NLP-CUET@LT-EDI-EACL2021: Multilingual Code-Mixed Hope Speech Detection using Cross-lingual Representation Learner

Eftekhar Hossain*, Omar Sharif† and Mohammed Moshiul Hoque†
†Department of Computer Science and Engineering
*Department of Electronics and Telecommunication Engineering
Chittagong University of Engineering and Technology, Bangladesh
{eftekhar.hossain, omar.sharif, moshiul_240}@cuet.ac.bd

Abstract

In recent years, several systems have been developed to regulate the spread of negativity and eliminate aggressive, offensive or abusive contents from the online platforms. Nevertheless, a limited number of researches carried out to identify positive, encouraging and supportive contents. In this work, our goal is to identify whether a social media post/comment contains hope speech or not. We propose three distinct models to identify hope speech in English, Tamil and Malayalam language to serve this purpose. To attain this goal, we employed various machine learning (support vector machine, logistic regression, ensemble), deep learning (convolutional neural network + long short term memory) and transformer (m-BERT, Indic-BERT, XLNet, XLM-Roberta) based methods. Results indicate that XLM-Roberta outdoes all other techniques by gaining a weighted f_1-score of 0.93, 0.60 and 0.85 respectively for English, Tamil and Malayalam language. Our team has achieved 1^{st}, 2^{nd} and 1^{st} rank in these three tasks respectively.

1 Introduction

Nowadays, online and social media platforms have enormous influence and impact on people's societal life. When people undergo a challenging or unfavourable time, they start to find emotional support from their friends, relatives or even virtual platforms to overwhelm this situation. Due to Covid-19 pandemic, various online forums have become a popular medium of seeking help, suggestion or support. Thus, researchers are trying to develop a computational model that can find positive and supportive social media information. In general, the hope speech contains words of inspiration, promise and suggestions. Chakravarthi (2020) considered those words as hope speech that offer suggestions, reassurance, support, insight and inspiration. Hope speech can be beneficial to save individuals who wish to harm themselves or even attempt to suicide. Such speech inspires people during the period of depression, loneliness and stress with the words of promise, suggestions and support (Herrestad and Biong, 2010). The major concern of this research is to originate a computational model on top of this dataset to identify hope speech from the social media posts/comments. Lack of resources on hope speech research, scarcity of training corpora and multilingual code-mixing are the key concerns to develop such models.

Machine learning (ML), and deep learning (DL) based techniques can be utilized to address the problem of hope speech detection. In recent years, transformers have gained immense popularity due to its ability to handle the dependencies between input and output with both attention and recurrence. Consequently, many NLP tasks have accomplished using the transformer-based model to obtain the state-of-the-art performance (Chen et al., 2021). The principal contributions in this research as listed below:

- Develop a model with cross-lingual contextual word embeddings (i.e. transformers) to identify the hope speech considering the code-mixed data for English, Tamil and Malayalam languages.
- Investigated the superiority of various ML, DL and transformer-based techniques with detail experimentation.

The rest of the paper organized as follows: works related to hope speech detection discussed in Section 2. Task and dataset are described in detail in Section 3. Section 4 explains the various techniques used to develop the model for performing the assigned task. Experimental

Proceedings of the First Workshop on Language Technology for Equality, Diversity and Inclusion, pages 168–174

findings and error analysis of the models are introduced in Section 5.

2 Related Work

With the substantial growth of the Internet and online contents, several methods have been developed to identify, classify and stop the expansion of negativity such as hate speech detection (Mandl et al., 2020; Chakravarthi et al., 2020), hostility detection (Sharif et al., 2021), aggressive language identification (Kumar et al., 2020), and flagging abusive contents (Akiwowo et al., 2020). However, very few researchers have put their focus on the other side that is hope speech detection. A little work has conducted till to date in this research avenue of NLP. Palakodety et al. (2019a) analyzed how hope speech can be utilized to mitigate tension between two rival (Pakistan and India) countries. Supporting texts regarding Rohingya community culled from social media in Hindi and English languages (Palakodety et al., 2019b). However, details of the dataset such as inter-annotator agreement, diversity of annotators were not clearly described. Chakravarthi (2020) developed a multilingual code-mixed hope speech dataset for Equality, Diversity and Inclusion (HopeEDI) in English, Tamil and Malayalam language. Data collected from social media like Facebook, YouTube in trending topics, i.e. COVID-19, LGBTIQ issues, and India-China war. Their models achieved the highest weighted f_1 score of 0.90, 0.56, 0.70 with a decision tree, naive Bayes, logistic regression techniques for English, Tamil and Malayalam languages.

3 Task and Dataset Descriptions

In this shared task, we have to perform multi-class classification where we aim to identify whether a given comment contains hope speech or not. Our system goal is to classify a post/comment into one of the three predefined classes: hope speech (HS), not hope speech (NHS) and not intended language (NIL). The shared task organizers (Chakravarthi and Muralidaran, 2021) developed a hope speech corpus in multilingual code-mixed setup. A total of 28541 (for English), 20198 (for Tamil) and 10705 (for Malayalam) texts are available in the corpus. This corpus partitioned into three independent sets: train, validation and test. Initially, the model is developed on top of the train set, and model hyperparameters are tuned based on the validation set's performance. Finally, the model evaluated on the unseen instances of the test set. Table 1 shows detail statistics of train, validation and test set for each class.

Further investigation on the training set revealed that the training set is highly imbalanced where several documents in 'not intended language' class are much lower than 'hope speech' and 'not hope speech' classes. The average number of words in 'not intended language' is approximately four words for Tamil and Malayalam languages. The model generalization capability on unseen data might degrade due to the lower number of examples on these classes. Detail analysis of the training set presented in table 2.

4 Methodology

This section provides a brief discussion of the schemes and techniques employed to address the task (Section 3). Initially, different feature extraction techniques are exploited with machine learning, and deep learning approaches for a baseline evaluation. Moreover, we also applied transformers to obtain a better outcome. Figure 1 shows the abstract process of hope speech detection. Architectures and parameters of different approaches are described in the following subsections.

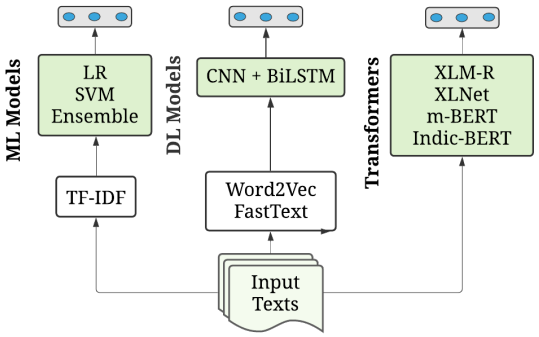

Figure 1: Abstract process of hope speech detection

4.1 Feature Extraction Techniques

ML and DL based techniques are incapable of processing strings or plain text from the raw forms. Thus, extracting of appropriate or relevant features is a prerequisite to train ML and DL based systems. We utilize TF-IDF (Tokunaga and Makoto, 1994), and FastText (Bojanowski et al., 2016) techniques for extracting features from the texts.

	English			Tamil			Malayalam		
	HS	**NHS**	**NIL**	**HS**	**NHS**	**NIL**	**HS**	**NHS**	**NIL**
Train	1962	20778	22	6327	7872	1961	1668	6205	691
Valid	272	2569	2	757	998	263	190	784	96
Test	250	2593	3	815	946	259	194	776	101

Table 1: Number of instances in train, validation and test sets for each language. Here HS, NHS and NIL indicates hope speech, not hope speech and not intended language respectively.

Language	**Classes**	**Total words**	**Unique words**	**Max. length (words)**	**Avg. words (per text)**
English	HS	49210	4811	197	25.08
	NHS	317854	19740	191	15.29
	NIL	325	239	47	14.77
Tamil	HS	56000	17274	193	8.85
	NHS	76302	23977	176	9.69
	NIL	7309	2093	48	3.72
Malayalam	HS	25144	11827	96	15.07
	NHS	60313	24607	95	9.72
	NIL	2644	1040	35	3.82

Table 2: Training set statistics for each language. Here HS, NHS and NIL indicates hope speech, not hope speech and not intended language respectively.

TF-IDF: TF-IDF is a measure that calculates the relevancy of a word to a document in a collection of documents. We calculate the TF-IDF value of unigram features for all the languages. During the calculation minimum and maximum document frequency value set at 1.

Word Embedding: Embedding features can capture the semantic meaning of a word. To get embeddings features for all the languages, we used Keras embedding layer with embedding size 100. During the training phase, Keras tries to find the optimal values of the embedding layer's weight matrix by doing simple matrix multiplication and thus create a mapping of each unique words into a vector of real numbers. We utilize the full vocabulary of the corpus and choose maximum input text length 100, 50, and 80 respectively for English, Tamil, and Malayalam data.

FastText: To alleviate the problem of out of vocabulary words in keras embeddings, we use FastText embedding. Instead of learning vectors directly for words, FastText represents each word as n-gram of characters. Therefore even if a word was not encountered during training, it could be split into n-gram to get its embedding. Pre-trained (Grave et al., 2018) embedding vectors are used to accomplish the tasks of each language. We retain the default embedding dimension 300 for FastText embedding.

4.2 ML Baselines

To address the problem, we investigate the performance of three traditional ML approaches, including logistic regression (LR), support vector machine (SVM) and Ensemble. Scikit-learn library is employed for the implementation of these models. TF-IDF features are used to train all the ML methods for three languages.

LR: LR is constructed by using 'lbfgs' solver along with 'l2' penalty. The regularization parameter C settled to 2, 5, and 1 respectively for English, Tamil, and Malayalam data.

SVM: For SVM, 'linear' kernel is utilized with C value of 10, 1 and 0.5 respectively for English, Tamil, and Malayalam language.

Ensemble: To perform classification task ensemble approach has proven superior compared to individual models outcome (Roy et al., 2018). We employ decision tree (DT) and random forest (RF) classifiers along with SVM and LR to develop an ensemble method. For RF, 100 'n_estimators' is chosen while 'gini' criterion used for both DT and RF. On the other hand, previously mentioned

parameters have retained in LR and SVM. The majority voting technique is utilized to get the prediction from the ensemble approach.

4.3 DL Baselines

A deep learning-based approach is applied with word embedding features to address the task. The model is developed in TensorFlow backend by using Keras library. The combination of convolutional neural network (CNN) and bidirectional long short term memory (BiLSTM) has achieved an outstanding result in many NLP tasks (Sharif et al., 2020). In this approach, we employ one BiLSTM layer on top of a convolution layer. Initially embedding features are feed to the CNN layer consisting of 128 filters. Following this, to choose appropriate features, a max-pooling is applied with window size 5. The resultant vector is then passed into the BiLSTM layer. In order to capture long term dependencies, 100 bidirectional cells are used in this layer. To mitigate the chance of overfitting BiLSTM layer dropout technique is utilized with a dropout rate of 0.2. Afterwards, the concatenated output of the BiLSTM layer transferred into a softmax layer for the prediction.

4.4 Transformers

We employed four pre-trained transformer models such as multilingual bidirectional encoder representations from transformers (m-BERT), Indic-BERT, XLNet, and XLM-Roberta (XLM-R) and fine-tuned them on the dataset with varying hyperparameters. For fine-tuning, maximum text length settled to 50 for Tamil and 100 for Malayalam and English. The models are fetched from Huggingface[1] transformers library and implemented using Ktrain (Maiya, 2020) package.

m-BERT: m-BERT (Devlin et al., 2018) is pre-trained on a large corpus of multilingual data. To accomplish our purpose, we employed 'bert-base-multilingual-cased' model and fine-tuned it on our dataset with batch size 12.

Indic-BERT: Indic-BERT (Kakwani et al., 2020) is a multilingual model pre-trained specifically on 12 major Indian languages. It has fewer parameters than other multi-lingual models (i.e. m-BERT, XLM-R). Nevertheless, it outperforms other transformers on various task (Kulkarni et al., 2021). The model is fine-tuned with the batch size of 8.

XLNet: XLNet (Yang et al., 2019) is an auto-regressive language model which utilizes the recurrence to output the joint probability of a sequence of words. It combines transformer mechanism with slight modification in language modelling approach. For the implementation 'xlnet-base-cased' model is used and for all the languages we choose batch size 12 for fine-tuning.

XLM-Roberta: XLM-R (Conneau et al., 2019) is referred as cross lingual representation learner. It is a multi-lingual transformer-based model pre-trained with more that 100 languages and achieves the state-of-the-art performance on cross-lingual NLP tasks. We used 'xlm-Roberta-base' model and select a batch size of 4 to fine-tuned it on our datasets.

All transformer models are fine-tuned using Ktrain 'fit_onecycle' method, trained for 30 epochs with a learning rate of $2e^{-5}$. The early stopping technique is employed to avoid the overfitting problem.

5 Results and Analysis

This section presents a comprehensive performance analysis of various machine learning, deep learning and transformer models for three languages (English, Tamil and Malayalam). Weighted f_1 score uses to determine the excellence of the models. In some cases, other evaluation metrics like precision and recall also considered. Table 3 presents the evaluation results of all models on the test set. It observed that ensemble achieved the highest f_1-score of 0.905 and 0.573 respectively for English and Tamil data in ML models. On the other hand, maximum f_1-score of 0.813 is obtained by SVM for the Malayalam language. For all the languages, LR also performed quite similar to SVM but failed to beat other models.

In deep learning, the combination of CNN and BiLSTM experiments with two different embedding features (i.e. Keras embedding and FastText). Both models obtained the highest f_1-score of around 0.90 for English while achieved approximately 0.79 for Malayalam dataset. However, the combination of CNN-BiLSTM model with FastText embedding features shows 1% rise in f_1-score (from 0.54 to 0.55) for Tamil language.

[1] https://huggingface.co/transformers/

Method	Classifiers	English			Tamil			Malayalam		
		P	R	F	P	R	F	P	R	F
ML	LR	0.914	0.869	0.886	0.569	0.563	0.562	0.815	0.798	0.804
	SVM	0.915	0.877	0.892	0.582	0.574	0.564	0.820	0.809	0.813
	Ensemble	0.904	0.920	0.905	0.584	0.584	0.573	0.80	0.811	0.794
DL	C + L (KE)	0.906	0.892	0.899	0.569	0.559	0.540	0.806	0.796	0.791
	C + L (FT)	0.898	0.899	0.898	0.565	0.557	0.548	0.789	0.788	0.786
Trans	m-BERT	0.928	0.927	0.928	0.588	0.591	0.588	0.808	0.823	0.804
	Indic-BERT	0.913	0.920	0.910	0.593	0.592	0.578	0.839	0.842	0.840
	Xlnet	0.931	0.929	0.930	0.558	0.560	0.558	0.779	0.797	0.781
	XLM-R	**0.931**	**0.931**	**0.931**	**0.610**	**0.609**	**0.602**	**0.859**	**0.852**	**0.854**

Table 3: Performance comparison of different models on test set where P, R, F denotes precision, recall and weighted f_1-score. Here, C+L means the combination of CNN and BiLSTM method and KE and FT represents Keras and FastText embeddings.

On the other hand, Transformer based models showed remarkable performance for all three languages. In English data, m-BERT, XLNet, and XLM-R got the highest f_1-score of approximately 0.93. However, considering both the precision and recall values, only XLM-R outperformed the other models. For the Tamil language, f_1-score of around 0.56 and 0.58 respectively obtained by XLNet and Indic-BERT. A slight rise of 1% in f_1-score (0.588) is noticed for m-BERT but it cannot beat XLM-R performance (f_1 score = 0.602). In case of Malayalam data, Indic-BERT (f_1 score = 0.84) shows an increase of 4% to 6% than the f_1-score obtained by m-BERT (0.804) and XLNet (0.781). Nevertheless, it failed to reach the outcome of XLM-R (f_1 score = 0.854), which outdoes all the models.

The results show that **XLM-R** outperformed ML, DL, and other transformer-based models for all three languages. The ability of cross-lingual understanding at different linguistic levels might be the reason for this superior performance of XLM-R.

5.1 Error Analysis

It is evident from the Table 3 that XLM-R is the best performing model to detect hope speech for English, Tamil and Malayalam languages. A detail error analysis is carried out using the confusion matrix to investigate more insights concerning the individual class performance (Figure 2). From the Figure 2(a), it is noticed that among 250 HS instances, 93 are misclassified as NHS. However, the model ultimately failed to detect any of the not-English data and incorrectly classified them as NHS. Similarly, 81 (out of 815) HS and 76 (out of 946) NHS are misclassified as a not-Tamil class in the Tamil language. However, the misclassification rate is comparatively low for nT class. Figure 2(c) shows that the model correctly classified 143 HS (out of 194) instances. Moreover, it wrongly classified 47 samples (as NHS) and 4 samples (as nM). Likewise, among 101 instances of NM, the model misclassified 20 texts while correctly identifying 81 instances. On the other hand, the NHS class received 668 correct classification out of 756 instances, and only misclassified 88 instances as other classes.

We noticed that the model mostly gets confused with HS and NHS class in all languages from the error analysis. The possible reason is that there may be plenty of code-mixed words common in both classes. Thus the system could not apprehend the inherent meaning of the sentences. The high-class imbalance may be another likely reason why the model gives the most priority to the not hope speech class and therefore incorrectly classified hope speech as not hope speech. Increasing the number of instances in the NHS class may mitigate the chance of excessive misclassification.

6 Conclusion

This paper describes and analyses the several ML, DL, and transformer-based methods that we have adopted to participate in the hope speech detection shared task at EACL 2021. Employing TF-IDF, embedding features initially, we performed experiments with ML (LR, SVM, ensemble) and DL (CNN+BiLSTM) approaches. The outcome shows that the ensemble technique achieved

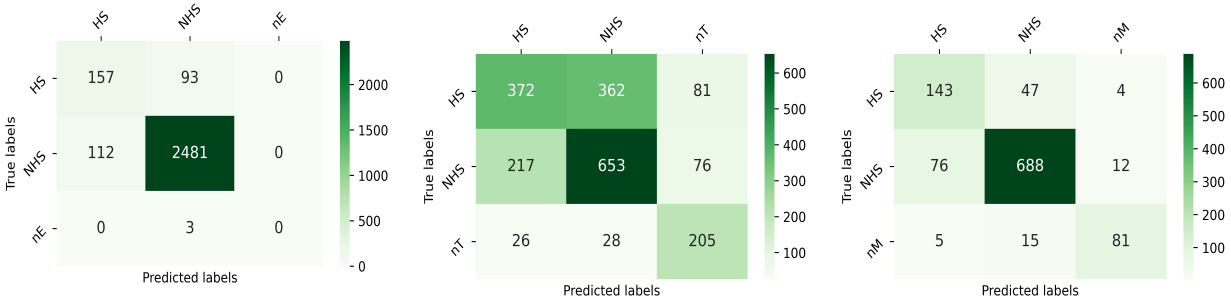

Figure 2: Confusion matrix of XLM-R technique for (a) English, (b) Tamil and (c) Malayalam languages.

higher performance compared to other ML/DL models. Further, transformer-based techniques are employed to improve the overall performance. The XML-R model outperformed all the models' performance by achieving the highest weighted f_1-score of 0.931, 0.854, and 0.602 respectively for English, Tamil Malayalam language. In the future, contextualized embeddings (such as ELMO, FLAIR) and transformers ensemble might explore to investigate the system's performance.

References

Seyi Akiwowo, Bertie Vidgen, Vinodkumar Prabhakaran, and Zeerak Waseem, editors. 2020. *Proceedings of the Fourth Workshop on Online Abuse and Harms*. Association for Computational Linguistics, Online.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomás Mikolov. 2016. Enriching word vectors with subword information. *CoRR*, abs/1607.04606.

Bharathi Raja Chakravarthi. 2020. HopeEDI: A multilingual hope speech detection dataset for equality, diversity, and inclusion. In *Proceedings of the Third Workshop on Computational Modeling of People's Opinions, Personality, and Emotion's in Social Media*, pages 41–53, Barcelona, Spain (Online). Association for Computational Linguistics.

Bharathi Raja Chakravarthi, M Anand Kumar, John Philip McCrae, Premjith B, Soman KP, and Thomas Mandl. 2020. Overview of the track on HASOC-Offensive Language Identification-DravidianCodeMix. In *Working Notes of the Forum for Information Retrieval Evaluation (FIRE 2020). CEUR Workshop Proceedings. In: CEUR-WS. org, Hyderabad, India*.

Bharathi Raja Chakravarthi and Vigneshwaran Muralidaran. 2021. Findings of the shared task on Hope Speech Detection for Equality, Diversity, and Inclusion. In *Proceedings of the First Workshop on Language Technology for Equality, Diversity and Inclusion*. Association for Computational Linguistics.

Ben Chen, Bin Chen, Dehong Gao, Qijin Chen, Chengfu Huo, Xiaonan Meng, Weijun Ren, and Yang Zhou. 2021. Transformer-based language model fine-tuning methods for covid-19 fake news detection.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomás Mikolov. 2018. Learning word vectors for 157 languages. *CoRR*, abs/1802.06893.

Henning Herrestad and Stian Biong. 2010. Relational hopes: A study of the lived experience of hope in some patients hospitalized for intentional self-harm. *International journal of qualitative studies on health and well-being*, 5(1):4651.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, Gokul N.C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. IndicNLPSuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4948–4961, Online. Association for Computational Linguistics.

Atharva Kulkarni, Meet Mandhane, Manali Likhitkar, Gayatri Kshirsagar, Jayashree Jagdale, and Raviraj Joshi. 2021. Experimental evaluation of deep learning models for marathi text classification.

Ritesh Kumar, Atul Kr. Ojha, Bornini Lahiri, Marcos Zampieri, Shervin Malmasi, Vanessa Murdock, and Daniel Kadar, editors. 2020. *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*. European Language Resources Association (ELRA), Marseille, France.

Arun S. Maiya. 2020. ktrain: A low-code library for augmented machine learning.

Thomas Mandl, Sandip Modha, Anand Kumar M, and Bharathi Raja Chakravarthi. 2020. Overview of the HASOC Track at FIRE 2020: Hate Speech and Offensive Language Identification in Tamil, Malayalam, Hindi, English and German. In *Forum for Information Retrieval Evaluation*, FIRE 2020, page 29–32, New York, NY, USA. Association for Computing Machinery.

Shriphani Palakodety, Ashiqur R. KhudaBukhsh, and Jaime G. Carbonell. 2019a. Kashmir: A computational analysis of the voice of peace. *CoRR*, abs/1909.12940.

Shriphani Palakodety, Ashiqur R. KhudaBukhsh, and Jaime G. Carbonell. 2019b. Voice for the voiceless: Active sampling to detect comments supporting the rohingyas. *CoRR*, abs/1910.03206.

Arjun Roy, Prashant Kapil, Kingshuk Basak, and Asif Ekbal. 2018. An ensemble approach for aggression identification in English and Hindi text. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018)*, pages 66–73, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Omar Sharif, Eftekhar Hossain, and Mohammed Moshiul Hoque. 2020. Techtexc: Classification of technical texts using convolution and bidirectional long short term memory network.

Omar Sharif, Eftekhar Hossain, and Mohammed Moshiul Hoque. 2021. Combating hostility: Covid-19 fake news and hostile post detection in social media.

Takenobu Tokunaga and Iwayama Makoto. 1994. Text categorization based on weighted inverse document frequency. In *Special Interest Groups and Information Process Society of Japan (SIG-IPSJ*. Citeseer.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *CoRR*, abs/1906.08237.