Digital "Mad Men": Training RNN as an Ad Copywriter

Christina Lin, Paul Shved, Sridatta Thatipamala

CS224N Spring 2019

Problem

Can we use neural style transfer for text to generate short text ads, given a set of keywords?

Generating ad copy can improve efficiency of a $100B+ industry!

Natural language generation challenges:
  ● Should sound grammatical/realistic
  ● Should not be generic or boring sentences
  ● Should contain the keywords that we want to include

Existing approaches:
  ● Language modeling: generic output, not much control over output
  ● Prototype-then-edit [2]: transfer new content to an existing "prototype" sentence

We investigated the prototype-then-edit approach

Approach

Task

Train a "neural editor" that accepts a prototype sentence and an "edit vector," and applies the edit to the sentence.

Prototype sentences

277,228 one-sentence Amazon product reviews.

Examples:
  ● "It's easy to use and easy to clean out too"
  ● "These Pyrex dishes were are great buy for the price"
  ● "Great for the price"

Ad keywords

814 keyword phrases of 1-2 words each. Used to synthesize edits to the prototype sentence

Examples:
  ● "car loans"
  ● "life insurance"
  ● "holiday quotes"

Model

Seq2Seq architecture. 3-layer BiLSTM encoder. 3-layer LSTM decoder with source attention, conditioned on edit vector

Model setup. We used Amazon reviews instead of Yelp reviews. Diagram from [2]

Edit vectors are constructed by concatenating the sum of added words and deleted words

Training

Pairs of Amazon reviews with Jaccard similarity of >= 0.4

Inference

For each Keyword, randomly sample 1,000 Reviews, decode, select best.

References


Results

Only about 2.3% of results produced desired edits

Source | Edits | Actual decoded target
--- | --- | ---

```
one rug now has an odd nap to it that makes it harder to vacuum
```

Add: ["coolkware"]

mine has this complicated "unsieve" carafe that makes it harder to do

```
to my surprise, you can invent a better shower curtain ring, and this is it
```

Add: ["keyboard"]

to my surprise, the keyboard is very nice, and more importantly to change easily

Most results:
  ● Achieved style transfer
  ● Did not actually add keywords

Analysis

We can trade off one metric for the other, but we can not achieve all

Baseline #2 performs well

Style Transfer Metrics

<table>
<thead>
<tr>
<th>Style Transfer Strength (Classification)</th>
<th>Content Preservation (Cosine Similarity)</th>
<th>Perplexity</th>
</tr>
</thead>
</table>

```
when it didn’t help, it didn’t help, stop, it just didn’t help
```

Quote: "when it didn’t help, it didn’t help, stop, it just didn’t help"