Generalization in NLI: Ways to (Not) Go Beyond Simple Heuristics

Prajjwal Bhargava [1] Aleksandr Drozd [2] Anna Rogers [3]

[1]The University of Texas at Dallas [2]Riken [3]University of Copenhagen

Introduction

Problem: DL models pick up spurious correlations in NLP datasets.

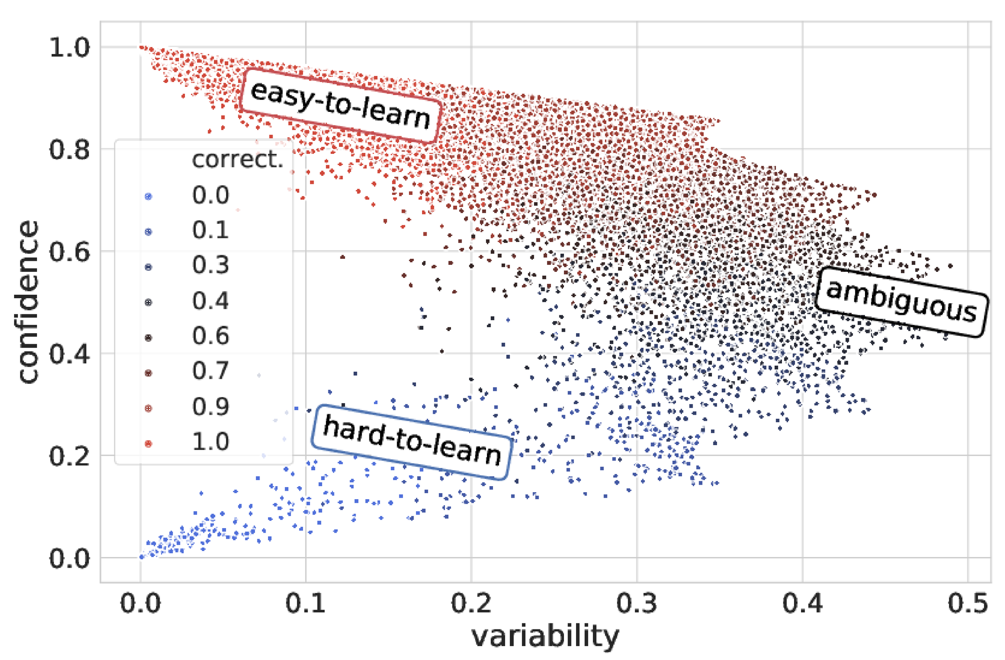
Case study: MNLI dataset "teaches" the following heuristics, tested by adversarial HANS dataset.

Heuristic	Premise	Hypothesis	Label
Lexical Overlap	The banker near the judge saw the actor.	The banker saw the actor.	E
Subsequence	The artist and the student called the judge.	The student called the judge.	E
Constituent	Before the actor slept, the senator ran.	The actor slept.	E

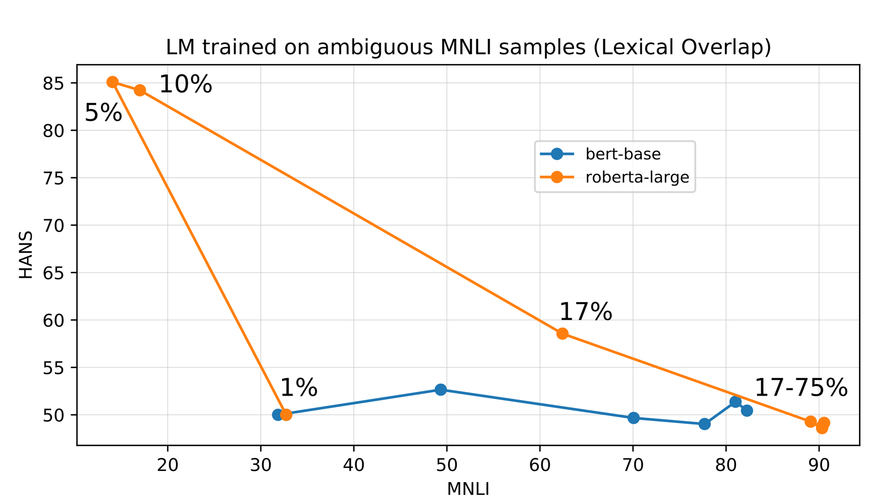
Motivation: How to learn to generalize to adversarial data when the training data has spurious patterns?

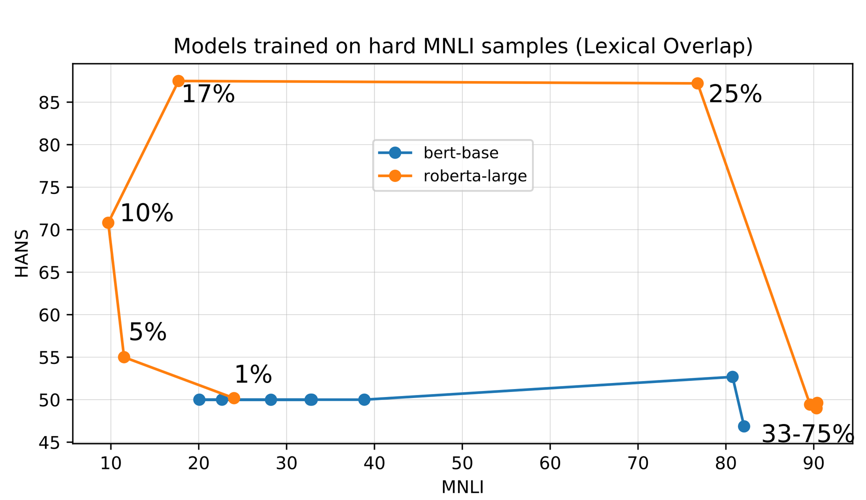
Subsampling Training Data with Cartography

Data cartography characterizes training data points via the model's confidence in the true class, and the variability of this confidence across epochs.



MNLI data map (with RoBERTa-large)



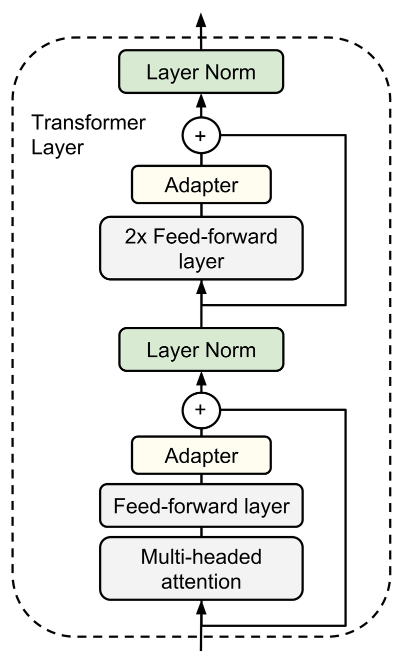


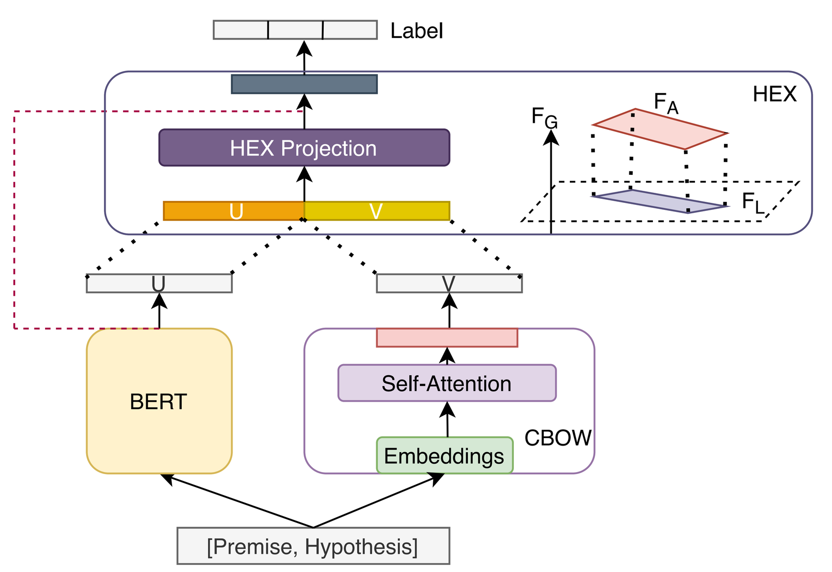
- For RoBERTa-large there does exist a MNLI subsample (at about 25% training data) yielding good performance on both HANS and MNLI.
- RoBERTa initially "learns" HANS at 5% of training data, but "loses" it before reaching even 60% accuracy on MNLI

Model based approaches

Adapter Networks:

- Motivation: Separate task-specific components could lead to increase in the amount of non-task-specific knowledge in the model.
- How: two linear layers (up and down) with a bottleneck of reduction factor of 16 and the ReLU non-linearity.



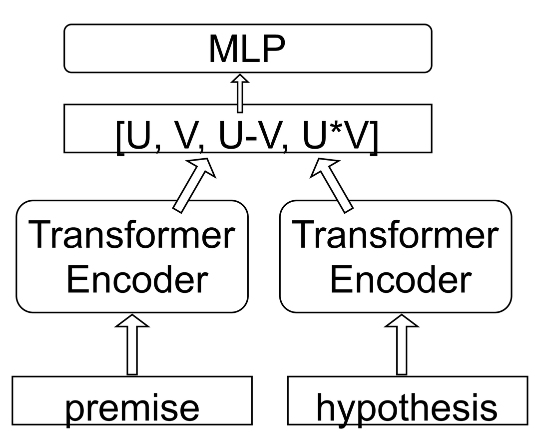


Explicit Debiasing:

- Motivation: If MNLI 'teaches' to rely on superficial features, we could try to avoid them.
- How: 'Naive' model learns superficial features. The transformer representation is projected orthogonally to that of the 'naive' model.

Siamese Transformer:

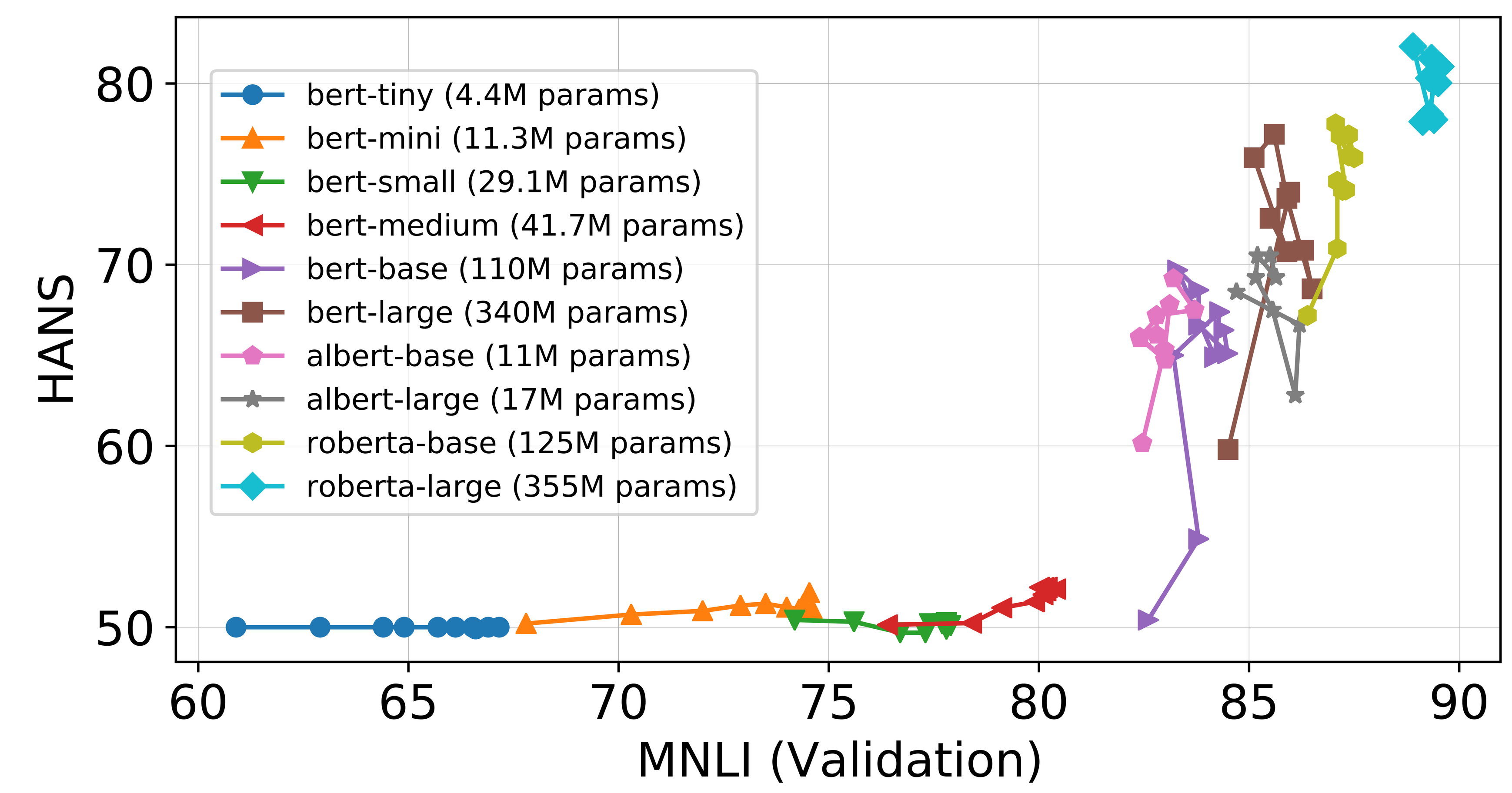
- Motivation: bottlenecking the interaction between premise and hypothesis to encourage the learning of more abstract patterns
- How: mean-pooled outputs of last transformer layer of two BERT encoders are fed as inputs to an MLP classifier



Architecture	Encoder	HF trainer MNLI	HF trainer HANS	Custom Trainer MNLI	Custom Trainer HANS
Siamese networks / frozen encoder	BERT-base	51.43	50.74	57.2 / 0.2	51.3 / 0.1
	BERT-large	51.72	51.12	61.4 / 0.1	51.6 / 0.1
Siamese networks / trainable encoder	BERT-base	58.9	52.79	76.5 / 0.03	51.3 / 0.03
	BERT-large	59.9	51.21	78.7	52.5
Adapter networks	BERT-base	82.6	50.97		
	BERT-large	84.75	57.17		
	RoBERTa-base	86.33	57.21		
	RoBERTa-large	90.4	75.93		
HEX debiasing	BERT-base	56.25	50.58		

Increasing Model Size

If pre-training "teaches" transferable linguistic knowledge, the models absorbing more data could be expected to generalize better.



Analysis: Bias Under Low Confidence

In low-confidence samples, BERT is biased towards entailment even if lexical overlap is *reduced*!

Motivation: Once the model learns that some pattern is a strong signal for a label, it will over-rely on it. But how much heuristic-matching evidence does it need?

Premise: do it now, $\overbrace{\text{think}}^{\text{ythink}}$ $\overbrace{\text{bout}}^{\text{about}}$ it later *Hypothesis:* $\overbrace{\text{think}}^{\text{zthink}}$ about it now, do it $\overbrace{\text{later}}^{\text{late (r}}$

Corruption	Labels	BERT	RoBERTa
Character insert	Entailment	+18.2	+11.9
	Neural	+13.78	+0.8
	Contradiction	−28.89	−8.4
Character substitute	Entailment	+35.5	+20.4
	Neural	+1.6	+5.9
	Contradiction	−23.9	−17.6
Character swap	Entailment	+23.8	+18.1
	Neural	−1.6	+3.3
	Contradiction	−15.5	−13.9

Code 🤗 github.com/prajjwal1/generalize_lm_nli

Code github.com/vecto-ai/langmo

Paper arxiv.org/abs/2110.01518