A flexible online system for curating reduced redundancy language exercises and tests

Torsten Zesch[1], Andrea Horbach[2], Melanie Goggin[3], and Jennifer Wrede-Jackes[4]

Abstract. We present a tool for the creation and curation of C-tests. C-tests are an established tool in language proficiency testing and language learning. They require examinees to complete a text in which the second half of every second word is replaced by a gap. We support teachers and test designers in creating such tests through a web-based system using Natural Language Processing (NLP) techniques. We provide support both for creating a test from a given text according to guidelines for different languages, as well as for automatically assessing the overall difficulty of the created test.

Keywords: C-tests, exercise generation, difficulty prediction.

1. Introduction

Reduced redundancy exercise formats like C-tests (Grotjahn, 2014) are common and established tools in language learning. C-tests are also frequently used in assessments because they correlate well with general language proficiency (Eckes & Grotjahn, 2006). A C-test consists of a text paragraph with a set of incomplete words containing gaps which the examinee must complete. The prefix of the incomplete word is shown as a hint for the learner. Figure 1 shows an example of an English C-test.

1. Language Technology Lab, University Duisburg-Essen, Germany; torsten.zesch@uni-due.de
2. Language Technology Lab, University Duisburg-Essen, Germany; andrea.horbach@uni-due.de
3. Institut für Optionale Studien, University Duisburg-Essen, Germany; melanie.goggin@uni-due.de
4. Institut für Optionale Studien, University Duisburg-Essen, Germany; jennifer.wrede@uni-due.de

Figure 1.   English C-test with 20 gaps (marked with curly brackets)

Wild boar in Berlin

With their hefty trunks, sharp tusks and adorable striped piglets, wild boars are easy to spot. If y{**ou**} were stan{**ding**} in cen{**tral**} *Berlin's Alexanderplatz* shopping squ{**are**}, they wo{**uld**} be ve{**ry**} hard t{**o**} miss. Wi{**ld**} boars a{**re**} thriving i{**n**} the ci{**ty**} and ha{**ve**} even be{**en**} spotted roa{**ming**} through t{**he**} busiest ar{**eas**}. The popul{**ation**} is curr{**ently**} estimated a{**t**} 3,000 and th{**ere**} are three isolated populations in forests of the capital. One person is licensed to cull the pigs in the city, but with restrictions on hunting and no natural predators, the wild boars are here to stay.

According to teachers and test designers (Arras, Eckes, & Grotjahn, 2002), the time-consuming process of designing C-tests is a major hindrance, especially if such tests are used as a type of exercise in language learning instead of summative assessment. Here, it is necessary to create a large number of tests in advance.

In order to decrease the workload of teachers and test creators, we have developed a flexible online tool that allows for the fast and dynamic creation and curation of C-tests. The features of the tool include: (1) automatic application of a general gap scheme, as well as specific gap-schemes for several languages; (2) one-click addition or deletion of gaps for fine-grained manual adjustment; (3) an option to manually adjust the number of deleted characters per gap as well as (4) to specify additional solutions for a gap; and (5) various import and export functionalities.

The tool is freely available at https://github.com/zesch/ctest-builder. We not only host a web-instance, but also give full access to the source code. This allows users to install their own instance, so that it can be adapted to new languages and to ensure that non-free texts do not have to be sent over the Internet.

In the following, we will discuss how C-tests are automatically created using NLP techniques. Subsequently, we will present ongoing and future work regarding how to further automate the C-test creation process.

2.   C-test creation

The core part of our tool is the creation of a C-test from plain text. We provide automatic gap assignment, which the user can adapt later according to their needs.

Incorporated into the tool are both generic and language-specific criteria for C-test creation. Figure 2 shows a screenshot of the tool.

Figure 2. Screenshot of the C-test tool
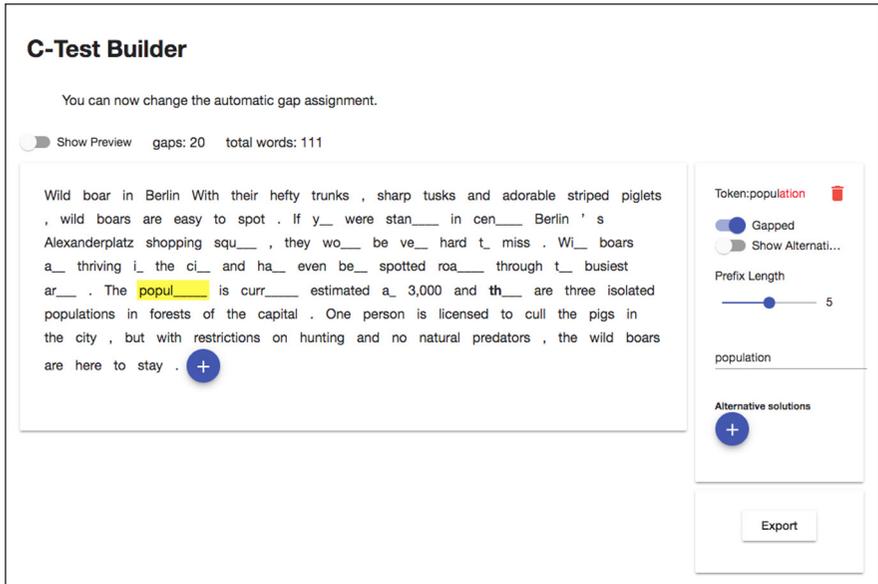
While the overall creation rule for C-tests is simple (gap the second half of every second word), there are a number of exceptions that require NLP to ensure a high quality in the automatic gap assignment methods, such as part-of-speech tagging (the identification of word classes like nouns, verbs, or adjectives) and named entity recognition (identifying, e.g. persons, locations, or organizations). We make sure that numbers and dates, abbreviations, punctuation, and named entities are not included in the gapping scheme as they often cannot be predicted or such prediction requires knowledge beyond mere language proficiency.

The process for gap assignment is language independent in its basic version, assuming that a suitable tokenizer (i.e. a tool that automatically identifies word boundaries) for the language in question is available. We currently assume that tokens can be separated by white-spaces and cannot yet deal with languages where this is not the case, such as in Chinese. For a number of languages (English, French, German, Spanish, and Italian), we have already implemented more specific gap assignment methods taking language-specific phenomena into account. However, this requires language-specific NLP tools that are not always available. Thus, we

also incorporate simple fallbacks, e.g. counting every capitalized word as a named entity.

An important phenomenon to consider for German, as well as English to some extent, is the frequent occurrence of noun compounds, such as *Haustürschlüssel* (literally 'house door key'). Applying the generic rule to split words in equal halves (*Haustürs*_____) often leads to gaps which are very hard to predict; therefore, compounds in German are usually treated in such a way that only the (right-most) head noun of the compound is gapped. In our example, this leads to the gapped word *Haustürschl*_____. We incorporate this behavior into our tool by employing automatic compound splitting.

In many romance languages, tokens containing clitics like the French *qu'aujourd'hui* need to be properly segmented before adding a gap. In the above example, proper tokenization should split the word into *qu'* and *aujourd'hui,* which would be reduced to *aujou*_____. Therefore, we use dedicated tokenization methods for each language.

In addition to the automatic gap assignment, users always have full control over the C-test and can manually adjust it. For example, users can mark words which should not be gapped, so that these words are not considered for automatic gapping. After the initial gap assignment, users may also modify the C-test by adding or deleting words or modifying the number of deleted characters for a gap.

The system automatically stores the correct solution for a gap based on the input text, so this information can be used later for automatic evaluation of a learner submission for the test. In some cases, however, more than one solution is possible. For example, in the English sentence *They returned to their ho*___, both *house* and *home* could in many contexts be a correct solution for the gap. Our tool gives teachers the option to specify such additional correct solutions for a gap. In the future, we also plan to identify alternative plausible solutions that are potentially unforeseen by the human test creator. One way to do this automatically would be the use of language models, which statistically predict the most likely words for a specific context.

The final C-tests can be exported in various formats to ease the integration with existing computer-assisted language learning systems. We support, for example, export in Moodle format as well as PDFs, which can be used to fill out the test pen-and-paper style. The online system can also be used as a convenient editor to import and adapt existing C-tests.

3.   Outlook and future work

The next steps we take will focus on two aspects related to creating C-Tests: predicting the difficulty of a C-test and searching the web for suitable textual material.

Predicting the difficulty of a C-test is a difficult task for humans (Beinborn, Zesch, & Gurevych, 2014). Therefore, field-tests with real learners are usually necessary before a test can be used to assign test takers to a proficiency level. Automatically assessing the difficulty of a C-test can therefore help to shorten the production cycle for new tests. To this end, we will include in the system a recently developed method (Beinborn et al., 2014) to reliably predict the difficulty of a given gap. This not only helps the exercise designer to adapt the overall difficulty of the test to the appropriate level for a given group of language learners, but also to identify individual problematic gaps.

Identifying suitable texts as input for the C-test tool is another important direction for future work. An appropriated text should be, as far as possible, thematically self-contained within a specific length constraint. It should have adequate complexity and linguistic difficulty. Finding such texts can be time-consuming for humans, while text mining and NLP methods can help by searching the web for good text candidates for a C-test. In the future, we plan to incorporate such a tool into the C-test builder.

4.   Acknowledgements

References

Arras, U., Eckes, T., & Grotjahn, R. (2002). C-Tests im Rahmen des „Test Deutsch als Fremdsprache" (TestDaF): Erste Forschungsergebnisse. In R. Grotjahn (Ed.), Der CTest: theoretische Grundlagen und praktische Anwendungen (pp. 175-209). AKSVerlag.

Beinborn, L., Zesch, T., & Gurevych, I. (2014). Predicting the difficulty of language proficiency tests. *Transactions of the association for computational linguistics, 2,* 517-529.

Eckes, T., & Grotjahn, R. (2006). A closer look at the construct validity of C-tests. *Language Testing, 26*(3), 290-325. https://doi.org/10.1191/0265532206lt330oa

Grotjahn, R. (Ed.). (2014). Der C-Test: aktuelle Tendenzen- The C-test: current trends [Bilingual]. Peter Lang GmbH, Internationaler Verlag der Wissenschaften. https://doi.org/10.3726/978-3-653-04578-9

Research-publishing.net