Devil's Advocate: Novel Boosting Ensemble Method from Psychological Findings for Text Classification

Hwiyeol Jo[*] [†]
NAVER, Clova AI
hwiyeolj@gmail.com

Jaeseo Lim[†]
Seoul National University
jaeseolim@snu.ac.kr

Byoung-Tak Zhang
Seoul National University
btzhang@bi.snu.ac.kr

Abstract

We present a new form of ensemble method– Devil's Advocate, which uses a deliberately dissenting model to force other submodels within the ensemble to better collaborate. Our method consists of two different training settings: one follows the conventional training process (Norm), and the other is trained by artificially generated labels (DevAdv). After training the models, Norm models are fine-tuned through an additional loss function, which uses the DevAdv model as a constraint. In making a final decision, the proposed ensemble model sums the scores of Norm models and then subtracts the score of the DevAdv model. The DevAdv model improves the overall performance of the other models within the ensemble. In addition to our ensemble framework being based on psychological background, it also shows comparable or improved performance on 5 text classification tasks when compared to conventional ensemble methods.

1 Introduction

Ensemble modeling is a technique that combines several submodels into a composite model. By diminishing model bias, and variance, ensemble techniques can improve overall model performance (Zhou, 2012). In addition, ensemble techniques are also used to get confidence scores of model predictions for explainable models (Haeusler et al., 2013; Li et al., 2014; Vasudevan et al., 2019). For these advantages, ensemble has been used as the de facto standard for many classification tasks.

Ensemble methods such as soft-voting, hard-voting (Hansen and Salamon, 1990), bagging (Breiman, 1996), and boosting (Schapire, 1990) attempt to build submodels which have different views on the same data, which produces more robust predictions.

Research in psychology has shown that a high level of cohesion and group thinking can lead to poor decisions and premature solutions (Janis, 1972; McGrath, 1984; Moorhead et al., 1991). People tend to follow majority in decision making even if the decisions are not reasonable. They are also more likely to rush to judgment and alternatives preferred by the majority (Nemeth, 2018). As Asch (1956) put, 35% of the responses agreed with the majority and nearly everyone followed the incorrect majority at least once. When it comes to group decision making, groups often fall into ideas that are sub-optimal rather than take advantages of using all of the ideas. Parallels can be drawn between this psychological phenomenon and some ensemble methods, especially in cases where the submodels all have similar architectures.

Devil's Advocate is one of the most prominent methods used for fostering healthy dissent in human group decision making (MacDougall and Baum, 1997; Nemeth et al., 2001). It involves taking a position counter to the majority position. That is, Devil's Advocate takes an alternative position from the norms taken for granted in order to deepen the discussion through reasonable opposition. By doing so, the dissenter can increase independence of individuals' thoughts (Nemeth and Nemeth-Brown, 2003). By leveraging this principle from human decision making, we attempt to model the settings of Devil's Advocate and to improve the quality of decision making (in the computational model) and performance.

The contributions of the present study can be summarized as follows[1]:

- We propose an ensemble method, which is theoretically based on psychological background, Devil's Advocate: a reasonable dissent can improve overall group decision making.
- On 5 different text classification datasets, our

[*] Work carried out at Seoul National University
[†] Equal Contribution

[1] http://github.com/HwiyeolJo/DevilsAdvocate

method shows comparable or improved performance when compared to conventional ensemble methods.

2 Preliminary

2.1 Devil's Advocate

Psychologists have made various attempts to improve the quality of decision making. Some tried to raise the quality through increasing the diversity in groups (Chatman et al., 1998). Other researchers have utilized the concept of 'an outsider in group', especially, Devil's advocate (Schweiger et al., 1986; Nemeth et al., 2001). Devil's Advocate is a person who takes a position that does not necessarily agree with the consensus, for the sake of rich discussion. By taking a counter position, the Devil's Advocate engages others in an argumentative discussion to challenge the uniform thought of the majority further, making the participants disagree with the consensus and challenge their point of view. The purpose of this idea is to assess the quality of the original thought and identify errors in argument.

2.2 Ensembles

Voting Algorithms (Hansen and Salamon, 1990); **Soft-Voting** simply involves averaging the prediction scores of submodels. When we train models, the model weights are initialized differently. Due to the effect of random initialization, the models have different views on the same data. **Hard-Voting** is a variation of soft-voting. In hard-voting, the prediction made by the majority of submodels is the resultant ensemble prediction. Although alternative ensemble methods have been developed, these simple voting models remain widely used due to their simplicity and high performance.

Bagging (Bootstrap AGGregatING) (Breiman, 1996) first generates a bootstrap sample from the training dataset. A classifier is then trained from the bootstrap sample. Through repeating this process, the method builds a number of classifiers and averages their prediction scores.

Boosting (Schapire, 1990) links weak classifiers in various ways to build a strong classifier. The main idea is to train a classifier by complementing the weaknesses of the previously trained classifier. Its variations, Adaboost (Freund and Schapire, 1997) and Gradient Boosting (Friedman, 2002), are famous but not widely used in deep learning since boosting requires weak classifiers.

3 Proposed Method: Devil's Advocate

3.1 Training Norm and DevAdv models

Our method requires at least 3 models. Normal models (Norm_n where $n \geq 2$) follow the conventional training process, while one model is used as a Devil's Advocate model (DevAdv). We first train Norm_n models, using a conventional Cross Entropy loss function (CE).

$$\text{TrainLoss}_{\text{Norm}_n} = \text{CE}(\text{Softmax}(\text{Scores}_{\text{Norm}_n}), l_{\text{true}})$$

where $\text{Scores}_{\text{Norm}_n}$ are prediction scores of Norm_n models, and l_{true} refers to true labels, respectively. Conversely, in order to create the DevAdv model, we randomly generate fake labels which do not intersect with the true labels. The generated labels are denoted as false labels (l_{false}). The loss function of DevAdv is as follows:

$$\text{TrainLoss}_{\text{DevAdv}} = \sum^{C-1} \text{CE}(\text{Softmax}(\text{Scores}_{\text{DevAdv}}), l_{\text{false}})$$

where C is the number of labels. Since the DevAdv model is trained using false labels, the model serves the Devil's Advocate, disagreeing with the prediction scores of the other models. Furthermore, the fake labels are randomly generated in each epoch, allowing the DevAdv model to offer a different view on the data with each training iteration.

In early-stopping, the validation performance of the DevAdv model is checked by assessing whether argmin ($\text{Scores}_{\text{DevAdv}}$) is the true label.

3.2 Group Discussion: Fine-tuning

For fine-tuning, we adopt an approach inspired by experiments of the human group decision making (i.e., group discussion) used in the original Devil's Advocate work. With the trained models (Norm_1, Norm_2, DevAdv), we design additional loss function as follows:

$$\begin{aligned}\text{DiscussLoss}_{\text{Norm}_1} = \\ \text{CE}(\text{Scores}_{\text{Norm}_1} + \text{Softmax}(\text{Scores}_{\text{DevAdv}}), l_{\text{true}}) \\ + \text{MSE}(\text{Scores}_{\text{Norm}_1}, \text{Scores}_{\text{Norm}_2})\end{aligned}$$

$$\begin{aligned}\text{DiscussLoss}_{\text{Norm}_2} = \\ \text{CE}(\text{Scores}_{\text{Norm}_2} + \text{Softmax}(\text{Scores}_{\text{DevAdv}}), l_{\text{true}}) \\ + \text{MSE}(\text{Scores}_{\text{Norm}_2}, \text{Scores}_{\text{Norm}_1})\end{aligned}$$

	DBpedia	Yahoo	Yelp	AGNews	IMDB
#Train/#Test	560K/70K	≈ 133K/24K	650K/50K	120K/7.6K	25K/25K
#Class	14	17	5	4	2

Table 1: The data information used in text classification.

Ensemble Method	DBpedia	Yahoo	Yelp	AGNews	IMDB
Single Model	98.44±.09	73.25±.27	63.24±.17	91.75±.16	89.97±.19
3Models-Soft-Voting	98.83±.05	75.33±.18	**64.60±.09**	92.44±.15	90.84±.16
3Models-Hard-Voting	98.78±.02	75.18±.14	64.02±.06	92.23±.14	90.74±.13
3Models-Bagging	**98.85±.03**	75.10±.23	64.40±.20	92.00±.14	90.24±.07
Devil's Adv. Ensemble (Ours)	98.84±.03	**76.26±.10**	64.58±.19	**92.71±.12**	**90.88±.10**
3Models-Soft-Voting +EmbPerturb	**98.91±.02**	75.69±.24	64.14±.45	92.53±.10	**90.99±.06**
Devil's Adv. Ens. +EmbPerturb	98.86±.00	**75.75±.40**	**64.70±.44**	**92.79±.08**	90.69±.10

Table 2: 5 times average performance on text classification datasets. Our method shows on par with or improved performance when compared to conventional ensemble methods. EmbPertub means the use of Miyato et al. (2016).

The model weights of the DevAdv model are fixed to prevent DevAdv from being trained like Norm. Also, softmax normalization is not applied to Norms' scores, not to limit the scores from 0 to 1; but to make the scores much higher than normalized DevAdv's score. Through CE loss, the DevAdv model prevents Norm models from being correctly fitted to the true labels. However, during the training process, Norm models eventually learn to correctly predict the true labels, even despite the disturbance by the DevAdv model. In the second MSE term of the above equation, each Norm model enhances the others with information (experience) learned from the first term. This term also prevents the models from catastrophic forgetting. With this loss function, we train the models again using the same train set. As a result, we expect to result in a more diverse range of views on the data.

When reporting the performance on the test set, we follow the soft-voting ensemble but utilize the DevAdv model by using its prediction scores reversely: $\text{argmax}(\sum_n^N \text{Scores}_{\text{Norm}_n} - \text{Scores}_{\text{DevAdv}})$.

4 Experiment

Data. We use GloVe (Pennington et al., 2014) as pretrained embeddings. To increase model performance, we apply a word vector post-processing method called extrofitting (Jo and Choi, 2018).

We prepare 3 topic classification datasets; DBpedia ontology (DBpedia) (Lehmann et al., 2015), YahooAnswers (Yahoo) (Chang et al., 2008), AGNews. We also prepared 2 sentiment classification datasets; Yelp reviews (Yelp) (Zhang et al., 2015), IMDB (Maas et al., 2011). The data information is presented in Table 1. Additionally, we seperate 15% from the training set of each dataset to create validation sets for all datasets. The validation set is used for early-stopping. We use all words as inputs, including all special symbols in a 300 dimensional embedding space.

Classifier. We choose TextCNN (Kim, 2014) as the submodel architecture of our proposed ensemble method. The model has two convolutional layers with 32 channels and 16 channels, respectively. We adopt multiple sizes of kernels–2, 3, 4, and 5, followed by ReLU activation (Hahnloser et al., 2000) and max-pooling. We concatenate the output after every max-pooling layer. We optimize the model parameters using Adam (Kingma and Ba, 2014) with a 1e-3 learning rate. We use 1 DevAdv model and 2 Norm models as a default.

Baseline Implementation. Soft-Voting is implemented by averaging the model prediction scores. Hard-Voting is implemented by selecting the majority predictions. The sampling rate of Bagging is 70% of training data with replacement, ensuring all the data being used at least once. Boosting cannot be compared as a baseline because our method consist of a single model architecture. In order to show the difference with Miyato et al. (2016), we report the performance with the embedding perturbation.

5 Result

The performance of our proposed ensemble methods is presented in Table 2. We confirm that our ensemble method is most effective when the dataset is relatively small. However, our method performs

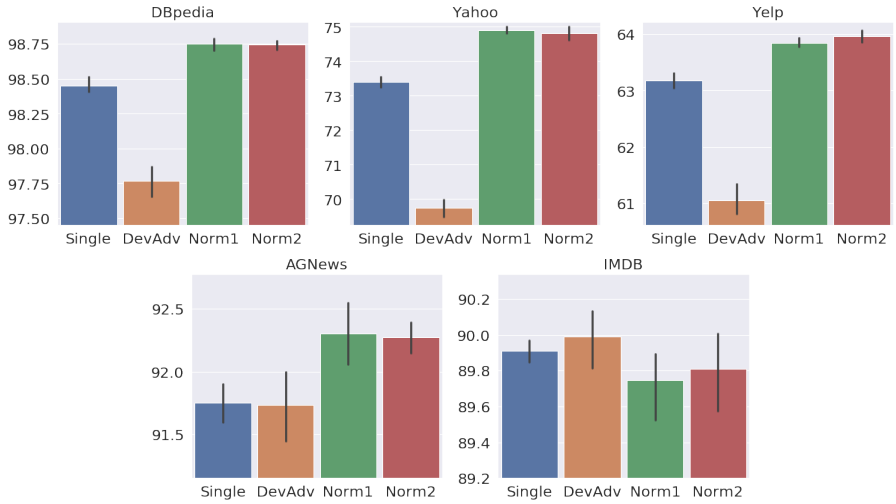

Figure 1: The performance of Single, DevAdv, Norm_1, and Norm_2 models. We confirm that DevAdv model provides further improvements to other models (Single $\rightarrow$ Norm_1, Single $\rightarrow$ Norm_2).

on par with soft-voting even on relatively large datasets. In addition, the performance gap between them on large datasets is within the error bounds.

We present the performance of the ensemble models on each dataset in Figure 1. With the help of DevAdv, Norm_1 and Norm_2 models perform much better than single model on most of the datasets. That is, the scores of the DevAdv model force the other classifiers to be improved in order to counteract this noise. This principle is also the core idea behind boosting.

On the IMDB dataset, DevAdv model does not augment the performance of the Norm models. Since IMDB has only 2 classes, the training process of the DevAdv model is not different from a conventional training process.

6 Related Work

Although our boosting method is inspired by the psychological background, Devil's Advocate, its implementation is related to **Data Augmentation** (in particular, **Negative Sampling** (Mikolov et al., 2013)), and **Adversarial Training** in terms of training DevAdv and fine-tuning, respectively.

Data augmentation is used in many machine learning tasks to artificially enlarge the size of the training set. In the text domain, using synonyms (Zhang et al., 2015), back translation (Sennrich et al., 2016), and paraphrasing (Kumar et al., 2019) have been proposed. However, these methods are only moderately effective since the meaning of words are sensitive to modification. Instead, we use a model trained through negative sampling.

Our method can then be compared to adversarial training, which uses a negative model to make other models more robust towards adversarial examples. However, as far as we know, Miyato et al. (2016) is the only work using an adversarial training framework for text classification. They used an adversarial training process at embedding-level, from the beginning of model training. In contrast, our proposed method utilizes a pretrained negative model to fine-tune other models. Furthermore, our negative model contributes to the final prediction, resulting in further improvements (see Table 3).

7 Ablation Studies

Training and Inference The false labels generated artificially serve to augment the data used for training the DevAdv model, which is trained using exclusively false labels. By limiting the number of false labels to 1, we confirm the effect of data augmentation. Table 3 shows that the effect of data augmentation is important when the number of classes in the dataset is large. On the other hand, datasets which have small numbers of classes (e.g., IMDB) are less affected.

Next, we remove the group discussion stage, which fine-tunes the Norm models interactively. By this ablation, we can see the effect of adversarial training, which trains a model in an unconventional way by using a negative model. The group discussion process (adversarial training) shows positive effects on performance except for Yelp. However, the performance gap is within the error range.

We also see that our method can be used with

Ablation	DBpedia	Yahoo	Yelp	AGNews	IMDB
Devil's Adv. Ensemble	**98.84±.03**	**76.26±.10**	64.58±.19	**92.71±.12**	90.88±.10
(−)DevAdv	98.71±.07	75.13±.21	64.34±.11	92.44±.11	90.66±.14
(−)Data Augmentation	98.81±.02	76.01±.20	64.49±.10	92.53±.09	**90.89±.17**
(−)DiscussLoss	98.76±.02	75.14±.12	**64.64±.15**	92.11±.13	90.63±.23
4Models-Soft-Voting	98.90±.04	76.37±.45	65.07±.15	92.68±.20	90.66±.25
Devil's Adv. Ens.(+)Norm$_3$	98.93±.03	76.50±.30	65.01±.03	92.80±.08	91.10±.08

Table 3: Ablation studies on the number of fake labels (data augmentation) and presence of group discussion (adversarial training). We also present the performance when the DevAdv model does not involve.

Model	Ensemble	DBpedia	Yahoo	Yelp	AGNews	IMDB
SmallCNN	Soft-Voting	98.60±.02	74.92±.47	63.51±.11	91.95±.12	90.57±.44
	Devil's Adv. Ens.	98.68±.03	76.10±.10	63.42±.23	92.42±.12	90.66±.10
Transformers	Soft-Voting	98.89±.02	71.72±.51	61.33±.20	91.20±.23	84.94±.22
	Devil's Adv. Ens.	98.83±.04	78.86±.12	61.45±.31	91.58±.05	84.76±.36

Table 4: The result of Devil's Advocate Ensemble on different model architectures: small sized CNN, and Transformers. Note that there is no advantage of DevAdv in IMDB dataset, which has only 2 classes.

more than 3 models (see Table 3). When we use KL divergence instead of MSE in discussion loss it slightly degrades the performance.

Model Architecture The small sized TextCNN (**SmallCNN**) model consists of multi-kernels which size is [2,3] (instead of [2,3,4,5]). Also, we reduce channel size from [32, 16] to [32], which has 1-depth convolutional layer only. The result is presented in Table 4. We also provide the performance of **Transformers** (Vaswani et al., 2017)-based model performance (see Table 4). The transformer classifier has the maximum 512 sequence length with 300 embedding dimensions and positional-embeddings. It also has 10 multi-head attentions but uses 1 encoder. Stacking more encoder layers harms the performance. The hyperparameters of these models are the same as those of main experiment with TextCNN.

Similar to the previous experiment, the performances on other models are on par with soft-voting. Nevertheless, the results indicates that our proposed ensemble (Devil's Advocate) can be applied to any kinds of model architecture. It is also interesting that Transformers shows overfitting on Yahoo dataset, but DevAdv makes the model being generalized.

8 Conclusion

In this paper, we propose a novel boosting ensemble approach, inspired by the Devil's Advocate. In addition to the implementation of the psychological background, the framework is designed to make submodels better collaborate with each other.

We first train a model with incorrect labels in order to make the model serves as Devil's Advocate (DevAdv), and the DevAdv interacts with the other conventionally trained models. In the experiments, we show DevAdv model improves performance of the other conventionally trained models.

Although the proposed models' performance does not significantly outperform other ensemble methods, we believe that our new ensemble approach makes valuable contributions to the future research: the use of negative model by taking advantages of data augmentation and adversarial training to provide different views of the same dataset, and the implementation of psychological-motivated idea can be properly applied to the NLP field/machine learning domain.

Acknowledgement

This work was partly supported by the Institute for Institute of Information & Communications Technology Planning & Evaluation (2015-0-00310-SW.StarLab/20%, 2017-0-01772-VTT/20%, 2018-0-00622-RMI/20%, 2019-0-01371-BabyMind/20%) grant funded by the Korean government. It was also supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2021R1A6A3A13039453)

References

Solomon E Asch. 1956. Studies of independence and conformity: I. a minority of one against a unanimous majority. *Psychological monographs: General and applied*, 70(9):1.

Leo Breiman. 1996. Bagging predictors. *Machine learning*, 24(2):123–140.

Ming-Wei Chang, Lev-Arie Ratinov, Dan Roth, and Vivek Srikumar. 2008. Importance of semantic representation: Dataless classification. In *AAAI*, volume 2, pages 830–835.

Jennifer A Chatman, Jeffrey T Polzer, Sigal G Barsade, and Margaret A Neale. 1998. Being different yet feeling similar: The influence of demographic composition and organizational culture on work processes and outcomes. *Administrative Science Quarterly*, pages 749–780.

Yoav Freund and Robert E Schapire. 1997. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139.

Jerome H Friedman. 2002. Stochastic gradient boosting. *Computational statistics & data analysis*, 38(4):367–378.

Ralf Haeusler, Rahul Nair, and Daniel Kondermann. 2013. Ensemble learning for confidence measures in stereo vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 305–312.

Richard HR Hahnloser, Rahul Sarpeshkar, Misha A Mahowald, Rodney J Douglas, and H Sebastian Seung. 2000. Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit. *Nature*, 405(6789):947.

Lars Kai Hansen and Peter Salamon. 1990. Neural network ensembles. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (10):993–1001.

Irving L Janis. 1972. Victims of groupthink: A psychological study of foreign-policy decisions and fiascoes.

Hwiyeol Jo and Stanley Jungkyu Choi. 2018. Extrofitting: Enriching word representation and its vector space with semantic lexicons. *ACL 2018*, page 24.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Ashutosh Kumar, Satwik Bhattamishra, Manik Bhandari, and Partha Talukdar. 2019. Submodular optimization-based diverse paraphrasing and its effectiveness in data augmentation. 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 3609–3619, Minneapolis, Minnesota. Association for Computational Linguistics.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. 2015. Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195.

Leijun Li, Qinghua Hu, Xiangqian Wu, and Daren Yu. 2014. Exploration of classification confidence in ensemble learning. *Pattern recognition*, 47(9):3120–3131.

Andrew L Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1*, pages 142–150. Association for Computational Linguistics.

Colin MacDougall and Frances Baum. 1997. The devil's advocate: A strategy to avoid groupthink and stimulate discussion in focus groups. *Qualitative health research*, 7(4):532–541.

Joseph Edward McGrath. 1984. *Groups: Interaction and performance*, volume 14. Prentice-Hall Englewood Cliffs, NJ.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Takeru Miyato, Andrew M Dai, and Ian Goodfellow. 2016. Adversarial training methods for semi-supervised text classification. *arXiv preprint arXiv:1605.07725*.

Gregory Moorhead, Richard Ference, and Chris P Neck. 1991. Group decision fiascoes continue: Space shuttle challenger and a revised groupthink framework. *Human Relations*, 44(6):539–550.

C Nemeth and Brendan Nemeth-Brown. 2003. Better than individuals. *Group creativity: Innovation through collaboration*, 4:63–84.

Charlan Nemeth, Keith Brown, and John Rogers. 2001. Devil's advocate versus authentic dissent: Stimulating quantity and quality. *European Journal of Social Psychology*, 31(6):707–720.

Charlan Jeanne Nemeth. 2018. *In defense of troublemakers: The power of dissent in life and business*. Basic Books.

Jeffrey Pennington, Richard Socher, and Christopher 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.

Robert E Schapire. 1990. The strength of weak learnability. *Machine learning*, 5(2):197–227.

David M Schweiger, William R Sandberg, and James W Ragan. 1986. Group approaches for improving strategic decision making: A comparative analysis of dialectical inquiry, devil's advocacy, and consensus. *Academy of management Journal*, 29(1):51–71.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96.

Vishal Thanvantri Vasudevan, Abhinav Sethy, and Alireza Roshan Ghias. 2019. Towards better confidence estimation for neural models. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7335–7339. IEEE.

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.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, pages 649–657.

Zhi-Hua Zhou. 2012. *Ensemble methods: foundations and algorithms*. Chapman and Hall/CRC.