On Isotropy Calibration of Transformers

Yue Ding[*2], **Karolis Martinkus**[*1], **Damián Pascual**[*1],
Simon Clematide[2], **Roger Wattenhofer**[1]
[1]ETH Zürich [2]University of Zürich
yue.ding@uzh.ch damianp@ethz.ch martinkus@ethz.ch
siclemat@cl.uzh.ch wattenhofer@ethz.ch

Abstract

Different studies of the embedding space of transformer models suggest that the distribution of contextual representations is highly anisotropic — the embeddings are distributed in a narrow cone. Meanwhile, static word representations (e.g., Word2Vec or GloVe) have been shown to benefit from isotropic spaces. Therefore, previous work has developed methods to calibrate the embedding space of transformers in order to ensure isotropy. However, a recent study (Cai et al., 2021) shows that the embedding space of transformers is locally isotropic, which suggests that these models are already capable of exploiting the expressive capacity of their embedding space. In this work, we conduct an empirical evaluation of state-of-the-art methods for isotropy calibration on transformers and find that they do not provide consistent improvements across models and tasks. These results support the thesis that, given the local isotropy, transformers do not benefit from additional isotropy calibration.

1 Introduction

The impressive performance of transformer models (Vaswani et al., 2017) across almost all areas of Natural Language Processing (NLP) has sparked in-depth investigations of these models. A remarkable finding is that the contextual representations computed by transformers are strongly anistropic (Ethayarajh, 2019), i.e., they are unevenly distributed and localized in a narrow cone of the embedding space. This discovery, labeled as the *representation degeneration problem* by Gao et al. (2019) is surprising since it suggests that most of the expressive capacity of these high-dimensional spaces is neglected by transformers.

Furthermore, previous work on static word representations, e.g., GloVE (Pennington et al., 2014) or Word2Vec (Mikolov et al., 2013), established that isotropy is a desirable property in non-contextual embedding spaces (Mu and Viswanath, 2018). Indeed, Mu and Viswanath (2018) and Liu et al. (2019a) showed that post-processing static word embeddings in order to increase isotropy improves their performance in downstream tasks. Based on these results, recent work has developed methods to correct the anisotropy of the contextual representations generated by transformers (Gao et al., 2019; Wang et al., 2019b; Li et al., 2020). These isotropy calibration methods have been reported to produce small gains in performance on some NLP tasks.

However, in a recent study, Cai et al. (2021) show that the space of contextual embeddings of transformers is locally isotropic. By analyzing low dimensional sub-spaces the authors identify isolated clusters and manifolds and argue that isotropy does exist in these manifolds. In the same line, Luo et al. (2021) and Kovaleva et al. (2021) find that in BERT (Devlin et al., 2019) almost all of the embeddings present large values in the same two components of the embedding vector. These large components distort our understanding of the embedding spaces by making all the representations have high cosine similarity. In this work, we perform an extensive empirical evaluation of isotropy calibration methods across different tasks and models to determine if they provide consistent improvements. Our results question the utility of isotropy calibration in transformers, implicitly supporting the argument that transformers do already benefit from local isotropy (Cai et al., 2021).

2 Related Work

Since the appearance of the transformer architecture and its multiple variants, of which BERT (Devlin et al., 2019) stands out as the most researched model, a lot of effort has been devoted to understanding their inner workings (Rogers et al., 2020). Unlike static word embeddings such as GloVE or Word2Vec, transformers build contextual embed-

[*] First three authors in alphabetic order

Proceedings of the Third Workshop on Insights from Negative Results in NLP, pages 1 - 9
May 26, 2022 ©2022 Association for Computational Linguistics

dings, i.e., dynamic representations that aggregate information from other context words. These representations have sparked a lot of research interest. Wu et al. (2020) showed that different transformer architectures produce similar contextual representations. Chronis and Erk (2020) studied the similarity and relatedness of contextual representations in the embedding spaces of BERT, while Brunner et al. (2019) studied how identifiable the intermediate representations of BERT are with respect to the input. Zhao et al. (2020) quantified the contextual knowledge of BERT and Zhao et al. (2021) analyzed the embedding spaces of BERT in order to quantify the non-linearity of its layers.

Following the discovery of anisotropy in transformers (Gao et al., 2019; Ethayarajh, 2019), different isotropy calibration methods have been developed to correct this phenomenon. Gao et al. (2019) and Zhang et al. (2020) introduced regularization objectives that affect the embedding distances. Zhou et al. (2021) presented a module inspired by batch-norm that regularizes the embeddings towards isotropic representations. Wang et al. (2019b) proposed to control the singular value decay of the output layer of transformers and Li et al. (2020) used normalizing flows to map transformer embeddings to an isotropic space. However, Cai et al. (2021) show that contextual representations are locally isotropic and suggest that this property allows transformers to exploit their full expressive capacity, questioning the utility of isotropy calibration.

3 Isotropy Calibration Methods

The output distribution of transformers is typically parameterized as a softmax function:

$$P(\boldsymbol{Y}_i = \boldsymbol{y}_i | \boldsymbol{h}_i) = \frac{\exp(\boldsymbol{h}_i^T \boldsymbol{W}_{\mathcal{I}(\boldsymbol{y}_i)})}{\sum_{j=1}^{N} \exp(\boldsymbol{h}_i^T \boldsymbol{W}_j)} ,$$

where $\boldsymbol{W} \in \mathcal{R}^{N \times d}$ is the output weight matrix, d is the embedding dimension, N is the output size, $\boldsymbol{y}_i$ is the i-th output, $\mathcal{I}(\boldsymbol{y}_i)$ is the index of $\boldsymbol{y}_i$ and $\boldsymbol{h}$ is the contextual embedding produced by the model. Since this constitutes a shared space between model embeddings $\boldsymbol{h} \in \boldsymbol{H}$ and output embeddings, isotropy at the output distribution can be enforced by calibrating either $\boldsymbol{H}$ or $\boldsymbol{W}$.

We experiment with three prominent methods for isotropy calibration on transformers:

Cosine Regularization. Gao et al. (2019) introduce a simple regularization term that minimizes the cosine similarity between any two output embeddings in order to increase the aperture of the cone that contains the embeddings. This regularization term is given by:

$$\mathcal{R}_{cos} = \lambda_c \frac{1}{|\mathcal{V}|^2} \sum_{i}^{n} \sum_{j \neq i}^{n} \hat{\boldsymbol{w}}_i^T \hat{\boldsymbol{w}}_j ,$$

where $\boldsymbol{w}_i$ is the embedding of the i-th token in the vocabulary $\mathcal{V}$, $\hat{\boldsymbol{w}} = \frac{\boldsymbol{w}}{||\boldsymbol{w}||}$ and λ_c is the regularization constant.

Spectrum Control. Wang et al. (2019b) increase isotropy by mitigating the fast decay of the singular value distribution of the output matrix $\boldsymbol{W}$. They decompose $\boldsymbol{W}$ using Singular Value Decomposition (SVD), such that $\boldsymbol{W} = \boldsymbol{U\Sigma V}^T$, where $\boldsymbol{\Sigma} \in \mathcal{R}^{d \times d}$ is the diagonal matrix of singular values. Then, they add a regularization term to guide the singular value distribution towards a prespecified slow-decaying prior distribution. This term spreads the variance away from the first few dominating singular values, increasing the isotropy of the space. They propose the following two regularization terms:

$$\mathcal{R}_{pol}(\boldsymbol{\Sigma}) = \lambda_p \sum_{k=1}^{d} (\sigma_k - c_1 k^{\gamma})^2 ,$$

for polynomial singular value decay; and

$$\mathcal{R}_{exp}(\boldsymbol{\Sigma}) = \lambda_e \sum_{k=1}^{d} (\sigma_k - c_1 \exp(-c_2 k^{\gamma}))^2 ,$$

for exponential decay, where λ_e, λ_p, c_1 and c_2 are regularization constants, σ_k is the k-th largest singular value and γ is a parameter which controls the rate of singular value decay.

Flow Model. Li et al. (2020) propose a method that leverages normalizing flows to learn an invertible mapping f_ϕ^{-1} between the embedding space of the transformer model and an isotropic (Gaussian) space $\mathcal{Z}$. First, an invertible flow model (Kingma and Dhariwal, 2018) f_ϕ is trained to generate transformer embedding vectors $\boldsymbol{h}$ from Gaussian noise $\boldsymbol{z}$:

$$\boldsymbol{z} \sim p_{\mathcal{Z}}(\boldsymbol{z}), \; \boldsymbol{h} = f_\phi(\boldsymbol{z}) .$$

Then, the model f_ϕ is inverted to map transformer embeddings $\boldsymbol{h}$ to the new (and isotropic) output embedding space $\mathcal{Z}$.

	SST-2	MRPC	CoLA	RTE	WNLI	STS-B	QNLI	MNLI		QQP
Model	Accuracy	F1	Mat. corr.	Accuracy	Accuracy	Pearson corr.	Accuracy	Match acc.	Mismatch acc.	Accuracy
BERT	**91.44 ±0.52**	**88.80±0.99**	**53.16±1.82**	**58.97±1.82**	53.52±4.88	**80.86 ±2.11**	88.78±0.57	81.02±0.17	81.78±0.40	89.31±0.06
+*Cosreg*	90.71 ±1.00	88.17 ±0.38	46.94 ±4.29	56.43 ±5.16	50.23 ±4.95	78.23 ±2.19	**89.58 ±0.19**	**81.20 ±0.41**	**82.04 ±0.21**	89.26 ±0.10
+*Spectrum-Pol*	90.86 ±1.35	81.22 ±0	0	49.58 ±3.62	**56.34 ±0**	NaN	81.24 ±4.45	64.33 ±27.80	64.76 ±27.48	87.15 ±2.23
+*Spectrum-Exp*	91.21 ±0.37	81.22 ±0	0	50.90 ±3.45	**56.34 ±0**	NaN	86.42 ±0.42	62.43 ±24.97	63.12 ±25.20	89.16 ±0.45
+*Flow*	91.09 ±0.54	86.99 ±0.89	51.19 ±1.81	54.27 ±1.46	48.36 ±5.86	78.88 ±3.46	86.21 ±3.38	80.65 ±0.46	81.15 ±0.21	**89.36 ±0.10**
RoBERTa	**92.97 ±0.63**	85.35 ±8.52	**53.67 ±3.32**	53.19 ±0.55	54.46 ±0.81	**83.10 ±2.87**	**91.00 ±0.46**	85.16 ±0.28	85.19 ±0.15	89.85 ±0.13
+*Cosreg*	92.66 ±0.23	**89.17 ±2.28**	48.99 ±5.61	**53.67 ±1.16**	53.52 ±1.41	28.44 ±44.84	90.89 ±0.19	**85.41 ±0.09**	**85.64 ±0.22** *	**89.87 ±0.12**
+*Spectrum-Pol*	88.08 ±0.99	81.22 ±0	0	52.71 ±0	**57.28 ±1.62** *	NaN	83.89 ±2.46	50.63 ±29.72	51.14 ±29.29	81.76 ±12.76
+*Spectrum-Exp*	90.71 ±1.09	81.22 ±0	0	52.95 ±0.42	56.34 ±0	NaN	82.25 ±3.14	84.46 ±0.51	84.77 0.41	80.95 ±13.89
DistilBERT	88.23 ±1.79	**87.97 ±1.02**	44.11 ±2.09	56.68 ±0.62	51.17 ±5.69	**23.63 ±41.08**	**87.53 ±0.13**	**78.84 ±0.27**	**79.50 ±0.32**	88.28 ±0.25
+*Cosreg*	88.53 ±1.55	87.88 ±1.36	**43.13 ±0.85**	**58.24 ±1.78**	52.11 ±2.44	-0.50 ±2.08	87.15 ±0.84	78.69 ±0.17	79.42 ±0.28	88.38 ±0.05
+*Spectrum-Pol*	88.80 ±0.37	81.22 ±0	0	54.15 ±2.50	**55.87 ±0.81**	NaN	85.47 ±0.96	78.39 ±0.17	79.13 ±0.05	**88.41 ±0.43**
+*Spectrum-Exp*	**88.92 ±0.67**	81.22 ±0	0	54.27 ±2.71	**55.87 ±0.81**	NaN	86.25 ±0.80	78.38 ±1.34	79.03 ±0.34	88.12 ±0.58

Table 1: Performance for different models and calibration methods on GLUE; * denotes significantly better performance than the corresponding uncalibrated model ($p < 0.05$, two-sample t-test). The NaN and 0 scores are caused by the model always predicting the same class.

4 Experiments

We evaluate the impact of each of these calibration methods on state-of-the-art transformer models in three prominent areas of Natural Language Processing: language understanding, machine translation, and summarization. For all of the models, we use the implementation and fine-tuning parameters from HuggingFace (Wolf et al., 2020) (cf. Appendix B). We run each experiment three times and report the mean and standard deviation. Fine-tuning time is reported on a Nvidia Titan RTX GPU.

To characterize the isotropy of the output embedding space we adopt the I_1 and I_2 isotropy measures from (Wang et al., 2019b), with $I_1(\boldsymbol{W}) \in [0, 1]$ and $I_2(\boldsymbol{W}) \geq 0$. Larger $I_1(\boldsymbol{W})$ and smaller $I_2(\boldsymbol{W})$ indicate more isotropic embeddings (cf. App. A for details).

4.1 Language Understanding

We consider three representative transformer models with different sizes, BERT-base (Devlin et al., 2019), RoBERTa (Liu et al., 2019b), and DistilBERT (Sanh et al., 2020). We evaluate these models on the development set of GLUE (Wang et al., 2019a), a well-known benchmark for language understanding that consists of nine different tasks. Due to the high computational cost of flow calibration and the large number of tasks, we apply this method only on BERT to save resources.

In Table 1 we report the performance per task of the calibrated and uncalibrated models. We observe the same pattern for all three models. In the overwhelming majority of cases, the calibrated models perform comparably to or worse than the uncalibrated ones, with calibration improving performance with statistical significance ($p < 0.05$, two-sample t-test) only in RoBERTa for WNLI with exponential decay and MNLI mismatched with cosine regularization. More specifically, cosine regularization and flow calibration (in BERT) do not affect performance much, while spectrum control in some cases produces severe performance degradation or even prevents learning, e.g., CoLA and STS-B. Furthermore, flow calibration adds a large training overhead, requiring on average 4.2 times more time per training epoch.

These results reveal that no isotropy calibration method yields consistently better performance than the uncalibrated models in language understanding tasks.

4.2 Machine Translation

We test multilingual BART (M-BART) (Liu et al., 2020) on English-Romanian and German-English WMT16 (Bojar et al., 2016) translation datasets. In Table 2 we report BLUE scores, compute time, and the isotropy metrics, for the uncalibrated and calibrated models. To reduce the high computational cost of flow calibration, we apply this method only on a reduced version of 50 000 samples for both tasks, English-Romanian and German-English translation. As a reference, we also provide the scores of the uncalibrated model on the small datasets. We find, that while cosine regularization does not significantly affect either BLEU scores or isotropy metrics, both variants of spectrum control improve isotropy but produce a performance degradation of over 3 and 5 BLEU points in the English-Romanian and German-English tasks respectively, while requiring 25% to 50% more computation

	EN-RO				DE-EN			
Model	BLEU (↑)	I_1(↑)	I_2(↓)	Time (min)	BLEU (↑)	I_1(↑)	I_2(↓)	Time (min)
M-BART	**26.15 ±0.08**	0.88 ±0.01	0.60 ±0	108 ±0	22.81 ±0.35	0.89 ±0.01	0.60 ±0	176 ±0
+*Cosreg*	26.07 ±0.10	0.88 ±0.01	0.60 ±0	110 ±0	**23.03 ±0.27**	0.89 ±0.01	0.60 ±0	188 ±1
+*Spectrum-Pol*	22.94 ±0.18	**1.00 ±0**	**0.02 ±0**	176 ±2	16.27 ±0.06	**1.00 ±0**	**0.02 ±0**	265 ±0
+*Spectrum-Exp*	22.92 ±0.05	**1.00 ±0**	**0.02 ±0**	170 ±1	16.24 ±0.12	**1.00 ±0**	**0.02 ±0**	230 ±18
M-BART (small dataset)	**9.09 ±1.02**	0.88 ±0	**0.60 ±0**	9 ±0	**11.61 ±2.25**	**0.88 ±0**	**0.60 ±0**	9 ±0
+*Flow*	8.57 ±2.52	**0.89 ±0**	**0.60 ±0**	95 ±0	10.93 ±0.70	**0.88 ±0**	**0.60 ±0**	96 ±1

Table 2: Multilingual BART performance, isotropy (I_1 and I_2) and fine-tuning time per epoch with different calibration methods for English - Romanian and German - English translation. Due to computational cost, the flow method was tested only on a smaller version of the EN-RO dataset with 50 000 sentences.

time. On the other hand, flow calibration yields comparable BLEU score to the uncalibrated model but requires on average 10.5 times more computation per epoch. These results suggest a negative and counter-intuitive relation between isotropy and downstream performance: when isotropy increases, performance decreases. We observe a similar trend for language understanding in Appendix C.

Overall, and in line with the results in the previous section, isotropy calibration in machine translation tends to degrade performance and increase the computational budget.

4.3 Summarization

We evaluate BART (Lewis et al., 2020) on the CNN/DM summarization task (Hermann et al., 2015); again we use a reduced dataset (20 000 articles) for flow calibration. The results in Table 3 show that none of the calibrated models performs significantly better than their uncalibrated counterparts in terms of ROUGE score (Lin, 2004) (cf. Appendix D). Cosine regularization does not affect performance nor isotropy, while spectrum control improves isotropy (I_1 and I_2) at the cost of a small performance drop. The flow model performs comparably to uncalibrated BART but requires 5.5 times more computation. Overall, we find no evidence that isotropy calibration provides gains in summarization.

	CNN / Daily Mail			
Model	R-1 (↑)	I_1(↑)	I_2(↓)	Time (min)
BART	**38.21 ±0.05**	0.95 ±0.01	0.25 ±0	246 ±8
+*Cosreg*	**38.21 ±0.05**	0.95 ±0.01	0.25 ±0	240 ±8
+*Spectrum-Pol*	37.36 ±0.08	**0.99 ±0**	**0.04 ±0**	245 ±20
+*Spectrum-Exp*	37.43 ±0.08	**0.99 ±0**	**0.04 ±0**	230 ±18
BART (small d.)	**36.56 ±0.25**	**0.94 ±0**	**0.25 ±0**	17 ±0
+*Flow*	36.15 ±0.30	**0.94 ±0**	**0.25 ±0**	95 ±2

Table 3: ROUGE-1 score, isotropy (I_1 and I_2), and fine-tuning time per epoch with different calibration methods on BART for summarization. Due to computational cost, the flow calibration method was tested on a smaller version of the dataset.

5 Discussion

Our extensive evaluation shows that none of the considered isotropy calibration methods produce consistent improvements over the uncalibrated models across tasks, domains and architectures. In fact, we observe a negative relation between isotropy calibration and downstream performance. The most aggressive method, i.e., spectrum control, produces the largest improvement in isotropy metrics as well as the most significant performance drop. On the other hand, the effect of cosine regularization and flow calibration is small in both, isotropy and performance.

According to Cai et al. (2021), the local isotropy of the embedding space of transformers may enable them to exploit their full expressive capacity. Furthermore, concurrent findings by Luo et al. (2021) and Kovaleva et al. (2021) reveal that certain components of the contextual embeddings consistently present very large magnitudes, which distort the cosine distances in the embedding space and questions their anisotropy. This could explain why additional isotropy calibration does not consistently improve the performance of transformers in downstream tasks.

In light of our results, we discourage isotropy calibration of transformers as a means of improving downstream performance. However, we believe that further investigation of the embedding space of transformers may be beneficial to increase our ability to interpret these models and improve their architecture.

References

Sanjeev Arora, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. 2016. A latent variable model approach to pmi-based word embeddings. *Transactions of the Association for Computational Linguistics*, 4:385–399.

Ondrej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurelie Neveol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. 2016. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation*, pages 131–198, Berlin, Germany. Association for Computational Linguistics.

Gino Brunner, Yang Liu, Damian Pascual, Oliver Richter, Massimiliano Ciaramita, and Roger Wattenhofer. 2019. On identifiability in transformers. In *International Conference on Learning Representations*.

Xingyu Cai, Jiaji Huang, Yuchen Bian, and Kenneth Church. 2021. Isotropy in the contextual embedding space: Clusters and manifolds. In *International Conference on Learning Representations*.

Gabriella Chronis and Katrin Erk. 2020. When is a bishop not like a rook? when it's like a rabbi! multi-prototype bert embeddings for estimating semantic relationships. In *Proceedings of the 24th Conference on Computational Natural Language Learning*, pages 227–244.

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.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings. 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 55–65, Hong Kong, China. Association for Computational Linguistics.

Jun Gao, Di He, Xu Tan, Tao Qin, Liwei Wang, and Tieyan Liu. 2019. Representation degeneration problem in training natural language generation models. In *International Conference on Learning Representations*.

Karl Moritz Hermann, Tomás Kociský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *NIPS*, pages 1693–1701.

Durk P Kingma and Prafulla Dhariwal. 2018. Glow: Generative flow with invertible 1x1 convolutions. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Olga Kovaleva, Saurabh Kulshreshtha, Anna Rogers, and Anna Rumshisky. 2021. BERT busters: Outlier dimensions that disrupt transformers. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3392–3405, Online. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pretraining for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. 2020. On the sentence embeddings from pre-trained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9119–9130, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Tianlin Liu, Lyle Ungar, and Joao Sedoc. 2019a. Unsupervised post-processing of word vectors via conceptor negation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6778–6785.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ziyang Luo, Artur Kulmizev, and Xiao-Xi Mao. 2021. Positional artefacts propagate through masked language model embeddings. In *ACL*.

Tomas Mikolov, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *ICLR*.

Jiaqi Mu and Pramod Viswanath. 2018. All-but-the-top: Simple and effective postprocessing for word representations. In *International Conference on Learning Representations*.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

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.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter.

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 *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 6000–6010.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2019a. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *7th International Conference on Learning Representations, ICLR 2019*.

Lingxiao Wang, Jing Huang, Kevin Huang, Ziniu Hu, Guangtao Wang, and Quanquan Gu. 2019b. Improving neural language generation with spectrum control. In *International Conference on Learning Representations*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

John Wu, Yonatan Belinkov, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. 2020. Similarity analysis of contextual word representation models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4638–4655.

Zhong Zhang, Chongming Gao, Cong Xu, Rui Miao, Qinli Yang, and Junming Shao. 2020. Revisiting representation degeneration problem in language modeling. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 518–527.

Mengjie Zhao, Philipp Dufter, Yadollah Yaghoobzadeh, and Hinrich Schütze. 2020. Quantifying the contextualization of word representations with semantic class probing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 1219–1234.

Sumu Zhao, Damián Pascual, Gino Brunner, and Roger Wattenhofer. 2021. Of non-linearity and commutativity in bert. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8.

Wenxuan Zhou, Bill Yuchen Lin, and Xiang Ren. 2021. Isobn: Fine-tuning bert with isotropic batch normalization. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(16):14621–14629.

A Isotropy Metrics

To characterize the isotropy of the output embedding space we adopt the I_1 and I_2 isotropy measures from (Wang et al., 2019b).

$$I_1(\boldsymbol{W}) = \frac{\min_{\boldsymbol{v}\in\boldsymbol{V}} Z(\boldsymbol{v})}{\max_{\boldsymbol{v}\in\boldsymbol{V}} Z(\boldsymbol{v})} ,$$

is based on the observation by (Arora et al., 2016), that the partition function $Z(\boldsymbol{v}) = \sum_{i=1}^{n} \exp(\boldsymbol{v}^T \boldsymbol{w}_i)$ should be close to a constant for any unit vector $\boldsymbol{v}$ if the embedding matrix $\boldsymbol{W}$ is isotropic. Here, we abuse notation and $\boldsymbol{w_i} \in \boldsymbol{W}$ is the i-th row of the embedding matrix $\boldsymbol{W}$. Following (Mu and Viswanath, 2018) we use the set of eigenvectors of $\boldsymbol{W}^T\boldsymbol{W}$ as $\boldsymbol{V}$. The second measure

$$I_2(\boldsymbol{W}) = \sqrt{\frac{\sum_{\boldsymbol{v}\in\boldsymbol{V}}(Z(\boldsymbol{v}) - \bar{Z}(\boldsymbol{v}))^2}{|V|\bar{Z}(\boldsymbol{v})^2}} ,$$

is the sample standard deviation of the partition function $Z(\boldsymbol{v})$ normalized by its average $\bar{Z}(\boldsymbol{v})$. This way, $I_1(\boldsymbol{W}) \in [0, 1]$ and $I_2(\boldsymbol{W}) \geq 0$. Larger $I_1(\boldsymbol{W})$ and smaller $I_2(\boldsymbol{W})$ indicate more isotropic embeddings.

B Model Hyperparameter Configuration

For all the models used in his work we use the implementation from HuggingFace and follow their instructions for the hyperparameters. In particular, we use the following configurations:

BERT and DistilBERT. Learning rate $2e^{-5}$ without scheduling, batch size 32, 3 training epochs for all GLUE tasks except for MRPC and WNLI, for which we train during 5 epochs.

RoBERTa. Learning rate of $1e^{-5}$ for all GLUE tasks except for SST-2 and STS-B, for which the learning rate is set to $1e^{-5}$, same number of epochs as for BERT and DistilBERT, batch size of 32.

M-BART and BART. Learning rate of $3e^{-5}$ with polynomial decay, batch size 48, and 5 training epochs.

C Isotropy Scores on GLUE

Here, in Table 4, we present the isotropy scores obtained in our evaluation of GLUE with BERT, RoBERTa, and DistilBERT, which were not included in the main text due to lack of space.

The isotropy metrics I_1 and I_2 show the opposite trend to the performance metrics. An improvement in isotropy reflects a decrease in downstream performance. This way, we see that across models and tasks, cosine regularization and flow calibration (for BERT) have a small impact on isotropy and that the performance of the models calibrated with these techniques is close to the that of the uncalibrated models. On the other hand, spectrum control produces a very significant increase in isotropy, with many tasks reaching a I_1 of 1.00; while in Table 1 we see how it produces strong performance degradation. This, further suggests a negative relation between isotropy and the downstream performance of transformers.

	SST-2		MRPC		CoLA	
Model	$I_1(\uparrow)$	$I_2(\downarrow)$	$I_1(\uparrow)$	$I_2(\downarrow)$	$I_1(\uparrow)$	$I_2(\downarrow)$
BERT	0.91 ±0.01	0.4 ±0	0.91 ±0.01	0.38 ±0.01	0.91 ±0.01	0.39 ±0.01
+*Cosreg*	0.91 ±0.2	0.39 ±0.02	0.92 ±0.01	0.39 ±0.2	0.91 ±0.01	0.39 ±0.01
+*Spectrum-Pol*	**1.00 ±0**	**0.007 ±0.003**	**1.00 ±0**	7e^{-4} ±3e^{-4}	**1.00 ±0**	**6**e^{-4} ±**1**e^{-4}
+*Spectrum-Exp*	0.99 ±0.01	0.02 ±0.02	**1.00 ±0**	**6**e^{-4} ±**2**e^{-4}	**1.00 ±0**	7e^{-4} ±3e^{-4}
+*Flow*	0.92 ±0.01	0.40 ±0	0.91 ±0.01	0.40 ±0	0.91 ±0.01	0.39 ±0.01
RoBERTa	0.91 ±0.01	0.39 ±0.01	0.92 ±0.01	0.39 ±0.01	0.91 ±0.01	0.40 ±0.01
+*Cosreg*	0.92 ±0.01	0.40 ±0.01	0.91 ±0.01	0.39 ±0.01	0.91 ±0.01	0.40 ±0.01
+*Spectrum-Pol*	**1.00 ±0**	0.008 ±0.002	**1.00 ±0**	5e^{-4} ±4e^{-4}	**1.00 ±0**	**5**e^{-4} ±**2**e^{-4}
+*Spectrum-Exp*	**1.00 ±0**	**0.005 ±0.004**	**1.00 ±0**	**1**e^{-4} ±**2**e^{-4}	**1.00 ±0**	6e^{-4} ±4e^{-4}
DistilBERT	0.91 ±0.01	0.38 ±0.01	0.92 ±0.01	0.39 ±0.01	0.92 ±0.01	0.38 ±0.01
+*Cosreg*	0.91 ±0.01	0.39 ±0.01	0.92 ±0.01	0.38 ±0.01	0.92 ±0.01	0.38 ±0.01
+*Spectrum-Pol*	**1.00 ±0.01**	0.012 ±0.016	**1.00 ±0**	**7**e^{-4} ±**5**e^{-4}	**1.00 ±0**	**11**e^{-4} ±**9**e^{-4}
+*Spectrum-Exp*	**1.00 ±0.01**	**0.009 ±0.010**	**1.00 ±0**	**7**e^{-4} ±**5**e^{-4}	**1.00 ±0**	**11**e^{-4} ±**9**e^{-4}
	RTE		WNLI		STS-B	
Model	$I_1(\uparrow)$	$I_2(\downarrow)$	$I_1(\uparrow)$	$I_2(\downarrow)$	$I_1(\uparrow)$	$I_2(\downarrow)$
BERT	0.92 ±0.01	0.39 ±0.02	0.91 ±0.01	0.39 ±0.02	0.95 ±0	0.22 ±0.01
+*Cosreg*	0.92 ±0.01	0.40 ±0.03	0.91 ±0.01	0.40 ±0.01	0.95 ±0.01	0.23 ±0.01
+*Spectrum-Pol*	**1.00 ±0**	**2**e^{-4} ±**1**e^{-4}	**1.00 ±0**	**1**e^{-4} ±**2**e^{-4}	**1.00 ±0**	0.002 ±0
+*Spectrum-Exp*	**1.00 ±0**	3e^{-4} ±2e^{-4}	**1.00 ±0**	2e^{-4} ±3e^{-4}	**1.00 ±0**	**13**e^{-4} ±**6**e^{-4}
+*Flow*	0.92 ±0.01	0.39 ±0.01	0.92 ±0.01	0.39 ±0.02	0.95 ±0.01	0.23 ±0.01
RoBERTa	0.91 ±0.01	0.40 ±0.01	0.91 ±0.01	0.39 ±0.01	0.95 ±0.01	0.23 ±0.01
+*Cosreg*	0.91 ±0	0.41 ±0	0.91 ±0.01	0.40 ±0.01	0.95 ±0	0.23 ±0.01
+*Spectrum-Pol*	**1.00 ±0**	**3**e^{-4} ±**2**e^{-4}	**1.00 ±0**	**3**e^{-4} ±**1**e^{-4}	**1.00 ±0**	**7**e^{-4} ±**3**e^{-4}
+*Spectrum-Exp*	**1.00 ±0**	**3**e^{-4} ±**2**e^{-4}	**1.00 ±0**	**3**e^{-4} ±**1**e^{-4}	**1.00 ±0**	15e^{-4} ±13e^{-4}
DistilBERT	0.92 ±0.01	0.38 ±0.01	0.92 ±0	0.39 ±0.01	0.95 ±0	0.22 ±0.01
+*Cosreg*	0.92 ±0	0.38 ±0.01	0.92 ±0.01	0.38 ±0.01	0.95 ±0	0.22 ±0.01
+*Spectrum-Pol*	**1.00 ±0**	**2**e^{-4} ±**3**e^{-4}	**1.00 ±0**	**1**e^{-4} ±**2**e^{-4}	**1.00 ±0**	**9**e^{-4} ±**1**e^{-4}
+*Spectrum-Exp*	**1.00 ±0**	**2**e^{-4} ±**3**e^{-4}	**1.00 ±0**	**1**e^{-4} ±**2**e^{-4}	**1.00 ±0**	**9**e^{-4} ±**1**e^{-4}
	QNLI		MNLI		QQP	
Model	$I_1(\uparrow)$	$I_2(\downarrow)$	$I_1(\uparrow)$	$I_2(\downarrow)$	$I_1(\uparrow)$	$I_2(\downarrow)$
BERT	0.92 ±0.01	0.39 ±0.01	0.93 ±0.01	0.32 ±0	0.92 ±0.01	0.39 ±0.01
+*Cosreg*	0.92 ±0.01	0.39 ±0.01	0.93 ±0.01	0.32 ±0.01	0.9 ±0	0.39 ±0.01
+*Spectrum-Pol*	0.99 ±0.01	0.06 ±0.02	0.95 ±0.01	0.21 ±0.04	0.92 ±0.02	0.39 ±0.06
+*Spectrum-Exp*	**1.00 ±0**	**5**e^{-4} ±**1**e^{-4}	**0.98 ±0.01**	**0.08 ±0.03**	**0.97 ±0.03**	**0.12 ±0.12**
+*Flow*	0.92 ±0.01	0.39 ±0.01	0.93 ±0	0.31 ±0	0.92 ±0.01	0.39 ±0.01
RoBERTa	0.91 ±0.01	0.40 ±0.01	0.93 ±0.01	0.32 ±0	0.92 ±0.01	0.39 ±0
+*Cosreg*	0.92 ±0.01	0.40 ±0.01	0.93 ±0.01	0.93 ±0.01	0.32 ±0.01	0.39 ±0
+*Spectrum-Pol*	**1.00 ±0**	**0.005 ±0.003**	0.96 ±0.03	0.15 ±0.13	**0.99 ±0.2**	**0.04 ±0.07**
+*Spectrum-Exp*	**1.0 ±0.01**	0.012 ±0.015	**0.98 ±0.01**	**0.10 ±0.04**	**0.99 ±0.01**	**0.04 ±0.06**
DistilBERT	0.92 ±0	0.38 ±0.01	0.93 ±0.01	0.32 ±0	0.92 ±0.1	0.38 ±0.01
+*Cosreg*	0.92 ±0.01	0.39 ±0.01	0.93 ±0	0.32 ±0	0.992 ±0.01	0.39 ±0.01
+*Spectrum-Pol*	0.99 ±0.01	0.03 ±0.04	0.93 ±0.01	0.29 ±0.01	0.93 ±0.03	0.36 ±0.17
+*Spectrum-Exp*	**1.00 ±0.01**	**0.02 ±0.03**	**0.97 ±0.1**	**0.13 ±0.01**	**0.95 ±0.01**	**0.25 ±0.01**

Table 4: Isotropy of the embedding space of the different transformer model and calibration method combinations on GLUE tasks.

D Complete Summarization Results

Here we report the complete summarization results, including the ROUGE-2 and ROUGE-L metrics, omitted in the main text.

	CNN / Daily Mail					
Model	R-1 (↑)	R-2 (↑)	R-L (↑)	$I_2(\uparrow)$	$I_2(\downarrow)$	Time (min)
BART	**38.21 ±0.05**	**17.62 ±0.03**	**27.06 ±0.08**	0.95 ±0.01	0.25 ±0	246 ±8
+*Cosreg*	**38.21 ±0.05**	**17.62 ±0.03**	**27.06 ±0.08**	0.95 ±0.01	0.25 ±0	240 ±8
+*Spectrum-Pol*	37.36 ±0.08	16.60 ±0.08	25.26 ±0.09	**0.99 ±0**	**0.04 ±0**	245 ±20
+*Spectrum-Exp*	37.43 ±0.08	16.62 ±0.01	26.30 ±0.05	**0.99 ±0**	**0.04 ±0**	230 ±18
BART (small dataset)	**36.56 ±0.25**	**15.62 ±0.07**	**25.05 ±0.07**	**0.94 ±0**	**0.25 ±0**	17 ±0
+*Flow*	36.15 ±0.30	15.40 ±0.23	24.79 ±0.19	**0.94 ±0**	**0.25 ±0**	95 ±2

Table 5: Complete BART summariation performance, embedding space isotropy and fine-tuning time per epoch using different calibration methods on the CNN / DailyMail dataset. Due to computational cost, the flow calibration method was tested on a smaller version of the dataset with 20 000 articles.

The performance in terms of ROUGE-2 and ROUGE-L scores follows the same patterns as ROUGE-1. Similar to language understanding and machine translation, increasing isotropy does not improve performance.