Building Competent Models: Ensembles for Unlearning Single Feature Artifacts

Abstract

Large, cornerstone datasets used for natural language inference (NLI) have documented dataset biases, often reflecting idiosyncrasies in the collection of the data or biases in the data sampling procedure. These dataset artifacts introduce the risk of learning spurious single feature heuristics during the training process, reflecting poorly on a model's performance out of sample. In this paper, I employ the competency problem framework introduced in (Gardner et al., 2021) to discover and describe dataset artifacts in the SNLI dataset (Bowman et al., 2015). I then fine-tune an implementation of ELECTRA-small (Clark et al., 2020b) on the SNLI data, and evaluate its reliance on learned single feature associations. Finally, I explore the use of a mixed capacity ensemble for unlearning dataset artifacts. Importantly, I discuss the relationship between mixed capacity ensembles and the competency framework, while highlighting its effectiveness at addressing single feature artifacts.

1 Introduction

Popular natural language inference (NLI) datasets are known to contain biases. Most commonly, datasets contain single words which have a high frequency of appearing alongside particular labels. For example, words like ”disgust” and ”boring” could appear attached to a poor review with an elevated frequency in a movie review dataset. While these words can be helpful for prediction in the context of the training data, learning a strong weight on a particular word can lead to poor performance out of sample. In this paper I synthesize the competency problems framework with work done on mixed capacity ensemble models to show a workflow for automatically identifying and mitigating dataset artifacts present in the SNLI dataset. In section 2 of this paper I characterize examples in the SNLI dataset (Bowman et al., 2015) which can be classified as ”competency problems”. The competency problem framework, defined further in section 2, provides an automated way of classifying single features in a dataset as sources of learned bias. In section 3, I train an ELECTRA-small model on the SNLI training data, and demonstrate evidence that the model learns harmful heuristics identified under the competency problem framework. I then propose and implement a mixed capacity ensemble (MCE) in section 4, which seeks to debias the ELECTRA-small model by allowing it to train in an ensemble with a lower capacity model. Finally, in section 5 I compare results from the two models, finding significant improvements in terms of bias and performance on examples high in dataset artifacts. I observe a small decline on in-domain accuracy, but the trade off is offset by improvements in out of domain accuracy and an overall reduction in learned bias.

2 Dataset Artifacts Under the Competency Assumption

Dataset artifacts are features or characteristics of a dataset which encourage a learner to develop misleading associations between those features and target labels in the data. Dataset artifacts often arise as a result of how data is collected. In this paper, I analyze the existence of artifacts in the SNLI (Bowman et al., 2015) dataset, a large corpus of text data, hand-annotated by human volunteers. Each example in the dataset is a premise-hypothesis pair, labeled as ”contradiction”, ”neu-

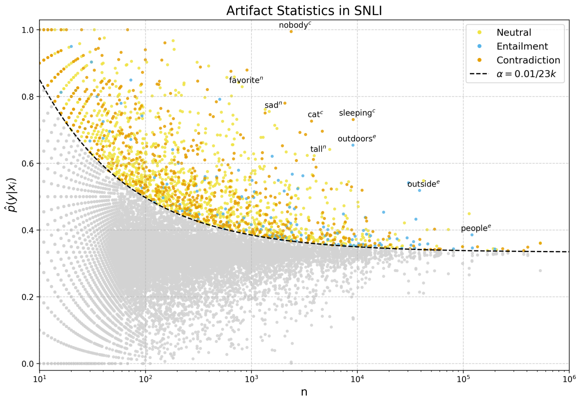

Figure 1: Dataset Artifacts in SNLI

tral", or "entailment". The examples in the SNLI dataset are generated by providing a human annotator with a premise, and asking them to return a hypothesis which contradicts the premise, entails the premise, and one hypothesis which is neutral.

A consequence of the SNLI sampling process is that human annotators are likely to fall back on a set of common heuristics when asked to generate an example hypothesis. For example, given a premise about an individual spending time with other people, a human labeler might use words like "nobody" or "no one" to generate a contradicting hypothesis. Heuristics like these can introduce dataset artifacts by generating data where single words are heavily correlated with a single label.

Fitting a model which has learned potentially strong associations between single words and labels is dangerous in practice. While these single word features are likely to be useful for performing in-domain prediction, over fitting on features related to these artifacts can lead to poor performance on out of domain tasks. Furthermore, reliance on single word features reflects a lack of true natural language understanding from the learner. No single word should contain enough information to make a confident prediction on any example. Single word features which drive the model to predict a particular label fall into a class of artifacts known as competency problems.

Introduced by (Gardner et al., 2021), a competency problem occurs when the marginal distribution of labels, given a single word, diverges from the uniform distribution. In the context of the SNLI dataset, that means for each single-word feature x_i, and each label $y \in \{0,1\}^3$, a competency problem arises when $p(y|x_i) \neq \frac{1}{3}$. I refer to this as the "competency assumption", as it reflects the assumption that a competent learner, given a single-word, over a large number of guesses should not guess one label more than another. This assumption reflects the fact that no single word should imply a particular label, as entailment is derived from deeper semantics embedded in each example. An additional benefit to defining competency problems, is that we now have a generalized method for exploring artifacts in the SNLI dataset. More specifically, the competency assumption implies the following hypothesis test:

Let $\hat{p}$ be the empirical probability of observing a label given a single word. By the competency assumption, I test the hypothesis that the true marginal distribution of the labels, given a single word, is uniform. That is $p_0 = \frac{1}{3}$. To test this hypothesis, I compute the following z-test statistic:

$$z^\star = \frac{\hat{p} - p_0}{\sqrt{p_0(1 - p_0))/n}} \qquad (1)$$

Figure 1 shows the result of testing this hypothesis on every word-label pair in the SNLI dataset. In the figure, the dotted line represents the rejection threshold with points above the line representing word-label pairs for which I reject the null hypothesis that $\hat{p} = \frac{1}{3}$. As alluded to, negation words such as "nobody" jump out as candidate artifacts, suggesting human annotators fall back on negation words for creating contradictions. Additionally, words such as sleeping and cat have heavy associations with the contradiction label. Sleeping is a word that can easily contradict premises where the subject is taking an action (other than sleeping). Cat is a word that may be used to contradict premises that are about dogs (or other pets). Entailment artifacts seem to follow the pattern of generic words. For example, words like outdoors and outside can be used to create entailment hypothesis for any premise that involves the subject engaging in actions outside. Words such as people can be used to create entailment for any premise that involves more than one person. Lastly, neutral labels seem to have a large number of artifacts which reflect subjective adjectives. Words like sad and favorite can be used to create neutral hypothesis in the case of premises that don't qualify how the subject feels in a given situation. Furthermore, words like tall can be used to create easy neutral hypothesis given premises which don't specify the height of a subject. This suggests that human an-

Train Set	Class	$\Delta\hat{p}_y$
SNLI	entailment	+15.1%
SNLI	neutral	+12.8%
SNLI	contradiction	+17.1%

Table 1: Average difference in predicted class between top 20 $z^\star$ and bottom 20 $z^\star$ tokens using ELECTRA-small fine-tuned on SNLI

Dominant Class	Accuracy	Avg. Miss	N
entailment	0.439	0.720	9,815
neutral	0.437	0.685	2,728
contradiction	0.437	0.670	3,754

Table 2: ELECTRA-small Model Performance on Test Examples that include top 20 $z^\star$ tokens

notators are likely to add unrelated or uncertain information to a premise when asked to generate a neutral hypothesis.

Now that word-label pairs failing the competency assumption have been identified, the next goal is to understand if a model, trained on the SNLI data, learns short-cuts from this set of competency problems.

3 Evaluating Model Bias

While it's clear that the SNLI dataset carries dataset artifacts introduced by heuristics used by human annotators, it remains to be shown that a learner actually learns these short-cuts. To evaluate the empirical presence of model bias learned from dataset artifacts, I fine-tune a pre-trained ELECTRA-small model (Clark et al., 2020b) using the SNLI dataset. In this exercise, I train the model for three epochs on the SNLI training set, using a learning rate of 5×10^{-5} with weight decay if 1×10^{-2}, and a batch size of sixteen. The model achieves a test accuracy of 0.895 on the SNLI test data.

To evaluate the model's reliance on dataset artifacts I replicate an experiment from (Gardner et al., 2021). Each word that is present more than twenty times across the premises and hypothesis in the SNLI data is used to create synthetic examples. Each synthetic example is created by placing a single word in either the premise or hypothesis slot, leaving the other slot empty. For each of these synthetic examples, I take a forward pass with the model and recover the logit predictions for each of the classes. These predictions are then averaged between the hypothesis only example and the premise only example. The goal of this experiment is to recover an estimate of the marginal probability of a label given each word. Table 1 presents the results of this experiment.

For each label, I take the top twenty highest scoring words and the bottom twenty lowest scoring words from our test of the competency assumption, and compare mean differences in $\hat{p}(y|x_i)$. More specifically, I construct column three of table 1 by computing $\Delta\hat{p}_y = \sum_{z_i^\star \geq z_{20}^\star} \hat{p}(y|x_i) - \sum_{z_i^\star \leq z_{N-20}^\star} \hat{p}(y|x_i)$. By comparing high-scoring tokens to a set of stable low-scoring tokens, I can recover an approximation of how much, on average, the model biases its predictions when given examples containing competency problems. Table 1 shows biases larger than 10% across all classes, with the strongest average bias being 17.1% for the contradiction class. The large positive bias estimates from this experiment provide evidence that ELECTRA-small is learning single-feature heuristics that have strong influence over its predictions. Returning briefly to the set of word-label artifact examples discussed in section 2, our experiment reveals strong evidence that these particular single-word features are being used as short-cuts by the model. Following this experiment, I recover a $\hat{p}(y|x_i)$ of 0.984 for cat, and 0.999 for nobody, two of our contradiction artifact examples. The word ”sleeping” is an exception, with a $\hat{p}(y = contradiction|x_i) = 0.303$. Our entailment artifacts also provide strong evidence of being learned by the model with $\hat{p}(y|x_i)$ of 0.888 for outdoors, 0.868 for outside, and 0.943 for people. Lastly, our neutral artifact examples continue to confirm intuition, having a $\hat{p}(y|x_i)$ of 0.859 for favorite, 0.961 for sad, and 0.698 for tall.

While this subset of examples have been selected to highlight extreme cases of competency artifacts, table 2 provides more general evidence of the consequences of learned artifacts.

Table 2 captures the accuracy and average miss, defined as $1 - \hat{p}_y^{golden}$, on examples from the SNLI test set which include words, for each label, scoring in the top 20 $z^\star$ from the earlier hypothesis test. The average accuracy on this subset of examples is approximately forty-six percentage points lower than the accuracy the model achieves on the full test set. In the case of the artifacts included in our experiment, predictions on the dominant class are only about ten percentage points better than a random guess, showing the pitfall of relying on these heuristics. Furthermore, the model has an average miss of 0.692 when test examples contain one of these artifacts. Not only do these artifacts have a strong sway over the model's predictions, these single-word features encourage the model to return confident and incorrect predictions on examples containing problematic words.

4 Building a Competent Model

Having shown that a model trained on the SNLI dataset will learn competency artifacts, I now look for a solution to debiasing the model. In (Gardner et al., 2021) the authors suggest using local edits to introduce examples in training which further balance the distribution of words over the labels in the data. However, local edits require human involvement which runs the risk of introducing new sources of bias and being potentially costly in terms of time and effort. Other authors have explored the possibility of learning a biased model, fit on known dataset artifacts, and then using the residual as a feature in the larger model to unlearn the dataset artifacts (He et al., 2019), or simply incorporating the biased model into an ensemble such that the larger model learns from it with each update (Clark et al., 2019). These methods are shown to be successful at mitigating model bias, however they require carefully tuning the model to unlearn known examples of dataset artifacts.

Instead, I propose learning a mixed capacity ensemble using model architecture introduced in (Clark et al., 2020a). To mitigate dataset artifacts, I train an ensemble of two models in parallel, a higher-capacity model and a lower-capacity model. The lower-capacity model is designed to attend to single-feature correlations present in the data. By explicitly allowing the lower-capacity model to encode bias, the ensemble indirectly penalizes reliance on these features for predictions. Over time, the higher-capacity model develops predictions that are independent of the features captured by the lower-capacity model. Each epoch I recover predictions from the low-capacity and higher-capacity models, and combine them as follows to recover predictions for the ensemble:

$$\hat{y}_i^e = softmax(log(f_h(x_i)) + log(f_l(x_i)) + log(p_y))$$
$$\hat{y}_i^l = softmax(log(f_l(x_i) + log(p_y))$$
$$\hat{y}_i^h = softmax(log(f_h(x_i) + log(p_y)))$$

I then compute the training loss below, with $w = 0.5$ to update the ensemble's weights. The lower-capacity model contributes to identifying high-probability artifacts. These are effectively "discounted" when the ensemble combines its output with the higher-capacity model's predictions. The inclusion of $log(p_y)$, a uniform prior over the classes, further discourages over-reliance on individual features.

$$L(\hat{y}_i^e, \hat{y}_i^l, y_i, w) = \sum_i^n L(\hat{y}_i^e, y_i) + wL(\hat{y}_i^l, y_i) \quad (2)$$

The core principle of this approach lies in the conditional independence of the two models' predictions given the true label y. For a given input x let $f_l(x) = x_l$ and $f_h(x) = x_h$ represent the lower-capacity and higher-capacity model outputs, respectively. As shown in (Clark et al., 2020a), recovering the ensemble prediction relies on x_h and x_l being conditionally independent, given y. Succinctly, the conditional independence assumption holds if:

$$P(y|x_h, x_l) \propto P(y|x_h)P(y|x_l)/P(y) \quad (3)$$

This factorization ensures that the ensemble prediction leverages complementary information from both models, while penalizing redundant reliance on features captured by x_l. Conditional independence ensures that the lower-capacity model captures the dataset artifacts, allowing the higher-capacity model to focus on deeper semantic patterns. In practice, conditional independence is encouraged through the structure of the loss function and the separation of responsibilities between the lower- and higher-capacity models. By ensuring the lower-capacity model captures artifacts, the higher-capacity model avoids learning redundant correlations.

For the empirical exercise, the higher-capacity model will be the ELECTRA-small model fine-tuned on the SNLI training data from section 3,

Test Set	ELECTRA-small	MCE
SNLI	0.889	0.875
HANS	0.493	0.500

Table 3: Model Accuracy on NLI Datasets

while the lower-capacity model will be a unigram bag-of-words multi-layer perceptron. For the lower-capacity model, I create a vocabulary of the 10,000 most common words in the SNLI dataset. Each example from the SNLI data is then tokenized into a feature vector by first combining the premise and hypothesis into a single string of text, and then mapping the count of each word in the example to its position in the vector. The final lower-capacity model is a fully-connected 2-layer neural network with a hidden dimension of 300 and ReLU activation functions.

The benefits to training a mixed-capacity ensemble (MCE) are two-fold. Firstly, the MCE is designed to automatically identify and learn dataset bias introduced by learning simplistic patterns in the training process. This addresses the concern of introducing bias through human edits and saves time on what would be spent identifying particular sources of bias. There is an added benefit of mitigating biases that may have been unknown to us prior to the modeling step. The second benefit to training the MCE is that it endogenously captures the competency assumption. By separating the treatment of single-feature artifacts and deeper semantic patterns, the MCE directly addresses the competency assumption, ensuring that predictions are not dominated by features where $P(y|x_i) \neq \frac{1}{3}$.

5 Results

The MCE described in section 4 is trained for three epochs with a learning rate of 5×10^{-5}, and weight decay of 1×10^{-2}. The lower-capacity model's loss is down-weighted, setting $w = 0.5$ as written in equation (2). Training the MCE requires specifying a prior over the classes $log(p_y)$. In line with the competency assumption, I allow the prior probability to be the log of the label probability drawn from the uniform distribution.

Table 3 compares the accuracy of the ELECTRA-small model to the MCE on the SNLI test data and the HANS test data (McCoy et al., 2019). The HANS dataset is a challenge dataset designed to challenge models which rely on commonly learned heuristics present in popular SNLI datasets. Both ELECTRA-small and MCE perform poorly on the HANS dataset, however the MCE improves on the performance of the ELECTRA-small. The poor performance on HANS reflects the dataset's emphasis on testing models' reliance on specific heuristics (e.g., lexical overlap). The slight improvement in MCE suggests its ability to reduce reliance on some heuristics, although additional tuning or constraints may be required to achieve substantial gains on this dataset. The ELECTRA-small model performs better on the SNLI test set by about one percentage point of accuracy. While the MCE performs slightly worse on SNLI compared to ELECTRA-small, this trade-off reflects the model's reduced reliance on dataset artifacts. In practice, this suggests greater robustness to out-of-distribution data, as indicated by the performance gains on HANS and bias-heavy subsets of SNLI.

Train Set	Class	$\Delta\hat{p}_y$
SNLI	entailment	+13.2%
SNLI	neutral	+9.5%
SNLI	contradiction	+15.6%

Table 4: Average difference in predicted class between top 20 $z^\star$ and bottom 20 $z^\star$ tokens using Mixed Capacity Ensemble

Table 4 revisits the model bias experiment from section 3, recalculating the 20 highest scoring words and the 20 lowest scoring words under the MCE's predicted probabilities. The MCE shows clear improvement on the influence of single word features. Under the MCE our predicted bias for the entailment class drops by two percentage points, our predicted bias for the neutral class drops by over three percentage points, and our predicted bias for the contradiction class drops by nearly two percentage points. The results in table 4 show that predictions on our 20 most problematic word-label pairs, by label, are becoming more comparable to our least problematic word-label pairs using the MCE.

Table 5 parallels the results from table 2, showing the performance of the MCE on the 20 highest scoring words from the ELECTRA-small model. These words have been fixed to match the test-

Dominant Class	Accuracy	Avg. Miss	N
entailment	0.875	0.169	9,815
neutral	0.875	0.167	2,728
contradiction	0.879	0.160	3,754

Table 5: MCE Model Performance on Test Examples that include top 20 $z^\star$ tokens

ing sets from Table 2 to allow us to compare results between the ELECTRA-small model and the MCE. The results in table 5 demonstrate major improvements in terms of accuracy and average miss from the MCE. Average accuracy nearly doubles on these ”hard” examples from the SNLI testing set, while the average miss made by the MCE is 0.165 compared to the average miss of 0.692 made by the ELECTRA-small on these same examples. These results are clear evidence that the MCE is improving on examples which are high in competency artifacts, which the ELECTRA-small fails on. This evidence is encouraging, and suggest that the MCE is likely a more robust model, out of sample, than the fine-tuned ELECTRA.

6 Conclusion

In this paper, I demonstrate that mixed capacity ensemble (MCE) models effectively mitigate the influence of dataset artifacts on model predictions. The MCE outperforms a fine-tuned ELECTRA-small model on in-domain examples with high artifact influence, showing reduced reliance on single-word features and addressing competency problems more robustly. These results indicate that MCEs represent a step forward in building more competent models by making competency an endogenous characteristic of their architecture. This paper also highlights the connection between the competency problems framework and the MCE architecture, offering an automated pipeline for artifact discovery and mitigation. While the implementation presented here does not explicitly model the conditional independence constraint modeled in (Clark et al., 2020a), the findings suggest that doing so would yield even greater improvements. As such, the results in this paper likely represent a lower bound on the potential of MCEs. By reducing bias and improving performance on artifact-heavy examples, this work lays the groundwork for developing more robust, fair, and generalizable natural language inference systems. Future research could extend this framework to broader applications, advancing the field’s ability to tackle dataset artifacts in a principled and automated way.

References

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In Lluís Màrquez, Chris Callison-Burch, and Jian Su, editors, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal, September. Association for Computational Linguistics.

Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. 2019. Don’t take the easy way out: Ensemble based methods for avoiding known dataset biases. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *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 4069–4082, Hong Kong, China, November. Association for Computational Linguistics.

Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. 2020a. Learning to model and ignore dataset bias with mixed capacity ensembles. In Trevor Cohn, Yulan He, and Yang Liu, editors, *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3031–3045, Online, November. Association for Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020b. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *ICLR*.

Matt Gardner, William Merrill, Jesse Dodge, Matthew Peters, Alexis Ross, Sameer Singh, and Noah A. Smith. 2021. Competency problems: On finding and removing artifacts in language data. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1801–1813, Online and Punta Cana, Dominican Republic, November. Association for Computational Linguistics.

He He, Sheng Zha, and Haohan Wang. 2019. Unlearn dataset bias in natural language inference by fitting the residual. In Colin Cherry, Greg Durrett, George Foster, Reza Haffari, Shahram Khadivi, Nanyun Peng, Xiang Ren, and Swabha Swayamdipta, editors, *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*, pages 132–142, Hong Kong, China, November. Association for Computational Linguistics.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy, July. Association for Computational Linguistics.