Automatically Generating ProFootballTalk Comments

Andrew Burkard
amb497@georgetown.edu

December 13, 2017

Abstract

In this paper I evaluate a few different techniques to generate a comment given an article from the website ProFootballTalk. A Markov chain and a variant of a Sequence-to-Sequence model with attention are compared to see how well they do at creating a realistic-sounding comment that is relevant to a given article. The Markov Chain generates interesting, if unrealistic results, while the neural network shows initial promise, but incomplete results.

1 Introduction

Automatically generating sentences that are syntactically and semantically correct, but also varied and realistic is one of the preeminent challenges in natural language processing. An even more difficult challenge is to create sentences that are relevant given a dynamic prompt. In this paper I evaluate a few different techniques to generate a comment given an article from the website ProFootballTalk. This corpus was chosen because the articles and even more so the commenters have a particular writing style that is characterized by "didactic misspelling, erratic punctuation, conspiratorial anxiety, and arrogant disdain for critical thought" [2].

2 Data

I collected the data from the `profootballtalk.nbcsports.com` RSS feed. This allowed me to gather 150,354 articles links and 2,770,700 associated comments from Oct 24, 2007 to Sep 22, 2017. Because of limitations in the RSS feed, I was only able to pull up to 30 comments per article. The comments contained very clean text with little to no markup. I then scraped the raw HTML for each article and passed it through a handy article extraction library [3]. This gave me an equally clean article dataset. Both articles and comments were passed through the English language tokenizer provided by the Spacy [1] package.

3 Methods

3.1 Markov Chain

The first method I attempted is also the simplest. In this model, a Markov chain is constructed from the corpus of comments where each state is a two word sequence w_i, w_{i+1}. To generate the next word in a sequence, we simply sample w_{i+2} from $P(w_{i+2}|w_i, w_{i+1})$, where the probability distribution is that of the corpus. The start and end of a comment are represented by special dummy tokens. The strengths of this model are that it is easy to construct, and it generates whole words, so its spelling ability meets that of the underlying corpus. The greatest weakness is that it doesn't learn long term dependencies as it only ever deals with sequences of three words at a time. Also, it is trained only on the comments corpus, so outputs cannot be generated given an article prompt.

3.2 Pointer Generator Model

Given the limitations of Markov approach, I went looking for a more sophisticated approach that was aware of long term dependencies and could respond differently to different inputs. I became

aware that my problem is really a special case of text summarization, where in place of summaries, we generate comments. This led me to consider a sequence-to-sequence model with attention, which has become state-of-the-art for neural summarization . However, a key potential weakness of the sequence-to-sequence model is that is purely abstractive. That is, the summaries it generates don't have to appear in the original article [4]. While this is often an advantage for the model, there are times, especially when creating comments, where we'd like to reference words in the original article. Fortunately for us, this is also an area of active research.

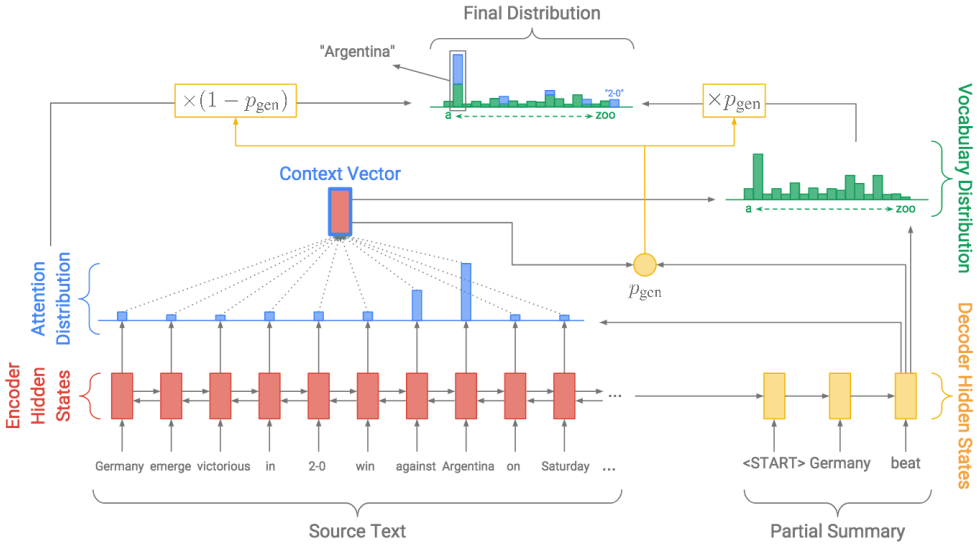

Figure 1: Pointer-Generator Model Architecture [5]

A "pointer-generator network" (See, Liu, & Manning, 2017) is a variant of the sequence-to-sequence model with attention that attempts to augment abstractive summarization with features extracted directly from the text. It does this by adding a layer in the decoder that calculates the probability of generating a new word or retrieving one from the text's attention distribution at every time step [5]. See Figure 1 from the author's original paper for a diagram of the network architecture. For the results in this paper, the hidden layer was trained with 128 nodes, input sequences were limited to 400 words, and output sequences were limited to 100 words. Additionally, the vocabulary was restricted to the top 50,000 words in the joint articles and comments corpus.

4 Results

The Markov chain produces comments that look like English sentences if the reader merely skims over them. However, upon closer inspection they clearly show their limitations. Often, they will form either long run-on sentences or brief clauses and phrases. Additionally, most comments are rather incoherent, although part of that may be due to the dataset. While the model does not link it's output to a prompt article, we can mimic that behavior somewhat by generating a large number of comments and filtering them by keywords.

canetic says : Mar 2 , 2010 12:00 PM who do the steelers have to say it , they get to the playoffs , while on another team either . Sad .
Hutch is out for calling someone else to follow his great - great WR who is willing to take the starting role to Aldon and do n't think a couple hundred people get money out of his time as Smith , Willis Mcgahee is still America . This deep into the street is a grown man and deserves to have something to him as the " inadvertent whistles
I have on your blog . :) At some point in Indy . Oh ? What has he made that comment above to the NFC East champion this year and that is the Steelers with do n't go on ... they ca n't separate from defenders " surrender the picks for a first round picks should not be just fine and a college game , regardless of how it works out for his 4th year .

Table 1: Example Markov Chain Comments.

Unfortunately, the Pointer-Generator did not converge before I ran out of time and resources as you can see from the training loss in Figure 2. Whether it would have done so with more epochs is an open question. However, we can at least see some patterns in the outputs of the latest checkpoints.

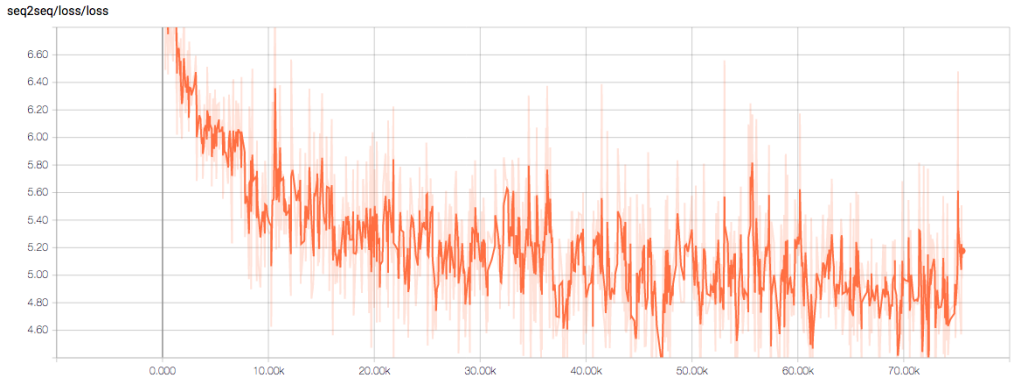

Figure 2: Training Loss for Pointer-Generator Model

The Pointer-Generator network clearly picked up on a common pattern in the comments. Because ProFootballTalk contains no built-in reply feature, commenters will often copy the username and date string at the top of another user's comment to reply to them. Unfortunately, the network has noticed this and overfit itself to it. The result is many repetitive comments with different usernames and dates. Furthermore, the usernames are usually replaced with an "[UNK]" token, as the original username tokens are almost never common enough to be in the top 50,000 most common words in the corpus. It remains to be seen whether more training epochs would resolve this issue. It should be said that when the network goes into pointing mode rather than generation mode it produces some decent sounding, if simplistic results, such as the middle comment referencing the Bears.

[UNK] says : Mar 28 , 2016 6:09 PM [UNK] says : Mar 28 , 2016 6:09 PM [UNK] says : Feb 12 , 2016 6:09 PM
[UNK] says : Jun 29 , 2011 8:37 PM [UNK] says : Jun 29 , 2011 8:37 PM [UNK] says : Jun 29 , 2011 8:37 PM I 'm not a fan of the Bears , but I do n't understand why the Bears are going to be a good team . I 'm not a fan of the Bears , but I do n't understand why the Bears are in the playoffs . _
[UNK] says : Oct 29 , 2016 AM The Jerry is the only one of the NFL . I know it is n't a fan . I 'm not sure you know it . I do n't know it . I do n't know it .

Table 2: Example Pointer-Generator Comments.

5 Conclusion

A Markov chain does a reasonably good job of approximating a ProFootballTalk comment, as long as the reader doesn't try to make sense of it. The neural pointer-generator network shows some promise when referring back to the original prompt article, but needs either some engineering enhancements or adjustments to the input data.

References

[1] Spacy: Industrial-strength natural language processing. https://spacy.io/.

[2] Jim Lohmar. Grit, grammar and road-grading: A conversation with pft commenter, 2014.

[3] Lucas Ou-Yang. Newspaper: Article scraping and curation. https://newspaper.readthedocs.io/en/latest/.

[4] Alexander M Rush, Sumit Chopra, and Jason Weston. A neural attention model for abstractive sentence summarization. *arXiv preprint arXiv:1509.00685*, 2015.

[5] Abigail See, Peter J Liu, and Christopher D Manning. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*, 2017.

A Appendix

I am responsible ... go to a remedial English reading class might be a gimmick , it 's all I am extremely proud .
This dude can ball ; gunslinger mentality . If you doubt it . They will do good on Sunday 's game . Most people who hate anyone during a first - time to learn from him
" Prohibition of Cannabis is actually burgandy and gold . I can see the Super-bowl " Sounds more like his . But if it was n't balanced officiating and that 's too funny .... RIP , Killer . Despite most fans of teams that benefited from , and the participants except LT and give them their riches . On February 12 , 2011 10:52 PM Has Manning ever would have either ; NFLPA on board to refer to him now , so no hate for T.O. I swear American workers enjoy on their street clothes

Table 3: Additional Markov Chain Comments.

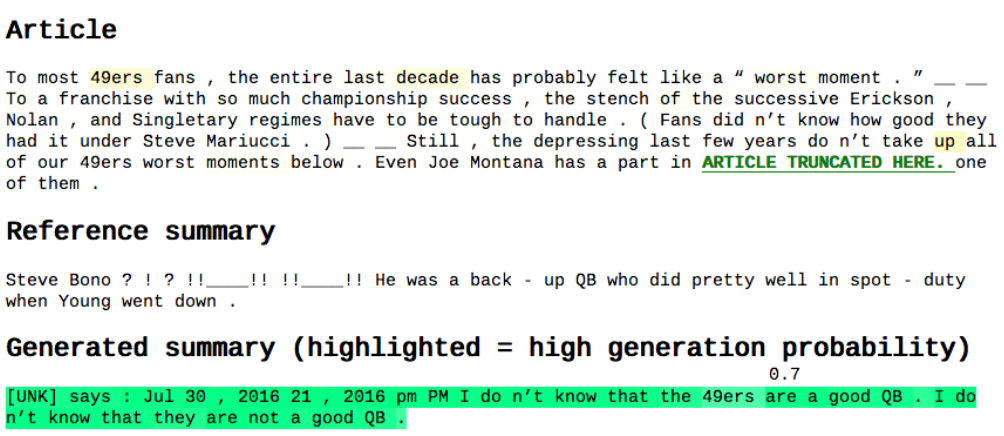

Figure 3: Attention Mechanism for Pointer Generator Model

[UNK] says : Jun 29 , 2011 8:37 PM I would rather have a lot of respect for him . He 's not going to play in the NFL . He 's not going to play in the NFL . He 's not going to play in the NFL . He is a good player . He is a good player . He is a good player . He 's a good player . He 's a good player . He 's a good player . He 's a good player .
[UNK] says : Sep 27 , 2011 AM [UNK] , I do n't know the Cowboys . I 'm sure that the Cowboys do n't have a lot of a couple - 8 - 8 .
[UNK] says : Jun 30 , 2011 8:37 PM [UNK] says : Jun 29 , 2011 8:37 PM I 'm not a fan of a fan of the NFL , but I do n't understand why the NFL is not going to be in the NFL . " I do n't know why the NFL is not going to be in the NFL .

Table 4: Additional Pointer Generator Comments.

Query	Results
broncos - denver + chicago =	bears 0.6081588268280029 packers 0.4379752278327942 lions 0.4198874235153198
tomlin - steelers + patriots =	belichick 0.610906720161438 reiss 0.5306036472320557 westhoff 0.5048067569732666
brees - saints + ravens =	flacco 0.5535164475440979 suggs 0.4450802803039551 raven 0.42625176906585693
peyton - good + mediocre =	eli 0.49270105361938477 danieal 0.4752071499824524 archie 0.4269319772720337
gronkowski + murderer =	hernandez 0.6410332322120667 gronk 0.5880988836288452 ninkovich 0.5786757469177246

Table 5: Word2Vec Analogies.