Behind the Scenes Of Sarcasm

Yair Gat and Ido Terner
Industrial Engineering and Management Department
The Technion - Israel Institute of Technology
Haifa 32000, Israel
{yairgat, ido.terner}@campus.technion.ac.il

Abstract

Sarcasm is a means of communication that involves a hidden insult. Often, understanding sarcasm requires a high level of comprehension. We present a method for approximating the impact of specific part-of-speech on the decision-making process on sarcasm detection models. Our method is based on INLP[1], a method for removing information from neural representations. We create a map of words and their part-of-speech by tagging the part-of-speech of words in a sentence using a pre-trained model[2]. We repeat training linear classifiers that predict the part of speech from GLOVE embedding and project the embedding on the classifier's null space. In doing so, we created a representation that omitted the meaning of the parts of speech. By measuring the changes in the sarcasm detection model we draw conclusions.

See code in github[1]

1 Introduction

The success of neural language models is limited by their difficulty in explaining. A question which has been commonly asked recently: What is stand behind the decision of neural networks? There is a vast amount of literature on feature removal. Unfortunately, many of these techniques are not feasible. We tackle this question using INLP[1] method that proposes a novel and easy-to-implement method of removing features from neural representations. Our method is designed to study the impact of specific parts of speech on a sarcasm detection model. Our method involves four steps: First, we create a parts-of-speech dataset that maps words to their part-of-speech. We do so, using a pre-trained model SequenceTagger model from Flair library. Second, we remove linear connections of specific parts-of-speech from the GLOVE embedding by INLP[1], by doing so we create a new representation that does not take into consideration the meaning of part-of-speech in the embedding of a given word. Third, we train a sarcasm detection model on our sarcasm data-set with no changes on Glove. Finally, we characterize the change in the model's prediction that results from replacing the embedding. If the resulting change in the sarcasm detection, we infer that the model uses the feature under consideration of part-of-speech meaning in the sentence. The full process is visualize in Figure 1.

We demonstrate the utility of our method on two specific part-of-speech, NN(singular noun) and JJ(adjectives).

2 Related Work

Iterative Null Space Projection (INLP) [1] removes a "protected attribute" from a representation vector, by repeatedly training linear classifiers that aim to predict that attribute from the representations we want to remove and projecting the representations on their null-space. The **Counterfactual Interventions Reveal the Causal Effect of Relative Clause Representations on Agreement Prediction** [3] proposes an intervention-based method, AlterRep, to test whether language models use the linguistic information encoded in their representations in a manner that is consistent with the grammar in terms of relative clauses. Their method is an application of the INLP method. To do so, they selectively remove information from the representation and observe the change in the behavior of the model on the main task.
Amnesic Probing: Behavioral Explanation with Amnesic Counterfactual [4] selectively remove information from the representation and observe the change in the behavior of the model on the

[1] https://github.com/iTerner/Sarcasm-Detection

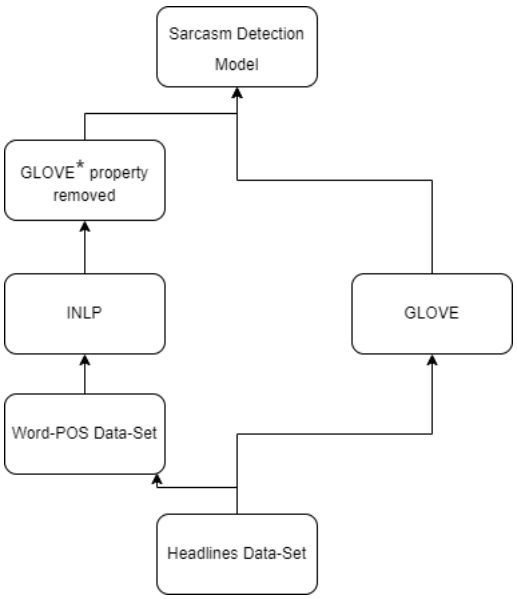

Figure 1: A schematic description of the proposed method. First, we train the sarcasm model on the glove representation. Then, we apply the INLP algorithm to get glove*, which is the embedding after applying INLP with a specific part-of-speech. And finally, evaluate the sarcasm model on both the glove and glove* test.

main task. Their methods are based on INLP, but focusing on removal under control. They suggest two types of control: Control over Information and Control over Selectivity. Among other things, they also demonstrate their method on the question "is part-of-speech information important for word prediction?".

3 Dataset

Past studies in Sarcasm Detection mostly make use of Twitter datasets collected using hashtag-based supervision but such datasets are noisy in terms of labels and language. Furthermore, many tweets are replies to other tweets, and detecting sarcasm in these requires the availability of contextual tweets.

To overcome the limitations related to noise in Twitter datasets, this News Headlines dataset for Sarcasm Detection[5] is collected from two news websites. TheOnion aims at producing sarcastic versions of current events and we collected all the headlines from News in Brief and News in Photos categories (which are sarcastic). We collect real (and non-sarcastic) news headlines from HuffPost.

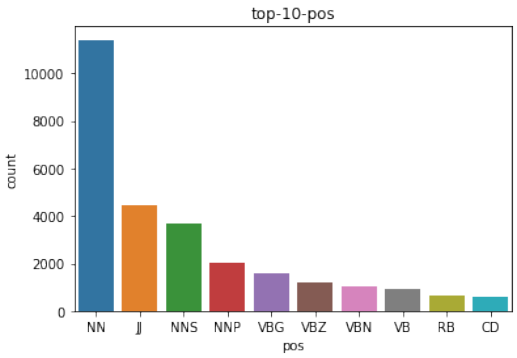

Figure 2: The top 10 part-of-speech frequencies in the sarcasm news headlines dataset.

4 Architecture

In this project, we wanted to leverage the sequence dependency that appears in sentences, so, we used a bi-directional LSTM encoder followed by an MLP decoder, which gave us the sarcasm prediction. The full architecture is shown in Figure 3.

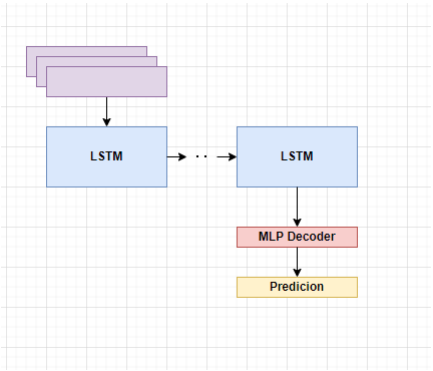

Figure 3: The architecture used in the project. The sentences (purple) are given as input to the LSTM encoder (blue), which followed by the MLP decoder (red) that gives us the prediction of whether or not the sentence is sarcastic (yellow).

5 INLP

INLP Nullspace Projection (INLP)[1]. Given a labeled dataset of representations H, and a property to remove, Z, INLP neutralizes the ability to linearly predict Z from H. It does so by training a sequence of linear classifiers (probes) $c_1, ..., c_k$ that predict Z, interpreting each one as conveying information on a unique direction in the latent space that corresponds to Z, and iteratively removing each of these directions. Concretely, we assume that the ith probe c_i is parameterized by a matrix W_i. In the ith iteration, c_i is trained to predict Z from H3, and the data is projected onto its nullspace using a pro-

jection matrix $P_{N(W_i)}$. This operation guarantees $W_i P_{N(W_i)} H = 0$, i.e., it neutralizes the features in the latent space which were found by Wi to be indicative to Z. By repeating this process until no classifier achieves above-majority accuracy, INLP removes all such features. In our case, the property we aim to remove is the part-of-speech of a word (in particular single noun and adjective).

6 Experiments and Results

In this section we will first describe the preprocessing stage of the data, and then decribe the training process and show the results on the sarcastic headline dataset.

6.1 Preprocessing

We process the data in the following way:

1. Convert each sentence to a list of the vector representation of each word from GLOVE, if a given sentence had the size of zero, we removed the sentence).

2. Apply INLP given specific properties. We use Logistic Regression as our linear classifer. By doing so, we create the GLOVE* vector representation for NN (single noun) and JJ (adjectives) (the two most common part-of-speech with appearance percentage of 39% and 12% respectively, see more in Figure 2) using the INLP method. we train the classifier for ten iterations until we got the expected accuracy (see Figure 4).

3. Split the data to train and test set where the train set size was 80% of the data and the test set was 20%. (there is no need for a validation set since we are not willing to optimize the test accuracy of the model).

4. Pad all the sentences with zeros according to the max sentence length in the set.

6.2 Results

We trained our sarcasm detection model for 10 epochs with Adam optimizer. We used a batch size of 64 and a learning rate of 10^{-3}. The full results can be shown in Table 1. The train loss and accuracy per epoch can be seen in Figure 5.

7 Discussion

In this project, we showed that sarcasm is expressed through singular nouns and adjectives, using the

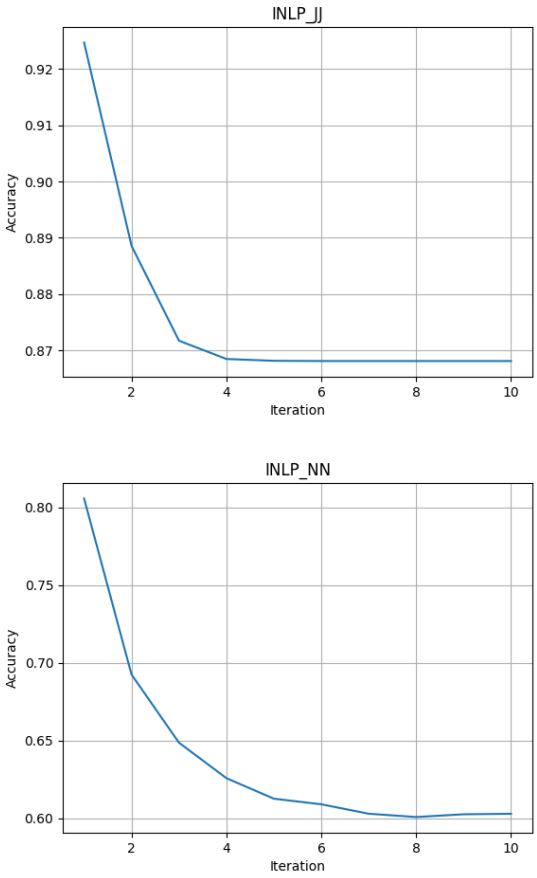

Figure 4: Accuracy of the linear classifier per iteration. As we can observe, the accuracy drops down to random guess chance.

Vector Representation	POS	Accuracy
GLOVE	None	87.086%
GLOVE* (NN)	NN	84.396%
GLOVE* (JJ)	JJ	80.413%

Table 1: The Accuracy results on the different kind of glove representation. The Raw sentences (GLOVE), the sentences after using INLP on NN (GLOVE* (NN)), and the sentences after using INLP on JJ (GLOVE* (JJ)).

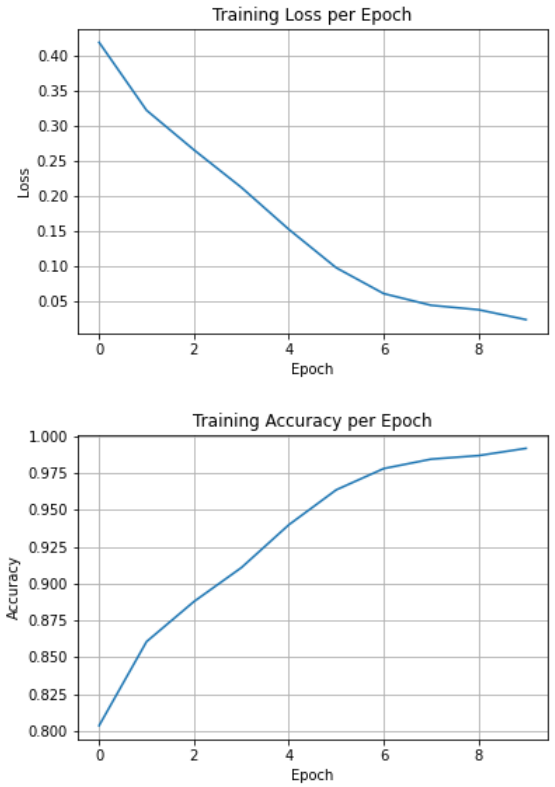

Figure 5: Train loss and accuracy of the sarcasm detection per epoch.

INLP method. We proved that by training a basic sarcasm model using GLOVE representation and evaluating the model on several test sets, such as GLOVE, GLOVE*(NN), and GLOVE*(JJ). We saw that after altering the vector representation, the accuracy of the model decreased, which indicate that sarcasm is expressed in both singular noun and adjectives. To train and evaluate, we used the sarcasm news headline dataset. We strongly believe that utilizing the INLP and similar methods could open a new fascinating branch in Natural Language Processing which is currently only in the beginning.

References

[1] Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. Null it out: Guarding protected attributes by iterative nullspace projection, 2020.

[2] Alan Akbik, Duncan Blythe, and Roland Vollgraf. Contextual string embeddings for sequence labeling. In *COLING 2018, 27th International Conference on Computational Linguistics*, pages 1638–1649, 2018.

[3] Shauli Ravfogel, Grusha Prasad, Tal Linzen, and Yoav Goldberg. Counterfactual interventions reveal the causal effect of relative clause representations on agreement prediction. *CoRR*, abs/2105.06965, 2021.

[4] Yanai Elazar, Shauli Ravfogel, Alon Jacovi, and Yoav Goldberg. When bert forgets how to POS: amnesic probing of linguistic properties and MLM predictions. *CoRR*, abs/2006.00995, 2020.

[5] Rishabh Misra and Jigyasa Grover. *Sculpting Data for ML: The first act of Machine Learning*. 01 2021.