Understanding and Supporting Vocabulary Learners via Machine Learning on Behavioral and Linguistic Data

by

Sungjin Nam

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Information)
in The University of Michigan
2020

Doctoral Committee:

Associate Professor Kevyn Collins-Thompson, Chair
Associate Professor Ryan Baker, University of Pennsylvania
Assistant Professor David Jurgens
Professor Rada Mihalcea

Sungjin Nam

sjnam@umich.edu

ORCID iD: 0000-0002-1893-4878

Dedication

To my family, and to my wife Hyeji Seo.

Acknowledgements

I want to thank many people who have worked with me and helped me during the doctoral program. I thank my committee members and close collaborators. My advisor, Kevyn Collins-Thompson, always provided his keen insights on problems and showed incredible patience when I explore research questions. David Jurgens shared his valuable insights on NLP problems in our regular meetings and my random visits to his office. Gwen Frishkoff guided me on how to think like an experimental psychologist. Invaluable comments and feedback for the dissertation from Rada Mihalcea and Ryan Baker helped me to write a better thesis.

During my study in the U.S., I worked with collaborators from various organizations, including researchers at Adobe Research (all names are in alphabetical order – Zoya Bylinskii, Rajiv Jain, Christopher Tensmeyer, Tong Sun, and Curtis Wigington), Echo 360 (Perry Samson), University of Michigan (Christopher Brooks, Steve Lonn, and Stephanie Teasley), University of Kentucky (Joseph Waddington), and Seoul National University (Joonhwan Lee and Jonghwan Oh). Although the studies I conducted with them are not included in this dissertation, these industry and academic research experience widened my view on diverse research topics.

I want to thank my friends I met during the study (Tawfiq Ammari, Ryan Burton, Yan Chen, Heeryung Choi, Joonyoung Chung, Jane Im, Jamin Koo, Jaseok Lee, Souneil Park, Woosuk Seo, Hari Subramonyam, Rohail Syed, Jungwon Yang, Rayoung Yang, and Sangseok You). Their friendship made me feel I was not alone for this journey.

And most importantly, I want to acknowledge endless support from my family and my wife. I could not complete this dissertation without them.

Throughout the studies included in this dissertation, I have received support and assistance from the following funding sources. The first and second studies of this dissertation were supported by the Institute of Education Sciences, U.S. Department of Education, through Grant R305A140647 to the University of Michigan. The third and fourth studies were supported in part by the Michigan Institute for Data Science, and by the Institute of Education Sciences, U.S. Department of Education, through Grant R305A140647 to the University of Michigan. The opinions expressed are those of the authors and do not represent views of the Institute or the U.S. Department of Education or the Michigan Institute for Data Science.

Table of Contents

List of Figures

List of Tables

List of Appendices

Abstract

This dissertation presents various machine learning applications for predicting different cognitive states of students while they are using a vocabulary tutoring system, DSCoVAR. We conduct four studies, each of which includes a comprehensive analysis of behavioral and linguistic data and provides data-driven evidence for designing personalized features for the system.

The first study presents how behavioral and linguistic interactions from the vocabulary tutoring system can be used to predict students' off-task states. The study identifies which predictive features from interaction signals are more important and examines different types of off-task behaviors. The second study investigates how to automatically evaluate students' partial word knowledge from open-ended responses to definition questions. We present a technique that augments modern word-embedding techniques with a classic semantic differential scaling method from cognitive psychology. We then use this interpretable semantic scale method for predicting students' short- and long-term learning.

The third and fourth studies show how to develop a model that can generate more efficient training curricula for both human and machine vocabulary learners.

The third study illustrates a deep-learning model to score sentences for a contextual vocabulary learning curriculum. We use pre-trained language models, such as ELMo or BERT, and an additional attention layer to capture how the context words are less or more important with respect to the meaning of the target word. The fourth study examines how the contextual informativeness model, originally designed to develop curricula for human vocabulary learning, can also be used for developing curricula for various word embedding models. We identify sentences predicted as low informative for human learners are also less helpful for machine learning algorithms.

Having a rich understanding of user behaviors, responses, and learning stimuli is imperative to develop an intelligent online system. Our studies demonstrate interpretable methods with cross-disciplinary approaches to understand various cognitive states of students during learning. The analysis results provide data-driven evidence for designing personalized features that can maximize learning outcomes. Datasets we collected from the studies will be shared publicly to promote future studies related to online tutoring systems. And these findings can also be applied to represent different user states observed in other online systems. In the future, we believe our findings can help to implement a more personalized vocabulary learning system, to develop a system that uses non-English texts or different types of inputs, and to investigate how the machine learning outputs interact with students.

Chapter 1

Introduction: Using Behavioral and Linguistic Data to Improve Learning Systems

The goal of this dissertation is to understand how users interact with learning systems and provide data-driven evidence to improve the overall learning experience. To achieve this, we use machine learning techniques on behavioral and linguistic data to predict various cognitive states of students while they are using a contextual vocabulary learning system. More specifically, this dissertation presents multiple machine learning applications that solve unique challenges in developing a contextual vocabulary learning system, including predicting students' disengagement, measuring partial knowledge of vocabulary with a fine-grained and interpretable method, and estimating the amount of contextual information in a sentence with respect to target words for learning. We believe the interdisciplinary approach of our studies can provide deeper understanding of how users interact with other information systems. Also, the interpretability of our machine learning models can help developers identify where their systems can be improved and help users understand the results they see from these systems.

Learning can occur in various types of information systems. For example, online learning systems like intelligent tutoring systems (ITS) or massive open online courses (MOOCs) provide digital environments for learning, helping students develop their knowledge of particular subjects through a carefully designed curriculum. Other information systems that are designed for more general-purpose information seeking, such as exploratory search systems or general search engines, are also valuable sources for exploring new information (Rieh et al., 2016). Users of these systems develop their knowledge through iterative search processes and by synthesizing pieces of information that they found useful while using the system. However, building an effective information system for learning cannot be done with accurate and efficient retrieval algorithms alone. It also requires deeper understanding of how users interact with the system (Sinatra et al., 2015; Waddington et al., 2016), detailed representation of the learning process (Durso and Shore, 1991; Shore and Durso, 1990; Van Inwegen et al., 2015), and presenting appropriate learning materials (Frishkoff et al., 2016a; Papoušek et al., 2016) to maximize users' learning outcomes.

In this dissertation, chapters address various cognitive states related to learning. We investigate how machine learning models can use behavioral and linguistic data to predict students' different hidden states during learning, and we derive data-driven insights on how to improve the learning experience. Building a prediction model for students' disengagement can let the system know when to intervene with a student and guide them to be more engaged with the task. It can also inform the teacher about different types of related behaviors, such as semantically related vs. lexical identical off-task responses (Chapter 3). Developing a fine-grained semantic representation of a student's partial word knowledge can be helpful for the system to correctly evaluate a user's progress and determine the best selection of next learning materials. Also, interpretable representations of open-ended responses can help both students and teachers easily understand where they stand and what needs to be improved from learning (Chapter 4). Predicting the amount of contextual informativeness of sentences would provide an automatic way to score the stimuli used in a contextual word learning system. It has significant potential for automatically developing curricula from a range of learning sources, from expert-edited textbooks to crowd-generated learning materials from the Internet. In Chapter 5, we develop a deep-learning model that can effectively predicts the amount of informativeness from context. Chapter 6 further examines the application of the model, especially for developing more efficient curricula for training word embedding models. We believe the findings from these studies can improve the design of various information systems that can be used for learning.

In the following sections, we explain more about the motivation to understand different states during learning, and our problem-solving strategies with machine learning methods (Section 1.1). Then we summarize the contribution of each study (Section 1.2), and covers the organization of the following chapters (Section 1.3).

1.1 Better understanding of User Behaviors and Linguistic Data

We learn about the world by interacting with information. We use information to identify problems (Wilson, 1997) and to decrease uncertainty of situations (Spink and Cole, 2006). Understanding user behavior and linguistic inputs better is important for identifying a student's current knowledge level and how she perceive the learning material provided from the system. Thus, it can be imperative for making an effective personalized vocabulary learning system. Using machine learning applications can help automatize this process and achieve scalability. However, it would require clear definitions of the user state that the system wants to identify, careful feature engineerings that can meaningfully represent the observed data, and building an interpretable model for students to understand their learning progress and for instructors to design future curricula.

1.1.1 Modeling User Engagement

Engagement is a crucial element that can ensure the effective delivery of information (Walonoski and Heffernan, 2006). It is a comprehensive mental activity that incorporates perception, attention, reasoning, volition, and emotions (Cocea and Weibelzahl, 2011). Well-designed systematic guidance, such as prompting motivational messages (Baker et al., 2006) or providing hints about which cognitive skills to use to solve a task (Roll et al., 2007; Arroyo et al., 2007), can help users learn more and stay longer in a learning system.

In an educational context, understanding the multidimensional construct of engagement can be helpful for developing personalized features and improving the learning outcomes from a system. Engagement can be shaped by different factors. Individual differences in users' intrinsic interests, motivations, and prior knowledge about the topic (Wilson, 1997), and preferences on positive results (White, 2013) or particular information foraging strategies (Chi and Pirolli, 2006; Kendal et al., 2004) can play important roles in determining patterns of user engagement in information systems. A user's engagement state may be observed with diverse behavioral patterns. For example, over multiple question items, a user can show their off-task state through repetitive responses or more random responses that are not in the context of the provided questions. Exploring how these factors are related to individual users' engagement states would provide important evidence for designing more engaging and effective learning systems.

In both on- and off-line learning environments, engagement is a crucial predictor of learning outcomes (Bizas et al., 1999; Herrington et al., 2003; Goldberg et al., 2011). Many studies have used behavioral signals, such as response time (Beck, 2005) and frequency of hint use or repetitive mistakes (Baker et al., 2004; Paquette et al., 2014), to predict users' engagement states in learning systems. These studies were often conducted with learning systems designed for STEM topics and structured question formats. However, with a vocabulary learning system with open-ended questions, designing predictive features for an engagement-related state has been less investigated and would require different features that focus on the linguistic properties of student responses.

In the first study, we present novel features for predicting students' off-task states from a vocabulary learning system. Features include single-trial online variables, which are derived from behavioral interactions and linguistic responses that are collected from answering a single question, and context-sensitive online variables, which further examine the linguistic relationship between past responses. We also include analysis on feature importance that describes which types of context-sensitive features are useful for predicting particular types of off-task responses.

1.1.2 Evaluating Partial Knowledge

Along with behavioral interactions, linguistic data can also provide much information on how users learn by using the system. For example, compared to multiple-choice responses, open-ended responses can provide more details about the user's current knowledge about the question (Durso and Shore, 1991; Adlof et al., 2016). Previous studies showed that different cognitive states during learning, such as confusion (Yang et al., 2015) or motivation (Chopra et al., 2018), can be predicted from the questions that a user asked in a forum.

Learning a new vocabulary happens incrementally (Frishkoff et al., 2011). Using machine learning methods to automatically evaluate students' open-ended responses would be important for the vocabulary learning system to track where students are doing well or poorly, and suggest better learning items that can maximize the efficiency of learning.

Unlike multiple-choice questions, responses from open-ended questions may contain rich information about students' knowledge, including partially correct responses. However, there are some existing challenges. First, automatically evaluating open-ended responses can be a difficult task. It requires a machine to have thorough knowledge in the domain and correctly understand the responses like human experts do. With fast retrieval of an accurate representation of the meaning of the target word, the vocabulary learning system can provide a correct evaluation of students' responses. Second, measuring a partial knowledge state is essential for accurately tracking learning progress. Existing studies have often used a limited number of categories (i.e., correct, partially correct, and incorrect) to represent students' knowledge states without detailed explanations (Durso and Shore, 1991; Dale, 1965). Having a fine-grained representation of partial knowledge is essential to identify the missing semantic component of a student's knowledge about a word.

In thes second study, we suggest a novel semantic representation method that combines a classic semantic differential scale method (Osgood et al., 1957) with a modern neural word embedding technique (Mikolov et al., 2013). The results of this study can provide an interpretable and scalable method for automatically evaluating students' responses from vocabulary learning systems. We also think that this method can be used in different tasks, including semantic evaluation of longer texts, or understanding semantic biases in different types of texts across time.

1.1.3 Informativeness of Learning Content

Suggesting adequate content is essential for retaining users' motivation (Reeve, 2012) and achieving a more satisfying user experience (Belkin, 2008) from an information system. Especially in learning, identifying easier (or harder) learning materials and choosing a more efficient learning curriculum can benefit both machine (Bengio et al., 2009) and human learners (Frishkoff et al., 2016a; Brown et al., 2005) with a more efficient learning process.

Identifying better learning materials is nontrivial. For contextual word learning, it requires a deeper understanding of the contextual information for the target word that the user is about to learn. If we can reliably predict the quality of the material, we can significantly improve the effectiveness of learning. For example, in our previous study, we found that scaffolding the difficulty levels of learning materials can improve learning outcomes in language learning (Frishkoff et al., 2015). Moreover, a well-designed training curriculum can also be helpful for a machine learning system to achieve a more efficient training process (Bengio et al., 2009). In the contextual vocabulary learning scenario, predicting the amount of contextual information from a text can enhance the outcome of contextual word learning for both human and machine language learners.

For both human (Landauer and Dumais, 1997) and machine learners (Mikolov et al., 2013), neighboring contextual information is an important source for learning the meaning of a target word. However, not all contexts contain much information about the target word; some contexts may contain more or less information than others for learning. Our previous study showed that using more informative sentences for the contextual word learning task can lead to more efficient learning results (Frishkoff et al., 2016b).

In the third study, we introduce a deep-learning NLP model that can predict the amount of contextual information from single- or multi-sentence contexts. The model achieves a significantly better performance than baseline models for predicting contextual informativeness scores with single-sentence contexts, and it achieves state-of-the-art performance with multi-sentence contexts. The model's output is easily interpretable, so it can tell which context words contribute more to the informativeness of the sentential context. Moreover, we show that the model can be useful for developing a more efficient training curriculum for simple word embedding models. We believe the results of this study would be useful for various potential applications, including machine reading, few-shot learning, personalized question sets for learning systems, and inferring knowledge levels from various user text inputs, by accurately capturing the contextual informativeness of a given context.

1.1.4 Curricula for Machine Learning Models

As we can reliably predict the amount of contextual information of the target word from sentence(s), we also investigate the model's applications. For example, many machine learning models tend to perform better with large sized training data. However, humans are known to be very good at making rich inferences from a small number of example data. In the context of machine learning research, this learning process is called few-shot learning (Fei-Fei et al., 2006; Lake et al., 2015). For few-shot learning models, identifying the quality of instructional materials becomes more critical for improving performance.

In the fourth study, we examine the use of the contextual informativeness model, originally developed for contextual word learning of human students. We use this model to create a more efficient curriculum for machine learning models, especially for word embedding models in various learning scenarios, including batch learning and few-shot learning. We show that sentences predicted as low informative are also less useful training materials for word embedding models, and simply filtering out the low informative sentences from the training set significantly improves the embedding models' performance. In the future, identifying more optimized curricula for word learning, and identifying factors related to the curriculum effect based on contextual informativeness would provide more exciting opportunities to understand how human and machine learners acquire knowledge of language.

1.2 Overall Contributions

This dissertation illustrates machine learning applications on behavioral and linguistic data from a vocabulary learning system to predict students' different cognitive states. Based on an interdisciplinary approach, the studies deliver thorough and interpretable analysis results that can improve understanding of students' learning behavior and help make learning systems better in the following ways.

Data-Driven Insights for Personalized Learning Systems First, the results from the studies provide data-driven insights on how to develop more sophisticated personalized learning systems through interpretable analysis results. For example, we conducted thorough ablation tests to investigate which predictive features are more helpful than others in our model. Detailed failure analyses also identified potential improvement points of our model. Based on these findings, each study suggests how to improve students' learning outputs from a vocabulary learning system. In

Chapter 3, feature analysis results show that the contextual features based on the linguistic relationship between responses are more important than the traditional single-trial features (Section 3.4.3). This is followed by a failure analysis that also shows major types of off-task responses (e.g., orthographically identical responses vs. semantically similar responses) and how the suggested model captures different off-task cases on the student level (Section 3.4.3). Chapter 4 identifies which of Osgood's semantic scales are more important for predicting students' learning (Section 4.4.3). These data-driven results suggest how the system can develop smaller and faster representation without sacrificing much prediction power on student short- and long-term learning (Section 4.4.3). Chapter 5 includes visualizations on how contextual informativeness is constructed from a sentence for the target word (Section 5.6). Such interpretable results can be very useful for educators and curriculum developers for contextual vocabulary learning by identifying more contributing context words that help students to infer the meaning of the target word. Chapter 6 compares the performance of word embedding models between various curricula based on the contextual informativeness scores and the number of sentences included (Section 6.4). This method can provide a useful strategy for identifying more informative materials for domain-specific models or better quality training materials for few-shot learning tasks.

Interdisciplinary Approaches Second, our studies compare existing studies from different research fields, such as psychology and machine learning, and combine different techniques to extract meaningful information from behavioral and linguistic data to predict students' various hidden states during learning. Our first study in Chapter 3 focuses on not only the behavioral features that are often used in existing ITS or learning psychology studies, but also the semantic relationships between a student's responses and the target, or across the series of responses that are automatically estimated using the machine learning algorithm (Section 3.3.3). The second study in Chapter 4 investigates the combination of traditional semantic differential scales and the neural word embedding method (Section 4.3.2), which can provide both scalability and interpretability when the system or domain expert users evaluate students' responses (Section 4.3.2). The third study in Chapter 5 compares our deep-learning based NLP model to a traditional approach based on lexical properties (Kapelner et al., 2018) for quantifying the relationship between context words and the target word (Section 5.4). The fourth study in Chapter 6 further examines how the model originally developed for human vocabulary learning can also be used for improving the efficiency of machine learning algorithms on word learning (Sections 6.5 and 6.6).

New Datasets Related to Vocabulary Learning Third, we are sharing new datasets that we collected from students while they are using our vocabulary learning system. Each study defines a novel problem and collects a related dataset from students while they are using our vocabulary learning system. Datasets include students' behavioral and linguistic interactions during a meaning-generation task (Chapter 3) and a series of practice question responses that can illustrate how students learn the meaning of new target words by using a contextual vocabulary learning system (Chapter 4). We believe this dataset could be useful for future studies on understanding contextual word learning behavior. In could also be used in designing an adaptive system that can improve students' engagement and learning outcomes in vocabulary learning. More technical details on how we collected online interaction data are described in Chapter 2.

Moreover, we also collected various crowdsourced annotations to quantify the amount of contextual informativeness of cloze sentences when they are used in an instructional setting. In Chapter 5, we present cloze sentences with human annotations for contextual informativeness scores that can be used to train an effective prediction model for automatically scoring the learning material for vocabulary learning. Further, the results from Chapter 6 show the model trained from this dataset also can be used to quantify the potential effectiveness of different corpora and to describe a new target word to learn. In this dissertation, we also share the details of designing the crowdsourcing task (Appendix A.2) and annotating protocols (Appendix A.2.2) for used in future studies. The collected datasets will be shared through our vocabulary system's web page[1].

General Insights on Understanding Online User Behaviors Lastly, the results from these studies can be useful for other information systems that are not limited to vocabulary learning. Predicting off-task behaviors during learning can help both instructors and the system by letting them know when to intervene. The behavioral and linguistic features can also be applicable to any learning system that uses open-ended questions. Detailed representation of semantic characteristics can be used in various domains that may require interpretable representations of text input. For example, summarizing the sentiment of a short paragraph, such

[1] `http://dscovar.org/`

as a product review or Twitter feed, can be useful for researchers or product managers who want to quickly understand the quality of unstructured text data without much investment in gaining technical knowledge on word embedding. The model for predicting contextual informativeness of text can be used for other tasks, such as measuring the quality of students' note-taking or determining the expertise level of a user from a community Q&A post, where the amount of contextual information can indicate users' different knowledge states. Developing curricula based on the contextual informativeness for machine learning models can introduce useful comparison between NLP models and human language learners. Applying our curricula building method to more sophisticated NLP models may also reveal interesting insights on how to build a more effective curriculum for vocabulary learning.

1.3 Thesis Organization

In Chapter 2, we introduce the contextual vocabulary tutoring system we used for the studies: Dynamic Support of Contextual Vocabulary Acquisition for Reading (DSCoVAR). This chapter covers a brief overview of DSCoVAR's teaching strategy, session structures, and how the system records the data used for each study.

Chapters 3, 4, 5, and 6 include four individual studies based on DSCoVAR. Each study has different goals, but they commonly investigate how machine learning models can capture different cognitive states of vocabulary learning students and use the outcome to improve the learning experience. The studies conduct comprehensive analyses on behavioral and linguistic signals to predict particular user states, and provide explainable results on how models suggested in each chapter work.

More specifically, in Chapter 3, we identify which behavioral and linguistic predictors are more important for predicting students' off-task states(Section 3.3.3 and Section 3.4.3) and illustrate how predictive features capture different types of off-task behaviors (Section 3.4.3).

Chapter 4 focuses on suggesting interpretable metrics for automatically evaluating a student's partial knowledge state in the vocabulary learning system (Section 4.3.2 and Section 4.3.2)). The study also compares which Osgood scale has more predictive power for predicting learning gain (Section 4.4.3 and Section 4.4.3).

Chapter 5 aims to develop a deep-learning model that can predict the amount of contextual information in sentences with respect to the target word (Section 5.4 and Section 5.5). Multiple experiments are also conducted to check the model's generalizability (Section 5.5.4) and interpretability (Section 5.6).

Chapter 6 extends the contextual informativeness model from Chapter 5, and investigates how the model's output can also be used to develop more efficient curricula for word embedding models in various settings, including batch learning settings (Section 6.5) and few-shot learning settings (Section 6.6).

Chapter 7 illustrates the impact of the studies in related research fields, including educational technology, psycholinguistics, and natural language processing. We also discuss example applications that can potentially benefit from our findings, and some limitations that can be addressed in future studies.

Lastly, Chapter 8 concludes the dissertation by summarizing the findings of individual studies and their broader implications.

Chapter 2

Describing the System: Dynamic Support of Contextual Vocabulary Acquisition for Reading

In this dissertation, we investigate behavioral interactions and linguistic inputs used in a vocabulary learning system called Dynamic Support of Contextual Vocabulary Acquisition for Reading (DSCoVAR). DSCoVAR is a vocabulary learning system that aims to teach students new vocabulary (target words) with contextual information provided within a sentence. The contextual word learning (CWL) system does not explicitly provide the meaning of the target word. Rather, students are asked to use linguistic context, such as the nearby semantic and syntactic cues, to infer the meaning of the unknown word. Through repeated practice, students learn the meaning of the word from various contexts. The following sections describe some technical details on how we implemented DSCoVAR, the structure of DSCoVAR that is related to experimental settings, and features from DSCoVAR that are closely related to the individual studies of the dissertation. [1]

2.1 Using a Computer System for Contextual Word Learning

Each tutoring system may employ different strategies for vocabulary learning. For example, associative word learning (AWL) applications use word pairs (e.g., semantic or lexical association pairs) to represent the meaning of the target word (e.g., flashcards) (Jenkins et al., 1978; Mastropieri et al., 1985). Although AWL is an effective strategy for learning simple, domain-specific, or concrete words (Solman and Wu, 1995), it may not be suitable for learning more complex and abstract words, such as Tier-2 words that have multidimensional meanings. These words are better learned through different contexts, or CWL. For example, the word *canny* has a similar meaning to *smart*, but it also implies an attitude or action that may be shrewd or dishonest. An example context like "a *canny* showman adept at manipulating the audience's feelings and expectations" provides contextual cues that the meaning of the word involves being manipulative or shrewd. This example illustrates how

[1] This chapter paraphrases a broader survey of DSCoVAR from the book chapter (Frishkoff et al., 2016a) and technical details of the system from the unpublished work (Frishkoff et al., n.d.). Gwen Frishkoff and Kevyn Collins-Thompson led the overall study design for DSCoVAR. Sungjin Nam was responsible for technical implementations, such as developing the system client and database, setting up the server instances, and evaluating the system's scalability feature.

CWL can provide deeper and more sophisticated knowledge about the target word, especially for more advanced vocabularies (Huang and Eslami, 2013).

There are existing CWL systems that can provide adaptive target words, based on pre-test results (Wang, 2016) or multimodal stimuli for learning (Ballard and Yu, 2003). DSCoVAR differs from the existing CWLs in the following ways.

First, DSCoVAR promotes *active inferencing.* Although learning through context can deliver rich information about the target word, it may limit students from actively guessing and learning the meaning of the word. However, the active inferencing skill is necessary if less information exists and to develop full knowledge about the word (Koren, 1999).

Second, DSCoVAR can provide a *real-time assessment* of student responses. DSCoVAR uses computational methods to identify if a student is disengaged with the task and to determine the partial word knowledge state. Knowing when a student has disengaged from the task can help the student finish the task and can identify other related states, such as confusion or frustration (Baker et al., 2010; Yang et al., 2015; Picard and Picard, 1997). Estimating partial knowledge can help provide immediate feedback during the training, and determine more personalized learning material based on the detailed analysis. Chapters 3 and 4 illustrate initial attempts to develop computational models for these features.

Third, DSCoVAR can provide contextual stimuli that was computationally predicted by the system. To automatize the stimuli generation or collection process for vocabulary learning, Chapter 5 investigates a NLP model that can predict the quality of stimuli of sentence stimuli used in DSCoVAR.

2.2 System Architecture

DSCoVAR consists of multiple components. This section briefly describes the role of each component and how it can affect the learning experience in DSCoVAR.

2.2.1 Database

DSCoVAR's database includes learning stimuli, students' demographic information and prior skill levels on vocabulary, and updates on how the students interacted with the system during learning. First, the database contains instructional materials, including the list of target words and pronunciations, training contexts (i.e., sentences containing contextual information about the target words), and pre- and post-test questions. Second, students' subject records include their log-in identification, demographics, grades, and language ability testing results measured before using DSCoVAR. Third, the database records how users interact with DSCoVAR. These records include information like timestamps for each question item loaded and submitted, types of errors and their frequencies, a list of submitted responses, and response accuracy.

2.2.2 Client Server

DSCoVAR is a web-based application. Students can use their desktop web browsers to access DSCoVAR. We used a standard LAMP stack (Linux Apache HTTP web server, MySQL database, PHP scripting) to build the client server.

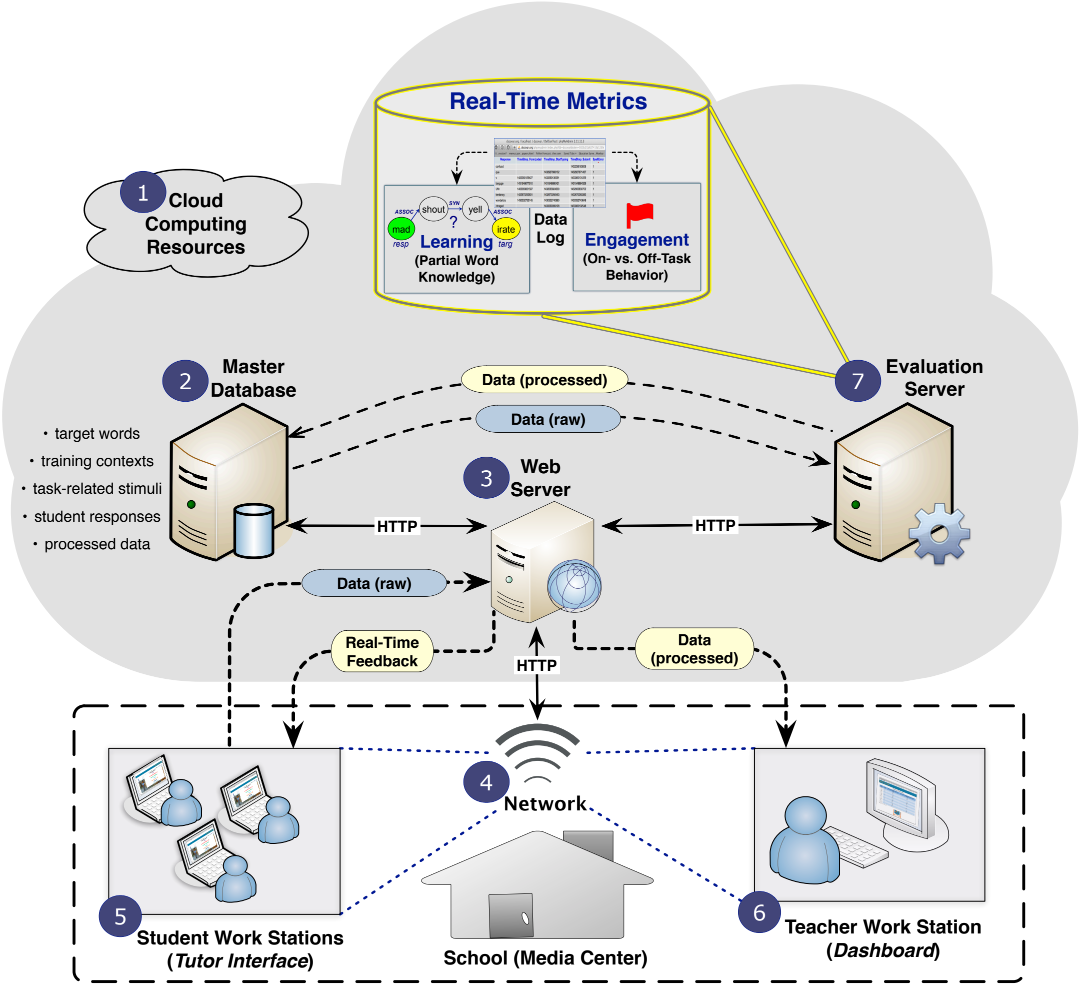


Figure 2.1: The overall system architecture of DSCoVAR. Students connect to DSCoVAR and practice their vocabulary knowledge through questions based on contextual sentences (3). The database (2) contains information on target words, contexts, and other stimuli. It also records how students interact with the system. As the students respond to the questions, their behavioral and linguistic interactions are evaluated at a separate server in real-time (7).

2.2.3 Evaluation Server

We also hosted a dedicated server to process computationally expensive evaluations. This includes real-time evaluation of student responses and predicting off-task states. In the training session, DSCoVAR provides real-time feedback to the student based on the similarity between the response and the target word. We used MESA (Collins-Thompson and Callan, 2007) to calculate the similarity between the two words. MESA uses a random-walk-based method on web-based resources, such as WordNet (Miller, 1995), to estimate the distance between the student's response and the target word. The same server instance can also be used to measure the partial knowledge state of student responses. For this task, we hosted the Word2Vec model as a service[2], which could be accessed through the *http* call (more details about the model can be found in Chapter 4). Other simple tasks, such as spell-checking, is also hosted in the evaluation server. DSCoVAR uses GNU Hunspell,[3] an open-source spellchecker, to validate single word responses. Students receive spelling suggestions, which is especially useful for younger or less skilled readers. Also, if a response is too short (e.g., fewer than 2 characters) or contains non-alphabetic characters, DSCoVAR generates an error message and prompts the student to enter a different response.

Both the client and evaluation servers are hosted in Amazon Web Services (AWS)[4], with dynamic scalability of the number of server instances (e.g., load balancing). For example, if server utilization exceeds a particular point, DSCoVAR automatically increases the number of instances to accommodate more access without modifying any current connections. This way, we can easily increase (or decrease) the capability of the client server to accommodate a wide range in the number of

[2] `https://github.com/nishankmahore/word2vec-flask-api`

[3] `http://hunspell.github.io/`

[4] `https://aws.amazon.com/`

users. Our stress test results show that each `c4.2xlarge` instance could handle 35 simultaneous users with less than 3 seconds of latency.

2.3 Session Structure

The goal of DSCoVAR is to teach students how to use contextual information in a sentence to infer the meaning of an unknown target word. DSCoVAR consists of multiple sessions. Pre-test and post-test sessions measure the student's knowledge of target vocabularies before and after using the system. Between the pre-test and post-test sessions, we have a training session, where the actual learning is happening. At the beginning of the training session, the system shows videos that contain example contextual word learning strategies. After the video, students can practice these strategies with practice questions that contain the actual target words to learn. Practice questions use various sentences that contain the target word. Each target word appears multiple times within different context sentences. Figure 2.3 illustrates the overall session structures.

2.3.1 Pre-test and Post-test Sessions

Questions in the pre- and post-test sessions include open-ended and multiple-choice questions. Both the open-ended and multiple-choice questions ask students to provide a synonym for the given target word.

Students answered the post-test questions twice, right after the training session (immediate post-test session) and a week after the training session (delayed post-test

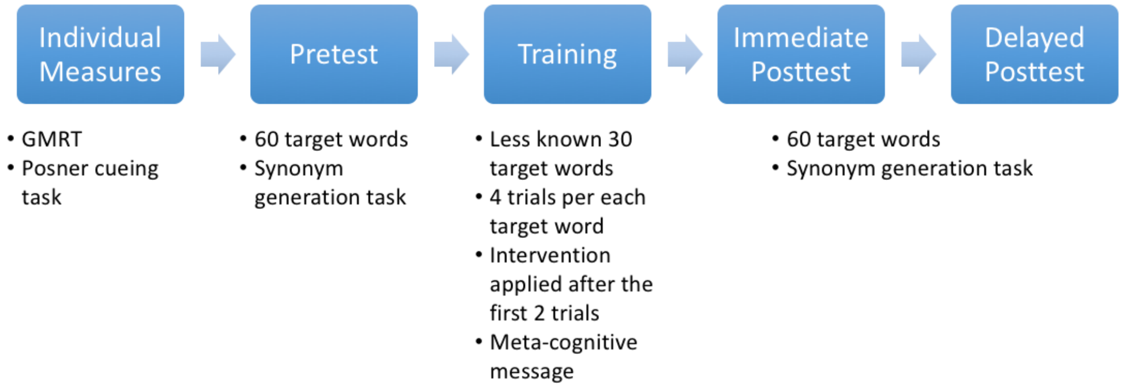

Individual Measures
• GMRT
• Posner cueing task
Pretest
• 60 target words
• Synonym generation task
Training
• Less known 30 target words
• 4 trials per each target word
• Intervention applied after the first 2 trials
• Meta-cognitive message
Immediate Posttest
Delayed Posttest
• 60 target words
• Synonym generation task

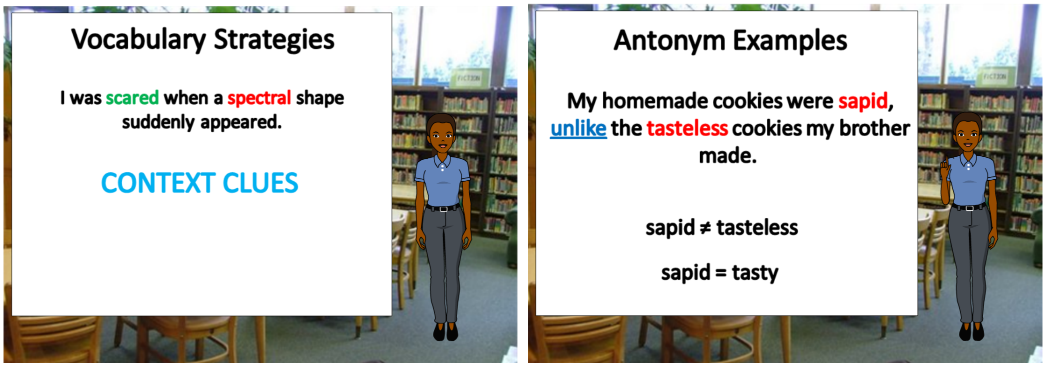
Figure 2.2: Session structure diagram. Individual skills are measured before the pre-test session. The pre-test session measures students' knowledge on target words. Students learn about target words in the training session. The post-test sessions measure the immediate and delayed knowledge, by comparing the results from the pre-test session.

session). Differences in student performance between the pre-test and immediate post-test sessions indicate a short-term learning gain. Students' performance on the delayed post-test measured the long-term learning effect of using DSCoVAR. Questions from the pre-test and post-test sessions are not subject to interventions.

2.3.2 Training Session

In the training session, students learn the meanings of target words through practice questions. Before answering the questions, students watch videos that teach them how to find contextual information from the sentence to infer the meaning of the unknown word (Figure 2.3.2).

In the training session questions, students are asked to provide a synonym of the



Vocabulary Strategies

I was scared when a spectral shape suddenly appeared.

CONTEXT CLUES

Antonym Examples

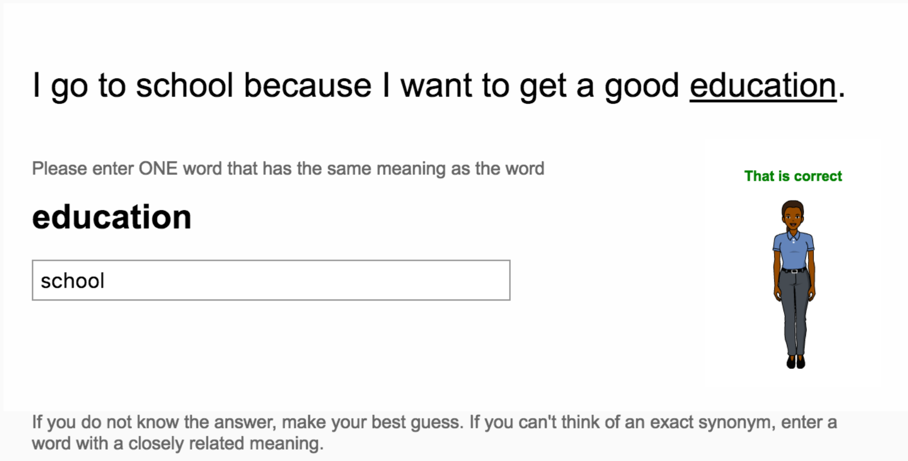
My homemade cookies were sapid, unlike the tasteless cookies my brother made.

sapid ≠ tasteless

sapid = tasty

Figure 2.3: Screenshots of the instruction video. The video describes how to use context cues to infer the meaning of the unknown target word (left). It also contains examples for different context cues (e.g., an antonym relationship between *sapid* and *tasteless*) so students can apply the strategies in their training session (right).

target word, using contextual information from a sentence that contains the target word. The amount of contextual information in the sentence will determine the question difficulty. Figure 2.3.2 presents an example of a question.



I go to school because I want to get a good education.

Please enter ONE word that has the same meaning as the word

education

school

That is correct

If you do not know the answer, make your best guess. If you can't think of an exact synonym, enter a word with a closely related meaning.

Figure 2.4: An example of the training session question. The student is asked to provide the synonym of the target word (e.g., education) based on the contextual information from an accompanying sentence and previous questions.

Students learn thirty unique target words in the training session (Chapter 4). For each target word, four different questions are provided to the students. In Chapter 4, we present the interpretable and fine-grained semantic scales that evaluate students' responses to the practice questions from the training session, and predict the students' short- and long-term learning gain derived from the pre- and post-test (e.g., immediate and delayed) sessions.

2.4 Learning Materials

For the training session, we developed a set of sentences that contain different amounts of contextual information with respect to the target word, which can be considered the difficulty level. DSCoVAR can control different orders of problem difficulties for the same target word. For example, the scaffolding condition suggests practice questions in ascending order of difficulty (i.e., easy to hard). Opposite scaffolding (i.e., hard to easy) or uniform (e.g., all medium) orders are also possible. We hired undergraduate research assistants to generate sentences with different amounts of contextual information. We provided a guide for how to differentiate the amount of contextual information in a sentence (Appendix A.1). To validate the level of contextual informativeness of generated sentences, we also designed crowdsourcing experiments to collect human annotations and quantify the informativeness levels. More details on instructions that we provided to crowdworkers (Appendix A.2.2), how we designed the crowdsourcing task, and methods to quantifying the cloze sentences (Appendix A.2) can be found in the later Appendix sections.

These sentences are used in DSCoVAR for students to practice contextual word learning. In Chapter 5, we also use this data to train a model that can automatically predict the amount of contextual information about the target word from single- and multi-sentence passages.

2.5 Behavioral and Linguistic Records from DSCoVAR

DSCoVAR records rich behavioral and linguistic interaction data from the students during learning (Section 2.2.1). For the behavioral data, DSCoVAR records temporal data, such as timestamps of mouse-clicks on answer options, typing responses in text boxes, and submitting responses. DSCoVAR includes some gaming prevention methods, such as a spell checker and a format checker for answer responses. Both checking modules guide students to input correctly spelled and formatted responses, and the system records the number of erroneous interactions.

Linguistic interactions include students' open-ended response contents. DSCoVAR guides students to answer the questions with a single word (i.e., provide a synonym for the target word in a sentence). In the training session, the similarity score between a student's response and the target word is recorded and used for providing immediate feedback for learning.

In Chapter 3, we use behavioral and linguistic interaction features from the pre-test session's open-ended questions to predict the off-task state of students. In Chapter 4, we use the results from pre- and post-test sessions' multiple-choice question results to measure students' learning gain from the system. We also investigate how to evaluate the semantic qualities of linguistic responses from the training session.

2.6 Is DSCoVAR Useful?

Our studies included in this dissertation does not directly investigate the effectiveness of DSCoVAR. However, we conducted several lab studies that investigate how contextual word learning strategy from DSCoVAR can help students learning difficult Tier-2 words. For example, in the first study, we tested if computationally scored response accuracy is helpful for immediate learning gain (Collins-Thompson et al., 2012). The results from this study were later used for DSCoVAR's real-time feedback based on classifying relatedness between a response and the target.

In the second study, we tested if the real-time feedback on response accuracy can improve students' short- and long-term learning (Frishkoff et al., 2016b). The results indicated that the group that received trial-by-trial feedback for responses outperformed the control group who did not receive the feedback.

The third study investigated if scaffolding can be helpful for more efficient vocabulary learning (Frishkoff et al., 2016a). Compared to using all easy contexts for training, the scaffolded condition showed lower performance in short-term learning, but higher performance in long-term learning evaluation.

From these studies, we show that contextual word learning implementation in

DSCoVAR can help K-12 students to learn new vocabulary and retain knowledge over the long-term.

Chapter 3

Predicting Off-task States from Behavioral and Linguistic Signals

3.1 Introduction[1]

Intelligent tutoring systems (ITS) aim to provide a student-centered environment for more effective learning. Compared to traditional learning environments, ITS can provide unique interaction opportunities between the learning system and students. For example, a typical ITS can determine an appropriate difficulty level of questions by modeling an individual student's previous knowledge level (Ma et al., 2016; Papoušek et al., 2016), or generate systematic feedback to student responses to help them develop their own learning strategies (Arroyo et al., 2007; Roll et al., 2011). By closely monitoring student behavior in ITS, educational researchers can observe

[1] This study was published as Sungjin Nam, Gwen Frishkoff, and Kevyn Collins-Thompson. 2018. Predicting students disengaged behaviors in an online meaning-generation task. *IEEE Transactions on Learning Technologies*, 11(3):362–375.

students' current progress on learning and anticipate their future performance. These advantages of ITS allow researchers to achieve deeper understanding of various behaviors of students while they interact with ITS and design more effective learning systems (Corbett and Anderson, 1994).

In order to provide a personalized learning experience, it is essential to estimate some model of each student's state. Measuring students' engagement level is one way to inform the ITS about the need for potential interventions. In many educational and psychological studies, engagement is considered as an important factor for predicting students' learning outcomes (Rowe et al., 2010). In ITS, retaining student engagement is also a critical factor for ensuring the effective delivery of educational materials (Walonoski and Heffernan, 2006). Previous studies have shown that engagement levels can be predicted based on various measures, such as student's response time for individual questions (Beck, 2004) or reading materials (Cocea and Weibelzahl, 2007), students' prior domain knowledge (Walonoski and Heffernan, 2006), and repetitive errors or help requests (Baker et al., 2004).

Our study is part of an effort to develop a web-based contextual word learning (CWL) system that aims to help students acquire strategies for learning the meaning of an unknown word based on contextual cues in the surrounding text. This study investigates how log data from such a vocabulary-learning ITS can be used to predict specific disengaged behaviors during an online meaning-generation task. Disengaged behaviors examined in this study include students' gaming behaviors from (Baker et al., 2004), such as systematic or repetitive incorrect attempts, and other motivation-related behaviors, like sharing responses with other students even if they were answering different practice questions from ITS. The meaning-generation task used in this study is a part of the pre-test phase of our vocabulary tutoring system. In this task, the CWL system asks free-response definition questions in which students type what they think the meaning of a new word is. Although this phase is not training oriented, it provides a well-defined yet challenging starting point for modeling disengaged behavior during a language-based task.

As we show later, disengaged behaviors in this scenario are characterized by a variety of response types, such as consistent use of nonsensical or irrelevant words, names of friends or celebrities, or repetitive patterns across multiple responses (e.g., a repeated word or a sequence like "one," "two," "three"). In this paper, we illustrate how to extract meaningful features from the log data, including event components and textual response features, and predict disengaged labels collected from human judges. Findings in this paper will help to achieve better understanding of students' disengaged behaviors in vocabulary-learning systems with open-ended questions, and potentially broader types of adaptive ITS in complex cognitive domains.

3.2 Related Work

Our approach builds on three main areas of research: (1) research in psychology, which describes the neurocognitive components of engagement, (2) studies in the learning sciences, which have identified several categories of disengagement, and (3) measurement and modeling of trial-based behavior within an intelligent tutoring system (ITS). In this section, we provide a brief summary of work in these three areas. Then we summarize our aims and approach within the current study.

3.2.1 Neurocognitive Components of Engagement

Engagement is a complex construct that reflects multiple underlying processes in the mind (Cocea and Weibelzahl, 2011). Specifically, studies in neuroscience points to two key components of engagement: motivation and cognitive control (Koechlin et al., 2003).

Motivation can be viewed as an emotion-driven tendency to act in a particular way. For example, fear is a mental state often triggers disengagement or withdrawal, whereas anger drives the compulsion to attack, and excitement motivates positive engagement or approach (Dolan, 2002). Emotional states are associated with distinct, but overlapping pathways in the brain, especially in subcortical networks (LeDoux, 2003). There is abundant evidence for the role of motivation in learning (Cocea and Weibelzahl, 2007; Johns and Woolf, 2006). The Yerkes-Dodson model (Teigen, 1994) shows an inverse quadratic relation between the level of motivation ("arousal") and performance across a variety of cognitive and perceptual tasks. This confirms that good learning outcomes require an optimal level of interest or engagement. Either too little (boredom) or too much (anxiety) can lead to disengagement and subsequent failures in school (Teigen, 1994).

Cognitive control is a second aspect of engagement, which involves top-down or strategic (i.e., "executive") attention (Van Veen and Carter, 2006), and is responsible for attentional focus, and for monitoring the alignment of a past or present action with a particular goal (Braver, 2012). Neural pathways in cognitive control converge in areas of the prefrontal cortex and are connected with subcortical pathways of motivation-emotion (Koechlin et al., 2003). Cognitive control is often essential for learning, particularly in tasks that require active decision-making (Gläscher et al., 2012), sustained attention to particular cues among multiple competing stimuli (Sarter et al., 2001), or integration of multiple cues in order to make an appropriate response (Badre and Wagner, 2004).

Together, motivation and cognitive control lead to behaviors that can be labeled as "engaged" or "disengaged". Studies of real-world (e.g., classroom) learning have shown that student engagement predicts learning outcomes, independent of prior knowledge or experience (Rowe et al., 2010). For example, more motivated students tend to choose deeper learning strategies, which typically require greater effort and engagement (Nesbit et al., 2006).

It is important to note that the relation between motivation and attention is context-sensitive, rather than simple and static. Consider a student who is highly motivated but has poor cognitive control. As the task continues, the student is likely to experience repeated failure, which can leads to frustration and disengagement from the task. This example shows why it is important to capture changes in student engagement with fine-grained measures throughout a task.

3.2.2 Levels of Engagement in Computerized Systems

Engagement can be characterized as a complex construct and can be represented at different levels of granularity (O'Brien and Toms, 2010). In studies of learning within computer-based systems, some researchers have characterized engagement at a relatively coarse-grained level. For example, (Kizilcec et al., 2013) characterized students in a massive open online course (MOOC) as more or less engaged, depending on their contributions to an online discussion group (also see (Sinha et al., 2014)). These studies have shown that disengagement is a strong predictor of student attrition (Kizilcec et al., 2013; Sinha et al., 2014).

Other studies have attempted to capture "trial-by-trial" changes (e.g., by every question item) in student engagement using more fine-grained measures. For example, Baker et al. (Baker et al., 2004) analyzed log data from student interactions with a graphical tutoring system. They showed that trial-specific features —– such as the latency, duration, and accuracy of individual responses —– were useful in predicting item-level student engagement.

Based on multiple studies of behavior within adaptive tutoring systems, Koedinger et al. (Koedinger et al., 2013) identified three types of disengaged behaviors. The first type is "gaming the system" (Baker et al., 2004; Baker, 2007; Walonoski and Heffernan, 2006) and occurs when students exploit patterns or regularities in an ITS in order to complete a task with minimal effort (e.g., "help abuse" (Baker et al., 2004)). Students may attempt to game the system when they are superficially motivated to complete the task (e.g., for a course grade), but are either unwilling or unable to engage the deeper strategies that promote genuine learning and mastery (Baker et al., 2008b). The second type includes behaviors that are off-task, that is, oriented away from the ITS, e.g., talking to one's neighbor, sleeping, or spacing out (Baker, 2007). Both gaming and off-task behaviors reflect a lack of motivation to engage in a key cognitive processes, especially when they impose a high cognitive load (Baker et al., 2008b; Walonoski and Heffernan, 2006). The third type is careless mistakes, such as typographical errors or accidental clicking on a web link. In some instances, these errors may reflect disengagement (e.g., a momentary lapse of attention). However, they may also reflect a failure to execute the intended action. Because accidental behaviors can be ascribed to more than one underlying cause (e.g., motor-control versus engagement), they may be harder to predict than intentional behaviors, such as gaming (Baker, 2007; Baker et al., 2004).

3.2.3 Modeling Trial-by-Trial Engagement

Trial-by-trial estimates of engagement are of interest for adaptive systems, because they can be used to determine the most effective way for the system to respond throughout a task (Hussar and Pasternak, 2010). Importantly, work by Baker, Koedinger, et al. (Baker et al., 2004, 2008b; Walonoski and Heffernan, 2006) has shown that different types of features are predictive of gaming versus other types of disengaged behavior. In particular, they have shown that patterns of response across trials (e.g., repetition of the same response (Baker et al., 2004)) can help to predict student behavior. These findings suggest that trial-by-trial measurement and modeling of engagement can benefit from the use of context-sensitive measures, as well as single-trial measures of behavior. In addition to online measures, studies have used various offline measures to capture student-level variables — such as skill level (estimated knowledge of a particular topic, working memory, etc.) — and item-level variables — such as problem difficulty (number of steps in a computation, written word frequency, etc.). Student- and item-level features can help capture sources of variance that are not well accounted for by online measures.

Previous studies presented the results related for detecting students' disengagement behaviors in ITS. Baker et al. (Baker et al., 2004) reported that they achieved AUC score of 0.82 for predicting harmful gaming responses, such as repetitively making errors or rapidly firing the help function accompanied with less learning gain. Paquette et al. (Paquette et al., 2015) reported AUC scores from 0.829 to 0.901 across student data from multiple ITS for predicting gaming behaviors defined in their previous study (Baker et al., 2008a) by using expert rule based gaming features. Cocea and Weibelzahl reported up to 89.8% accuracy (equals to 10.2% error rate; reported recall rates were up to 0.94) on predicting disengaged behaviors, defined as spending too much or little time on learning materials, in computer programming tutoring systems (Cocea and Weibelzahl, 2009).

3.2.4 Overview of Current Study

Previous studies on modeling student disengagement typically focused on other domains, such as science, technology engineering, and mathematics (STEM) topics, (Baker et al., 2004; Paquette et al., 2014; Baker et al., 2008a; Cocea and Weibelzahl, 2009). However, modeling students' behavior in a vocabulary-learning system with open-ended questions may require additional domain-specific features to address important usage scenarios. Features based on students' text responses, like similarity between responses and target words or the number of erroneous attempts that try using non-alphabetic characters or misspelled answers, may more fully represent students' interaction with such a language-oriented system.

In the present study, we used data-driven methods to predict different patterns of engagement within a vocabulary assessment task. During this task, the ITS presented a word (known as the target word) and prompted students to type in the word's meaning. Response data were logged and used to generate a set of trial-specific measures, including response time and task-related errors. Log data were then provided to human experts, who were asked to flag responses that were consistent with disengaged (gaming or off-task) behavior.

We had three specific research questions (RQ) and corresponding hypotheses.

- RQ1: Can we use trial-specific measures, based on responses to generating synonym questions, to predict variability in student engagement on a trial-by-trial basis?

Previous studies have shown that student interactions with an ITS (i.e., log data) can be used to predict disengaged behaviors (Baker et al., 2004, 2008b; Walonoski and Heffernan, 2006). In the present study, we extended this prior work by using free-text data to predict variability in engagement. To this end, we computed each response's semantic features (e.g., similarity to the target word meaning) and orthographic features (e.g., spelling similarity to a presented question item), as well as standard log data such as response latency. This expected to provide a rich set of features for prediction and analysis of student behavior.

- RQ2: Do context-sensitive measures predict variability in student engagement that is not accounted for by single-trial features?

In this study, we expected to replicate findings from Baker et al. (Baker et al., 2004), which showed that context-sensitive measures are important predictors of trial-specific engagement. To extend this prior work, we investigated how linguistic measures, such as orthographic and semantic similarity measures among recent responses, can improve the performance of predicting students' disengaged behaviors.

- RQ3: How can we characterize patterns of disengaged behavior among students with a strong tendency toward disengaged behavior?

Lastly, we also investigate how each contextual feature type captures particular patterns of disengaged responses, including repetitive responses and semantically related sequential responses.

3.3 Method

In this section, we describe methods for the acquisition and analysis of different types of student behaviors within a vocabulary-training ITS. Section 3.3.1 describes the procedures for acquisition of raw data, including free-text responses from a Meaning-Generation task, which are the main focus of our analysis. Section 3.3.2 explains the methods for gold-standard labeling of log data from the Meaning-Generation task and the identification of predictive features, including online (single-trial and context-sensitive) measures, as well as offline (student- and item-level) measures. Finally, Section 3.3.4 describes the statistical models.

3.3.1 Study Design

Participants

Thirty-three student participants (from 4th to 6th grade) were recruited from a small laboratory school, which is located on a university campus in a medium-sized city in the northeastern United States. Prior to the main task, students completed the online version of the Gates-Macginitie Reading Test (GMRT), a standardized test of reading comprehension ability (MacGinitie et al., 2000).

Data from 8 participants were excluded from the final analysis because they did not complete the GMRT. The resulting dataset included data from fourteen girls and eleven boys. There were ten 6th grade students, ten 5th grade students, and five 4th grade students. All selected participants were native English speakers, and did not have a history of developmental or reading disability. The twenty-five participants included in our analysis scored well above average on the GMRT: the median (composite) score was 75 (mean = 70.24; s.d. = 27.79).

Stimuli

Students were presented with 60 SAT-level (or so-called Tier 2) English words (Blachowicz et al., 2006). These stimuli "target words" were balanced between 20 adjectives (e.g., defiant), 20 nouns (e.g., eminence), and 20 verbs (e.g., languish). Individual students typically differ in their degree of familiarity and knowledge with a particular word, reflecting different frequencies and types of exposure to words. Given this observation, we selected 60 target words that, on average, we expected to be difficult, but not necessarily novel; these items are sometimes referred to as "frontier" words, which have been viewed useful words to target within a vocabulary intervention (Beck et al., 2013). To identify appropriate frontier words, we used grade-specific language norms and selected a mixture of three types of words: (1) Known words, which the participants would be able to recognize and define ($\sim 20 - 30\%$, based on word norms), (2) Familiar words, which they would recognize, but be unable to define, ($\sim 20 - 30\%$), and (3) Unknown words, which would be novel, that is, indistinguishable from nonce words ($\sim 40 - 60\%$). Our strategy for word selection made it likely that there would be variability in word knowledge across both students and items.

Experimental Tasks

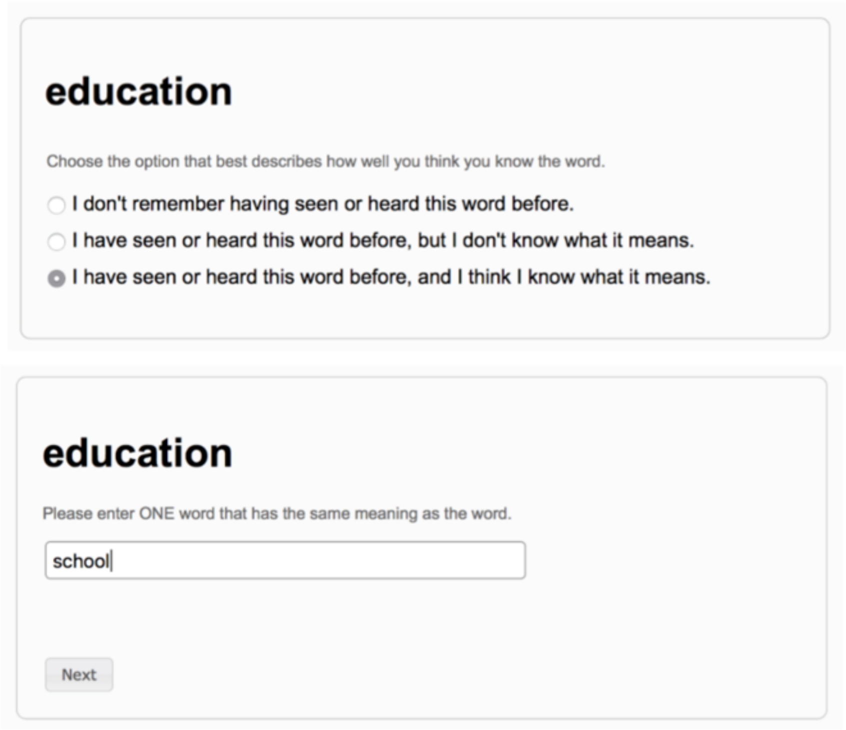
As mentioned previously, data for the present analysis were collected during the first (pre-test) session of a classroom experiment using a vocabulary-training ITS. The pretest session comprised two parts. Part 1 included a familiarity-rating question and a meaning-generation question. Part 2 was a synonym-selection task: accuracy on Part 2 is used to evaluate learning outcomes (Frishkoff et al., 2016a). Because we restrict our attention to patterns of behavior during the meaning-generation task, Part 2 data were excluded from the present analysis.

Familiarity-Rating Task Students were presented with each of the 60 target words and were asked to indicate if the word was completely *unknown* ("I have never seen or heard this word before"), *familiar* ("I have seen or heard this word before, but I do not know what it means"), or *known* ("I have seen or heard this word before, and I know what it means"). Familiarity ratings were used as one of the predictor variables in the present analysis, to capture individual differences in prior knowledge of words.

Meaning-Generation Task Immediately after each familiarity rating question, students were asked to type the meaning (synonym or near-synonym) of each target word. Students were instructed to enter only single-word responses and to avoid use of non-alphabet characters, including hyphens (e.g., compound words) and spaces (e.g., multi-word responses). Students were required to click an “Next” button to submit each typed response.

The present version of the ITS used jQuery’s text field validation module to verify that each response was a single word that is different from the target word. If a student provided an ill-formatted response, the system generated an error message and asked the student to enter a response that consists of a single word, with no spaces or hyphens (in response to typographic errors), or a response that is different from the target word (if the student tried to game the system by retyping the target).

Additionally, PHP’s Pspell extension was used for spell checking. If the student provided a well-formatted (one-word) response that was orthographically incorrect (i.e., a non-word string or a misspelled word), the ITS responded with an error message and provided up to three spelling suggestions that were orthographically similar with the provided response. After the student provided a well-formatted, orthographically correct response, the ITS proceeded to the next item.



education

Choose the option that best describes how well you think you know the word.

○ I don't remember having seen or heard this word before.

○ I have seen or heard this word before, but I don't know what it means.

● I have seen or heard this word before, and I think I know what it means.

education

Please enter ONE word that has the same meaning as the word.

school

Figure 3.1: Examples for familiarity-rating (top) and meaning-generation (bottom) tasks from the pre-test session.

3.3.2 Data Annotation

Log data recorded during the Meaning-Generation task were used to identify patterns of student behavior that were judged to be disengaged (either gaming or off-task). Operational definitions and rules for application of labels are described in the present section.

Log Data

Log data comprised a total of 1,500 items, including free-text responses from 25 participants to 60 target words. Note that each item was associated with at least one response since the task involves forced generation (Frishkoff et al., 2016a). In the present sample, students provided an average of 1.26 responses to each target word (SD = 0.981; median = 1).

For each item, the ITS recorded the following: the question onset (i.e., target item load time), the student response (i.e., typed response string), the response onset (i.e., the first typed in time), the response offset (i.e., response submission time), and the number and types of error messages.

Gold-Standard Labeling of Log Data

Two native English speakers, one female and one male, provided gold-standard labels for each response item in the log data. Both judges were undergraduate students at the University of Michigan. Judges were informed that the student task was to provide the meaning (synonym or near-synonym) for each word. They had no additional knowledge of the experiment procedures, study methods, or hypotheses.

Instructions to judges were modeled after Baker et al. (Baker et al., 2004). For each of the 1500 items, they were asked to detect the "disengaged" response for one or more of the following reasons:

- The response seemed "less serious or completely irrelevant" for a given target word,
- The response was part of a series of "patterned responses over different question items", or
- The response was part of a series of "repetitive false submissions with invalid answers".

The log data was provided as a single spreadsheet file. For labeling, each judge was instructed to sort the data in two formats. Format 1 consisted of responses submitted to the system by each student (ordered by question onset time, grouped within students). This ordering enabled the judges to detect response patterns over time within each student (Table 3.1). Note that a single label was generated for each item, even when the item triggered multiple disengaged responses (e.g., due to repetitive spelling or related validation errors). Format 2 consisted of responses submitted to the system in strict temporal order, without subject-level grouping. This ordering enabled judges to detect patterns that were common across students at around the same time, which could suggest answer sharing activities between students (Table 3.2).

Judges could go back and forth between different sorting formats to generate the labels. Disengagement behavior labels were only counted for the analysis when both judges agreed on their decisions for the same question item. Inter-rater agreement between the two judges was moderately high (Cohen's kappa, 0.734). Both judges agreed on 276 of 1,500 responses (about 18.4%) as representing instances of disengaged behavior.

Table 3.1: Examples of response sequences with disengaged responses (labeled with *) occurring within highly disengaged students. The table shows how disengaged behavior can vary including highly repetitive responses (A1, A2, A3), random irrelevant words (B1, B3), and sequences (B2).

Student					
A1	A2	A3	B1	B2	B3
blah*	not*	run*	dark	Twelve*	cow*
blah*	sure	run*	pandas*	Mimic	dragon*
hastily	not*	run*	penguins*	Thirteen*	pear*
blah*	added	run*	donkey*	Fourteen*	orange*
blah*	not*	run*	bob	hello*	block
blah*	not*	hi*	scared	Flag	argue

Table 3.2: Examples of disengaged responses from study data (labeled with *) occurring across different students within a similar time frame. Students S3 and S4 provided very similar responses almost concurrently even though they were confronted with different target words. Both judges considered these responses as disengaged behavior, suspecting the possibility that the students were talking with each other.

Seq#	**Students**	**Target Word**	**Response**
11	S1	reticent	receive
12	S2	perturbed	clean
13	S3	tenable	rain*
14	S4	vie	Rain*

3.3.3 Predictor Variables

To represent and model student behaviors that were subsequently labeled as "engaged" or "disengaged", we considered three types of predictor variables (Table 3.3). The main predictors were based on data from the Meaning-Generation task; henceforth, we refer to these as "online" (fixed effect) variables. These include Single-Trial Online Variables (STOV) — which reflect individual responses on each trial, without consideration of prior or future responses — and Context-Sensitive Online Variables (CSOV) — which are defined on each trial, but also reflect patterns of behavior across trials. Another set of variables are "offline" with respect to the Meaning-Generation task (random effect variables); they include subject-level factors (`Grade`, `Skill`), item-level factors (`Target`), and self-rated `Familiarity` with each target word. For some online variables, we also computed the mean and standard deviation (SD) to reflect the response patterns from question items with multiple attempts.

Single-Trial Online Variables (STOV)

STOV characterize participant responses to a particular item (`Target`) during the Meaning-Generation task. Each STOV is either an error-based feature, a temporal feature, an orthographic feature, or a semantic feature.

Error-based features represent objective failures to comply with the task instructions. `NoErrForm` is the number of ill-formatted responses to a particular item (e.g., hyphenated and multi-word responses, or responses with non-alphabetical

Table 3.3: Predictor Variables. *STOV*: single-trial online variables. *CSOV*: context-sensitive online variables. *OFFV*: offline variables. (Responses are “accepted” if they generate no spelling or formatting errors)

Type	**Name**	**Descriptions**
STOV	NoErrSpell	Number of misspelled responses to target word
	NoErrForm	Number of ill-formatted responses to target word
	RTStart	Response times for starting typing the response (in milliseconds), natural log-transformed (average (mean), standard deviation (SD), and the accepted response (final))
	RTFinish	Response times for finishing typing in the response (in milliseconds), natural log-transformed (mean, SD, final)
	RspLen	Number of characters of the responses that were submitted to the system (mean, SD, final)
	SimOrth	Orthographic similarity between the typed responses and the spelling of the target word, measured in trigram cosine similarity (mean, SD, final)
	SimSem	Semantic similarity between the accepted typed response and the meaning of the target word, measured by MESA (Frishkoff et al., 2008) (final)
CSOV	PattOrth.pX	Orthographic similarity between the accepted typed response to current item and responses to X previous items (mean, SD)
	PattSem.pX	Semantic similarity between the accepted typed response to current item and responses to X previous items (mean, SD)
OFFV	Target	Individual target words (Target_{01}... Target_{60})
	Familiarity	Self-rated familiarity with the target word (unknown, familiar, or known)
	Grade	Student’s grade level (4^{th}, 5^{th}, 6^{th})
	Skill	Student’s composite reading comprehension skill (GMRT score (MacGinitie et al., 2000))

strings), and NoErrSpell is the number of misspelled responses to that item (i.e., misspellings).

Temporal features represent the latency of each response and include response time to type the first character of a response string (RTStart) and response time to press the "Next" key to submit the completed response string (RTFinish). Both features are measured in milliseconds, and values are natural log-transformed prior to analysis.

Orthographic features include the length of the response (i.e., the number of characters; RspLen) and the orthographic overlap between the response string and the target item (SimOrth), based on trigram cosine similarity, which measures spelling similarity between the response and the target word by three adjacent characters.

Finally, the accuracy of the response is represented by the semantic similarity between the response word and the target word. In the present analysis we use Markov Estimation of Semantic Association (MESA) (Frishkoff et al., 2008) to compute semantic similarity (SimSem).

Context-Sensitive Online Variables (CSOV)

CSOV also characterize free-text responses on a particular trial. However, unlike STOV, CSOV are defined with respect to student responses on previous trials. In the present study, we define two such variables. The first, PattOrth, represents the orthographic overlap between the response to the current item and the responses on one or more consecutive trial(s). The second, PattSem, represents the semantic overlap between the current response and responses on previous trials. In each case, mean and standard deviation were computed.

We also determined the optimal window size by comparing results from the variable selection process (described later in Section 3.3.4) for window sizes of one, three, five, and seven. For example, if the window size is seven, CSOV are defined over eight consecutive trials (where the eight trial is the current trial, and trials 1-7 represent information from previous trials). In this case, CSOV values are only computed for trials 8-60. Thus, models with CSOV were trained and evaluated with the dataset that excludes trials 1-7.

Offline Variables

Offline variables include subject- and item-level predictors. Subject-level predictors vary across students, but not across items. In the present analysis, we included two subject-level predictors: `Grade` (4th, 5th, or 6th) and `Skill` (whether the GMRT composite score is above median or below). GMRT is consisted of three scores: Reading, Comprehension, and Vocabulary scores. In this study, we used a score from Comprehension as a random effect variable since it is a composite score of other two. Item-level features vary across items. In the present analysis, `Target` (target words) was included as an item-level feature. Finally, `Familiarity` (known, familiar, unknown) was used to represent individual differences in self-rated knowledge of each item. Familiarity was considered "offline" because it was acquired outside the Meaning-Generation task.

3.3.4 Modeling Methods

In this section, we describe methods for identification of predictive features and selection of accurate and robust models. All analyses were conducted in R (R Core Team, 2015).

Identification of Predictive Features

To select the structures of main predictors (fixed effect) for subsequent modeling, we applied a two-stage process. In both stages, the hill-climbing (HC) algorithm (Margaritis, 2003) and a step-wise selection process were applied to the dataset including all 25 participants. In the first stage, we performed structure learning using the HC algorithm to automatically extract the pairwise interactions of fixed-effect variables that can be used to predict disengaged behavior labels. Further higher order interaction structures were not considered in this study for easier interpretation. In the second stage, the pairwise interactions identified in step one were entered into a step-wise variable selection process with other online and offline variables. The direction of edges in pairwise interactions was ignored at this stage.

Interaction Structure Learning. To identify pairwise interactions between fixed effect variables, we used the HC algorithm, implemented in R's `bnlearn` package (Scutari, 2009). The algorithm was applied for 1,000 iterations using the `boot.strength` function. This process results in a Bayesian network structure that contains the probability of each edge and direction estimated by bootstrap samples. Non-significant edges were filtered out from the averaged output by using the `averaged.network` function. Lastly, the Bayesian network output was summarized by extracting Markov blanket nodes of disengaged behavior labels. A Markov blanket is a set of variables that contains enough information to predict the value of the particular node, which includes parents, children, and children's other parents of the node that is going to be predicted (Pearl, 2014).

Variable Selection. As a second stage of structure learning, we used a step-wise process with the Akaike information criterion (AIC) that maximized model fit. To extract features for the logistic regression (LR) models, we used R's built-in step function. This step-wise algorithm repeatedly added and dropped fixed-effect variables based on the model's AIC score.

In the case of mixed effect logistic regression (MLR) models, fixed effect variables were initially selected by backward-fitting process, removing each variable if it does not improve the model's AIC score. After this process, each random effect variable was tested. As an initial starting point, `Familiarity` was added to the MLR model. Other random effect variables, such as `Target`, `Grade`, and `Skill`, were added to the model if the model's F score changed significantly ($p < 0.05$) by adding the variable. Lastly, another backward-fitting process for fixed effect variables was conducted to see if any fixed effect variables could improve the AIC score with the updated random effect variable setting. This process was done by using the `fitLMER.fnc` function of `LMERConvenienceFunctions` package (Tremblay et al., 2015).

Statistical Modeling

To predict disengaged response patterns (see Section 3.3.2 for response labeling), we compared two families of statistical models: logistic regression (LR) and mixed-effects logistic regression (MLR). LR is a regression model that is widely used for classifying data with binary labels. MLR is a more general form of regression model that incorporates random effect variables to capture variations among repeated measures. Although it may not be conventional to use LR models with data that contains repeated measures over items and students, we included LR models in our analysis because several of our CSOV features, such as those computed from both a student's current response and response history, do in fact capture per-student and per-item correlation.

We used R's built-in `glm` function to compute LR models and `glmer`, as defined in `lme4` package (Bates et al., 2014), to compute MLR models. Coefficients for fixed and random effect variables of MLR models were estimated with $nAGQ = 1$, which represents the number of points per axis for testing the Gauss-Germite approximation to the log-likelihood (Bates et al., 2014).

To estimate model accuracy in generalizing to previously unseen students, we used cross-validation over the set of students. Parameters for regression models were estimated from the training set, and performance was averaged over held-out sets. We computed the average error rate (proportion of incorrect classifications), precision, recall, and F1 score results across 25 folds that were created from leaving out each student's data as a held-out validation set. Binary values of predicted labels in the first two evaluation methods were decided using a threshold probability of 0.5: if the predicted probability of disengagement for the item was bigger than 0.5, the item was considered to be representing disengaged behavior. After this, feature analysis was performed to identify the relative importance of variables for predicting disengaged behaviors.

3.4 Results

We now discuss the learned prediction models (Section 3.4.1), compare the models' disengaged behavior prediction accuracy (Section 3.4.2), and assess the relative importance of different variable sets in selected prediction models for disengaged behaviors (Section 3.4.3).

3.4.1 Structure Learning

Interaction Structures from the Hill-climbing Algorithm

The HC algorithm identified Bayesian networks that capture conditional dependencies between the fixed-effect variables. Each predictor set included single-trial online variables (STOV) or additional context-sensitive online variables (STOV+CSOV). The resulting Bayesian network structures are shown in Figure 3.2.

From Figure 3.2, we could extract the number of pairwise interactions among STOVs. The results show conditional dependency relationships between variables, which can be translated into pairwise interactions in the regression model, including relationships between time for start typing responses and orthographic similarity,

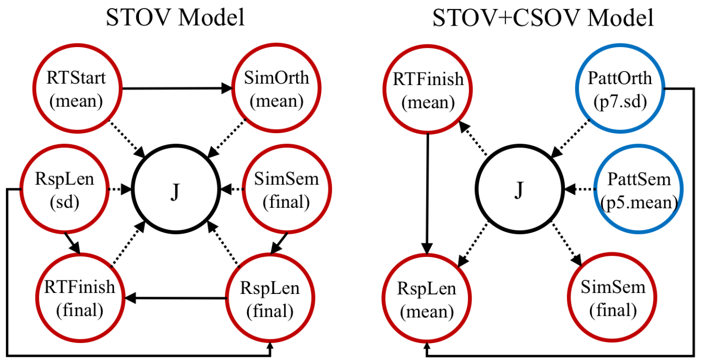


Figure 3.2: The Bayesian network structures learned for fixed-effect variables using single-trial online variables only (left) and single-trial + context sensitive online variables (right). Solid lines represent the pairwise interaction structures extracted from the HC algorithm, and tested in a step-wise process. Compared to the structure from STOV only, adding CSOVs to the model reduces the number of pairwise interaction structures extracted from the HC algorithm (STOVs: red circles; CSOVs: blue circles; disengagement label: black circles (J)).

time for complete typing response and response length, and response length and semantic similarity.

When single-trial online variables and context-sensitive online variables were combined, the interaction variable structure was slightly simpler, and included dependencies between mean response time for finishing typing in and mean response length, and mean response length and standard deviation of orthographic similarity between previous responses and the target word.

These interaction structures were passed on to a step-wise process and treated as candidate predictive features with other fixed effect online variables. The results for selected predictors are presented in the next section.

Table 3.4: Regression coefficients for logistic regression (LR) and mixed-effects logistic regression (MLR) models. Interaction structure for MLR model with single-trial online variables (*MLR:STOV*) derived same results with *MLR:STOV* model without the interaction structure. *S+C* indicates the *STOV+CSOV* structure. (†: The model includes interactions) ($^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$, $^{\cdot}p < 0.1$)

	Predictors	**LR**				**MLR**		
Type	Name	STOV	STOV†	S+C	S+C†	STOV	S+C	S+C†
	(Intercept)	5.16***	14.20***	2.49$^{\cdot}$	2.73 *	4.16***	-0.63	0.15
STOV	RTStart.mean		-0.39***					
	RTStart.sd	0.36		0.71*	0.83**			
	RTStart.final	-0.39***				-0.33***		
	RTFinish.mean							
	RTFinish.sd			-0.81*	-0.76$^{\cdot}$			
	RTFinish.final	-0.56***	-1.39***	-0.29$^{\cdot}$	-0.25	-0.55***		
	RspLen.mean	-0.34***	-0.53*	-0.30***	-0.46***	-0.22***	-0.27***	-0.41***
	RspLen.sd	0.53***	1.15**	0.55***	0.59***	0.50***	0.63***	0.56***
	RSpLen.final		-1.65***					
	SimOrth.mean		-7.73***				-7.42*	-5.79***
	SimOrth.sd	-3.87		-6.88*	-6.99*			
	SimOrth.final	-8.08***		-5.66***	-5.45**	-8.63***	-5.82***	
	SimSem.final	-0.20***	-0.04	-0.21***	-0.21***	-0.14***	-0.16***	-0.16***
STOV : STOV	RTFinish.final : RspLen.final		0.16***					
	RspLen.sd : RspLen.final		-0.11$^{\cdot}$					
	RspLen.final : SimSem.final		-0.04$^{\cdot}$					
CSOV	PattOrth.p3.sd			-2.84				
	PattOrth.p5.mean			4.24	3.05			
	PattOrth.p7.sd			6.40**	-1.93		7.84***	2.70
	PattSem.p3.mean			0.15***	0.11 *		0.17***	0.17***
	PattSem.p3.sd			0.08*	0.08*			
	PattSem.p5.mean				0.11*			
	PattSem.p7.mean			0.08$^{\cdot}$				
STOV : CSOV	RspLen.mean : PattOrth.p7.sd				1.27***			0.94**

Selected features in LR models

A step-wise variable selection process was followed with interaction structures suggested from the HC algorithm. In the logistic regression model with single response variables (*LR:STOV*), coefficients indicated that the response was significantly likely to be disengaged when following behavioral patterns were observed ($p < 0.05$):

- If there was a short response time for both initiating the accepted response (`RTStart.final`) and then completing it (`RTStart.final`);
- If the average length of responses was short (`RspLen.mean`) or variation among length of responses was large (`RspLen.sd`);
- If the last submitted response (`SimOrth.final`) was orthographically dissimilar to the target word;
- If the accepted response was semantically dissimilar to the target word (`SimSem.final`).

During the step-wise variable-selection process, variables for the number of misspelled or illegally formatted responses were dropped. If we add interaction to the model structure, interactions between the finishing time and response length of the accepted response was found to be statistically significant ($p < 0.001$). This means short and quickly typed responses tend to be classified as disengaged.

When introducing the context-sensitive online variables to the model (*LR:STOV+CSOV*), most single-trial online variables remained statistically significant ($p < 0.05$). In the model, context-sensitive variables described additional information of disengaged behaviors. Disengaged behaviors were more likely to be observed if the mean (`SemPatt.p3.mean`) or standard deviation (`PattSem.p3.sd`) of semantic similarity between the current response and the previous responses ($p < 0.05$) were high. Interaction between average length of the response and standard deviation of orthographic similarity among the previous responses was found to be significant as well (`RspLen.mean : PattOrth.p7.sd`; $p < 0.05$) in STOV+CSOV condition. This means disengaged behaviors were more likely to be observed if the response was short and placed within an orthographically less diverse response pattern.

Selected features in MLR models

As with LR models, we built MLR models using two different variable sets (*STOV* and *STOV+CSOV*) and interaction structures. Results from the step-wise algorithm with MLR models were also similar to those for LR models. Some noticeable differences were that the p-values for selected fixed-effect predictors were more stable than the ones in the LR models; less number of variables were selected; and none of the response time type variables (`RTStart` and `RTFinish` series) were significant predictors if the model included context-sensitive information (*STOV+CSOV* models). Lastly, all pairwise interaction structures from single-trial online variables were dropped during the step-wise process.

Each offline variable was added as a random intercept and also evaluated in the step-wise process. Table 3.5 explains variances and standard deviations of

Table 3.5: Results for selected random intercepts in MLR models. Adding CSOV to MLR models decreased the amount of information explained by some offline variables like `Familiarity` and `Skill`. (†: The model includes interactions)

	Variance (Std. Dev.)		
(Int.)	STOV	STOV+CSOV	STOV+CSOV†
Familiarity	1.54 (1.24)	0.95 (0.97)	0.81 (0.90)
Target	NA	NA	NA
Grade	0.14 (0.38)	0.14 (0.38)	0.14 (0.37)
Skill	0.28 (0.53)	NA	NA

selected random intercepts. In a single-trial online variable condition ($STOV$), `Familiarity`, `Grade`, and `Skill` were selected as random intercepts. In models with context-sensitive information ($STOV$+$CSOV$), *Familiarity* and *Grade* were selected as random intercepts. Adding context-sensitive online variables, which contain information about the relationship between responses, to MLR models decreased the amount of information that was explained by `Familiarity` and `Skill` variable.

Overall, a step-wise process selected the list of predictors that increase LR and MLR models' goodness of fit that measured in AIC score. The nature of the resulting single-trial online variables suggests that disengaged behaviors can be explained by such measures as time elapsed for initiating or completing the answer, length of the response, and orthographic or semantic relationship between the response and the target word. Multiple context-sensitive variables were also found to be significant predictors for both logistic regression and mixed-effect logistic regression models, suggesting that information about performance in previous questions also can be useful for predicting disengaged behaviors.

3.4.2 Model Evaluation

In this section, we evaluate various models' prediction performance and identify which model is better than others for predicting student behaviors labeled as disengaged behavior. All measures are reported by using a leave-one-subject-out cross-validation process.

Overall Error Rate

Table 3.6: Average classification error rates for prediction models (lower numbers are better). Logistic regression models (LR) were performing marginally better than mixed effect models (MLR). Adding context-sensitive information (*STOV+CSOV*) and interaction structures for fixed-effect variables provided marginal improvements for models' average error rates (†: model includes interactions, ^{a}: single-trial online variable (*STOV*) models evaluated with *STOV* dataset, ^{b}: *STOV* models evaluated with *STOV+CSOV* dataset that does not include the first seven items) (Scores in bold: the best performing model with a given variable set; NA: MLR models without any significant fixed-effect interactions)

	STOV^{a}		**STOV^{b}**		**STOV+CSOV**	
Models	Mean	95% CI	Mean	95% CI	Mean	95% CI
Baseline	0.184	[0.101, 0.267]	0.192	[0.106, 0.279]	0.192	[0.106, 0.279]
LR	0.154	[0.104, 0.204]	0.155	[0.106, 0.205]	0.112	[0.067, 0.156]
LR†	**0.148**	**[0.098, 0.198]**	**0.152**	**[0.104, 0.200]**	**0.109**	**[0.066, 0.152]**
MLR	0.155	[0.103, 0.206]	0.155	[0.106, 0.205]	0.116	[0.071, 0.162]
MLR†	NA	NA	NA	NA	0.118	[0.071, 0.164]

First, the models' performance was measured with average error rates. Error rate in this paper was defined as following:

$$error\ rate = \frac{number\ of\ incorrectly\ classified\ items}{total\ number\ of\ items}$$

We used a baseline prediction model that simply guesses the majority label (i.e., non-disengaged responses) for a given response, which is an initial target to surpass for prediction models. For the dataset used to train single-trial online variable models (*STOV*), the baseline error rate was 18.4%, the proportion of disengagement labels in the collected data. The baseline for the dataset used to train models with additional context-sensitive online variables (*STOV+CSOV*) was slightly different at 19.2%, since the CSOVs require responses having a history window of at least seven prior responses.

Our results showed that the average scores of all STOV models performed better than the baseline model (Table 3.6). Using either LR or MLR prediction models with STOV predictors reduced the average error rate compared to the baseline from 15.8% to 16.3%. For LR model, adding interaction structures learned from the hill-climbing algorithm improved the prediction performance only by an additional 3.9% over the regular STOV models (step-wise process for MLR model dropped all interaction structures).

Models including context-sensitive online variables used a slightly smaller dataset than the *STOV* model evaluation results. Using the same dataset from *STOV+CSOV* evaluation on *STOV* models yield from 19.3% to 20.8% better performance than the baseline model.

The largest improvements came from adding context-sensitive online variables to prediction models (*STOV+CSOV*). Compared to the STOV model, the performance of STOV+CSOV models was improved from 25.2% to 27.3%. Also, in *STOV+CSOV* models, adding interaction variables among fixed-effect variables only brought -1.7%

to 2.7% improvements. Lastly, we found that LR and MLR models did not have significantly different prediction accuracy for this task.

Precision and Recall Measures

For further comparison of models with different measures, we also analyzed models with standard evaluation metrics, such as precision, recall, and F_1 scores. From the data collected from 25 students, we had eight students who did not have any labels for disengaged behaviors. With students' data without any disengagement labels, it was impossible to measure precision and recall if disengaged behaviors are considered as positive cases. For example, if we consider disengagement labels to be positive cases, data from those eight students do not contain any real positive conditions, which makes the denominator value of recall zero. Moreover, if the classifier correctly guesses all true labels from those students, as all negative cases (engaged state) throughout the task, it also makes the denominator value of precision zero. Therefore, in this section, precision, recall, and F_1 results were calculated by treating disengagement behaviors as negative cases.

The perfect recall value of the baseline model means that it never misses the positive cases (labels for engaged state) since it always predicts the item as positive. Since other LR and MLR prediction models contain inevitable noise and fail to predict a few instances of on-task states, it seems like they perform worse than the baseline model. However, this does not mean that our models also perform poorly than the baseline model in other evaluation measures

Overall, precision, recall, and F_1 scores showed similar patterns of performance

Table 3.7: Precision, recall, and F_1 scores for prediction models. Mixed effect models (MLR) perform better than logistic regression models (LR). Adding context-sensitive variables (*STOV+CSOV*) marginally increased the precision, recall, and F_1 scores. Adding interaction structure was also maginally helpful for increasing precision and F_1 scores. (†: model includes interactions; Scores in bold: the best performing model with the scoring theme; NA: MLR models without any significant fixed-effect interactions)

STOV						
	Precision		Recall		F_1	
Models	Mean	95% CI	Mean	95% CI	Mean	95% CI
Baseline	0.816	[0.733, 0.899]	1.000	[1.000, 1.000]	0.881	[0.818, 0.944]
LR	0.853	[0.786, 0.921]	0.959	[0.939, 0.980]	0.892	[0.846, 0.938]
LR†	0.855	[0.787, 0.922]	**0.967**	**[0.951, 0.982]**	**0.896**	**[0.850, 0.942]**
MLR	**0.866**	**[0.797, 0.932]**	0.956	[0.932, 0.979]	**0.896**	**[0.852, 0.940]**
MLR†	NA	NA	NA	NA	NA	NA

STOV+CSOV						
	Precision		Recall		F_1	
Models	Mean	95% CI	Mean	95% CI	Mean	95% CI
Baseline	0.808	[0.721, 0.894]	1.000	[1.000, 1.000]	0.875	[0.810, 0.939]
LR	0.889	[0.838, 0.940]	0.979	[0.972, 0.986]	0.927	[0.896, 0.957]
LR†	0.893	[0.843, 0.942]	0.979	[0.971, 0.987]	0.929	[0.899, 0.958]
MLR	0.888	[0.837, 0.940]	**0.981**	**[0.973, 0.989]**	0.927	[0.896, 0.958]
MLR†	**0.894**	**[0.845, 0.943]**	0.980	[0.971, 0.990]	**0.930**	**[0.901, 0.959]**

to the average error rate results. Models with context-sensitive online variables performed marginally better than single-trial online variable models (Table 3.7). For example, the *LR:STOV+CSOV* model with interactions performed better than the *LR:STOV* model with interactions by 4.4% in precision, 3.7% in recall, and 3.7% in F_1 score. In terms of precision and F_1 score, all LR and MLR models performed better than the baseline model. For example, the *LR:STOV+CSOV* model with interaction structure achieved a 9.5% better precision rate and a 5.81% better F_1 score than the baseline model. MLR models performed similar or marginally better than LR models in all variable conditions. Adding interaction structures for the context-sensitive online variable condition also only marginally improved the models' performance.

Evaluation on Disengaged Student Subset

An ROC curve is a collection of true positive rate and false positive rate pairs based on different classifier thresholds. With an ROC curve, we can compute the area under the ROC curve statistic (AUC) as a robust overall evaluation metric for classifier effectiveness. However, regardless of how we conceptualize the positive cases (i.e., consider labels for either engaged or disengaged behaviors as the negative case), AUC is measurable only when both positive and negative cases exist in real label data. Therefore, analysis in this section measured the AUC score for prediction models with subset of student data that did not include those students who did not show a single disengaged behavior labels (17 of 25 students). Although this analysis setting may reduce the explanatory power of our results by conducting the analysis with a smaller-sized sample, we think it may provide additional information on how our models would perform with highly disengaged student data.

Results in Table 3.8 show AUC statistics from ROC curve and average error rates for students who exhibit at least one disengaged behavior event. In terms of AUC score, both LR models and MLR models performed substantially above the baseline. We confirmed that adding variables for context-sensitive information to models helped to improve the AUC scores and average errors. All MLR models performed better than LR models with AUC scores. However, adding interaction structures of fixed-effect variables did not increased the AUC scores. In terms of average error rate,

Table 3.8: The Area Under the ROC curve (AUC) statistic and average error rate for the disengaged student subset. Overall, including context-sensitive online variables improves the prediction performance (*STOV+CSOV*). In terms of AUC score, mixed effect models (MLR) perform better than logistic regression models (LR). Including additional interaction structure only improves the average error rate of LR models. (†: model includes interactions) (Scores in bold: the best model performance with a given variable set; NA: MLR models without any significant fixed-effect interactions; AUC: higher is better; Avg.Error: lower is better)

			AUC			
	STOV^a		**STOV^b**		**STOV+CSOV**	
Models	Mean	95% CI	Mean	95% CI	Mean	95% CI
Baseline	0.500	[0.500, 0.500]	0.500	[0.500, 0.500]	0.500	[0.500, 0.500]
LR	0.764	[0.695, 0.834]	0.764	[0.695, 0.834]	0.820	[0.760, 0.879]
LR†	0.759	[0.691, 0.828]	0.759	[0.691, 0.828]	0.807	[0.732, 0.883]
MLR	**0.822**	**[0.760, 0.884]**	**0.821**	**[0.759, 0.883]**	**0.865**	**[0.824, 0.907]**
MLR†	NA	NA	NA	NA	0.838	[0.767, 0.909]

			Avg. Error			
	STOV^a		**STOV^b**		**STOV+CSOV**	
Models	Mean	95% CI	Mean	95% CI	Mean	95% CI
Baseline	0.271	[0.172, 0.369]	0.283	[0.182, 0.384]	0.283	[0.182, 0.384]
LR	0.217	[0.168, 0.265]	0.225	[0.180, 0.271]	0.156	[0.106, 0.207]
LR†	0.206	[0.159, 0.253]	**0.205**	**[0.163, 0.248]**	**0.148**	**[0.104, 0.191]**
MLR	**0.201**	**[0.146, 0.256]**	0.212	[0.158, 0.266]	0.149	[0.103, 0.194]
MLR†	NA	NA	NA	NA	0.159	[0.112, 0.205]

MLR model only outperformed LR models in single-trial online variable condition. Adding interaction structures improved performance of LR models.

3.4.3 Feature Importance Analysis

Results from Section 3.4.2 illustrated that both LR and MLR models perform similar or better by including interaction structures for both STOV and STOV+CSOV conditions. In this section, we identify which features are more important than others in terms of improving the prediction accuracy. We also examine how different types of context-information variables are helpful for predicting particular response patterns of disengaged behaviors.

Individual Features

In order to identify more details about each feature's contribution to the prediction model, we conducted a feature ablation analysis that removed a single feature at a time from the model and evaluated the resulting loss in prediction accuracy, averaged across leave-one-subject-out cross-validation folds. All models used in this analysis included interaction structures (if possible) to illustrate more information on importance levels of various predictive features.

Table 3.9 presents the list of predictive features ordered by importance level, measured by changes in average classification error rate of selected models. Due to the discrepancy between the model-fitting process, which used AIC statistics to maximize goodness-of-fit through a step-wise process, and the prediction process, which used average classification error rates over cross-validation, some features provided better (*harmful*) or same (*neutral*) average error rates if they were taken off the model.

Overall, the results are in accord with what we already observed in Section 3.4.1. In Table 3.9, we can observe that adding context information to the LR model makes temporal features (`RTStart` or `RTFinish` series) less critical for prediction. In both single-trial online variable models (*LR:STOV* and *MLR:STOV*), the results show that temporal features and orthographic features are helpful for improving the models' prediction performance. In models containing context-sensitive online variables (*STOV+CSOV*), variables like semantic similarity among recent responses were considered as more important features than other STOV features.

Contextual Features

The results from previous sections repeatedly showed that introducing context-sensitive variables (CSOV) significantly improved the model's performance on predicting labels for disengaged behaviors. To get more detailed insights into which types of CSOVs are responsible for this improvement, we also evaluated the average increase in prediction error (across leave-one-subject-out folds) when a particular CSOV variable type is removed from the model. In this analysis, the *MLR:STOV+CSOV* model with interaction structure was used for evaluation. In the previous section, this model showed that its model structure is more stable than the corresponding LR model (Table 3.9).

For this analysis, we categorized context-sensitive online variables into two different types: (1) CSOVs related to orthographic similarity scores (e.g., (PattOrth.p3.mean)), and (2) CSOVs related with semantic similarity scores (e.g., (PattSem.p7.sd) and (RspLen.mean:PattSem.p7.sd)). This included fixed effect interaction structures that associated with the CSOV. Analyzing the model's performance without each type of CSOV will demonstrate how different relationship among the current response and previous responses can be useful for predicting different patterns of individual student's disengaged behaviors.

The results of our analysis are summarized in Figure 3.3, which compares the individual student level of average prediction error when each context-sensitive online

Table 3.9: Feature importance for LR and MLR models. Each cell indicates how much the average classification error rate across leave-one-subject-out cross-validation changes if the predictor is removed from the model. Temporal and orthographic features were commonly found to be helpful predictors in the single-trial online variable condition (*STOV*). Semantic and orthographic similarity based features were additional helpful predictors if context-sensitive online variables are added (*STOV+CSOV*), while temporal features were excluded. (^{c}: Context-information variable (CSOV); Variable names in bold: helpful variables that are commonly found in both LR and MLR models with the same variable set condition)

	LR:STOV		**MLR:STOV**		**LR:STOV+CSOV**		**MLR:STOV+CSOV**	
	Variable	Changes	Variable	Changes	Variable	Changes	Variable	Changes
helpful	RTStart.mean	+0.012	**RspLen.mean**	+0.013	**RspLen.mean**	+0.006	**PattSem.p3.mean**^{c}	+0.023
	RTFinish.final	+0.012	SimSem.final	+0.006	**SimSem.final**	+0.005	**RspLen.mean**	+0.015
	RspLen.final	+0.009	RTStart.final	+0.005	**RspLen.mean : PattOrth.p7.sd**^{c}	+0.005	**SimOrth.final**	+0.010
	RTFinish.final : RspLen.final	+0.009	**RTFinish.final**	+0.003	PattSem.p5.mean^c	+0.004	**RspLen.mean : PattOrth.p7.sd**^{c}	+0.010
	RspLen.mean	+0.003	SimOrth.final	+0.003	**PattSem.p3.mean**^{c}	+0.003	**RspLen.sd**	+0.008
	RspLen.sd	+0.003	**RspLen.sd**	+0.002	SimOrth.sd	+0.002	**SimSem.final**	+0.008
	SimOrth.mean	+0.003			**SimOrth.final**	+0.002	PattOrth.p7.sd^c	+0.003
	RspLen.sd : RspLen.final	+0.003			**RspLen.sd**	+0.002		
					RTFinish.final	+0.001		
neutral	RspLen.final : SimSem.final	0.000			RTStart.sd	0.000		
					RTFinish.sd	0.000		
					PattSem.p3.sd^c	0.000		
harmful	SimSem.final	-0.001			PattOrth.p5.mean^c	-0.001		
					PattOrth.p7.sd^c	-0.002		

variable type is included in the MLR model. Students are sorted by ascending proportion of disengaged responses (and thus, the baseline error rate of individuals). They are also labeled (x-axis) according to the same rubric used for Table 3.1, into lower disengaged response rates (O, under 30% — the first quartile of disengaged behaviors ratio); high disengagement rates (over 30%) with performance increase from including orthographic similarity measures to the model (A); or including semantic similarity measures to the model(B). Generally, we considered A group as students with many repeated responses and B group as students with semantically similar responses.

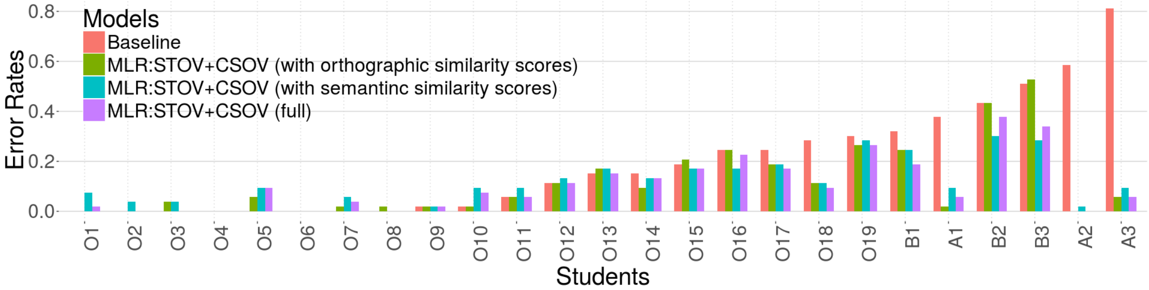


Figure 3.3: Subject-level error rate for predicting disengaged responses. The results show the significant reduction in error rate from adding context-sensitive information variables (CSOV) for students with high disengagement rates (A and B student groups). Features based on orthographic similarity score was helpful for decreasing error rate on A students. Semantic similarity based features helped to improve the model's performance on B students.

First, we see that there is wide variation in disengaged behavior across individual students. In particular, while many students showed little or no disengaged behavior, the right-most six students (the 3rd quartile of total samples) exhibited disengagement rates of 40% to 80%. Second, we see how prediction with linguistic context measures, either orthographic or semantic similarity related measures, gives a modest but consistent reduction in error for students with higher disengagement rates. Third, context-sensitive online variables that relate with orthographic similarity measures give a very significant reduction in error for some category A disengagement students, those who showed many repeated responses. Interestingly, using other CSOV based on semantic similarity measures provided similar improvements. Lastly, context-sensitive online variables that relate with semantic similarity measures provided relatively smaller improvements for capturing disengaged responses from B students (e.g., name of animals or fruits, parts of numeric sequences).

Thus, we can conclude that different types of linguistic context measures can be useful for improving the overall accuracy of the model and predicting complementary types of disengaged behaviors.

3.5 Discussion

Disengaged behaviors do not necessarily represent a particular cognitive state; such behavior could arise from a variety of mental activities that relate to perception, attention, reasoning, volition, and emotions (Cocea and Weibelzahl, 2009). Because of the complex nature of disengaged responses, the instruction for disengaged labeling was kept deliberately flexible, relying on human cognition to recognize potential occurrences. Thus, while features from log data may not explain why users engage or disengage with the system, such features can provide behavioral representations of student engagement in an intelligent tutoring system.

As a part of developing a web-based contextual word learning system, this study provides a starting point for modeling disengaged behavior during a vocabulary learning task. The performance of selected LR and MLR predictive models ranged from average error rates of 0.109 to 0.148, and recall of 0.956 to 0.981 for the entire participant dataset. AUC scores for students who showed at least one disengaged behaviors ranged from 0.759 to 0.865. These results are slightly better or similar to previous studies (Baker et al., 2004; Paquette et al., 2014; Cocea and Weibelzahl, 2009), depending on the different evaluation metrics that previous studies used. Thus, we argue that the modeling process and developed features proposed in this paper, such as single response variables (STOV) or context information variables (CSOV), are effectively predicting disengaged behaviors in a language learning ITS.

Recorded responses based on gaming or off-task behaviors would be less directly toward the task, while the careless mistakes include incorrect, but task-related responses. Our results showed that our model could capture different types of disengaged behaviors and responses. Identifying off-task responses are also important since it can also be adapted in a 'productive and constructive' way (Baker et al., 2013). For example, as the system observes disengaged behaviors, it can issue a prompt message that can bring the student's attention back to the system, such as text messages (Arroyo et al., 2007) or animated visual cues (Baker et al., 2006). Making students identify their disengaged behaviors can help to reduce the off-task states and increase learning gain (Baker et al., 2013). For ITS, being able to predict the student's off-task state would be an important step to making the tutoring system more adaptive.

Results in this paper suggest several more points for discussion. First, the parsimonious characteristics of MLR for selecting predictors led the model to use fewer predictors than LR models while achieving similar prediction accuracy. This result could be helpful for people who are designing the log data structure of ITS for predicting disengaged behaviors.

Second, our results on variable selection and feature importance analysis showed that some features are consistent regardless of regression technique we applied. In Section 3.4.1, we found that variables related to response length, semantic similarity between the target word and current response, and orthographic similarity between the target word and current response were common predictors across all models. In Section 3.4.3, we analyzed further details of individual feature importance. We found that variables related to response length and orthographic similarity between the target word and the current response (STOV) were commonly helpful predictors across all models. Variables for semantic similarity between the target word and the current response (STOV) and the average semantic similarity among previous responses and the current response (CSOV) were helpful in both logistic regression (LR) and mixed-effect logistic regression (MLR) models when context-based information variables were included (STOV+CSOV).

Third, variables related to response latency were less important predictors when context-based information was included in the model. In *MLR:STOV+CSOV* models, they did not incorporate response-time variables (`RTStart` and `RTFinish` measures) as significant predictors. The results from feature importance analysis also showed that response time variables are relatively less important when CSOVs are included in *LR:STOV+CSOV* models. This is interesting because in multiple previous ITS studies (Johns and Woolf, 2006; Beck, 2004), response time was considered an important predictor of behavior. However, this outcome may caused by the nature of the experimental task. For example, the Meaning Generation task from this study was not a time-sensitive task and would require much more cognitive resources from participants than other multiple choice questions (Beck, 2004). It would require additional study to see if CSOVs can provide better prediction performance than using variables related to response latency in a time-sensitive vocabulary learning task.

Fourth, orthographic similarity based context-sensitive variables were not as effective as semantic similarity based context-sensitive variables. For example, in Section 3.4.3, responses from `A` type students, who were considered as disengaged by providing repeated responses, were equally captured with both orthographic and semantic based context-sensitive features. We think this can be related to high semantic similarity among orthographically similar responses, as orthographically identical responses would elicit perfect semantic similarity scores from MESA. In future work, using different window sizes for each context-sensitive variable type could increase prediction performance by capturing more details of the student's mixed patterns of disengaged behaviors.

Fifth, combining models can increase the prediction performance. In this paper, because of the window size of context-sensitive online variables, *STOV+CSOV* models were trained and tested on the data without a sequence of the earliest questions. In a future study, multiple prediction models may be applied based on the student's current question order, such as using the single-trial online variables in early questions and using the model with context-sensitive online variables in later sequences.

Lastly, mixed-effect models performed similar to or only marginally outperformed fixed-effect models. This is likely because the correlations between responses in item-level and individual student-level may have already been captured by the context-based variables, which use an individual student's response history. Moreover, both step-wise process and hill-climbing algorithm, the two structure-learning algorithms used in this paper, rely on a model's AIC score. Using AIC score in the feature selection process deals with the trade-off between the complexity of model structure and goodness of fit. In other words, as we saw from the results of individual feature analysis (Section 3.4.3, the variable structures identified for the suggested models may not be the optimal model structure for maximizing the model's prediction performance. Further investigation is needed with more data collection and detailed feature analysis. Using more robust modeling methods, such as bootstrapping and Markov chain Monte Carlo for estimating the parameters of mixed-effect models, or different structure learning methods like LASSO or Elastic Net are other possible options to derive a more accurate prediction model.

3.6 Limitations

Several aspects of this study could be refined in future studies. First, the models here were limited to predict disengaged responses with log data from the pre-test task. In future work, we will examine how disengaged behaviors are associated with learning outcomes in other learning-oriented task settings. Second, disengagement labels generated from human judges do not address every type of disengaged behavior. Capturing other types of disengaged behaviors, such as a student cheating with his or her smartphone during the task or communicating with neighboring students, could be addressed by including external observations during the task or using more sophisticated latent-variable learning algorithms with larger datasets that can reveal these types of patterns from the log data. Third, acquiring disengagement labels can become more affordable in a crowdsourcing setting. However, it would require more carefully described instructions or simpler task design to collect reliable judgments from anonymous crowdworkers. Fourth, this study's relatively small number of participants also may not be representative of broader classes of behavior. For example, certain kinds of disengaged behaviors may be associated with different demographic groups, such as students with less experience with technology or poor core reading skills. Further studies with larger, more diverse student populations will help give a more complete picture of this complex phenomenon.

3.7 Conclusion

This study focused on developing and evaluating prediction models for students' disengaged responses in a meaning-generation task of a contextual word learning tutoring system. Our suggested model performed significantly better than the majority-class baseline in predicting disengaged behaviors. Compared to previous studies, the performance was similar or slightly better than disengaged behavior detectors on non-vocabulary tutoring systems. The models were developed based on data-driven methods. From different features that can be derived from a vocabulary learning system, we found that adding context-based features to the prediction model greatly improved prediction accuracy. Additional marginal improvements were also found when pairwise interaction structures were introduced to the prediction model with single-response-based variables. We also observed that context information variables based on linguistic relationships between responses were effective at capturing different types of disengaged responses, such as repeating the same answer or providing semantically sequential responses in the Meaning Generation task.

A central problem in the science of learning is to determine how much assistance (e.g., instructional help or support) to provide during learning (Teigen, 1994; Koedinger et al., 2013). Findings from this study will be useful for understanding the relationship between disengaged behaviors and learning outcomes. The results can be helpful to develop a real-time detector of student engagement which can make our contextual word learning (CWL) adapt more effectively to individual students' skill levels and performance.

3.8 Author Contributions

Sungjin Nam was the main contributor to the study, including developing the experimental tutoring system, designing the study, collecting the annotations for log data, conducting statistical analysis, and writing the manuscript. Dr. Kevyn

Collins-Thompson and Dr. Gwen Frishkoff contributed to designing the tutoring system, developing the questionnaire used in the experiment, and revising the manuscript. We also thank Dr. Charles Perfetti and his lab team at the University of Pittsburgh, particularly Adeetee Bhide and Kim Muth, and the helpful personnel at all of our partner schools.

Chapter 4

Capturing Partial Word Knowledge State with Word Embeddings and Semantic Differential Scales

4.1 Introduction[1]

Studies of word learning have shown that knowledge of individual words is typically not all-or-nothing. Rather, people acquire varying degrees of knowledge of many words incrementally over time, by exposure to them in context (Frishkoff et al.,

[1] This study was published as Sungjin Nam, Gwen Frishkoff, and Kevyn Collins-Thompson. 2017. Predicting short-and long-term vocabulary learning via semantic features of partial word knowledge. In *Proceedings of the 10th International Conference on Educational Data Mining*, pages 80–87.

2011). This is especially true for so-called “academic” words that are less common and more abstract — e.g., *pontificate*, *probity*, or *assiduous* (Frishkoff et al., 2016b). Binary representations and measures model word knowledge simply as correct or incorrect on a particular item (word), but in reality, a student’s knowledge level may reside between these two extremes. Thus, previous studies of vocabulary acquisition have suggested that students’ partial knowledge be modeled using a representation that adding an additional label corresponding to an intermediate knowledge state (Durso and Shore, 1991) or further, in terms of continuous metrics for semantic similarity (Collins-Thompson and Callan, 2007).

In addition, there are multiple dimensions to a word’s meaning (Osgood et al., 1957). Measuring a student’s partial knowledge on a single scale may only provide abstract information about the student’s general answer quality and not give enough information to specify *which* dimensions of word knowledge a student already has learned or needs to improve. In order to achieve detailed understanding of a student’s learning state, online learning systems should be able to capture a student’s “learning trajectory” that tracks their partial knowledge on a particular item over time, over multiple dimensions of meaning in a multidimensional semantic representation.

Hence, multidimensional representations of word knowledge can be an important element for building an effective intelligent tutoring system (ITS) for reading and language. Maintaining a fine-grained semantic representation of a student’s degree of word knowledge can be helpful for the ITS to design more engaging instructional content, more helpful personalized feedback, and more sensitive assessments (Ostrow et al., 2015; Van Inwegen et al., 2015). Selecting semantic representations to model, understand, and predict learning outcomes is important to designing a more effective and efficient ITS.

In this paper, we explore the use of multidimensional semantic word representations for modeling and predicting short- and long-term learning outcomes in a vocabulary tutoring system. Our approach derives predictive features using a novel application of existing methods in cognitive psychology combined with methods from natural language processing (NLP). First, we introduce a new multidimensional representation of a word based on the Osgood semantic differential (Osgood et al., 1957), an empirically based, cognitive framework that uses a small number of scales to represent latent components of word meaning. We compare the effectiveness of model features based on this Osgood-based representation to features based on a different representation, the widely-used Word2Vec word embedding (Mikolov et al., 2013). Second, we evaluate our prediction models using data from a meaning-generation task that was conducted during a computer-based intervention. Our study results demonstrate how similarity-based metrics based on rich semantic representation can be used to automatically evaluate specific components of word knowledge, track changes in the student's knowledge toward the correct meaning, and compute a rich set of features for use in predicting short- and long-term learning outcomes. Our methods could support advances in real-time, adaptive support for word semantic learning, resulting in more effective personalized learning systems.

4.2 Related Work

The present study is informed by three areas of research: (1) studies of partial word knowledge; (2) the Osgood framework for multiple dimensions of word meaning, and (3) computational methods for estimating semantic similarity.

4.2.1 Partial Word Knowledge

The concept of partial word knowledge has interested vocabulary researchers for several decades, particularly in the learning and instruction of "Tier 2" words (Yonek, 2008). Tier 2 words are low-frequency and typically have complex (multiple, nuanced) meanings. By nature, they are rarely learned through "one-shot" learning. Instead, they are learned partially and gaps are filled in over time.

Words in this intermediate state, neither novel nor fully known, are sometimes called "frontier words" (Dale, 1965). Durso and Shore operationalized the frontier word as a word the student had seen previously but was not actively using it (Durso and Shore, 1991). Based on this definition, the student may have had implicit memory of frontier words, such as general information like whether the word indicates a good or bad situation or refers a person or an action. They discovered that students are more familiar with frontier words than other types of words in terms of their sounds and orthographic characteristics (Durso and Shore, 1991). This previous work suggested that the concept of frontier words can be used to represent a student's partial knowledge states in a vocabulary acquisition task (Dale, 1965; Durso and Shore, 1991).

In some studies, partial word knowledge has been represented using simple, categorical labels, e.g., multiple-choice tests that include “partially correct” response options, as well as a single “best” (correct) response. In other studies, the student is presented with a word and is asked to say what it means (Adlof et al., 2016). The definition is given partial credit if it reflects knowledge that is partial or incomplete. For example, a student may recognize that the word *probity* has a positive connotation, even if she cannot give a complete definition. However, single categorical or score-based indicators may not explain which specific aspects of vocabulary knowledge the student is missing. Moreover, these studies relied on human ratings to evaluate students’ responses for unknown words (Durso and Shore, 1991). Although widely used in psychometric and psycholinguistic studies (Coltheart, 1981; Osgood et al., 1957), hiring human raters is expensive and may not be done in real time during students’ interaction with the tutoring system.

To address these problems, we propose a data-driven method that can automatically extract semantic characteristics of a word based on a set of relatively simple, interpretable scales. The method benefits from existing findings in cognitive psychology and natural language processing. In the following sections, we illustrate more details of related findings and how they can be used in an intelligent tutoring system setting.

4.2.2 Semantic Representation & the Osgood Framework

To quantify the semantic characteristics of a student’s intermediate knowledge of vocabulary, this paper uses a “spatial analogue” for capturing semantic characteristics of words. In (Osgood et al., 1957), Osgood investigated how the meaning of a word can be represented by a series of general semantic scales. By using these scales, Osgood suggested that the meanings of any word can be projected and explored in a continuous semantic space.

Osgood asked human raters to evaluate a set of words using a large number of scales (e.g., tall-short, fat-thin, heavy-light) and captured the semantic representation of a word (Osgood et al., 1957). Respondents gave Likert ratings, which indicated whether they thought that a word meaning was closer to one extreme (-3) or the other (+3), or basically irrelevant (0). A principal components analysis (PCA) was used to represent the latent semantic features that can explain the patterns of response to individual words within this task.

In our study, we suggest a method that can automatically extract similar semantic information that can project a word into a multidimensional semantic space. By using semantic scales selected from (Osgood et al., 1957), we verify if such representation of semantic attributes of words is useful for predicting students' short- and long-term learning.

4.2.3 Semantic Similarity Measures

Studies in NLP have suggested methods to automatically evaluate the semantic association between two words. For example, Markov Estimation of Semantic Association (MESA) (Collins-Thompson and Callan, 2007; Frishkoff et al., 2011) can estimate the similarity between words from a random walk model over a synonym network such as WordNet (Miller, 1995). Other methods like latent semantic analysis (LSA) are based on co-occurrence of the word in a document corpus. In LSA, semantic similarity between words is determined by using a cosine similarity measure, derived from a sparse matrix constructed from unique words and paragraphs containing the words (Landauer, 2006).

For this paper, we use Word2Vec (Mikolov et al., 2013), a widely used word embedding method, to calculate the semantic similarity between words. Word2Vec's technique (Li et al., 2015) transforms the semantic context, such as proximity between words, into a numeric vector space. In this way, linguistic regularities and patterns are encoded into linear translations. For example, using outputs from Word2Vec, relationships between words can be estimated by simple operations on their corresponding vectors, e.g., $Madrid - Spain + France = Paris$, or $Germany + capital = Berlin$ (Mikolov et al., 2013).

Measures from these computational semantic similarity tools are powerful because they can provide an automated method for evaluation of partial word knowledge. However, they typically produce a single measure (e.g., cosine similarity or Euclidean distance), representing semantic similarity as a one-dimensional construct. With such a measure, it is not possible to determine represent partial semantic knowledge and changes in knowledge of latent semantic features as word knowledge progresses from unknown to frontier to fully known. In following sections, we describe how we address this problem, using novel methods to to estimate the contribution of Osgood semantic features to individual word meanings.

4.2.4 Overview of the Study

Based on findings from existing studies, this study will suggest an automatized method for evaluating students' partial knowledge of vocabulary that can be used to predict students' short-term vocabulary acquisition and long-term retention. To investigate this problem, we will answer the following research questions with this paper.

- RQ1: Can semantic similarity scores from Word2Vec be used to predict students' short-term learning and long-term retention?

Previous studies in vocabulary tutoring systems tend to focus on how different experimental conditions, such as different spacing between question items (Pavlik and Anderson, 2005), difficulty levels (Ostrow et al., 2015), and systematic feedback (Frishkoff et al., 2016b), affect students' short-term learning. This study will answer how computationally estimated trial-by-trial scores in a vocabulary tutoring system can be used to predict students' short-term learning and long-term retention.

- RQ2: Compared to using regular Word2Vec scores, how does the model using Osgood's semantic scales (Osgood et al., 1957) as features perform for immediate and delayed learning prediction tasks?

As described in the previous section, the initial outcome from Word2Vec returns hundreds of semantic dimensions to represent the semantic characteristics of a word. Summary statistics for comparing such high-dimensional vectors, such as cosine similarity or Euclidean distance, only provide the overall similarity between words.

If measures from Osgood scales work in a similar level to models using regular Word2Vec scores for predicting students' learning outcomes, we can argue that it can be an effective method for representing students' partial knowledge of vocabulary.

4.3 Method

4.3.1 Word Learning Study

This study used a vocabulary tutoring system called Dynamic Support of Contextual Vocabulary Acquisition for Reading (DSCoVAR) (Frishkoff et al., 2016a)). DSCoVAR aims to support efficient and effective learning vocabulary in context. All participants accessed DSCoVAR in a classroom-setting environment by using Chromebook devices or the school's computer lab in the presence of other students.

Study Participants

Participants included 280 middle school students (6th to 8th grade) from multiple schools, including children from diverse socio-economic and educational backgrounds. Table 4.1 provides a summary of student demographics, including location (P1 or P2), age and grade level, sex. Location P1 is a laboratory school affiliated with a large urban university in the northeastern United States. Students from location P1 were generally of high socio-economic status (e.g., children of University faculty and staff). Location P2 includes three public middle schools in a southern metropolitan area of the United States. All students from location P2 qualified for free or reduced lunch. The study included a broad range of students so that the results of this analysis were more likely to generalize to future samples.

Table 4.1: The number of participants by grade and gender

	6th grade		**7th grade**		**8th grade**	
Group	Girl	Boy	Girl	Boy	Girl	Boy
P1	16	28	19	23	18	13
P2	53	51	12	6	21	20

Study Materials

DSCoVAR presented students with 60 SAT-level English words (also known as Tier 2 words). These "target words," lesser-known words that the students are going to learn, were balanced between different parts of speech, including 20 adjectives, 20 nouns, and 20 verbs. Based on previous works, we expected that students would have varying degrees of familiarity with the words at pre-test, but that most words would be either completely novel ("unknown") or somewhat familiar ("partially known") (Frishkoff et al., 2016a). This selection of materials ensured that there would be variability in word knowledge across students for each word and across words for each student.

In DSCoVAR, students learned how to infer the meaning of an unknown word in a sentence by using surrounding contextual information. Having more information in a sentence (i.e., a sentence with a high degree of contextual constraint) can decrease the uncertainty of inference. Instructions used for creating sentences for practice questions can be found in Appendix A.1.

In this study, the degree of sentence constraint was determined using standard cloze testing methods: quantifying the diversity of responses from 30 human judges when the target word is left as a fill-in-the-blank question. If the lexical entropy of collected responses was low, the sentence was considered as high constrained (or easy) sentence. For example, high constrained sentences collected less diverse responses from crowdworkers since more contextual information included in the sentence can restrict the range of likely responses. If these high constrained sentences are used as a stimuli in practice questions with target words, it would be easier for students to infer the meaning of the target word, since the sentence contains relatively more contextual information about the target word. Details for collecting cloze responses can be found in Appendix A.2.2.

Study Protocol

The word learning study comprised four parts: (1) a pre-test, which was used to estimate baseline knowledge of words, (2) a training session, where learners were exposed to words in meaningful contexts, (3) an immediate post-test, and (4) a delayed post-test, which occurred approximately one week after training.

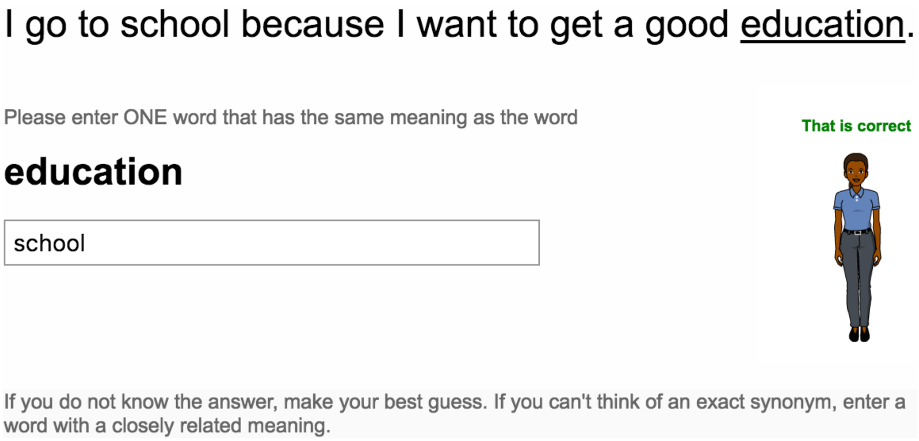
Pre-test The pre-test session was designed to measure the students' prior knowledge of the target words. For each target word, students were asked to answer two types of questions: familiarity-rating questions and synonym selection questions. In familiarity rating questions, students provided their self-rated familiarity levels (unknown, known, and familiar) for presented target words. In synonym-selection questions, students were asked to select a synonym word for the given target word from five multiple choice options. The outcome from synonym-selection questions provided more objective measures for students' prior domain knowledge of target words.

Training Approximately one week after the pre-test session, students participated in the training. During training, students learned strategies to infer the meaning of an unknown word in a sentence by using surrounding contextual information.

A training session consisted of two parts: an instruction video and practice questions. In the instruction video, students saw an animated movie clip about how to identify and use contextual information from the sentence to infer the meaning of an unknown word. In the practice question part, students could exercise the skill that they learned from the video. DSCoVAR provided sentences that included a target word with different levels of surrounding contextual information. The amount of contextual information for each sentence was determined by external crowd workers.

In the practice question part, each target word was presented four times within different sentences. Students were asked to type a synonym of the target word, which was presented in the sentence as underlined and bold. Over two weeks, students participated in two training sessions with a week's gap between them. Each training session contained the instruction video and practice questions for 30 target words. An immediate post-test session followed right after each training session.

Students were randomly selected to experience different instruction video conditions (full instruction video vs. restricted instruction video). Additionally, various difficulty level conditions and feedback conditions (e.g., DSCoVAR provides a feedback message to the student based on answer accuracy vs. no feedback) were tested within the same student. However, in this study, we focused on data from



I go to school because I want to get a good education.

Please enter ONE word that has the same meaning as the word

education

school

If you do not know the answer, make your best guess. If you can't think of an exact synonym, enter a word with a closely related meaning.

Figure 4.1: An example of a training session question. In this example, the target word is "education" with a feedback message for a high-accuracy response.

students who experienced a full instruction video and repeating difficulty conditions. Repeating difficulty conditions included questions with all high or medium contextual constraint levels. By doing so, we wanted to minimize the impact from various experimental conditions for analyzing post-test outcomes. Moreover, we filtered out response sequences that did not include all four responses for the target word. As a result, we analyzed 818 response sequences from 7,425 items in total.

Immediate and Delayed Post-test The immediate post-test occurred right after the students finished the training; the delayed post-test was conducted one week later. Data collected during the immediate and delayed post-tests were used to estimate short-and long-term learning, respectively. Test items were identical to those in the pretest session, except for item order, which varied across tests. For analysis of the delayed post-test data, we only used the data from target words for which the student

- bad – good
- passive – active
- powerful – helpless
- big – small
- helpful – harmful
- complex – simple
- fast – slow
- noisy – quiet
- new – old
- healthy – sick

Figure 4.2: Ten semantic scales used for projecting target words and responses (Osgood et al., 1957).

provided a correct answer in the earlier, immediate post-test session. As a result, 449 response sequences were analyzed for predicting the long-term retention.

4.3.2 Semantic Score-Based Features

We now describe the semantic features tested in our prediction models.

Semantic Scales

For this study, we used semantic scales from Osgood's study (Osgood et al., 1957). Ten scales were selected by a cognitive psychologist as being considered semantic attributes that can be detected during word learning (Figure 4.2). Each semantic scale consists of pairs of semantic attributes. For example, the *bad–good* scale can show how the meaning of a word can be projected on a scale with *bad* and *good* located at either end. The word's relationship with each semantic anchor can be automatically measured from its semantic similarity with these exemplar semantic elements.

Basic Semantic Distance Scores

To extract meaningful semantic information, we have applied the following measures that can be used to explain various characteristics of student responses for different target words. In this study, we used a pre-trained model for Word2Vec,[2] built based on the Google News corpus (100 billion tokens with 3 million unique vocabularies, using a negative sampling algorithm), to measure semantic similarity between words. The output of the pre-trained Word2Vec model contained a numeric vector with 300 hundred dimensions.

First, we calculated the relationship between word pairs (i.e., a single student response and the target word, or a pair of responses) in both the regular Word2Vec (W2V) score and the Osgood semantic scale (OSG) score. In the W2V score, the semantic relationship between words was represented with a cosine distance between word vectors, denoted as:

$$D_{w2v}(w_1, w_2) = 1 - |sim(V(w_1), V(w_2))|. \tag{4.1}$$

In this equation, the function V returned the vectorized representation of the word (w_1 or w_2) from the pre-trained Word2Vec model. By calculating the cosine similarity (sim) between two vectors, we could extract a single numeric similarity score between two words. This score was converted into a distance-like score by taking the absolute value of the cosine similarity score and subtracting from one.

For the OSG score, we extracted two different types of scores: a non-normalized

[2] API and pre-trained model for Word2Vec was downloaded from this URL: https://github.com/3Top/word2vec-api

score and a normalized score. A non-normalized score showed how a word is similar to a single anchor word (e.g., *bad* or *good*) from the Osgood scale.

$$S_{osg}^{non}(w, a_{i,j}) = sim(V(w), V(a_{i,j})) \tag{4.2}$$

$$D_{osg}^{non}(w_1, w_2; a_{i,j}) = |S_{osg}^{non}(w_1, a_{i,j})| - |S_{osg}^{non}(w_2, a_{i,j})| \tag{4.3}$$

In equation 4.2, $a_{i,j}$ represents the j-th anchor word in the i-th Osgood scale. The similarity between the anchor word and the evaluating word w was calculated with cosine similarity of Word2Vec outcomes for both words. In a non-normalized setting, the distance between two words given by a particular anchor word was calculated by the difference of absolute cosine similarity scores (equation 4.3).

The second type of OSG score is a normalized score. By using Word2Vec's ability to represent the semantic relationship between words through simple arithmetic calculations of word vectors (Mikolov et al., 2013), the normalized OSG score provided a relative location of the word from two anchor ends of the Osgood scale.

$$S_{osg}^{nrm}(w, a_i) = sim(V(w), V(a_{i,1}) - V(a_{i,2})) \tag{4.4}$$

$$D_{osg}^{nrm}(w_1, w_2; a_i) = |S_{osg}^{nrm}(w_1, a_i) - S_{osg}^{nrm}(w_2, a_i)| \tag{4.5}$$

In equation 4.4, the output represents the cosine similarity score between the word w and two anchor words ($a_{i,1}$ and $a_{i,2}$). For example, if the cosine similarity score of $S_{osg}^{nrm}(w, a_i)$ is close to -1, it means the word w is close to the first anchor word $a_{i,1}$. If the score is close to 1, it is vice versa. In equation 4.5, the distance between two words was calculated as the absolute value of the difference between two cosine similarity measures.

Deriving Predictive Features

Based on semantic distance equations explained in the previous section, this section explains examples of predictive features that we used to predict students' short-term learning and long-term retention.

Distance Between the Target Word and the Response. For regular Word2Vec score models and Osgood scale score models, distance measures between the target word and the response (by using equations 4.1 and 4.5) were used to estimate the accuracy of the response to a question. This feature represents the trial-by-trial answer accuracy of a student response. Each response sequence for the target word contained four distance scores.

Difference Between Responses. Another feature that we used in both types of models was the difference between responses. This feature captures how a student's current answer is semantically different from the previous response. From each response sequence, we could extract three derivative scores from four responses. An example for deriving distance based features is illustrated at Figure 4.3.

Convex Hull Area of Responses. Alternative to the difference between responses feature, Osgood scale models were also tested with the area size of convex hull that can be generated by responses calculated with non-normalized

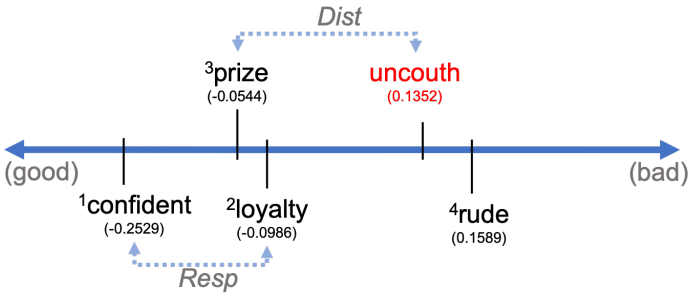


Figure 4.3: An example of distance based features by using normalized OSG scores from equation 4.4. *Dist* represents the distance between the target word and the response. *Resp* represents the difference between responses. This example illustrates how the student's responses get closer to the the target word *uncouth* over trials (noted as superscript numbers 1-4) in a *good–bad* Osgood scale.

Osgood scale scores (equation 4.3). For example, for each Osgood scale, a non-normalized score provided two-dimensional scores that can be used for geometric representation. By putting the target word in an origin position, a sequence of responses can create a polygon that can represent the semantic area that the student explored with responses. Since some response sequences were unable to generate the polygon by including less than three unique responses, we added a small, random noise that uniformly distributed (between -10^{-4} and 10^{-4}) to all response points. Additionally, a value of 10^{-20} was added to all convex hull area output to create a visible lower-bound value.

Unlike the measure of difference between responses, this feature also considers angles that can be created between responses and the target word. This representation can provide more information than just using difference between responses. An example of this representation can be found in Figure 4.3.2.

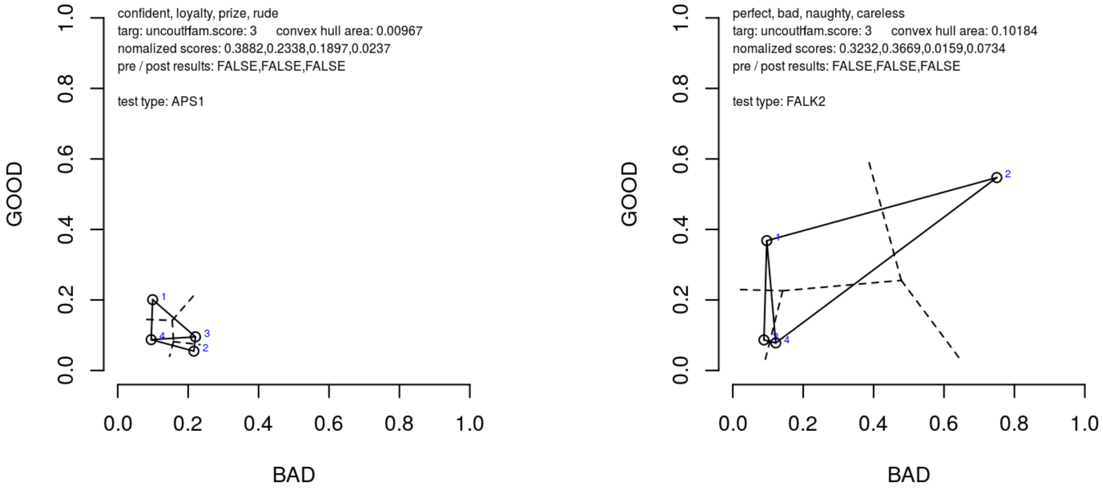


Figure 4.4: Response sequences represented in a non-normalized Osgood scale (bad-good) for the target word ‘uncouth’. Convex hull is calculated from the area inside the polygon that generated by response points.

4.3.3 Building Prediction Models

To predict students’ short-term learning and long-term retention, we used a mixed-effect logistic regression model (MLR). MLR is a general form of logistic regression model that includes random effect factors to capture variations from repeated measures.

Off-line Variables

Off-line variables capture item- or subject-level variances that can be observed repeatedly from the data. In this study, we used multiple off-line variables as random effect factors.

First, results from familiarity-rating and synonym-selection questions from the pre-test session were used to include item- and subject-level variances. Both variables include information on the student's prior domain knowledge level for target words. Second, the question difficulty condition was considered as an item group level factor. In the analysis, sentences for the target word that were presented to the student contained the same difficulty level, either high or medium contextual constraint levels, over four trials. Third, a different experiment group was used as a subject group factor. As described in Section 4.3.1, this study contains data from students in different institutions in separate geographic locations. The inclusion of these participant groups in the model can be used to explain different short-term learning outcomes and long-term retention by demographic groups.

Model Building

In this study, we compared the performance of MLR models with four different feature types. First, the baseline model was set to indicate the MLR model's performance without any fixed effect variables but only with random intercepts. Second, the response time model was built to be compared with semantic score-based models. Many previous studies have used response time as an important predictor of student engagement and learning (Beck, 2005; Ma et al., 2016). In this study, we used two types of response time variables, the latency for initiating the response and finishing typing the response, as predictive features. Both variables were measured in milliseconds over four trials and natural log transformed for the analysis. Third, semantic features from regular Word2Vec scores were used as predictors. This model was built to show how semantic scores from Word2Vec can be useful for predicting students' short- and long-term performance in DSCoVAR. Lastly, Osgood scale-based features were used as predictors. This model was compared with the regular Word2Vec score model to examine the effectiveness of using Osgood scales for evaluating students' performance in DSCoVAR. For these semantic-score based models, we tested out different types of predictive features that were described in Section 4.3.2. All models shared the same random intercept structure that treated each off-line variable as an individual random intercept.

For Osgood scale models, we also derived reduced-scale models. Reduced-scale models were compared with the full-scale model, which uses all ten Osgood scales. In this case, using fewer Osgood scales can provide easier interpretation of semantic analysis for intelligent tutoring system users.

Model Evaluation

To compare performance between different models, this study used various evaluation metrics, including AUC (an area under the curve score from a response operating characteristic (ROC) curve), F_1 (a harmonic mean of precision and recall), and error rate (a ratio of the number of misclassified items over total items). 95% confidence interval of each evaluation metric was calculated from the outcome of a ten-fold cross-validation process repeated over ten times.

To select the semantic score-based features for models based on regular Word2Vec scores and Osgood scale scores, we used rankings from each evaluation metric. The model with the highest overall rank (i.e., sum the ranks from AUC, F_1, and error rate, and select the model with the lowest rank-sum value) was considered the best-performing model for the score type (i.e., models based on the regular Word2Vec score or Osgood scale score). More details on this process will be illustrated in the next section.

4.4 Results

4.4.1 Selecting Models

In this section, we selected the best-performing model based on the models' overall ranks in each evaluation metric. All model parameters were trained in each fold of repeated cross-validation. We calculated 95% confidence intervals for comparison. To calculate the confidence interval of F_1 and error rate measures, the maximum (F_1) and minimum (error rate) scores of each fold were extracted. These maximum and minimum values were derived from applying multiple cutoff points to the mixed-effect regression model.

Predicting Immediate Learning

First, we built models that predict the students' immediate learning from the immediate post-test session. From models based on regular Word2Vec scores (W2V), the model with the distance between the target and responses and the difference between responses (*Dist+Resp*) provided the highest rank from various evaluation metrics (Table 4.2). From models based on Osgood scales (OSG), the model with the difference between responses (*Resp*) provided the highest rank.

The selected W2V model provided significantly better performance than the baseline model. The selected OSG model also showed significantly better performance than the baseline model, except for the AUC score. The selected W2V model was significantly better than the model using response time features in the AUC score and error rates.

The selected W2V model showed significantly better performance than the OSG model only with the AUC score. Figure 4.5 shows that the W2V model has a slightly larger area under the ROC curve than the OSG model. In the precision and recall curve, the selected W2V model provides more balanced trade-offs between precision and recall measures. The selected OSG model outperforms the W2V model in precision only in a very low recall measure range.

Table 4.2: Ranks of predictive feature sets for regular Word2Vec models (W2V) and Osgood score models (OSG) in the immediate post-test data. All models are significantly better than the baseline model. (Bold: the selected model with highest overall rank.)

	W2V models		
Features	AUC	F_1	Err
baseline	0.68 [0.67, 0.69] (5)	0.74 [0.73, 0.74] (5)	0.33 [0.33, 0.34] (5)
RT	0.69 [0.68, 0.70] (4)	0.75 [0.75, 0.76] (3)	0.31 [0.31, 0.32] (4)
Dist	0.72 [0.71, 0.74] (1)	0.76 [0.75, 0.76] (2)	0.29 [0.28, 0.30] (2)
Resp	0.70 [0.69, 0.71] (3)	0.75 [0.74, 0.76] (4)	0.31 [0.30, 0.32] (3)
Chull	NA	NA	NA
Dist+Resp	**0.72 [0.71, 0.73] (2)**	**0.76 [0.75, 0.77] (1)**	**0.29 [0.28, 0.30] (1)**
Dist+Chull	NA	NA	NA
	OSG models		
Features	AUC	F_1	Err
baseline	0.68 [0.67, 0.69] (5)	0.74 [0.73, 0.74] (5)	0.33 [0.33, 0.34] (7)
RT	0.69 [0.68, 0.70] (2)	0.75 [0.74, 0.76] (2)	0.31 [0.31, 0.32] (2)
Dist	0.67 [0.66, 0.68] (7)	0.73 [0.73, 0.74] (7)	0.33 [0.32, 0.34] (6)
Resp	**0.69 [0.68, 0.70] (1)**	**0.75 [0.75, 0.76] (1)**	**0.31 [0.30, 0.32] (1)**
Chull	0.69 [0.68, 0.70] (3)	0.74 [0.73, 0.75] (4)	0.32 [0.31, 0.33] (4)
Dist+Resp	0.68 [0.67, 0.69] (4)	0.74 [0.73, 0.75] (3)	0.31 [0.31, 0.32] (3)
Dist+Chull	0.67 [0.66, 0.68] (6)	0.74 [0.73, 0.74] (6)	0.33 [0.32, 0.34] (5)

Predicting Long-Term Retention

We also built prediction models to predict the students' long-term retention in the delayed post-test session. In this analysis, a student response was included only when the student provided correct answers to the immediate post-test session questions. Among W2V score-based models, the best-performing model contained the same feature types as the immediate post-test results (Table 4.3). By using the distance between the target and responses and difference between responses (*Dist+Resp*), the model achieved significantly better performance than the baseline model, except for the AUC score.

For OSG models, the model with a convex hull area of responses (*Chull*) provided the highest overall rank from evaluation metrics (Table 4.3). The results were significantly better than the baseline model, and marginally better than the W2V model. Both selected W2V and OSG models were marginally better than the response time model, except the error rate of the OSG model was significantly better.

In Figure 4.5, the selected W2V model slightly outperforms the OSG model in mid-range true positive rates, while the OSG model performed slightly better in a higher true positive area. Precision and recall curves show similar patterns to those we observed from the immediate post-test prediction models. The OSG model only outperforms the W2V model in a very low recall value area.

Comparing Models

Compared to the selected W2V model in the immediate post-test condition, the selected W2V model in the delayed post-test retention condition showed a

Table 4.3: Ranks of predictive feature sets for W2V and OSG models in the delayed post-test data. All models are significantly better than the baseline model. (Bold: the selected model with highest overall rank.)

	W2V models		
Features	AUC	F$_1$	Err
baseline	0.65 [0.64, 0.67] (5)	0.75 [0.74, 0.76] (5)	0.33 [0.32, 0.34] (5)
RT	0.67 [0.65, 0.68] (3)	0.76 [0.76, 0.77] (4)	0.31 [0.30, 0.32] (3)
Dist	0.66 [0.64, 0.68] (4)	0.77 [0.76, 0.78] (3)	0.31 [0.30, 0.32] (4)
Resp	0.69 [0.67, 0.71] (1)	0.77 [0.76, 0.78] (2)	0.30 [0.29, 0.31] (2)
Chull	NA	NA	NA
Dist+Resp	**0.68 [0.66, 0.70] (2)**	**0.78 [0.77, 0.79] (1)**	**0.30 [0.29, 0.31] (1)**
Dist+Chull	NA	NA	NA

	OSG models		
Features	AUC	F$_1$	Err
baseline	0.65 [0.64, 0.67] (5)	0.75 [0.74, 0.76] (7)	0.33 [0.32, 0.34] (7)
RT	0.67 [0.65, 0.68] (3)	0.76 [0.76, 0.77] (5)	0.31 [0.30, 0.32] (5)
Dist	0.66 [0.64, 0.68] (4)	0.78 [0.77, 0.79] (3)	0.30 [0.29, 0.31] (3)
Resp	0.63 [0.61, 0.65] (7)	0.76 [0.75, 0.77] (6)	0.32 [0.31, 0.33] (6)
Chull	**0.69 [0.68, 0.71] (1)**	**0.78 [0.77, 0.79] (2)**	**0.28 [0.27, 0.29] (1)**
Dist+Resp	0.64 [0.62, 0.66] (6)	0.77 [0.76, 0.78] (4)	0.31 [0.29, 0.32] (4)
Dist+Chull	0.69 [0.67, 0.71] (2)	0.78 [0.78, 0.79] (1)	0.29 [0.27, 0.30] (2)

significantly lower AUC score, marginally higher F_1 score, and marginally higher error rate. In terms of OSG models, the selected OSG model for delayed post-test retention showed a significantly better F_1 score and error rates than the selected OSG model in the immediate post-test condition. Based on these results, we can argue that Osgood scale scores can be more useful for predicting student retention in the delayed post-test session than predicting the outcome from the immediate post-test.

In terms of selected feature types, the best-performing OSG models used features based on the difference between responses (*Resp*) or the convex hull area (*Chull*) that was created from the relative location of the responses. On the other hand, selected W2V models used both the distance between the target word and responses and difference between responses (*Dist+Resp*). When we compared both W2V and OSG models using the difference between responses feature, we found that performance is similar in the immediate post-test data. However, the OSG model was significantly better in the delayed post-test data. These results show that Osgood scale scores can be more useful for representing the relationship among response sequences.

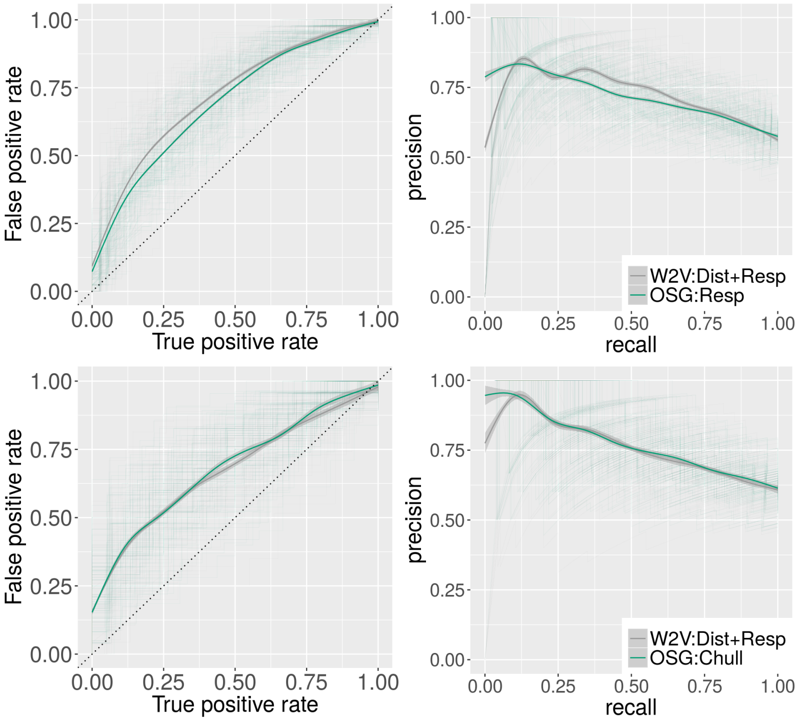


Figure 4.5: ROC curves and precision and recall curves for selected immediate post-test prediction models (left) and delayed post-test prediction models (right). Curves are smoothed out with a local polynomial regression method based on repeated cross-validation results.

4.4.2 Failure Analysis

From the previous section, we identified that the regular Word2Vec score based model and the Osgood scale score based model provide error rates around 30%. However, selected predictive models contain some errors. Different false predictions can be made on particular response patterns or participant groups. In this section, we compared patterns of false positive and false negative errors observed in the repeated cross-validation process from the regular Word2Vec score based model (W2V), the Osgood semantic scales based model (OSG) , and the response time based model (RT). For W2V and OSG models, we will use the selected models from the previous section.

Overall Comparison Between Models

To analyze, we used 0.5 probability as a cutoff to distinguish positive and negative predictions from the prediction model. For example, if the mixed-effect logistic regression model returned the probability of higher than 0.5, we considered the prediction as a positive prediction. By using the fixed threshold, the outcome can be different from the previous section, which compared multiple cutoff points that can maximize or minimize scores of F_1 measure and error rates. Confidence intervals of outcomes from repeated cross-validation process will be compared between models.

We found that the OSG model show significant higher recall rates (W2v: 0.71 [0.70, 0.73] vs. OSG: 0.75 [0.74, 0.77]) in predicting retention, while the W2V model have marginally higher precision rate for predicting the immediate post-test outcomes (W2V: 0.68 [0.66, 0.69] vs. OSG: 0.66 [0.65, 0.67]). The OSG model showed marginally higher false discovery rate (W2V: 0.49 [0.46, 0.52] vs. OSG: 0.53 [0.50, 0.56] – for the immediate post-test outcome; W2V: 0.53 [0.49, 0.57] vs. OSG: 0.54 [0.51, 0.58] – for the delayed post-test retention), which is a ratio of the number of false positive cases over true positive cases. False omission rate, which is a ratio of the number of false negative cases over true negative cases, was higher in the OSG model with the immediate post-test data (W2V: 0.60 [0.56, 0.64] vs. OSG: 0.61 [0.57, 0.65]). However, with the delayed post-test retention data, the W2V model showed higher false omission rate W2V: 0.93 [0.80, 1.06] vs. OSG: 0.87 [0.74, 1.01]). This means that the OSG model made relatively more false positive errors. False negative errors were more likely to be observed in the delayed post-test results.

The RT model showed similar performance with W2V and OSG models in the immediate post-test data. However, the RT model performed significantly inferior in recall (0.74 [0.72, 0.76]) and error rate (0.40 [0.39, 0.41]) from the OSG model in the delayed post-test data (recall: 0.81 [0.79, 0.83], error rate:0.37 [0.36, 0.38]).

Hand-picked Examples

To explore more details of these false prediction cases, we handpicked some false positive and false negative examples. We could find some false positive cases occur in repetitive but higher quality answers. For the target word *uncouth*, some students provided *rude* as a single unique response that repeated over all four trials. However, while both the W2V and the OSG model predicted these response sequences as positive case, some items turned out as false positives in the delayed-post test.

Table 4.4: False prediction examples. For false positive cases, we could see that the model tend to predict as the student learned the meaning of the word in the delayed-post test when he or she provided the (correct) repetitive responses. On the contrary, there were also some false negative cases when the responses tend to unique between each other. These results show that including additional engagement related or linguistic features would benefit the model's performance.

False Pred. Type	**Target word**	**Responses**
False Positive	uncouth	rude–rude–rude–rude
	gramercy	thanks–thanks–thanks–thanks
False Negative	uncouth	disgusting–embarrassing–limits–disrespectful
	gramercy	thankfulness–grateful–thank–gratefulness

Similar examples were observed in the target word *gramercy* with repetitively responding by entering *thanks*.

Some False negative cases showed an opposite pattern. For target words *uncouth* and *gramercy*, few false negative cases were found in response sequences containing more than two unique responses (Table 4.4). These results indicate that our prediction models may benefit from including additional features, such as response time or orthographic similarity, to address more information on students' engagement states and further linguistic characteristics (Nam et al., 2018) while answering the questions.

Comparison of Off-line Variables

We also explored how prediction models performed differently by individual subject or item group factors. As a result, we could observe that if the student is better prepared with the target word (e.g., higher familiarity score or provided correct answer in the pre-test's synonym selection question), confronted with easier questions (e.g., questions with high contextual constrain conditions), and recruited from the university owned laboratory school (*P1* group), models tend to yield lower error rates, false discovery rates, and false omission rates. For example, the OSG model with the immediate post-test data provided decreasing error rates by higher familiarity score (0.39 [0.38, 0.41], 0.33 [0.31, 0.35], and 0.28 [0.26, 0.30] (low to high)). Similar pattern was observed with the delayed post-test retention data (0.45 [0.44, 0.47], 0.37 [0.34, 0.39], and 0.21 [0.19, 0.24], low to high familiarity scores). All differences between familiarity scores were significant.

Another interesting comparison was by the number of unique responses provided from the student. If one or two unique responses existed in the response sequence, the OSG model with the immediate post-test data provided significantly higher recall and precision scores than if the number of unique responses were greater than two (recall: 0.54 [0.52, 0.56] vs. 0.93 [0.92, 0.94], precision: 0.61 [0.59, 0.64] vs. 0.68 [0.67, 0.70]). False discovery rates (0.69 [0.62, 0.76] vs. 0.48 [0.45, 0.52]) and error rates (0.38 [0.37, 0.39] vs. 0.32 [0.31, 0.33]) were also significantly lower in less number of unique response cases. However, the false omission rate was higher in this case (0.63 [0.59, 0.68] vs. 0.68 [0.54, 0.82]). Similar patterns were observed with the delayed post-test data (recall: 0.52 [0.49 0.55] vs. 0.99 [0.99, 1.00], precision: 0.59 [0.56 0.63] vs. 0.69 [0.67, 0.70], false discovery rates: 0.96 [0.75 1.17] vs. 0.48 [0.45, 0.52], false omission rates: 0.87 [0.73 1.00] vs. 2.33 [2.10, 2.56], and error rates: 0.42 [0.40 0.45] vs. 0.32 [0.30, 0.34]).

From these results, we can conclude that models performed better with the response data collected from better prepared student groups, easier questions, or containing less number of unique responses.

4.4.3 Comparing the Osgood Scales

To identify which Osgood scales are more helpful than others for predicting students' performance, we conducted a scale-wise importance analysis. The results from this section reveal which Osgood scales are more important than others, and how the performance of prediction models with a reduced number of scales is comparable with the full-scale model.

Identifying More Important Osgood Scales

In this section, based on the selected Osgood score model from Section 4.4.1, we identified the level of contribution for features based on each Osgood scale. For example, the selected OSG model for predicting the immediate post-test data uses the difference between responses in ten Osgood scales as features. In order to diagnose the importance level of the first scale (*bad–good*), we can retrain the model with features based on the nine other scales and compare the performance of the newly trained model with the existing full-scale model.

In Table 4.5, we picked the top five scales that were important in individual prediction tasks. We found that *big-small*, *helpful-harmful*, *complex-simple*, and *fast-slow* were commonly important Osgood scales for predicting students' performance in immediate post-test and delayed post-test sessions. Scales like *bad-good* and *passive-*

active were only important scales in the immediate post-test prediction. Likewise, *new-old* was an important scale only in the delayed post-test prediction.

Table 4.5: Scale-wise importance of each Osgood scale. Scales were selected based on the sum of each evaluation metric's rank. (Bold: Osgood scales that were commonly important in both prediction tasks; *: top five scales in each prediction task including tied ranks)

	Imm. post-test				**Del. post-test**			
Scales	AUC	F_1	Err	All	AUC	F_1	Err	All
bad-good	1	1	1	1*	4	10	4	6
passive-active	2	4	3	2*	8	6	6	7
powerful-helpless	7	9	6	7.5	10	8	10	10
big-small	3	3	4	3*	1	3	2	2*
helpful-harmful	4	6	5	5.5*	2	1	1	1*
complex-simple	8	5	2	5.5*	3	5	7	4.5*
fast-slow	5	2	7	4*	6	4	3	3*
noisy-quiet	6	8	8	7.5	7	9	9	9
new-old	9	7	9	9	5	2	8	4.5*
healthy-sick	10	10	10	10	9	7	5	8

Performance of Reduced Models

Based on the scale-wise importance analysis results, we built reduced-scale models that only contain features with more important Osgood scales. The prediction performance of reduced-scale models was similar or marginally better than full-scale OSG models. For example, the OSG model for predicting the immediate post-test outcome with the top two scales (*bad–good* and *passive–active*) were marginally better than the full-scale model (AUC: 0.71 [0.70, 0.72], F_1: 0.76 [0.75, 0.77], error rate: 0.30 [0.29, 0.30]). Similar results were observed for predicting retention in the delayed post-test (selected scales: *helpful–harmful*, *big–small*) (AUC: 0.71 [0.69, 0.72], F_1: 0.79 [0.78, 0.80], error rate: 0.28 [0.27, 0.29]). Although differences were small, the results indicate that using a small number of Osgood scales can be similarly effective to the full-scale model.

4.5 Discussion and Conclusions

In this paper, we introduced a novel semantic similarity scoring method that uses predefined semantic scales to represent the relationship between words. By combining Osgood's semantic scales (Osgood et al., 1957) and Word2Vec (Mikolov et al., 2013), we could automatically extract the semantic relationship between two words in a more interpretable manner. To show this method can effectively represent students' knowledge in vocabulary acquisition, we built prediction models that can be used to predict the student's immediate learning and long-term retention. We found that our models performed significantly better than the baseline and the response-time-based models. Our model also performed significantly worse than the Word2Vec model in predicting students' performance in the immediate post-test task, but marginally better with the delayed post-test task. In the future, we believe results from using an Osgood scale-based student model could be used to provide a more personalized learning experience, such as generating questions that can improve an individual student's understanding for specific semantic attributes.

Based on our findings, we have identified the following points for further discussion. First, in Section 4.4.1, we found that models using Osgood scale scores perform similarly with models using regular Word2Vec scores for predicting students' long-term retention of acquired vocabulary. However, we think our models can be further improved by incorporating additional features. For example, non-semantic score-based features like response time and orthographic similarity among responses can be useful features for capturing different patterns of false predictions of current models. Moreover, some general measures to capture a student's meta-cognitive or linguistic skills could be helpful to explain different retention results found even if students provided the same response sequences. Similarly, in Section 4.4.1, we found that Osgood scores can be a better metric to characterize the relationship between responses in terms of predicting students' retention. A composite model that uses both regular Word2Vec score-based feature (target-response distance) and Osgood scale score-based feature (response-response distance) may also provide better prediction performance.

Second, we found that models with a reduced number of Osgood scales performed marginally better than the full-scale model. However, differences were very small. Since this study only used some of the semantic scales from Osgood's study (Osgood et al., 1957), further investigation would be required to examine the validity of these scales, including other scales not used for this study, for capturing the semantic attributes of student responses during vocabulary learning.

Also, there were some limitations in the current study and areas for future work. First, expanding the scope of analysis to the full set of experimental conditions used in the study may reveal more complex interactions between these conditions and students' short- and long-term learning. Second, this study used a fixed threshold of 0.5 for investigating false prediction results. However, an optimal threshold for each participant group or prediction model could be selected, especially if there are different false positive or negative patterns observed for different groups of students. Lastly, this study collected data from a single vocabulary tutoring system that was used in a classroom setting. Applying the proposed method to data that was collected from a non-classroom setting or other vocabulary learning system would be useful to show the generalization of our suggested method.

4.6 Author Contributions

Sungjin Nam was the main contributor to the study, including developing the experimental tutoring system, designing the study, conducting statistical analysis, and writing the manuscript. Dr. Kevyn Collins-Thompson and Dr. Gwen Frishkoff contributed to designing the tutoring system, developing the cloze sentences used in the experiment, and revising the manuscript.

Chapter 5

Attention-based Learning Models for Predicting Contextual Informativeness

5.1 Introduction

We learn the vast majority of our new vocabulary with significant help from context. Humans acquire the meanings of unknown words partially and incrementally Frishkoff et al. (2008) by repeated exposure to clues in the surrounding text or conversation. As part of literacy training, contextual word learning methods can help students by teaching them different techniques for inferring the meaning of unknown words by recognizing and exploiting semantic cues such as synonyms and cause-effect relationships (Heilman et al., 2010). However, not all contexts are

1) My friends, family, and I all really like *tesgüino.*
2) There is a bottle of *tesgüino* on the table.
3) Brewers will ferment corn kernels to make *tesgüino.*

Figure 5.1: The three sentences have the same length but provide very different information – contextual informativeness – about the meaning of the target word, *tesgüino.* We introduce a new dataset and computational models to quantify the degree and nature of this target-specific informativeness for learning.

equally informative for learning a word's meaning. As Figure 5.1 shows, there can be wide variation in the amount and type of information about a 'target' word to be learned, via semantic constraints implied by the context.

Humans are very good at 'few-shot learning' of new vocabulary from such examples, but the instructional *quality* of initial encounters with a new word is critical. Identifying the degree and nature of contextual informativeness in authentic learning materials is an important problem to solve for designing effective curricula for contextual word learning (Webb, 2008; Frishkoff et al., 2015). As we elaborate in Section 5.2, predicting and characterizing contextually informative passages is quite different from other context-based prediction tasks such as n-gram prediction or cloze completion. It also has broad potential applications for both human and machine learning.

In this study, we introduce a new dataset and models for predicting the degree and nature of the *contextual informativeness* of a passage with respect to the meaning of a target word to be learned. First, we introduce a new dataset of contextual informativeness ratings for learning target words in single-sentence contexts. Second, we show that recent advances in deep semantic representations are highly effective for this task. We develop models based on ELMo (Peters et al., 2018) and

BERT (Devlin et al., 2019), combined with an attention layer. We demonstrate that the learned models generalize effectively across very different datasets, showing state-of-the-art performance on both our single-sentence context dataset and the multi-sentence context dataset of Kapelner et al. (2018). Third, beyond predicting a score, we provide a quantitative evaluation of how models capture the contributions of a particular passage to correctly infer a target word's meaning, demonstrating that attention activation provides fine-grained, interpretable characterizations of contextual informativeness. Further, using the dataset of Santus et al. (2015), we show that informativeness learned through this mechanism is robust across various semantic relations. Our results are applicable not only to developing educational curricula for vocabulary instruction, but also to NLP tasks like few-shot learning of new words or concepts from text.

5.2 Related Work

Our study focuses on measuring *contextual informativeness* with respect to a specific target word. This has some connection to the *predictability* of the word, or the "likelihood of a word occurring in a given context" in psycholinguistics (Warren, 2012), but with important differences that we describe further below. Generic definitions of informativeness have been defined as the density of relevant information that exists in a dialog (Gorman et al., 2003), or the number of different semantic senses included in a sentence (Lahiri, 2015). Entropy-based measures like *KL-divergence* have been used to represent reader 'surprise' from reading a new text compared to their prior knowledge (Peyrard, 2018). Compared to these generic definitions, the contextual aspect relative to a target concept is a critical distinction: different words in the same sentence may have very different degrees of semantic constraint imposed by the rest of the sentence.

Computational lexical semantics has long studied how to characterize word meaning in context (Turney and Pantel, 2010; Mikolov et al., 2013; Pennington et al., 2014) and how contextual informativeness can be used to select word meaning (Szarvas et al., 2013; Melamud et al., 2016; McCann et al., 2017; Peters et al., 2018). However, these models typically assume informative contexts are given, and do not predict or characterize the varying degrees of informativeness with respect to a target word or concept.

5.2.1 Contextual Informativeness in Reading and Literacy Research

Beck et al. (1983) characterized the informativeness of contexts for learning new words, distinguishing between pedagogical (specifically chosen to teach meaning) vs. natural (unintentionally informative). They reviewed two basal reading series and found that sentences fell into four contextual informativeness categories: misdirective (actually leading to erroneous learning: about 3% of observations); non-directive (ambiguous sentences with little information value about the target word: 27%); general (sentences that help place the target word in a general category: 49%);

and directive (sentences that happen to point the student to the target word's specific, correct meaning: 13%). [1]

More recent research has shown that both high- *and* low-informative contexts play important roles in optimizing long-term retention of new vocabulary, as they invoke different but complementary learning mechanisms. Low-informative contexts force more retrieval from memory, while high-informative contexts elicit a variety of inference processes that aid deeper word comprehension (van den Broek et al., 2018). Exposing a reader to the right carefully-chosen curriculum of different contexts can lead to significantly better long-term retention of new words. For example, Frishkoff et al. (2016a) showed that using a scaffolded series of informative contexts (initially highly informative, then progressively less informative) resulted in the best long-term retention of new words (+15%), compared to all other curriculum designs.

5.2.2 Models for Contextual Informativeness

The predictability of a word given its surrounding context is often represented as a probability calculated from a large corpus (Jurafsky et al., 2001). Language models, in particular, can provide useful information on which words may come after the given context. However, language modeling alone may not adequately capture semantics for contextual informativeness: additional longer-range dependencies, or more sophisticated semantic relations and world knowledge may be needed (Shaoul et al., 2014). Our work focuses on the instructional aspect, which investigates

[1] In our study, the terms non-directive, general, and directive map to low, medium, and high information respectively. We omit the misdirective case for now given its relative rarity.

whether and how context words facilitate making correct inferences about the meaning of the target word (Beck et al., 1983).

Recent models like context2vec (Melamud et al., 2016) or ELMo (Peters et al., 2018) use LSTM layers (Hochreiter and Schmidhuber, 1997) to capture semantic information from a sequence of words. Transformer-based models like BERT (Devlin et al., 2019) can be also used to represent contexts that consist of word sequences. Unlike LSTM-based models, the latter can be more effective in understanding long-range dependencies or unusual relationships between words. In this paper, we use two contextual pre-trained embeddings, ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019) and compare the performance in our model.

In the most closely related work to ours, a study by Kapelner et al. (2018) addressed the task of predicting contextual informativeness. They explored a heavily feature-engineered approach using predictive models based on random forests that combined over 600 different pre-specified text features. Our approach differs in significant ways. First, our approach learns effective feature representations automatically using attention-based deep learning. We also show that a hybrid model that combines and analyzes the benefits of both types of feature representation attains the best overall performance. Second, we explore and evaluate the *interpretability* of the resulting models, to characterize *how* a particular context gives information about a given target word. Third, their specific goal was to achieve high precision in finding highly informative contexts from authentic online texts. In contrast, we focus on predicting *and* characterizing a range of low- and high-informative contexts.

The REAP project Collins-Thompson and Callan (2004) used NLP methods to identify appropriate contexts for vocabulary learning, but focused on filtering entire web pages by tagging sentences with specific criteria, not individual prediction of informative contexts. Similarly Hassan and Mihalcea (2011) used a feature-engineering approach with supervised learning to develop a classifier to label entire documents as ‘learning objects’ for concepts (e.g., computer science).

Our suggested model was inspired in part by Liu et al. (2018) who used an attention-based model to classify customer sentiment towards particular product aspects, by capturing the relationship between context words and a target word.

5.2.3 Similar Tasks

Various NLP tasks involve examining the meaning of a word in a particular context. First, several lexical tasks focus on predicting acceptable words for a given informative context. Lexical substitution tasks have the model choose the correct word that can replace an existing word in a sentence, which can be shown once per sentence (McCarthy and Navigli, 2007), or multiple times in different locations (Kremer et al., 2014). Lexical completion tasks, like the Microsoft Sentence Completion Challenge (Zweig and Burges, 2011), have the model generate a word that can correctly fill in the blank without providing an example target word. However, our task aims to predict the degree of informativeness of a context, assuming the amount of contextual information can vary depending on the selected target word.

Second, previous studies proposed tasks for predicting the semantic properties of predetermined concepts (Wang et al., 2017) or named entities (Pavlick and Pasca, 2017), including particular semantic senses for a target. However, they do not address the case where the target concepts or entities are not presented in the training set. Our annotated task focuses on having a model predict the degree of semantic constraint in a single- or multi-sentence context without using predefined lists of concepts for evaluation.

Third, nonce word tasks include various learning scenarios for unseen words. Studies like Lazaridou et al. (2017); Herbelot and Baroni (2017) investigated how contextual information can be used to infer the meaning of synthetically generated target words. However, they also relied on the assumption that the provided context contains enough information to make an inference, by manually selecting the training sentences for synthetic words. In contrast, our contextual informative task involves diverse examples where some contexts can be less or more helpful. Our model also attempts to characterize the nature of the explicit cues that exist for learning the target word.

5.3 Contextual Informativeness Datasets

We used two different datasets to train and evaluate our model's performance on predicting contextual informativeness. These datasets have different context lengths (single- vs. multi-sentence contexts), labeling methods (relative vs. absolute assessment scales), and number of included contexts per target word. Using datasets with varied characteristics helps test the generality of our suggested model structure and whether it can effectively predict contextual informativeness in different situations. Both datasets use as gold-standard labels for contextual informativeness, a value on a numerical scale that is based on the perceived learning effectiveness of the context for the given target word. This effectiveness summarizes, for example, the precision and variety of any cues that are present in the context that help a reader infer the precise, correct meaning of the target word. Specific examples of cues might include synonymy, antonymy, cause-effect, whole-part, frequent co-occurrence, or other relationships that help comprehend the meaning of a new word. However, because of the virtually unlimited nature of these cues, for both datasets, annotators were not given explicit relation types as a basis for judgment.

5.3.1 Dataset 1: Single-sentence Contexts

Our first dataset is a new collection of pedagogical single-sentence contexts for contextual vocabulary learning. This annotated data consists of 1783 sentences. Each sentence contains exactly one target word drawn from a set of 60 words (20 nouns, 20 verbs, and 20 adjectives). These target words were ‘Tier 2’ words (critical for literacy but rarely encountered in speech), carefully normed to achieve a balanced set of psychometric properties (abstract/concrete, age of acquisition, etc.).

With these target words, researchers (not the authors) with literacy research background generated sentences with high, medium, or low informativeness[2]. The sentences were normed to control variability in semantic and syntactic properties, such as length and difficulty. The average length of these sentences was 12.49 words

[2] The phrase *level of semantic constraint* is sometimes used to describe the level of contextual information.

($\sigma^2 = 2.75$ words), and the average relative location of the target word was 64.37% from the beginning ($\sigma^2 = 2.83\%$). Further details are in Appendix A.1.

Annotating Perceived Informativeness. For these generated sentences, we cross-checked the original researcher-provided labels with additional crowdsourced annotations using the best-worst scaling (BWS) method. BWS is often preferable to other strategies like ranking with a Likert scale in cases where annotators can reliably distinguish between items (e.g., which sentence is more informative), while keeping the size of annotations manageable. Previous studies like Kiritchenko and Mohammad (2017) and Rouces et al. (2018) have used BWS annotation to create semantic lexicons. For our task, we asked non-expert crowdworkers to "find the most- and least-informative sentences" with respect to the word's meaning. For each question item, workers selected the best or worst informative sentence from a set of four sentences.

Annotation results for BWS scores were highly reliable. Following best practices for measuring annotation replicability as outlined in Kiritchenko and Mohammad (2016), we simulated whether similar results would be obtained over repeated trials. Annotations were randomly partitioned into two sets and then each used to compute the informativeness scores, comparing the rankings of two groups. We repeated this process 10 times, and found the average of Spearman's rank correlation coefficients was 0.843 (σ^2=0.018, all coefficients were statistically significant ($p < 0.001$)), indicating high replicability in the scores. Inter-rater agreement rates for the best and worst sentence picks for each tuple were 0.376 and 0.424 with Krippendorff's α. More details are in Appendix A.2.

5.3.2 Dataset 2: Multi-sentence Contexts

The meaning of a target word can be also determined from information in multiple surrounding sentences. To test the generalizability of our models to the multi-sentence scenario, we used an existing dataset from the only previous study, to our knowledge, on contextual informativeness Kapelner et al. (2018). This dataset contains 67,833 multi-sentence contexts selected from the DictionarySquared database ($\mu = 81$ words, $\sigma^2 = 42$). Each context contains exactly one of 933 unique target words, which were selected to range across difficulty levels. Like the single-sentence dataset, this dataset was also designed for contextual vocabulary learning, but in contrast to our BWS procedure, crowdworkers annotated informativeness of context passages for a target word with an ordinal four-point Likert scale roughly corresponding to the four categories in Beck et al. (1983).

5.4 Model Structure

Our approach is based on a deep learning model with some pre-trained components (orange blocks in Figure 5.2) that can easily fetch vector representations of contexts and the target word. Specifically, we use pre-trained versions of ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019). During training, we updated the pre-trained models' parameters; however, to avoid overfitting, we only fine-tuned selected parameters: for ELMo-based models, we updated parameters that determine the aggregating weights of LSTM and word embedding layers; for BERT-based models, we updated the parameters for the last encoding layer. For each model,

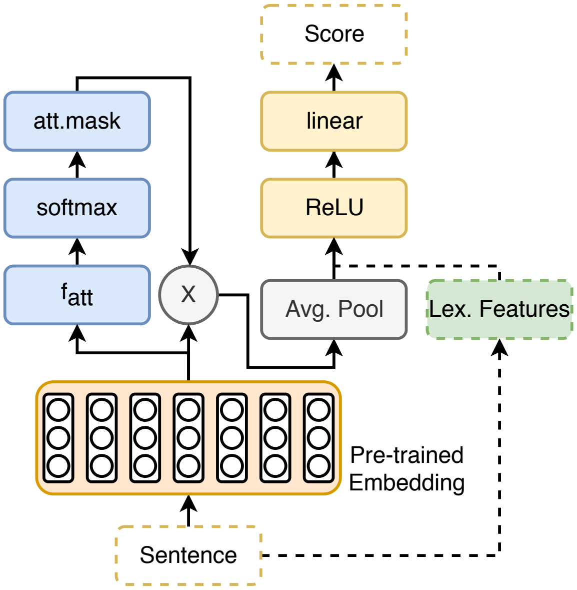


Figure 5.2: The structure of our proposed model. The model consists of a pre-trained embedding (orange) with masked attention block (blue) to create attention weighted context vectors, and regression block (yellow) to predict the numeric contextual informativeness score. For the multi-sentence context dataset, we also tested lexical features from Kapelner et al. (2018) (green) as a complementary model's additional input.

we treated the target word as unknown (`<UNK>` for ELMo) or masked (`[MASK]` for BERT) token so that the model must use contextual information to infer the meaning of the 'unknown' target word.

The input for the attention layers (blue blocks) is the vector for the target word and context tokens. Using a multiplicative attention mechanism (Luong et al., 2015), we calculated the relationship between the token that replaced the target word and context words (f_{att}). We used $softmax$ to normalize the output of the attention layer. The output of the softmax layer masked non-context tokens as zero, to eliminate the weights for padding and the target word ($att.mask$). The masked attention output was then multiplied with the contextual vectors from the pre-trained model to generate attention-weighted context vectors. For the multi-sentence context dataset, we also tested lexical features from Kapelner et al. (2018) as input features (green block), by concatenating with attention-weighted context vectors (e.g., `+Att+Lex` in Figure 5.3).

The regression layers (yellow blocks) used an average pooling result of attention-weighted context vectors. First, the ReLU layer was applied, followed by a fully-connected linear layer that estimated the score of contextual informativeness on a continuous scale. We used root mean square error (RMSE) as a loss function. Full details for the model and hyperparameters are in Appendix B.1.[3]

5.5 Experiment 1: Predicting Contextual Informativeness

In the first experiment, we examined the prediction performance of our attention-based model by (a) comparing its effectiveness to both simple and more complex baseline classifiers; (b) comparing the effectiveness of deep vs lexical-based representations and their combination; (c) cross-prediction, training on one dataset and testing on the other; and (d) examining variations in the deep learning design, using ELMo or BERT, and the role of the attention block in prediction performance.

[3] Data and code will be open sourced upon acceptance.

5.5.1 Experimental Setup

For baselines, we used (1) a dummy model that always predicts the average informativeness score from a fold (`Base:Avg`); (2) a ridge regression model using the co-occurrence information of context words (`Base:BoW`), which represents a prediction *independent* of the target word; and (3) a linear regression model based on sentence length (`Base:Length`). Such baselines represent simple estimations of contextual informativeness without using external resources like ELMo or BERT.

Prediction performance is measured using two scores: RMSE and ROCAUC. RMSE shows how the model's prediction scores diverge from the true scores. For ROCAUC, we set specific thresholds to investigate how the model performed in predicting binary labels for high (e.g., top 20% or 50%) or low (e.g., bottom 20%) informative sentences. Per Section 5.2.1, selecting a range of context informativeness levels can be important in learning applications, while the high-precision setting resembles the goal of the previous study by (Kapelner et al., 2018). All reported results are based on 10-fold cross validation. Each fold was randomly selected based on the target word, to ensure the model did not see the sentence with the same target word during the training process.

5.5.2 Results: Single-sentence Contexts

In Figure 5.3, with single-sentence contexts, the sentence-length baseline (`Base:Length`) shows that, as expected, raw word count provides some information about contextual informativeness. However, in most cases, ELMo- and BERT-based models with attention block (`+Att`) performed significantly better than the

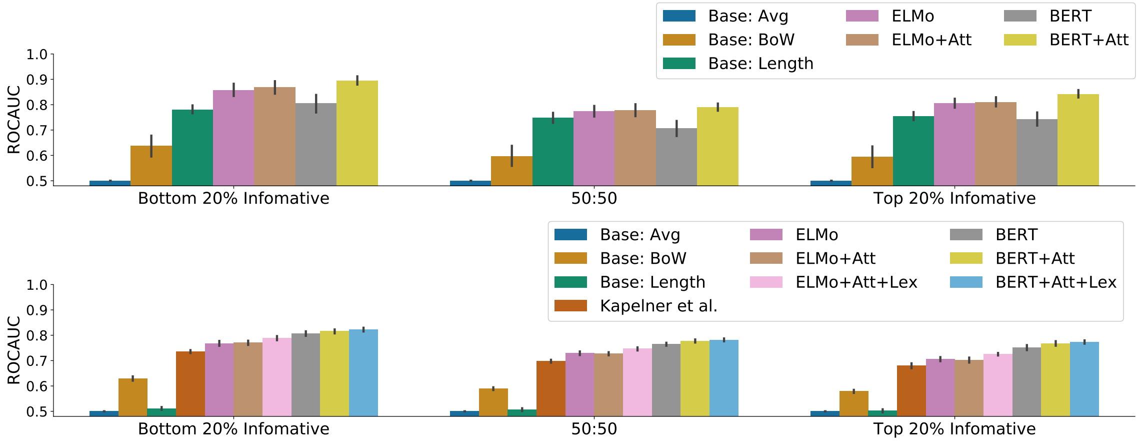


Figure 5.3: Binary classification results with the single-sentence context dataset (top) and multi-sentence context dataset (bottom; higher means better). ELMo and BERT-based models performed significantly better than the baseline models in most cases. Adding the attention block (`+Att`) also improved the prediction performance, especially for BERT-based models. For the multi-sentence contexts, the complementary model using lexical features from Kapelner et al. (2018) (`+Att+Lex`) showed the best prediction performance.

baselines (e.g., 95% CI of ROCAUC scores in 50:50 classification were 0.736-0.775 (`Base:Length`), 0.787-0.834 (`ELMo+Att`), and 0.824-0.860 (`BERT+Att`)). Adding the attention block to the model introduced either slight (ELMo-based) or significant (BERT-based) improvement (Appendix B.3.1).

5.5.3 Results: Multi-sentence Contexts

For training and evaluation based on the multi-sentence context dataset from Kapelner et al. (2018), we added our replication of the random forest model from that paper as an additional baseline model, using lexical features provided by the authors. Our baseline replicated their results, with very similar R^2 scores (e.g., 0.179 vs. 0.177).

In Figure 5.3, with multi-sentence contexts, all ELMo- and BERT-based models performed significantly better than the baseline models. Moreover, all BERT-based models outperformed the ELMo-based models significantly. (e.g., 95% CIs in 50:50 classification were 0.691-0.705 (`Kapelner et al.`), 0.720-0.734 (`ELMo+Att`), and 0.770-0.785 (`BERT+Att`)) The attention block (`+Att`) provided marginal gain for BERT-based models in prediction performance. The complementary model, which concatenated attention-weighted context vectors with lexical features (`+Att+Lex`), provided the best overall prediction performance for both ELMo- and BERT-based models (Appendix B.3.2).

5.5.4 Cross-Prediction

We also tested the generalization ability of our model by cross-training. Table 5.1 demonstrates that the model trained with the multi-sentence context dataset was effective at predicting the contextual informativeness scores for the single-sentence context dataset, showing that our model captures some essential aspects of contextual informativeness. In this scenario, both models performed better than random chance (0.50), while the BERT-based model performed better than the ELMo-based model. The weaker performance in this transfer learning setting suggests that models are learning context-type-specific cues of contextual informativeness, rather than general strategies, implying that general contextual informativeness models should see a variety of contexts to perform well.

ROCAUC	↓ 20% Info	50:50	↑ 20% Info
ELMo+Att	0.711	0.678	0.651
BERT+Att	0.784	0.752	0.715

Table 5.1: Models trained on the multi-sentence dataset were effective at predicting contextual informativeness scores for the single-sentence context dataset. However, less effective prediction results were observed in the other direction (Appendix B.3.3).

5.6 Experiment 2: Evaluating Attention Weight Quality

In the second experiment, we examined how the model's attention mechanism can provide interpretable details on informative contexts, by identifying contextual cues that facilitate more precise inference of the meaning of the target word.

5.6.1 Quantifying the Quality of Attention

To evaluate our model's attention output across different types of contextual relationships, we used the EVALution dataset (Santus et al., 2015), originally designed for evaluating whether word embedding models capture nine different semantic relations between word pairs. It includes over 1800 unique words and seven thousand word-pairs, used in automatically generated example sentences. Each sentence mirrors lexico-syntactic patterns expressing a particular semantic relationship (e.g., Hearst, 1992; Pantel and Pennacchiotti, 2006; Snow et al., 2005), so a contextual cue to the meaning of a target word is easily identifiable. Although formulaic, this regular structure allows us to control for many contextual confounds and to test which types of relational information are identified by the model.

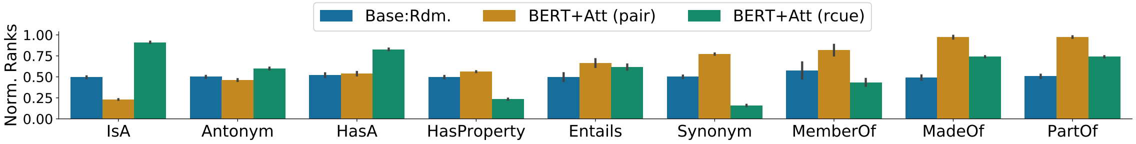


Figure 5.4: Comparing the normalized rank scores from the BERT-based model on the EVALution dataset (Santus et al., 2015). Higher means better. The BERT-based model with the attention block was able to capture important contexts, such as the pair word (*pair*) and relational cues (*rcue*) better than randomly sampled rank scores. For 4 of 9 relations, both *pair* and *rcue* contexts were captured better than the random baseline.

For example, in the sentence "An account is a type of record," the meaning of the target word "account" is informed by a single contextual *pair word* "record", and a *relational cue* word "type" that says how the pair word relates to the target semantically. If the attention mechanism reflects an interpretable explanation of informativeness, it should put more weights on the pair word and relational cue word in the context. Our analysis examines whether our model attention weights rank these salient contextual cues highly, by measuring the normalized rankings $1-\frac{rank(x)-1}{N-1}$ of the pair or relational cue words' attention weights from each sentence. For comparison, we used a baseline where the pair or relational cue word is assigned a random rank.

5.6.2 Results

Figure 5.4 compares the performance of the `BERT+Att` model (trained with the multi-sentence context dataset) and the randomized rank baseline on the EVALution dataset (Santus et al., 2015). The `BERT+Att` model captures the pair word better than the baseline in 7 of 9 relations, and the relational cue words in 6 of 9 relations.

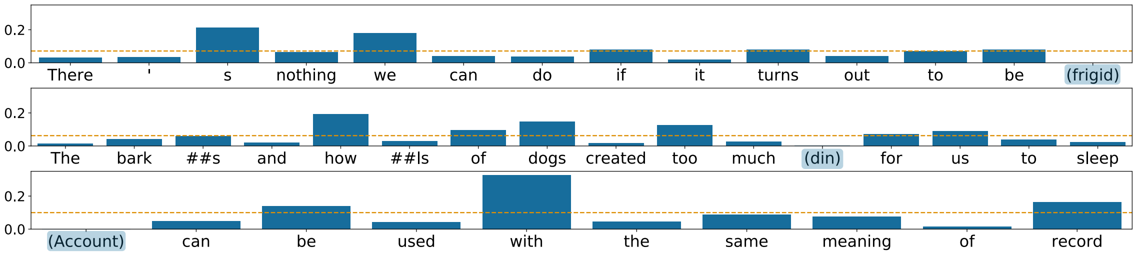


Figure 5.5: Example attention weight distributions from the BERT-based model trained with multi-sentence contexts. Our model successfully captures meaningful attention weights in the high-informative sentences. The dotted line marks the average weight value for each sentence. Each target word (highlighted) was masked for the model.

Surprisingly, the model did not perform well at ranking the pair word highly in the IsA and Antonym relationships, despite these pair words providing the most evidence of meaning. In contrast, the model was able to correctly rank meronyms (e.g., MadeOf, PartOf , and MemberOf) as having high semantic information. The `ELMo+Att` model captured the pair words better, but did worse with the relational cue words (Appendix B.3.4).

Qualitative Evaluation Figure 5.5 shows how the model learned a relationship between the target word and context words with example sentences[4]. For the (first) low-informative sentence, the model put more weight on function tokens (e.g., *'s, we*), suggesting that the sentence lacked sufficient content words to constrain the meaning. However, for the (second) high informative sentence, the model tended to put more weight on individual content words (e.g., *howls, dogs, too*) that help infer the meaning of the target word. The last sentence shows the output for a synonym-

[4] The first two sentences are drawn from our single-sentence context dataset.

relation sentence (Santus et al., 2015). The model's attention activation successfully highlighted the pair word (*record*) but not the relational cue word (*same*). Further analysis is needed to examine why some context cues were not highlighted, and how methods for fine-grained interpretation of contextual informativeness could help curriculum instruction or improve prediction performance.

5.7 Discussion

We view our study as bridging recent deep learning advances in semantic representation with new educational applications. Predicting, characterizing, and ultimately, optimizing the quality of a learner's initial encounters with content has many potential applications for both human and machine learners. Our results showed that attention-based deep learning models can provide both effective and interpretable results for capturing the instructional aspect of contextual informatinveness.

For human learners, contextual informativeness models could be applied to diverse sources of classroom-generated text, such as video transcripts or class notes, to find the most supportive lecture contexts that help learn or review specific terminology. Search engines could emphasize the most contextually informative educational Web content examples for a given term or concept. Custom pre-trained models (e.g., BioBERT (Lee et al., 2019) for biomedical terms and content) would enable more domain-specific applications.

On the algorithmic side, accurate prediction and characterization of contextual informativeness would be highly valuable in NLP applications, including finding educationally supportive sentences for automatic summarization, or automatic curriculum design for few-shot learning NLP tasks (Herbelot and Baroni, 2017), where the quality of the training examples is critical. Conversely, our new dataset could be a valuable resource for systematically evaluating the few-shot learning capabilities of sophisticated language modeling systems (e.g., Brown et al., 2020).

We also discuss here a few limitations of this study that may inspire further research. First, a more complete model of contextual informativeness would include an individual component capturing a specific user's background knowledge of the target concept, although such models can be challenging to learn and evaluate. For example, individual differences in word knowledge (Borovsky et al., 2010) or language proficiency (Elgort et al., 2018) may result in different levels of comprehension or faster processing of orthographic information. In the annotation phase, we tried to minimize these confounds by collecting multiple responses per sentence and limiting the geographic location of annotators to native English speaking countries. However, further comparison between different learner profiles, such as L1 vs. L2 learners, could benefit developing more personalized and/or group-oriented prediction.

Second, because our initial students were English-language learners, our efforts focused on developing English sentences. Further studies based on non-English datasets, accounting for more richly inflected languages, more complex grammatical rules, or semantic biases associated with different cultures (Osgood et al., 1957; Bolukbasi et al., 2016) would be a valuable complement to the science of reading literature.

5.8 Conclusion

Both humans and machines rely on context for acquiring a word's meaning—yet not all contexts are equally informative for learning. We introduced a new high-quality dataset labelled for contextual informativeness, and showed that attention-based deep learning models can effectively capture and predict a general conception of contextual informativeness, in a way that generalizes across significantly different datasets. Moreover, we showed that deep neural representations learned automatically can replace or augment a complex, feature-engineered model for this contextual informativeness task. Further, we demonstrated that learned attention mechanisms can provide interpretable explanations that match human intuition as to which specific context words help human readers infer the meanings of new words.

5.9 Author Contributions

Sungjin Nam was the main contributor to the study, including designing the study, organizing the crowdsourcing task for sentence annotations, developing the prediction model, and writing the manuscript. Dr. Kevyn Collins-Thompson and Dr. David Jurgens contributed to conducting the experiment and revising the manuscript.

Chapter 6

Smaller and Stronger: Developing Curricula for Word Embedding Models Based on Contextual Informativeness Scores

6.1 Introduction

We learn a significant number of new vocabulary terms through understanding contextual information (Landauer and Dumais, 1997). Through iterative practice, students learn the meanings of words from various contexts and develop deeper and more sophisticated knowledge of advanced vocabularies (Huang and Eslami, 2013). Various machine learning algorithms have also suggested different methods

Low Informative Sentence: We couldn't agree on whether the *din* was acceptable.
High Informative Sentence: The barks and howls of dogs created too much *din* for us to sleep.

Figure 6.1: Each sentence differs in the amount of contextual information for the same target word *din*.

to capture the semantic information of words based on contextual information (Blei et al., 2003; Mikolov et al., 2013; Devlin et al., 2019). Without explicit definitions or associated words presented for the target (unknown) word, contextual word learning utilizes linguistic context, such as nearby semantic and syntactic cues, for learners to infer the meaning of the target word.

Studies have shown that identifying the level of contextual informativeness is crucial for designing more effective curricula for contextual word learning for human learners (Webb, 2008; Frishkoff et al., 2015). However, not all contexts are equally informative for instructional purposes. As Figure 6.1 shows, the amount of contextual information with respect to a target word may vary greatly. It is a non-trivial task to automatically quantify the amount of information included in a target word's context and use it to develop better curricula for vocabulary learning (Kapelner et al., 2018).

Compared to many machine learning models, humans can learn from a relatively small number of examples (Fei-Fei et al., 2006; Lake et al., 2015). When availability of training data is limited, such that learning takes place over only a small number of examples for human language learners or automated few-shot learning algorithms, the quality of the instructional materials in the training set becomes critical to achieve better learning outcomes. Thus here we investigated whether contextual informativeness scores used for designing a vocabulary learning curriculum for human learners can be also useful for word representation learning algorithms in multiple learning scenarios.

We examined how a deep learning model developed for predicting contextual informativeness in human language acquisition can also benefit machine learning algorithms. We compared the performance of word embedding models in two scenarios, batch learning and few-shot learning, according to different training curricula that were based on predicted contextual informativeness scores. The results show that training only on low-informative sentences significantly lowered performance on downstream tasks. Further, simply detecting and removing the least-informative 50% of examples in the training corpus provided significantly better performance for word embedding models.

6.2 Related Work

We first review work on the prediction task itself, contextual informativeness in the context of word learning, and then discuss connections with curriculum learning for NLP.

Contextual Informativeness & Word Learning. Repetitive exposure to contextual cues from text or conversation can provide much information about the meaning of unknown words (Frishkoff et al., 2008). Contextual word learning is an instructional method that teaches students how to infer the meaning of unknown words by recognizing and utilizing semantic cues, such as synonyms and cause-effect relationships (Heilman et al., 2010).

Previous studies such as Kapelner et al. (2018) have shown that machine learning models can be useful for determining high-quality instructional material for vocabulary learning. However, their model was less generalizable since it relied on a large number of pre-defined lexical features, made it hard to interpret which context words were more important, and focused on precisely identifying the high-informative contexts only. In Section 6.3.1, we describe the deep learning-based model used for our curriculum construction that addresses these issues and attains better performance.

Recent studies have also investigated how to develop meaningful curricula for vocabulary learning by humans. For example, Frishkoff et al. (2016b) showed that using an ordered series of informative contexts (e.g., starting from highly informative, followed by less informative examples) provided better long-term retention of new words than several other curriculum designs. Since different levels of informativeness can elicit different learning behaviors (e.g. low-informative contexts facilitate retrieval from memory, while high-informative contexts can promote various inference processes and comprehension (van den Broek et al., 2018)) it is important to distinguish different levels of informative learning materials in contextual word learning.

Curriculum Learning in NLP. Previous machine learning studies have focused on achieving more efficient training (Bengio et al., 2009). For example, multiple NLP studies investigated the role of training curriculum on performance. Among others, Sachan and Xing (2016) showed that easy (e.g., smaller losses), but diverse training examples can provide better performance for NLP models to solve QA tasks. The properties of the learning target, such as topical domain or part-of-speech, may also affect the stability of word embedding models (Wendlandt et al., 2018; Antoniak and Mimno, 2018). Particular context words can be more important than others for predicting the target word, and weighting each context word differently can improve training efficiency (Kabbach et al., 2019).

However, unlike these studies, our study focuses on the idea that applying vocabulary learning strategies that help human students can also benefit machine learning models. By using a model that predicts contextual informativeness, we score sentences from the corpus and developed simple filtering heuristics for producing higher-quality training curricula. We also examine how contextual informativeness scores can be used for more efficient training of word embedding models.

6.3 Models

We now describe the two types of models used in this study, for (a) contextual informativeness prediction and (b) word embedding. We used the contextual informativeness prediction model described in Chapter 5 (Section 5.4) to score sentences from a corpus. Based on the predicted score, we applied a variety of filtering heuristics to the training corpus and used these to train different word embeddings. We did a task-based evaluation of curriculum quality according to its effect on several important downstream similarity tasks that used the word embeddings.

6.3.1 Contextual Informativeness Model

We used the deep learning model from Chapter 5, which originally developed for predicting the contextual informativeness of sentences in an instructional context to score sentences from the corpus.

The model used a pre-trained BERT (Devlin et al., 2019) with an additional masked attention mechanism. The model was trained with the existing dataset from Kapelner et al. (2018), but did not used lexical features for prediction. We chose this variant of the contextual informativeness model since it provided the best performance in a generalized task (e.g., Table 5.1 or Figure 5.4). A diagram for the model we used for this study can be found in Appendix C.1.

6.3.2 Word Embedding Models

To examine the effect of different curricula based on contextual informativeness scores, we tested three word embedding models. As further described in Sections 6.5 and 6.6, we used FastText and Word2Vec models for a batch learning scenario analysis, and Nonce2Vec model for a few-shot learning scenario.

First, *Word2Vec* is a word embedding model that captures the meaning of the word based on the surrounding context words using an algorithm like skip-gram (Mikolov et al., 2013). We chose Word2Vec since it is widely used and has been a strong traditional baseline in many NLP tasks.

Second, *FastText* is an extension of Word2Vec that incorporates a character-level encoding (Bojanowski et al., 2017), intended to handle rare words better, or even represent out-of-vocabulary words through n-gram information.

Third, *Nonce2Vec* is another variant of Word2Vec that specializes in learning with a smaller training corpus for nonce words (Herbelot and Baroni, 2017). This model employs a higher initial learning rate and customized decay rate, to provide provide a riskier but more effective learning strategy for unseen words with a small number of training sentences.

6.4 Curriculum Building

Our curriculum used sentences from the *ukWaC* corpus (Ferraresi et al., 2008) to train the word embedding models. ukWaC is a corpus of British English collected from web pages with the *.uk* domain, using medium-frequency words from the British National Corpus as seed words. Since the corpus is collected from a broad set of web pages, its sentences contain different types of writings with various contextual informativeness levels.

Semantic Similarity Tasks. To test the performance of word embedding models, we employed three semantic similarity tasks. First, *SimLex-999* includes 999 pairs of nouns, verbs, or adjectives. It also distinguishes more associated word pairs. Second, as the name suggests, *Simverb-3500* includes verb pairs. It uses the same guidelines as SimLex-999 for collecting human annotations (Gerz et al., 2016). Third, *WordSim-353* includes scores for noun pairs (Finkelstein et al., 2002).

These tasks use human annotations on semantic similarity between word pairs as their gold standard. We trained word embedding models for each task, calculated the cosine similarity between words, and analyzed the correlation with the human-annotated scores (Spearman's r).

Training Sentences for Word Embeddings. To carefully control the learning materials and compare outcomes between curricula, we chose sentences from the corpus as follows.[1]

First, we divided sentences into *non-target sentences* and *target sentences. Non-target sentences* are sentences that do not contain any target word from the three tasks (about 8.1M sentences, 2.4M unique tokens, 133.7M tokens in total). Non-target sentences were used to train the background model, which represents the model's prior knowledge about other words before they learn about target words. On the other hand, each *target sentence* contains the target word only once. Later, target sentences are scored with the contextual informativeness model and used for developing the training curricula. All sentences containing multiple target words were removed from the analysis.

Second, we selected sentences that are 10–30 words long, which is similar to the length of average English sentences (15-20 words; (Cutts, 2013)). This criterion filtered out sentences that are too short or too long. It also controlled potential correlation between the sentence lengths and contextual informativeness scores, and kept the number of words between curricula relatively similar.

Third, we only analyzed target words that had more than a certain number of sentences in the corpus. We sampled 512 sentences per target word. Each sentence

[1] Details on the distribution of informativeness scores and relationship with sentence length are in Appendix C.2.

contained the target word only once. If a target word had less than 512 training sentences meeting all criteria above, we excluded the target word from the analysis. As a result, we analyzed 94.16% (SimLex-999; 968 of 1028 unique target words), 80.89% (SimVerb-3500; 669 of 827), and 95.65% (WordSim-355; 418 of 437) of target words of each task.

6.5 Experiment 1: Batch Learning

For this study, we conducted two experiments. Each experiment tested different learning scenarios, *batch learning* scenario and *few-shot learning* scenario, for word embedding models. We evaluated how each curriculum based on contextual informativeness scores can affect word embedding models' performance across different numbers of target sentences used per target word. The first experiment tested *Word2Vec* and *FastText* models.

6.5.1 Experimental Setup

First, we trained the background model with *non-target sentences*. This simulates a model with existing knowledge of English words excluding the target words. Second, we developed the training curriculum by selecting k ($k = 2^i, i = 1, ..., 9$) *target sentences* per each target word. For the experiment, we tested five simple heuristics to build curricula (Table 6.1). The results compare how curricula based on contextual informativeness scores can bring different results for updating the same background word embedding model. Figure 6.2 visualizes the curriculum development process.

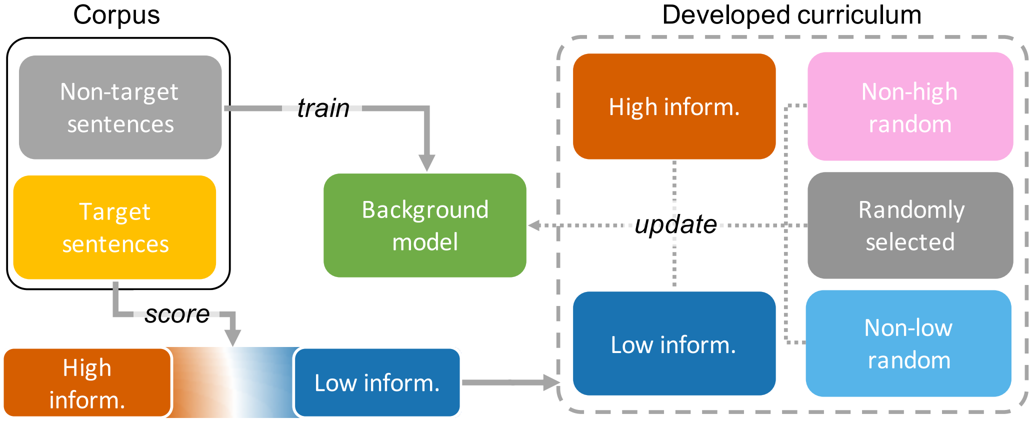


Figure 6.2: An illustration of curriculum developing process. First, we separated sentences into *non-target* and *target sentences*, based on if the sentence contains any target word from the semantic similarity tasks. Second, *target sentences* were scored by the contextual informativeness model. *Non-target* sentences were used to train the background model. Third, different curriculum heuristics were developed to update the background model with target sentences.

6.5.2 Results: Similarity Tasks

To evaluate the learning performance, we calculated Spearman's rank coefficient scores for each semantic similarity task between cosine similarity scores of word pair vectors and human-annotated scores. Word2Vec is known to perform at various levels for the similarity tasks.[2] Although we used a different corpus and parameters, we were able to achieve similar performance with our Word2Vec models with a larger number of target sentences.

Figure 6.3 shows that the contextual informativeness model can successfully distinguish less helpful training sentences from the corpus. In most cases, the low informative sentence curriculum provided significantly worse performance.

[2] Reported Spearman's r scores for Word2Vec models were 0.414 (Hill et al., 2015, SimLex-999;), 0.274 (SimVerb-3500; Gerz et al., 2016), and 0.655 (WordSim-353; Hill et al., 2015)

Curriculum	**Description**
Low Informative	Selecting k low informative sentences per target word
High Informative	Selecting k high informative sentences per target word
Rand. Select	Random k sentences per target word
Rand. Non-Low	Random k sentences per target word from the top half informative sentences (256)
Rand. Non-High	Random k sentences per target word from the bottom half informative sentences (256)

Table 6.1: Five curriculum building heuristics based on contextual informative scores.

Figure 6.4 shows more clearly that for SimLex-999 and SimVerb-3500 tasks, the non-low informative sentence models tend to perform better, especially with more sentences per target word (e.g., $> 2^4$). However, the non-high informative sentence models tended to perform worse. These results indicate that contextual informativeness scores may improve the word embedding models' performance by removing less informative contexts from the training set. We also noted that FastText models were significantly better than the Word2Vec models when trained with the lower number of training sentences. However, using all training sentences, the performance of these models eventually converged within a similar score range. The results confirm that FastText models were better for representing rare words (Bojanowski et al., 2017) and as such, the differences between curricula are smaller for FastText models than for Word2Vec models.

6.5.3 Results: By Part of Speech

We further analyzed the results from the SimLex-999 task with different target word attributes, such as part of speech (POS) and associated pairs.

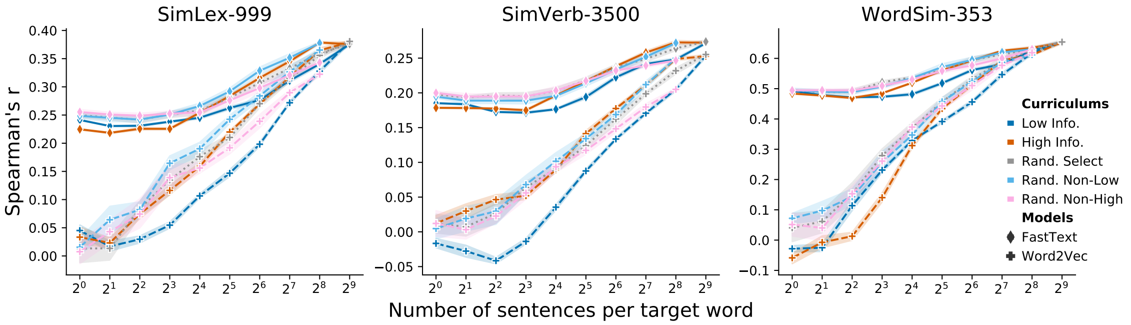


Figure 6.3: Batch learning results on FastText♦ and Word2Vec✚ models with SimLex-999, SimVerb-3500, and WordSim-353 tasks. Shades represent the 95% CI. The results show that the contextual informativeness score can successfully distinguish less helpful sentences for the batch learning models. The FastText models are significantly better than Word2Vec models, with fewer training sentences. Notice that scales are different between the tasks.

The results by different target word POS were similar to the previous results from Simverb-3500 and WordSim-353 tasks (Figure 6.5). For example, with the verb pairs (222) the high informative sentence models performed significantly better than the low informative sentence models. The distinction between the high informative models and the randomly selected sentence models were also significant. For the noun pairs (666), the non-low informative random sentence models performed consistently better than other models. For the adjective pairs (111), we observed unusual examples of early learning (e.g., 2^0–2^3) from the low informative sentence Word2Vec models. However, it did not link to as much continuous improvement as the model trained with a larger number of sentences.

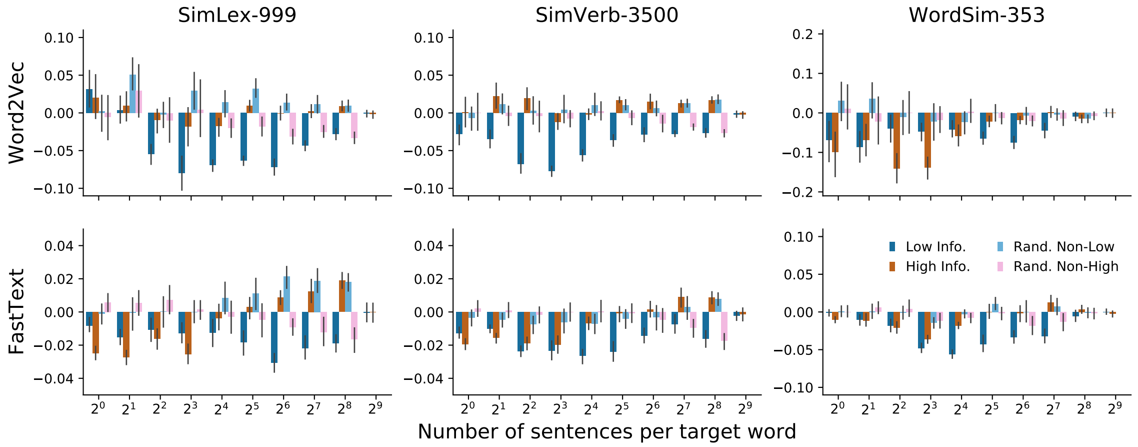


Figure 6.4: Differences between curricula relative to random curriculum performance. Black lines represent 95% CI. Filtering out low informative sentences increased embedding models' performance in many cases. For SimLex-999 and SimVerb-3500, as the number of sentences per target word increased, the non-low informative sentence models performed significantly better than others, while the non-high informative sentence models performed worse. Note that scales are smaller for FastText models (which does better for rare words) but still shows a consistent trend.

6.5.4 Results: By Associated Pairs

For the associated and non-associated pairs, we observed that contextual informativeness scores worked consistently well in distinguishing less vs more helpful training sentences for both models.

For example, with the non-associated target word pairs (666), the non-low informative random sentence models performed at least as well as the random sentence models. The high informative models showed consistently better performance than the low informative models. For the associated pairs (333), unlike other analysis results, FastText models did not show early learning advantages, showing that the associated pair task is a hard task for both Word2Vec and FastText

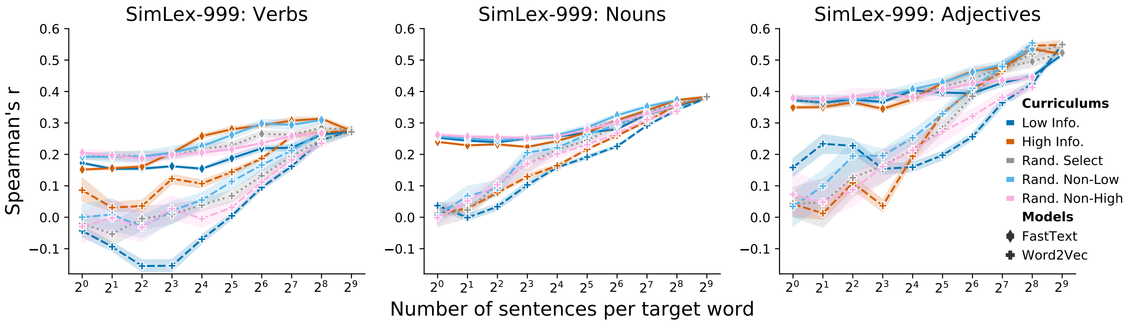


Figure 6.5: Batch learning results on FastText♦ and Word2Vec✚ with different POS of SimLex-999 target words. In most sentences per target word conditions, filtering out low informative sentences provided significantly better performance than the model trained only with low informative sentences. For verbs, the high informative sentence models showed significantly better performance than the randomly selected sentence models. For nouns, the non-low informative sentence models performed significantly better than models trained on random sentences.

word embedding models. For this analysis, Word2Vec models trained with the non-low informative random sentences performed best.

With the batch learning setting using Word2Vec and FastText models, we found that our contextual informativeness model effectively distinguished less useful sentences from the training set. In most cases, the non-low informative random sentence models performed best. The performance of the all-high informative sentence models and the all-low informative sentence models were also significantly different in many cases. Based on these results, we suggest that filtering out low-informative sentences from the training set can significantly improve model performance. In the next section, we looked into few-shot learning using Nonce2Vec.

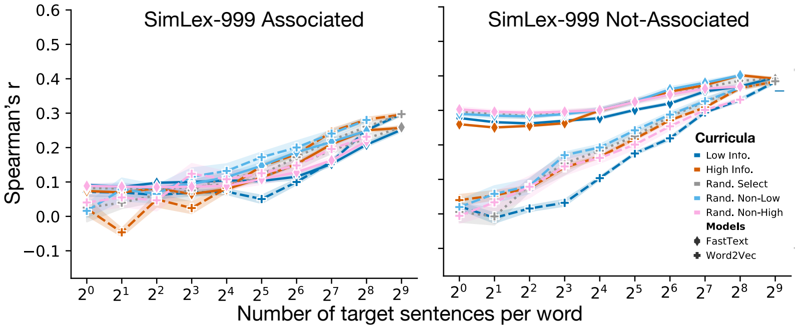


Figure 6.6: Batch learning results on FastText♦ and Word2Vec✚ with associated and non-associated target word pairs (SimLex-999). The non-low informative random sentence models performed best in both tests. FastText models did not show an advantage with smaller sentence sizes for the associated word pairs.

6.6 Experiment 2: Few-Shot Learning

The second experiment tested the effect of curriculum design with the Nonce2Vec model, which is explicitly designed for few-shot learning.

6.6.1 Experimental Setup

We follow the same experimental setup for training Nonce2Vec as in Herbelot and Baroni (2017), which first trains a background model and then tests he effect of adding new sentences. The background corpus contained non-target sentences and ∼60% of target sentences per target word; training curricula were derived from other 40% of target sentence and used to update the nonce word vectors. More details on Nonce2Vec model are in Appendix C.5.

We used 2, 4, or 6 target sentences per target word to train the model. For a more robust comparison, we selected each curriculum stochastically. For example, we first created a sampling pool of 50 sentences (about 10% of sentences per target word) for each curriculum type (with overlaps)[3]. From each pool, we then randomly sampled 2, 4, or 6 target sentences per target word to develop the curriculum for each iteration.

For evaluation, we compared the median ranks of newly-learned nonce word vectors and the gold-standard word vectors included in the background model. Ideally, if the nonce word learning process of Nonce2Vec was perfect, the embedding vector for the nonce word and the gold-standard word from the background model (e.g., *insulin* vs. *insulin_gold*) should be identical ($rank = 1$). Similarly, lower median rank scores for the target words (e.g., SimLex-999) would indicate better embedding quality derived from a curriculum.

We also compared Spearman's rank correlation scores of each semantic similarity task. For Nonce2Vec models, we took the average of two cosine similarity scores for a word pair, as we conducted the nonce learning for each target word separately. For example, for the word pair *old–new*, we first conducted nonce word learning for the word *old* and calculated the cosine similarity score with the word *new* from the background model. Then we conducted the same process vice versa. The average of these two cosine similarity scores was calculated for the word pair.

6.6.2 Results: Nonce2Vec on SimLex-999

We used SimLex-999 for the few-shot learning analysis, as the task includes word pairs with various lexical attributes. Figure 6.7 shows how Nonce2Vec models ($epoch = 5$, $learningrate = 0.5$) performed with different curricula.

First, we compared the quality of updated nonce word vectors by comparing

[3] For non-low and non-high informative curricula, we firstly filtered out 256 low or high informative sentences and then sampled 50 sentences from the rest.

the similarity ranking with the gold-standard word vector. We observed similar patterns from the batch learning results. The median rank scores indicate that the low-informative sentences performed significantly worse than other curricula. Also, the non-low informative random sentences tended to perform better than others. The high-informative sentence models showed significantly better results than the low informative sentence models, but not much different from the randomly selected sentence models. These results show that our contextual informativeness model can effectively distinguish less-helpful sentences for the few-shot learning tasks. Moreover, based on the random non-low results, we see that excluding the least informative ones, together with using diverse levels of contextual informativeness stimuli (medium and high), can improve word embedding model performance.

Second, we analyzed the rank correlation between the Nonce2Vec word vectors and human-annotated scores from SimLex-999. The low-informative sentence models performed marginally worse than other curricula. The non-low informative models and randomly selected sentence models performed similarly. However, we did not observe significant differences between curricula.

6.7 Discussion

Our results show that filtering out low informative sentences from the training set provides better learning outcomes across various sentence sizes and similarity tasks, which suggests three potential modeling improvements.

First, more detailed analysis on where the BERT-based contextual

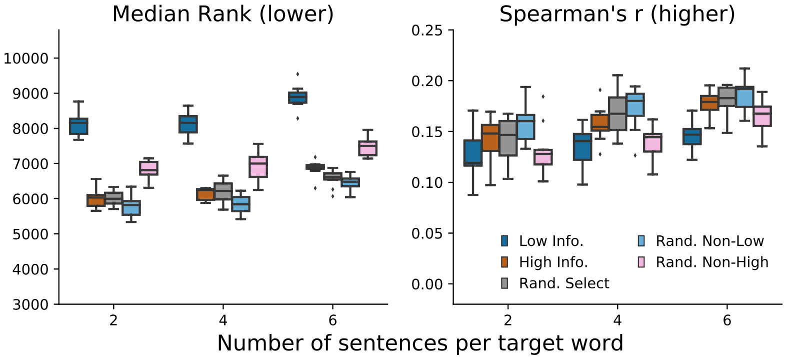


Figure 6.7: Few-shot learning results (Nonce2Vec) on SimLex-999 dataset. The low informative sentence and non-low informative sentence curricula provided the worst and best embedding quality respectively, compared to the gold-standard word vectors. Spearman's r score tended to increase as the number of sentences per target word increased, while there were no significant differences observed between curricula.

informativeness model fails in predicting contextual informativeness scores could improve the derived curriculum. For example, the predicted scores were less accurate for some sentences that contain less-frequent words or many special characters. This may due to BERT's small vocabulary size. Grammatically complex (or incorrect) sentences can be also difficult for current pre-trained NLP models to precisely encode contextual information. Example-based analysis, such as word-level permutation, to investigate which context words most impact prediction results (Kaushik et al., 2019), might be helpful for identifying various types of difficult cases.

Second, there is room to optimize the curriculum development strategies. As previous studies in vocabulary acquisition (Frishkoff et al., 2016a) and machine learning (Tsvetkov et al., 2016; Sachan and Xing, 2016) suggest, developing more effective curricula can be non-trivial. Different levels of informative contexts have different roles in learning (van den Broek et al., 2018). As we observed from Sec. 6.5.2

and 6.6.2, the non-low informative sentence models, which filtered out the less-informative sentences but kept diverse levels of informativeness scores for target sentences, performed better than the high-informative sentence models in almost every case. Developing more sophisticated learning strategies, such as using high-informative sentences for the initial higher loss state and low-informative sentences in a more mature model state, could be an interesting curriculum learning problem and opportunity to compare with human students learning in a scaffolded condition.

Third, we could further analyze what accounts for the curriculum effects we observed. For example, Word2Vec and FastText models' performance seemed to reach a plateau relatively quickly (e.g., $\geq 2^8$ sentences per target word (Figure 6.5.2)). Bigger models, such as transformer or LSTM based models, may have more performance headroom for testing the curriculum effect with a larger number of target sentences. With respect to curriculum content, target words with different parts of speech could be found in sentences within different grammatical structures. Some context words might be more informative than other context words (Kabbach et al., 2019). High-informative sentences may contain fewer redundant or frequent words than low-informative sentences. In this way, deeper understanding of the relationship between context words and a target word in a sentence could improve curriculum quality.

6.8 Conclusion

When learning vocabulary, the right curriculum can substantially improve students' learning. We demonstrate that an analogous approach for curriculum selection can help improve the representations learned by various types of word embedding models. In a series of experiments across batch learning and few-shot learning, we test the effectiveness of rating contexts by informativeness to prioritize those contexts likely to aid to learning word meaning. Our results show that (i) sentences predicted as low-informative by the model are indeed generally less effective for training word embedding models and (ii) in most cases, filtering out low-informative sentences from the training set substantially improves word representations for downstream tasks. In the future, we will further investigate how to build optimized curricula for word learning, and identify factors related to the curriculum effect based on contextual informativeness.

6.9 Author Contributions

Sungjin Nam was the main contributor to the study, including designing the study, organizing the crowdsourcing task for sentence annotations, developing the prediction model, and writing the manuscript. Dr. Kevyn Collins-Thompson and Dr. David Jurgens contributed to conducting the experiment and revising the manuscript.

Chapter 7

Discussion

This dissertation explored how we can predict students' latent cognitive states while they use an intelligent tutoring system, by analyzing implicit signals like behavioral interactions and linguistic inputs. Our findings would help instructors identify students' current states while using the system and provide data-driven evidence for system developers to design personalized features that can maximize their learning experience in various ways. In the following sections, we will discuss the broader implications of the findings from our studies and some future study plans.

7.1 Broader Implications

Identifying off-task states can provide important information to the system on when to intervene with a student during a learning task. We believe our findings in Chapter 3 can be transferable to other intelligent tutoring systems using linguistic inputs from students. For example, features based on semantic or lexical similarity scores between open-ended responses can be effective for identifying various types of disengaged behaviors in tutoring systems for non-vocabulary-related topics, such as social science or natural science, that focus on learning about the new concept. The features can also help to identify repetitive or related responses that are not addressing what actual questions ask to students.

Studies from other disciplines have suggested similar interpretable scales for distributional semantic models to understand changes in semantic senses of a word over time (Rudolph and Blei, 2018; Hamilton et al., 2016) or to quantify racial or gender bias incorporated in the models (Bolukbasi et al., 2016; Caliskan et al., 2017). However, these studies only examined the limited range of semantic senses that they wanted to capture from the computational model. Our findings in Chapter 4 present a more generalizable approach based on a classic semantic differential method (Osgood et al., 1957) which focused on providing a set of semantic senses that can be applicable in a wide range of words. We believe that Osgood's semantic scales can be used for various applications. For example, combining semantic differential scales and NLP models can be used to solve word-level semantic tasks like polysemy or antonym distinction, or other word disambiguation tasks that may benefit from using discrete semantic scales for effectively narrowing down the semantic search space. Moreover, we think the method could be used for characterizing the semantic content from longer texts, such as developing fine-grained semantic representations of product reviews or questions in community Q&A, to let system administrators understand the overall quality of text data and let users explore contents from other uses more effectively.

In Chapter 5, we presented a model for predicting contextual informativeness. We achieved competitive performance by using a vector representation of sentential context provided by pre-trained NLP models. Compared to a previous study, which required hundreds of lexical features to be processed for the prediction (Kapelner et al., 2018), our model provides a more flexible adaptation for texts from diverse domains by learning an effective representation automatically. For example, the prediction performance of our model can be easily improved as a more sophisticated pre-trained language model is developed in the future. Moreover, our results show that our model can be generalizable across different datasets, such as single-sentence contexts that researchers manually generated (Frishkoff et al., 2016a) or multi-sentence contexts crawled from the Internet (Kapelner et al., 2018). We believe our model can also be applied to calculate the contextual informativeness of other texts, like measuring the quality of a student's note, identifying more sophisticated questions from discussion forums, or selecting more informative instructional material to understand other domain-specific concepts.

Lastly, in Chapter 6, we showed how the contextual informativeness model could be used to distinguish less helpful sentences from the training data, and develop more efficient curricula for word embedding models. These results suggest that contextual word learning of human language learners and machine learning algorithms may share some similar attributes. In the future, existing studies on human language acquisition can be adapted for designing a more sophisticated curriculum developing strategies for machine learning models. Identifying the specific role of low and high informativeness sentences (Webb, 2008) in the machine learning curriculum can help achieve more robust and efficient learning strategies for NLP models. Identifying factors related to the curriculum effect based on contextual informativeness may also provide further insights for improving the performance of machine learning models and accurately modeling the human language acquisition process.

7.2 Limitations and Future Studies

Implementing the findings included in the dissertation would introduce various intelligent and adaptive features into a vocabulary tutoring system. For example, based on the off-task state prediction results, the system may adjust the question difficulty (e.g., showing easier or more informative sentences for off-task students in their next iterations), or display prompting messages that can encourage students to keep working on the task. As the student's partial knowledge state is evaluated from open-ended responses, the system can intelligently choose the next stimuli that allow the student to focus on improving the specific partial knowledge that the student currently lacks. Predicting the amount of contextual informativeness would help to automatically determine the quality of learning materials for contextual word learning can greatly improve the efficiency of developing personalized curricula. At the same time, being able to train efficient machine learning models through curriculum learning can provide more light-weighted and accessible educational applications for mobile devices or other less expensive computing devices.

Moreover, these findings can also be applied to other educational applications. Comparing the behavioral and linguistic sequence of responses, such as responses that are not related to the task but related to each other, would provide essential signals on users' off-task states. The semantic differential scaling method can be used flexibly. Once the instructor or system designer identifies semantic anchor terms that cover various semantic dimensions to evaluate, we believe the method can be applied to evaluate text responses from educational applications in diverse domains. The contextual informativeness model can be used to assess the instructional quality of various texts that are related to learning, including student notes, textbook phrases, or online forum posts. We can also select quality subsets from a corpus and quickly develop a domain-specific NLP model by using fewer but more informative sentences. And most importantly, integrating these features into a single system and investigating the interactions between developed features would significantly increase the effectiveness of a personalized learning system. For example, we can investigate the relationship between the off-task state and question difficulty based on the contextual informativeness score. There may be more optimal semantic scales to support predicting off-task responses or the amount of contextual informativeness of texts from various domains. These results will provide a more in-depth understanding of how students learn from an intelligent tutoring system, and how their learning behaviors are affected by machine-learning-based features in the educational application.

There are some limitations of studies that can be improved in the future. First, we presented data-driven evidence for designing a better vocabulary learning system. Yet, we did not test the effectiveness of suggested interventions or new design features. Many previous studies investigated different personalization features in information systems, like customized intervention messages (Szafir and Mutlu, 2012; Arroyo et al., 2007; Roll et al., 2007), contents (Shen et al., 2005; Teevan et al., 2010; Lagun and Lalmas, 2016), or user interfaces (Sarrafzadeh et al., 2016; Gajos et al., 2008; Reinecke and Bernstein, 2011), to maximize the user's satisfaction (O'Brien and Toms, 2008) and information gain (Baker et al., 2006). Integrating the findings from this dissertation into a contextual vocabulary learning system, such as DSCoVAR, would be worth investigating in the future. It would require careful experimental design to examine the effectiveness of various adaptive features and compare these results with the findings from previous studies.

Second, the studies in Chapters 3, 4, 5, and 6 only examined the English learning context. The linguistic features in these chapters were based on models that trained with the English language. Further examination on non-English-language vocabulary learning applications would examine the generalizability of the suggested features and may introduce unique challenges, especially with learning low-resource languages like Inuit or Sindhi. Moreover, if a vocabulary learning system is used by diverse groups of students, additional factors would be needed for accurate modeling of student behaviors. Different types of native languages or age groups can be important indications that may explain behavioral and linguistic interactions with a system. For example, Osgood illustrated how particular semantic scales can be more or less important to represent semantic senses of words in different languages and cultures (Osgood et al., 1957). A careful data collection approach and sophisticated machine learning method would be required to solve such a problem.

Third, some of computational linguistic models rely on relatively simple word embedding models (Mikolov et al., 2013; Herbelot and Baroni, 2017). For example, Chapters 3 and 4 used Word2Vec to extract predictive features. Using more sophisticated and context aware embedding models like ELMo (Peters et al., 2018) or BERT (Devlin et al., 2019) would provide more accurate representation of student responses collected from a vocabulary learning system, and improve its capacity to correctly understanding multi-word responses. Chapter 6 used Word2Vec and Nonce2Vec to evaluate the application of generated curricula. However, recent algorithms like transformers (Vaswani et al., 2017) can be used to represent more sophisticated and non-linear relationships between the target and context words, unlike Word2Vec based models that simply averages the information of the words within the context window to induce the semantic information of the target word (Mikolov et al., 2013). Further examination of curricula based on contextual informativeness and recent computational linguistic models would provide a more in-depth understanding of how recent deep-learning architectures utilize less or more rich contextual information to represent language.

Lastly, it would be interesting to investigate students' tendency in reacting to the stimuli from the vocabulary tutoring system. For example, off-task behaviors may be observed more from stimuli (e.g., cloze sentences) that are related to particular semantic information like a person's name or the names of activities. Partial knowledge measured on some scales can be easier to improve than other semantic senses, since students may be more sensitive to information related to particular semantic scales. Crowdsourced annotations for contextual informativeness can be interpreted in a similar way. Crowdworkers may be more susceptible to the particular lexical characteristics or semantic senses that they are more knowledgeable when they recall the potential responses for the cloze sentences. Investigating how human bias or subjectivity can affect language learning applications would be an interesting subject of future research. In the future, including these individual-level factors in online interactions or annotation data will enable deeper analysis on students' vocabulary learning behavior and improve understanding of how humans perceive contextual informativeness from a passage.

Chapter 8

Conclusion

This dissertation illustrated multiple machine learning applications that can help improve our understanding of students' vocabulary learning processes, and contribute to developing more automatized and intelligent vocabulary learning systems. We conclude the dissertation by summarizing the motivations, results, and overall contributions of these included studies.

Vocabulary Learning System: DSCoVAR We first introduced our vocabulary learning system, Dynamic Support of Contextual Vocabulary Acquisition for Reading (DSCoVAR): An Intelligent Tutoring System (Chapter 2). DSCoVAR is a contextual word learning system that lets students practice active inference techniques for learning the meaning of new words from sentential context. It is an online tutoring system that students can access through their desktop web browser. For the study, the system collected online interaction signals from students during their learning, including behavioral interaction and linguistic input data. We also generated sentence stimuli from research assistants and evaluated each sentence's informativeness quality based on crowdsourcing tasks. Based on these data, subsequent chapters investigated how to use behavioral and linguistic data to predict different cognitive states observed while using a vocabulary learning system.

Understanding Off-Task Behaviors Chapter 3 examined a prediction model to predict students' off-task states while using a vocabulary learning system. Based on existing studies on student engagement or gaming the system (Baker et al., 2010; Beck, 2005; Paquette et al., 2014) in online learning systems, we used online interaction signals collected from the system, and suggested predictive models using various features, including semantic and lexical features to capture characteristics of student responses.

An investigation of feature importance revealed that features about students' response history mostly improve the prediction performance. We could also identify different types of off-task responses: lexical repetitive responses or semantically related responses were the most common off-task responses in our vocabulary learning system. In the future, the findings from this study could be used to determine when the system could intervene with students to improve their engagement with the learning system to potentially further improve learning outcomes.

Capturing Partial Word Knowledge State Chapter 4 explored a more interpretable and scalable method to represent students' partial knowledge state during the vocabulary learning. Semantic differential scales (Osgood et al., 1957) are easily interpretable, but can be expensive to develop since they rely on human annotations. On the other hand, a word embedding model (Mikolov et al., 2013) can be trained with a large corpus quickly, but its high-dimensional vector representation can be often less interpretable. In this study, we tested how ten handpicked Osgood scales can be used to represent student responses in Word2Vec embedding space, and predicted students' short- and long-term learning outcome from a vocabulary learning system.

As a result, we found that combining Osgood's scales with Word2Vec can provide both interpretable representations and effective prediction results, compared to using a less interpretable vector output from the original Word2Vec model. In the future, we may expand this approach with more extensively selected semantic scales and more recent computational linguistic models to analyze longer passages in various domains.

Predicting Contextual Informativeness of Sentences Chapter 5 investigated how to build a model that can score contextual informativeness of sentence stimuli. Contextual word learning is known to be effective method to learn the meaning of words, especially if the word has complex or abstract meanings. However, not all contexts are equally informative for vocabulary learning. Being able to predict the amount of informativeness automatically can greatly reduce the cost of generating or collecting sentence stimuli for a contextual vocabulary learning system.

Compared to an existing study using a "kitchen sink" approach to feature representation (Kapelner et al., 2018), our deep learning approach using pre-trained language models showed better prediction performance and more generalizable results across different datasets. In the future, we will test how automatically generated curriculum can make human vocabulary learning more efficient.

Curriculum Development for Human and Machine Learners Like humans, many machine learning models also use contextual information to understand language. In Chapter 6, we showed that the contextual informativeness model from Chapter 5, originally developed for students' vocabulary learning, can be also helpful for developing more efficient curricula of machine learning models.

The results presented that sentences predicted as low-informative by the contextual informativeness model are also less effective for training word embedding models in batch learning (e.g., Word2Vec (Mikolov et al., 2013) and FastText (Bojanowski et al., 2017)) or few-shot learning (e.g., Nonce2Vec (Herbelot and Baroni, 2017)) settings. Moreover, simply filtering out low-informative sentences from the training set significantly improved the embedding models' performance. We think the effectiveness of curriculum learning can be further improved by identifying more optimized methods for machine learning models. Identifying factors related to the curriculum effect would be important to compare how humans and machines acquire knowledge of language.

Overall Contributions Our studies provided thorough and interpretable analysis that can improve understanding of students' latent learning states and provide data-driven evidence to develop more personalized features for a vocabulary learning system. It was possible by bridging research fields like psychology and machine learning and combining different techniques to extract meaningful information from behavioral and linguistic data that we collected from a vocabulary learning system. Sharing data that we collected from the studies, including online interaction data and sentence curriculum data, would help other researchers to investigate contextual word learning behaviors with online tutoring systems. Findings from our studies would also be applicable to understand various user states in other online systems, including user engagement, tracking learning progress, selecting better text content, and training more light and effective machine learning models based on behavioral and linguistic interaction signals. In the future, we could investigate how to integrate these findings into a single personalized vocabulary learning system, how to apply the findings to different types of inputs, such as non-English texts or interaction data from different learning systems, and investigate how the outcomes of our models may interact with students and shape their learning experience.

Bibliography

Suzanne Adlof, Gwen Frishkoff, Jennifer Dandy, and Charles Perfetti. 2016. Effects of induced orthographic and semantic knowledge on subsequent learning: A test of the partial knowledge hypothesis. *Reading and Writing*, 29(3):475–500.

Maria Antoniak and David Mimno. 2018. Evaluating the stability of embedding-based word similarities. *Transactions of the Association for Computational Linguistics*, 6:107–119.

Ivon Arroyo, Kimberly Ferguson, Jeffrey Johns, Toby Dragon, Hasmik Meheranian, Don Fisher, Andrew Barto, Sridhar Mahadevan, and Beverly Park Woolf. 2007. Repairing disengagement with non-invasive interventions. In *Artificial Intelligence in Education*, volume 2007, pages 195–202.

David Badre and Anthony D Wagner. 2004. Selection, integration, and conflict monitoring: assessing the nature and generality of prefrontal cognitive control mechanisms. *Neuron*, 41(3):473–487.

Ryan S Baker. 2007. Modeling and understanding students' off-task behavior in intelligent tutoring systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, pages 1059–1068. ACM.

Ryan S Baker, Albert T Corbett, and Kenneth R Koedinger. 2004. Detecting student misuse of intelligent tutoring systems. In *International Conference on Intelligent Tutoring Systems*, pages 531–540. Springer.

Ryan S Baker, Albert T Corbett, Kenneth R Koedinger, Shelley Evenson, Ido Roll, Angela Z Wagner, Meghan Naim, Jay Raspat, Daniel J Baker, and Joseph E Beck. 2006. Adapting to when students game an intelligent tutoring system. In *International Conference on Intelligent Tutoring Systems*, pages 392–401. Springer.

Ryan S Baker, Albert T Corbett, Ido Roll, and Kenneth R Koedinger. 2008a. Developing a generalizable detector of when students game the system. *User Modeling and User-Adapted Interaction*, 18(3):287–314.

Ryan S Baker, AT Corbett, I Roll, KR Koedinger, V Aleven, Mihaela Cocea, A Hershkovitz, AMJB de Caravalho, A Mitrovic, and M Mathews. 2013. Modeling and studying gaming the system with educational data mining. In *International Handbook of Metacognition and Learning Technologies*, pages 97–115. Springer.

Ryan S Baker, Sidney K D'Mello, Ma Mercedes T Rodrigo, and Arthur C Graesser. 2010. Better to be frustrated than bored: The incidence, persistence, and impact of learners' cognitive–affective states during interactions with three different computer-based learning environments. *International Journal of Human-Computer Studies*, 68(4):223–241.

Ryan S Baker, Jason A Walonoski, Neil T Heffernan, Ido Roll, Albert T Corbett, and Kenneth R Koedinger. 2008b. Why students engage in “gaming the system” behavior in interactive learning environments. *Journal of Interactive Learning Research*, 19(2):185–224.

Dana H Ballard and Chen Yu. 2003. A multimodal learning interface for word acquisition. In *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP’03).*, volume 5, pages 1–4. IEEE.

Douglas Bates, Martin Mächler, Ben Bolker, and Steve Walker. 2014. Fitting linear mixed-effects models using lme4. *arXiv preprint arXiv:1406.5823*.

Isabel L Beck, Margaret G McKeown, and Linda Kucan. 2013. *Bringing Words to Life: Robust Vocabulary Instruction*. Guilford Press.

Isabel L Beck, Margaret G McKeown, and Ellen S McCaslin. 1983. Vocabulary development: All contexts are not created equal. *The Elementary School Journal*, 83(3):177–181.

Joseph E Beck. 2004. Using response times to model student disengagement. In *Proceedings of the ITS2004 Workshop on Social and Emotional Intelligence in Learning Environments*, pages 13–20.

Joseph E Beck. 2005. Engagement tracing: Using response times to model student disengagement. *Artificial Intelligence in Education: Supporting Learning Through Intelligent and Socially Informed Technology*, 125:88.

Nicholas J. Belkin. 2008. Some(what) grand challenges for information retrieval. *SIGIR Forum*, 42(1):47–54.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 41–48. ACM.

Emmanuel Bizas, Panagiotis Simos, Cornelis J Stam, Steve Arvanitis, Dimitris Terzakis, and Sifis Micheloyannis. 1999. Eeg correlates of cerebral engagement in reading tasks. *Brain Topography*, 12(2):99–105.

Camille L Blachowicz, Peter J Fisher, Donna Ogle, and Susan Watts-Taffe. 2006. Vocabulary: Questions from the classroom. *Reading Research Quarterly*, 41(4):524–539.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3(Jan):993–1022.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in Neural Information Processing Systems*, pages 4349–4357.

Arielle Borovsky, Marta Kutas, and Jeff Elman. 2010. Learning to use words: Event-related potentials index single-shot contextual word learning. *Cognition*, 116(2):289–296.

Todd S Braver. 2012. The variablennature of cognitive control: A dual mechanisms framework. *Trends in Cognitive Sciences*, 16(2):106–113.

Gesa SE van den Broek, Atsuko Takashima, Eliane Segers, and Ludo Verhoeven. 2018. Contextual richness and word learning: Context enhances comprehension but retrieval enhances retention. *Language Learning*, 68(2):546–585.

Jonathan C Brown, Gwen A Frishkoff, and Maxine Eskenazi. 2005. Automatic question generation for vocabulary assessment. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 819–826. Association for Computational Linguistics.

Tom B. Brown et al. 2020. Language models are few-shot learners. *arXiv preprint arxiv.org/abs/2005.14165*.

Aylin Caliskan, Joanna J Bryson, and Arvind Narayanan. 2017. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186.

Ed H Chi and Peter Pirolli. 2006. Social information foraging and collaborative search. *Human Computer Interaction Consortium, Colorado, USA*.

Shivangi Chopra, Hannah Gautreau, Abeer Khan, and Lukasz Golab. 2018. Gender differences in undergraduate engineering applicants: A text mining approach. In

Proceedings of the 11th International Conference on Educational Data Mining, pages 44–54.

Mihaela Cocea and Stephan Weibelzahl. 2007. Eliciting motivation knowledge from log files towards motivation diagnosis for adaptive systems. In *International Conference on User Modeling*, pages 197–206. Springer.

Mihaela Cocea and Stephan Weibelzahl. 2009. Log file analysis for disengagement detection in e-learning environments. *User Modeling and User-Adapted Interaction*, 19(4):341–385.

Mihaela Cocea and Stephan Weibelzahl. 2011. Disengagement detection in online learning: validation studies and perspectives. *Learning Technologies, IEEE Transactions on*, 4(2):114–124.

Kevyn Collins-Thompson and Jamie Callan. 2004. Information retrieval for language tutoring: an overview of the REAP project. In *SIGIR 2004: Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Sheffield, UK, July 25-29, 2004*, pages 544–545. ACM.

Kevyn Collins-Thompson and Jamie Callan. 2007. Automatic and human scoring of word definition responses. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 476–483.

Kevyn Collins-Thompson, Gwen Frishkoff, and Scott Crossley. 2012. Definition response scoring with probabilistic ordinal regression. In *Proceedings of the International Conference on Computers in Education*, volume 6, pages 101–105.

Max Coltheart. 1981. The MRC psycholinguistic database. *The Quarterly Journal of Experimental Psychology*, 33(4):497–505.

Albert T Corbett and John R Anderson. 1994. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4(4):253–278.

Scott A Crossley, Kristopher Kyle, and Danielle S McNamara. 2016. The tool for the automatic analysis of text cohesion (taaco): Automatic assessment of local, global, and text cohesion. *Behavior Research Methods*, 48(4):1227–1237.

Scott A Crossley, Kristopher Kyle, and Danielle S McNamara. 2017. Sentiment analysis and social cognition engine (seance): An automatic tool for sentiment, social cognition, and social-order analysis. *Behavior Research Methods*, 49(3):803–821.

Martin Cutts. 2013. *Oxford guide to plain English*. OUP Oxford.

Edgar Dale. 1965. Vocabulary measurement: Techniques and major findings. *Elementary English*, 42(8):895–948.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of* *the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Raymond J Dolan. 2002. Emotion, cognition, and behavior. *Science*, 298(5596):1191–1194.

Francis T Durso and Wendelyn J Shore. 1991. Partial knowledge of word meanings. *Journal of Experimental Psychology: General*, 120(2):190.

Irina Elgort, Marc Brysbaert, Michaël Stevens, and Eva Van Assche. 2018. Contextual word learning during reading in a second language: An eye-movement study. *Studies in Second Language Acquisition*, 40(2):341–366.

Li Fei-Fei, Rob Fergus, and Pietro Perona. 2006. One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):594–611.

Adriano Ferraresi, Eros Zanchetta, Marco Baroni, and Silvia Bernardini. 2008. Introducing and evaluating ukwac, a very large web-derived corpus of english. In *Proceedings of the 4th Web as Corpus Workshop (WAC-4) Can we beat Google*, pages 47–54.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2002. Placing search in context: The concept revisited. *Placing Search in Context: The Concept Revisited*, 20(1):116–131.

Gwen Frishkoff, Kevyn Collins-Thompson, and Sungjin Nam. 2016a. Dynamic support of contextual vocabulary acquisition for reading: An intelligent tutoring system for contextual word learning. In Scott A Crossley and Danielle S McNamara, editors, *Adaptive Educational Technologies for Literacy Instruction.*, chapter 5, pages 69–81. Taylor & Francis, Routledge, New York, NY, USA.

Gwen Frishkoff, Kevyn Collins-Thompson, Sungjin Nam, Adeetee Bhide, Kim Muth, Leslie Hodges, and Charles Perfetti. 2015. The role of informativeness in contextual word learning: Evidence from a web-based tutoring system. In *Annual Meeting of the American Educational Research Association*. AERA.

Gwen Frishkoff, Kevyn Collins-Thompson, Charles Perfetti, and Jamie Callan. 2008. Measuring incremental changes in word knowledge: Experimental validation and implications for learning and assessment. *Behavior Research Methods*, 40(4):907–925.

Gwen Frishkoff, Sungjin Nam, and Kevyn Collins-Thompson. n.d. Design and implementation of an intelligent tutor for contextual word learning. Unpublished.

Gwen A Frishkoff, Kevyn Collins-Thompson, Leslie Hodges, and Scott Crossley. 2016b. Accuracy feedback improves word learning from context: Evidence from a meaning-generation task. *Reading and Writing*, 29(4):609–632.

Gwen A Frishkoff, Charles A Perfetti, and Kevyn Collins-Thompson. 2011. Predicting robust vocabulary growth from measures of incremental learning. *Scientific Studies of Reading*, 15(1):71–91.

Krzysztof Z Gajos, Jacob O Wobbrock, and Daniel S Weld. 2008. Improving the performance of motor-impaired users with automatically-generated, ability-based interfaces. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, CHI '08, pages 1257–1266. ACM.

Daniela Gerz, Ivan Vulić, Felix Hill, Roi Reichart, and Anna Korhonen. 2016. Simverb-3500: A large-scale evaluation set of verb similarity. *arXiv preprint arXiv:1608.00869*.

Jan Gläscher, Ralph Adolphs, Hanna Damasio, Antoine Bechara, David Rudrauf, Matthew Calamia, Lynn K Paul, and Daniel Tranel. 2012. Lesion mapping of cognitive control and value-based decision making in the prefrontal cortex. *Proceedings of the National Academy of Sciences*, 109(36):14681–14686.

Benjamin S Goldberg, Robert A Sottilare, Keith W Brawner, and Heather K Holden. 2011. Predicting learner engagement during well-defined and ill-defined computer-based intercultural interactions. In *Affective Computing and Intelligent Interaction*, pages 538–547. Springer.

Jamie C Gorman, Peter W Foltz, Preston A Kiekel, Melanie J Martin, and Nancy J Cooke. 2003. Evaluation of latent semantic analysis-based measures of team communications content. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 47(3):424–428.

William L Hamilton, Jure Leskovec, and Dan Jurafsky. 2016. Diachronic word embeddings reveal statistical laws of semantic change. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1489–1501.

S. Hassan and R. Mihalcea. 2011. Learning to identify educational materials. *ACM Trans. Speech Language Process.*, 8(2).

Marti A Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th Conference on Computational Linguistics-Volume 2*, pages 539–545. Association for Computational Linguistics.

Michael Heilman, Kevyn Collins-Thompson, Jamie Callan, Maxine Eskenazi, Alan Juffs, and Lois Wilson. 2010. Personalization of reading passages improves vocabulary acquisition. *International Journal of Artificial Intelligence in Education*, 20(1):73–98.

Aurélie Herbelot and Marco Baroni. 2017. High-risk learning: Acquiring new word vectors from tiny data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 304–309.

Jan Herrington, Ron Oliver, and Thomas C Reeves. 2003. Patterns of engagement in authentic online learning environments. *Australian Journal of Educational Technology*, 19(1):59–71.

Felix Hill, Roi Reichart, and Anna Korhonen. 2015. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Shufen Huang and Zohreh Eslami. 2013. The use of dictionary and contextual guessing strategies for vocabulary learning by advanced english-language learners. *English Language and Literature Studies*, 3(3):1.

Cory Hussar and Tatiana Pasternak. 2010. Trial-to-trial variability of the prefrontal neurons reveals the nature of their engagement in a motion discrimination task. *Proceedings of the National Academy of Sciences*, 107(50):21842–21847.

Joseph R Jenkins, Darlene Zabish Pany, and Janice Vanderploeg Schreck. 1978. Vocabulary and reading comprehension: Instructional effects. *Center for the Study of Reading Technical Report; no. 100.*

Jeffrey Johns and Beverly Woolf. 2006. A dynamic mixture model to detect student motivation and proficiency. In *Twenty-First AAAI Conference on Artificial Intelligence*, pages 163–168.

Daniel Jurafsky, Alan Bell, Michelle Gregory, and William D Raymond. 2001. Probabilistic relations between words: Evidence from reduction in lexical production. *Typological Studies in Language*, 45:229–254.

Alexandre Kabbach, Kristina Gulordava, and Aurélie Herbelot. 2019. Towards incremental learning of word embeddings using context informativeness. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 162–168.

Adam Kapelner, Jeanine Soterwood, Shalev Nessaiver, and Suzanne Adlof. 2018.

Predicting contextual informativeness for vocabulary learning. *IEEE Transactions on Learning Technologies*, 11(1):13–26.

Divyansh Kaushik, Eduard Hovy, and Zachary C Lipton. 2019. Learning the difference that makes a difference with counterfactually-augmented data. *arXiv preprint arXiv:1909.12434*.

Rachel L Kendal, Isabelle Coolen, and Kevin N Laland. 2004. The role of conformity in foraging when personal and social information conflict. *Behavioral Ecology*, 15(2):269–277.

Svetlana Kiritchenko and Saif Mohammad. 2017. Best-worst scaling more reliable than rating scales: A case study on sentiment intensity annotation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 465–470.

Svetlana Kiritchenko and Saif M Mohammad. 2016. Capturing reliable fine-grained sentiment associations by crowdsourcing and best–worst scaling. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 811–817, San Diego, California. Association for Computational Linguistics.

René F Kizilcec, Chris Piech, and Emily Schneider. 2013. Deconstructing disengagement: Analyzing learner subpopulations in massive open online courses. In *Proceedings of the Third International Conference on Learning Analytics & Knowledge*, LAK '13, pages 170–179. ACM.

Etienne Koechlin, Chrystele Ody, and Frédérique Kouneiher. 2003. The architecture of cognitive control in the human prefrontal cortex. *Science*, 302(5648):1181–1185.

Kenneth R Koedinger, Emma Brunskill, Ryan S Baker, Elizabeth A McLaughlin, and John Stamper. 2013. New potentials for data-driven intelligent tutoring system development and optimization. *AI Magazine*, 34(3):27–41.

Peter Kolb. 2008. Disco: A multilingual database of distributionally similar words. In *Proceedings of KONVENS*.

Shira Koren. 1999. Vocabulary instruction through hypertext: Are there advantages over conventional methods of teaching?. *Teaching English as a Second or Foreign Language*, 4(1):1–13.

Gerhard Kremer, Katrin Erk, Sebastian Padó, and Stefan Thater. 2014. What substitutes tell us - analysis of an "all-words" lexical substitution corpus. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 540–549.

Dmitry Lagun and Mounia Lalmas. 2016. Understanding user attention and engagement in online news reading. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*, pages 113–122. ACM.

Shibamouli Lahiri. 2015. Squinky! a corpus of sentence-level formality, informativeness, and implicature. *arXiv preprint arXiv:1506.02306*.

Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. 2015. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338.

Thomas K Landauer. 2006. *Latent Semantic Analysis*. Wiley Online Library.

Thomas K Landauer and Susan T Dumais. 1997. A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, 104(2):211.

Angeliki Lazaridou, Marco Marelli, and Marco Baroni. 2017. Multimodal word meaning induction from minimal exposure to natural text. *Cognitive Science*, 41:677–705.

Joseph LeDoux. 2003. The emotional brain, fear, and the amygdala. *Cellular and Molecular Neurobiology*, 23(4-5):727–738.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*.

Yitan Li, Linli Xu, Fei Tian, Liang Jiang, Xiaowei Zhong, and Enhong Chen. 2015. Word embedding revisited: A new representation learning and explicit matrix factorization perspective. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence, Buenos Aires, Argentina*, pages 3650–3656.

Qiao Liu, Haibin Zhang, Yifu Zeng, Ziqi Huang, and Zufeng Wu. 2018. Content attention model for aspect based sentiment analysis. In *Proceedings of the 2018*

World Wide Web Conference, pages 1023–1032. International World Wide Web Conferences Steering Committee.

Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal. Association for Computational Linguistics.

Yijun Ma, Lalitha Agnihotri, McGraw Hill Education, Ryan Baker, and Shirin Mojarad. 2016. Effect of student ability and question difficulty on duration. In *Proceedings of the 9th International Conference on Educational Data Mining*, pages 135–142.

Walter H MacGinitie, Ruth K MacGinitie, Katherine Maria, Lois G Dreyer, and Kay E Hughes. 2000. Gates-MacGinitie Reading Test (GMRT) fourth edition. `http://www.riversidepublishing.com/products/gmrt/`. Accessed: 2016-08-29.

Dimitris Margaritis. 2003. *Learning Bayesian network model structure from data*. Ph.D. thesis, Carnegie Mellon University.

Margo A Mastropieri, Thomas E Scruggs, Joel R Levin, Jan Gaffney, and Barbara McLoone. 1985. Mnemonic vocabulary instruction for learning disabled students. *Learning Disability Quarterly*, 8(1):57–63.

Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. In *Advances in Neural Information Processing Systems*, pages 6297–6308.

Diana McCarthy and Roberto Navigli. 2007. Semeval-2007 task 10: English lexical substitution task. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 48–53. Association for Computational Linguistics.

Danielle S McNamara, Arthur C Graesser, Philip M McCarthy, and Zhiqiang Cai. 2014. *Automated Evaluation of Text and Discourse with Coh-Metrix*. Cambridge University Press.

Oren Melamud, Jacob Goldberger, and Ido Dagan. 2016. context2vec: Learning generic context embedding with bidirectional LSTM. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 51–61.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Sungjin Nam, Gwen Frishkoff, and Kevyn Collins-Thompson. 2017. Predicting short- and long-term vocabulary learning via semantic features of partial word knowledge. In *Proceedings of the 10th International Conference on Educational Data Mining*, pages 80–87.

Sungjin Nam, Gwen Frishkoff, and Kevyn Collins-Thompson. 2018. Predicting students disengaged behaviors in an online meaning-generation task. *IEEE Transactions on Learning Technologies*, 11(3):362–375.

James H Neely. 1991. Semantic priming effects in visual word recognition: A selective review of current findings and theories. In *Basic Processes in Reading: Visual Word Recognition*, pages 264–336. Routledge.

John C Nesbit, Philip H Winne, Dianne Jamieson-Noel, Jillianne Code, Mingming Zhou, Ken MacAllister, Sharon Bratt, Wei Wang, and Allyson Hadwin. 2006. Using cognitive tools in gstudy to investigate how study activities covary with achievement goals. *Journal of Educational Computing Research*, 35(4):339–358.

Heather L O'Brien and Elaine G Toms. 2008. What is user engagement? a conceptual framework for defining user engagement with technology. *Journal of the American Society for Information Science and Technology*, 59(6):938–955.

Heather L O'Brien and Elaine G Toms. 2010. The development and evaluation of a survey to measure user engagement. *Journal of the American Society for Information Science and Technology*, 61(1):50–69.

Charles E Osgood, George J Suci, and Percy H Tannenbaum. 1957. *The Measurement of Meaning*. University of Illinois Press.

Korinn Ostrow, Christopher Donnelly, Seth Adjei, and Neil Heffernan. 2015. Improving student modeling through partial credit and problem difficulty. In

Proceedings of the Second ACM Conference on Learning@Scale, L@S '15, pages 11–20. ACM.

Patrick Pantel and Marco Pennacchiotti. 2006. Espresso: Leveraging generic patterns for automatically harvesting semantic relations. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, pages 113–120. Association for Computational Linguistics.

Jan Papoušek, Vít Stanislav, and Radek Pelánek. 2016. Impact of question difficulty on engagement and learning. In *International Conference on Intelligent Tutoring Systems*, pages 267–272. Springer.

Luc Paquette, Ryan S Baker, Adriana de Carvalho, and Jaclyn Ocumpaugh. 2015. Cross-system transfer of machine learned and knowledge engineered models of gaming the system. In *International Conference on User Modeling, Adaptation, and Personalization*, pages 183–194. Springer.

Luc Paquette, Adriana de Carvahlo, Ryan Baker, and Jaclyn Ocumpaugh. 2014. Reengineering the feature distillation process: A case study in detection of gaming the system. In *Proceedings of the 7th International Conference on Educational Data Mining*.

Ellie Pavlick and Marius Pasca. 2017. Identifying 1950s American jazz musicians: Fine-grained IsA extraction via modifier composition. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 2099–2109.

Philip I Pavlik and John R Anderson. 2005. Practice and forgetting effects on vocabulary memory: An activation-based model of the spacing effect. *Cognitive Science*, 29(4):559–586.

Judea Pearl. 2014. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Maxime Peyrard. 2018. A formal definition of importance for summarization. *arXiv preprint arXiv:1801.08991*.

Rosalind W Picard and Roalind Picard. 1997. *Affective Computing*, volume 252. MIT press Cambridge.

R Core Team. 2015. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Johnmarshall Reeve. 2012. A self-determination theory perspective on student engagement. In *Handbook of Research on Student Engagement*, pages 149–172. Springer.

Katharina Reinecke and Abraham Bernstein. 2011. Improving performance, perceived usability, and aesthetics with culturally adaptive user interfaces. *ACM Transactions on Computer-Human Interaction*, 18(2):8.

Soo Young Rieh, Kevyn Collins-Thompson, Preben Hansen, and Hye-Jung Lee. 2016. Towards searching as a learning process: A review of current perspectives and future directions. *Journal of Information Science*, 42(1):19–34.

Ido Roll, Vincent Aleven, Bruce M McLaren, and Kenneth R Koedinger. 2007. Can help seeking be tutored? searching for the secret sauce of metacognitive tutoring. In *Artificial Intelligence in Education*, volume 2007, pages 203–10.

Ido Roll, Vincent Aleven, Bruce M McLaren, and Kenneth R Koedinger. 2011. Improving students' help-seeking skills using metacognitive feedback in an intelligent tutoring system. *Learning and Instruction*, 21(2):267–280.

Jacobo Rouces, Nina Tahmasebi, Lars Borin, and Stian Rødven Eide. 2018. Generating a gold standard for a swedish sentiment lexicon. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation*, LREC '18.

Jonathan P Rowe, Lucy R Shores, Bradford W Mott, and James C Lester. 2010. Integrating learning and engagement in narrative-centered learning environments. In *International Conference on Intelligent Tutoring Systems*, pages 166–177. Springer.

Maja Rudolph and David Blei. 2018. Dynamic embeddings for language evolution. In *Proceedings of the 2018 World Wide Web Conference*, pages 1003–1011.

Mrinmaya Sachan and Eric Xing. 2016. Easy questions first? a case study on curriculum learning for question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 453–463.

Enrico Santus, Frances Yung, Alessandro Lenci, and Chu-Ren Huang. 2015. Evalution 1.0: an evolving semantic dataset for training and evaluation of distributional semantic models. In *Proceedings of the 4th Workshop on Linked Data in Linguistics: Resources and Applications*, pages 64–69.

Bahareh Sarrafzadeh, Alexandra Vtyurina, Edward Lank, and Olga Vechtomova. 2016. Knowledge graphs versus hierarchies: An analysis of user behaviours and perspectives in information seeking. In *Proceedings of the 2016 ACM on Conference on Human Information Interaction and Retrieval*, CHIIR '16, pages 91–100. ACM.

Martin Sarter, Ben Givens, and John P Bruno. 2001. The cognitive neuroscience of sustained attention: Where top-down meets bottom-up. *Brain Research Reviews*, 35(2):146–160.

Rebecca Scarborough. 2010. Lexical and contextual predictability: Confluent effects on the production of vowels. *Laboratory Phonology*, 10:557–586.

Marco Scutari. 2009. Learning Bayesian networks with the bnlearn r package. *arXiv preprint arXiv:0908.3817*.

Cyrus Shaoul, Harald Baayen, and Chris Westbury. 2014. N-gram probability effects in a cloze task. *The Mental Lexicon*, 9(3):437–472.

Xuehua Shen, Bin Tan, and ChengXiang Zhai. 2005. Context-sensitive information retrieval using implicit feedback. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '05, pages 43–50. ACM.

Wendelyn J Shore and Francis T Durso. 1990. Partial knowledge in vocabulary acquisition: General constraints and specific detail. *Journal of Educational Psychology*, 82(2):315.

Gale M Sinatra, Benjamin C Heddy, and Doug Lombardi. 2015. The challenges of defining and measuring student engagement in science. *Educational Psychologist*, 50(1):1–13.

Tanmay Sinha, Nan Li, Patrick Jermann, and Pierre Dillenbourg. 2014. Capturing "attrition intensifying" structural traits from didactic interaction sequences of MOOC learners. In *Proceedings of the EMNLP 2014 Workshop on Modeling Large Scale Social Interaction in Massively Open Online Courses*, EPFL-CONF-208843, pages 42–49.

Rion Snow, Daniel Jurafsky, and Andrew Y Ng. 2005. Learning syntactic patterns for automatic hypernym discovery. In *Advances in Neural Information Processing Systems*, pages 1297–1304.

Robert T Solman and Huei-Min Wu. 1995. Pictures as feedback in single word learning. *Educational Psychology*, 15(3):227–244.

Amanda Spink and Charles Cole. 2006. Human information behavior: Integrating diverse approaches and information use. *Journal of the American Society for information Science and Technology*, 57(1):25–35.

Daniel Szafir and Bilge Mutlu. 2012. Pay attention!: designing adaptive agents that monitor and improve user engagement. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 11–20. ACM.

György Szarvas, Chris Biemann, and Iryna Gurevych. 2013. Supervised all-words lexical substitution using delexicalized features. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1131–1141.

Jaime Teevan, Susan T Dumais, and Eric Horvitz. 2010. Potential for personalization. *ACM Transactions on Computer-Human Interaction*, 17(1):4.

Karl Halvor Teigen. 1994. Yerkes-dodson: A law for all seasons. *Theory & Psychology*, 4(4):525–547.

Antoine Tremblay, Dalhousie University, Johannes Ransijn, and University of Copenhagen. 2015. *LMERConvenienceFunctions: Model Selection and Post-hoc Analysis for (G)LMER Models*. R package version 2.10.

Yulia Tsvetkov, Manaal Faruqui, Wang Ling, Brian MacWhinney, and Chris Dyer. 2016. Learning the curriculum with bayesian optimization for task-specific word representation learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 130–139.

Peter D Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*, 37:141–188.

Seppo Vainio, Jukka Hyönä, and Anneli Pajunen. 2009. Lexical predictability exerts robust effects on fixation duration, but not on initial landing position during reading. *Experimental Psychology*, 56(1):66–74.

Eric G Van Inwegen, Seth A Adjei, Yan Wang, and Neil T Heffernan. 2015. Using partial credit and response history to model user knowledge. In *Proceedings of the 8th International Conference on Educational Data Mining*, pages 313–319.

Vincent Van Veen and Cameron S Carter. 2006. Conflict and cognitive control in the brain. *Current Directions in Psychological Science*, 15(5):237–240.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

R Joseph Waddington, Sungjin Nam, Steven Lonn, and Stephanie D Teasley. 2016. Improving early warning systems with categorized course resource usage. *Journal of Learning Analytics*, 3(3):263–290.

Jason A Walonoski and Neil T Heffernan. 2006. Detection and analysis of off-task gaming behavior in intelligent tutoring systems. In *International Conference on Intelligent Tutoring Systems*, pages 382–391. Springer.

Su Wang, Stephen Roller, and Katrin Erk. 2017. Distributional modeling on a diet: One-shot word learning from text only. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 204–213.

Yen-Hui Wang. 2016. Promoting contextual vocabulary learning through an adaptive computer-assisted efl reading system. *Journal of Computer Assisted Learning*, 32(4):291–303.

Paul Warren. 2012. *Introducing Psycholinguistics*. Cambridge University Press.

Stuart Webb. 2008. The effects of context on incidental vocabulary learning. *Reading in a Foreign Language*, 20(2):232–245.

Laura Wendlandt, Jonathan K Kummerfeld, and Rada Mihalcea. 2018. Factors influencing the surprising instability of word embeddings. In *Proceedings of NAACL-HLT*, pages 2092–2102.

Ryen White. 2013. Beliefs and biases in web search. In *Proceedings of the 36th international ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 3–12. ACM.

Tom D Wilson. 1997. Information behaviour: an interdisciplinary perspective. *Information Processing & Management*, 33(4):551–572.

Diyi Yang, Miaomiao Wen, Iris Howley, Robert Kraut, and Carolyn Rose. 2015. Exploring the effect of confusion in discussion forums of massive open online courses. In *Proceedings of the Second ACM Conference on Learning@Scale*, L@S '15, pages 121–130. ACM.

Lisa Marie Yonek. 2008. *The Effects of Rich Vocabulary Instruction on Students' Expository Writing*. Ph.D. thesis, University of Pittsburgh.

Geoffrey Zweig and Christopher JC Burges. 2011. The Microsoft Research Sentence Completion Challenge. *Microsoft Research Technical Report MSR-TR-2011–129.*

Appendix A

Creating and Annotating Cloze Sentence Dataset

A.1 Process for Generating Cloze Sentences

For the single-sentence context dataset, Our researchers manually generated sentences in different contextual informativeness levels with respect to the target word. We provided general instructions (Table A.1) for creating low, medium, and high informative cloze sentences. We also included example sentences and descriptions A.2) in different contextual informativeness levels and target words' part-of-speech to help researchers to create cloze sentences correctly. We additionally provided example phrases (Table A.3) and poor example sentences (Table A.4) for low contextual informative cases, to further control the quality of cloze sentences.

Lv.	Instructions
High	1. Use simple sentence structures. Note that target grade level for these contexts is 4th grade. All sentences should be between 9 and 13 words long (mean 11). 2. Use only easy, familiar words (except for target of course). Note that the average grade level for our contexts in the previous study was 4th grade. 3. Target word should be placed towards the end of the sentence whevever possible. You may find that this requirement completes with #1. Do your best. 4. Refer to list of synonyms, near synonyms, and cohorts. 5. Remember to avoid difficult (Tier 2) words. Look for simple synonyms and related words using LSA and/or a good thesaurus. 6. Each sentence should work with BOTH the very rare words and their Tier 2 synonyms. Please take the time to try to understand correct usage. Use the following website to look up actual usage of words if you're not sure: http://www.onelook.com (Onelook.com links to two sites that are particularly helpful: Vocabulary.com and https://www.wordnik.com, which provide a good range of example sentences, as well as definitions and explanations of correct usage).
Med	(in addition to 1-6 above) 7. It is not easy to know a priori whether a sentence will turn out (based on cloze data) to be medium or high constraint. In addition, this classification depends on the metric (e.g., whether we're looking at lexical/cloze data or at a derived measure that captures semantic constraint).Nonetheless, I've tried to assemble what I think are good a priori examples of Med vs. High C. Don't sweat the difference too much. We'll need cloze data to determine which way they fall.
Low	1. Use simple sentence structures. Note that target grade level for these contexts is 4th grade. All sentences should be between 9 and 13 words long (mean 11). 2. Use only easy, familiar words (except for target of course). Note that the average grade level for our contexts in the previous study was 4th grade. 3. Target word should be placed towards the end of the sentence whevever possible. You may find that this requirement completes with #1. Do your best. 4. For low-constraint sentences, avoid any content words (adjectives, nouns, or verbs) that could prime specific concepts.

Table A.1: The instructions for generating cloze sentences in different contextual informativeness levels.

Lv.	POS	Example sentence / Descriptions
High	Noun	*We covered our ears to block the loud ___ from the **crowd**.* Bolded words all prime the concept ‘noise’. ‘Ears’ and ‘loud’ are super constraining. Try out Hi-constraining sentences on a few friends, colleagues to ensure that they don’t come up with other concepts you didn’t think of when you created sentence. The cloze probability for ‘noise’ was 70%
	Adj	*Wendy **used to be fat, but** after her **illness** she looked ___.* ‘but’ indicates that missing word is opposite of ‘fat’ (thin, skinny, gaunt). Note that ‘illness’ suggests negative rather than positive characteristic (so ‘skinny’ or ‘gaunt’ were both common responses). The joint cloze probability for ‘thin’ + ‘skinny’ was > 70%.
	Verb	*The burglar was caught by police while trying to ___ the jewelry.* ‘burglar’, ‘police’ suggest criminal activity; ‘try to’ suggest activity was thwarted, so ‘buy’ for example would be pragmatically odd. The cloze probability for ”steal” was around 70%.
Med	Noun	*I enjoyed my flight to Paris except for all the ___.* ‘enjoy ___’ or ‘except ___’ suggests that ___ refers to something undesirable/unpleasant. On a flight there are usually lots of people, incl. screaming babies & small children → ‘noise’. However, there is also ‘turbulence’ and ‘delays’, two other concepts that were also provided by several respondents on the cloze task. The cloze probability for ‘noise’ was 13% (cloze for ‘turbulence’ was $\sim$25%)
	Adj	*The doctor warned the woman she was too ___ from a poor diet.* Could be ‘thin’ or ‘skinny,’ but also ‘fat’ or ‘sick.’ The joint cloze probability for ‘thin’ + ‘skinny’ was 29%.
	Verb	*Sam said he would have a million dollars if he ___ it.* People said ‘invest’, ‘earn’, and ‘save’ as often as they said ‘steal.’ The cloze probability for ‘steal’ was around 30%.
Low	Noun	*The group **did not choose that one** because of all the ___.* ‘did not choose’ is not very constraining since we don’t know what ‘that one’ refers to.
	Adj	*I was **surprised** to find that **it** was ___.* ‘surprise’ doesn’t suggest anything in particular about the characteristics of the noun (it); ‘it’ can refer to almost anything.
	Verb	*We were **interested** to learn that Sally had **decided not to** ___ .* ‘interested to learn’ is could refer to almost anything; same with ‘decided not to ___.’

Table A.2: The list of good cloze sentence examples with descriptions used for generating cloze sentences with different semantic constraint levels.

POS	Example Phrases
Noun	___ appeared/disappeared, turn into ___, ___ came into sight, think about/imagine ___, found/lost/discovered ___, remember/recall ___, buy/lend ___, write about ___, have ___, forget about ___
Adj	is/seems/looks/appears ___, decide/judge whether it is ___, become/turn ___, think/believe/know it is ___, say/argue it is ___
Verb	decide to ___, hard/impossible to ___, used to ___, ___ more/less often (in the future), agree to ___, see/watch someone ___, try to ___, have/need to ___, imagine what it's like to ___, learn (how) to ___, remember (how) to ___

Table A.3: Example phrases provided for generating low constraining sentences.

POS	Example sentence / Descriptions
Noun	*Morgan did not like Bob because she thought he was a(n) ___.* 'did not like' suggests that the target word refers to something undesirable/displeasing; linking target word to 'Bob' may bias reader to think of 'male' traits, occupations.
Adj	*I was disappointed to find that the boat was ___.* 'disappointed' suggests that the target word refers to something undesirable; 'the boat' is much too constraining (the target word can only refer to properties of boats).
Verb	*We were thrilled to learn that Sally had decided not to ___ .* 'thrilled to learn' has positive connotations (so it increases the constraint)

Table A.4: The poor examples for the low informative sentences with descriptions.

A.1.1 Iterative Refinement of Contexts

Most of our target words were lower-frequency, “Tier 2” words that are critical in writing but are rarely encountered in everyday speech. It is therefore not surprising that researchers (even those with excellent vocabulary) sometimes generated contexts that misuse target words. This might happen because researchers relied on the synonym and cohort words to come up with new examples, rather than retrieving them from high-quality published sources. In any case, new human-generated contexts must be vetted for correct usage.

A.2 Annotating the Single-Sentence Context Dataset

All crowdsourcing tasks were conducted on FigureEight[1] platform. Two types of crowdsourcing annotation were performed. The first task is the Best-Worst Scaling (BWS) annotations as described in Section 5.3.1 of the main paper. Workers were shown 10 tuples at a time per page of annotation and paid $0.25 per page. To ensure quality, during annotation one control question was randomly inserted per page, which had a known judgment (e.g., being the most informative sentence of a tuple). Workers were required to maintain at least 80% accuracy on these during annotation to continue annotating.

The choice of using 4-tuple sets for the task is based on previous studies and our own pilot testing. Each tuple set included four randomly selected different sentences

[1] `https://www.figure-eight.com/`

as options to be selected as the best or least informative sentences; each tuple was then scored by 3 different crowdworkers. We sampled sentences using the following criteria brought from Kiritchenko and Mohammad (2016).

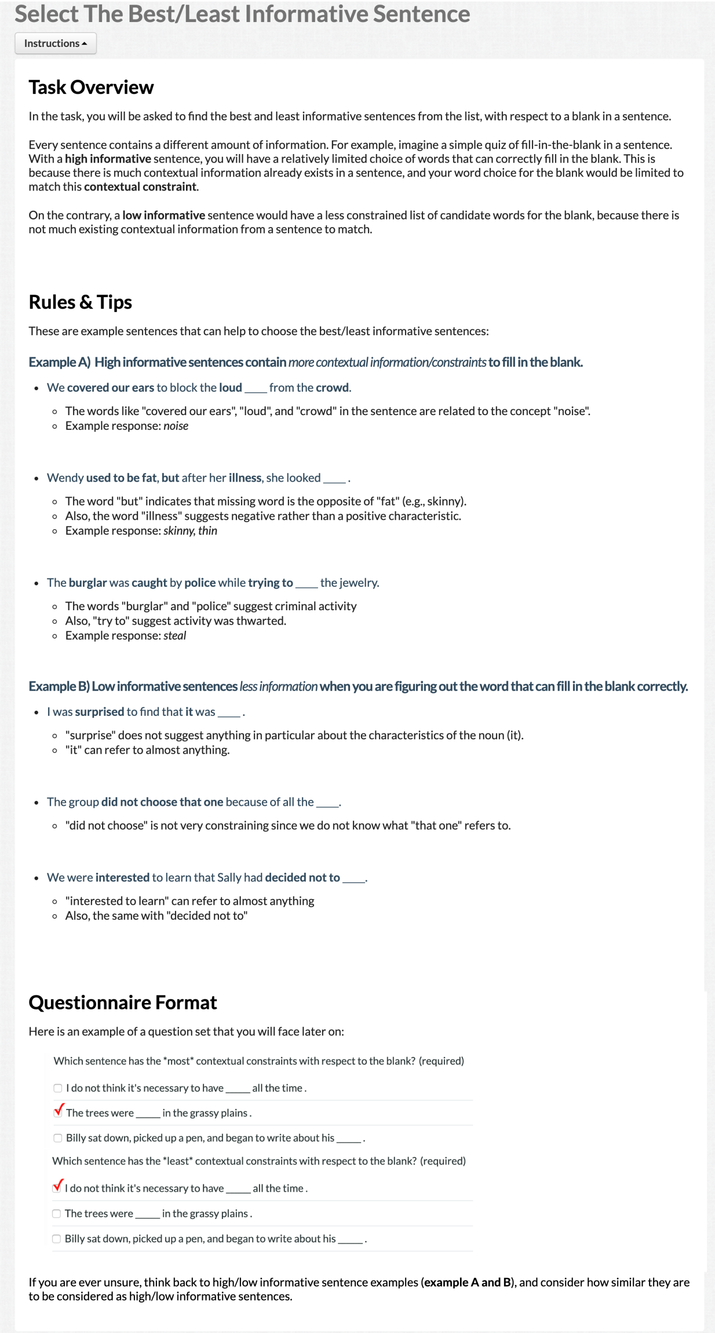
- No two k-tuples have the same k terms
- No two terms within a k-tuple are identical
- Each term in the term list appears approximately in the same number of k-tuples
- Each pair of terms appears approximately in the same number of k-tuples.

From the entire questionnaire, each sentence appeared in 8 different tuple sets (as $2 \times n$ 4-tuple sets were included in the entire questionnaire set, following Kiritchenko and Mohammad (2016)). In total, each sentence appeared as an option of the tuple set 24 times. If the sentence was marked as the most informative or the least informative sentence from the question, each rating was converted to an integer score ($+1$ or -1) respectively, while the unmarked case was considered to be 0.

Figure A.1 includes the instruction page that we presented to crowdworkers. It includes the definition of high or low contextual informativeness with examples.

A.2.1 Measuring Contextual Informativeness through Semantic Density

To complement the BWS-based scores of contextual informativeness, we also considered a second method of measuring contextual informativeness by examining



Select The Best/Least Informative Sentence

Task Overview

In the task, you will be asked to find the best and least informative sentences from the list, with respect to a blank in a sentence.

Every sentence contains a different amount of information. For example, imagine a simple quiz of fill-in-the-blank in a sentence. With a **high informative** sentence, you will have a relatively limited choice of words that can correctly fill in the blank. This is because there is much contextual information already exists in a sentence, and your word choice for the blank would be limited to match this **contextual constraint**.

On the contrary, a **low informative** sentence would have a less constrained list of candidate words for the blank, because there is not much existing contextual information from a sentence to match.

Rules & Tips

These are example sentences that can help to choose the best/least informative sentences:

Example A) High informative sentences contain ***more contextual information/constraints*** **to fill in the blank.**

- We **covered our ears** to block the **loud** ____ from the **crowd**.
    - The words like "covered our ears", "loud", and "crowd" in the sentence are related to the concept "noise".
    - Example response: *noise*
- Wendy **used to be fat**, **but** after her **illness**, she looked ____ .
    - The word "but" indicates that missing word is the opposite of "fat" (e.g., skinny).
    - Also, the word "illness" suggests negative rather than a positive characteristic.
    - Example response: *skinny, thin*
- The **burglar** was **caught** by **police** while **trying to** ____ the jewelry.
    - The words "burglar" and "police" suggest criminal activity
    - Also, "try to" suggest activity was thwarted.
    - Example response: *steal*

Example B) Low informative sentences *less information* **when you are figuring out the word that can fill in the blank correctly.**

- I was **surprised** to find that **it** was ____ .
    - "surprise" does not suggest anything in particular about the characteristics of the noun (it).
    - "it" can refer to almost anything.
- The group **did not choose that one** because of all the ____.
    - "did not choose" is not very constraining since we do not know what "that one" refers to.
- We were **interested** to learn that Sally had **decided not to** ____.
    - "interested to learn" can refer to almost anything
    - Also, the same with "decided not to"

Questionnaire Format

Here is an example of a question set that you will face later on:

Which sentence has the *most* contextual constraints with respect to the blank? (required)

☐ I do not think it's necessary to have _____ all the time .

☑ The trees were _____ in the grassy plains .

☐ Billy sat down, picked up a pen, and began to write about his _____ .

Which sentence has the *least* contextual constraints with respect to the blank? (required)

☑ I do not think it's necessary to have _____ all the time .

☐ The trees were _____ in the grassy plains .

☐ Billy sat down, picked up a pen, and began to write about his _____ .

If you are ever unsure, think back to high/low informative sentence examples (**example A and B**), and consider how similar they are to be considered as high/low informative sentences.

Figure A.1: Instructions used for collecting cloze sentence annotations.

the types of words annotators would insert into the sentence-sentence contexts in place of the target word. Here, each sentence is treated a cloze task where the target word is hidden and annotators are asked to suggest a substitute word in its place. Then, we computationally measure how different or similar are the semantics of the substitute words to quantify contextually informativeness. The intuition behind this scoring is highly informative contexts will select for words with very similar semantics that appear a dense cluster in a semantic space, where as uninformative contexts will select for words scattered throughout the semantic space. We refer to this measurement of contextual informativeness as *semantic density.*

Crowdworkers provided a single word that correctly completes the sentence. We collected 30 responses each for 1783 cloze sentences (53490 responses total). Collected cloze responses can be considered as how crowdworkers interpreted the contextual informativeness with respect to the blank in a cloze sentence. Crowdworkers were restricted to workers in the US, Canada, and the UK. Instructions used for collecting cloze responses can be found in Table A.5.

The contextual informativeness of the cloze responses is then converted into a semantic density score. The semantic density score relies on the crowdworkers' lexical prediction activity for cloze sentences (Vainio et al., 2009). This can be a complex activity that requires multiple mental stages, such as comprehending the given sentence and recall the vocabulary that fits. Thus, the scoring results may be different form how people perceive the level of informativeness semantically (Scarborough, 2010; Neely, 1991)

To calculate the semantic density score, we first use GloVe (Pennington et al.,

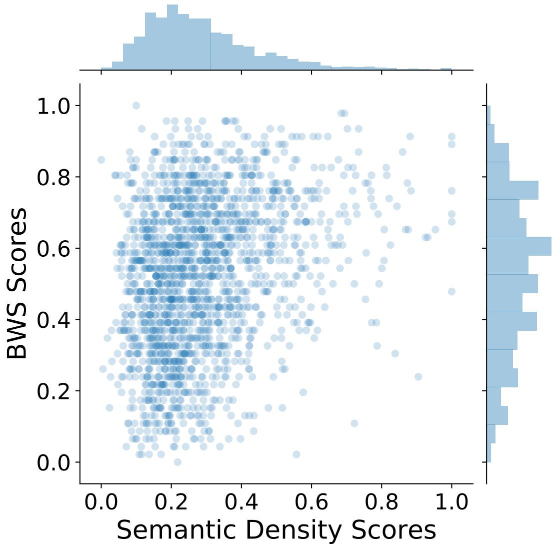


Figure A.2: Distributions of the semantic density score and the BWS score (Both were min-max normalized). Two scores were moderately correlated.

2014) to retrieve the vector for a cloze response within a sentence, and calculate the weighted average of cosine similarity between cloze-response pairs for each cloze sentence (Equation A.1). Although the outcome form GloVe might not provide the most accurate semantic similarity scores between cloze responses, we could separate distributional embeddings used for the prediction model and semantic density scores. The *BWS* score was moderately correlated with the *semantic density* score ($r = 0.292, p < 0.001$ in Spearman's rank correlation).

$$S(X) = \frac{1}{\binom{N}{2}} \sum_{i=1}^{N} \sum_{j=1}^{N} sim(x_i, x_j) \quad (i \neq j; N = 30) \tag{A.1}$$

A.2.2 Instructions for Collecting Cloze Responses

Overview	This task is a very simple fill-in-the-blanks quiz where you will see some text and be asked to enter the missing word. You will be provided a few sentences and there will be one word missing, represented by "_____". Your job is to guess what that word is.
Rules	Do not search the web to find hints or solutions to the question. All answers must be a single word. This means your answer should just be one word (only alphabets A-Z).
Example	The quick brown ___ jumped over the fence. *Missing word: fox*

Table A.5: Instructions for Collecting Cloze Responses

S_{Low}: We weren't able to tell if he was a ____ or not. *Original Target Word:* recluse *BWS Score:* 0.239, *Sem. Den. Score:* 0.027 *Crowd Substitutes:* cop(2), idiot(2), criminal(2), ...
S_{High}: The barks and howls of dogs created too much ____ for us to sleep. *Original Target Word:* din *BWS Score:* 0.913, *Sem. Den. Score:* 0.881 *Crowd Substitutes:* noise(28), racket(1), rouse(1)

Table A.6: Each sentence may have a different amount of information with respect to a target word. In the proposed task, we let systems rate the informativeness of the sentential context with respect to a target word (shown as a blank), ranging from low contextual informativeness (top) to high (bottom). Numbers in parentheses show the frequency of responses that we collected from a crowdsourcing task.

Appendix B

Informativeness Model Analysis

B.1 Hyperparameters

During the training process of ELMo-based and BERT-based models, we fine-tuned the pre-trained models. Because of the differences in the number of trainable parameters of each pre-trained model, we used different learning rates for each ELMo-based (1e−3) and BERT-based (3e−5) model. Other hyper-parameters remained constant across models (batch size: 16, iteration: 5 (for the single-sentence) or 3 (for the multi-sentence context dataset). The dimension of ReLu layer was 256. The dimensions for the attention block layers used same dimensions with the pre-trained embeddings (ELMo: 1024, BERT: 768).

For the baseline model using co-occurrence information, the ridge regression model was trained with `scikit-learn`'s default alpha value. Co-occurrence matrix was built for words that appeared more than five times in the training data.

The replicated random forest model from Kapelner et al. (2018) followed the original paper's setting, including setting the number of estimators as 500, and bootstrapping sample size as 10000.

B.2 Computing Resource for Training

For this study, we used a single NVIDIA 2080 TI GPU with Intel i7 CPU. For training the model with the single-sentence context dataset, it took about 1 minute per fold (90% of the data). For the multi-sentence context dataset, it took approximately 30 minutes per fold.

We used pre-trained versions of ELMo (Peters et al., 2018, `https://tfhub.dev/google/elmo/2`) and BERT (Devlin et al., 2019, `https://tfhub.dev/tensorflow/bert_en_cased_L-12_H-768_A-12/1`) from the public repository. Our ELMo-based model with attention block had about 426k trainable parameters, while the BERT-based counterpart had about 7.3M trainable parameters.

B.3 Additional Analysis Results

The following sections include additional analysis results that we did not included in Sections 5 and 6.

B.3.1 Single-Sentence Contexts

Table B.1 shows more details of suggested models' performance on both semantic density and BWS informative scores of the single-sentence dataset.

(Sem. Den.)	RMSE	↓ 20% Info	50:50	↑ 20% Info
Base:Avg.	0.163 (0.151, 0.176)	0.500 (0.500, 0.500)	0.500 (0.500, 0.500)	0.500 (0.500, 0.500)
Base:BoW	0.162 (0.149, 0.175)	0.556 (0.522, 0.591)	0.559 (0.520, 0.597)	0.550 (0.507, 0.593)
Base:Length	0.198 (0.183, 0.214)	0.572 (0.535, 0.608)	0.573 (0.552, 0.595)	0.556 (0.515, 0.597)
ELMo	0.166 (0.154, 0.178)	0.608 (0.557, 0.659)	0.616 (0.565, 0.667)	0.628 (0.589, 0.667)
ELMo+Att	0.162 (0.149, 0.174)	0.623 (0.564, 0.682)	0.624 (0.585, 0.664)	0.636 (0.600, 0.672)
BERT	0.203 (0.195, 0.211)	0.590 (0.545, 0.636)	0.581 (0.563, 0.599)	0.580 (0.544, 0.616)
BERT+Att	0.164 (0.151, 0.176)	0.631 (0.599, 0.664)	0.607 (0.571, 0.643)	0.623 (0.591, 0.656)
(BWS)	RMSE	↓ 20% Info	50:50	↑ 20% Info
Base:Avg.	0.315 (0.304, 0.326)	0.500 (0.500, 0.500)	0.500 (0.500, 0.500)	0.500 (0.500, 0.500)
Base:BoW	0.308 (0.298, 0.319)	0.638 (0.587, 0.689)	0.594 (0.542, 0.646)	0.598 (0.552, 0.643)
Base:Length	0.321 (0.311, 0.331)	0.781 (0.762, 0.799)	0.755 (0.736, 0.775)	0.749 (0.725, 0.773)
ELMo	0.179 (0.170, 0.188)	0.858 (0.828, 0.887)	0.806 (0.784, 0.827)	0.775 (0.748, 0.802)
ELMo+Att	0.166 (0.159, 0.173)	0.868 (0.838, 0.898)	0.810 (0.787, 0.834)	0.778 (0.749, 0.807)
BERT	0.201 (0.192, 0.210)	0.806 (0.763, 0.849)	0.743 (0.710, 0.775)	0.707 (0.672, 0.743)
BERT+Att	0.154 (0.146, 0.162)	0.895 (0.875, 0.916)	0.842 (0.824, 0.860)	0.791 (0.773, 0.809)

Table B.1: Average RMSE and binary classification results (ROCAUC) with the single-sentence context dataset. Informativeness scores were defined as semantic density (top) and BWS scores (bottom). For both ELMo and BERT-based models, adding the attention block (`+Att`) improved the performance for predicting continuous score range (*RMSE*) and various classification scenarios (`ROCAUC`). Numbers in parentheses are the 95% confidence interval.

B.3.2 Multi-Sentence Contexts

Lexical Features from Kapelner et al. (2018) Kapelner et al. (2018) used lexical features including n-gram frequencies from Google API, Coh-Metrix features (McNamara et al., 2014), sentiments (Crossley et al., 2017), psycholinguistic features (Crossley et al., 2016), and other lexical sophistication features (Crossley et al., 2016; Kolb, 2008).

In their original paper, the authors reported that the most important lexical features from contexts include top 10 words that include synonymous words from the target, top 10 context words that frequently collocate with the target words, frequency of the target word, context words' politeness, age of acquisition, and meaningfullness of context words.

Table B.2 shows more details of suggested models' performance on the multi-sentence context dataset.

B.3.3 Cross-Predicting Contexts

Table B.3 shows that the ELMo-based and the BERT-based model that trained with the single-sentence context dataset did not performed well in predicting the multi-sentence context dataset.

B.3.4 EVALUtion dataset

Table B.4 shows the list of relations and templates that Santus et al. (2015) used to create example sentences.

Table B.5 includes the randomized rank scores for each relation that used as a baseline for EVALution dataset (Santus et al., 2015). Table B.6 shows the results from models trained with the single-sentence context data. And Table B.7 shows the results from models trained with the multi-sentence context data.

	RMSE	↓ 20% Info	50:50	↑ 20% Info
Base:Avg.	0.173, (0.170, 0.176)	0.500 (0.500, 0.500)	0.500 (0.500, 0.500)	0.500 (0.500, 0.500)
Base:Length	0.173, (0.170, 0.176)	0.511 (0.505, 0.517)	0.507 (0.500, 0.514)	0.502 (0.495, 0.509)
Base:BoW	0.209, (0.207, 0.211)	0.630 (0.620, 0.640)	0.589 (0.582, 0.596)	0.579 (0.572, 0.586)
Kapelner et al. (2018)	0.157, (0.154, 0.159)	0.736 (0.729, 0.743)	0.698 (0.691, 0.705)	0.680 (0.669, 0.692)
ELMo	0.152, (0.146, 0.159)	0.768 (0.757, 0.779)	0.729 (0.721, 0.737)	0.705 (0.696, 0.715)
ELMo+Att	0.153, (0.149, 0.156)	0.770 (0.760, 0.780)	0.727 (0.720, 0.734)	0.701 (0.689, 0.713)
ELMo+Att+Lex	0.152, (0.149, 0.155)	0.789 (0.779, 0.799)	0.746 (0.739, 0.754)	0.725 (0.719, 0.731)
BERT	0.139, (0.136, 0.142)	0.807 (0.797, 0.817)	0.764 (0.757, 0.772)	0.751 (0.739, 0.763)
BERT+Att	0.138, (0.136, 0.140)	0.816 (0.806, 0.825)	0.777 (0.770, 0.785)	0.768 (0.757, 0.778)
BERT+Att+Lex	0.145, (0.142, 0.149)	0.822 (0.814, 0.831)	0.782 (0.775, 0.788)	0.773 (0.765, 0.781)

Table B.2: Average RMSE and binary classification results (ROCAUC) with the multi-sentence context dataset (Kapelner et al., 2018). The BERT-based model with the attention block performed significantly better than the baseline and ELMo-based models, in terms of RMSE and ROCAUC scores. Adding features from the original paper (Kapelner et al., 2018) (`+Lex`) also increased the prediction performance. Numbers in parentheses are the 95% confidence interval.

	↓ 20% Info	50:50	↑ 20% Info
ELMo+Att	0.577	0.556	0.532
BERT+Att	0.502	0.497	0.492

Table B.3: ROCAUC results for predicting the multi-sentence contexts from models trained with the single-sentence context dataset. Prediction performances are slightly better than random (e.g., $ROCAUC = 0.5$))

Relation	Pairs	Relata	Senence template
IsA (hypernym)	1880	1296	X is a *kind* of Y
Antonym	1660	1144	X can be used as the *opposite* of Y
Synonym	1086	1019	X can be used with the *same* meaning of Y
Meronym	1003	978	X is ...
- PartOf	654	599	... *part* of Y
- MemberOf	32	52	... *member* of Y
- MadeOf	317	327	... *made* of Y
Entailment	82	132	If X is *true*, then also Y is *true*
HasA (possession)	544	460	X can *have* or can *contain* Y
HasProperty (attribute)	1297	770	Y is to *specify* X

Table B.4: The list of relations, number of pairs, and sentence template examples from Santus et al. (2015). In a sentence template, X means the target word, while Y is the pair, and italicized words are relational cues.

Relations	`Rdm:pair`	`Rdm:rcue`
Antonym	0.507 (0.491, 0.523)	0.494 (0.478, 0.510)
Entails	0.482 (0.414, 0.550)	0.452 (0.380, 0.524)
HasA	0.485 (0.457, 0.513)	0.507 (0.479, 0.535)
HasProperty	0.498 (0.479, 0.516)	0.507 (0.489, 0.525)
IsA	0.508 (0.492, 0.523)	0.502 (0.487, 0.517)
MadeOf	0.550 (0.512, 0.587)	0.487 (0.446, 0.527)
MemberOf	0.425 (0.307, 0.543)	0.494 (0.375, 0.612)
PartOf	0.501 (0.473, 0.529)	0.476 (0.449, 0.502)
Synonym	0.493 (0.474, 0.512)	0.503 (0.484, 0.522)
Overall Avg.	0.503 (0.495, 0.510)	0.498 (0.491, 0.506)

Table B.5: Random baseline performance for EVALution dataset (Santus et al., 2015).

Relations	`ELMo+Att:pair`	`ELMo+Att:rcue`	`BERT+Att:pair`	`BERT+Att:rcue`
Antonym	0.727 (0.724, 0.730)	0.603 (0.599, 0.606)	0.353 (0.346, 0.359)	0.501 (0.494, 0.508)
Entails	0.364 (0.346, 0.382)	0.210 (0.183, 0.237)	0.451 (0.415, 0.486)	0.514 (0.487, 0.540)
HasA	0.509 (0.504, 0.515)	0.257 (0.252, 0.261)	0.527 (0.515, 0.540)	0.430 (0.427, 0.433)
HasProperty	0.405 (0.403, 0.408)	0.795 (0.794, 0.797)	0.031 (0.027, 0.035)	0.592 (0.590, 0.595)
IsA	0.400 (0.397, 0.403)	0.795 (0.793, 0.796)	0.024 (0.020, 0.027)	0.437 (0.432, 0.441)
MadeOf	0.504 (0.499, 0.510)	0.259 (0.253, 0.265)	0.007 (0.002, 0.012)	0.740 (0.734, 0.747)
MemberOf	0.400 (0.400, 0.400)	0.800 (0.800, 0.800)	0.056 (0.014, 0.098)	0.569 (0.536, 0.601)
PartOf	0.504 (0.501, 0.507)	0.250 (0.248, 0.252)	0.023 (0.017, 0.029)	0.711 (0.703, 0.719)
Synonym	0.771 (0.768, 0.774)	0.477 (0.473, 0.481)	0.567 (0.557, 0.577)	0.608 (0.600, 0.617)
Overall Avg.	0.546 (0.542, 0.549)	0.592 (0.587, 0.597)	0.215 (0.209, 0.220)	0.540 (0.537, 0.543)

Table B.6: Comparing the average of normalized rank scores of attention weights from single-sentenced trained models across various relations in EVALution dataset (Santus et al., 2015). Higher score means better. The `ELMo+Att` model perform better than the *BERT+Att* model in both `pair` and `rcue` scores. However, the performance is less accurate than multi-sentence dataset models in Table B.7.

Relations	`ELMo+Att:pair`	`ELMo+Att:rcue`	`BERT+Att:pair`	`BERT+Att:rcue`
Antonym	0.505 (0.504, 0.507)	0.253 (0.251, 0.254)	0.462 (0.451, 0.474)	0.599 (0.588, 0.610)
Entails	0.954 (0.919, 0.989)	0.394 (0.377, 0.411)	0.664 (0.616, 0.711)	0.616 (0.585, 0.647)
HasA	0.310 (0.304, 0.317)	0.758 (0.755, 0.762)	0.538 (0.518, 0.559)	0.828 (0.818, 0.837)
HasProperty	0.998 (0.996, 1.000)	0.209 (0.205, 0.212)	0.562 (0.557, 0.566)	0.234 (0.228, 0.240)
IsA	0.990 (0.987, 0.993)	0.212 (0.209, 0.215)	0.230 (0.227, 0.234)	0.911 (0.904, 0.919)
MadeOf	0.998 (0.995, 1.001)	0.692 (0.681, 0.704)	0.974 (0.958, 0.989)	0.741 (0.735, 0.746)
MemberOf	0.994 (0.981, 1.006)	0.200 (0.200, 0.200)	0.819 (0.747, 0.891)	0.431 (0.386, 0.477)
PartOf	0.995 (0.992, 0.998)	0.705 (0.697, 0.712)	0.975 (0.966, 0.983)	0.740 (0.734, 0.746)
Synonym	0.777 (0.775, 0.780)	0.235 (0.231, 0.239)	0.771 (0.764, 0.778)	0.160 (0.153, 0.167)
Overall Avg.	0.808 (0.802, 0.814)	0.328 (0.324, 0.333)	0.542 (0.535, 0.548)	0.585 (0.577, 0.592)

Table B.7: Comparing the average of normalized rank scores of attention weights from multi-sentenced trained models across various relations in EVALution dataset (Santus et al., 2015). Higher score means better. A better model should pay more weights to the word that pairs with the target word (`pair`) or words that can indicate the type of relationship between the target and the pair word (`rcue`). Overall, the `ELMo+Att` performs better in capturing the `pair` words, while the `BERT+Att` model is better in capturing the `rcue` words.

Appendix C

Curriculum Learning Analysis

C.1 Contextual Informativeness Model

We used a deep learning based contextual informativeness model (Figure C.1) from Chapter 5.

We used root mean square error (RMSE) as a loss function. During training, we fine-tuned BERT's last encoding layer. After the validation process, we chose hyper-parameters as following:

- Batch size: 16
- Iteration: 3
- Learning rate: 3e−5
- ReLu layer dimension: 256.
- Attention block layers dimension: 768 (same as BERT's output)

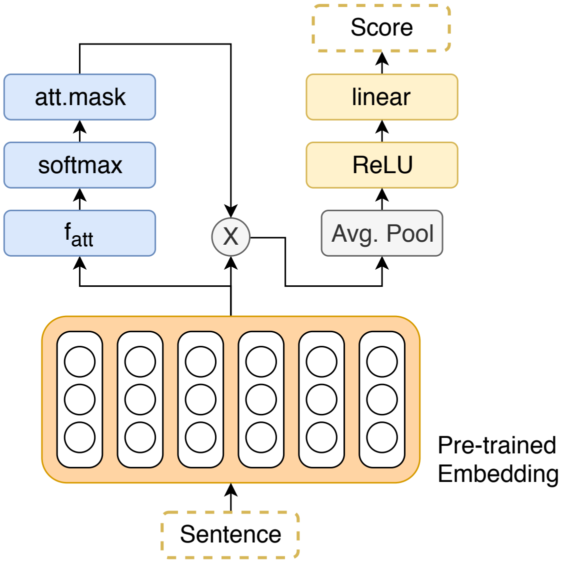


Figure C.1: The contextual informativeness prediction model used a pre-trained BERT model (Devlin et al., 2019) (orange block) to initially represent context words. Masked attention blocks (blue) provided the attention distribution of context words with respect to the target word, and calculated the attention-weighted context representation. Lastly, the regression block (yellow) predicted the contextual informativeness score of a passage. More details are in Appendix C.1.

For the study, we used our replication of the random forest model from Kapelner et al. (2018). We followed the original paper's setting, including setting the number of estimators as 500, and bootstrapping sample size as 10000. We validate our comparison by replicating their original model results R^2=0.179 with very similar results R^2=0.177 in our replication.

When lexical features from Kapelner et al. (2018) were used with contextual embedding features, we could observe a small performance increase (RMSE: 0.145, (0.142, 0.149), ↓ 20% Info.: 0.822 (0.814, 0.831), 50:50: 0.782 (0.775, 0.788), ↑ 20% Info.: 0.773 (0.765, 0.781)).

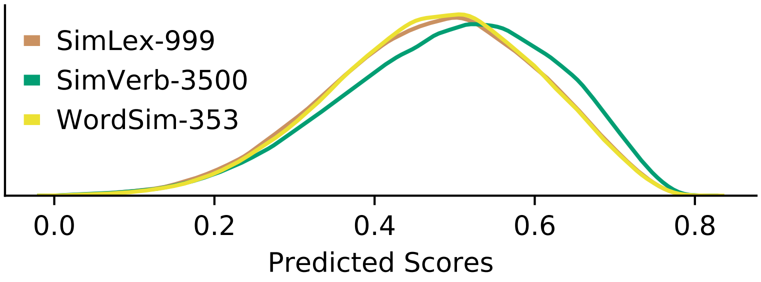


Figure C.2: Distributions of predicted contextual informativeness scores for target sentences from ukWaC corpus. Predicted scores were centered around the center, with enough number of relatively low or high informative sentences ($\mu \approx 0.5$, $\sigma \approx 0.13$).

C.2 Contextual Informativeness Scores for Target Sentences

Before the experiments, we briefly explored properties of target sentences. Figure C.2 shows the distribution of the predicted scores from the contextual informativeness model. For all three semantic similarity tasks, contextual informativeness scores for target sentences were distributed around the score of 0.5, and showed that there were enough number of low or high informative sentences to test different curriculum building heuristics ($\sigma \approx 0.13$).

Figure C.3 shows that sentence lengths and predicted informative scores of SimLex-999's target sentences in ukWaC corpus are not correlated. This ensures that similar amount of lexical information was provided to the embedding models throughout the different curricula. Target sentences for SimVerb-3500 and WordSim-353 tasks also showed similar results.

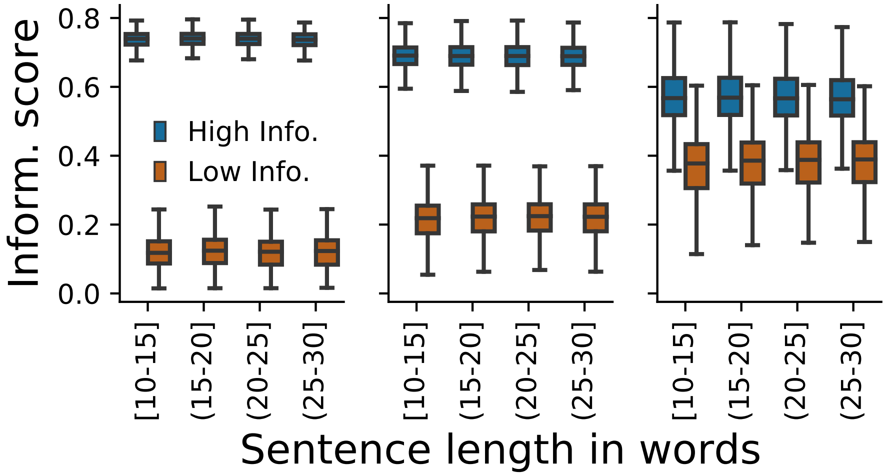


Figure C.3: Relationship between sentence length and contextual informativeness predictions. Each plot represents when 4, 32, and 256 sentences are selected per target word. Distributions of the predicted scores are not affected by the length of sentences.

C.3 SimLex-999: by POS and Association Pairs

Figure C.4 shows the comparison between curricula from the random curriculum by different POS and association pairs of SimLex-999 target words. We can observe similar patterns from other similarity tasks' results, as the low informative models perform worse than other curriculum and the non-low informative sentence models perform better than other cases.

C.4 Word2Vec and FastText Models

For consistent analysis results, we used the same hyper-parameters to train the background models for Word2Vec, FastText, and Nonce2Vec:

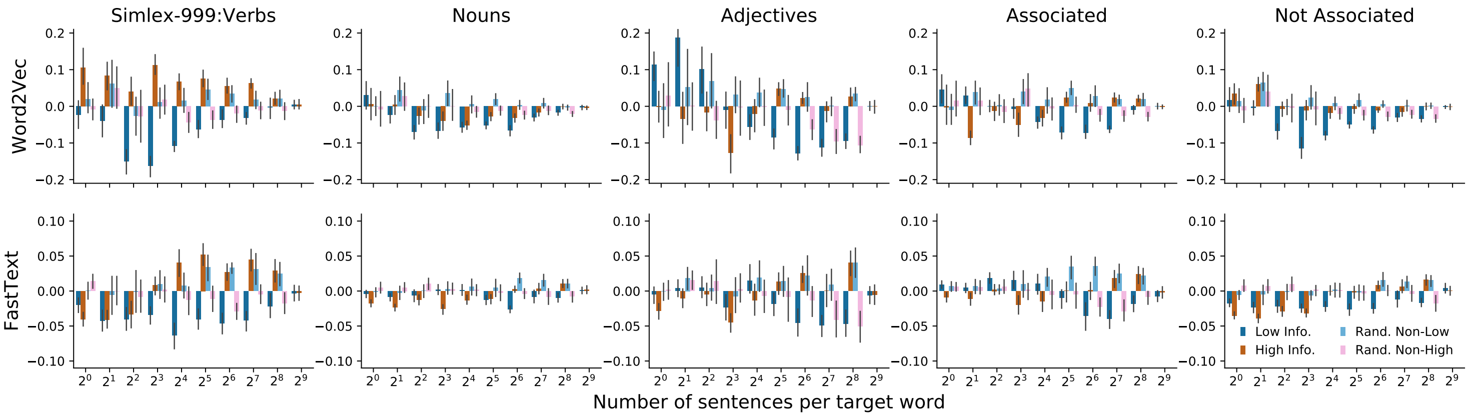


Figure C.4: Performance differences between curricula from the random curriculum in different POS and association pairs of SimLex-999 target words. Notice scales are different between Word2Vec and FastText models.

- Skip-gram algorithm
- Embedding dimension: 400
- Window size: 5
- Negative sampling words: 5
- Minimum word counts: 50
- Alpha: 0.025
- Sampling rates: 0.001

When updating Word2Vec and FastText models, we changed the minimum word count to 0 for learning to accommodate learning with small-sized training sentences.

C.5 Nonce2Vec Models

We used Nonce2Vec models for the few-shot learning analysis. Compared to Word2Vec and FastText, Nonce2Vec had unique training process. We also tested more numbers of hyper-parameters for Nonce2Vec, since the model's results were much more sensitive to parameter settings.

C.5.1 Background Model

We used a regular Word2Vec model (with the same parameters) as a background model with both target and non-target sentences. Nonce2Vec model takes the nonce word (i.e., target word) as an input. The model simulates first-time exposure to the target word by changing the label of the existing target word's vector from the background model, and adds a newly initialized word vector for the target word. For example, if the target word is *insulin*, Nonce2Vec copies the target word's vector from the background model with a different label, like *insulin_gold*. Then the model randomly initialize the original target word's vector, and learns the new vector representation for the target word with a small number of target sentences Herbelot and Baroni (2017).

C.5.2 Nonce2Vec Results in Different Hyper-parameters

For updating Nonce2Vec models, we followed the settings from Herbelot and Baroni (2017), using 15 window words, 3 negative sampling words, 1 minimum word counts, and 10000 sampling rates. We also tested different learning rates {0.5, 1, 2}, and epochs {1, 5} for Nonce2Vec models.

Figures C.5 and C.6 shows Nonce2Vec models' results in different hyper-parameter settings. Higher epoch setting (e.g., $epoch = 5$) tended to show more stable results. The learning rate of 1.0 showed the best performance in median rank scores, but performed worse in Spearman's r scores.

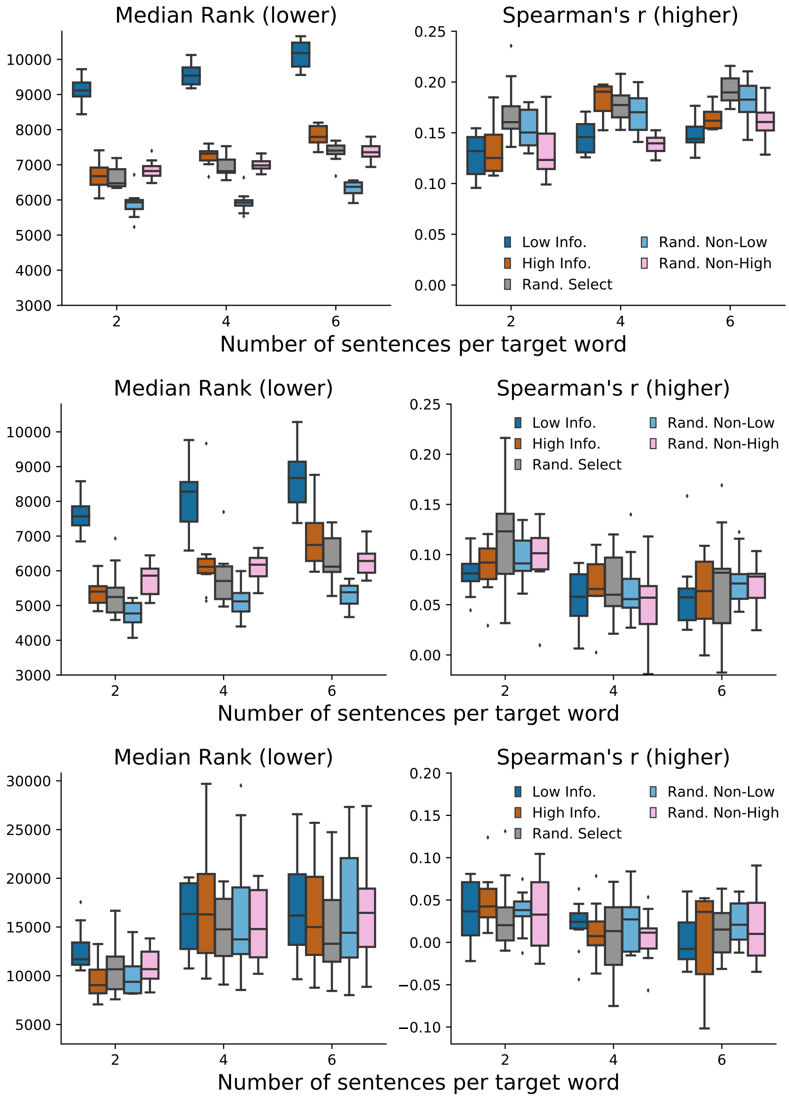


Figure C.5: Few-shot learning results (Nonce2Vec) on SimLex-999 dataset, with different learning gains $\{0.5, 1.0, 2.0\}$ and *epoch* $= 1$ settings.

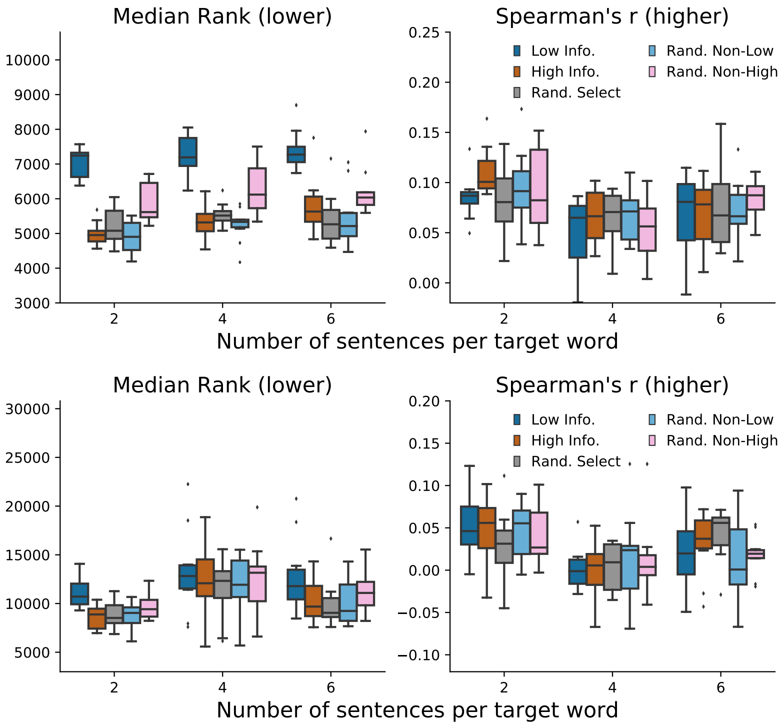


Figure C.6: Few-shot learning results (Nonce2Vec) on SimLex-999 dataset, with different learning gains {1.0, 2.0} and *epoch* = 5 settings. A higher epoch setting tended to show more stable results.