How to automatically generate exercises of different target difficulty from a single text?

![Static approach vs. Proposed approach]

- Language learning requires a lot of effort and motivation
- Personalized exercises can help keeping learners motivated
- For example, using their favorite book as a basis for exercises

1. Create standard C-Test
2. Assess difficulty
3. Manipulate C-Test
4. Go to 2. if not
   - a) Reached target difficulty $\tau$
   - b) No manipulation possible
5. Return resulting C-Test

C-Tests are fill-the-gap exercises where the second half of a word is turned into a gap for every second word in a text. To provide some contextual information, the first and the last sentence of a text do not contain any gaps. Due to the first half remaining as a hint, C-Tests have less ambiguity but still require orthographic, morphological, syntactic, and semantic competencies.

**C-Test Difficulty Prediction**

- Reproduction study of the work done by Lisa Beinborn (2016)
- Seemingly small changes may lead to different results (e.g., using a newer system dictionary)
- Achieved similar performance as the original system

**Overall Architecture**

1. Motivation
2. Overall Architecture
3. C-Test Difficulty Prediction
4. C-Test Difficulty Manipulation
5. Evaluation of Achievable Target Difficulty
6. User Study for C-Tests of Different Target Difficulties
7. Conclusion

**Motivation**

**Overall Architecture**

1. Create standard C-Test
2. Assess difficulty
3. Manipulate C-Test
4. Go to 2. if not
   - a) Reached target difficulty $\tau$
   - b) No manipulation possible
5. Return resulting C-Test

**C-Test Difficulty Prediction**

- Sample four texts $(T_1, T_2, T_3, T_4)$ of medium difficulty from the Brown corpus
- Use $T_1$ as the reference C-Test (same for all participants)
- For $(T_2, T_3, T_4)$, create an easy $(\tau = 0.1)$, hard $(\tau = 0.5)$, and default version with SEL and SIZE
- Two groups of 30 participants each solve either SEL or SIZE modified C-Tests
- Each participant solves four C-Tests and provides feedback on a five-point Likert-scale, their error-rate, and by ranking all C-Tests according to their perceived difficulty.

**Conclusion**

- Both manipulation strategies were able to create C-Tests of a target difficulty $\tau$ and were also perceived accordingly
- This allows us to create language learning exercises from a learner-preferred basis of texts to keep them motivated
- Work towards personalized learning process for different learners

**What are C-Tests?**

- Automatic evaluation on the Gutenberg, Reuters, and Brown corpus
- Assess influence of the underlying text for a target difficulty $\tau$
- Create maximally $(\tau_{\text{max}} = 1.0)$ and minimally $(\tau_{\text{min}} = 0.0)$ difficult C-Tests and estimate their difficulty using SEL and SIZE
- Most texts produce C-Tests with $\tau \in [0.0, 0.4]$ Error-rate ranges $\tau_{\text{max}} - \tau_{\text{min}}$ for different corpora