# Distributionally Robust Recurrent Decoders with Random Network Distillation

**Antonio Valerio Miceli Barone**
University of Edinburgh
amiceli@ed.ac.uk

**Alexandra Birch**
University of Edinburgh
a.birch@ed.ac.uk

**Rico Sennrich**
Universität Zürich
sennrich@cl.uzh.ch

## Abstract

Neural machine learning models can successfully model language that is similar to their training distribution, but they are highly susceptible to degradation under distribution shift, which occurs in many practical applications when processing out-of-domain (OOD) text. This has been attributed to "shortcut learning": relying on weak correlations over arbitrary large contexts. We propose a method based on OOD detection with Random Network Distillation to allow an autoregressive language model to automatically disregard OOD context during inference, smoothly transitioning towards a less expressive but more robust model as the data becomes more OOD, while retaining its full context capability when operating in-distribution. We apply our method to a GRU architecture, demonstrating improvements on multiple language modeling (LM) datasets.

## 1 Introduction

Neural language models have become the main component of modern natural language processing systems, with larger and larger models being used as feature extractors for downstream tasks (Devlin et al., 2019), as probability estimators for ranking and ensembling (Gulcehre et al., 2015) or as language generators (Bahdanau et al., 2015; Vaswani et al., 2017; Brown et al., 2020).

Despite their success, neural machine learning models can suffer large performance degradation when they are applied to out-of-domain data which is substantially different than their training data (Lapuschkin et al., 2019; Hupkes et al., 2019; Recht et al., 2019).

Unlike the older statistical language models, Recurrent LMs (RNNLMs) (Mikolov et al., 2010) and their successors Transformers LMs (Vaswani et al., 2017) can consider the entire prefix of a sentence when predicting or generating the next token. By being able to relate a very high-dimensional input to the output, these models can learn many subtle correlations which are highly useful as long as the input is in-distribution, unfortunately these correlations tend to be brittle to distribution shift, causing a model that depends on them to go astray. This phenomenon is known as "shortcut learning" (Geirhos et al., 2020) and it has been found to also occur in humans and animals, but it is especially prevalent in artificial neural networks. Research on this problem has explored models invariant or equivariant w.r.t. certain transformations by means of compositional representations (Sabour et al., 2017; Soulos et al., 2019; Liu et al., 2020), causal modeling (Schölkopf et al., 2021), or both (Arjovsky et al., 2019; Krueger et al., 2020), but these works focus on classification tasks often on synthetic datasets and can't be straightforwardly applied to black-box language models. Approaches specific to LMs have focused on robustness where the data domains are known and represented in the training data (Oren et al., 2019; Gerstenberger et al., 2020).

In this work we propose a method that uses Random Network Distillation (RND) (Burda et al., 2018) to dynamically adapt the amount of context that the model relies upon during inference based on an estimate of how much this context is out-of-distribution (OOD). This way the model can still make use of all available context when operating within a familiar context space, exploiting long-distance weak correlations, but it reduces to a less expressive and more robust model when operating OOD, relying only on the strongest correlations. As a proof of concept we implement our approach on a GRU recurrent language model (Cho et al., 2014). While Transformer decoders outperform RNNs when trained on large training sets, RNNs remain competitive on smaller datasets ($< 10^7$ tokens) where OOD phenomena are easier to measure, furthermore they are easier to optimize, simplifying architecture and hyperparameter



*Proceedings of the 7th Workshop on Representation Learning for NLP*, pages 1 - 8
May 26, 2022 

search. We evaluate our method on language modeling tasks on English datasets, obtaining improvements when evaluating on eight OOD domains.We report additional preliminary sequence-to-sequence results on Transformer-RNN models (Zhang et al., 2018) in appendix A. We leave extensions of our method to full Transformers as future research.

## 2 Background

**Recurrent Language Model** Given a sequence $x(t)$ of tokens encoded as one-hot vectors, an autoregressive causal recurrent language model estimates at each step $t$ a probability distribution $\Pr(x(t+1)|x(0), \ldots, x(t)) = y(t+1)$ over the next token conditional on the observed prefix which is summarized as a fixed-dimensional state $h(t+1) \in \mathcal{R}^d$ computed according to the recurrence relation:

$$u(t) = \text{Emb}(x(t), \theta) \tag{1}$$
$$h(0) = 0^{\otimes d} \tag{2}$$
$$h(t+1) = \text{RNN}(h(t), u(t), \theta) \tag{3}$$
$$y(t+1) = \text{Proj}(h(t+1), \theta) \tag{4}$$

where Emb is an embedding layer, RNN is a recurrent cell (in our case, a GRU), Proj is a readout layer (we use a mixture-of-softmaxes layer (Yang et al., 2018)) and $\theta$ represents all the trainable parameters. The initial state $h(0)$ is fixed at zero.

An interesting property of this model is that close to the beginning of the sequence the state vector $h(t)$ has a small norm, and the entropy of the predicted token distribution is usually high because many tokens are plausible, while as more and more tokens are observed the state norm grows (token embeddings are approximately "added" to the state (Levy et al., 2018)) up to a point, and at the same time the entropy of the predicted token distribution decreases as the model becomes more confident of its prediction due to the larger observed context (Figure 1). Indeed, in a softmax readout layer:

$$\text{Proj}(h) = \text{softmax}(W \cdot h + b)$$

where $W$ is the output projection matrix and $b$ is the output bias vector, increasing the norm of the state vector $h$ will usually cause the probability distribution to become sharper unless $W \cdot h$ happens to approximately cancel out the bias vector $b$, which in high dimensions requires a rather specific alignment. A mixture-of-softmaxes readout also

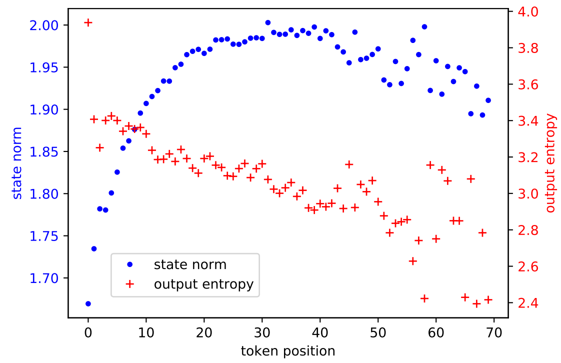


Figure 1: L2-norm of top GRU state (blue dots) and output softmax entropy (red crosses) over BPE token position, averaged over in-domain test set (Penn Treebank). Sentences are evaluated independently starting from the zero state. Norm and entropy correlate at $-0.58$. Model trained on Penn Treebank (sec. 4).

exhibits this property. Furthermore, it has been observed that the state of a RNN is usually dominated by the most recently observed inputs as the contribution of past inputs decreases exponentially over time (Jaeger, 2001; Pascanu et al., 2013; Levy et al., 2018; Zhang and Sennrich, 2019). Therefore, we hypothesize that the norm of the state vector corresponds to the amount of context that the model is considering for its future predictions, and this in turn controls the confidence of the model in its predictions.

**Random Network Distillation** In order to estimate how much the state of our RNNLM has deviated from the training distribution we choose the Random Network Distillation (RND) approach (Burda et al., 2018; Ciosek et al., 2020). Given a representation $h$, we define an OOD detector as

$$\text{OOD}(h) = |T(h) - S(h, \phi)|^2 \tag{5}$$

where $T(h)$ is a randomly initialized and frozen feed-forward teacher network that pseudo-randomly maps the state $h$ to a high-dimensional output and $S(h, \phi)$ is a feed-forward student network with parameters $\phi$ trained to copy the teacher by minimizing eq. 5 on the training set. At inference time the distillation error of eq. 5 provides an OOD estimate of $h$. This works by deliberately exploiting the fragility of neural networks w.r.t. distribution shift: while in principle the student could learn to copy the teacher for all possible inputs, in practice it only learns to do so on the training set (in-domain by definition) and becomes increasingly uncorrelated to it as the input becomes more OOD.

See Ciosek et al. (2020) for an extensive analysis. We chose this method because it can be applied to internal representations, is completely unsupervised and does not require any OOD tuning data. RND has been proposed initially in the context of reinforcement learning where the OOD signal can be used as a "curiosity" reward to stimulate exploration, and it has been subsequently studied in the context of OOD estimation for image classification. To our knowledge, we are the first to apply it to NLP, and to use it to actively compensate for distribution shift rather than just measure it.

## 3 Proposed approach

Our approach consists of estimating how much out-of-distribution the state of the model is and scaling it towards the all-zero initial state accordingly, effectively purging the OOD context out of the memory of the model and forcing it to rely only on the strongest, usually short-distance, correlations that survive the purge. As the state is pushed towards zero, the model also becomes more conservative in its predictions, avoiding the typical overconfidence of neural networks in OOD conditions. Specifically, for our language modeling experiments, we train a GRU RNNLM as usual, then we freeze it and train a RND OOD estimator on the RNNLM states on the same training set. Then during inference we modify the recurrence relation (eq. 3) to

$$\tilde{h} = \mathrm{RNN}(h(t), u(t), \theta) \tag{6}$$

$$h(t+1) = \tilde{h} \cdot \alpha \exp(-\beta \cdot \mathrm{OOD}(\tilde{h})) \tag{7}$$

where we use a simple exponential scaling with $\alpha$ and $\beta$ hyperparameters[1] which we set to 1. When the OOD signal is zero the model behaves like the baseline RNNLM, when it is high instead it behaves more like a unigram language model. This way, we can retain the expressivity of "shortcut learning" when it is beneficial, and hopefully avoid its influence when it is detrimental.

## 4 Experiments

**Setup** For all our language modelling experiments we use two-layer stacked GRUs, with a PyTorch implementation based on code by Zhang and Sennrich (2019)[2]. We train separate models on the Penn Treebank and Wikitext-2 corpora using the default hyperparameters provided by the codebase. We also train models on BPE subtokenized (Sennrich et al., 2016) versions of the corpora using SentencePiece[3]. These models are used both as baselines and to provide the initial models for our approach. For our approach we train one RND OOD model for each layer of the RNNLM, the teachers are 2-layer LeakyReLU MLPs (Maas et al., 2013) with layer normalization (Ba et al., 2016) and the students are like the teachers followed by 4 Resnet blocks (He et al., 2016) with 2 LeakyReLU MLP layers each. All hidden dimensions are set to match the RNNLM state dimension. For consistency with the original codebase, we use SGD with decaying learning rate and early stopping to train the baseline RNNLMs, while we switch to Adam (Kingma and Ba, 2015), with constant learning rate and early stopping when training the RND OOD estimator. GRU hyperparameters are the default ones from the reported Penn Treebank and Wikitext-2 models of the baseline implementation. The code to run the experiments is available.[4]

**Perplexity estimation** We investigate OOD performance with two standard corpora, Penn Treebank and Wikitext2. We evaluate each of the models both in-distribution, on the default test set of its training corpus, and out-of-distribution, on the test set of the other corpus. We also use additional test sets adapted from machine translation robustness evaluations, specifically the English sides of the De-En test sets of Müller et al. (2020), which is a collection of corpora from diffent domains (I.T., Koran, law, medical and movie subtitles) and the English sides of the MTNT Ja-En and Fr-En test sets of Michel and Neubig (2018), which are corpora scraped from Reddit and have been used for the WMT-19 robustness shared task (Li et al., 2019).

We report the results in tables 1 and 2. We find that for the word-level models trained on Penn Treebank our approach improves the perplexity consistently both in-distribution and out-of-distribution for all the test sets we considered. For the word-level models trained on Wikitext-2 our approach preserves perplexity in-distribution and improves it on most OOD test sets, namely the Penn Treebank test set, the Ja-En test set of the MTNT corpus and all the tests sets of Müller et al. (2020) except

[1] $\alpha$ can also be tuned by SGD on the training set, but we found this to be unnecessary.

[2] https://github.com/bzhangGo/lrn

[3] https://github.com/google/sentencepiece

[4] https://github.com/Avmb/lm-robustness

| | in-domain | | (Müller et al., 2020) | | | | | MTNT | |
|---|---|---|---|---|---|---|---|---|---|
| | Penn | WT-2 | IT | Koran | Law | Med | Sub | fr-en.en | ja-en.en |
| | | | | | word-level | | | | |
| Baseline | 68.04 | 55.73 | 59.37 | 50.12 | 64.12 | 35.10 | 47.81 | 76.75 | 66.08 |
| RND | 67.86 | **55.00** | **58.18** | **49.12** | **62.94** | **34.67** | **47.01** | **75.33** | **64.73** |
| RND (abl.) | **67.84** | 55.41 | 59.02 | 49.76 | 63.67 | 34.99 | 47.55 | 76.25 | 65.64 |
| | | | | | BPE-level | | | | |
| Baseline | **27.85** | 1371.16 | 5657.39 | 5493.64 | 4123.78 | 5657.54 | 4048.14 | 2837.97 | 4051.66 |
| RND | 28.16 | **1197.07** | **4828.55** | **4774.78** | **3552.22** | **4520.38** | **3558.99** | **2519.75** | **3551.63** |
| RND (abl.) | 27.93 | 1287.26 | 5178.40 | 5159.36 | 3815.87 | 4898.91 | 3792.19 | 2675.51 | 3794.22 |

Table 1: Perplexity of language models trained on the Penn Treebank dataset.

| | in-domain | | (Müller et al., 2020) | | | | | MTNT | |
|---|---|---|---|---|---|---|---|---|---|
| | WT-2 | Penn | IT | Koran | Law | Med | Sub | fr-en.en | ja-en.en |
| | | | | | word-level | | | | |
| Baseline | **64.69** | 361.84 | 162.01 | 159.02 | 178.92 | 103.87 | **96.65** | **177.73** | 184.69 |
| RND | **64.69** | **333.52** | **156.73** | 156.59 | **171.42** | **102.33** | 100.46 | 175.34 | **180.54** |
| RND (abl.) | **64.69** | 338.33 | 157.96 | **155.75** | 172.94 | 102.74 | 98.82 | 174.20 | 180.91 |
| | | | | | BPE-level | | | | |
| Baseline | **29.39** | 190.91 | 648.84 | **694.86** | 339.46 | 355.74 | **563.92** | **495.39** | **497.27** |
| RND | 29.73 | **183.23** | 637.63 | 712.93 | 335.86 | 348.16 | 656.86 | 530.45 | 526.30 |
| RND (abl.) | 29.46 | 185.48 | **632.52** | 695.54 | **334.37** | **347.75** | 624.27 | 515.16 | 512.96 |

Table 2: Perplexity of language models trained on the Wikitext-2 dataset.

| | | | (Müller et al., 2020) | | | | | MTNT | |
|---|---|---|---|---|---|---|---|---|---|
| Training | WT-2 | Penn | IT | Koran | Law | Med | Sub | fr-en.en | ja-en.en |
| WT-2 | **0.0240*** | 0.1137 | 0.0735 | 0.1155 | 0.0767 | 0.0485 | 0.0936 | 0.1070 | 0.0896 |
| Penn | 0.0237 | 0.0252* | 0.0244 | **0.0233** | 0.0244 | 0.0234 | 0.0236 | 0.0240 | 0.0238 |
| WT-2 (BPE) | **0.0220*** | 0.0534 | 0.1054 | 0.1824 | 0.0697 | 0.0657 | 0.1472 | 0.1328 | 0.1196 |
| Penn (BPE) | **0.0256** | 0.0257* | 0.0321 | 0.0279 | 0.0313 | 0.0359 | 0.0300 | 0.0308 | 0.0302 |

Table 3: OOD estimates, averaged over GRU layers and tokens in each test set. * denotes the in-domain test sets.

the subtitles test set. The Penn Treebank results are somewhat anomalous in that the perplexity of some OOD test sets is lower than the perplexity of the in-distribution test sets (and in fact the perplexity of the Wikitext-2 test set is even lower that the perplexity of the same test set evaluated by its own in-domain model). This effect is caused by the limited vocabulary of the Penn Treebank training set which causes many of the tokens of the OOD test sets to be replaced by UNKs, which are easy to predict. To avoid this artifact, we evaluate BPE-level models, which are open vocabulary and hence do not introduce any UNKs. For the BPE-level models we find that for both the baselines and the RND approach the perplexities on the OOD datasets are much higher than the perplexities on the in-domain test sets. Comparing our approach to the baselines, we observe a minimal degradation of perplexity in-distribution and substantial improvements on all OOD test sets when training on Penn Treebank, while when training on Wikitext-2 we observe more mixed results.

In order to analyse if the model is learning sensible values for scaling the out-of-distribution states, we compute the OOD scores estimated by the RND OOD detectors, averaged over the two GRU layers and over all the tokens in each test set. We report these scores in table 3. The models trained on Wikitext-2 (both the word-level and BPE-level versions) always estimate the lowest OOD scores on the in-domain test set, as expected. The Penn Treebank word-level model performs poorly, estimating similar scores for all the test sets, consistent with the aforementioned vocabulary collapse to UNKs, the BPE-level model instead is generally able to distinguish in-domain and out-of-domain test sets, albeit by a small margin and fails on one test set (Wikitext-2).

**Ablation** One could hypothesize that the improvements obtained by our model are due to just increasing the entropy of the output distribution rather than dropping unnecessary context from the RNN state. We evaluate a variant of our model where we apply the OOD scaling only on the output of the top-layer RNN but not to the internal states. This increases the output entropy without affecting the context remembered by the model between time steps. This ablation generally improves over the baseline but performs worse than our full model except for the model trained on Wikitext-2 BPE where the results are mixed.

## 5 Conclusions and future work

We proposed a method to improve the robustness of language models to distribution shift caused by train/test domain mismatch. Our model contracts the RNN state based on an unsupervised out-of-distribution estimator in order to reduce the model dependency on weak long-distance correlations, which are useful in-distribution but tend to be spurious in out-of-distribution conditions. We obtain perplexity improvements on multiple out-of-domain test sets without substantial degradation on in-domain test sets.

While our approach is based on Recurrent decoders, its general principles may be applicable to other neural architectures. For instance, the self-attention heads of a Transformer might modulated by an OOD detector in order to avoid attending to out-of-distribution parts of a sentence. We anticipate that extending our method to these kind of models will be a promising research direction.

## Broader impact and ethical concerns

This work provides improvements for language model technology on application domains not well represented in the training data.

We expect that our approach might promote an increased deployment and usage of such technology. We do not expect our approach to introduce any bias against any specific group of users. Our approach adds only small computational costs over baseline language models and therefore is unlikely to prevent users with limited computational budgets from benefiting from the technology.

## Acknowledgments

This project received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement 825299 (GoURMET), the European Research Council (ERC StG BroadSem 678254; ERC CoG TransModal 681760) and funding by the UK Engineering and Physical Sciences Research Council (EPSRC) fellowship grant EP/S001271/1 (MTStretch).

## References

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. 2019. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. 2016. Layer normalization.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. 2018. Exploration by random network distillation. *arXiv preprint arXiv:1810.12894*.

Mauro Cettolo, Jan Niehues, Sebastian Stüker, Luisa Bentivogli, and Marcello Federico. 2014. Report on the 11th iwslt evaluation campaign, iwslt 2014. In *Proceedings of the International Workshop on Spoken Language Translation, Hanoi, Vietnam*, volume 57.

Mia Xu Chen, Orhan Firat, Ankur Bapna, Melvin Johnson, Wolfgang Macherey, George F. Foster, Llion Jones, Niki Parmar, Mike Schuster, Zhifeng Chen, Yonghui Wu, and Macduff Hughes. 2018. The best of both worlds: Combining recent advances in neural machine translation. *CoRR*, abs/1804.09849.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in*

*Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

Kamil Ciosek, Vincent Fortuin, Ryota Tomioka, Katja Hofmann, and Richard Turner. 2020. Conservative uncertainty estimation by fitting prior networks. In *International Conference on Learning Representations*.

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.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. 2020. Shortcut learning in deep neural networks. *arXiv preprint arXiv:2004.07780*.

Alexander Gerstenberger, Kazuki Irie, Pavel Golik, Eugen Beck, and Hermann Ney. 2020. Domain robust, fast, and compact neural language models. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7954–7958.

Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loic Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2015. On Using Monolingual Corpora in Neural Machine Translation. *arXiv e-prints*, page arXiv:1503.03535.

K. He, X. Zhang, S. Ren, and J. Sun. 2016. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

Dieuwke Hupkes, Verna Dankers, Mathijs Mul, and Elia Bruni. 2019. The compositionality of neural networks: integrating symbolism and connectionism. *CoRR*, abs/1908.08351.

Herbert Jaeger. 2001. The "echo state" approach to analysing and training recurrent neural networks-with an erratum note. *Bonn, Germany: German National Research Center for Information Technology GMD Technical Report*, 148(34):13.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

David Krueger, Ethan Caballero, Jörn-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Rémi Le Priol, and Aaron C. Courville. 2020. Out-of-distribution generalization via risk extrapolation (rex). *CoRR*, abs/2003.00688.

Sebastian Lapuschkin, Stephan Wäldchen, Alexander Binder, Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. 2019. Unmasking clever hans predictors and assessing what machines really learn. *CoRR*, abs/1902.10178.

Omer Levy, Kenton Lee, Nicholas FitzGerald, and Luke Zettlemoyer. 2018. Long short-term memory as a dynamically computed element-wise weighted sum. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 732–739, Melbourne, Australia. Association for Computational Linguistics.

Xian Li, Paul Michel, Antonios Anastasopoulos, Yonatan Belinkov, Nadir Durrani, Orhan Firat, Philipp Koehn, Graham Neubig, Juan Pino, and Hassan Sajjad. 2019. Findings of the first shared task on machine translation robustness. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 91–102, Florence, Italy. Association for Computational Linguistics.

Qian Liu, Shengnan An, Jian-Guang Lou, Bei Chen, Zeqi Lin, Yan Gao, Bin Zhou, Nanning Zheng, and Dongmei Zhang. 2020. Compositional generalization by learning analytical expressions. *arXiv preprint arXiv:2006.10627*.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal. Association for Computational Linguistics.

Andrew L. Maas, Awni Y. Hannun, and Andrew Y. Ng. 2013. Rectifier nonlinearities improve neural network acoustic models. In *in ICML Workshop on Deep Learning for Audio, Speech and Language Processing*.

Paul Michel and Graham Neubig. 2018. Mtnt: A testbed for machine translation of noisy text. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černockỳ, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *Eleventh annual conference of the international speech communication association*.

Mathias Müller, Annette Rios, and Rico Sennrich. 2020. Domain robustness in neural machine translation. In *Proceedings of the 14th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pages 151–164, Virtual. Association for Machine Translation in the Americas.

Yonatan Oren, Shiori Sagawa, Tatsunori B. Hashimoto, and Percy Liang. 2019. Distributionally robust language modeling. 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 4227–4237, Hong Kong, China. Association for Computational Linguistics.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. On the difficulty of training recurrent neural networks. In *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, ICML'13, page III–1310–III–1318. JMLR.org.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. 2019. Do imagenet classifiers generalize to imagenet? *CoRR*, abs/1902.10811.

Sara Sabour, Nicholas Frosst, and Geoffrey E. Hinton. 2017. Dynamic routing between capsules. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 3859–3869, Red Hook, NY, USA. Curran Associates Inc.

Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. 2021. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Paul Soulos, Tom McCoy, Tal Linzen, and Paul Smolensky. 2019. Discovering the compositional structure of vector representations with role learning networks. *CoRR*, abs/1910.09113.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, pages 5998–6008. Curran Associates, Inc.

Zhilin Yang, Zihang Dai, Ruslan Salakhutdinov, and William W. Cohen. 2018. Breaking the softmax bottleneck: A high-rank RNN language model. In *International Conference on Learning Representations*.

Biao Zhang and Rico Sennrich. 2019. A lightweight recurrent network for sequence modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1538–1548, Florence, Italy. Association for Computational Linguistics.

Biao Zhang, Deyi Xiong, and Jinsong Su. 2018. Accelerating neural transformer via an average attention network. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1789–1798, Melbourne, Australia. Association for Computational Linguistics.

# Appendices

## A Sequence-to-sequence experiments

We performed additional experiments on sequence-to-sequence (seq2seq) tasks. We obtained negative results, which we report here.

**Architecture** Our models use a Transformer-GRU architecture. The encoder is a standard bidirectional Transformer while the decoder is a two-layer stacked GRU (sec. 2). The recurrent cell also accesses contextual embeddings of a source sentence tokens via an attention mechanism implemented as in Luong et al. (2015), except that instead of a single attention head we use a Transformer multihead attention layer, similar to Chen et al. (2018). The RND OOD model has the same architecture as in the LM experiments, although for simplicity we train it jointly with the MT models rather than in a separate stage, we make sure not to propagate gradients between the RND OOD model and the translation model hence there is no tradeoff between their training objectives. The implementation is based on the Fairseq (Ott et al., 2019) Transformer and LSTM architectures, using the hyperparameters for their default IWSLT14 configuration.

**Machine translation** We trained De→En translation models on the IWSLT14 training set (Cettolo et al., 2014) with the standard Fairseq preprocessing pipeline[5]. We used on the standard test set produced by the preprocessing script as our in-domain test set and the Müller et al. (2020) test sets as our OOD test sets. We report BLEU scores in table 4. The baseline and the RND model have nearly identical scores on the in-domain test set, while they deviate up to about 1 BLEU point on the OOD test sets, although in a non-systematic way.

| | in-domain | (Müller et al., 2020) | | | | |
|---|---|---|---|---|---|---|
| | IWSLT14 | IT | Koran | Law | Med | Sub |
| Base | 32.95 | 11.03 | **5.72** | 11.35 | 13.76 | **19.19** |
| RND | **32.97** | **12.07** | 5.13 | **12.02** | **13.93** | 18.71 |

Table 4: Machine translation results

**Sentence reversal** We considered a synthetic task intended to elicit the RND OOD activity. The source segments consist each of a number of concatenated sentences separated by a separator token, the target segments are made of the same sentences, where each sentence is reversed at token level, but the sentences are concatenated in the same order as the source. Since reversing a sentence does not depend on the previous sentences in the segment, the previous sentences become distractors that pollute the decoder GRU state with irrelevant information. The model can learn to compensate in in-domain conditions where the test set is sampled from the same distribution of the training set, but we hypothesize that in OOD scenarios with longer segments composed by a higher number of sentences this spurious information will greatly decrease accuracy. We test whether the RND OOD mechanism is effective at discarding this spurious information.

We consider two versions of the task, in one we sample the source segments from a synthetic vocabulary of 256 tokens, with uniform probability per token, 32 tokens per sentence, 8 sentences per training segment. We test in-domain at 8 sentences and OOD at 10 and 12 sentences per segment. In the second version, we train on concatenations of 4 consecutive sentences of the English side of the IWSLT14 De-En training set, and we test at $4, 6, 8, 10$ and 12 sentences per segment. We use the same hyperparameters of our translation experiments, during inference we constrain the decoder to match the source length.

All the models achieve near perfect ($> 99.9$) BLEU scores in-domain, while OOD the scores quickly decrease as the number of sentences per segment increases, as expected. Unfortunately we find no systematic difference between baseline and RND OOD models.

**Discussion** Unlike our language modeling experiments, we did not observe systematic improvements from using the RND out-of-distribution detector to contract the state of the GRU decoder in our sequence-to-sequence results. There are multiple possible hypotheses for this discrepancy, such as encoder effects, generating outputs by beam search rather than scoring natural text, or the target distribution being more peaked around the mode. We plan to investigate this effect in the future.

[5]prepare-iwslt14.sh