The 27th International Conference
on Computational Linguistics (COLING 2018)

Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Par4Sim - Adaptive Paraphrasing for Text Simplification

Par4Sim - የሚጣጣም መልሶ መጻፍ፤ ፅሁፍን ለማቅለል

Seid Muhie Yimam
Chris Biemann

Language Technology Group
Department of Informatics,
MIN Faculty Universität Hamburg, Germany

Aug 20-26, 2018

Introduction

- Issues with annotation tools
  - Expensive
  - Requires expertise
  - Concept drift
  - Disconnected from the real-world applications

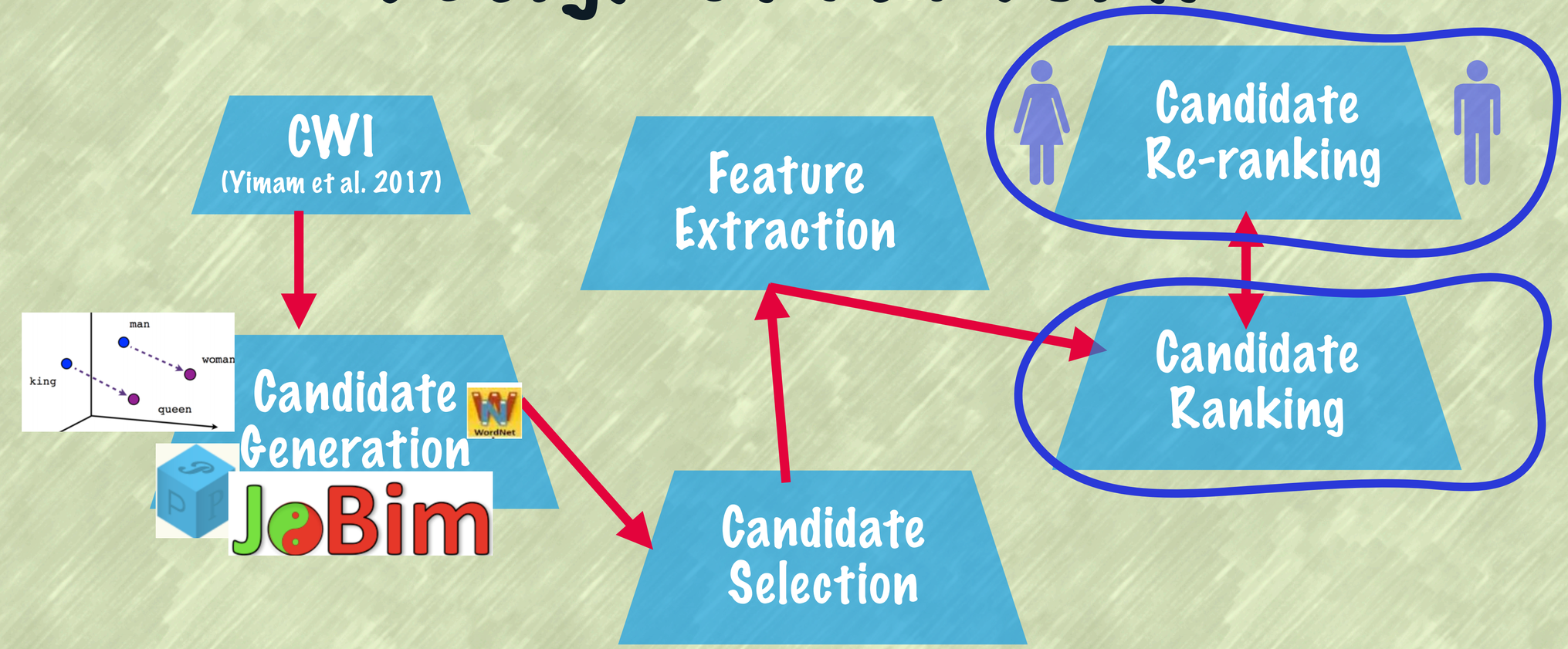
Introduction ...

Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

- Our approach:
  - Integrate ML model in applications
  - Adaptive and personalized
  - Iterative and interactive
  - Use-case — text simplification

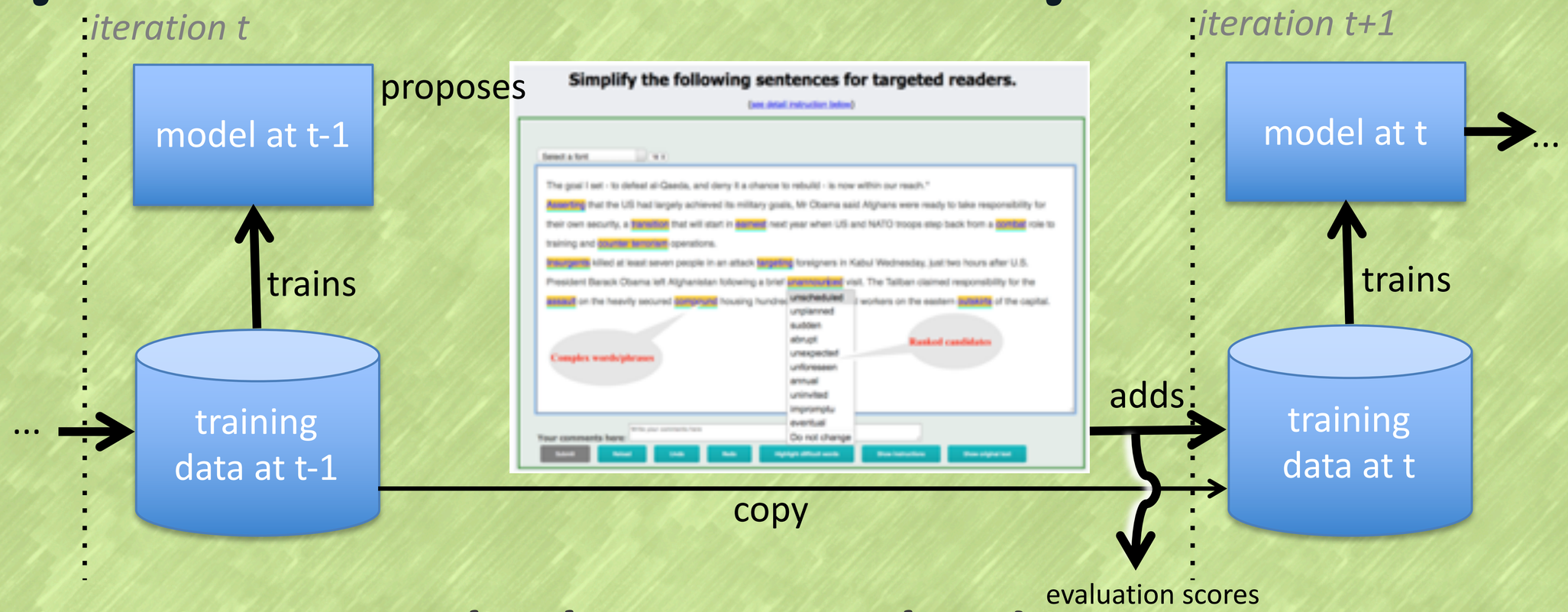
Design of Par4Sim

Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG



Seid Muhie Yimam, Sanja Štajner, Martin Riedl, and Chris Biemann. 2017. CWIG3G2 - Complex Word Identification Task across Three Text Genres and Two User Groups. IJCNLP-2017 Taipei, Taiwan.

Adaptive models for text simplification



* First iteration: use baseline language model ranking
* Training data is built in batches
* No overlap of data between iterations

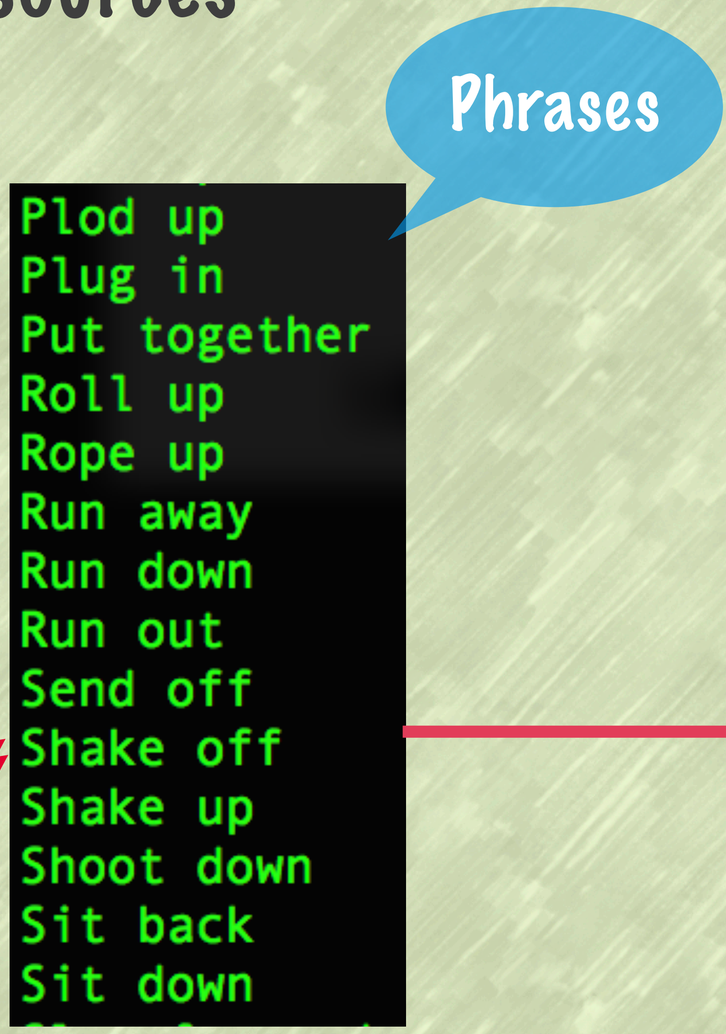
CWI component - (Yimam et al. 2017)

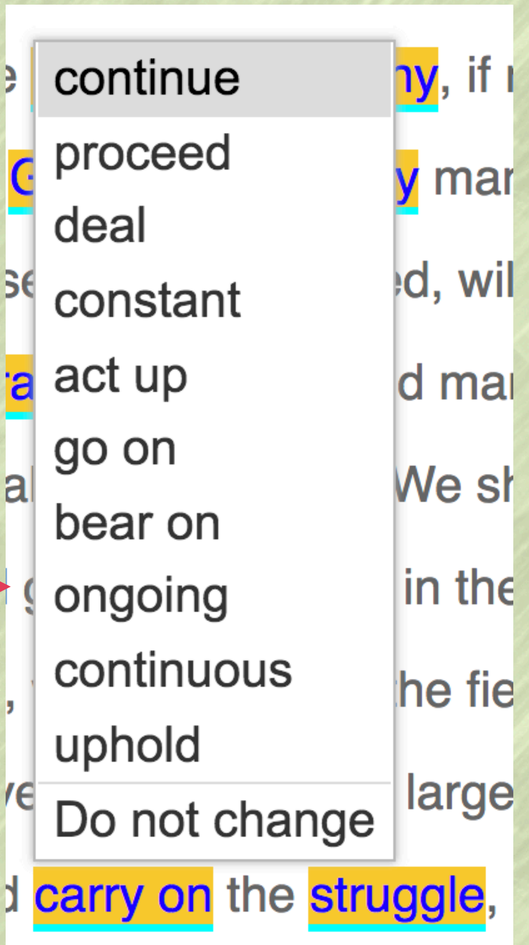
* ID1 Both China and the Philippines flexed their muscles on Wednesday.
  31 37 flexed 2 7 9
  * flexed is marked as complex phrase by 2 native and 7 non-native English speakers
* ID1 Both China and the Philippines flexed their muscles on Wednesday.
  31 51 flexed their muscles 4 2 6
  * flexed their muscles is marked by 4 native and 2 non native English speakers.

Candidate generation

- Lexical and Distributional resources
  - WordNet
  - DT (JoBimText)
- Paraphrase Database

  (PPDB) 2.0 and Simple PPDB
- Phrase2Vec





Feature extraction

* Frequency and length
* Lexical and distributional thesaurus resources
* PPDB 2.0 (http://paraphrase.org/) and simple PPDB:
* Word embeddings feature

Candidate ranking

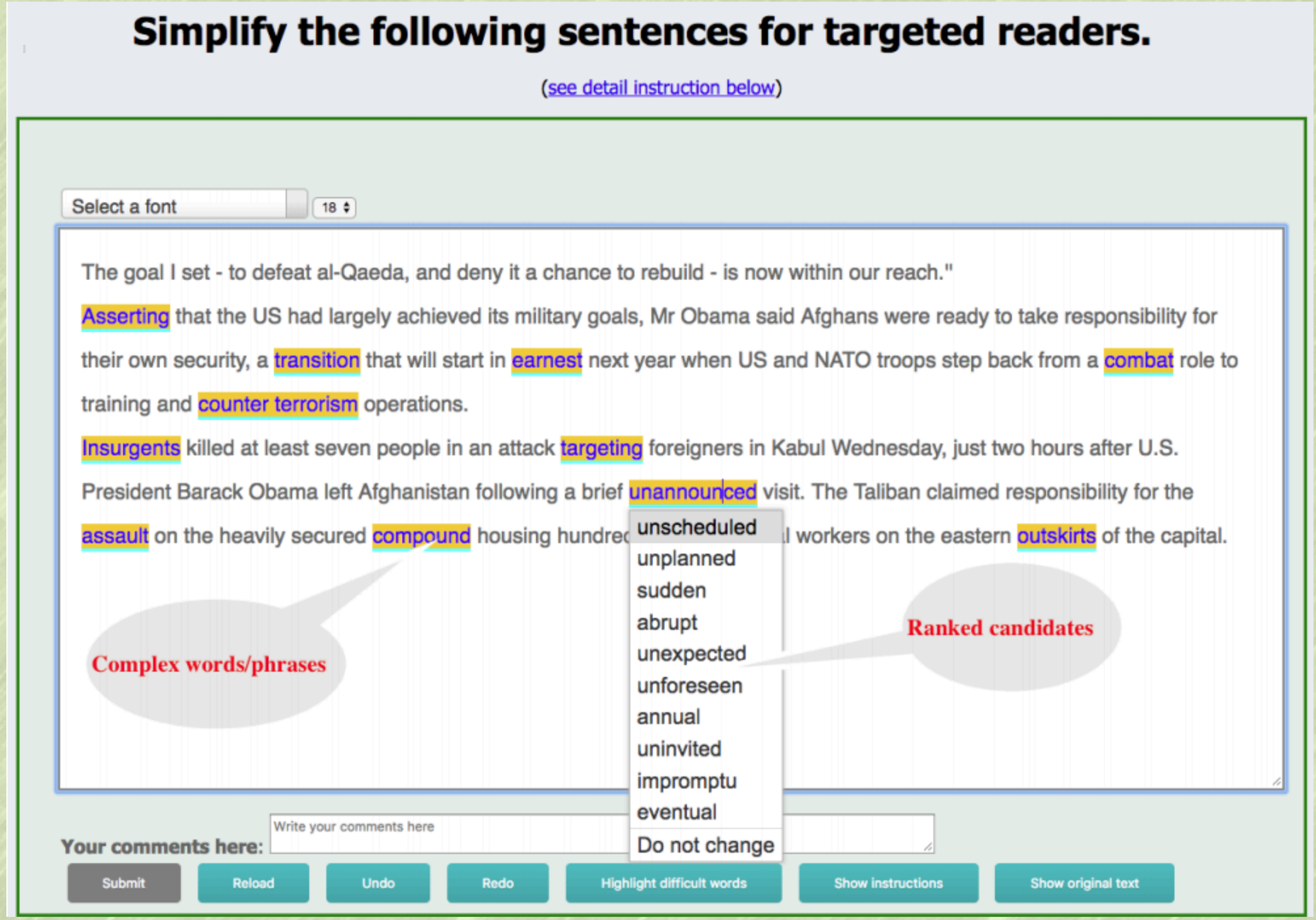
- LambdaMART from Ranklib [1]
  - Evaluation metrics — Normalized Discounted Cumulative Gain (NDCG)
  - Graded judgments — from the number of workers suggesting the candidate for the given CP target

[1] https://sourceforge.net/p/lemur/wiki/RankLib/

Crowdsourcing task setup

- Using Amazon Mechanical Turk (**MTurk**)
- With the external HIT of MTurk
  - Workers can login and access HITs from MTurk
  - Task runs on **our server**
    - **Log** every user actions (candidate selection, text highlighting, text input,...)
    - Generate **new model** once all HITs in a **batch** are completed

UI of Par4Sim



Instructions and operations for MTurk Workers

* Target of simplification: children, language learner, and people with reading impairments
* Reload text
* Undo and redo
* Highlight difficult words
* Show instruction / Original texts
* Show animation

Instructions of Par4Sim - MTurk

Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Instructions

In this HIT, you will see texts which contain 5-10 sentences. Your task is to make these texts **simpler** to understand for **children**, **language learners**, or **people with reading impairment**, as much as possible. You do so by replacing **difficult** or **complex** words and phrases by the simpler ones, which fit the context well and preserve the original meaning.

To make the simplification task easier, we provide you with built-in **suggestion system**. It helps you edit the text in the following way:

1. When you open the HIT, some words will be highlighted. Click the highlighted word and it will show you a list of words or phrases (possible **suggestions for replacing the original word or phrase**). When one of the suggestions is simpler (even if it is wrong in grammar or plural and singular forms), and fit the context well, please click on that suggestion. You can still correct the replaced word (for its form or tense, for example). Select **Do not change** if none of the suggestions seems to fit.

2. In case you find some words or phrases that are difficult to understand but are not yet highlighted, you can select them by double clicking on the word. If the system can provide you with a list of suggestions, the word/phrase you selected will become highlighted and you will see the list of suggestions for replacing it.

3. In case you do not like any of the suggested words/phrases, you can use the **back space key** to delete the original word/phrase and write your own suggestion.

How to work with the buttons:

- **Reload**: Get the original text again. This will remove all your changes.
- **Undo/Redo**: Undo or redo your changes
- **Highlight difficult words**: For the existing text, get suggestions for replacements from the system.
- **Show instructions / Show animation**: It shows you the detailed instructions or the animation.
- **Show original text / Hide original texts** : It shows/hides the original text to compare with your editing.

Start Experiment

If you have questions or comments about this task, please provide your comments or questions in the provided text field.

You will be able to submit the text after making enough changes and the **Submit** button is active (in blue background). Until then, the submit button will remain inactive (gray background). Having the **Submit** button active does not mean that your work is completed or your answer is accepted. It only shows that there is a reasonable progress in editing the text.

In case the text is already simple (in your opinion) butt the **Submit** button is not yet active, tell us in the comment text field so that the **Submit** button will be active.

Please **NEVER SELECT WRONG SUGGESTIONS** after clicking the highlighted words. If none of the suggestions is valid, click on **Do not change** and type your own substitute if you like.

Collected training instances

Highlighted based on the CWI dataset

Number of workers selected this candidate

Complex Sentence: *Hajar said his cousin was not* ***affiliated*** *with any terrorist group.*

Simplified Sentence 1: *Hajar said his cousin was not* ***associated*** *with any terrorist group.* → *6*

Simplified Sentence 2: *Hajar said his cousin was not* ***merged*** *with any terrorist group.* → *2*

Simplified Sentence 3: *Hajar said his cousin was not* ***aligned*** *with any terrorist group.* → *1*

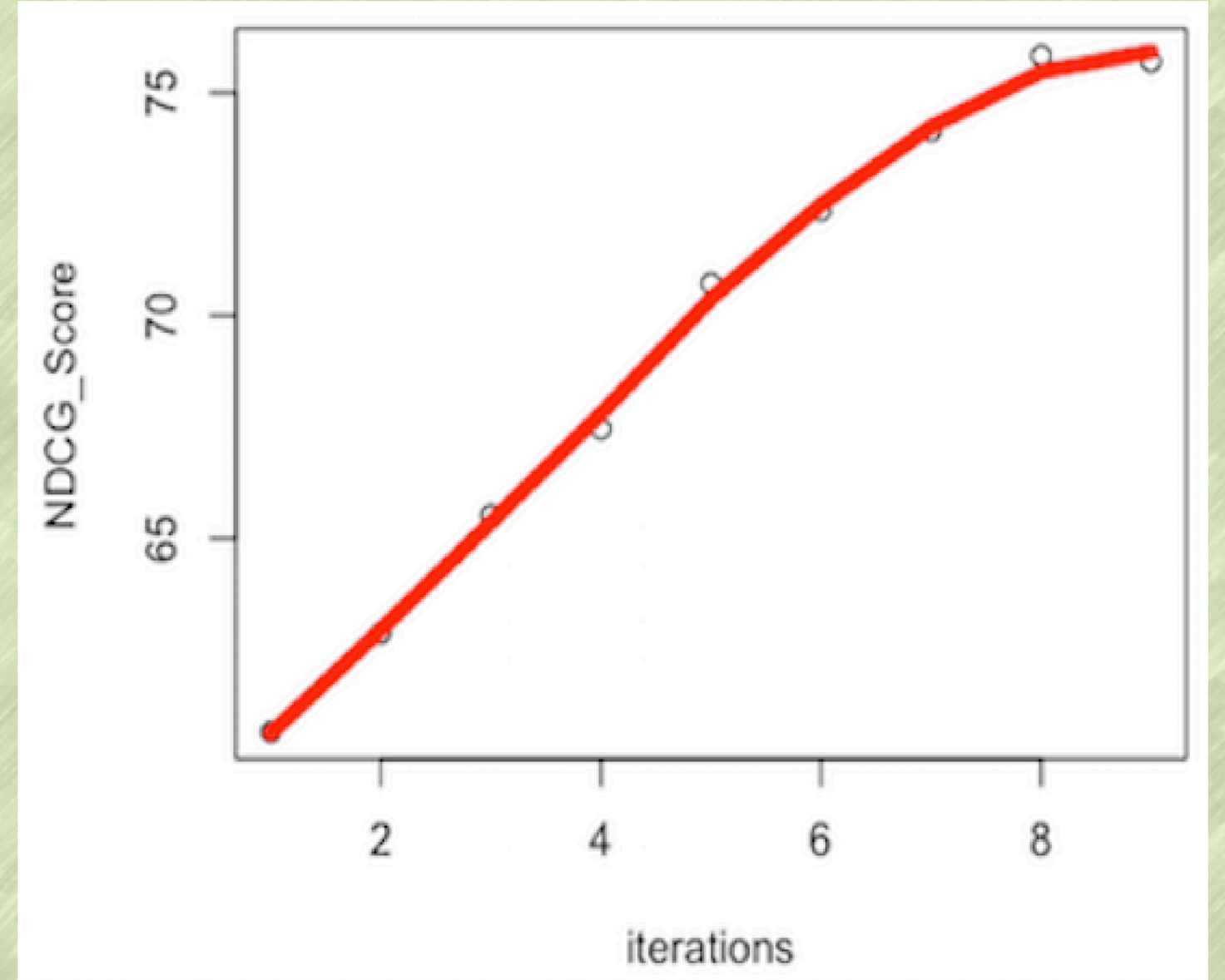
Simplified Sentence 4: *Hajar said his cousin was not* ***partnered*** *with any terrorist group.* → *1*

Experimental results

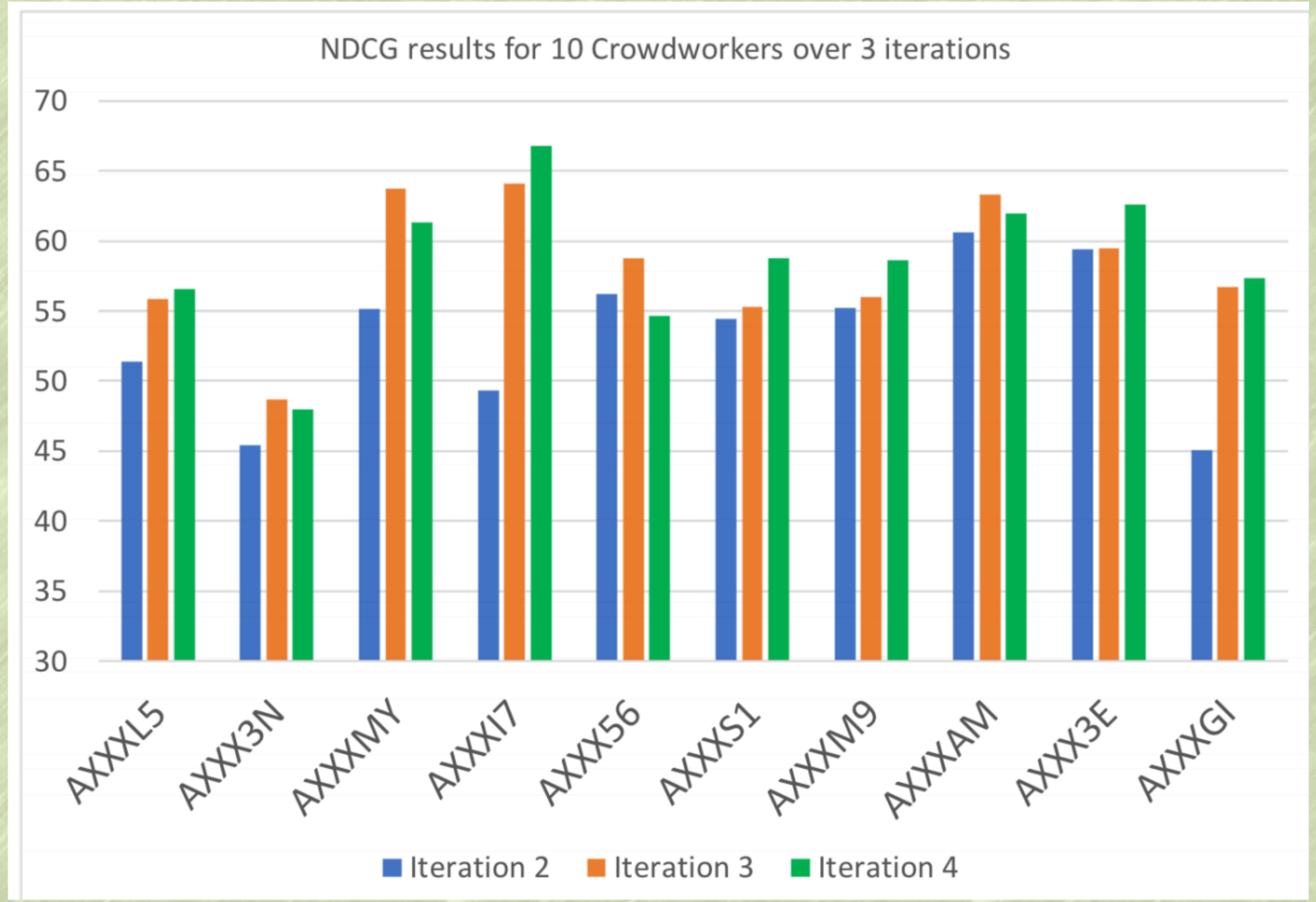
Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Testing	NDCG@10								
	Training instances on previous iterations								
	#sentences	baseline	1	≤ 2	≤3	≤4	≤5	≤6	≤7	≤8
1	115	-	-	-	-	-	-	-	-	-
2	214	60.66	62.88	-	-	-	-	-	-	-
3	207	61.05	63.39	65.52	-	-	-	-	-	-
4	210	58.21	60.73	65.93	67.46	-	-	-	-	-
5	233	56.10	62.53	65.66	66.00	70.72	-	-	-	-
6	215	62.18	61.05	66.51	67.86	69.88	72.36	-	-	-
7	213	57.00	62.07	64.02	64.88	67.28	69.27	74.14	-	-
8	195	56.56	59.53	62.11	63.03	64.54	67.40	71.05	75.83	-
9	224	56.14	63.48	65.58	65.87	69.18	69.51	71.31	71.40	75.70

Learning-curve



NDCG@10 over 3 iterations for 10 workers

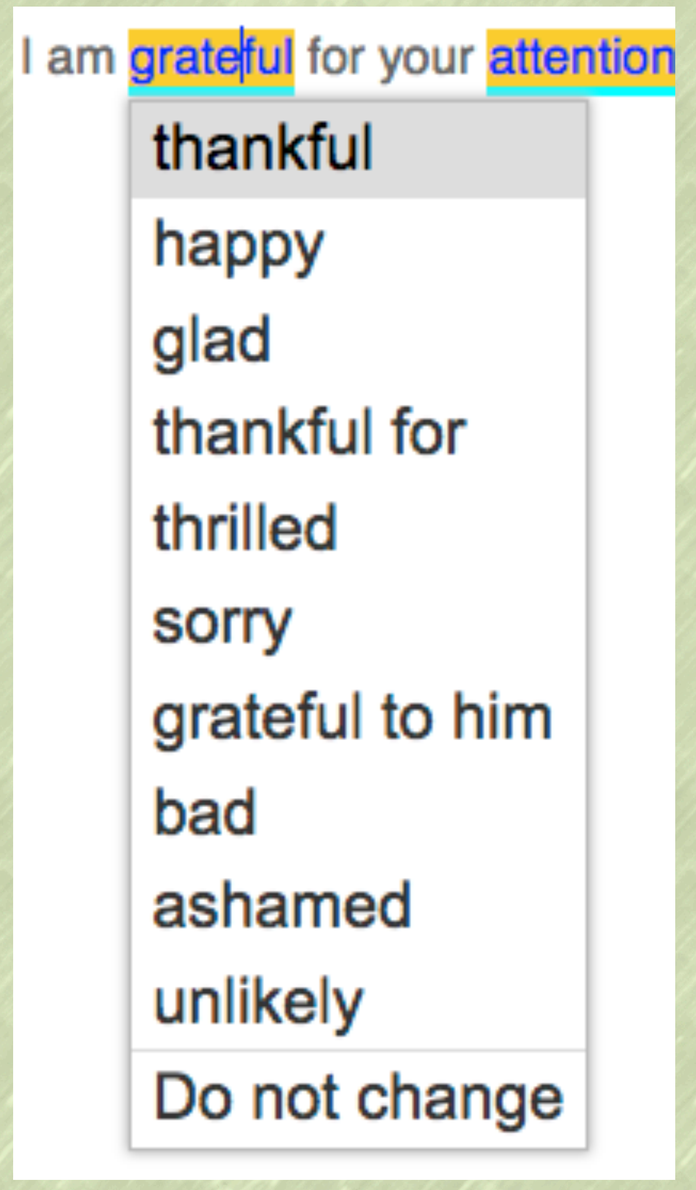


Discussion

- Traditional NLP models —> collect-train-evaluate
- Adaptive approaches —> a) integrated to applications b) personalized c) improve through usage (incremental)
- We show adaptive approaches using **MTurk**
  - For text simplification
- Par4Sim can be extended for **technical document writing**

Documentation, dataset, demo

PAR4SIM


I am grateful for your attention
thankful
happy
glad
thankful for
thrilled
sorry
grateful to him
bad
ashamed
unlikely
Do not change

LambdaMART

Algorithm: LambdaMART

Initialization

set number of trees N, number of training samples m, number of leaves per tree L, learning rate η
for $i = 0$ to m **do**
  $F_0(x_i) = \text{BaseModel}(x_i)$ //If BaseModel is empty, set $F_0(x_i) = 0$
end for

Boosted Tree Step

for $k = 1$ to N **do**
  for $i = 0$ to m **do**
    $y_i = \lambda_i$
    $w_i = \frac{\partial y_i}{\partial F_{k-1}(x_i)}$
  end for

Calculate lambda and weight

  $\{R_{lk}\}_{l=1}^{L}$ // Create L leaf tree on $\{x_i, y_i\}_{i=1}^{m}$

Create regression trees for lambda

  $\gamma_{lk} = \frac{\sum_{x_i \in R_{lk}} y_i}{\sum_{x_i \in R_{lk}} w_i}$ // Assign leaf values based on Newton step.
  $F_k(x_i) = F_{k-1}(x_i) + \eta \sum_l \gamma_{lk} I(x_i \in R_{lk})$ // Take step with learning rate η.

Calculate leaf node values and update the predicted score

end for

Evaluation — NDCG@10

- Ratings up to a specified rank position

$$Cumulative\,Gain\,at\,p = CG_p = \sum_{i=1}^{p} rating(i)$$

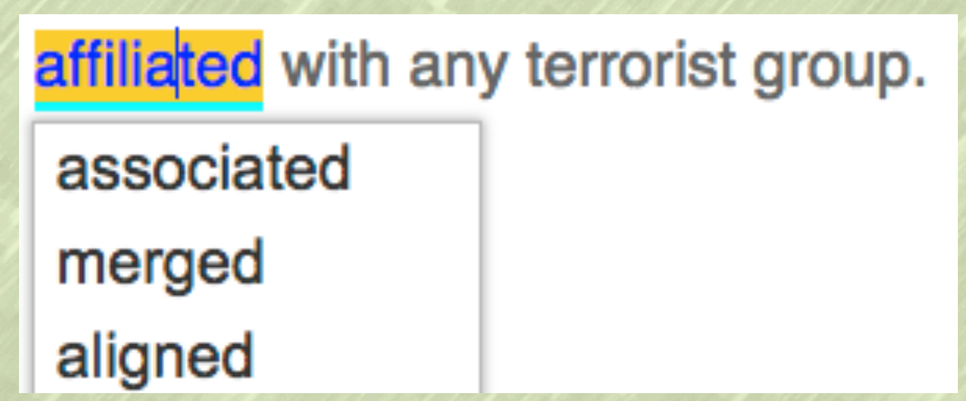
- The Discounted Cumulative Gain (DCG), which penalizes, or discounts (logarithmically), each rating based on its position in the results

$$Discounted\,CG_p = DCG_p = \sum_{i=1}^{p} \frac{rating(i)}{log_2(i+1)}$$

- Ideal DCG (IDCG), which is the DCG of the best possible results based on the given ratings

$$Ideal\ DCG_p = IDCG_p = \sum_{i=1}^{|REL|} \frac{rating(i)}{log_2(i+1)}$$

$$Normalized\ DCG_p = \frac{DCG_p}{IDCG_p}$$


affiliated with any terrorist group.
associated
merged
aligned