HW3: Paper Reading

Emily Tseng
et397@cornell.edu

February 28, 2020

1 Introduction

In this assignment, we report on the following 3 papers:

1. Lei et al. (2016): *Rationalizing Neural Predictions*
2. Esmaeili et al. (2018): *Structured Neural Topic Models for Reviews*
3. Yu et al. (2016): *The Neural Noisy Channel*

2 Lei et al. (2016): *Rationalizing Neural Predictions*

Lei et al. (2016) report an approach to learning interpretable justifications for model predictions: *rationales*, or short, coherent pieces of an input text sufficient to producing the same prediction.

What is the model? Roughly how many parameters does the model have? The model is comprised of two modular neural components, a generator and an encoder. The generator $gen(x)$ produces a distribution of possible rationales $p(z|x)$, and the encoder $enc(z)$ makes a prediction y given a rationale. Given an input x (e.g. a full beer review), the generator $gen(x)$ produces a distribution of possible rationales $p(z|x)$ (e.g. sets of sentence fragments from the beer review), which is then consumed by the encoder $enc(z,x)$ to produce a target vector (e.g. a sentiment prediction for the beer review). The approach is generic—both the generator and the encoder can be implemented using a variety of RNNs or CNNs. The generator and encoder are trained jointly, in an end-to-end fashion.

What is the latent structure? The rationale z is the latent variable; it modulates how the input sequence x is translated to a prediction.

What is the inference approach? If they use amortized inference, what is the amortized model? How many parameters does it have? This paper proposes turning the inference problem of estimating $p(z|x;y)$ into an optimization over θ_g and θ_e via the gradient sampling method explained below. The number of parameters still depends on the models chosen for gen and enc.

What is the training objective? Conceptually, the generator and encoder are jointly trained over the following:

$$cost(z, x, y) = \|enc(z, x) - y\|_2^2 + \lambda_1 \|z\| + \lambda_2 \sum_t |z_t - z_{t-1}|$$

Here, the first term indirectly guides the generator towards producing rationales sufficient as a replacement for the input text, by penalizing encoder outputs where the generator output z produces a bad prediction. The rest of the objective regularizes towards desiderata further, by penalizing rationale selections that are too long (second term) and emphasizing continuity within a rationale (third term). (Note that these work because rationales are represented as binary vectors indicating whether a given token from the input has been selected as a rationale).

However, because rationales z are not given during training, the paper instead proposes a minimization of expected cost:

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

This is intractable because summation over Z is exponential given the size of the input data, so as an approximation, the paper proposes a sampled gradient descent method similar to REINFORCE. We describe it below.

What training / inference approximations does the paper make? We can approximate the gradient of the expectation w.r.t. the parameters of the generator from N sampled rationales z thus:

$$\begin{aligned}\frac{\partial \mathbb{E}_{z \sim gen(x)} [cost(z, x, y)]}{\partial \theta_g} &= \mathbb{E}_{z \sim gen(x)} [cost(z, y) \frac{\partial \log p(z|x)}{\partial \theta_g}] \\ &\approx \frac{1}{N} \sum_{i=1}^{N} cost(z_i, x_i) \frac{\partial \log p(z_i|x_i)}{\partial \theta_g}\end{aligned}$$

A similar approach for the gradient w.r.t. encoder parameters θ_e can be similarly approximated. The paper notes a sampled approximation of the expected cost objective as obtained separately for each input x to fit within an overall stochastic gradient descent learning method.

Does the paper evaluate interpretability? If so how? Yes—the entire paper is itself a proposal for a scheme for interpretability. Rationales must be coherent, as assessed by human qualitative review, and sufficient replacements for the input to generate the same downstream predictions, as assessed by quantitative success metrics. The authors report evaluations of the scheme on two tasks: (1) multi-aspect sentiment analysis on the BeerAdvocate review dataset and (2) similar-questions retrieval on the AskUbuntu question answering forum.

For (1), the authors constructed the task as a regression for per-aspect sentiment ratings (e.g. look, 5 stars; aroma, 2 stars). They evaluated on two metrics: the *mean squared error* of the regressions over all aspects and the *precision of the rationales generated*, calculated from sentence-level annotations per aspect. They also conducted a cursory qualitative review of the extracted rationales. On both sets of metrics, their approach outperformed a baseline bigram SVM model; their approach also outperformed an attention-based model on precision.

For (2), the authors constructed a question retrieval task: given a real-world dataset from AskUbuntu of long questions fraught with irrelevant details, the model should learn to extract rationales that represent the fraction of the original text sufficient to represent its content. The encoder was constructed to optimize the cosine similarity between similar questions vs. random non-similar ones. (This was optimized via hinge loss).

Extracted rationales were evaluated using the *mean average precision (MAP)* of the retrieval, contextualized against the performance when the full body of a question is used and the performance when just the question title is used. Rationales were able to achieve close to title precision, approx. 56.5%.

3 Esmaeili et al. (2018): *Structured Neural Topic Models for Reviews*

Esmaeili et al. (2018) propose VALTA (Variational Aspect-based Latent Topic Allocation), an approach that uses autoencoding topic models to learn aspect-based representations of reviews.

What is the model? Roughly how many parameters does the model have? VALTA is a combination of (1) an inference model with parameters θ_i and (2) a sentence-level generative network with parameters θ_g. Assume review $x_{i,u}$ written by user u about item i. We take $x_{i,u,s}$ to mean the sentence s of review $x_{i,u}$, and $z_{i,u,s}$ to mean the aspect assignment of sentence $x_{i,u,s}$. (In this paper's simplification, a sentence is assumed to discuss just one aspect.) With this, we can represent the aspect-specific topic distributions of $x_{i,u}$ as $\psi_{i,u}$. The structure of the model is depicted in Figure 1, where A denotes the total number of aspects, K denotes the number of sub-aspects, V is the vocabulary size and H is the hidden state size. Here $\omega_{i,u,s}$ denotes the aspect log-probabilities for a given $x_{i,u,s}$.

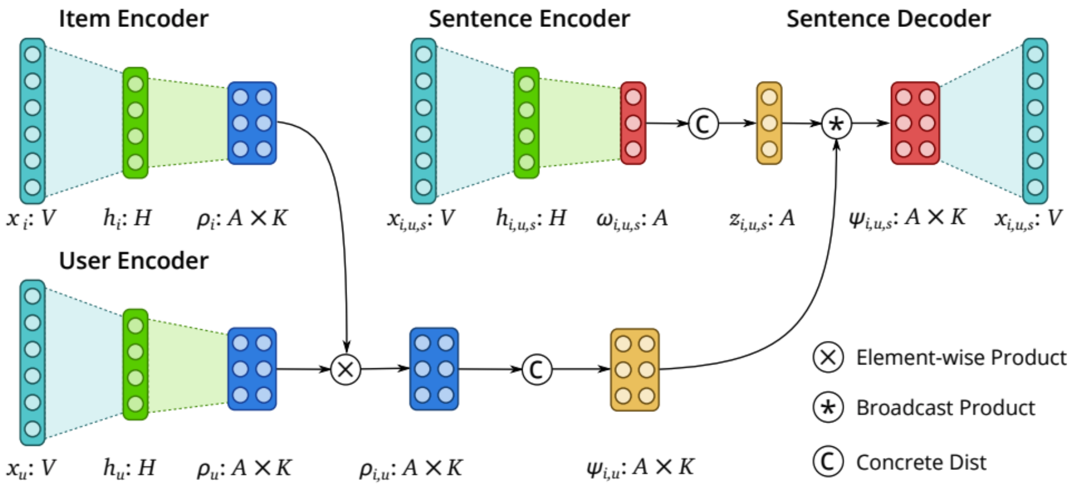

Figure 1: Figure 1 from Esmaeili et al. (2018), depicting the structure of VALTA.

The two component networks are thus:

$$\begin{aligned} q_{\theta_i}(\psi_{i,u}, z_{i,u} | x_i, x_u, x_{i,u}) &= q_{\theta_i}(\psi_{i,u} | x_i, x_u) \prod_s q_{\theta_i}(z_{i,u,s} | x_{i,u,s}) \\ p_{\theta_g}(x_{i,u}, z_{i,u}, \psi_{i,u}) &= \prod_s p_{\theta_g}(x_{i,u,s} | z_{i,u,s}, \psi_{i,u}) p(z_{i,u,s}) \prod_a p(\psi_{i,u,a}) \end{aligned}$$

What is the latent structure? VALTA has two latent variables: z, a set of aspect assignments for $x_{i,u,s}$, and ψ, a set of topic proportions for $x_{i,u}$.

What is the inference approach? The inference network learns a joint embedding $x_{i,u}$ of the separated distributions x_i and x_u, corresponding to item- and user-level aspect preferences. To use these embeddings to infer $p(\psi)$ (topic proportions) and $p(z)$ (per-sentence aspect assignment proportions), which would be best represented as Dirichlet and discrete distributions, respectively, the paper makes use of a technique to transform the hidden representations of $x_{i,u}$ into a Concrete distribution via a Gumbel softmax reparameterization to form $\psi_{i,u}$. We describe this further below.

If they use amortized inference, what is the amortized model? How many parameters does it have? I am not sure whether this constitutes amortized inference. Technically I suppose $\psi_{i,u}$ is a set of parameters for the user and the item shared/amortized for each sentence, but this is also inherent to the setup of the problem.

What is the training objective? First, users' overall ratings for an item $\hat{r}_{i,u}$ are predicted from importance-weighted ratings per aspect. Given global rating bias β_0 and item and user biases β_i and β_u, we can define:

$$\hat{r}_{i,u} = \beta_0 + \beta_i + \beta_u + \frac{1}{A}\sum_{a=1}^{A} \lambda_{i,u,a}\hat{r}_{i,u,a}$$

Aspect importance vectors $\lambda_{i,u,a}$ are derived from the sentence encoder (Figure 1). We define that encoder as $f_{\theta_g}^{aspect}(\cdot)$, and use it to extract λs from collections of user-item pairs:

$$\begin{aligned}
\lambda_i &= f_{\theta_g}^{aspect}(h_i) \\
\lambda_u &= f_{\theta_g}^{aspect}(h_u) \\
\lambda_{i,u} &= \frac{1}{2}(\lambda_i + \lambda_u)
\end{aligned}$$

Similarly, aspect ratings $\hat{r}_{i,u,a}$ are predicted by re-using the inputs to the concrete distributions ($\rho_{i,u}$ in Figure 1):

$$\hat{r}_{i,u,a} = \sum_{k=1}^{K} \rho_{i,u,a,k}$$

Using this method to predict ratings from (user, item) review pairs, we can define an overall objective for VALTA thus:

$$\begin{aligned}
\mathcal{L}(\theta_i, \theta_g) &= \mathcal{L}_{gen}^{x} + \mathcal{L}_{mse}^{r} + \mathcal{L}_{KL}^{\psi} + \mathcal{L}_{KL}^{z} \\
\mathcal{L}_{gen}^{x}(\theta_i, \theta_g) &= \mathbb{E}[\log \prod_s p_{\theta_g}(x_{i,u,s} | \psi_{i,u}, z_{i,u,s})] \\
\mathcal{L}_{mse}^{r}(\theta_g) &= \log p_{\theta_g}(r_{i,u} | x_i, x_u) \\
\mathcal{L}_{KL}^{z}(\theta_i, \theta_g) &= -\,\mathbb{E}[log \prod_s \frac{q_{\theta_i}(z_{i,u,s} | x_{i,u,s})}{p_{\theta_g}(z_{i,u,s})}] \\
\mathcal{L}_{KL}^{\psi}(\theta_i, \theta_g) &= -\,\mathbb{E}[log \frac{q_{\theta_i}(\psi_{i,u} | x_i, x_u)}{p_{\theta_g}(\psi_{i,u})}]
\end{aligned}$$

What training / inference approximations does the paper make? The paper approximates $p(z)$ and $p(\psi)$, the aspect assignment and topic proportions, respectively, as a relaxation implemented via Gumbel softmax. The paper goes on to describe several temperature experiments, and their effect on the interpretability of the generated topics.

Does the paper evaluate interpretability? If so how? Yes. The paper defined interpretability as the ability for the model to correctly label aspects in sentences and produce coherent topics. The authors also showed the learned representations could be used on simple classification tasks to enable aspect-based comparison. Finally, the authors showed the generative element of the model—its ratings predictions—could outperform mean squared error baselines on four datasets on four datasets: (1) BeerAdvocate, a body of beer reviews; (2) restaurant reviews from Yelp; (3) Clothing and (4) Movie review datasets from Amazon.

4 Yu et al. (2016): *The Neural Noisy Channel*

Yu et al. (2016) report a neural network implementation of the noisy channel model for seq2seq estimation that uses a latent alignment variable z to control when each output token is to be generated as the input token is read. Variational decomposition of the turns the decoding step from an intractable argmax to a tractable beam search.

What is the model? Roughly how many parameters does the model have? The model uses recurrent neural networks to parameterize the source and channel models of a noisy channel model. Consider the Bayes' rule decomposition of p(output sequence y | input sequence x):

$$p(y|x) = \frac{p(x|y)p(y)}{p(x)}$$

From this, we have the conditional *channel model* $p(x|y)$ and the unconditional *source model* $p(y)$. In practice, this is intractable at the decoding step, $argmax_y p(x|y)p(y)$. To solve this, the authors build on the Segment to Segment Neural Transduction model (SSNT) and incrementally construct the conditioning context by using the latent variable z. We describe this below.

What is the latent structure? The latent structure in use in this paper involves a latent alignment variable z, which indicates when each token of the output sequence is to be generated as the input sequence is read left to right. $z_j = i$ denotes that the output token at position j, y_j, is generated when the input sequence up through position i has been read. z assumes a sequence is read just once, left to right, and is thus restricted as a monotonically increasing alignment. Thus we have:

$$p(y|x) = \sum_z p(y,z|x)$$

$$p(y,z|x) = \prod_{j=1}^{|y|} p(z_j|z_{j-1}, x_1^{z_j}, y_1^{j-1}) p(y_j|x_1^{z_j}, y_1^{j-1})$$

The first term in the product is the probability of the alignment, decomposed into a sequence of conditionally independent SHIFT and EMIT operations that decide whether to read another token or stop. The probability of emitting an output is calculated:

$$p(a_{i,j} = EMIT|x_1^i, y_1^{j-1}) = \sigma(MLP(W_t[h_i; s_j] + b_t))$$

The probability of shifting instead of emitting is $1 - p(a_{i,j} = EMIT)$. Thus the alignment probabilities are:

$$p(z_j = i|z_{j-1}, y_1^{j-1}, x_1^i) = \begin{cases} 0 & \text{if } i < z_{j-1} \\ p(a_{i,j} = EMIT) & \text{if } i = z_{j-1} \\ (\prod_{i'=z_{j-1}}^{i-1} p(a_{i',j} = SHIFT))p(a_{i,j} = EMIT) & \text{if } i > z_{j-1} \end{cases}$$

The second term is the probability of the next token y_j, calculated by concatenating the aligned hidden state vectors s_j and h_{z_j} and pushing the results through a softmax layer:

$$p(y_j|x_1^{z_j}, y_1^{j-1}) \propto \exp(W_w[h_{z_j}; s_j] + b_w)$$

Training the SSNT minimizes the negative log-likelihood of the parallel corpus S using gradients of the following objective:

$$\mathcal{L}(\theta) = -\sum_{(x,y)\in S} \log p(y|x;\theta)$$

The parameters θ are pulled from the component alignment and word probability models.

What is the inference approach? If they use amortized inference, what is the amortized model? How many parameters does it have? The variational piece of this model happens at the decoding step, where we attempt to get $\hat{y} = argmax_y p(x|y)p(y)$. Instead of performing an exhaustive search of the entire vocabulary every time we extend to the next token in a sequence, we make use of an auxiliary direct model $q(y, z|x)$ to explore probable extensions of partial hypotheses. This is done via a dynamic programming technique using a Viterbi matrix and the following training objective to rank candidate beams:

$$O_{x_1^i, y_1^i} = \lambda_1 \log p(y_1^j|x_1^i) + \lambda_2 \log p(x_1^i|y_1^j) + \lambda_3 \log p(y_1^j) + \lambda_4|y_1^j|$$

The linear combination weights λ are tuned as hyperparameters during training.

What is the training objective? The variational decoding step described above uses the aforementioned objective to rank candidate beams; additionally, the SSNT itself uses the negative log-likelihood objective described earlier to train over stochastic gradient methods.

What training / inference approximations does the paper make? The key approximation in this paper is the auxiliary model $q(y, z|x)$ used at the decoding step, which uses a tuned beam search algorithm to partially explore the space of possible prefix hypotheses. Beam sizes K_1 and K_2 are hyperparameters to be tuned, but the assumption that this method can approximate the true argmax of the candidate output sequences is core to the paper.

Does the paper evaluate interpretability? If so how? The paper does not appear to address interpretability.

References

Esmaeili, B., Huang, H., Wallace, B. C., and van de Meent, J.-W. (2018). Structured neural topic models for reviews. *arXiv preprint arXiv:1812.05035*.

Lei, T., Barzilay, R., and Jaakkola, T. (2016). Rationalizing neural predictions. *arXiv preprint arXiv:1606.04155*.

Yu, L., Blunsom, P., Dyer, C., Grefenstette, E., and Kocisky, T. (2016). The neural noisy channel. *arXiv preprint arXiv:1611.02554*.