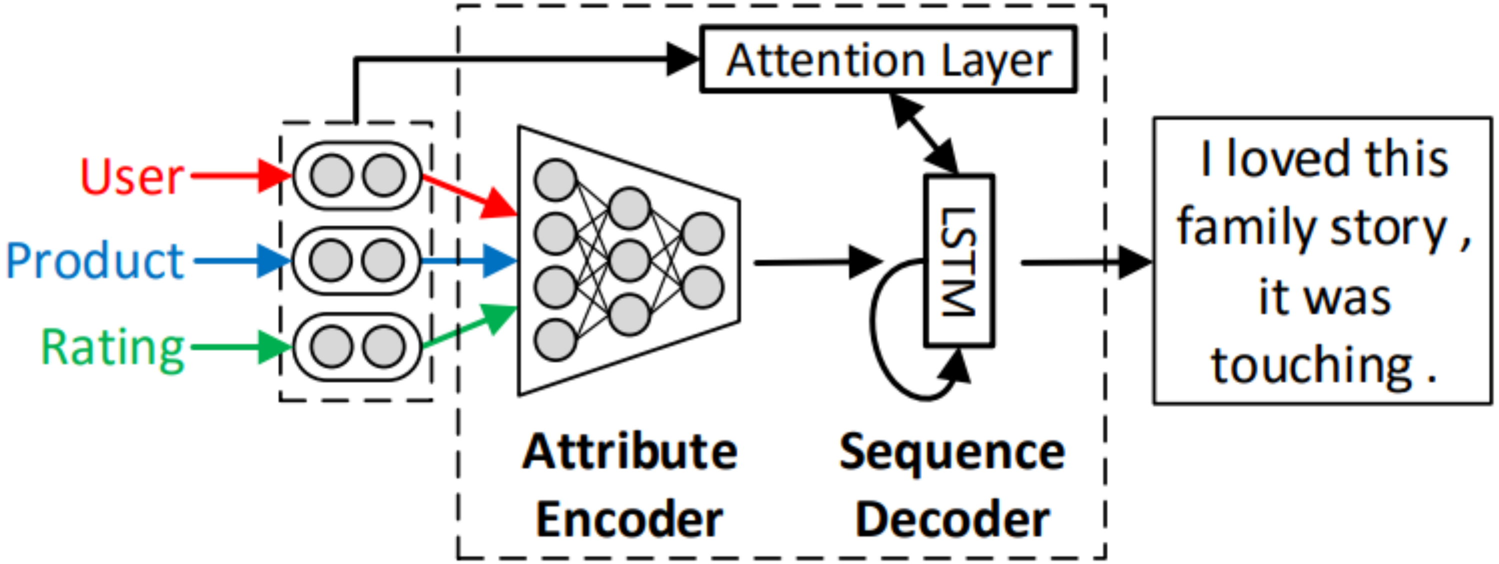
Learning to Generate Product Reviews from Attributes

Authors: Li Dong, Shaohan Huang, Furu Wei, Mirella Lapata, Ming Zhou and Ke Xu

Presenter: Yimeng Zhou

Introduction

- Presents an **attention-enhanced attribute-to-sequence model** to generate product reviews for given attribute information such as user, product and rating.



Introduction

- Challenges:
  - Variety of candidate reviews that satisfy the input attributes.
  - Unknown or latent factors that influence the generated reviews, which renders the generation process non-deterministic.
- Rating explicitly determine the usage of sentiment words.
- User and product implicitly influence word usage.

Compared to Prior work

- Most previous work focuses on using rule-based methods or machine learning techniques for sentiment classification, which classifies reviews into different sentiment categories
- In contrast, this model is mainly evaluated on the review generation task rather than classification. Moreover, it uses an attention mechanism in encoder-decoder model

Model - Overview

- Input attributes $a = (a_1, \cdots, a_{|a|})$
- Generate product review $r = (y_1, \cdots, y_{|r|})$ to maximize the conditional probability *p(r|a)*
  - |a| is fixed to 3 with userID, productid and rating.

Model - Overview

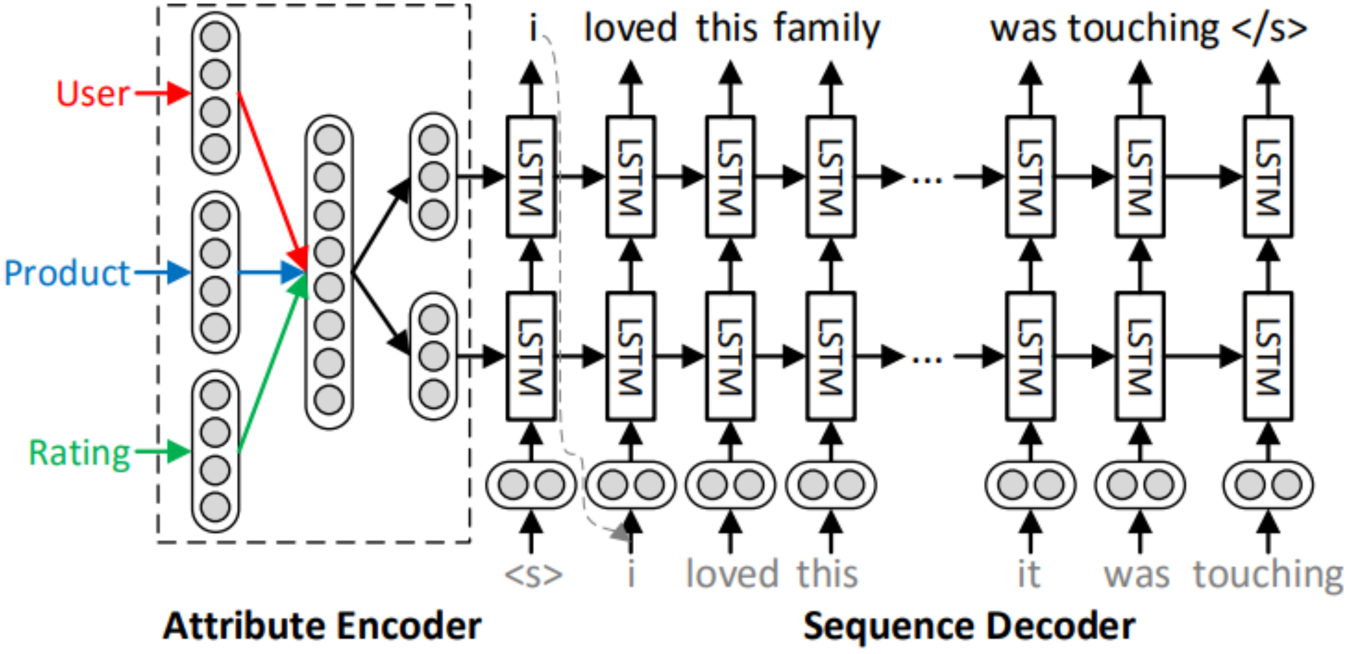
- The model learns to compute the likelihood of generated reviews given input attributes.
- This conditional probability *p(r|a)* is decomposed to

$$p\left(r|a\right) = \prod_{t=1}^{|r|} p\left(y_t|y_{<t}, a\right) \tag{1}$$

where $y_{<t} = (y_1, \cdots, y_{t-1})$.

Model – Three parts

- Attribute Encoder
- Sequence Decoder
- Attention Mechanism
  - Att2seq model without attention mechanism



Model – Attribute Encoder

- Use multilayer perceptrons to encode input attributes into vector representations that are used as latent factors for generating reviews.
- Input attributes a are represented by low-dimensional vectors. The attribute a_i's vector $g(a_i)$ is computed via

$$\mathbf{g}(a_i) = W_i^a \mathbf{e}(a_i)$$

- Where $W_i^a \in \mathbb{R}^{m \times |a_i|}$ is a parameter matrix and $e(a_i)$ is a one-hot vector representing the presence or absence of a_i.

Model – Attribute Encoder

- Then these attribute vectors are concatenated and fed into a hidden layer which outputs the encoding vector. The output of the hidden layer is computed as:

$$\mathbf{a} = \tanh\left(H[\mathbf{g}\left(a_1\right), \cdots, \mathbf{g}\left(a_{|a|}\right)] + \mathbf{b}_a\right) \quad (3)$$

Model – Sequence Decoder

- The decoder is built by stacking multiple layers of recurrent neural networks with long short-term memory units to better handle long sequences.
- RNNs use vectors to represent information for the current time step and recurrently compute the next hidden states.

Model – Sequence Decoder

- The LSTM introduces several gates and explicit memory cells to memorize or forget information, which enables networks learn more complicated patterns
- The n-dimensional hidden vector in layer l and time step t is computed via

$$\mathbf{h}_t^l = f\left(\mathbf{h}_{t-1}^l, \mathbf{h}_t^{l-1}\right)$$

Model – Sequence Decoder

- The LSTM unit is given by

$$\begin{pmatrix} \mathbf{i} \\ \mathbf{f} \\ \mathbf{o} \\ \mathbf{g} \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} W^l \begin{pmatrix} \mathbf{h}_t^{l-1} \\ \mathbf{h}_{t-1}^l \end{pmatrix} \tag{5}$$

$$\mathbf{p}_t^l = \mathbf{f} \odot \mathbf{p}_{t-1}^l + \mathbf{i} \odot \mathbf{g}$$

$$\mathbf{h}_t^l = \mathbf{o} \odot \tanh\left(\mathbf{p}_t^l\right)$$

where $\tanh$, sigm, and $\odot$ are element-wise operators, and $W^l \in \mathbb{R}^{4n \times 2n}$ is a weight matrix for the l-th layer.

Model – Sequence Decoder

- Finally, for the vanilla model without using an attention mechanism, the predicted distribution of the t-th output word is:

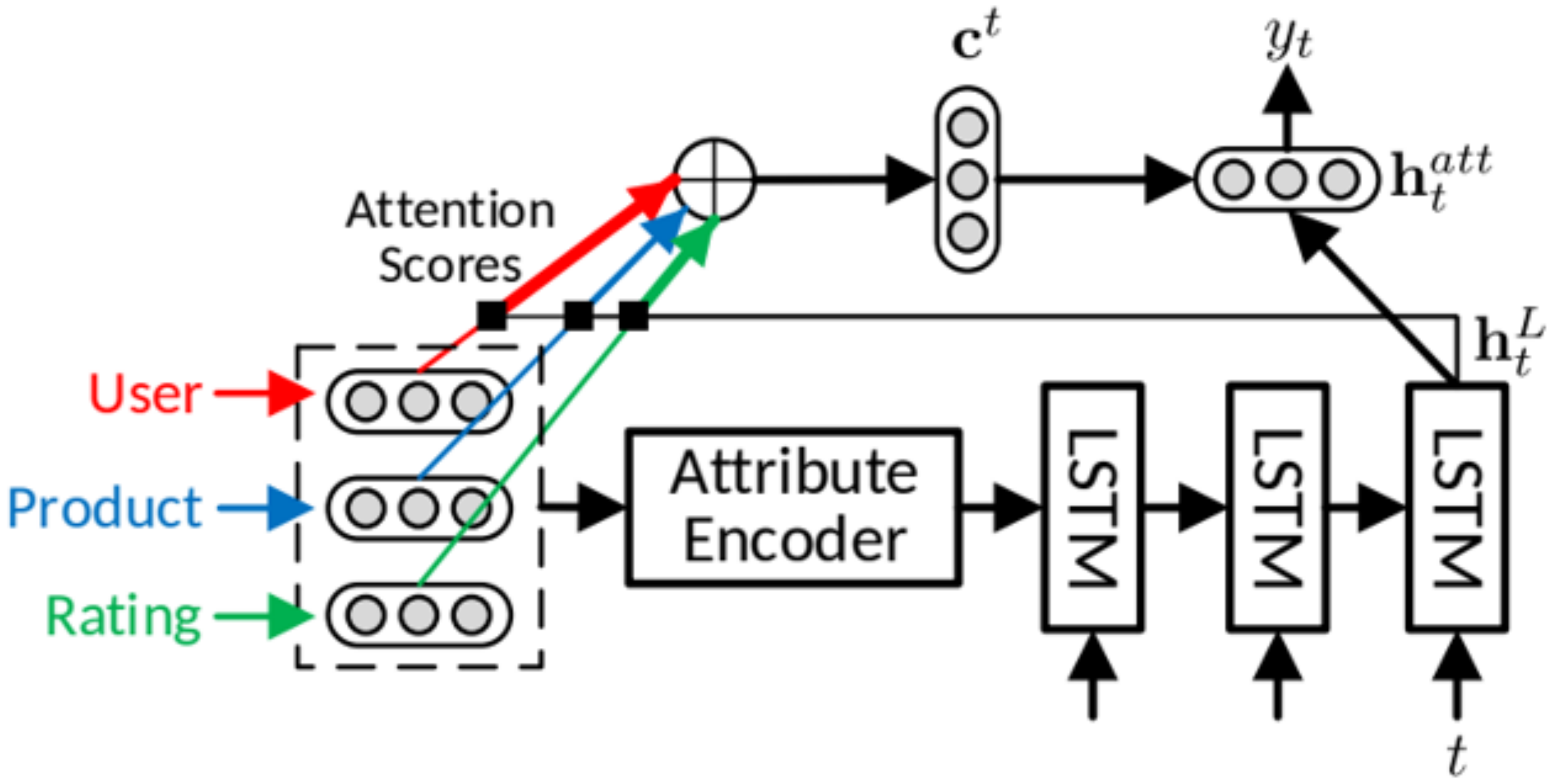
$$p\left(y_t | y_{<t}, a\right) = \operatorname{softmax}_{y_t}\left(W^p \mathbf{h}_t^L\right) \qquad (6)$$

where $W^p \in \mathbb{R}^{|V_r| \times n}$ is a parameter matrix.

Model – Attention Mechanism

- Better utilize encoder-side information
- The attention mechanism learns soft alignments between generated words and attributes, and adaptively computes encoder-side context vectors used to predict the next tokens.

Model – Attention Mechanism



Model – Attention Mechanism

- For the t-th time step of the decoder, we compute the attention score of attribute a_i via

$$s_i^t = \exp\left(\tanh\left(W^s\left[\mathbf{h}_t^L, \mathbf{g}\left(a_i\right)\right]\right)\right)/Z \qquad (7)$$

- Z is a normalization term that ensures $\sum_{i=1}^{|a|} s_i^t = 1$

Model – Attention Mechanism

- Then the attention context vector c^t is obtained by

$$\mathbf{c}^t = \sum_{i=1}^{|a|} s_i^t \, \mathbf{g}(a_i)$$

which is a weighted sum of attribute vectors.

Model – Attention Mechanism

- Further employ the vector to predict the t-th output token as

$$\mathbf{h}_t^{att} = \tanh\left(W_1\mathbf{c}^t + W_2\mathbf{h}_t^L\right) \tag{9}$$

$$p\left(y_t|y_{<t}, a\right) = \mathrm{softmax}_{y_t}\left(W^p\mathbf{h}_t^{att}\right) \tag{10}$$

where $W^p \in \mathbb{R}^{|V_r|\times n}$, $W_1 \in \mathbb{R}^{n\times m}$ and $W_2 \in \mathbb{R}^{n\times n}$ are three parameter matrices.

Model – Attention Mechanism

- Aim at maximizing the likelihood of generated reviews given input attributes for the training data.
- The optimization problem is to maximize

$$\sum_{(a,r)\in\mathcal{D}} \log p\left(r|a\right)$$

- Avoid overfitting: insert dropout layers between different LSTM layers as suggested in Zaremba et al. (2015).

Experiments

- Dataset: built upon Amazon product data including reviews and metadata spanning.
- The whole dataset is randomly split into three parts TRAIN, DEV and TEST (70%. 10%, 20%)
- Parameter settings:
  - Dimension of Attributes vectors:64
  - Dimension of word embeddings and hidden vectors:512
  - Uniform distribution [-0.08,0.08]
  - Batch size, smoothing constant, learning rate: 50, 0.95, 0.0002
  - Dropout rate: 0.2
  - Gradient values: [-5, 5]

Results

Method	BLEU-4 (%)	BLEU-1 (%)
Rand	0.86	20.36
MELM	1.28	21.59
NN-pr	1.53	22.44
NN-ur	3.61	26.37
Att2Seq	4.51	30.24
Att2Seq+A	**5.03***	**30.48***

Table 1: Evaluation results on the TEST set of Amazon data. *: significantly better than the second best score ($p < 0.05$).

Results - Polarities

Method	MELM	Att2Seq	Att2Seq+A
Accuracy (%)	59.00	88.67	**93.33***

Table 2: We manually annotate some polarity labels (positive or negative) for generated reviews and compute accuracy by comparing them with the input ratings. *: significantly better than the second best accuracy ($p < 0.05$).

Results – Ablation

Method	BLEU-4 (%)	BLEU-1 (%)
Att2Seq+A	5.01	30.23
AvgEnc	4.07	28.13
NoStack	4.73	29.58
w/o user	4.10	26.87
w/o product	4.13	27.15
w/o rating	4.12	27.98

Table 3: Model ablation results on the DEV set.

Results – Attention Scores

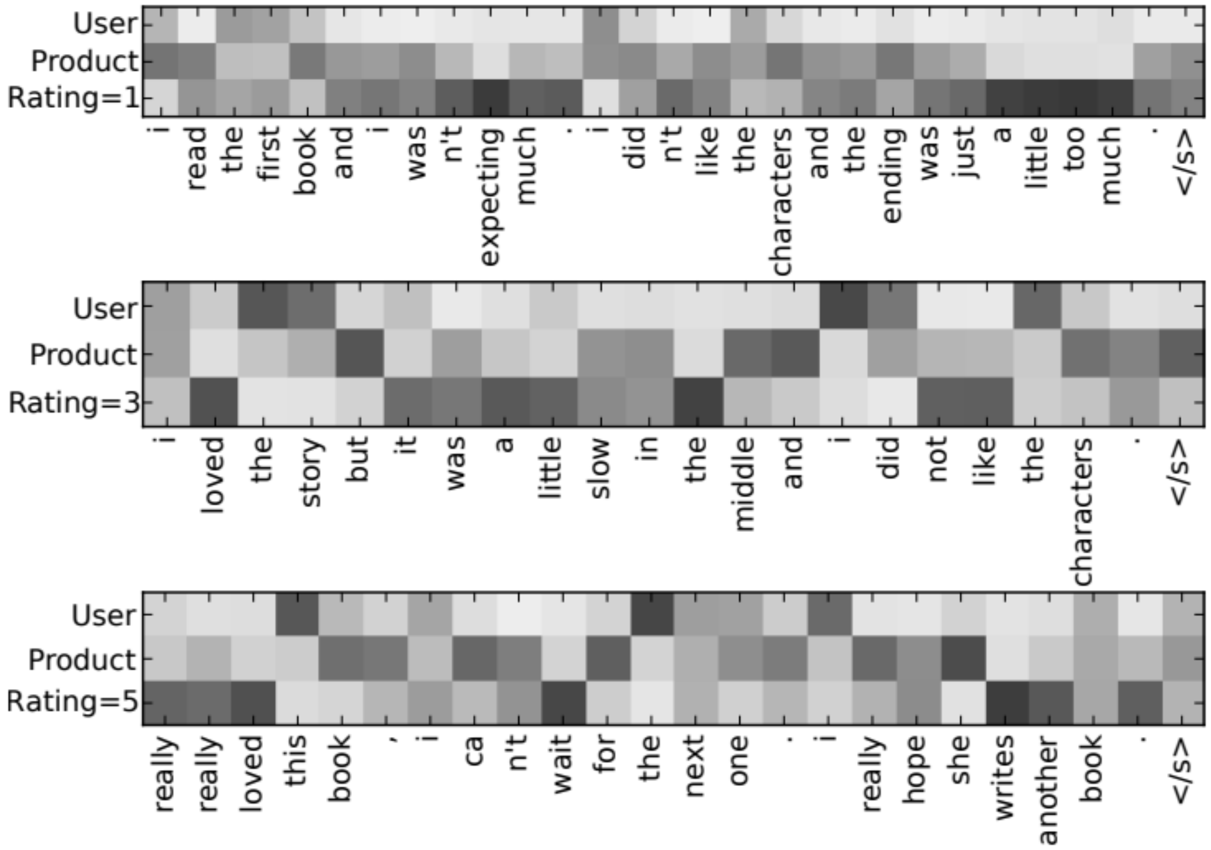


Figure 4: Examples of attention scores (Equation (7)) over three attributes. Darker color indicates higher attention score.

Results – Control Variable

U	P	R	Generated Review
A	V	1	i'm sorry to say this was a very boring book. i didn't finish it. i'm not a new fan of the series, but this was a disappointment.
A	V	3	this was a nice story. i liked the characters and the story line. i'm not sure i'd read another by this author.
A	V	5	this was a very good book. i enjoyed the characters and the story line. i'm looking forward to reading more in this series.
B	W	5	i couldn't put it down. it was a great love story. i can't wait to read the next one.
C	W	5	enjoyable story that keeps you turning the pages. the characters are well developed and the plot is excellent. i would recommend this book to anyone who enjoys a good love story.
D	W	5	i loved this book. i could not put it down. i loved this story and the characters. i will be reading the next book.
E	X	1	i read this book because i was looking for something to read. this book was just too much like the others. i thought the author was going to be a good writer, but i was disappointed.
E	Y	1	i was disappointed. i read the first chapter and then i was bored. i read the whole thing, but i just couldn't get into it.
E	Z	1	this book was just too much. i read the whole thing, but i didn't like the way the author ended it. i was hoping for a different ending.

Table 4: **U**: User. **P**: Product. **R**: Rating. This table shows some generated examples of the Att2Seq+A model. In every group, two attributes are kept unchanged, while the other attribute has different values. For instance, in the first group, we use different ratings ranging from 1 (the lowest score) to 5 (the highest score) with the same user and product to generate reviews. The users and products are anonymized by A-E and V-Z.

Improvements

- Use more fine-grained attributes as the input of our model.
  - Conditioned on device specification, brand, user’s gender, product description, etc.
  - Leverage review texts without attributes to improve the sequence decoder.

Conclusion

- Proposed a novel product review generation task, in which generated reviews are conditioned on input attributes,
- Formulated a neural network based attribute-to-sequence model that uses multilayer perceptrons to encode input attributes and employs recurrent neural networks to generate reviews.
- Introduced an attention mechanism to better utilize input attribute information.

Thank you!