Rationalizing Neural Predictions Reimplementation

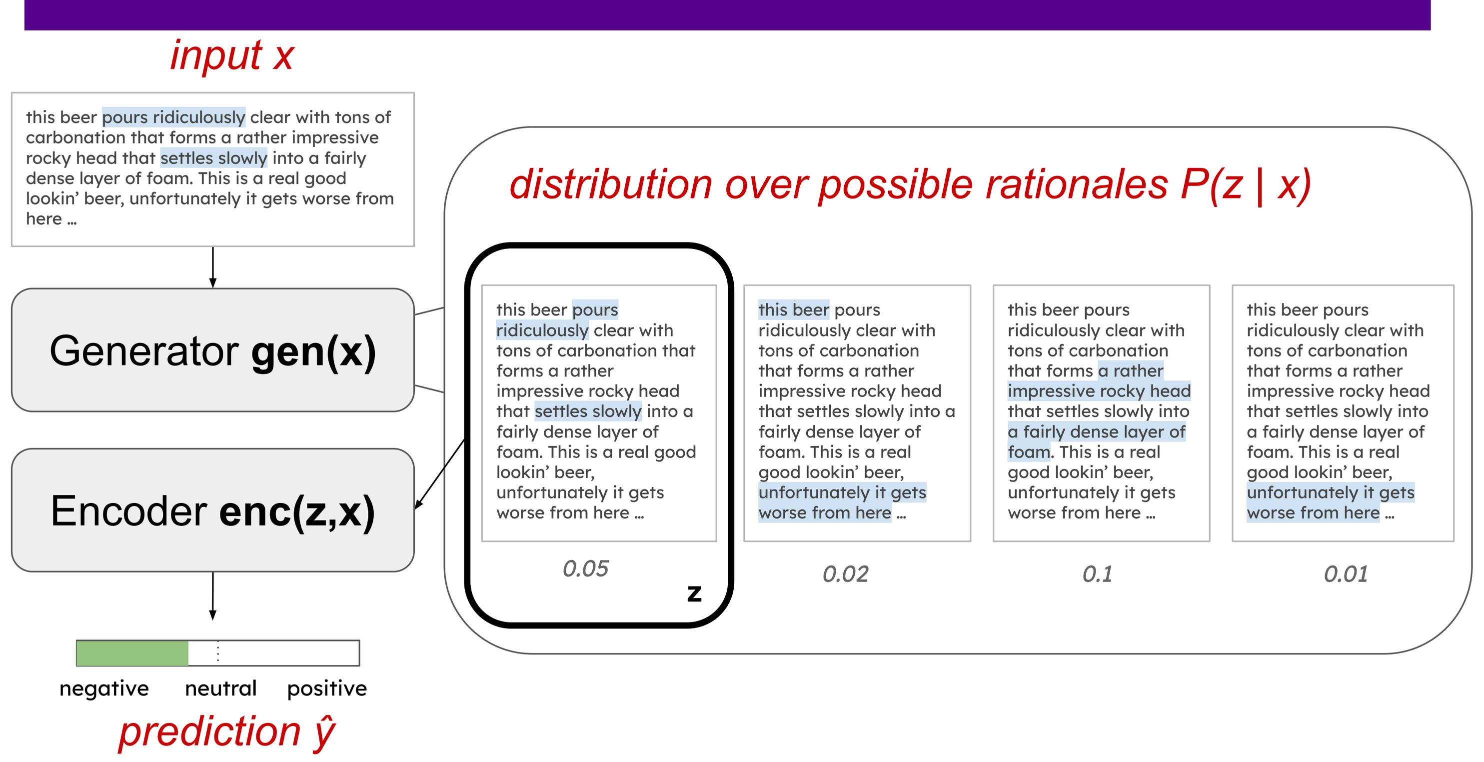
Xu Han, Jason Wang, Chloe Zheng
DS-GA 1011 NLP Tal Linzen
Advisor: Will Merrill

Background

Neural NLP models often come with a significant trade-off between accuracy and interpretability. How can we increase interpretability without significant accuracy loss?

- Tao Lei, Regina Barzilay and Tommi Jaakkola combine two modular components, a generator and an encoder, to extract short, continuous pieces of input text as justification for the sentiment prediction
- Originally built with Theano, we reimplemented the model with Pytorch
- We train the model on two datasets:
  a. Multi-aspect BeerAdvocate reviews (replication)
  b. Single-aspect Rotten Tomatoes Reviews (extension)

Methods



Generator: Produces set of rationales for analysis
Encoder: Predict sentiment based on selected rationale

Loss function

$$\mathcal{L}(\mathbf{z},\mathbf{x},\mathbf{y}) = \|\mathbf{enc}(\mathbf{z},\mathbf{x}) - \mathbf{y}\|_2^2$$

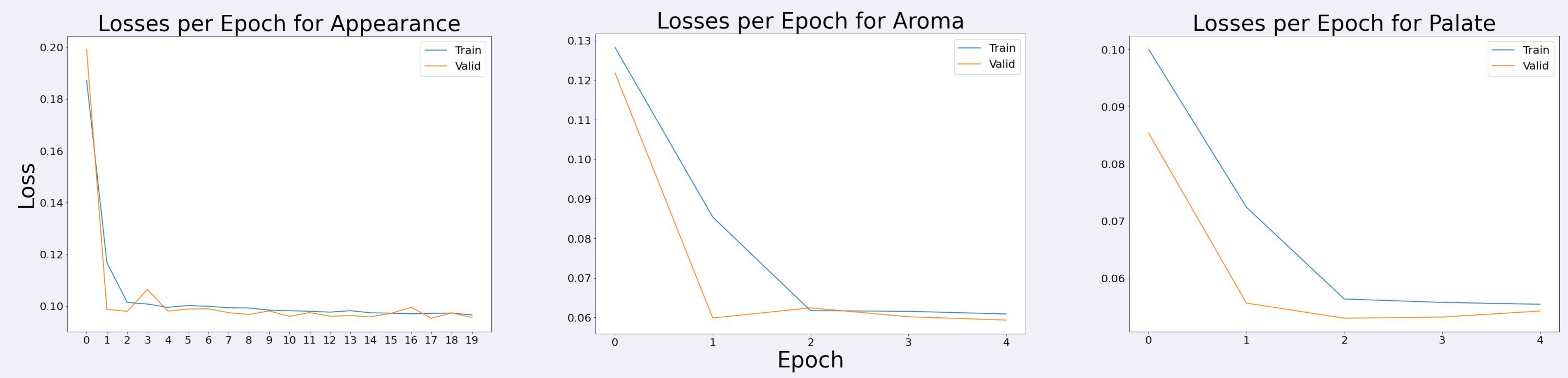
$$\Omega(\mathbf{z}) = \lambda_1\|\mathbf{z}\| + \lambda_2 \sum_t |\mathbf{z}_t - \mathbf{z}_{t-1}|$$

$$\text{cost}(\mathbf{z},\mathbf{x},\mathbf{y}) = \mathcal{L}(\mathbf{z},\mathbf{x},\mathbf{y}) + \Omega(\mathbf{z})$$

$$\min_{\theta_e,\theta_g} \sum_{(\mathbf{x},\mathbf{y})\in D} \mathbb{E}_{\mathbf{z}\sim\mathbf{gen}(\mathbf{x})}\left[\text{cost}(\mathbf{z},\mathbf{x},\mathbf{y})\right]$$

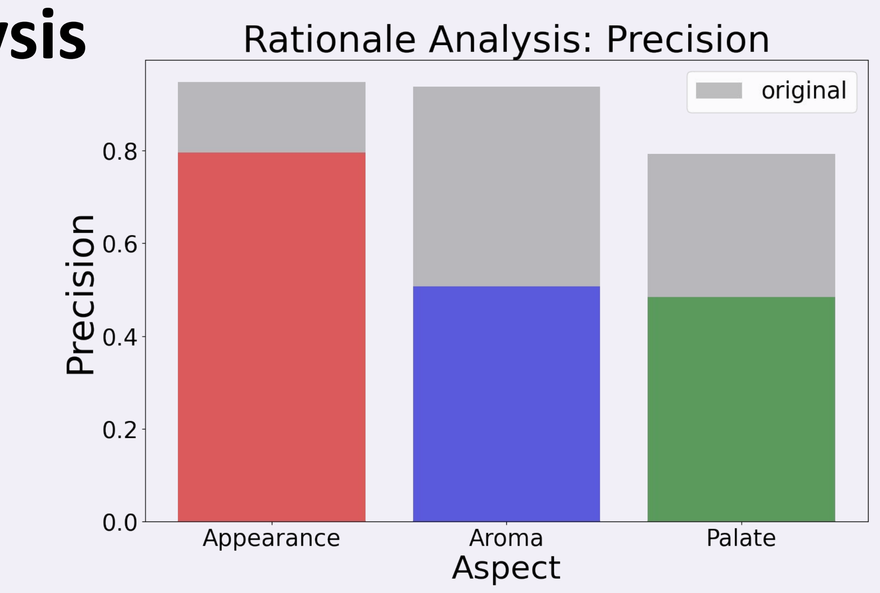
Results

Training Curves



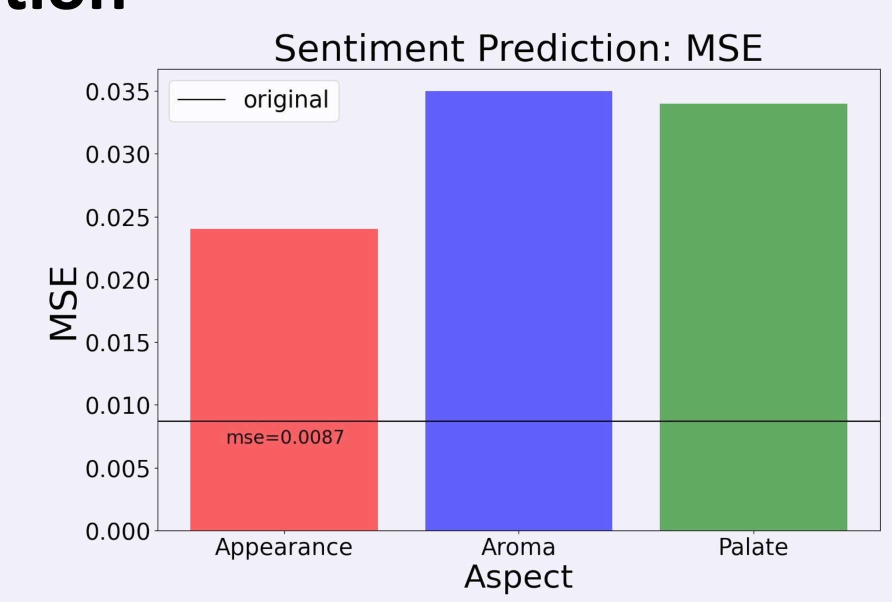
Rationale Analysis

*"Poured into a snifter. **Produces a small coffee head that reduces quickly. Black as night.** Pretty typical imp. **Roasted malts** hit on the nose. A **little sweet chocolate follows.** Big toasty character on the taste. In between i'm getting plenty of dark chocolate and some bitter espresso. It finishes with hop bitterness. **Nice smooth mouthfeel with perfect carbonation for the** style. Overall a nice stout i would love to have again, maybe with some age in it."*



Sentiment Prediction

Aspect	Sentiment
Appearance	4
Aroma	3.5
Palate	5



Rotten Tomatoes Extension

*"Maybe my favorite TV Series in the last two-plus years. **Every performance is just perfect, the set dec, the cinematography.. everything is so intentional.** Yes, it's a slow burn, and each episode builds ever so slightly on the last but trust me the finale will have you in an hour-long anxiety attack. Perfect season of television."*

Metric	Extension
MSE	0.0685
Precision	0.5362

Data and Metrics

BeerAdvocate Dataset includes beer reviews , sentiment labels [0,1] for 3 aspects, and rationale annotations

Rotten Tomatoes Dataset includes movie name, rating {0-5}, and review. We convert ratings to be [0,1] and annotated 100 reviews for rationale data

Metrics:
Sentiment Prediction - Mean Squared Error (MSE)
Rationale Analysis - Precision

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 \qquad \mathit{Precision} = \frac{TP}{TP + FP}$$

Conclusions and Future Directions

Conclusions:

- Pytorch version of Generator-Encoder model can generate rationales for sentiment prediction
- Unable to replicate exact metrics reported by original paper - could be due to lack of full dataset, Theano/PyTorch differences, different hyperparameters
- Rationale generation is extremely sensitive to hyperparameters
- Successfully adapted the model on Rotten Tomatoes dataset and generated intuitively reasonable rationales, but there was no baseline to compare to

Future directions:

- Identify additional metrics to measure quality of rationale prediction
- Identify what is a quality rationale annotation
- Train and test on a larger dataset with rationale annotations
- Compare performance datasets with multilingual languages
- Compare our Rotten Tomatoes results with other sentiment prediction and rationale generation models