Towards Automatic Scoring of Cloze Items by Selecting Low-Ambiguity Contexts

Tobias Horsmann, Torsten Zesch

Language Technology Lab
University of Duisburg-Essen
Germany

{tobias.horsmann, torsten.zesch}@uni-due.de

Abstract

In second language learning, cloze tests (also known as fill-in-the-blank tests) are frequently used for assessing the learning progress of students. While preparation effort for these tests is low, scoring needs to be done manually, as there usually is a huge number of correct solutions. In this paper, we examine whether the ambiguity of cloze items can be lowered to a point where automatic scoring becomes possible. We utilize the local context of a word to collect evidence of low-ambiguity. We do that by seeking for collocated word sequences, but also taking structural information on sentence level into account. We evaluate the effectiveness of our method in a user study on cloze items ranked by our method. For the top-ranked items (lowest ambiguity) the subjects provide the target word significantly more often than for the bottom-ranked items (59.9% vs. 36.5%). While this shows the potential of our method, we did not succeed in fully eliminating ambiguity. Thus, further research is necessary before fully automatic scoring becomes possible.

Keywords: cloze tests, language proficiency tests, automatic scoring.

Tobias Horsmann and Torsten Zesch 2014. Towards automatic scoring of cloze items by selecting low-ambiguity contexts. *Proceedings of the third workshop on NLP for computer-assisted language learning.* NEALT Proceedings Series 22 / Linköping Electronic Conference Proceedings 107: 33–42.

1 Introduction

Cloze items (Taylor, 1953, 1956; O'Toole and King, 2011) are frequently used to test language proficiency. A cloze item consists of a sentence with usually one word being blanked. The learner's task is to find the correct word for the blank:

(1) He sold his ____ yesterday below price.

As we can see from example (1), blanks can be quite ambiguous, i.e. a very high number of correct solutions exists. In this example, a wide range of nouns are acceptable solutions including *books*, *house*, or *bike*, but also *daughter* or *kidney* cannot be ruled out in this (quite limited) context. Such ambiguity is not only a problem for language learners, but even native speakers frequently fail when facing such a task (Klein-Braley and Raatz, 1982).

If cloze items should be automatically generated and scored, ambiguous items pose a serious problem, as we only know for sure one correct answer namely the one that was used in the original sentence. Students might get frustrated if they provide a valid solution that is not recognized by the system. The same problem affects an alternative solution to the problem: providing a list of alternative answer options - called distractors (Sumita et al., 2005). Determining whether a distractor is actually another valid solution is equivalent to the problem described above. Thus, finding good distractors is still an unsolved problem that attracts a lot of research (Lee and Seneff, 2007; Smith and Avinesh, 2010; Sakaguchi et al., 2013; Zesch and Melamud, 2014). Furthermore, providing distractors for cloze items also considerably changes the nature of the task, as distractors are recognition stimuli, i.e. the student recognizes the correct answer rather than having to actively produce it (Gonzáles, 1996).

Now consider example sentence (2):

(2) I went to the ____ today and now I have sand in my shoes.

Most people would come up with *beach* or maybe *desert*, while other solutions are highly unlikely, which means that the blank is less ambiguous than example (1). This leads to our research question, whether it is possible to find contexts that are specific enough to only allow one correct solution. Such a setup would dramatically simplify automatic scoring.

We limit the scope of this work to determine low-ambiguity contexts for single-word nouns and leave other parts of speech for future work. In the next section, we describe how such contexts can be detected.

2 Detecting Low-Ambiguity Contexts

In order to determine low-ambiguity contexts, we introduce several detectors that collect evidence of low ambiguity. Figure 1 shows the process chain: First, a target word is chosen for which a context of low ambiguity shall be found. If a sentence contains the target word, we run all detectors in parallel and combine their scores to form the final score. We manually determined the perceived reliability of the detectors and assigned them to three classes *strong*, *medium*, and *weak*. Each of the classes correspond to a certain weight of the detector in the final score: a medium detector is 5 times as important as a weak one, and a strong detector is 20 times more important than a weak one. Table 1 provides an overview of the class assignments for detectors. As this is basically a linear

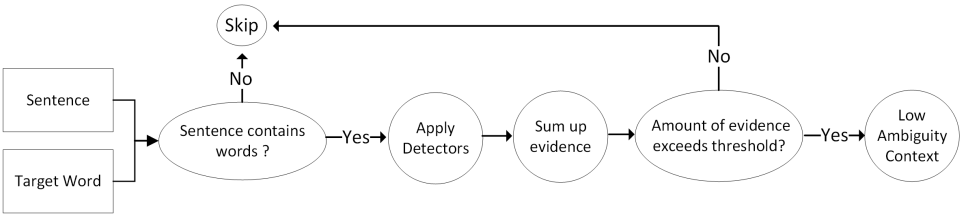

Figure 1: Low Ambiguity Context Determination Process Chain

Detector	Reliability	Weigth
Distributional Thesaurus	*weak*	1
POS Pattern Sliding Window	*weak*	1
Bigram Sliding Window	*medium*	5
Skip-Bigrams	*medium*	5
Hearst-Patterns	*medium*	5
3-5 Gram Sliding Window	*strong*	20
Word Repetition	*strong*	20

Table 1: List of detectors with their reliability and assigned weigth

regression with hand-assigned weights, the weights can (and should) be learned if labeled training data is available.

In the remainder of this section, we describe our detectors in more detail. The detectors are not mutually exclusive and a word might be detected by several detectors.

2.1 Collocation-based Detectors

Subsequently, we will introduce three detectors that are variations of testing a word for being collocated with one or more nearby words. These detectors are motivated by the research field of word prediction where an algorithm tries to determine the next word a user might want to type on a keyboard based on the already entered word sequence (Li and Hirst, 2005; Trnka, 2008; Aliprandi et al., 2007). We hypothesize that if a word is easy to predict it should also be less ambiguous in that context. We further expect that longer collocated sequences are easier to predict than shorter ones.

Another way to look at this problem is its relation to language model perplexity (Chen et al., 1998). Classically, perplexity makes a statement about how well a model predicts test data. In our task, we try to do the exact opposite. We want to determine a test set on which we would achieve a low perplexity on our (static) language model.

3-5 Gram Sliding Window This detector tests if the target word and its neighboring words are collocated, taking into account three, four, and five grams.

We start by moving a 5-gram context-window over the sentence and signal a match if the association strength of the ngram exceeds the threshold. Otherwise, we move the window one position to the right and repeat until either a match occurs or the target word became the most left-hand word in the window. If no match occurs, the window size is decreased and the procedure repeated.

An example sentence that demonstrates the detection of a three word collocation is depicted in

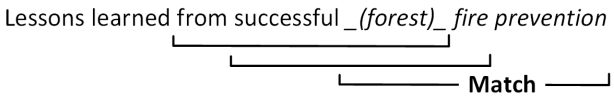

Figure 2: Sliding Window: Collocation detection with window size of three

Figure 2. The target word is *forest*. The window has to be shifted two times until it is over the phrase *forest fire prevention* in order to match.

In order to quantify word association strength, we use pointwise mutual information (PMI) (Church and Hanks, 1990) and obtain the required word frequencies from the Google Web1T corpus (Brants and Franz, 2006). As PMI is only defined for bigrams, sequences of length $n \geq 3$ have to be split into two units that are pseudo-bigrams.[1] For example, calculating the PMI for *forest fire prevention*, would either lead to a split such as [*forest fire*] [*prevention*] or [*forest*] [*fire prevention*].

Following Korkontzelos et al. (2008), we use the so called "pessimistic split" to compute PMI by using the split with the highest likelihood of all possible splits:

$$\text{PMI}_{pess}(w_1, ..., w_n) = log \frac{P(w_1,...,w_n)}{P(w_1,..,w_i)P(w_{i+1},..,w_n)}$$

Using the highest likelihood split decreases the number of false alarms by the detector. This is a pessimistic, conservative way of calculating PMI (Hartmann et al., 2012) because the best-split of many actually collocated words will not have co-occurrences high enough to exceed the threshold.[2] Hence, we increase the precision in detection of true collocations, but lower the recall. In combination with the high threshold, we limit the detected word-sequences to those which are extremely frequent to occur in daily life. We expect that the frequency and commonness of such a word sequence will provide the most ideal circumstance to restore a deleted word.

Bigram Sliding Window We also use a bigram sliding window detector, but found that it does not work as reliable as the 3-5 gram detector. Thus, we assign a *medium* reliability. Also, PMI values are usually lower so that we need to set a different threshold (5 instead of 10) in order to ever trigger the detector.

Skip Bigrams Words that occur frequently together do not need to be directly adjacent, but can be separated by other words. To detect such cases, this detector calculates the PMI value between the target word and adjectives or verbs occurring in an n-word window around it. We consider the four words to the left and to the right of the target word. The words directly adjacent to the target word are omitted as this case is already handled by the *Bigram Sliding Window* detector.

For example, we want to detect cases such as:

It aims to be both, ***empirical****, but novel ____. , → research.*

The word *empirical* is clearly referencing to *research* in this case.

We assign a *medium* reliability to this detector and used the same PMI threshold of 5 as for the other bigram detector.

[1] "pseudo-bigram" because such a bigram does not necessarily contain two words, it might only be one, but also three or more depending on the length of the word sequence.

[2] We use a threshold of 10

2.2 POS Pattern Sliding Window

This detector works like the other sliding window detectors, but relies on part-of-speech (POS) ngrams instead of token ngrams. We use a list of POS pattern from Arranz et al. (2005) that were originally used for detecting multiword expressions. We found that the approach did not work well, as contexts with low and high ambiguity can have the same POS patterns. For example, "balance of international payment" (NN-IN-JJ-NN) has the same POS pattern as "car with expensive tires" but is less ambiguous depending on the position of the cloze item. We thus assign a *low* reliability, but still think that the detector can provide supporting evidence for other more important detectors.

2.3 Word Repetition

If the deleted word is repeated later in the sentence, it proposes itself as possible candidate for solving the blank. The detector matches if the target word occurs a second time in the sentence in any inflection form.

Go through the ____ and mark each affected slice., → *slices*

We found that this is a *strong* indicator.

2.4 Distributional Thesaurus

A sentence may contain many words that are semantically related to the target word, but that do not appear in a collocation:

The ***compressor*** *is necessitated by ____ which injects fuel into the* ***cylinder*** → air

In the example, we show semantically related words in bold-face. They help to guide the reader towards the right choice for the blank.

We detect semantically related words using the distributional thesaurus from Biemann and Riedl (2013) that was computed over the top 1 million words of the English Google Books Corpus (Goldberg and Orwant, 2013). We retrieve the fifty highest ranked words that are associated with our target word in the thesaurus. These words are compared with all words occurring in the sentence. If two or more associated words are found in the sentence the detector matches.

Unfortunately, the detector fires quite often, even in cases of a rather weak relationship between two words. We thus assign a *low* reliability.

2.5 Hearst Patterns

Hearst patterns (Hearst, 1992) are lexico-syntactic pattern like "X such as Y, and Z" that are frequently used to detect hierarchical relationships between words. We argue that wherever such a pattern can be found in a sentence, the ambiguity should be reduced as the reader can be guided by the explicitly stated relationships.

Already small amounts of ____
such as ***beer*** *or* ***wine*** *can affect your ability to drive.* → alcohol

This structure allows the reader to abstract to the more general, deleted word. In the case of *beer* and *wine*, the reader may conclude that the deleted word is *alcohol*. Thus, if the more general word is deleted, the list of more specific words allows determining the deleted word. Note that the other direction is usually more difficult, e.g. 'fruit → orange' is harder than 'orange → fruit' as there are many different fruit, but only one more general concept.

We found that the detector works quite well, but produces some false alarms. We thus assign a *medium* reliability.

3 Experimental Setup

In order to evaluate our methods, we have to determine how often human subjects are able to complete the cloze items with the target word. If subjects consistently only give the correct answer, our method would work perfectly. However, as this might be too optimistic, we will measure how often the correct word (the originally deleted word) is provided and how many different alternatives human subjects provide.

In order to create the evaluation dataset, we randomly select sentences from the UkWaC corpus (Baroni et al., 2009) which contains general-purpose text obtained from uk-domain websites. The sentences are filtered by their reading-difficulty using the Flesch–Kincaid test (Kincaid et al., 1975): we remove all sentences requiring more than 10 years of school-education. We then only keep sentences that contain a noun from a randomly chosen subset of common nouns. We then apply our method to the remaining sentences obtaining weight scores from 0 to 52, where a higher number means more evidence (less ambiguity). From this ranking, we select the 25 top-ranked sentences and the 25 bottom-ranked sentences. The bottom-ranked sentences have a score-range of 11 to 16, thus, none of the *strong* detectors contributed to their score. We compiled a cloze test from both collections (replacing the common noun with a blank) and asked volunteer participants to solve the items.

Overall, 30 native speakers of English completed the study, which was conducted online. Before the actual study, participants were shown a detailed manual describing the study. Participants were asked not to cheat and not to use any search engines or ask other people for second opinions. If they could not come up with an answer, participants were instructed to move to the next sentence. The fifty sentences were offered over 4 web pages; each page briefly repeated the manual.

4 Results & Discussion

Figure 3 shows the result of the study in terms of how often participants provided the target word. For the top-ranked items, participants provided the target word on average in 59.9% of the cases, while the average was only 36.5% in the bottom-ranked group. This clearly shows that our method is effective in reducing the ambiguity of cloze items. However, we did not succeed in fully eliminating ambiguity.

The achieved ambiguity reduction can also be measured by considering how many different answers were provided for solving a cloze item. The top-ranked cloze items had an average of 4.5 different answers per item including the target word. Thus, three to four alternative answers were always provided although the frequency for choosing the target word clearly outweighs those of the alternatives. For the bottom-ranked cloze items, participants provided on average almost twice as much different answers (8.4).

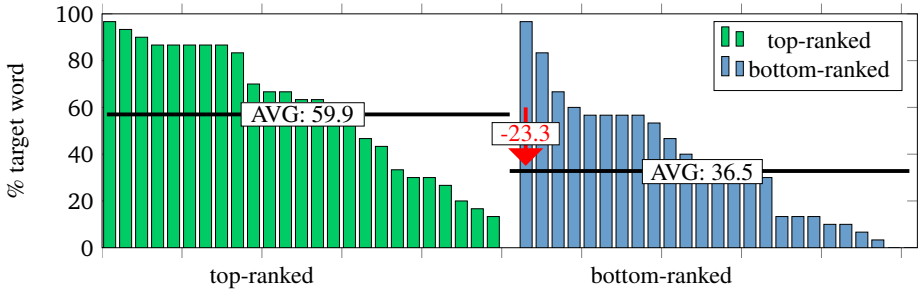

Figure 3: Ratio of participants providing correct answer for top-ranked and bottom-ranked items

4.1 Error Analysis

Our methods worked well in some cases but also yielded poorly performing cases. We discuss subsequently examples for both.

Top-ranked (correct) The following sentences are examples where the user study confirmed a low-ambiguity context with at least 80% of all human subjects providing the target word.

> (A) It is important that you spend some ____ over the next few weeks to help your kitten adjust to its new family so please take the time to read our information before you get your new kitten . → *time* (96%)
>
> (B) Effects : A trip can take from 20 minutes to an ____ to start and usually lasts about 12 hours . → *hour* (93%)
>
> (C) In 1974 the former girls' premises were occupied by Orange Hill junior high ____ and the boys ' by Orange Hill senior high school . → *school* (86%)

The cloze items in sentence (A-C) is placed within very common phrases. In (A) and (C) the local word context alone was sufficiently specific to fill the item with its original word. In sentence (B) we have even more hints: (i) the target word is repeated within the sentence, and (ii) the determiner "an" right in front of the blank further limits the possible candidate answers.

Top-ranked (wrong) For the following sentences only 20% or less of the participants provided the target word:

> (E) They have various difficulties including learning difficulties, physical difficulties, emotional and behavioural ____ as well as mental health problems. → *problems* (20%)
>
> (F) All stores offer a wide and varied ____ of hot tubs and spas, please call us to arrange a dip at anytime on 0800 085 8880 or go to the showroom locator to find your nearest branch. → *range* (16%)
>
> (G) The ground floor has a well equipped living room, dining ____ and kitchen and the private seating area is only a few steps away, in the garden. → *area* (13%)

The sentences in the top-ranked group are mainly from matches of collocation-based detectors and word redundancy. In sentence (E) *difficulties* (30%) is provided most frequently, but was not used in

the original sentence probably due to stylistic considerations. In (G) *room* (80%) is selected due to a similar pattern. In sentence (F) the most frequent answer is *selection* (63%) instead of *range*. *Selection* is a synonym of *range* in this context. In order to correctly rank cases like this we need to take competing candidates into account in the ranking process.

Bottom-ranked (wrong) Two items stand out in the bottom-ranked group, because participants provided the target word in over 80% of cases so they should actually be in the top-ranked group.

> (H) I managed to ride the bike to work with the handlebars pointing in a different ____ to the wheel and almost in tears. → *direction* (96%)
>
> (I) The disease usually starts in the wrists, hands or feet, and can spread to other joints and other ____ of the body. → *parts* (83%)

In case of sentence (H) *pointing in a different direction* missed the threshold, but is actually a quite strong collocation. In sentence (I), the enumeration of body parts provided the needed context to determine the deleted word. Our *Hearst* detector matched, but its weight was not enough to put the item in the top-ranked group.

Bottom-ranked (correct) The following two sentences were not answered by a single participants with the target word. Although they belong in the bottom-ranked group, the especially low score deserves some discussion.

> (J) This new edition contains many more illustrations and anecdotes, and two new chapters on ____'s surviving Bristol Channel pilot cutters and their restoration and model making of these craft. → today (0%)
>
> (K) Anta Scotland Ltd Specialise in fabrics and ceramics and have various contemporary versions of traditional ____ such as tartans. → designs (0%)

In case of (J), the possessive case of the deleted word confused almost all participants. This blank remained unanswered quite often (40%). If an answer was provided it was rather a proper noun. Sentence (K) was most frequently answered with *fabrics* (40%). The *Hearst* detector matched, but we assume that our participants were unfamiliar with the word *tartans*, and thus could not take advantage of this structural hint. This illustrates why we only put medium weights on the *Hearst* detector. The conditions under which a detector is useful remains difficult to predict.

5 Conclusion & Future Work

In this paper, we discussed methods for determining cloze items with reduced ambiguity. We introduced seven detectors in order to find such items. We found that our methods are able to significantly reduce the ambiguity of blanks, but that we could not reach our goal of a single valid answer per item.

In future work, we want to improve the detectors in order to further reduce ambiguity of selected sentences. We also need to address the problem that cloze items might become to easy, as e.g. word repetition is a strong detector, but obviously not very useful when generating a language proficiency test.

References

Aliprandi, C., Carmignani, N., and Mancarella, P. (2007). In *International Journal of Computing and Information Sciences*, volume 5, pages 79–85.

Arranz, V., Atserias, J., and Castillo, M. (2005). Multiwords and word sense disambiguation. In Gelbukh, A., editor, *Computational Linguistics and Intelligent Text Processing*, volume 3406 of *Lecture Notes in Computer Science*, pages 250–262. Springer Berlin Heidelberg.

Baroni, M., Bernardini, S., Ferraresi, A., and Zanchetta, E. (2009). The wacky wide web: A collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, 3:209–226.

Biemann, C. and Riedl, M. (2013). Text: now in 2d! a framework for lexical expansion with contextual similarity. *Journal of Language Modelling*, 1:55–95.

Brants, T. and Franz, A. (2006). Web 1t 5-gram corpus version 1.1. *Linguistic Data Consortium*.

Chen, S., Beeferman, D., and Rosenfeld, R. (1998). Evaluation Metrics for Language Models. In *DARPA Broadcast News Transcription and Understanding Workshop (BNTUW)*, Lansdowne, Virginia, USA.

Church, K. W. and Hanks, P. (1990). Word association norms, mutual information, and lexicography. *Comput. Linguist.*, 16(1):22–29.

Goldberg, Y. and Orwant, J. (2013). A dataset of syntactic-ngrams over time from a very large corpus of english books.

Gonzáles, A. B. (1996). Testing english as a foreign language: an overview and some methodological considerations. *Revista española de lingüística aplicada*, 11:71–94.

Hartmann, S., Szarvas, G., and Gurevych, I. (2012). Mining multiword terms from wikipedia. In Pazienza, M. T. and Stellato, A., editors, *Semi-Automatic Ontology Development: Processes and Resources*, pages 226–258. IGI Global, Hershey, PA, USA.

Hearst, M. A. (1992). Automatic acquisition of hyponyms from large text corpora. In *In Proceedings of the 14th International Conference on Computational Linguistics*, pages 539–545.

Kincaid, P. J., Fishburne, R. P. J., Rogers, R. L., and Chissom, B. S. (1975). Derivation of new readability formulas (automated readability index, fog cound and flesch reading ease formula) for navy enlisted personnel. *Naval Technical Training Command: Research Branch Report 8-75*.

Klein-Braley, C. and Raatz, U. (1982). Der c-test: ein neuer ansatz zur messung von allgemeiner sprachbeherrschung. *AKS-Rundbrief*, pages 23–37.

Korkontzelos, I., Klapaftis, I., and Manandhar, S. (2008). Reviewing and evaluating automatic term recognition techniques. In Nordström, B. and Ranta, A., editors, *Advances in Natural Language Processing*, volume 5221 of *Lecture Notes in Computer Science*, pages 248–259. Springer Berlin Heidelberg.

Lee, J. and Seneff, S. (2007). Automatic generation of cloze items for prepositions. *Interspeech*.

Li, J. and Hirst, G. (2005). Semantic knowledge in word completion. In *Proceedings of the 7th international ACM SIGACCESS conference on Computers and accessibility*, Assets '05, pages 121–128, New York, NY, USA. ACM.

O'Toole, J. M. and King, R. A. R. (2011). The deceptive mean: Conceptual scoring of cloze entries differentially advantages more able readers. *Language Testing*.

Sakaguchi, K., Arase, Y., and Komachi, M. (2013). Discriminative approach to fill-in-the-blank quiz generation for language learners. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 238–242.

Smith, S. and Avinesh, P. (2010). Gap-fill tests for language learners: Corpus-driven item generation. In *Proceedings of ICON-2010: 8th International Conference on Natural Language Processing*.

Sumita, E., Sugaya, F., and Yamamoto, S. (2005). Measuring non-native speakers' proficiency of english by using a test with automatically-generated fill-in-the-blank questions. In *Proceedings of the second workshop on Building Educational Applications Using NLP*, EdAppsNLP 05, pages 61–68, Stroudsburg, PA, USA. Association for Computational Linguistics.

Taylor, W. (1953). Cloze procedure: A new tool for measuring readability. *Journalism Quarterly*, 30:415–433.

Taylor, W. (1956). Recent developments in the use of cloze procedure. *Journalism Quarterly*, 33:42.

Trnka, K. (2008). Adaptive language modeling for word prediction. In *Proceedings of the ACL-08: HLT Student Research Workshop*, pages 61–66, Columbus, Ohio. Association for Computational Linguistics.

Zesch, T. and Melamud, O. (2014). Automatic Generation of Challenging Distractors Using Context-Sensitive Inference Rules. In *Proceedings of the 9th Workshop on Innovative Use of NLP for Building Educational Applications at ACL*, Baltimore, USA.