Interpreting Neural CWI Classifiers' Weights as Vocabulary Size

Yo Ehara
Shizuoka Institute of Science and Technology / 2200-2, Toyosawa, Fukuroi, Shizuoka, Japan.
ehara.yo@sist.ac.jp

Abstract

Complex Word Identification (CWI) is a task for the identification of words that are challenging for second-language learners to read. Even though the use of neural classifiers is now common in CWI, the interpretation of their parameters remains difficult. This paper analyzes neural CWI classifiers and shows that some of their parameters can be interpreted as vocabulary size. We present a novel formalization of vocabulary size measurement methods that are practiced in the applied linguistics field as a kind of neural classifier. We also contribute to building a novel dataset for validating vocabulary testing and readability via crowdsourcing.

1 Introduction

The readability of second-language learners has attracted great interest in studies in the field of natural language processing (NLP) (Beinborn et al., 2014; Pavlick and Callison-Burch, 2016). As NLP mainly addresses automatic editing of texts, readability assessment studies in this field have focused on identifying complex parts by assuming that the words identified are eventually simplified so that learners can read them. To this end, complex word identification (CWI) (Paetzold and Specia, 2016; Yimam et al., 2018) tasks have been studied extensively. Recently, a personalized CWI task has been proposed, where the goal of the task is to predict whether a word is complex for each learner in a personalized manner (Paetzold and Specia, 2017; Lee and Yeung, 2018). Neural models are also employed in these studies and have achieved excellent performance.

The weights, or parameters, of a personalized high-performance neural CWI, obviously include information on how to measure the word difficulty and learner ability from a variety of features. If such information could be extracted from the model in a form that is easy to interpret, it would not only be use (Hoshino, 2009; Ehara et al., 2012, 2013, 2014; Sakaguchi et al., 2013; Ehara et al., 2016, 2018; Ehara, 2019). To this end, this paper proposes a method for interpreting the weights of personalized neural CWI models. Let us suppose that we have a corpus and that its word frequency ranking reflects its word difficulty. Using our method, a word's difficulty can be interpreted as the *frequency rank* of the word in the corpus and a learner's ability can be interpreted as the *vocabulary size* with respect to the corpus, i.e., the number of words known to the learner when counted in a descending order of frequency in the corpus.

Our key idea is to compare CWI studies with *vocabulary testing* studies in applied linguistics (Nation, 2006; Laufer and Ravenhorst-Kalovski, 2010). Second-language vocabulary is extensive and occupies most of the time spent in learning a language. Vocabulary testing studies focus on measuring each learner's second language vocabulary quickly. One of the major findings of these studies is that a learner needs to "know" at least from 95% to 98% of the tokens in a target text to read. Here, to measure if a learner "knows" a word, vocabulary testing studies use the learner's vocabulary size and word frequency ranking of a balanced corpus. Hence, by formalizing the measurement method used in vocabulary testing studies as a neural personalized CWI, we can interpret neural personalized CWI models' weights as vocabulary size and word frequency ranking.

Our contributions are summarized as follows:

1. To predict whether a learner knows a word through the use of a vocabulary test result in hand, vocabulary size-based methods were previously used for vocabulary testing. We show that this method can represent a special case of typical neural CWI classifiers that take a specific set of features as input. Furthermore, we theoretically propose novel methods that enable the weights of certain neural classifiers

Proceedings of the 15th Workshop on Innovative Use of NLP for Building Educational Applications, pages 171–176

to become explainable on the basis of the vocabulary size of a learner.

2. To validate the proposed models, we want a dataset in which each learner/test-taker takes both vocabulary and reading comprehension tests. To this end, we build a novel dataset and make it publicly available.

2 Related Work

2.1 Vocabulary size-based testing

Vocabulary size-based testing studies (Nation, 2006; Laufer and Ravenhorst-Kalovski, 2010) measure the vocabulary size of second-language learners. Assuming that all learners memorize words in the same order, i.e., that the difficulty of words is identical for each learner, all words are ranked in one dimension using this method. Subsequently, it is determined whether or not a learner knows a target word by checking if the vocabulary size of the learner is greater than the easiness *rank* of the word.

The vocabulary size-based method can be formalized as follows. Let us consider the case in which we have J learners $\{l_1, l_2, \ldots, l_j, \ldots, l_J\}$ and I words $\{v_1, v_2, \ldots, v_i, \ldots, v_I\}$. j is the index of the learners and i is the index of the words. When there is no ambiguity, we denote word v_i as word i and learner l_j as learner j, for the sake of simplicity. We write the *rank* of word v_i as r_i and the vocabulary *size* of learner l_j as s_j. Then, to determine whether learner l_j knows word v_i, the following decision function f is used:

$$f(l_j, v_i) = s_j - r_i \tag{1}$$

Interpretting Eq. 1 is simple: if $f(l_j, v_i) \geq 0$, then learner l_j knows word v_i; if $f(l_j, v_i) < 0$, then learner l_j does not know word v_i.

The performance of Eq. 1 depends solely on how to determine the vocabulary size of learner l_j, s_j, and the easiness rank of word v_i, r_i. As several methods have previously been proposed to estimate this, we describe them in the following subsections.

2.1.1 Measuring rank of word v_i

Easiness *rank*s of words are important in vocabulary size-based testing. To this end, word frequency rankings from a balanced corpus, especially the British National Corpus (BNC Consortium, 2007), are used: the more frequent words in the corpus are ranked higher and considered to be easier. Some previous studies in the field manually adjust the BNC word frequency rankings to make them compatible with language teachers' intuitions. BNC collects British English. Recent studies also take into account word frequency obtained from the Corpus of Contemporary American (COCA) English (Davies, 2009) by simply adding the word frequencies of both corpora in order to obtain a word frequency ranking.

2.1.2 Measuring the vocabulary size of learner l_j

An intuitive and simple method for measuring the vocabulary size of learner l_j is as follows. First, we randomly sample some words from a large vocabulary sample of the target language. Second, we test whether learner l_j knows each of the sampled words and identify the ratio of words known to the learner. Third, we estimate the learner's vocabulary size as the ratio $\times$ the number of correctly answered questions.

This is how the Vocabulary Size Test (Beglar and Nation, 2007) works. Using the frequency ranking of $20,000$ words from the BNC corpus, the words are first split into 20 levels, with each level consisting of $1,000$ words. It is assumed that the $1,000$ words grouped in the same level have similar difficulty. Then, from the $1,000$ words at each level, 5 words are carefully sampled and a vocabulary test is built that consists of 100 words in total. Finally, the number of words that learner l_j correctly answered $\times$ 200 is estimated to be the vocabulary size of learner l_j. This simple method was later validated by a study from another independent group (Beglar, 2010) and is widely accepted.

Examples of the Vocabulary Size Test are publicly available (Nation, 2007). Each question asks learners taking the test to choose the correct answer by selecting one of the four offered options that has the same meaning as one of the underlined words in the question. It should be noted that, in the Vocabulary Size Test, each word is placed in a sentence to disambiguate the usage of each word and each option can directly be replaced with the underlined part without the need to grammatically rewrite the sentence, e.g., for singular/plural differences. Although a typical criticism of vocabulary tests relates to the fact that they do not take contexts into account, each question in the Vocabulary Size Test is specifically designed to account for such criticism by asking the meaning of a word within a sentence.

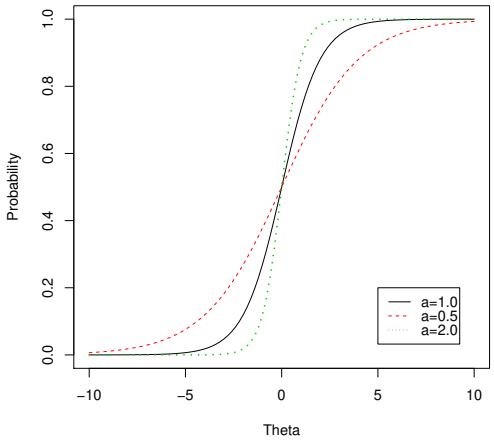

Figure 1: Probability against θ when changing the value of a.

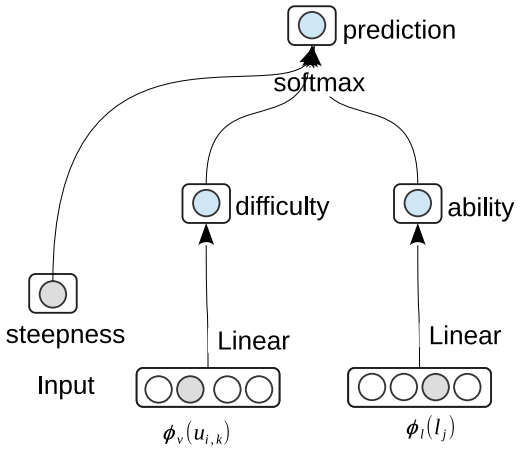

Figure 2: Neural network illustration of a vocabulary size-based prediction.

3 Proposed Formulation

The following notations are used. We have J learners $\{l_1, l_2, \ldots, l_j, \ldots, l_J\}$ and I words $\{v_1, v_2, \ldots, v_i, \ldots, v_I\}$. j is the index of the learners and i is the index of the words. When there is no ambiguity, we denote word v_i as word i and learner l_j as learner j, for the sake of simplicity. Let K_i be the number of occurrences of word v_i. While we do not use in our experiments, for generality, we explicitly write the index for each of the occurrences, i.e., k. Let $u_{i,k}$ be the k-th occurrence of word v_i in the text. Let b_j be the ability of learner j and let $d_{i,k}$ be the difficulty of the k-th occurrence of word v_i.

A dichotomous decision using a neural network-based formulation is typically modeled using a probabilistic formulation. Let $y_{j,i,k}$ be a binary random variable that takes 1 if learner l_j knows the k-th occurrence of word v_i, otherwise it takes 0. Subsequently, it is typical to use a function that maps a real number to the $[0, 1]$ range so that the real number can be interpreted as a probability. To this end, typically σ is the logistic sigmoid function, i.e., $\sigma(x) = \frac{1}{\exp(-x)+1}$ is used. Then, the probability that learner l_j knows the k-th occurrence of word v_i, namely, $u_{i,k}$, can be modeled as in Eq. 2.

$$p(y_{i,k,j} = 1 | u_{i,k}, l_j) = \sigma(a(b_j - d_{i,k})) \quad (2)$$

Qualitative characteristics of Eq. 2 are explained as follows. Let $\theta = a(b_j - d_{i,k})$. The logistic sigmoid function maps an arbitrary real number to the $[0, 1]$ range and makes it possible to interpret the real number as a probability. Here, θ is mapped to the range. As θ increases, the larger the probability becomes. We can see that $a > 0$ is the parameter that determines the steepness of the slope. A large a results in a steep slope. When a is large enough, 4.0 for example, numerically, the function is very close to the *identity* function that returns 0 if $\theta < 0$ and 1 if $\theta \geq 0$.

Probability in a dichotomous classification is most ambiguous when it takes 0.5. By focusing on the point the vertical line takes 0.5, we can see that the sign of $b_j - d_{i,k}$ determines whether or not the probability is larger than 0.5.

3.1 Vocabulary size-based classification as neural classification

These characteristics of Eq. 2 enable it to express the decision function employed in the previous vocabulary size-based decision function Eq. 1 as its special case. Let us consider the case when a is large and the curve is very steep, say $a = 10$, for example. Then, by setting $b_j = s_j$ and $d_{i,k} = r_i$ for all k for word v_i, the decision about whether learner j knows the k-th occurrence of word v_i in Eq. 1 is virtually identical to that of Eq. 2. In this manner, the previous vocabulary size-based decision functions for whether learner l_j knows word v_i in applied linguistics can be converted to a neural network-based classifier and vice versa.

We can see that there exists a *freedom* in the parameters. In the above example, we can achieve the same setting by setting $b_j = 0.1 s_j$, $d_{i,k} = 0.1 r_i$ and $a = 100$. In this way, the same vocabulary size classification can be achieved by different parameter values.

This freedom in terms of parameters is the key for conversion: by setting an appropriate a, we can convert neural classifier parameters as each learner's vocabulary size and the rank of each word.

3.2 Rewriting parameters

While b_j and $d_{i,k}$ are parameters, we rewrite them using one-hot vectors that are widely used to describe neural network-based models. Let us introduce two types of feature functions: ϕ_l and ϕ_v. The former returns the feature vector of learner l_j, and the latter returns the feature vector of the k-th occurrence of word v_i, $u_{i,k}$.

Then, the ability and difficulty parameters of Eq. 2 can be written as the inner product of a weight vector and a feature vector. Let us introduce $\mathbf{w}_l$ as the weight vector for ϕ_l. Let $\mathbf{h}$ be a function that returns the one-hot representation of the augment. We write $\mathbf{h}_l(l_j)$ to denote a function that returns J-dimensional one-hot vector, where only the j-th element is 1 while the other elements are 0. Then, we can rewrite b_j as the inner product of the weight vector and the one-hot vector as $b_j = \mathbf{w}_l^\top \mathbf{h}_l(l_j)$.

In the same way, $d_{i,k}$ can be rewritten as the inner product of its weight vector and feature vector. Being reminded that K_i denotes the number of occurrences of word v_i, we consider a very long $\sum_{i=1}^{I} K_i$-dimensional one-hot vector $\mathbf{h}_v(u_{i,k})$, where only one element that corresponds to the k-th element of word v_i is 1 and all other elements are 0. Then, by introducing a weight vector $\mathbf{w}_v$ that has the same dimension with $\mathbf{h}_v(u_{i,k})$, we can rewrite $d_{i,k}$ as $d_{i,k} = \mathbf{w}_v^\top \mathbf{h}_v(u_{i,k})$. Using these expressions, Eq. 2 can be illustrated using a typical neural network illustration as in Fig. 2.

Overall, the equation using one-hot vector representation can be described as follows:

$$\begin{aligned} &p(y_{i,k,j} = 1 | u_{i,k}, l_j) \\ &\quad = \sigma(a(\mathbf{w}_l^\top \mathbf{h}_l(l_j) - \mathbf{w}_v^\top \mathbf{h}_v(u_{i,k}))) \end{aligned} \tag{3}$$

3.3 Weights as learner vocabulary sizes and word frequency ranks

Eq. 3 provides us with a hint to convert neural classifier weights into vocabulary sizes and word frequency rankings. To this end, we can do the following. First, we use Eq. 3 to estimate parameters: a, $\mathbf{w}_l$, and $\mathbf{w}_v$. Typically, for a binary classification setting using the logistic sigmoid function, cross-entropy loss is chosen as the loss function. We use $L(a, \mathbf{w}_l, \mathbf{w}_v)$ to denote the sum of the cross-entropy loss function for each of the following: all data, all learners, and all occurrences of all words.

From a, $\mathbf{w}_l$ and $\mathbf{w}_v$, we can estimate the frequency rank of word v_i as follows: $a\mathbf{w}_v^\top \mathbf{h}_v(u_{i,k})$. Hence, by comparing the estimate with the observed ranking value r_i of word v_i, we can also tune all parameters. We can simply employ $R(a, \mathbf{w}_v) = \sum_{i=1}^{I} \sum_{k=1}^{K_i} ||a\mathbf{w}_v^\top \mathbf{h}_v(u_{i,k}) - r_i||^2$ for a loss function that measures how distant the estimated rank and the observed rank are. Of course, we can compare $a\mathbf{w}_l^\top \mathbf{h}_l(l_j)$ and s_j, the observed vocabulary size of learner l_j. However, since the observed vocabulary size of each learner is usually much more inaccurate than the ranking of a word, we do not use this term. As ranks usually take large values but never larger than 1, we can use the logarithm of the rank of word v_i for r_i instead of its raw values.

3.4 Proposed Model

Practically, it is important to note that the one-hot vector $\mathbf{h}_v(u_{i,k})$ in L and R functions can be replaced with any feature vector of $u_{i,k}$ or with the k-th occurrence of word v_i. In our experiments, we simply used this replacement.

We propose the following minimization problem that simultaneously tunes both parameters. We let the parameter $\gamma \in [0, 1]$ be the parameter that tunes the two loss functions, namely, L and R. Note that, as the optimal value of a is different for term L and for term R, we modeled the two terms separately: a_1 and a_2, respectively. Since most of Eq. 4 consists of continuous functions, then Eq. 4 can easily be optimized as a neural classifier using a typical deep learning framework, such as **PyTorch**.

$$\min_{a_1, a_2, \mathbf{w}_l, \mathbf{w}_v} \gamma L(a_1, \mathbf{w}_l, \mathbf{w}_v) + (1 - \gamma) R(a_2, \mathbf{w}_v) \tag{4}$$

For the input, we prepare the vocabulary test results of J learners, the vocabulary feature function $\mathbf{h}$, and the vocabulary ranking r_i. By preparing these data for input, we can train the model through estimating the $\mathbf{w}$ parameters by minimizing $Eq.\ 4$. The tuning of the γ value can be conducted using validation data that are disjointed from both the training and test data. Or, γ can also be tuned by jointly minimizing γ with other parameters in Eq. 4. Finally, in the test phase, using the trained parameter a_1 and $\mathbf{w}_l$ – we can estimate learner l_j's vocabulary size as $a_1 \mathbf{w}_l^\top \mathbf{h}_l(l_j)$. Using the trained parameter $a_2, \mathbf{w}_v$, we can estimate the rank of the

first occurrence of a new word v_i, which did not appear in the training data, as $a_2 \mathbf{w}_v^\top \mathbf{h}_v(u_{i,1})$.

4 Dataset

4.1 Description

To evaluate Eq. 4, we need a real dataset that covers both vocabulary size and reading comprehension tests, assuming that the text coverage hypothesis of 98% holds true. To our knowledge, there is no such dataset widely available. There are certain existing vocabulary test result datasets, such as (Ehara, 2018), as well as many reading comprehension test result datasets - however; we could not find a dataset in which a second-language learner subject is asked to provide both vocabulary size and reading comprehension test results.

To this end, this paper provides such a dataset. Following (Ehara, 2018), we used the Lancers crowdsourcing service to collect 55 vocabulary test results as well as answers to 1 long and 1 short reading comprehension question from 100 learners. We paid around 5 USD for each participant. In comparison to the dataset by (Ehara, 2018), the number of vocabulary test questions was reduced so that subjects would have enough time to solve the reading comprehension test. For the vocabulary test part, we used the Vocabulary Size Test (Beglar and Nation, 2007). The reading comprehension questions were taken from the sample set of the questions in the Appendix section in (Laufer and Ravenhorst-Kalovski, 2010). The correct options for these questions are on a website that can also be reached from the description of (Laufer and Ravenhorst-Kalovski, 2010) [1].

In the same manner as (Ehara, 2018), all participants were required to have ever taken the Test of English for International Communication (TOEIC) test provided by English Testing Services (ETS) and to write scores on a self-report basis. This requirement filters out learners who have never studied English seriously but try to participate for economical merits.

In the dataset, each line describes all the responses from a learner. The first columns, which contain the term TOEIC in their headings, provide TOEIC scores and dates. Then, the 55 vocabulary testing questions follow. The columns that start with "l" denote the responses on the long reading comprehension test and those with "s" denote the responses on the short one.

[1]For more detailed information for the dataset, refer to http://yoehara.com/vocabulary-prediction/.

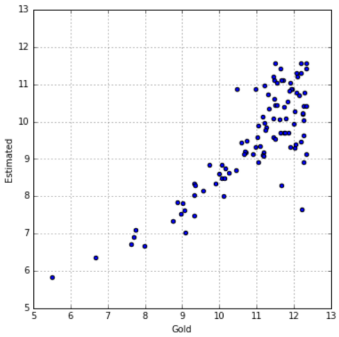

Figure 3: Estimated LFRs against Gold LFRs.

4.2 Preliminary Experiments

Finally, we show preliminary experiments by using our dataset. We used 33 words from the dataset, i.e., 3,300 responses. Hereafter, we simply denote the logarithm of frequency ranks in a descending order as "LFR". For r_i, we used the LFR of the BNC corpus (BNC Consortium, 2007). For features of $\mathbf{h}_v$, we used the logarithm of the frequency of the COCA corpus (Davies, 2009). We obtained parameters by optimizing the minimization parameters Eq. 4. Then, for 100 words *disjoint from the* 33 *training words*, we plotted the estimated LFR values against the gold LFR values in Fig. 3. We can easily see that they have a good correlation. The Spearman's correlation coefficient for Fig. 3 was 0.70, which can be construed as a strong correlation (Taylor, 1990).

5 Conclusions

In this paper, we theoretically showed that previous vocabulary size-based classifiers can be seen as a special case of a neural classifier. We also built a dataset necessary for this evaluation and made it publicly available in the form of an attached dataset. Future work include more detailed experiments on language learners' second language vocabularies.

Acknowledgments

This work was supported by JST ACT-I Grant Number JPMJPR18U8 and JSPS KAKENHI Grant Number JP18K18118. We used the AI Bridging Cloud Infrastructure (ABCI) by the National Institute of Advanced Industrial Science and Technology (AIST), Japan. We thank anonymous reviewers for their insightful and constructive comments.

References

David Beglar. 2010. A Rasch-based validation of the Vocabulary Size Test. *Language Testing*, 27(1):101–118.

David Beglar and Paul Nation. 2007. A vocabulary size test. *The Language Teacher*, 31(7):9–13.

Lisa Beinborn, Torsten Zesch, and Iryna Gurevych. 2014. Predicting the Difficulty of Language Proficiency Tests. *Transactions of the Association for Computational Linguistics*, 2:517–530.

The BNC Consortium. 2007. *The British National Corpus, version 3 (BNC XML Edition).*

Mark Davies. 2009. The 385+ million word corpus of contemporary american english (1990–2008+): Design, architecture, and linguistic insights. *International journal of corpus linguistics*, 14(2):159–190.

Yo Ehara. 2018. Building an English Vocabulary Knowledge Dataset of Japanese English-as-a-Second-Language Learners Using Crowdsourcing. In *Proc. of LREC*.

Yo Ehara. 2019. Neural rasch model: How word embeddings affect to word difficulty? In *Proc. of the 16th International Conference of the Pacific Association for Computational Linguistics (PACLING)*.

Yo Ehara, Yukino Baba, Masao Utiyama, and Eiichiro Sumita. 2016. Assessing Translation Ability through Vocabulary Ability Assessment. In *Proc. of IJCAI*.

Yo Ehara, Yusuke Miyao, Hidekazu Oiwa, Issei Sato, and Hiroshi Nakagawa. 2014. Formalizing Word Sampling for Vocabulary Prediction as Graph-based Active Learning. In *Proc. of EMNLP*, pages 1374–1384.

Yo Ehara, Issei Sato, Hidekazu Oiwa, and Hiroshi Nakagawa. 2012. Mining Words in the Minds of Second Language Learners: Learner-Specific Word Difficulty. In *Proceedings of COLING 2012*, pages 799–814, Mumbai, India. The COLING 2012 Organizing Committee.

Yo Ehara, Issei Sato, Hidekazu Oiwa, and Hiroshi Nakagawa. 2018. Mining words in the minds of second language learners for learner-specific word difficulty. *Journal of Information Processing*, 26:267–275.

Yo Ehara, Nobuyuki Shimizu, Takashi Ninomiya, and Hiroshi Nakagawa. 2013. Personalized Reading Support for Second-language Web Documents. *ACM Trans. Intell. Syst. Technol.*, 4(2):31:1–31:19.

Ayako Hoshino. 2009. *Automatic Question Generation for Language Testing and its Evaluation Criteria*. Ph.D. thesis, Graduate School of Interdisciplinary Information Studies, The University of Tokyo.

Batia Laufer and Geke C. Ravenhorst-Kalovski. 2010. Lexical Threshold Revisited: Lexical Text Coverage, Learners' Vocabulary Size and Reading Comprehension. *Reading in a Foreign Language*, 22(1):15–30.

John Lee and Chak Yan Yeung. 2018. Personalizing lexical simplification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 224–232, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

I. Nation. 2006. How Large a Vocabulary is Needed For Reading and Listening? *Canadian Modern Language Review*, 63(1):59–82.

I. Nation. 2007. Vocabulary size test. `https://www.wgtn.ac.nz/lals/about/staff/paul-nation#vocab-tests`.

Gustavo Paetzold and Lucia Specia. 2016. Collecting and Exploring Everyday Language for Predicting Psycholinguistic Properties of Words. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1669–1679, Osaka, Japan. The COLING 2016 Organizing Committee.

Gustavo Paetzold and Lucia Specia. 2017. Lexical Simplification with Neural Ranking. In *Proc. of EACL*, pages 34–40, Valencia, Spain.

Ellie Pavlick and Chris Callison-Burch. 2016. Simple PPDB: A Paraphrase Database for Simplification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 143–148, Berlin, Germany. Association for Computational Linguistics.

Keisuke Sakaguchi, Yuki Arase, and Mamoru Komachi. 2013. Discriminative Approach to Fill-in-the-Blank Quiz Generation for Language Learners. In *Proc. of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 238–242, Sofia, Bulgaria. Association for Computational Linguistics.

Richard Taylor. 1990. Interpretation of the correlation coefficient: a basic review. *Journal of diagnostic medical sonography*, 6(1):35–39.

Seid Muhie Yimam, Chris Biemann, Shervin Malmasi, Gustavo H. Paetzold, Lucia Specia, Sanja Štajner, Anaïs Tack, and Marcos Zampieri. 2018. A Report on the Complex Word Identification Shared Task 2018. *arXiv:1804.09132 [cs]*. ArXiv: 1804.09132.