Language Generation
Announcements

• Reading: language generation paper for today
  • This is the baseline model for the E2E language generation challenge
  • Your HW4 implementation is based on this model

• Wed, Nov 20\textsuperscript{th}: Or Biran, Elementary Cognition: Dialog systems.

• Final exam: In-class, Dec. 9\textsuperscript{th}: see syllabus

• Monday, Nov 25\textsuperscript{th}: Bias
We teach A.I. Systems Everything, Including our Biases
Beam search complexity

- **Time complexity:** linear because it only expands $b$ nodes at each level
  - Worst case: $O(Bm)$ where $B$ is beam and $m$ is maximum depth of any path

- **Space complexity:** linear
  - Worst case: $O(Bm)$

- Not optimal
Today

Extractive summarization of news articles

Language generation

The E2E task

Baseline model used in the task
Another neural summarization approach

- Extractive summarization of news
  - Single document summarization

- Data source: Daily News
  - Bulleted highlights of each article

- Neural Summarization by Extracting Sentences and Words
  - Cheng and Lapata, Edinburgh
Example from Daily News

AFL star blames vomiting cat for speeding
Adelaide Crows defender Daniel Talia has kept his driving license, telling a court he was speeding 36km over the limit because he was distracted by his sick cat.
The 22-year-old AFL star, who drove 96km/h in a 60km/h road works zone on the South Eastern expressway in February, said he didn’t see the reduced speed sign because he was so distracted by his cat vomiting violently in the back seat of his car.
In the Adelaide magistrates court on Wednesday, Magistrate Bob Harrap fined Talia $824 for exceeding the speed limit by more than 30km/h.
He lost four demerit points, instead of seven, because of his significant training commitments.

- Adelaide Crows defender Daniel Talia admits to speeding but says he didn’t see road signs because his cat was vomiting in his car.
- 22-year-old Talia was fined $824 and four demerit points, instead of seven, because of his ‘significant’ training commitments.

Figure 1: DailyMail news article with highlights. Underlined sentences bear label 1, and 0 otherwise.

Cheng and Lapata 2016
AFL star blames vomiting cat for speeding
Adelaide Crows defender Daniel Talia has kept his driving license, telling a court he was speeding 36km over the limit because he was distracted by his sick cat.
The 22-year-old AFL star, who drove 96km/h in a 60km/h road works zone on the South Eastern expressway in February, said he didn’t see the reduced speed sign because he was so distracted by his cat vomiting violently in the back seat of his car.
In the Adelaide magistrates court on Wednesday, Magistrate Bob Harrap fined Talia $824 for exceeding the speed limit by more than 30km/h.
He lost four demerit points, instead of seven, because of his significant training commitments.

- Adelaide Crows defender Daniel Talia admits to speeding but says he didn’t see road signs because his cat was vomiting in his car.
- 22-year-old Talia was fined $824 and four demerit points, instead of seven, because of his ’significant’ training commitments.

Figure 1: DailyMail news article with highlights. Underlined sentences bear label 1, and 0 otherwise.

Paraphrasing Compression Fusion

Cheng and Lapata 2016
Two Tasks

• Input: Document D: \( \{s_1, \ldots, s_m\} \) consisting of words \( w_1, \ldots, w_n \)

• Sentence extraction
  • Select a subset of \( j \) sentences, \( j < m \)
  • Score each sentence and predict label \( y_L \in \{0,1\} \)
  • Objective: Maximize all sentence labels given D and weights \( \theta \)

• Word extraction
  • Find a subset of words in D and their optimal ordering
  • Language generation task with output vocabulary restricted to input D vocabulary
  • Objective: Maximize the likelihood of generated sentences, further decomposed by considering conditional dependencies among their words
Training Data

• Sentence extraction
  • Highlights are abstracts
  • Find the s in D that most closely matches a highlight sentence
    • Positive, unigram and bigram matches, #entities
  • 200K document/summary pairs, summary size = 30% document

• Word extraction
  • Retain highlights with all words from D
  • Find neighbors of words not in D and substitute
  • 170K document/summary pairs
Neural Summarization Architecture

• Hierarchical document reader
  • Derive meaning representation of document from its constituent sentences

• Attention based hierarchical content extractor

• Encoder-decoder architecture
Document Reader

- CNN sentence encoder
  - Useful for sentence classification
  - Easy to train

- LSTM document encoder
  - Avoids vanishing gradients
**CNN**

\[ f^i_j = \tanh(W_{j:j+c-1} \otimes K + b) \]

- Where \( W \in \mathbb{R}^{n \times d} \) and \( d = \) word embedding dimension, \( n = \) #words in sentence
- \( K \) a kernel of width \( c \), \( b \) the bias
- \( f^i_j \) = the jth item in the ith feature map \( f^i \)
- Perform max pooling over time to obtain a single feature to represent the sentence

\[ s_{i,K} = \max_j f^i_j \]
These are words in the sentence.
Recurrent document encoder

• LSTM to compose a sequence of sentence vectors into a document vector

• The hidden states of the LSTM = a list of partial representations
  • Each focuses on the corresponding input sentence given previous content

• Altogether constitute document representation
these are words in the sentence
Sentence Extractor

• Applies attention to directly extract sentences after reading them

\[ \tilde{h}_t = \text{LSTM}(p_{t-1}s_{t-1}, \tilde{h}_{t-1}) \]

\[ p(y_L(t) = 1|D) = \sigma(\text{MLP}(\tilde{h}_t : h_t)) \]

• $\tilde{h}$ extractor hidden state, $h$ encoder hidden state
  • Attends to relation between extractor and encoder hidden state

• MLP takes as input concatenated $\tilde{h}$ and $h$

• $p_t$ degree to which extractor believes previous sentence should be extracted
Word Extractor

• Instead of extracting sentence, extracts next word
• Uses hierarchical attention to attend to sentence and word within sentence
• Output vocabulary restricted to input sentence

• -> conditional language model with vocabulary constraint
Datasets

- Daily Mail
  - 200K training
  - 500 test

- DUC 2002
  - 567 documents with 2 summaries each
Results

<table>
<thead>
<tr>
<th></th>
<th>DUC 2002</th>
<th>ROUGE-1</th>
<th>ROUGE-2</th>
<th>ROUGE-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>LEAD</td>
<td>43.6</td>
<td>21.0</td>
<td>40.2</td>
<td></td>
</tr>
<tr>
<td>LREG</td>
<td>43.8</td>
<td>20.7</td>
<td>40.3</td>
<td></td>
</tr>
<tr>
<td>ILP</td>
<td>45.4</td>
<td>21.3</td>
<td>42.8</td>
<td></td>
</tr>
<tr>
<td>NN-ABS</td>
<td>15.8</td>
<td>5.2</td>
<td>13.8</td>
<td></td>
</tr>
<tr>
<td>TGRAPH</td>
<td>48.1</td>
<td>24.3</td>
<td></td>
<td>—</td>
</tr>
<tr>
<td>URANK</td>
<td>48.5</td>
<td>21.5</td>
<td></td>
<td>—</td>
</tr>
<tr>
<td>NN-SE</td>
<td>47.4</td>
<td>23.0</td>
<td>43.5</td>
<td></td>
</tr>
<tr>
<td>NN-WE</td>
<td>27.0</td>
<td>7.9</td>
<td>22.8</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>DailyMail</th>
<th>ROUGE-1</th>
<th>ROUGE-2</th>
<th>ROUGE-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>LEAD</td>
<td>20.4</td>
<td>7.7</td>
<td>11.4</td>
<td></td>
</tr>
<tr>
<td>LREG</td>
<td>18.5</td>
<td>6.9</td>
<td>10.2</td>
<td></td>
</tr>
<tr>
<td>NN-ABS</td>
<td>7.8</td>
<td>1.7</td>
<td>7.1</td>
<td></td>
</tr>
<tr>
<td>NN-SE</td>
<td>21.2</td>
<td>8.3</td>
<td>12.0</td>
<td></td>
</tr>
<tr>
<td>NN-WE</td>
<td>15.7</td>
<td>6.4</td>
<td>9.8</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: ROUGE evaluation (%) on the DUC-2002 and 500 DailyMail samples.
Later results

(Nallapati et al 2017):

- RNN over sentence embeddings and output concatenated, representation of entire document by averaging, representation of summary so far by summing outputs

(Kedzie, McKeown and Daume 2018):

- Simpler is better
  - Averaging word embeddings, pre-trained fine, no need for summary so far
  - Order most important for news
State of the Art


• Is this a good task?

• Could you imagine other summarization tasks for which there might be data?
Language Generation

• The E2E Challenge
  • 62 submissions by 17 institutions, 11 countries, 1/3 from industry

• Restaurant recommendations

• Generation of one or more sentences from an input meaning representation (MR)

• Large and varied dataset
Input Data

• Unordered sets of attributes

• MR:
  • Name[The Wrestlers], pricerange[cheap], customerrating[1 of 5]

• Output
  • The Wrestlers offer competitive prices but it isn’t highly rated by customers.
# Domain Ontology

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Data Type</th>
<th>Example value</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>verbatim string</td>
<td><em>The Eagle, ...</em></td>
</tr>
<tr>
<td>eatType</td>
<td>dictionary</td>
<td><em>restaurant, pub, ...</em></td>
</tr>
<tr>
<td>familyFriendly</td>
<td>boolean</td>
<td><em>Yes / No</em></td>
</tr>
<tr>
<td>priceRange</td>
<td>dictionary</td>
<td><em>cheap, expensive, ...</em></td>
</tr>
<tr>
<td>food</td>
<td>dictionary</td>
<td><em>French, Italian, ...</em></td>
</tr>
<tr>
<td>near</td>
<td>verbatim string</td>
<td><em>market square, Cafe Adriatic, ...</em></td>
</tr>
<tr>
<td>area</td>
<td>dictionary</td>
<td><em>riverside, city center, ...</em></td>
</tr>
</tbody>
</table>
| customerRating   | enumerable    | *1 of 5 (low), 4 of 5 (high), ...*   |}

Dusek et al 2018
How was data gathered

- Crowd sourcing on CrowdFlower
- Experiment with 2 kinds of prompts
  - List of randomly ordered attributes
  - Pictorial representations
- Payment: .02/page containing 1 MR, 20 seconds/hit

Picture: Serving low-cost Japanese style cuisine, Loch Fynne caters for everyone, including families with small children.


Training Examples

• Crowd sourced ~50K instances
  • 6K MRs
  • 5 slots/MR
  • Largest dataset of its kind
    • Sfrest: 5K instances, 1K MR
    • Bagel: 404 instances, 380 MR

• Average of 8.27 references per MR
Delexicalization

• **MR:** name[Green Man], food[French], priceRange[more than 30 pounds], area[city centre], familyFriendly[no], near[All Bar One]

• **Lex:** *Green Man* is a French restaurant in the city centre. It is not child friendly and is located near *All Bar One*. It costs more than thirty pounds.

• **Delex:** *X-name* is a french restaurant in the city centre. It is not child friendly and is located near *X-near*. It costs more than thirty pounds.
Traditional language generation

• Content selection
  • (Done for us in the E2E task)

• Aggregation
  • Which pieces of content go into which sentence?

• Realization
  • How does a tree get realized in English?
The baseline system

• Two step generation
  • Sentence planning and surface realization are separated

• Joint one-step approach
  • Directly produces a natural language string
inform(name=X-name,type=placetoeat,eattype=restaurant, area=riverside,food=Italian)

X is an Italian restaurant near the river.

Figure 1: Example DA (top) with the corresponding deep syntax tree (middle) and natural language string (bottom)
Seq2seq model

- Encoder decoder RNN (Cho et al 2014, Sutskever et al 2014)

- Need to convert input dialog act (DA) and output tree into sequences
DA: Sequence representation

- Triple: DA type, slot, value
- Concatenate triples all slots
- Each token is an embedding
Syntax trees as sequences

• \(<\text{root}> \ <\text{root}> \ ((X\text{-name } n:\text{subj}) \text{ be } v:\text{fin} \ ((\text{Italian } adj:\text{attr}) \text{ restaurant } n:\text{obj} \ (\text{river } n:\text{near+X})))\)
Seq2seq model

- **Encoder**
  - \( X = \{x_1, x_2, \ldots, x_n\} \)
  - RNN to encode into a sequence of encoder output/hidden states \( h = \{h_1, h_2, \ldots, h_n\} \)
    - Where \( h_t = \text{lstm}(x_t, h_{t-1}) \)

Dusek and Jurcicek 2016
Seq2seq model

- Decoder
  - Output $y = \{y_1, y_2, \ldots, y_n\}$
  - $P(y_t | y_1, \ldots, y_{t-1}, x) = \text{softmax}((s_t \circ c_t) W_y)$
  - $S_t$ is the decoder state
    - $S_0 = h_n$
    - $S_t = \text{Istn}(((y_{t-1} \circ c_t)W_s, s_{t-1})$

Dusek and Jurcicek 2016
Beam search in this context

• Last time:

Filter k-max

• $\pi \ (i+1) \leftarrow \text{K-argmax} \ g(y_{i+1}, y_c, x) + s(y, x)$

• What was $y_c$?
• What would we use here in place of $y_c$?
Beam search and re-ranker

- Most common errors (semantic errors)
  - Missing an attribute
  - Added an attribute (hallucination)
  - Wrong value for an attribute
- Re-ranker scores n-best output by penalizing those that added or missed an attribute
  - Vector of realized attributes compared to vector of input attributes
  - Hamming distance is the penalty
- Learn a classifier for each output over the scores
  - Sigmoid ($h_n \cdot W_R + b$)
Computing Hamming distance

Figure 4: The reranker
Beam search on your homework

• Suggest using log likelihood normalized by length

• If you’d like to do something more sophisticated, such as this, can earn you extra credit
<table>
<thead>
<tr>
<th>Setup</th>
<th>BLEU</th>
<th>NIST</th>
<th>ERR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mairesse et al. (2010)*</td>
<td>~67</td>
<td>-</td>
<td>0</td>
</tr>
<tr>
<td>Dušek and Jurčíček (2015)</td>
<td>59.89</td>
<td>5.231</td>
<td>30</td>
</tr>
<tr>
<td>Greedy with trees</td>
<td>55.29</td>
<td>5.144</td>
<td>20</td>
</tr>
<tr>
<td>+ Beam search (b. size 100)</td>
<td>58.59</td>
<td>5.293</td>
<td>28</td>
</tr>
<tr>
<td>+ Reranker (beam size 5)</td>
<td>60.77</td>
<td>5.487</td>
<td>24</td>
</tr>
<tr>
<td>(beam size 10)</td>
<td>60.93</td>
<td>5.510</td>
<td>25</td>
</tr>
<tr>
<td>(beam size 100)</td>
<td>60.44</td>
<td>5.514</td>
<td>19</td>
</tr>
<tr>
<td>Greedy into strings</td>
<td>52.54</td>
<td>5.052</td>
<td>37</td>
</tr>
<tr>
<td>+ Beam search (b. size 100)</td>
<td>55.84</td>
<td>5.228</td>
<td>32</td>
</tr>
<tr>
<td>+ Reranker (beam size 5)</td>
<td>61.18</td>
<td>5.507</td>
<td>27</td>
</tr>
<tr>
<td>(beam size 10)</td>
<td>62.40</td>
<td>5.614</td>
<td>21</td>
</tr>
<tr>
<td>(beam size 100)</td>
<td>62.76</td>
<td>5.669</td>
<td>19</td>
</tr>
</tbody>
</table>

Table 1: Results on the BAGEL data set

Dusek and Jurcicek 2016
E2E 2018 Challenge Take-aways

• Seq2seq score high on automatic metrics and human evaluations of naturalness
  • Other approaches: statistical/ML, template filling (learned and manual)

• But seq2seq often fail to correctly express a meaning representation

• Seq2seq can be outperformed by hand-engineered
  • On overall quality, complexity, length and diversity of output