEPLER**
BERT	0.083	0.14	0.10	0.18
RoBERTa	0.064	0.12	0.081	0.19

Table 6: Accuracy of classifying universally quantified versions of minority and majority generic statements in the test set as false, using the PLM generics classifiers, before and after injecting factual knowledge with KEPLER.

#	BERT	RoBERTa
1	Eyes	Heads
2	Manes	Eyes
3	Heads	Manes
4	Teeth	Tails
5	Tails	Teeth

Table 7: The top 5 words predicted by BERT and RoBERTa, with injected factual knowledge, for filling the mask in the generic "most lions have [MASK]".

#	BERT	RoBERTa
1	Mountain	Male
2	Female	Female
3	White	Mountain
4	All	Young
5	Young	All

Table 8: The top 5 words predicted by BERT and RoBERTa, with injected factual knowledge, for filling the mask in the generic "[MASK] lions are animals".

ample, the token "stripes" should be ranked among the top tokens that the PLM predict when asked to fill in the mask token in the statement "most tigers have [MASK]". On the other hand, the PLM should *not* predict the token "stripes" when asked to fill in the blank for the statement 'few tigers have [MASK]".

For this probing task, we evaluate statements using four quantifiers: *few, many, most*, and *some*. We quantify both minority and majority characteristic generics and mask the final token in the statement (corresponding to the object or predicative complement). We generate probing datasets using the template: *quantifier* + (generic statement - final token) + [MASK].

We report the mean reciprocal rank of both minority and majority characteristic generics for each quantifier. If the model successfully learns how to interpret each quantifier, the masked final tokens should be ranked higher together with a quantifier that makes the generic statement true and lower with the quantifier that make the generic statement false. Table 9 shows the results of this probing task.

The results show that for minority characteristic generics the PLMs correctly assign a higher masked token MRR to true statements quantified by *few* or *some* than to statements quantified by *many* or *most*. Conversely, for majority characteristic generics the PLMs also correctly assign higher MRR when statements are quantified by *many* or *most* instead of with *few* or *some*. However, statements with the quantifier *some* are still ranked relatively high, indicating that the PLMs struggle more to interpret that quantifier correctly.

The MRR for masked tokens in true statements is higher for majority characteristics than minority characteristics, but the MRR for false statements is relatively lower on minority characteristics. Despite learning the distinction between different kinds of quantifiers, the MRR across the models and quantifiers is arguably still too high with quantifiers that falsify a generic statement.

As an example, Table 7 shows the tokens predicted for the statement "most lions have [MASK]". Here we expect the original token, "manes", not to feature among the top predicted tokens for the quantifier *most* because the injected factual data should make salient the knowledge that it is only a minority population of lions that have manes. However the two PLMs still rank "manes" as second and third most likely token, respectively. In contrast, when the first token in the statement is masked, i.e., "[MASK] lions are animals", the only quantifier in the top 5 is *all*, at position 4 and 5.

6 Conclusion

We investigated the generic overgeneralization (GOG) effect in PLMs and demonstrated that PLMs do overgeneralize and treat quantified statements as if they were generics. We introduced datasets on minority and majority characteristic generics that

	Minority Generics with Quantifier				Majority Generics with Quantifier			
Model	**Few**	**Some**	Many	Most	Few	Some	**Many**	**Most**
BERT	0.58	0.69	0.43	0.45	0.51	0.65	0.71	0.74
RoBERTa	0.61	0.70	0.38	0.40	0.49	0.63	0.76	0.80

Table 9: Mean Reciprocal Rank (MRR) scores of masked final tokens using PLMs with knowledge injection under different quantifiers. Scores indicate how each model perform on the probing task for each quantifier when applied to minority and majority characteristic generics.

can be used to evaluate the GOG effect, as well as a source of factual knowledge about kinds to evaluate PLM knowledge embedding methods. Our results suggest that knowledge injection reduces the GOG effect in PLMs but does not eliminate it, and that majority characteristic generic statements are more susceptible to overgeneralization bias. Probing the models after knowledge injection, we were able to determine which quantifiers make minority or majority characteristic generics to remain as true quantified generic statements and which quantifiers make the generics to become non-generic statements.

Our paper makes the case for future research on methods for injecting commonsense into PLMs more effectively so that they can perform better natural language inference based on the knowledge presented. This would be an important step towards advancing commonsense reasoning in PLMs.

Acknowledgement

This work is based on research supported in part by the National Research Foundation of South Africa (Grant Number: 129850) and the Hasso Plattner Institute for Digital Engineering through the HPI Research School at the University of Cape Town.

References

Sumithra Bhakthavatsalam, Chloe Anastasiades, and Peter Clark. 2020. GenericsKB: A knowledge base of generic statements. *CoRR*, abs/2005.00660.

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, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 632–642. The Association for Computational Linguistics.

Yi-Jyun Chen, Ching-Yu Helen Yang, and Jason S. Chang. 2020. Improving phrase translation based on sentence alignment of Chinese-English parallel corpus. In *Proceedings of the 32nd Conference on Computational Linguistics and Speech Processing (ROCLING 2020)*, pages 6–7, Taipei, Taiwan. The Association for Computational Linguistics and Chinese Language Processing (ACLCLP).

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy. Association for Computational Linguistics.

Yuanfei Dai, Shiping Wang, Neal N. Xiong, and Wenzhong Guo. 2020. A survey on knowledge graph embedding: Approaches, applications and benchmarks. *Electronics*, 9(5).

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.

Annemarie Friedrich and Manfred Pinkal. 2015. Discourse-sensitive automatic identification of generic expressions. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 1272–1281. The Association for Computer Linguistics.

Peizhu Gong, Jin Liu, Yihe Yang, and Huihua He. 2020. Towards knowledge enhanced language model for machine reading comprehension. *IEEE Access*, 8:224837–224851.

Venkata Subrahmanyan Govindarajan, Benjamin Van Durme, and Aaron Steven White. 2019. Decomposing generalization: Models of generic, habitual and episodic statements. *Trans. Assoc. Comput. Linguistics*, 7:501–517.

Michelle A. Hollander, Susan A. Gelman, and Jon Star. 2002. Children's interpretation of generic noun phrases. *Developmental Psychology*, 38(6):883–894.

Daniel Karczewski, Edyta Wajda, and Radosław Poniat. 2020. Do all storks fly to africa? universal statements and the generic overgeneralization effect. *Lingua*, 246:102855.

Sangeet Khemlani, Sarah-Jane Leslie, Sam Glucksberg, and Paula Rubio Fernandez. 2007. Do ducks lay eggs? How people interpret generic assertions. *Proceedings of the Annual Meeting of the Cognitive Science Society*, 29.

Dimitra Lazaridou-Chatzigoga. 2019. Genericity. In Chris Cummins and Napoleon Katsos, editors, *The Oxford Handbook of Experimental Semantics and Pragmatics*, pages 155–177. Oxford University Press.

Sarah-Jane Leslie, Sangeet Khemlani, and Sam Glucksberg. 2011. Do all ducks lay eggs? The generic overgeneralization effect. *Journal of Memory and Language*, 65(1):15–31.

Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019a. Linguistic knowledge and transferability of contextual 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 1073–1094, Minneapolis, Minnesota. Association for Computational Linguistics.

Y. Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, M. Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. RoBERTa: A robustly optimized BERT pretraining approach. *ArXiv*, abs/1907.11692.

Yuzhang Liu, Peng Wang, Yingtai Li, Yizhan Shao, and Zhongkai Xu. 2020. AprilE: Attention with pseudo residual connection for knowledge graph embedding. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 508–518, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Peter LoBue and Alexander Yates. 2011. Types of common-sense knowledge needed for recognizing textual entailment. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA - Short Papers*, pages 329–334. The Association for Computer Linguistics.

Todor Mihaylov and Anette Frank. 2018. Knowledgeable reader: Enhancing cloze-style reading comprehension with external commonsense knowledge. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 821–832. Association for Computational Linguistics.

Sean Monahan, Michael Mohler, Marc T. Tomlinson, Amy Book, Maxim Gorelkin, Kevin Crosby, and Mary Brunson. 2015. Populating a knowledge base with information about events. *Theory and Applications of Categories*.

Tuan-Phong Nguyen, Simon Razniewski, and Gerhard Weikum. 2021. Advanced semantics for commonsense knowledge extraction. In *WWW '21: Proceedings of the Web Conference 2021*.

Guanglin Niu, Bo Li, Yongfei Zhang, Shiliang Pu, and Jingyang Li. 2020. AutoETER: Automated entity type representation for knowledge graph embedding. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1172–1181, Online. Association for Computational Linguistics.

Sandeep Prasada, Sangeet Khemlani, Sarah-Jane Leslie, and Sam Glucksberg. 2013. Conceptual distinctions amongst generics. *Cognition*, 126(3):405–422.

Hannah Rashkin, Antoine Bosselut, Maarten Sap, Kevin Knight, and Yejin Choi. 2018a. Modeling naive psychology of characters in simple commonsense stories. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 2289–2299. Association for Computational Linguistics.

Hannah Rashkin, Maarten Sap, Emily Allaway, Noah A. Smith, and Yejin Choi. 2018b. Event2mind: Commonsense inference on events, intents, and reactions. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 463–473. Association for Computational Linguistics.

Nils Reiter and Anette Frank. 2010. Identifying generic noun phrases. In *ACL 2010, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, July 11-16, 2010, Uppsala, Sweden*, pages 40–49. The Association for Computer Linguistics.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, 8:842–866.

Yun Tang, Jing Huang, Guangtao Wang, Xiaodong He, and Bowen Zhou. 2020. Orthogonal relation transforms with graph context modeling for knowledge graph embedding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2713–2722, Online. Association for Computational Linguistics.

Ruize Wang, Duyu Tang, Nan Duan, Zhongyu Wei, Xuanjing Huang, Jianshu Ji, Guihong Cao, Daxin Jiang, and Ming Zhou. 2021a. K-Adapter: Infusing Knowledge into Pre-Trained Models with Adapters. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1405–1418, Online. Association for Computational Linguistics.

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li, and Jian Tang. 2021b. KEPLER: A Unified Model for Knowledge Embedding and Pre-trained Language Representation. *Transactions of the Association for Computational Linguistics*, 9:176–194.

Donghan Yu, Chenguang Zhu, Yiming Yang, and Michael Zeng. 2022. Jaket: Joint pre-training of knowledge graph and language understanding. In *AAAI 2022*.

Mengfei Zhou, Anette Frank, Annemarie Friedrich, and Alexis Palmer. 2015. Semantically enriched models for modal sense classification. In *LSD-Sem@EMNLP*.