Proposal: Adversarial Analysis of Natural Language Inference Systems

Tiffany Chien
University of California, Berkeley

Jugal Kalita
University of Colorado, Colorado Springs

Abstract

The release of large natural language inference (NLI) datasets like SNLI and MNLI led to rapid development and improvement of completely neural systems for the task. Most recently, pre-trained, Transformer-based models like OpenAI-GPT and BERT have reached near-human performance on these datasets. However, various work has questioned the degree to which performance on these datasets really indicates understanding of the task (and language in general). Adversarial (challenge) datasets have been created that cause models that perform well on these datasets to fail dramatically. Although extra training on this data generally improves model performance on just that type of data, transferring that learning to unseen examples is still partial at best. This work compares the failures of state-of-the-art and older models, to determine how differences in architecture (non-recurrent) and training (pre-training) affect what the models are better and worse at. Then, we use a visualization tool on the new Transformer-based models to deeply examine what information they pay attention to, and how that leads to failure.

Introduction

In recent years, deep learning models have achieved and continued to improve on state-of-the-art results on many NLP tasks. However, models that perform extremely well on standard datasets have been shown to be rather brittle and easily tricked. In particular, the idea of *adversarial* examples or attacks was brought over from computer vision, and various methods of slightly perturbing inputs have been developed that cause models to fail catastrophically (McCoy, Pavlick, and Linzen 2019; Glockner, Shwartz, and Goldberg 2018; Naik et al. 2018).

Adversarial attacks need to be studied from a security perspective for the deployment of real-world systems, but they are also a powerful lens into *interpretability* of black-box deep learning systems. By examining the failures of state-of-the-art models, we can learn a lot about what they are really learning, which may give us insights into improving their robustness and general performance.

One philosophical generalization about the cause of failure for all current NLP systems is a lack of deep, 'real' understanding of language. We will focus on the task of natural language inference (NLI, also known as textual entailment (RTE)), which is a basic natural language understanding task that is thought to be a key stepping stone to higher-level understanding tasks like question answering and summarization. The setup of the NLI task is to determine whether a *hypothesis* is true given a *premise*, answering *entailment*, *contradiction*, or *neutral* (the hypothesis's truth value can't be determined).

The current top-performing systems for NLI rely on pre-training on generic tasks, followed by fine-tuning on a labeled task-specific dataset. This is in contrast to older (before late 2018) models, which were primarily task-specific architectures trained primarily on task-specific labeled datasets. In addition, the Transformer architecture (Vaswani et al. 2017) now outperforms the previously dominating recurrent architectures (LSTM and variants). We want to investigate why exactly new models perform better, i.e. what particular kinds of reasoning they capture better.

Our goal is to pit the new state-of-the-art models against various adversarial attacks, and carefully examine when and why they fail. We will focus on semantic phenomena, like negation (Kang et al. 2018), compositionality (Nie, Wang, and Bansal 2018), and commonsense reasoning (Glockner, Shwartz, and Goldberg 2018). Many failure analyses have been performed before on older model architectures, but our contribution will be to do the same on the newest state-of-the-art models, and then compare the results with older models. Furthermore, we will go past observing when models fail and how to make them fail, and approach the question of why using visualization tools that display the models' internals.

Related Work

One important question to ask when studying adversarial vulnerability is whether the failure is caused by the model design itself, or the limits of its training data. One way to begin disentangling this blame, described by Liu, Schwartz, and Smith (2019) under a metaphor of inoculation, is to expose a small part of the challenge dataset to the model during training, and re-test its evaluation performance on the original test set and the challenge dataset.

1. If the model still fails the challenge dataset, the weakness probably lies in its design/architecture or training process.
2. If the model can now succeed at the challenge dataset

(without sacrificing performance on the original dataset), then the original dataset is at fault.

3. If the model does better on the challenge dataset but worse on the original dataset, the challenge dataset is somehow not representative of the phenomenon it was trying to test, for example having annotation artifacts or being very skewed to a particular label.

Unfortunately, even if adversarial training does improve model performance, it is fundamentally impossible to devise and train on all possible linguistic phenomena. The transferability of adversarial robustness to new kinds of examples has been tested by some of the above works, by withholding some example generation methods while training on others. Nie, Wang, and Bansal find that knowledge of each of their rule-based templates was almost completely non-transferable to others. In fact, training on some specific templates caused overfitting and hurt overall robustness. McCoy, Pavlick, and Linzen find more mixed results, with some cases of successful transfer. However, many of their syntactic heuristics are pretty similar and overlapping, so generalizability is not as difficult.

Many standard datasets for different tasks have been shown to have blatant annotation artifacts, allowing models to learn features that are strong in the training (and testing) data, but that have nothing to do with actually performing the task. Gururangan et al. (2018) find many of these artifacts in standard NLI datasets (SNLI and MNLI). For example, *neutral* hypotheses tend to be longer in length, because an easy way to generate a hypothesis that isn't necessarily entailed by the premise is to add extra details. Meanwhile, strong negation words like *nobody, no, never* are strong indicators of *contradiction*. With these artifacts in mind, they split the data into "hard" and "easy" versions, and model performance decreased by about 15% on the hard test set. These findings suggest that it is not the models' faults for failing on adversarial examples, given that there exist easier ways to get high accuracy than truly understanding anything. But it also means that current evaluation metrics greatly overestimate models' abilities and understanding.

Experimental Setup

We will test a variety of models against a variety of adversarial datasets, and then compare and visualize the results.

Models

The three newest models that we will study all gain most of their power from pre-training on a generic language task with a huge unlabeled dataset.

1. **OpenAI-GPT** (Radford et al. 2018) pre-trains on the standard left-to-right language modelling task.
2. **BERT** (Devlin et al. 2018) pre-trains on a bidirectional word-masking language modelling task, in addition to sentence pair prediction (whether the second sentence is likely to directly follow the first).
3. **MT-DNN** (Liu et al. 2019) combines BERT and multi-task learning during its pre-training, and is the reigning champion on the nine-task GLUE (General Language Understanding Evaluation) benchmark (Wang et al. 2018).

All of these models are based on the Transformer architecture (Vaswani et al. 2017), a non-recurrent, purely attention-based architecture. One important difference between the Transformer and recurrent architectures is how *word order* information is encoded. Parsing and understanding word order is obviously crucial to language understanding. Recurrent architectures explicitly pass the hidden states of previous words as an input to future hidden states. The Transformer instead uses 'position encodings', fixed (not learned) sinusoids with frequencies determined by a word's position in the input sentence. These position encodings are then added to the input embeddings. We want to explore how this comparatively simple method compares to explicit recurrence.

We will compare with three recurrent models (the second two build on the first).

1. **ESIM** Enhanced Sequential Inference Model (Chen et al. 2016) uses a bidirectional LSTM to encode sentences, and uses attention across those representations.
2. **S-TLSTM** Syntactic TreeLSTM (Chen et al. 2016) is identical to ESIM except it uses a TreeLSTM that takes a dependency parse as input.
3. **KIM** Knowledge-based Inference Model (Chen et al. 2018) enhances ESIM by incorporating knowledge from WordNet in a variety of ways (involving additions to the model architecture).

S-TLSTM and KIM use explicit extra information (parses and knowledge base data, respectively), which is the complete opposite of the unsupervised pre-training by the newest models. Goldberg found that BERT performed remarkably well on syntactic tests, so it will be interesting to see how they compare.

Adversarial Datasets

The creation of adversarial examples in NLP comes from hypotheses and observations about the simplistic assumptions that models make in place of real understanding. We list five datasets below for context, but we will just be using the last three (because of likely time constraints).

- McCoy, Pavlick, and Linzen (2019) show that models use specific fallible, surface-level syntactic heuristics, by testing them on a challenge dataset (HANS) that violates those heuristics. Their heuristics can be categorised into lexical overlap, subsequence, and constituent. They find that while all the models they tested did worse on their dataset, BERT performed best.
- Naik et al. (2018) manually examined and categorized 100 errors that a model made, and automatically generated examples in each category, including antonyms, numerical reasoning, word overlap (append "and true is true" to hypothesis) negation words (append "false is not true"), length mismatch (append "and true is true" 5 times), and spelling errors.

- **Kang et al.** (2018) created a handful of rule templates based on knowledge bases (WordNet, PPDB (paraphrase database)), in addition to a simple hand-defined negation rule. They then trained their model in a GAN framework, fighting off against the rule-based generator.
- **Nie, Wang, and Bansal** (2018) conduct a series of experiments that demonstrate that models do not correctly use compositionality information. For example, they shuffle the words in the training dataset (changing their compositional meaning), but models trained on the shuffled data still performed equally well when evaluated on the original dataset. They generate a 'compositionality-sensitivity' test by selecting examples where bag-of-words models perform poorly.
- **Glockner, Shwartz, and Goldberg** (2018) creates a dataset testing lexical inference by replacing single words in SNLI examples' hypotheses (keeping the premises the same). The new words come from an online English learning resource, and they all already exist in SNLI.

Other than McCoy, Pavlick, and Linzen, these were before GPT and BERT, so they analyze older models only.

Visualization Tool

After comparing the different models' performance on the different challenge examples, we will look deeper into the Transformer-based models using a visualization tool developed by Vig (2019). The tool specifically focuses on visualizing attention, displaying what different layers and 'attention heads' are paying attention to for specific inputs.

Planned Timeline

By week:

3. Get computer/GPU server set up.
   Replicate MNLI results for 3 pre-trained models.
4. Replicate MNLI results for 3 recurrent models.
   Evaluate all models on negation and knowledge base dataset (Kang et al. 2018).
5. Evaluate all models on compositionality (Nie, Wang, and Bansal 2018) and lexical inference (Glockner, Shwartz, and Goldberg 2018) datasets.
6. Analyze results (compare different models' performance on different datasets).
   Prepare mid term results, presentation, and writeup.
7. For datasets that the pretrained models fail on, try fine-tuning them on a portion of the challenge data, and re-evaluate.

8–9. Use visualization tool to examine failures (details depend on results of above experiments).

10. Prepare final results, presentation, and writeup.

Conclusion

In this work, we aim to chip away at the problem of model interpretability in modern neural models, by examining successful models' failures on adversarial examples. We utilize the ideas of others' analyses but focus on the most recent and hyped state-of-the-art models. We complement larger-scale targeted testing with manually examining visualizations of common failures. Interpretability leads to more useful (and safe) models for practical use, as well as helping to generate ideas for future higher-performing models, and we hope that the relentless pursuit of higher accuracy on existing datasets will continue to be balanced with critical analysis of what models really understand.

References

Chen, Q.; Zhu, X.; Ling, Z.; Wei, S.; Jiang, H.; and Inkpen, D. 2016. Enhanced LSTM for Natural Language Inference. *arXiv:1609.06038 [cs]*. arXiv: 1609.06038.

Chen, Q.; Zhu, X.; Ling, Z.-H.; Inkpen, D.; and Wei, S. 2018. Neural Natural Language Inference Models Enhanced with External Knowledge. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2406–2417. Melbourne, Australia: Association for Computational Linguistics.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv:1810.04805 [cs]*. arXiv: 1810.04805.

Glockner, M.; Shwartz, V.; and Goldberg, Y. 2018. Breaking NLI Systems with Sentences that Require Simple Lexical Inferences. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 650–655. Melbourne, Australia: Association for Computational Linguistics.

Goldberg, Y. 2019. Assessing BERT's Syntactic Abilities. *arXiv:1901.05287 [cs]*. arXiv: 1901.05287.

Gururangan, S.; Swayamdipta, S.; Levy, O.; Schwartz, R.; Bowman, S. R.; and Smith, N. A. 2018. Annotation Artifacts in Natural Language Inference Data. *arXiv:1803.02324 [cs]*. arXiv: 1803.02324.

Kang, D.; Khot, T.; Sabharwal, A.; and Hovy, E. 2018. AdvEntuRe: Adversarial Training for Textual Entailment with Knowledge-Guided Examples. *arXiv:1805.04680 [cs]*. arXiv: 1805.04680.

Liu, X.; He, P.; Chen, W.; and Gao, J. 2019. Multi-Task Deep Neural Networks for Natural Language Understanding. *arXiv:1901.11504 [cs]*. arXiv: 1901.11504.

Liu, N. F.; Schwartz, R.; and Smith, N. A. 2019. Inoculation by Fine-Tuning: A Method for Analyzing Challenge Datasets. *arXiv:1904.02668 [cs]*. arXiv: 1904.02668.

McCoy, R. T.; Pavlick, E.; and Linzen, T. 2019. Right for the Wrong Reasons: Diagnosing Syntactic Heuristics in Natural Language Inference. *arXiv:1902.01007 [cs]*. arXiv: 1902.01007.

Naik, A.; Ravichander, A.; Sadeh, N.; Rose, C.; and Neubig, G. 2018. Stress Test Evaluation for Natural Language Inference. *arXiv:1806.00692 [cs]*. arXiv: 1806.00692.

Nie, Y.; Wang, Y.; and Bansal, M. 2018. Analyzing Compositionality-Sensitivity of NLI Models. *arXiv:1811.07033 [cs]*. arXiv: 1811.07033.

Radford, A.; Narasimhan, K.; Salimans, T.; and Sutskever, I. 2018. Improving Language Understanding by Generative Pre-Training.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention Is All You Need. *arXiv:1706.03762 [cs]*. arXiv: 1706.03762.

Vig, J. 2019. Visualizing Attention in Transformer-Based Language Representation Models. *arXiv:1904.02679 [cs, stat]*. arXiv: 1904.02679.

Wang, A.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; and Bowman, S. R. 2018. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. *arXiv:1804.07461 [cs]*. arXiv: 1804.07461.