https://pair-code.github.io/lit

The Language Interpretability Tool (🔥LIT)

Google Research People + AI Research

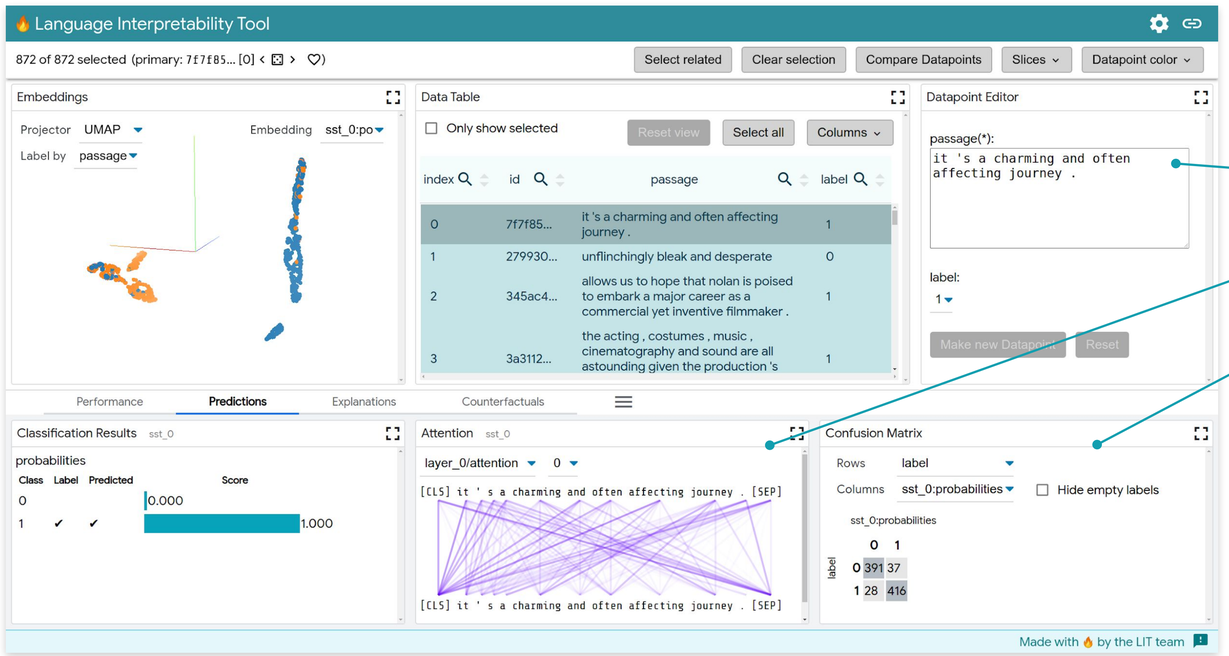
Ian Tenney, James Wexler, Jasmijn Bastings, Tolga Bolukbasi, Andy Coenen, Sebastian Gehrmann, Ellen Jiang, Mahima Pushkarna, Carey Radebaugh, Emily Reif, Ann Yuan
People + AI Research Initiative, Google Research

Overview

The Language Interpretability Tool is a visual, interactive model understanding tool for NLP.

Understand model behavior through interactive exploration, including model internals and counterfactual analysis. Ask powerful questions like:

- → **What kind of examples** does my model perform poorly on?
- → **Why did my model** make this prediction? Can I attribute it to adversarial behavior, or priors from the training set?
- → **Does it behave consistently** if I change things like textual style, verb tense, or pronoun gender?



Key Features

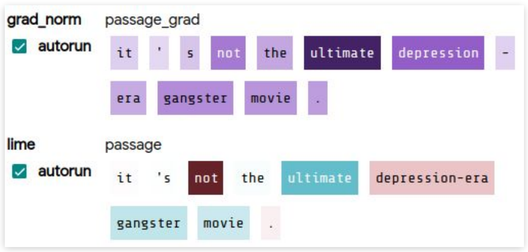
Explore, debug, and understand your models with:

- **Counterfactual generation** via manual edits or algorithmic generator plug-ins
- **Local explanations** via salience maps, attention, and rich visualizations
- **Aggregate analysis** with custom metrics, slice-by-feature, and embedding clusters
- **Side-by-side mode** to compare two models, or a pair of examples
- **Highly extensible** to new model types, including classification, regression, structured prediction, and seq2seq
- **Framework agnostic** and compatible with TensorFlow, PyTorch, and more

Case Studies

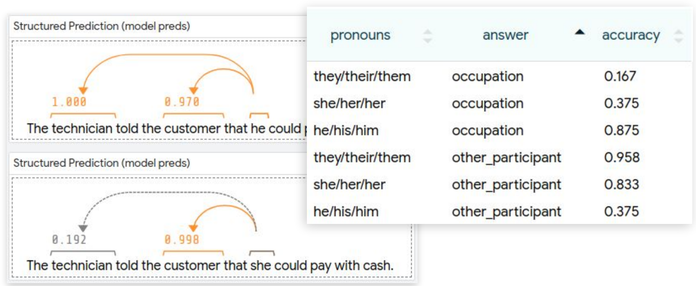
Sentiment Analysis

Find important words from salience maps, and validate with input perturbations



Gender Bias in Coreference

Use intersectional analysis to identify performance disparities



Try it Yourself

Use LIT to **accelerate error analysis,** and get a feel for your model and data.

Probe your model for fairness and robustness, and build trust with a human in the loop.

Reproduce and share analysis through web-based demos and modular, reusable components.

Simple Python API for models, datasets, and interpretability components:

```
models = {'foo': FooModel(...),
          'bar': BarModel(...)}
datasets = {'baz': BazDataset(...)}
server = lit.Server(models, datasets)
server.serve()
```

Full documentation, demos, and more at https://pair-code.github.io/lit