Manipulating the Difficulty of C-Tests

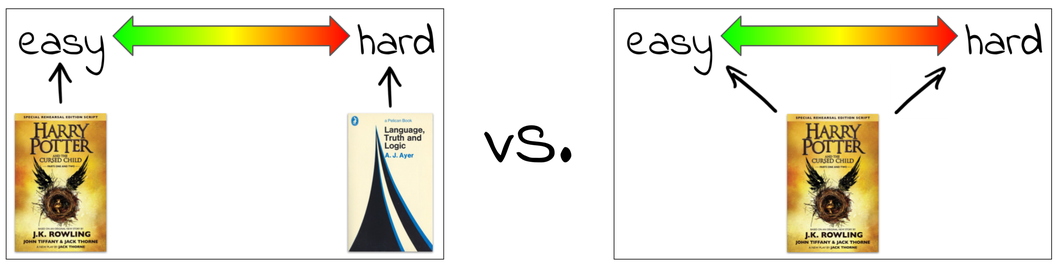
Ji-Ung Lee, Erik Schwan, Christian M. Meyer
Ubiquitous Knowledge Processing Lab (UKP)
Department of Computer Science, Technische Universität Darmstadt
https://www.ukp.tu-darmstadt.de

TECHNISCHE UNIVERSITÄT DARMSTADT

UKP

1 Motivation

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

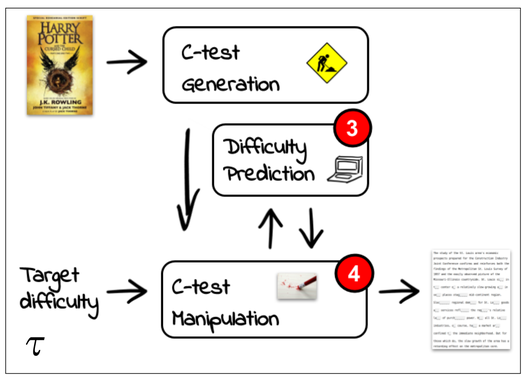


Static approach — Proposed approach

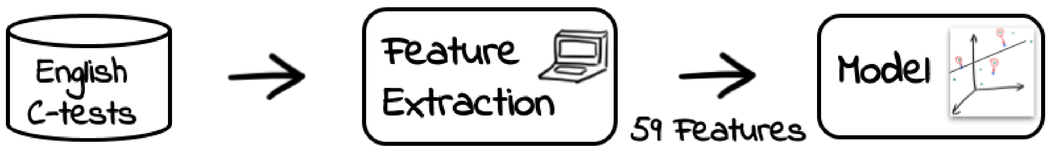
- Language learning requires a lot of effort and motivation
- Personalized exercises can help keeping learners motivated
- For example, using their favorite book as a basis for exercises

2 Overall Architecture

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



3 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

	Original data			New data		
Model	ρ	RMSE	qwκ	ρ	RMSE	qwκ
SVM (original)	.50	.23	.44	–	–	–
SVM (reproduced)	.49	.24	.47	.50	.21	.39
MLP	.42	.25	.31	.41	.22	.25
BiLSTM	.49	.24	.35	.39	.24	.27

4 C-Test Difficulty Manipulation

Algorithm 1 Gap size strategy (SIZE)

procedure IncreaseDifficulty(T, τ)
  $G_{\text{SIZE}} \leftarrow G_{\text{DEF}}$
  $D \leftarrow d(T)$
  while $D < \tau$ **do**
    $g^* = (i, \ell) \leftarrow \arg\max_{g \in G_{\text{SIZE}}} \Delta_{\text{inc}}(g)$
    $\ell \leftarrow \ell + 1$
    $D \leftarrow D + \Delta_{\text{inc}}(g)$
  return G_{SIZE}

Algorithm 2 Gap selection strategy (SEL)

procedure GapSelection(T, τ)
  $G_{\text{FULL}} \leftarrow \{(i, \lceil \frac{|w_i|}{2} \rceil) \mid 1 \le i \le 2n\}$
  $G_{\text{SEL}} \leftarrow \emptyset$
  while $|G_{\text{SEL}}| < n$ **do**
    $G_{\le\tau} \leftarrow \{g \in G_{\text{FULL}} \mid d(g) \le \tau\}$
    if $|G_{\le\tau}| > 0$ **then**
      $g^* \leftarrow \arg\min_{g \in G_{\le\tau}} |d(g) - \tau|$
      $G_{\text{SEL}} \leftarrow G_{\text{SEL}} \cup \{g^*\}$
      $G_{\text{FULL}} \leftarrow G_{\text{FULL}} \setminus \{g^*\}$
    $G_{>\tau} \leftarrow \{g \in G_{\text{FULL}} \mid d(g) > \tau\}$
    if $|G_{>\tau}| > 0$ **then**
      $g^* \leftarrow \arg\min_{g \in G_{>\tau}} |d(g) - \tau|$
      $G_{\text{SEL}} \leftarrow G_{\text{SEL}} \cup \{g^*\}$
      $G_{\text{FULL}} \leftarrow G_{\text{FULL}} \setminus \{g^*\}$
  return G_{SEL}

- SIZE: Modifying the gap size
  - Keep the initial gaps G_{DEF}
  - Only change the gap size ℓ
  - Increase (decrease) gap size for higher (lower) C-Test difficulty Δ_{inc} (Δ_{dec})
- SEL: Changing the gap selection
  - Create all possible gaps G_{FULL}
  - Select gaps closest to target difficulty

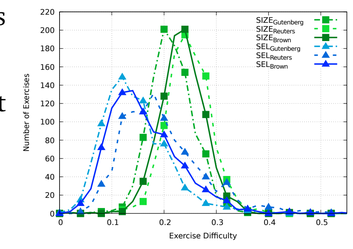
What are C-Tests?

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, morphologic, syntactic, and semantic competencies.

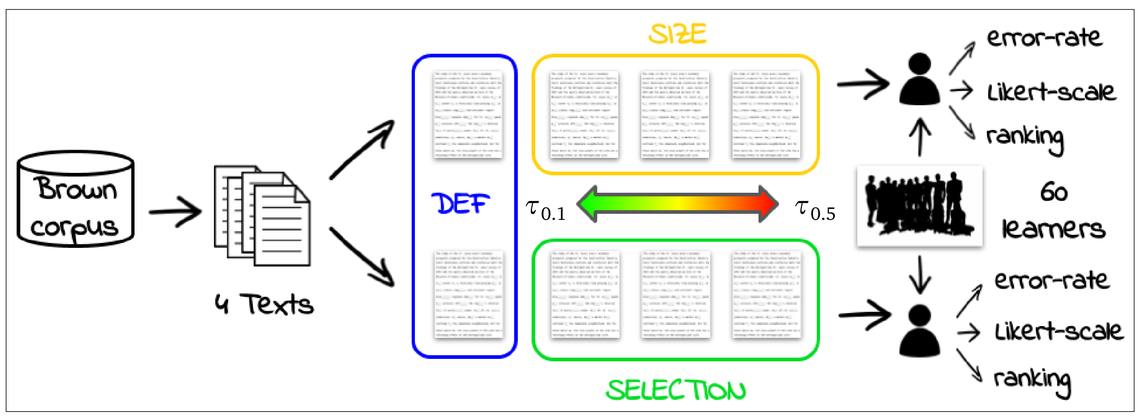
> St. Louis si__ in t__ center o_ a relatively slow-growing a__ in so__ places stag____ mid-continent region . Slac_____ regional dem___ for St. Lo___ goods a__ services refl____ the reg___'s relative la__ of purch_____ power. N__ all St. Lo___ industries, o_ course, ha__ a market ar__ confined t_ the immediate neighborhood.

5 Evaluation of Achievable Target Difficulty

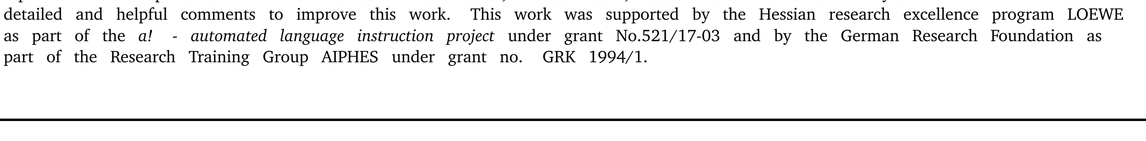
- Automatic evaluation on the Gutenberg, Reuters, and Brown corpus
- Assess influence of the underlying text for a target difficulty τ
- Create maximally ($\tau_{\max} = 1.0$) and minimally ($\tau_{\min} = 0.0$) difficult C-Tests and estimate their difficulty using SEL and SIZE
- Most texts produce C-Tests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



6 User Study for C-Tests of Different Target Difficulties



- 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.



a) Error-rate — b) Likert-scale — c) Ranking

7 Conclusion

- Both manipulation strategies were able to create C-Tests of a target difficulty τ 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

Code and Data

https://github.com/UKPLab/acl2019-ctest-difficulty-manipulation

We thank Lisa Beinborn and the language center of Technische Universität Darmstadt for providing us with the necessary data to reproduce their experiments. We also thank Claudia Schulz, Tristan Miller, Chris Stahlhut and the anonymous reviewers for their detailed and helpful comments to improve this work. This work was supported by the Hessian research excellence program LOEWE as part of the *a! - automated language instruction project* under grant No.521/17-03 and by the German Research Foundation as part of the Research Training Group AIPHES under grant no. GRK 1994/1.