1 Background

1.1 Representation learning

The development of rich distributed representations of text, also known as embeddings, is intimately linked to major advances in natural language processing (NLP) over the last decade. Generally, these representations are generated by embedding the high-dimensional language space in a vector space. Modern techniques of learning such mappings are based on neural networks and thus highly data-driven. Massive neural architectures are trained on millions of sentences to perform tasks such as language modeling and machine translation, during which they discover and encode linguistic features of the data in their hidden representations. The desired mapping is then extracted by ignoring the downstream task and repurposing the models solely for generating these embeddings.

1.2 Impact

Embeddings are a cornerstone of the modern NLP pipeline. The first step of any NLP system almost always involves the conversion of a string of raw text into the corresponding sequence of embeddings. However, a series of groundbreaking work around contextualized word embeddings, starting with CoVe and ELMo and more recently the Transformer-based architectures BERT and GPT-2, have redefined the role of embeddings from a humble preprocessing step to a means of advancing the state-of-the-art. For example, the authors of BERT show that simply using these contextual embeddings and fine-tuning an additional task-specific layer on top suffices to set new records on 11 types of tasks, including question answering and natural language inference [2].

1.3 Interpretability

There has been work showing that LSTMs can capture certain structurally-sensitive dependencies such as agreement [6, 7], and that Transformer models learn to encode syntactic information in its representations [4, 10]. However, it is still not fully understood exactly why these models work so well. The
current NLP paradigm of employing black-box neural models is naturally antithetic to interpretability, but the ability to understand and explain our models' decisions is very valuable nonetheless. Recent studies have shown that these models sometimes achieve spuriously good performances by learning heuristics inherent to the datasets \[8\], and are vulnerable to simple adversarial attacks \[5\]. The lack of interpretability in our models prevents us from validating their correctness, forcing us to adopt a reactive approach to any problems that may arise after they are productionized.

2 Research Directions

Currently, the lines of inquiry I intend to explore are:

1. **How do representations respond to distortions in the input?**

   \[3\] demonstrates that exposure bias is not a significant problem in autoregressive training of RNNs using maximum likelihood estimation. This suggests that RNNs may not be sensitive to small distortions in the input, but preliminary experiments have shown that even with major distortions, e.g. intervening halfway through decoding to feed the network several random tokens before allowing it to continue, the network is still able to “fail gracefully” in producing a reasonable completion of the sentence fragment before the intervention. I would like to investigate this phenomenon at a more fine-grained level: for certain linguistic constructions, e.g. subject-verb agreement, what kinds of input distortions will cause the network to make an error? What information encoded in the representations is being gained or lost when we perturb the inputs?

2. **How does the information content of representations change during fine-tuning?**

   The pretrained linguistic knowledge encoded in contextual embeddings, e.g. BERT, is transferred to drive downstream task performance via fine-tuning. For BERT, \[2\] simply augments the model with an additional task-specific layer and then fine-tunes parameters on the training set of the target task. Meanwhile, \[9\] examines several different fine-tuning approaches, including the multi-task learning and few-shot learning settings. I would like to explore how the information content of the original BERT layers change during these fine-tuning procedures, and what kinds of linguistic information these procedures successfully transfer to the downstream tasks.

To answer these questions, I will use methods such as probing classifiers, principal components analysis and representational similarity analysis \[1\].

3 Deliverables

- A research paper in fulfilment of the written report requirement
- Source code for the project, including model architectures, weights, datasets, usage scripts
References


