Quantifying the Impact of Disfluency on Spoken Content Summarization

Maria Teleki, Xiangjue Dong, James Caverlee

Texas A&M University

Research Questions

- ***RQ1: How Do Disfluencies Impact Summarization Quality?*** We synthetically inject disfluency events (repeats, interjections, false starts, and their combinations) at a range of severity levels and measure their impact on summarization quality.
- ***RQ2: Can Summarization Quality be Improved By Directly Modeling Disfluency?*** We explore the use of a state-of-the-art disfluency detection model [2] to improve the summarization quality by either (1) removing the disflencies, or (2) tagging the disfluencies.

Original

Hello and welcome to our podcast! Let's get right to it. Today we're going to be interviewing a very special guest, someone I know you guys have been excited about having on the show.

Repeats with N=3

Hello and welcome to our podcast! Let's get get get get right to it. Today we're going to be interviewing a a a a very special guest, someone I know you guys have been excited about having on the show.

Interjections with N=3

Hello and welcome to our podcast! Let's get right uh okay okay to it. Today we're going to be interviewing a very special um so I mean guest, someone I know you guys have been excited about having on the show.

False Starts with N=3

Hello and welcome to our podcast! Let's get right to it. Today we're today we're today we're today we're going to be interviewing a very special guest, someone I know you guys have been excited about having on the show.

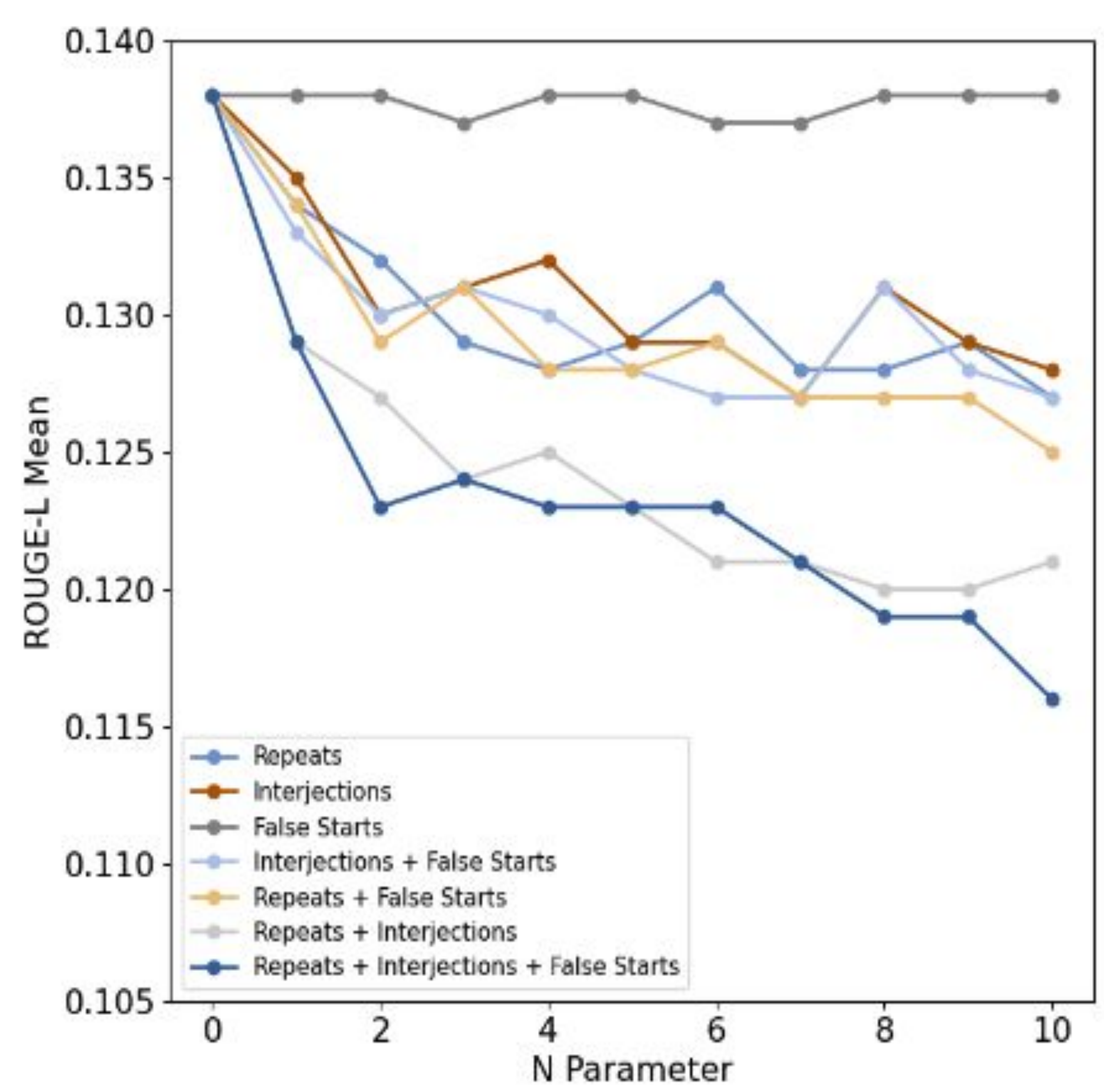
RQ1: Synthetic Disfluency Injection *(N)*

- We use 1,020 podcasts from the **Spotify Podcasts Dataset** [1] for our experiments for consistency with the 2020 TREC Podcasts Track summarization task [2].
- For **repeats and interjections**, we sample from *X~N (μ=10, σ=1)* to determine the position at which the term(s) should be injected into the transcript N times; interjections are uniformly randomly selected from: *uh, um, well, like, so, okay, I mean, you know.*
- For **false starts**, sentences >4 words are non-uniformly sampled with 80/20 probability with replacement, and the selected sentences have a false start (first 2 words of sentence) injected N times.
- We **vary N from 1 to 10 to isolate the impact of increased disfluency** and stress test the summarization systems.

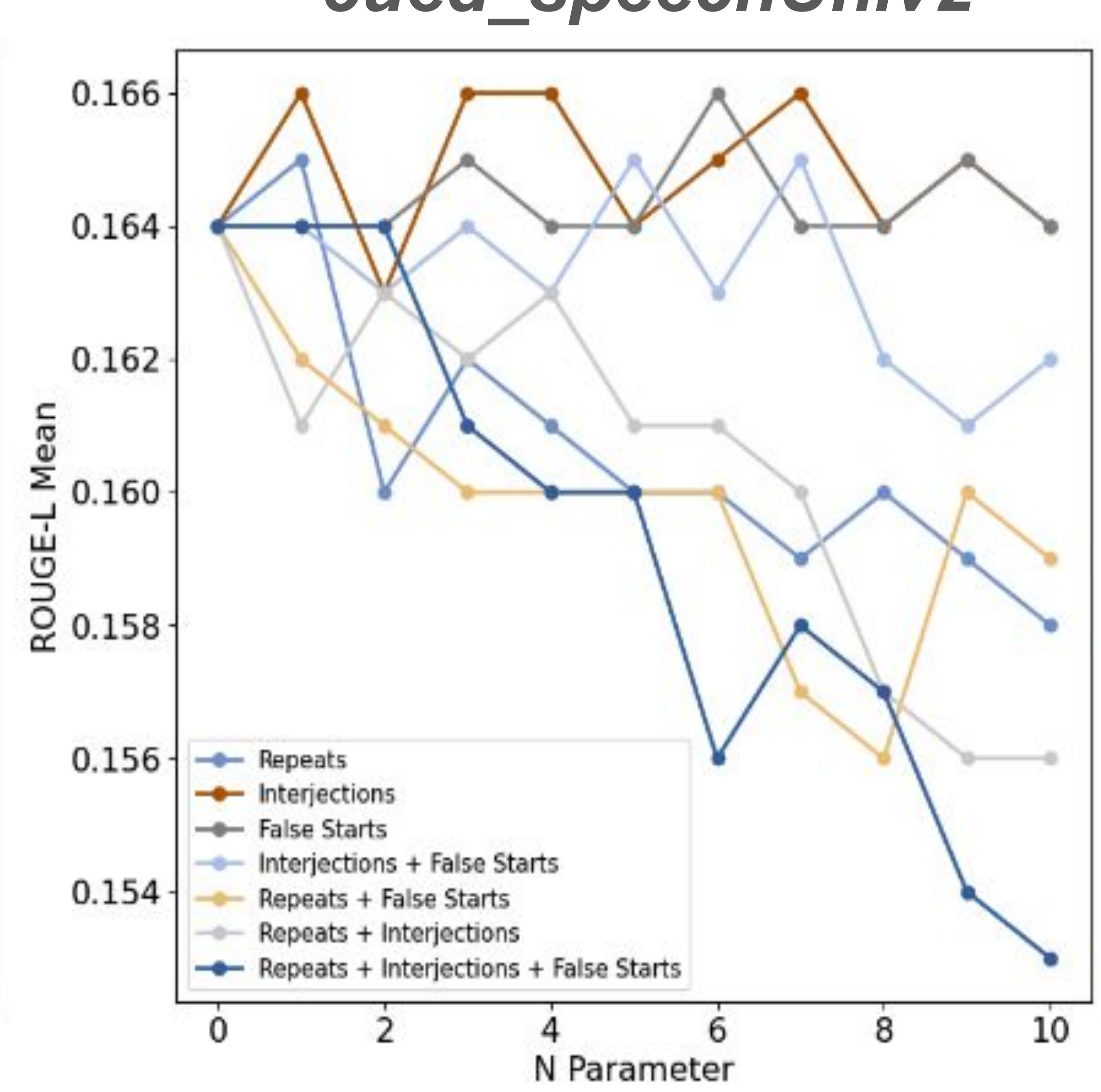
RQ1: Stress Testing Summarization Models *(N=0 to N=10)*

- We consider 6 models: 1min baseline, cued_speechUniv2, BART, T5, Pegasus, Llama 2-Chat.
- **Overall drop in ROUGE-L** with increased N.
- T5 and Pegasus are the least resilient, BART is moderately resilient, and cued_speechUniv2 and Llama 2-chat are the most resilient.

BART



cued_speechUniv2



RQ2: Repairing & Tagging Transcripts for Fine-Tuning *(N=2)*

- We use a ***disfluency annotation model*** [3] to label disfluencies.
- We then examine the impact of: (i) repairing the transcripts via disfluency removal, and (ii) tagging the disfluencies in the transcripts (<DIS>).

train	test	BART			T5			Pegasus		
		R-L	R-1	R-2	R-L	R-1	R-2	R-L	R-1	R-2
train$_R$	test$_R$	0.172	0.240	0.085	0.145	0.197	0.059	0.129	0.174	0.049
	test	**0.177**	**0.244**	**0.090**	0.146	0.196	0.060	**0.131**	0.177	**0.052**
	test$_T$	0.174	0.241	0.086	0.148	0.198	0.063	0.096	0.133	0.037
train	test$_R$	0.170	0.236	0.083	0.146	0.198	0.060	0.122	0.165	0.045
	test	0.175	0.242	0.088	**0.149**	**0.200**	0.062	0.126	0.169	0.049
	test$_T$	0.172	0.238	0.085	0.147	0.194	**0.065**	0.090	0.124	0.032
train$_T$	test$_R$	0.172	0.238	0.083	0.142	0.193	0.057	0.129	**0.193**	0.048
	test	0.173	0.240	0.085	0.143	0.194	0.057	0.127	**0.193**	0.047
	test$_T$	0.169	0.235	0.081	0.145	0.196	0.058	0.115	0.146	0.038

We find that training on the repaired transcripts (train$_R$) and testing on the original transcripts (test) yields the best results.

References

[1] Clifton, Ann and Reddy, Sravana and Yu, Yongze and Pappu, Aasish and others. 2020. 100,000 podcasts: A spoken English document corpus.

[2] Rosie Jones, Ben Carterette, Ann Clifton, Maria Eskevich, and others. 2020. TREC 2020 Podcasts Track Overview. In Text Retrieval Conference.

[3] Paria Jamshid Lou and Mark Johnson. 2020. Improving disfluency detection by self-training a self-attentive model. In Association for Computational Linguistics, pages 3754–3763.

Texas A&M University

Department of Computer Science & Engineering

Code *M. Teleki*