Supervised Visualization of Vocabulary Knowledge towards Explainable Support of Second Language Learners

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

Abstract

In second language learning, it is crucial to identify gaps in knowledge of the language between second language learners and native speakers. Such a gap exists even when learning a single word in a second language. As the semantic broadness of a word differs from language to language, language learners must learn how broadly a word can be used in a language. For example, certain languages use different words for "period" in "a period of time" or "period pains" yet both are nouns. Learners whose native languages are such languages typically have only partial knowledge of a word, even though they think they know the word "period," producing a gap between them and native speakers. Language learners typically want explanations for these word usage differences, which even native speakers find it difficult to explain and find it costly to annotate. To support language learners in noticing these challenging differences easily and intuitively, this paper proposes a novel supervised visualization of the usages of a word. In our method, the usages of an inputted word in large corpora written by native speakers are visualized, taking the semantic proximity between the usages into account. Then, for the single inputted word, our method makes a personalized prediction of word usages that each learner may know, based on his/her results of a quick vocabulary test, which takes approximately 30 minutes. The experiment results show that our method produces better usage frequency counts than raw usage frequency counts in predicting vocabulary test responses, implying that word usage prediction is accurate.

Introduction

Acquiring a second language requires repeated efforts to narrow the gap between language learners' knowledge of the language and that of native speakers. Making such gaps intuitively understandable greatly helps language learners self-teach the language and also helps researchers build effective language tutoring systems. Some gaps such as vocabulary size, or time spent in language learning are intuitively easy to understand and, hence, are well studied. However, in second language learning, most gaps are related to meaning and semantics and are inherently abstract. Hence, visualizing these gaps is essential to make these gaps intuitively understandable.

The *broadness* of a word, or how a word can be used in the language to express different concepts, is one such abstract gap (Read 2000). Because the meaning of a word differs from language to language, when learning a word in a second language, there typically exists a gap between what learners think the word means and how the word is actually used in the language. Polysemous words are examples that are easy to understand: "book" can mean an item associating with reading, or it can mean to make a reservation. Other than these examples, to which the part-of-speech tagging techniques in natural language processing (NLP) seem applicable, some examples are more subtle: some languages always use different words for "time" in "in a short time" or "for a time," in which the word "time" refers to a period, and "time and space" or "time heals all wounds," in which "time" is used as an abstract concept. In another example, many languages use different words for "period" in "a period of time", and "period" in "period pains". In this way, the granularity of the word's senses should be distinguished for second language acquisition, as it varies from word to word.

Polysemous words encode different concepts in one word: hence, they have been one of the central topics in knowledge engineering. A substantial amount of work has been conducted to automatically recognize polysemous words for practical applications by using machine learning, including those in the previous AAAI-MAKE workshops (Ramprasad and Maddox 2019; Hinkelmann et al. 2019; Laurenzi et al. 2019). However, even among few such applications for second language acquisition (Heilman et al. 2007; Dias and Moraliyski 2009) in the artificial intelligence (AI) community, the challenging problem of different granularity of the word's senses in second language acquisition has not been addressed. In second language acquisition, as learners are typically not linguistic experts, i.e., novices, hence, systems to support their learning need to be intuitively understandable. Our goal is to make the gaps among word usages intuitively understandable, even for novice language learners.

To this end, this paper proposes a novel supervised visualization method for word usages to assist in learning the different usages of a word. Our method first searches all usages of the target word in a large corpus written by native speakers. Then, it calculates the vector representation of each usage, or occurrence, of each word by using a contextualized word embedding method (Devlin et al. 2019). Contextualized word embedding methods (Peters et al. 2018; Devlin et al. 2019) are recently proposed methods to embed each occurrence of a word, capturing the context of each usage of the word.

Then, our method is *trained* to visualize the contextualized word embedding vectors by plotting each usage as a point in a two-dimensional space. Unlike a typical visualization method that merely projects the vectors to a two-dimensional space, our method is trained to fit and visually explain a given supervision dataset. This means that the same vectors are visualized in different ways if the supervision dataset differs. Here, the supervisions are a vocabulary test result dataset that consists of a matrix-format data, recording which learner answered correctly/incorrectly to which word question. The method visualizes the areas a learner user may know by classifying each usage point in the visualization into known/not known to the learner. This classification is conducted in a personalized manner because learners' language skills and specialized fields are different. The learner only needs to take a 30-minute vocabulary test for this purpose.

Figure 1 shows an example visualization using our method. "To haunt" has two different meanings in English, the first being "to chase" and the other "to curse," or to be affected by ghosts or misfortune. Each point shows the usage of the word in a corpus written by native speakers. The differences in point colors indicate whether they are predicted to be known to the learner. The right side of the figure, within the dotted curve, is predicted to be known to the learner. In this way, our method visualizes the semantic area the learner knows.

Our contribution is as follows:

- For second language vocabulary learning, we propose a novel supervised visualization model that captures word broadness via a personalized prediction of learner's knowledge of usages.

- As our visualization uses a vocabulary test result dataset as supervisions, learners can understand which usage of the inputted word is predicted to be known/not known to him/her. Unlike previous methods that output automatic explanation of machine-learning models, our method is much more intuitive and novice-friendly for language learners in the sense that language learners do not need to know about machine learning models.

- We evaluated our method in terms of predictive accuracy of vocabulary test result dataset and achieved better results compared to baselines.

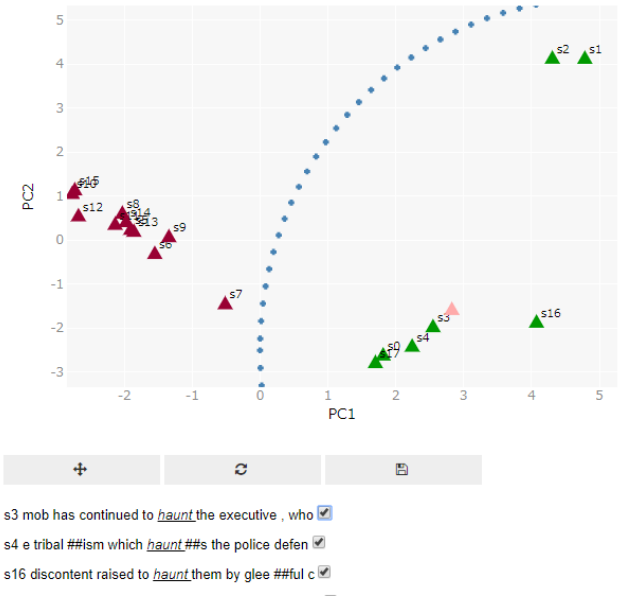

Figure 1: Usage of "haunt" predicted to be familiar to the learner.

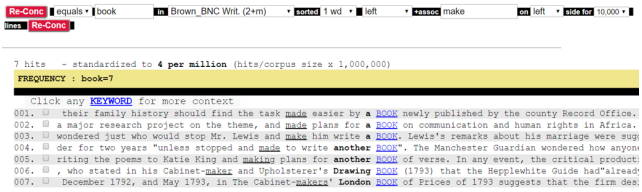

Figure 2: An example of concordancer.

Related Work

Explainable machine learning studies

While deep learning-based methods outperformed conventional machine learning methods such as support vector machines (SVMs) in many tasks, parameters of deep learning methods are typically more difficult to interpret compared to those in conventional models. To this end, in the machine learning and artificial intelligence community, a number of methods have been proposed to extract explanations from trained machine-learning models, or training models taking explainability into accounts (Ribeiro, Singh, and Guestrin 2016; Koh and Liang 2017; Lundberg and Lee 2017; Ribeiro, Singh, and Guestrin 2018).

However, the purpose of these methods is to explain machine-learning models to help machine-learning engineers and researchers in understanding the models. Obviously, second language learners are usually not machine-learning engineers and researchers. Therefore, methods of these studies have different purposes, and it is difficult to apply these methods to help their understanding of the models. Language learners are typically even not interested in the models. Rather, learners' interests reside in understanding their current learning status and what they should learn

to improve it. Hence, to meet learners' needs, a model is desirable for a learner to see his/her current learning status and what he/she needs to learn in the near future.

Word Embedding Visualization Studies

Word embedding techniques are techniques that have been extensively studied in natural language processing (NLP) to obtain vector representations of words typically using neural networks. The word2vec is a seminal paper in these lines of studies (Mikolov et al. 2013). The following papers report improvement of their accurateness to represent words as vectors, typically by comparing the distances between word vectors with human judgments on semantic proximity between words (Pennington, Socher, and Manning 2014). Early studies on word embeddings address how to make one vector for each word. As one vector representation is modeled to point one meaning, this limitation is obviously problematic to deal with polysemous words. Several previous studies tackled this problem and proposed methods to estimate the number of a word's meanings and to estimate an embedding for each meaning of the word (Athiwaratkun, Wilson, and Anandkumar 2018). However, recently, contextualized word embeddings (Peters et al. 2018; Devlin et al. 2019) became quickly popular. With these methods, we can obtain an embedding for each *usage*, or occurrence, of a word, considering the context of the occurrence of the word in a running sentence. These methods can also be seen as a method to estimate word embeddings for polysemous words, with an extreme assumption that each occurrence of a word has different meanings. As contextualized word embeddings are shown to be successful in many tasks, in current NLP, the former strategy to estimate both the number of meanings of a word and an embedding for each meaning is employed only when it is necessary.

Following the rise of word embedding techniques, visualization studies were propose to visualize word embeddings. The study by (Smilkov et al. 2016) simply reported that their development of a tool to visualize embeddings for different words. The study by (Liu et al. 2017) introduces applying visualization of word embeddings to analyze semantic relationships between words. Both paper deals with principal component analysis (PCA) and t-SNE (Maaten and Hinton 2008) for visualization. To our knowledge, we are the first to visualize contextualized word embeddings, in which each *occurrence* of a word, rather than a word, is visualized, with a practical purpose on language education.

In addition to the visualization, our method can also predict the usages that each learner is familiar/unfamiliar with, in a personalized manner, when vocabulary test result data of dozens of learners are provided, such as the data in (Ehara 2018). While there exist previous studies (Ehara 2018; Lee and Yeung 2018; Yeung and Lee 2018) for predicting the *words* that each learner is familiar/unfamiliar with using such data by using simple machine-learning classification, our method tackles a more difficult problem that deals with predicting which *usages* of a word is known/unknown to the learner.

Concordancer studies

While our proposed method is novel as a visualization, software tools that search the usages of an inputted word for educational purposes and display them itself are not novel: such software is known as *concordancers*. Concordancers target learners, educators, and linguists as primary users. They are interactive software tools that retrieve all usages of the inputted word in a large corpus and display the list of the usages, each of which comes with the surrounding word patterns (Hockey and Martin 1987). Concordancers were also studied to support translators, who are second language learners in many cases (Wu et al. 2004; Jian, Chang, and Chang 2004; Lux-Pogodalla, Besagni, and Fort 2010).

Figure 2 shows a screenshot from a current concordancer [1]. In this screenshot, the word "book" is searched. Then, the list of word usages is shown. Each word usage comes with surrounding words so that language learners can see how the word is used. While the list is sorted in alphabetical order of the previous word, we can see that the list shows "a book" and "the book" in totally different positions and are not helpful for language learners. While some concordancers support listing the usage of "book" as nouns by attaching texts with part-of-speeches in advance, this is not helpful to see the different usages of the word when the part-of-speeches of the usages are identical. For example, the word "bank" have polysemous meanings sharing the same part-of-speech: one as financial organizations, and another as embankments.

Personalized complex word identification studies

In this study, a part of our goals is to identify complex *usages* of a word in a running text. In other words, for one word, one usage of the word in running text is complex for a learner, and another usage of the word is not. There are previous studies that identify complex *words* in a personalized manner in the NLP literature (Ehara et al. 2012; Lee and Yeung 2018). These studies predict the words that each learner knows based on each learner's result of a short vocabulary test, which a learner typically takes 30 minutes to solve. Also, there are also many studies that identify complex *usages* in a *non*-personalized manner, as summarized in (Paetzold and Specia 2016; Yimam et al. 2018).

However, to our knowledge, the task of identifying complex *usages* in a *personalized* manner is novel. Our method is also novel in that it trains how to visualize the usages so that learners can visually understand the usage differences by using the learners' vocabulary test data.

Preliminary System and Experiments

Before entering the technical details of our method described in the Proposed Method section, we first show the preliminary system and some experiment results to introduce the motivation of the proposed method.

The preliminary system visualizes contextualized word embeddings by using the conventional visualization of principal component analysis (PCA). Figure 3 shows the layout

[1] https://lextutor.ca/conc/eng/

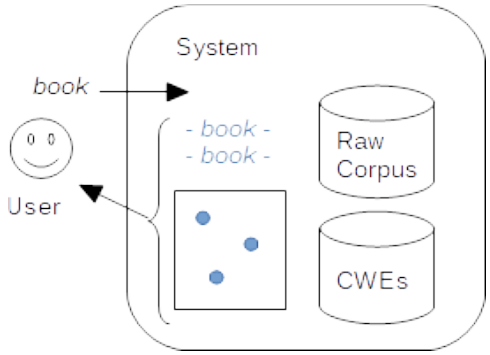

Figure 3: System layout. CWE means contextualized word embeddings.

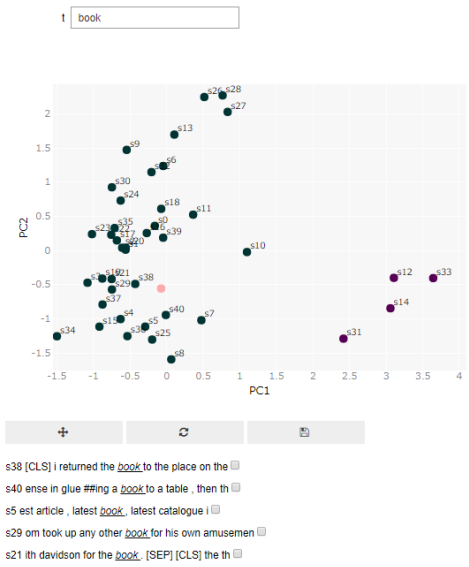

Figure 4: Example of searching the word *book*.

of the preliminary system. Once a user provides a word to the system, it automatically searches the word in the corpus in a similar way to typical concordancers. Unlike concordancers, the system has a database that stores contextualized word embeddings for each *usage* or occurrence of each word in the corpus. We used half a million sentences from the British National Corpus (BNC Consortium 2007) as the raw corpus. We built the database by applying the **bert-base-uncased** model of the PyTorch Pretrained the BERT project [2](Devlin et al. 2019) to the corpus. We used the last layer, which was more distant from the surface input, as the embeddings.

Choice of dimension reduction methods

Principal component analysis (PCA) and t-SNE (Maaten and Hinton 2008) are famous dimension reduction methods, and t-SNE is notable for its intuitiveness and well clustered

[2]https://github.com/huggingface/pytorch-pretrained-BERT

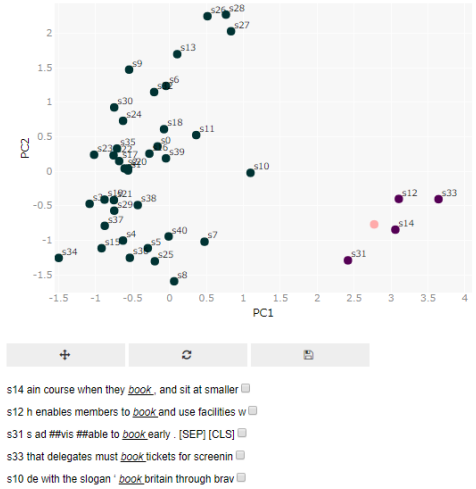

Figure 5: Another example of searching the word *book*.

points (Maaten and Hinton 2008).

Knowing t-SNE, we did not employ t-SNE for visualization for the following reasons: First, in our visualization, the distances between usage points are important. While t-SNE often produces intuitive clusters between data points, the distance between points in the visualization is complicated compared to those of PCA. Hence, to interpret distances between points, PCA is This is stated in the original t-SNE (Maaten and Hinton 2008) paper. Moreover, many blog posts such as [3] for engineers address this fact to encourage the proper understanding of t-SNE. For these reasons, we employed PCA for the basis of our visualization.

Second, even if the data to visualize is fixed, t-SNE returns different results depending on its hyperparameter called **perplexity**. In contrast, PCA returns the same results if the data to visualize is fixed. This dependence on the hyperparameter is elaborated in the original t-SNE paper (Maaten and Hinton 2008) in the first place. We can also find some blog posts targeting engineers that advocates to carefully set the perplexity parameter such as [4]. Various results on fixed data can be useful when the data is difficult to be pre-processed so that the following dimension-reduction methods are easy to handle. However, in this study, the data to be visualized are embeddings vectors; hence, the data can be easily pre-processed before we feed them into the data. Hence, for the purpose of this study, the feature that the results vary on fixed data is unlikely to be useful. Rather, this may possibly complicate the interpretation of the visualization.

Third, practically, t-SNE is computationally heavy compared to PCA. Computing a t-SNE visualization involves calculations for every pair of the given data points. While how to deal with this heavy computational complexity is addressed in studies such as (Tang et al. 2016), practically, t-

[3]https://mlexplained.com/2018/09/14/paper-dissected-visualizing-data-using-t-sne-explained/

[4]https://distill.pub/2016/misread-tsne/

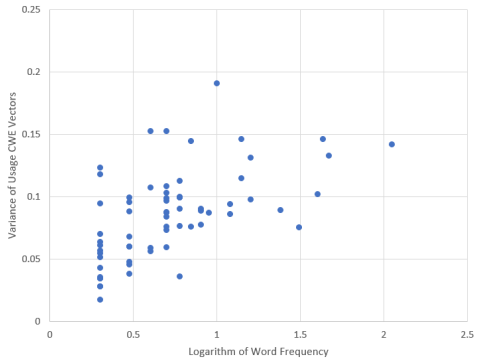

Figure 6: Variance of usage vectors vs. log word frequency.

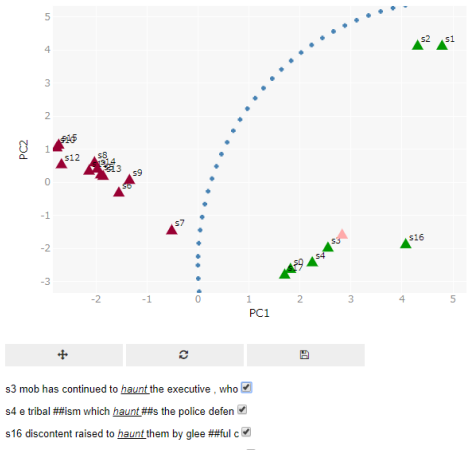

Figure 7: Recap: usage of "haunt" predicted to be known to the learner.

SNE is usually computationally heavy when compared to PCA. Strictly speaking, PCA has a similar complexity as it involves the computation of singular values and vectors in singular value decomposition (SVD). However, the calculation of SVD has a number of applications other than PCA-based visualization, sophisticated calculation methods for large data were previously proposed (Halko et al. 2011).

Preliminary System by using PCA

We built a preliminary system and conducted some experiments to see how contextualized word embedding vectors are plotted in the system. Figure 4 depicts such an example of searching for the word *book*. Users can directly type the word in the textbox shown at the top of Figure 4. Below is the visualization of the usages found and their list. Each dark-colored point is linked to each usage. Two dark colors are used to color each usage point according to the results of a Gaussian mixture model (GMM) clustering with 2 components, as this value was reported to work well (Athiwaratkun, Wilson, and Anandkumar 2018). The light-red colored point is the *probe point*: the usages are listed in the nearest order of the probe point. No usage is linked to the probe point. Users can freely and interactively drag and move the probe point to change the list of usages below the visualization. Each line of the list shows the usage identification number and the surrounding words of the usage, followed by a checkbox to record the usage so that learners can refer to it later. In Figure 4, the probe point is on the left part of the visualized figure. In the first several lines of the list, the system successfully shows the usages of the word *book* as a publication. In contrast, Figure 5 depicts the case in which the users drag the probe point from the left to the right of the visualization. The first several lines of the list show the usages of the word *book*, which means to *reserve*. We can see that the words surrounding the word *book* vary: merely focusing on the surrounding words, such as "to" before *book*, cannot distinguish the usages of *book*, which means to *reserve*, from the usages of *book* for reading.

Clustering Results

The GMM clustering was accurate but not perfect: 0 errors in the 42 usages of "book", 1 error in the 22 usages of "bank", when manually checked in the excerpt. Hence, learners can choose not to use this, as in the video. Figure 6 shows the variance of the usage vectors of each word against its log frequency in the excerpt. It showed a statistically significant moderate correlation ($r = 0.56, p < 0.01$ by F-test), implying that frequent words tend to have complex usages.

Motivating Examples

From the example of "book" in the previous sections, we can easily see that the usages of "book" about reading are more frequent than those of "book" about a reservation. Hence, when counting the number of usages, it is intuitive to assume that learners are not familiar with all usages but are familiar with the usages *within a certain radius* in the vector space. This is the motivation of our method descried in the next section.

Before entering the technical details of our visualization method in the next section, we show some usage prediction result examples of our method in a manner similar to the previous examples of "book" so that readers can intuitively understand our motivation, as shown in Figure 7 and Figure 8. The markers are changed to triangular to denote that the colors reflect prediction results, rather than the GMM-based clustering results explained above. The coloring and darkness of the points in the visualization follow those of the previous examples; the red light-colored point is the probe point, and the other dark points denote usages. Figure 7 shows an example of the familiar usage prediction in case of searching the word "haunt". The right-hand side of the cross-marked circle is the area in which usages are predicted to be familiar to this learner. The probe point is located within the circle. We can see that the usages of "haunt" about chasing are listed below. Figure 8 shows another example of "haunt". As the probe point is located outside of the circle, in the left side of the visualization, the list below shows the list of the usages predicted to be unfamiliar to this learner. We can see that"haunt" about "to curse" are mainly listed.

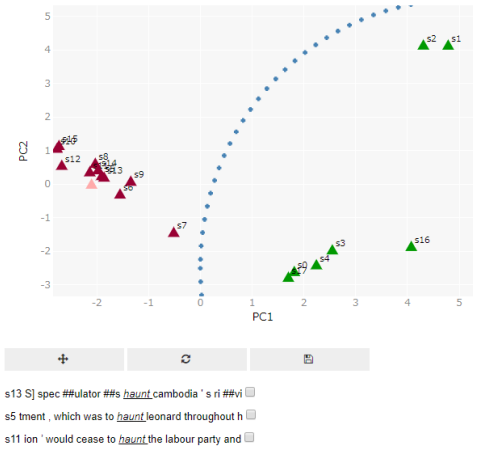

Figure 8: Usage of "haunt" predicted not to be known to the learner.

Proposed Method

As stated in the Related Work section, some previous studies address methods to predict the words that a learner knows based on his/her short vocabulary test result. However, since our application requires personalized prediction of the *usages* of the word that the learner does not know. Hence, we propose a novel model that does this.

Let us write the set of words as $\{v_1, \ldots, v_I\}$, where I is the number of words (in type), and write the set of learners as $\{l_1, \ldots, l_J\}$, where J is the number of learners. Then, in previous studies, based on the Rasch model (Rasch 1960; Baker 2004), the following logistic regression model Equation 1 is used to predict whether learner l_j knows word v_i or not. Here, $\sigma(x) := \frac{1}{1+\exp(-x)}$ and $y_{i,j}$ is the response of the learner in the vocabulary test; $y_{i,j} = 1$ if learner l_j answered correctly to the question of word v_i, and $y_{i,j} = 0$ otherwise. We have two types of parameters to tune: a_{l_j} is the *ability* of learner l_j and d_{v_i} is the *difficulty* of word v_i.

$$P(y_{i,j} = 1 | l_j, v_i) = \sigma(a_{l_j} - d_{v_i}) \quad (1)$$

Here, how to model d_{v_i}, or the difficulty parameter of word v_i, is the key to our purpose. Previous studies report that the negative logarithm of the word frequency correlates well with the perceived difficulty of words (Tamayo 1987; Beglar 2010). As in Figure 1, our key idea is to count the frequency of word usages only within a certain distance from the typical usage of the word. Hence, we propose the following model to implement this idea.

For each v_i, we have n_i vectors that are vector representation of each of the n_i occurrences of word v_i. We write these vectors as $X_i = \{\vec{x}_{1,i}, \ldots, \vec{x}_{n_i,i}\}$. Each vector $\vec{x}_{k,i}$ is T_1 dimensional. Among X_i, let $\vec{c}_i$ be the one closest to their center $\frac{1}{n_i}\sum_{k=1}^{n_i} \vec{x}_k$. Let freq$(v_i)$ be the frequency of the vectors in X_i within distance ϵ measured from the central vector $\vec{c}_i$. We write this frequency simply as freq$(v_i) = N(\vec{c}_i, \epsilon, X_i)$. Here, n is the number of usages of word v_i and let each $\vec{x}_k$ be each usage vector obtained from contextualized word embedding methods. Let $\mathrm{ReLU}(z) = \max(0, z)$ be the rectified linear unit function, and M be a large positive constant, such as 100. Let $\mathbf{G}$ be a linear projection matrix from a T_1 dimensional space to a T_2 dimensional space. Let $d_e(\vec{a}, \vec{b})$ be the Euclidean distance between two vectors. By using these formulations, we modeled the difficulty of words as follows:

$$d_{v_i} = -\log(\text{freq}(v_i) + 1) \quad (2)$$

$$\text{freq}(v_i) = N(\vec{c}_i, \epsilon, X_i) \quad (3)$$

$$\approx \sum_{k=1}^{n_i} \tanh\left(M \cdot \mathrm{ReLU}(\epsilon - d_e(\mathbf{G}\vec{c}_i, \mathbf{G}\vec{x}_{k,i}))\right) \quad (4)$$

The tricky part is that Equation 3 can be approximately written as Equation 4, whose parameter can be easily tuned and optimized by using neural machine learning framework such as **PyTorch**. In Equation 4, due to the ReLU function, negative values within the function is simply ignored. Hence, as d_e is the Euclidean distance, if $\epsilon = 0$, i.e., the size of the circle is 0, the terms inside ReLU is negative, and freq$(v_i) = 0$. If $\epsilon - d_e(\mathbf{G}\vec{c}_i, \mathbf{G}\vec{x}_{k,i}) > 0$, due to M and $\tanh$, the resulting value is almost 1. This means that we are counting only the cases that ϵ surpasses d_e, i.e., counting the usages within ϵ measured from $\vec{c}_i$.

Notably, the following characteristics are important to understand our model.

Not merely a logistic regression

Notably, the proposed model is *not* merely a logistic regression. Our model has more parameters such as $\epsilon, \vec{c}_i, a_{l_j}, \mathbf{G}$. Because of having different extra parameters compared to the logistic regression, to train our model, we typically need to use a neural network machine learning framework to model and optimize, such as **PyTorch**. To optimize using such models, as it is difficult to differentiate the loss functions of such models by hand, the model loss function is desirable to be mostly continuous and smooth so that its parameters can be tuned using auto-gradient. We specifically designed Equation 4 to meet these conditions. In the experiments, we used the Adam optimization method (Kingma and Ba 2015) to optimize the loss function.

Trainable G

As Equation 4 is mostly continuous and smooth, matrix $\mathbf{G}$ can also be trained by using deep-learning framework software. As $\mathbf{G}$ is a projection matrix from T_1 to T_2, if we set $T_2 = 2$ to consider a projection to a two-dimensional space, training $\mathbf{G}$ via supervisions means training visualization via supervisions. Here, in our task setting, the supervisions are vocabulary test dataset of second language learners, i.e., a matrix in which the (j, i)-th element denotes whether learner l_j correctly answered the question of word v_i.

ϵ_j: Personalized ϵ

In Equation 4, for easier understanding, we write ϵ to be a constant that does not depend on learner index j. In reality, we can *personalize* ϵ by making ϵ dependent to learner index j as ϵ_j; in this case, each learner l_j has his/her own region that he/she can understand, and the radius of this region is

ϵ_j. This personalized version is the one that we used in the experiments.

Experiments

Quantitative Results of Prediction

Quantitative evaluation of this personalized prediction of usages of a word is difficult; to this end, we need to test each learner multiple times for different usages of the same word. However, when tested with the same word multiple times, learners easily notice that the word has multiple meanings. Hence, instead, we evaluated the accuracy of personalized prediction of the words that the learner knows under an experiment setting similar to (Ehara 2018). Our proposed method is based on neural classification with a novel extension to adjusted counting the frequency of the usages within distance ϵ_j. Since a typical logistic-regression classifier is identical to one-layer neural classifier, comparing our model with a typical logistic-regression classifier using a frequency feature in terms of accuracy can be used to indirectly evaluate how the idea of adjusted frequency is a practical method for evaluation.

The proposed model estimates the number of occurrences, i.e., usages, that each learner knows. In other words, this can be regarded as modifying the word frequency so that the model fits to the given vocabulary test dataset. In this regard, we can evaluate how well the proposed model can correct word frequency when an unbalanced corpus is given. Each document in the British National Corpus (BNC) (BNC Consortium 2007) is annotated with a domain Table 1. We evaluated how the proposed model can correct the word frequency in the "arts" domain.

We used the vocabulary test result data in which each of 100 learners answered 31 vocabulary questions on the publicly available dataset (Ehara 2018). In $3,100$ vocabulary test responses, we used $1,800$ to train the model, and the rest was used for the test. The baseline model is simply a logistic regression in which the logarithm of word frequency is the only feature. The logarithm of word frequency has been used as a simple rough measure for word difficulty and previously used to analyze and predict word difficulty based on vocabulary test data (Beglar 2010; Ehara et al. 2013; Lee and Yeung 2018; Yeung and Lee 2018). The proposed model counts only the number of usages within the radius ϵ_j. We used the PyTorch neural network framework to automatically tune the radius ϵ_j and the center of the sphere by using its powerful automatic gradient support (Paszke et al. 2017).

First, we perform experiments on $T_1 = T_2$ and $\mathbf{G} = \mathbf{I}$, a setting where no projection is performed and the model deals with T_1 dimensional hyperspheres. Table 2 shows the results. It can be seen that the accuracy of the prediction of the word test data of language learners using the biased text of arts domain only is lower than that using the word frequency of all domains. The proposed method was able to improve the accuracy of the word frequency of the arts domain only by counting the frequency in the region on the contextual word expression vector space where the examinee is estimated to be reacting. This effect was also observed for all domains. This seems to be the effect of frequency counting excluding the cases where the proposed method is outlier. The improvement in accuracy before and after correction ($p < 0.01$, Wilcoxon test) was statistically significant when modifying word frequencies in the arts domain alone or in all domains.

Table 1: Number of sentences in each domain of the BNC corpus in the total of $100,000$ sentences.

imaginative	21,946
arts	18,289
natural sciences	5,256
social science	7,777
commerce	4,378
leisure	20,300
belief and thought	3,441
world news	764
applied science	2,625
world affairs	15,224

Table 2: Accuracy of predicting learners' vocabulary test responses by using the raw freqiencies and the corrected frequencies by the proposed model in each domain.

Domain	Correction	Accuracy
Arts	Raw	0.61
Arts	Corrected	**0.64**
All domains	Raw	0.67
All domains	Corrected	**0.72**

"Trained" visualization

In the above experiments, we considered the case where no projection was conducted, by fixing $\mathbf{G} = I$. Next, let us consider the case where $\mathbf{G}$ is a projection to a two-dimensional space, i.e., $\mathbf{G}$ is a $2 \times T_1$ matrix. Tuning $\mathbf{G}$ and radius ϵ_j to fit the vocabulary test dataset by using Equation 4 means that we can actually *train* the visualization to explain the vocabulary test dataset in a supervised manner.

Figure 9 and Figure 10 show the result of the visualization. The initial value of $\mathbf{G}$ was set to a two-dimensional projection matrix by principal component analysis (PCA). Though the initial value is the projection by PCA, it should be noted that the projection matrix $\mathbf{G}$ itself is trained from the vocabulary test dataset as well as the radius ϵ_j.

From Figure 9 and Figure 10, we can see that the proposed method counts only the main meanings within the red circle. To qualitatively evaluate the results, in Table 3, the two farthest or closes example occurrences of "period" from its center point, i.e., the center of the red circle in Figure 9 are shown. It can be seen that the farthest cases are examples of the use of technical terms such as "period pain" and "magnetic field period", while the closest two cases are examples of nouns representing periods such as "this period" and "the period".

Table 3: Farthest (F.) and closest (C.) two occurrences of "period" from the center of the circle in Figure 9.

F.	period pains can be severe and disruptive.
F.	to produce a slight spread of magnetic field period .
C.	design during this period was in the plan .
C.	the pub designer of the period ,

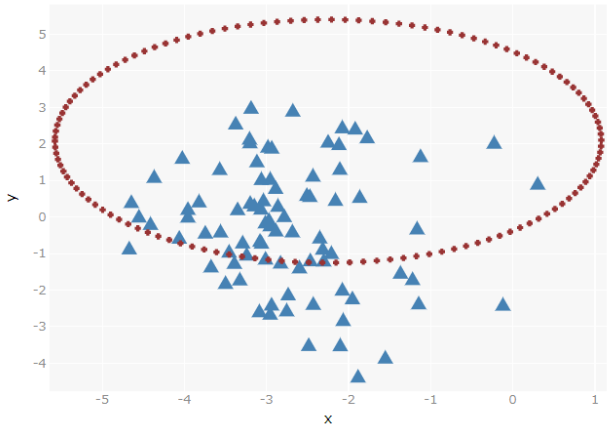

Figure 9: Trained visualization example of "period". Each triangle point represents an occurrence, or a usage, of the word "period" in the "arts" domain in the BNC corpus. The entire projection of the original contextualized word embedding vectors to the two-dimensional space, namely $\mathbf{G}$, and the radius ϵ_j was *optimized* to fit the vocabulary test dataset (ref. Equation 1, Equation 3, and Equation 4). Intuitively, a large ϵ_j denotes that learner l_j has a high language ability as he/she is estimated to understand many of the occurrences of the word "period" within the red circle.

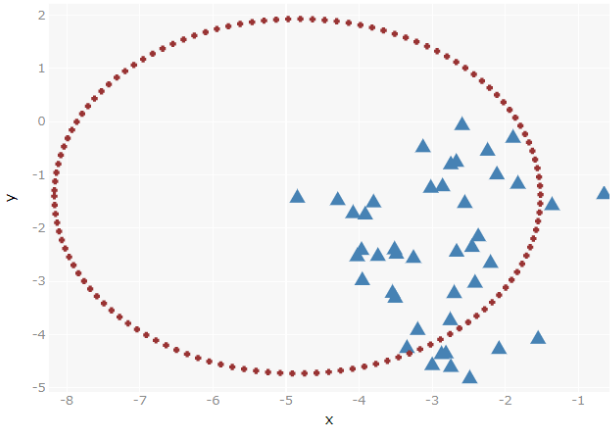

Figure 10: Trained visualization example of "figure". The setting of the training is identical to that of Figure 9.

Conclusions

In this paper, we propose a supervised visualization method to predict which usages of a word are known to each learner, by using a vocabulary test result dataset as supervisions. Our neural method automatically tunes the projection matrix to visualize and the radius of each learner in the visualization so that the counted frequency within the circles fits to the supervisions. Experiments on actual subject response data show that the proposed method can predict subject response more accurately by modifying the frequency even when the use cases are biased to a specific domain. As a future work, we are planning to make our method more interactive.

References

Athiwaratkun, B.; Wilson, A.; and Anandkumar, A. 2018. Probabilistic FastText for multi-sense word embeddings. In *Proc. of ACL*.

Baker, F. B. 2004. *Item Response Theory : Parameter Estimation Techniques, Second Edition*. CRC Press.

Beglar, D. 2010. A rasch-based validation of the vocabulary size test. *Language Testing* 27(1):101–118.

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

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of NAACL*.

Dias, G., and Moraliyski, R. 2009. Relieving polysemy problem for synonymy detection. In *Portuguese Conference on Artificial Intelligence*, 610–621. Springer.

Ehara, Y.; Sato, I.; Oiwa, H.; and Nakagawa, H. 2012. Mining words in the minds of second language learners: learner-specific word difficulty. In *Proc. of COLING*.

Ehara, Y.; Shimizu, N.; Ninomiya, T.; and Nakagawa, H. 2013. Personalized reading support for second-language web documents. *ACM Transactions on Intelligent Systems and Technology* 4(2).

Ehara, Y. 2018. Building an english vocabulary knowledge dataset of japanese english-as-a-second-language learners using crowdsourcing. In *Proc. LREC*.

Halko, N.; Martinsson, P.-G.; Shkolnisky, Y.; and Tygert, M. 2011. An algorithm for the principal component analysis of large data sets. *SIAM Journal on Scientific computing* 33(5):2580–2594.

Heilman, M.; Collins-Thompson, K.; Callan, J.; and Eskenazi, M. 2007. Combining Lexical and Grammatical Features to Improve Readability Measures for First and Second Language Texts. In *Proc. of NAACL*, 460–467. Rochester, New York: Association for Computational Linguistics.

Hinkelmann, K.; Blaser, M.; Faust, O.; Horst, A.; and Mehli, C. 2019. Virtual Bartender: A Dialog System Combining Data-Driven and Knowledge-Based Recommendation. In *AAAI Spring Symposium: Combining Machine Learning with Knowledge Engineering*.

Hockey, S., and Martin, J. 1987. The Oxford Concordance Program Version 2. *Digital Scholarship in the Humanities* 2(2):125–131.

Jian, J.-Y.; Chang, Y.-C.; and Chang, J. S. 2004. TANGO: Bilingual collocational concordancer. In *Proc. of ACL demo.*, 166–169.

Kingma, D. P., and Ba, J. 2015. Adam: A method for stochastic optimization. In *Proc. of ICLR*.

Koh, P. W., and Liang, P. 2017. Understanding Black-box Predictions via Influence Functions. In *Proc. of ICML*, 1885–1894.

Laurenzi, E.; Hinkelmann, K.; Jüngling, S.; Montecchiari, D.; Pande, C.; and Martin, A. 2019. Towards an Assistive and Pattern Learning-driven Process Modeling Approach. In *AAAI Spring Symposium: Combining Machine Learning with Knowledge Engineering*.

Lee, J., and Yeung, C. Y. 2018. Personalizing lexical simplification. In *Proc. of COLING*, 224–232.

Liu, S.; Bremer, P.-T.; Thiagarajan, J. J.; Srikumar, V.; Wang, B.; Livnat, Y.; and Pascucci, V. 2017. Visual exploration of semantic relationships in neural word embeddings. *IEEE trans. on vis. and comp. g.* 24(1):553–562.

Lundberg, S. M., and Lee, S.-I. 2017. A Unified Approach to Interpreting Model Predictions. In Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Proc. of NIPS*, 4765–4774.

Lux-Pogodalla, V.; Besagni, D.; and Fort, K. 2010. FastKwic, an "intelligent" concordancer using FASTR. In *Proc. of LREC*.

Maaten, L. v. d., and Hinton, G. 2008. Visualizing data using t-sne. *Journal of machine learning research* 9(Nov):2579–2605.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In *Proc. of NIPS*, 3111–3119.

Paetzold, G., and Specia, L. 2016. Benchmarking lexical simplification systems. In *LREC*.

Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; and Lerer, A. 2017. Automatic differentiation in pytorch.

Pennington, J.; Socher, R.; and Manning, C. 2014. Glove: Global vectors for word representation. In *Proc. of EMNLP*, 1532–1543.

Peters, M. E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; and Zettlemoyer, L. 2018. Deep contextualized word representations. *Proc. of NAACL*.

Ramprasad, S., and Maddox, J. 2019. CoKE: Word Sense Induction Using Contextualized Knowledge Embeddings. In *AAAI Spring Symposium: Combining Machine Learning with Knowledge Engineering*.

Rasch, G. 1960. *Probabilistic Models for Some Intelligence and Attainment Tests*. Copenhagen: Danish Institute for Educational Research.

Read, J. 2000. *Assessing Vocabulary*. Cambridge University Press.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proc. of KDD*, 1135–1144. ACM. event-place: San Francisco, California, USA.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2018. Anchors: High-Precision Model-Agnostic Explanations. In *Proc. of AAAI*.

Smilkov, D.; Thorat, N.; Nicholson, C.; Reif, E.; Viégas, F. B.; and Wattenberg, M. 2016. Embedding Projector: Interactive Visualization and Interpretation of Embeddings. In *In Proc. of NIPS 2016 Workshop on Interpretable Machine Learning in Complex Systems*.

Tamayo, J. M. 1987. Frequency of use as a measure of word difficulty in bilingual vocabulary test construction and translation. *Educational and Psychological Measurement* 47(4):893–902.

Tang, J.; Liu, J.; Zhang, M.; and Mei, Q. 2016. Visualizing large-scale and high-dimensional data. In *Proc. of WWW*, 287–297.

Wu, J.-C.; Chuang, T. C.; Shei, W.-C.; and Chang, J. S. 2004. Subsentential translation memory for computer assisted writing and translation. In *Proc. of ACL demo.*, 106–109.

Yeung, C. Y., and Lee, J. 2018. Personalized text retrieval for learners of chinese as a foreign language. In *Proc. of COLING*, 3448–3455.

Yimam, S. M.; Biemann, C.; Malmasi, S.; Paetzold, G. H.; Specia, L.; Štajner, S.; Tack, A.; and Zampieri, M. 2018. A report on the complex word identification shared task 2018. In *Proc. of BEA*.