Data and Model Distillation as a Solution for Domain-transferable Fact Verification

Mitch Paul Mithun, Sandeep Suntwal, Mihai Surdeanu
University of Arizona, Tucson, Arizona, USA
{mithunpaul,sandeepsuntwal,msurdeanu}@email.arizona.edu

Abstract

While neural networks produce state-of-the-art performance in several NLP tasks, they generally depend heavily on lexicalized information, which transfer poorly between domains. We present a combination of two strategies to mitigate this dependence on lexicalized information in fact verification tasks. We present a data distillation technique for delexicalization, which we then combine with a model distillation method to prevent aggressive data distillation. We show that by using our solution, not only does the performance of an existing state-of-the-art model remain at par with that of the model trained on a fully lexicalized data, but it also performs better than it when tested out of domain. We show that the technique we present encourages models to extract transferable facts from a given fact verification dataset.

1 Introduction

Neural networks have matched, and in several cases even surpassed, human performance in several supervised learning problems. However, such successes come at a cost. These neural networks typically need a great deal of human support in the form of man power required for curating domain specific datasets. Further, it has been shown (Gururangan et al., 2018; Poliak et al., 2018; Thorne and Vlachos, 2020) that several such models depend heavily on certain statistical nuances found in these datasets, information that transfers poorly between domains. The ideal solution to this problem is the creation of models that do not rely on such statistical nuances in the given datasets, but instead encode the true underlying semantics of the task, that are in turn transferable to other domains.

Fact verification is the task of verifying the truthfulness of claims by estimating their assertions against credible evidences. Specifically, given a pair of claim and evidence statements, they have to be classified into one of the 3 class labels, *agree*, *disagree*, or *neutral*. Fact verification datasets, which often constitute real life news articles, have the added advantage of being used in practical problems such as fake news detection. More recently, several neural network models (Nie et al., 2020; Liu et al., 2020, inter alia) built on top of the transformers (Vaswani et al., 2017), have achieved excellent performance in fact verification tasks.

However these methods are not devoid of the shortcomings that besiege other neural networks in natural language processing tasks. It has been shown that these approaches depend heavily on lexical artifacts that transfer poorly between domains (Panenghat et al., 2020; Karimi Mahabadi et al., 2020; Schuster et al., 2019). For example, Suntwal et al. (2019) observed that out of all the statements containing the phrase 'American Author' in the FEVER dataset (Thorne et al., 2018), 91% of them belonged to one class label. Further, they demonstrated that neural methods put unnecessary emphasis on such lexical artifacts, which limits their transfer to other fact verification datasets such as the Fake News Challenge (FNC) (Pomerleau and Rao, 2017).

To mitigate the dependency on such artifacts, Suntwal et al. (2019) proposed a *data distillation* (or delexicalization) approach, which replaces some lexical artifacts such as named entities with their type and a unique id to indicate occurrence of the same artifact in claim and evidence. While promising, the risk of this direction is discarding too much information through the delexicalization process. For example, replacing *China* with its named entity (NE) type (`COUNTRY`) in an evidence sentence discards the fact that the text is about an Asian country which might be relevant in the context.

In this work we propose a solution that combines data distillation with *model distillation* to reduce the risk of over delexicalization. In particular, we introduce a teacher-student architecture inspired

Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 4546–4552
June 6–11, 2021.

from that of (Tarvainen and Valpola, 2017). In our architecture, the student model is trained on delexicalized data (to take advantage of data distillation), but is also guided by a teacher trained on the original lexicalized data (as a form of model distillation) to mitigate the possibility of discarding too much lexical information. The contributions of our work are as follows:

(1) To our knowledge, we are the first to explore the combination of data and model distillation as a strategy to improve domain transfer of fact verification methods. Note that while our training process is more costly due to the combination of the student and teacher models, the output is a single individual model (the student), which has the same runtime cost as an individual classifier. Further, our approach is classifier agnostic, and can be coupled with any fact verification method.

(2) We investigate the domain transfer of our method between two fact verification tasks (FNC and FEVER), where we train on one and test on the other. For these experiments we couple our method with the state of the art fact verification approach based on transformers (Vaswani et al., 2017). Our results indicate that our method achieves a cross-domain accuracy of 73.17% in one of the experiments and 74.58% in the other, outperforming other methods that do not use the data distillation-model distillation combination.

All the software for our proposed approach is open-source and publicly available on GitHub at: `https://github.com/clulab/releases/tree/master/naacl2021-student-teacher`.

2 Methodology

2.1 Data distillation

Suntwal et al. (2019) demonstrated that named entities are most prone to overfitting for fact verification. Based on this observation, we also replace named entities with their type (and a unique id). However, unlike their work, we have observed in early experiments that more fine-grained NE types yield better models. In particular, we utilize the FIGER named entity recognizer (NER) (Ling and Weld, 2012) to detect and replace named entities with their most specific label returned by the NER. Further, we also process the text with the CoreNLP NER (Manning et al., 2014) to delexicalize additional NER classes not covered by FIGER. We include in this list mentions of date, time, money, number, and ordinal.

	Claim	Evidence
Plain text	Mark Zuckerberg made the Forbes list of The World's Most Powerful People	In December 2016, Zuckerberg was ranked 10th on Forbes list of The World's Most Powerful People.
Distilled text	personC1 made the Forbes list of written_workC1's Most Powerful People .	In December 2016, personC1 was ranked 10th on Forbes list of written_workC1 's Most Powerful People.

Table 1: The claim and evidence before and after the data distillation process.

Next, we align the named entities between the claim and the evidence. That is, any named entity that appears first in the claim is assigned an id postfixed with #Cn; if an entity mention appears only in evidence then it is postfixed with #En, where C indicates that the entity appeared first in the claim, E indicates that the entity first appeared in the evidence, and n indicates the n_{th} observed entity. Table 1 shows an example output for this data distillation process.

2.2 Model distillation

We propose a model distillation strategy to mitigate the risk of overly aggressive data distillation. In particular, we introduce a teacher-student architecture (shown in figure 1) (Hinton et al., 2015; Tarvainen and Valpola, 2017; Laine and Aila, 2016; Sajjadi et al., 2016), where the teacher is trained on the original, lexicalized data, and the student is trained on the data delexicalized with the approach described in the previous sub-section.

The intuition behind our model distillation approach is that the proposed teacher model will "pull" the student model towards the original underlying semantics, which are partially obscured to the student due to the delexicalization of its training data. More formally, this is captured through a consistency loss that minimizes the difference in predicted label distributions between the student and the teacher. The consistency loss is implemented as a mean squared error between the label scores predicted by the student and the teacher. Additionally, both the student and the teacher components include a regular classification loss on their respective data, which is implemented using cross entropy. This encourages both the student and the teacher to

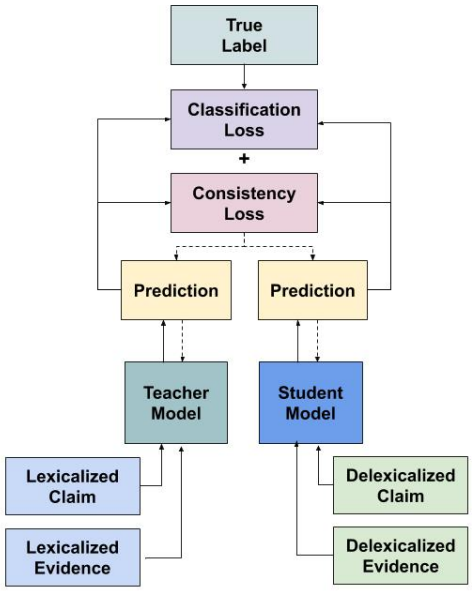

Figure 1: The teacher-student architecture for model distillation.

learn as much as possible from their own views of the data.

2.3 Classifiers

We experiment with a state-of-the-art method for fact verification, transformers (Vaswani et al., 2017), which has achieved state-of-the-art results not only in the task of fact verification but in several other NLP tasks. Specifically, we use the PyTorch implementation of BERT (Devlin et al., 2019) from huggingface (Wolf et al., 2019).

We experimented with several pre-trained BERT-base models and found that the one which gave the highest performance was the BERT-cased model when used with a sequence length of 128. Further, to distinguish the vocabulary of the delexicalized data from the lexicalized data we augment the base vocabulary of BERT with tokens specific to the delexicalized data. For example, as mentioned before, during delexicalization we use `personC1` to denote the first occurence of the named entity in the claim paragraph. However, to ensure that the BERT BasicTokenizer does split `personC1` into `person` and `C1`, we added the token "C1" to the BERT vocabulary. Tokenizers for each of the lexicalized and delexicalized dataset are initially created using BERT BasicTokenizer, but then use the aforementioned vocabulary created for the specific data type.

3 Experiments

3.1 Data

We use two distinct fact verification datasets for our experiments, FEVER (Thorne et al., 2018) and FNC (Pomerleau and Rao, 2017).

The Fact Extraction and Verification (FEVER) dataset: This dataset consists of 145,449 data points each having a claim and evidence pair. These claim-evidence pairs typically contain one or more sentences compiled from Wikipedia using an information retrieval (IR) module and are classified into three classes: *supports, refutes* and *not enough info*. The evidence for data points that had the gold label of *not enough info* were retrieved (using a task-provided IR component) either by finding the nearest neighbor to the claim or randomly. Even though the training partition of the FEVER dataset was publicly released, the gold test labels used in the final shared task were not. We therefore built our own test partition by dividing the randomized training partition into 80% (119,197 data points) and 20% (26,252 data points).

The Fake News Challenge (FNC) dataset: This dataset comprises claim-evidence pairs that were divided into four classes, *agree, disagree, discuss* and *unrelated*. These claim-evidence pairs were created using the headlines and content section of real news articles respectively. While the training partition of the publicly available dataset comprised 49,972 data points, the testing partition had 25,413 data points. We further divided the training partition into 40,904 data points for training and 9,068 data points for development.

Cross-domain labels: In order to evaluate the proposed methods in a cross-domain setting, we modified the label space of the source domain to match that of the target domain. In particular, when training on FEVER and testing on FNC, the data points in FEVER that belong to the class *supports* were relabeled as *agree*, and those in *refutes* as *disagree*. Further, the data points belonging to the third class *not enough info* (NEI) were divided into *discuss* and *unrelated*. Specifically, of all the claim-evidence pairs that belonged to the NEI class, the ones whose evidences were retrieved using the nearest neighbor technique component of FEVER, were labeled to now belong to the *discuss* class since they were more likely to be topically relevant to the claim. The rest were assigned the label *unrelated*. Similarly in the other direction, i.e., when

	Configuration			
Train Domain Eval Domain	FEVER FEVER	FEVER FNC	FNC FNC	FNC FEVER
BERT Lex	94.15%	68.93%	96.39%	73.21%
BERT Delex (OA-NER)	82.31%	53.59%	65.85%	46.47%
BERT Delex (OA-NER + SS)	75.26%	46.71%	45.51%	51.77%
BERT Delex (FIGER)	91.97%	54.27%	96.22%	62.99%
BERT TS (FIGER)	89.42%	**73.14%***	98.89%	**74.58%***

Table 2: In-domain and cross-domain accuracies for various methods. All scores reported are averaged across three random seeds. "BERT Lex" is the stand alone model trained on the original lexicalized data; "BERT Delex" is the standalone model trained on delexicalized data. OA-NER delexicalizes the data using the Overlap Aware Named Entity Recognizer; SS uses Super Sense tags — two delexicalization techniques mentioned in Suntwal et al. (2019). FIGER delexicalizes the data using a fine-grained named entity recognizer (Ling and Weld, 2012). ; and "BERT TS" denotes the student in the proposed teacher-student architecture. * indicates that the corresponding result is significantly better than its baseline ("BERT lex" in the same column), under a bootstrap resampling test with 1,000 samples, and p-value < 0.035.

training on FNC and testing on FEVER, the data points that had the labels of *discuss* and *unrelated* were combined and given the label of *not enough info*.

3.2 Settings

In all the experiments, the performance of the underlying model on the respective lexicalized data is considered as the baseline. For example when training a teacher-student model on FEVER, the baseline is the model that was trained using the original text of the FEVER dataset. In the baseline model, we use the default hyper parameters set in the huggingface repository (Wolf et al., 2019).

We focus our analysis on cross-domain evaluation, i.e., we train all models on one dataset (e.g., FEVER) and evaluate their accuracy on the other dataset (e.g., FNC).

3.3 Results

Table 2 summarizes the results of our experiments with various models tested in-domain and cross-domain. All scores reported are averaged across three random seeds. We use 'BERT Lex' as the baseline model which is the stand alone model trained on the original lexicalized data. 'BERT Delex' denotes the standalone models trained on delexicalized data, along with the corresponding delexicalization techniques used. OA-NER uses the Overlap Aware Named Entity Recognizer for delexicalization of data and SS uses Super Sense tags (Suntwal et al., 2019). FIGER delexicalizes the data using a fine-grained named entity recognizer (Ling and Weld, 2012). 'BERT TS' denotes the student in the proposed teacher-student architecture. Since the delexicalization used by the best performing 'BERT Delex' models in the cross-domain setting was FIGER, we chose it as the preferred delexicalization technique for this student.

Note that the lexicalized models, which perform well in-domain, tend to transfer poorly to a new domain. For example, the BERT model trained on lexicalized FEVER data, gave an accuracy of 94.15% when tested on FEVER, but reduced to 68.93% when tested on FNC. This verifies our findings that the signal the model learns from unmasked text does not generalize well.

In contrast, in all our experiments, the student models trained under the teacher-student architecture outperform the other models trained using lexicalized data, in a cross-domain setting. For example, the student model of the teacher-student architecture trained on FEVER, gave an accuracy of 89.42% when tested on FEVER and an accuracy of 73.14% when tested on FNC. Similarly in the other direction, when the same model was trained on FNC, it gave an accuracy of 98.89% when tested on FNC, and an accuracy of 74.58% when tested on FEVER. Note that in both the directions the accuracy of the student model of the teacher-student architecture surpasses the corresponding accuracy of the model trained on lexicalized data in a cross-domain setting. These experiments were repeated under a bootstrap resampling test with 1,000 samples, and p-value < 0.035 to ensure statistical significance.

3.4 Discussion

We believe that the improved performance of the student model in the TS architecture is due to the

fact that the TS architecture provides additional information over the ground labels. The key addition of our TS approach is that the delexicalized student learns to mimic the label probability distributions of the teacher through the consistency loss. As discussed earlier, we conjecture that this pulls the student model closer to the teacher. Another possible interpretation is that the model distillation has a regularization effect since the consistency loss essentially averages the behavior of both models.

Importantly, our results indicate that too much delexicalization risks discarding useful information. We believe this is why the standalone delexicalized model performs worse out of domain, and why the TS delexicalized student performs better. Understanding *how much* delexicalization to apply given a task opens up interesting avenues for future research. Nevertheless, overall this paper demonstrates that data distillation and model distillation can be combined as a strategy to improve domain transfer of fact verification methods.

Lex	TS Student
Apple	year
year	said
Rivers	`country`
said	`person`
Islamic	according
State	`organization`
according	news
says	Islamic
Watch	`engineer`
report	`actor`

Table 3: Top 10 tokens with the highest attention weights by each of the trained models. 'Lex' is the stand alone model trained on the original lexicalized data and 'TS Student' denotes the student in the proposed teacher-student architecture.

Lastly, we also inspected the word-level attention weights (Bahdanau et al., 2014) to further understand what these models are learning. Specifically, we analyze the weights assigned by the last attention head in the last layer of the respective transformer models. Table 3 shows the tokens that were assigned highest weights by the model trained on lexicalized data and the teacher-student model.[1] It can be seen that the tokens that were given the highest weights by the model trained on lexicalized data contain more named entities (e.g., *Apple*, *State*). This suggests potential overfitting, since the specific named entities should not be relevant for the fact verification task.

On the other hand, the tokens that were given the highest weights by the teacher-student model contain more generic named entity labels (e.g., country, person). Also we found that out of all the attention weights assigned by the model trained on lexicalized data, 15.60% were given to named entities. Further, in the TS student model only 7.44% was assigned to named entity labels. These findings demonstrate that by using the data distillation and model distillation techniques we are able to reduce the importance that models place on lexical artifacts. This not only helps them achieve accuracies at par with their counterparts trained on plain text data in an in-domain setting, but also outperform them in a cross-domain setting.

4 Conclusion

We present a new strategy to improve domain transfer of fact verification methods, which combines data distillation and model distillation. We show that the performance of existing state-of-the-art models degrades significantly on a cross-domain setting, hence motivating the necessity of robust data distillation techniques such as delexicalization to minimize overfitting on lexical artifacts. We further combine delexicalization with a teacher-student architecture as a form of model distillation to reduce the risk of over-delexicalization. We hope that this solution will encourage the development of architectures capable of reducing the dependency of models on lexical artifacts in an effort to learn domain transferable knowledge in the task of fact verification.

Acknowledgments

This work was supported by the Defense Advanced Research Projects Agency (DARPA) under the World Modelers program, grant number W911NF1810014, and by the Bill and Melinda Gates Foundation HBGDki Initiative. Mihai Surdeanu declares a financial interest in lum.ai. This interest has been properly disclosed to the University of Arizona Institutional Review Committee and is managed in accordance with its conflict of interest policies. The authors would also like to thank Becky Sharp and Marco Valenzuela-Escárcega for all their valuable comments and reviews.

[1] Stop words and other BERT specific tokens like [SEP], [CLS], [PAD], etc., are removed from this list.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

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.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. 2018. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112, New Orleans, Louisiana. Association for Computational Linguistics.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Rabeeh Karimi Mahabadi, Yonatan Belinkov, and James Henderson. 2020. End-to-end bias mitigation by modelling biases in corpora. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8706–8716, Online. Association for Computational Linguistics.

Samuli Laine and Timo Aila. 2016. Temporal ensembling for semi-supervised learning. *arXiv preprint arXiv:1610.02242*.

Xiao Ling and Daniel S Weld. 2012. Fine-grained entity recognition. In *AAAI*, volume 12, pages 94–100.

Xiaodong Liu, Yu Wang, Jianshu Ji, Hao Cheng, Xueyun Zhu, Emmanuel Awa, Pengcheng He, Weizhu Chen, Hoifung Poon, Guihong Cao, and Jianfeng Gao. 2020. The Microsoft toolkit of multi-task deep neural networks for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 118–126, Online. Association for Computational Linguistics.

Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60.

Yixin Nie, Lisa Bauer, and Mohit Bansal. 2020. Simple compounded-label training for fact extraction and verification. In *Proceedings of the Third Workshop on Fact Extraction and VERification (FEVER)*, pages 1–7, Online. Association for Computational Linguistics.

Mithun Paul Panenghat, Sandeep Suntwal, Faiz Rafique, Rebecca Sharp, and Mihai Surdeanu. 2020. Towards the necessity for debiasing natural language inference datasets. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 6883–6888.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018. Hypothesis only baselines in natural language inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 180–191, New Orleans, Louisiana. Association for Computational Linguistics.

Dean Pomerleau and Delip Rao. 2017. Fake news challenge.

Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. 2016. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. *Advances in neural information processing systems*, 29:1163–1171.

Tal Schuster, Darsh Shah, Yun Jie Serene Yeo, Daniel Roberto Filizzola Ortiz, Enrico Santus, and Regina Barzilay. 2019. Towards debiasing fact verification models. 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 3419–3425, Hong Kong, China. Association for Computational Linguistics.

Sandeep Suntwal, Mithun Paul, Rebecca Sharp, and Mihai Surdeanu. 2019. On the importance of delexicalization for fact verification. 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 3413–3418, Hong Kong, China. Association for Computational Linguistics.

Antti Tarvainen and Harri Valpola. 2017. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in neural information processing systems*, pages 1195–1204.

James Thorne and Andreas Vlachos. 2020. Avoiding catastrophic forgetting in mitigating model biases in sentence-pair classification with elastic weight consolidation. *arXiv preprint arXiv:2004.14366*.

James Thorne, Andreas Vlachos, Oana Cocarascu, Christos Christodoulopoulos, and Arpit Mittal. 2018. The fact extraction and VERification (FEVER) shared task. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 1–9, Brussels, Belgium. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, pages arXiv–1910.