Natural Language Generation of *The Office* Using a Neural Machine Translation Context

Winton Yee, winton@stanford.edu **Samuel Kwong, samkwong@stanford.edu**

Abstract

The shortcomings of many NLG techniques are especially noticeable in film dialogue scripts. In a script, each line of dialogue is made by a particular character, with the next line of dialogue often being a response to that speaker from another speaker. NLG models such as LSTM-RNNs fail to capture flow of exchange. To address this challenge, we propose an NLG model using a Neural Machine Translation context.

Data

Our dataset contains every line from every episode of the U.S. television show, *The* Office, a total of 59,909 lines. Each entry contains dialogue, the speaker and stage directions.

```
Dwight: Hey, we nailed the sale!
Michael: Where were you this morning?
Dwight: I overslept. Damn rooster didn't crow.
Michael: Why do you lie, liar?
```

Approach

Common pitfalls of using common NLG techniques such as LSTM-RNNs to produce dialogues are that speakers do not speak in an orderly fashion (i.e. no back-and-forth exchange) and what each speaker says rarely is a believable response to what has been said before, as shown below:

```
Oscar: Can I make you a day off between us? [Michael sets over]
Andy: I'm gonna have some attention. They good? One, two, he Halpert is that stuff, and it's been good enough to be earbud] Plus I have a couple of little meet, a few office.
Michael: Toby sounds good.
Pam: [leaving the cigarette approaches from Michael's desk] Hey!
```

Although text is effectively generated, it has little coherence to form an exchange of dialogue, let alone a plot to the script.

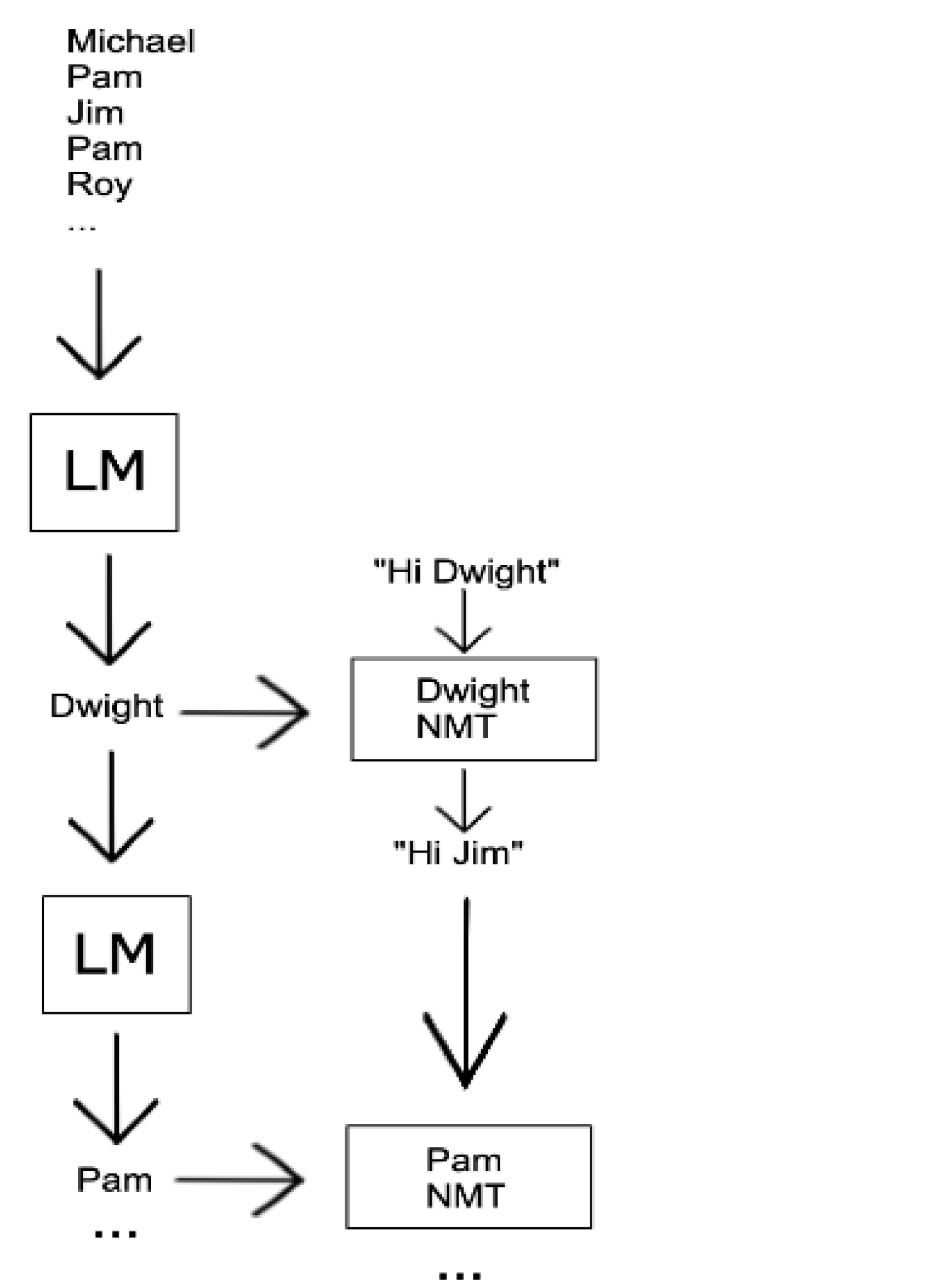
To combat this problem, we first enlist a language model to decide who is the next speaker for each line of our generated dialogue script. Then, we use a the corresponding NMT for each character to produce their line of output.

Each **source sentence** is the previous speaker's line of dialogue, while the **target sentence** is the line of dialogue the current speaker has said in the dataset. Thus, when generating output, each model's generation of text is dependent on what the previous speaker said.

Overall, our architecture combines multiple steps to produce an end-to-end trained model that takes into account the flow of exchange in dialogue.

1) a word-based RNN language model that generates who is the next speaker for each line of dialogue, treating each speaker's name as an entry in our vocabulary
2) twenty-five custom NMT models trained specifically for each of the twenty-five characters in our dataset, for their style of response

Model



Results & Analysis

Below are two sample texts:

Text1

```
Jim: I think you should have a second?
Ryan: Did you get a drink of a thing.
Pam: I don't think you should go it.
Michael: Excellent!
Pam: I mean, I don't think we should have a second?
Michael: Mmm.
Pam: You don't have cash.
Michael: No, no. You know what, that is that?
Jim: No.
Michael: Definitely.
```

Text 2

```
Dwight: I don't think you can do this to you down.
Jim: I'm going to you?
Dwight: I can do my aphrodisiac.
Jim: I think you should have to go back to her.
Dwight: I don't want to be standing to work?
Jim: I'm not gonna do it.
Dwight: I can get it. You know what? You just want to be a bad or time you ever learned off how you were private salivating that Hold a witness.
Jim: I'm going to coffee.
```

We used human evaluators to measure the quality of our generated text, compared to baselines and actual scripts. We asked them to rate texts on coherence and humor, on a scale of 1 to 5. Scores are shown below.

Model	Coherence	Humor
Char-based	1.31	1.27
Word-based	1.23	1.15
NMT-language	3.27	3.08
Actual scenes	5.00	4.12

We found our model succeeded in simulating back-and-forth conversation due to the word-based language model. We were also able to simulate the long-range nature of a conversation. For example, the first text references a "second drink" multiple lines apart. This may be due to the fact the "same" sentence is translated multiple times. Additionally, questions are often followed by answers. This may be because "source" questions and translated to "target" answers.

References

- Kaparthy, Andrej. (2015). The Unreasonable Effectiveness of Recurrent Neural Networks
- Ehsan, U., Harrison, B., Chan, L., and Riedl, M. (2017). Rationalization: A Neural Machine Translation Approach to Generating Natural Language Explanations