Analyzing the Causal Relation Between Linguistic Knowledge and the Performance of Language Models Using Structural Equation Modeling

by

Han Namgi

Dissertation

submitted to the Department of Informatics
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

The Graduate University for Advanced Studies, SOKENDAI
September 2021

A dissertation submitted to Department of Informatics,
School of Multidisciplinary Sciences,
The Graduate University for Advanced Studies, SOKENDAI,
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

Advisory Committee

1. Prof. Akiko AIZAWA	SOKENDAI, National Institute of Informatics, The University of Tokyo
2. Prof. Yusuke MIYAO	The University of Tokyo
3. Prof. Junichi YAMAGISHI	SOKENDAI, National Institute of Informatics
4. Prof. Jun SUZUKI	Tohoku University
5. Prof. Noriko KANDO	SOKENDAI, National Institute of Informatics

Acknowledgements

First of all, I should offer my best gratitude to my supervisor, Dr. Yusuke Miyao, for his patient guidance, continuous support, and heartful encouragement, and on and on. I can not express all his contribution to my research life, thesis, and academic experience here. However, I should appreciate him once more since I learned from him was literally everything I need to be a researcher.

I would like to express my deep gratitude to all members of my committee. Dr. Akiko Aizawa, Dr. Junichi Yamagishi, Dr. Jun Suzuki, and Dr. Noriko Kando. They have given me fruitful comments and insightful feedback throughout the whole of my doctoral course. Without a doubt, their suggestions and encouragements significantly have improved this thesis.

I conducted many parts of this study with Knowledge and Information Research Team in Artificial Intelligence Research Center, The National Institute of Advanced Industrial Science and Technology. I would like to thank all former and current Knowledge and Information Research Team members, especially Dr. Hiroya Takamura, Dr. Hiroshi Noji, Dr. Pascual Martinez-Gomez, and Mr. Goran Topic. I should note their advice and participation in this study with my sincere gratitude.

During my doctoral life, I owe much debt to all former and current mynlp lab members. I can not list everyone, but I am particularly grateful to Dr. Wai Lok Tam, Dr. Sho Hoshino, Dr. Nguyn Quý, Dr. Sang Phan, Dr. Fei Cheng, Dr. Katsuhiko Hayashi, Mr. Akira Miyazawa, Mr. Juan Ignacio Navarro Horñiacek, Mr. Takahiro Kondo, Mr. Takuto Asakura, Mr. Wenjie Zhong, and Ms. Nao Yoshida. Each discussion, conversation, and moment with them has helped me to do my best for this study.

At last, I would like to credit non-companions of my study but my precious family; my father Sang-Joon Han, my mother Myeong-Ok Cho, my brother Jung-Hoon Han, and my nephew I-Deun Han. Also, I would like to appreciate, if I can call him this way, my friend, Sang-Yong Sim. Their emotional supports have sustained me during my whole foreign academic life.

The Graduate University for Advanced Studies, SOKENDAI

Abstract

School of Multidisciplinary Sciences
Department of Informatics

Doctor of Philosophy

Analyzing the Causal Relation Between Linguistic Knowledge and the Performance of Language Models Using Structural Equation Modeling

by Han Namgi

Explaining the reason for the high performance of one system is as important as achieving high performance by using that system. Recently the language model, a vector representation of natural languages such as word2vec and BERT, has become an indispensable tool for natural language processing. While researchers have reported the state-of-the-art accuracy for a variety of downstream tasks by using language models, our understanding of this phenomenon usually depends on the observation for accuracy. However, the accuracy does not explain why one language model can obtain good accuracy and another can not. Furthermore, it is hard to find the reason for the good or bad performance of one language model for various downstream tasks from the accuracy. In other words, it indicates the lack of interpretability for language models.

Previous studies have tried to explain the quality of one language model in the aspect of encoded linguistic knowledge on that language model. However, their essential assumption, "encoded linguistic knowledge on one language model should affect the accuracy of the downstream task solved by that language model", has not been proved empirically and causally with enough samples. We present a novel framework employing the statistical method, Partial Least Squares Path Modeling (PLSPM), to explain the causal relationship between encoded linguistic knowledge and the accuracy of downstream tasks on the target language model. Our proposed framework starts from a causal diagram consisting of causal assumptions between variables, including encoded linguistic knowledge and the accuracy of downstream tasks. By validating whether the suggested causal diagram can produce similar covariance matrices with observed variables, we can examine our causal assumptions, for example, causal relationships between encoded linguistic knowledge and the accuracy of downstream tasks.

We present the usefulness of our proposed framework by following steps. First, we show that our PLSPM framework can prove the causal diagram consisting of traditional assumptions for encoded linguistic knowledge. In our PLSPM models, causal assumptions between encoded linguistic knowledge and accuracies for downstream tasks are expressed as linear regression equations. For fitting PLSPM models for our proposed causal diagrams, we prepare accuracies of one word analogy dataset measuring encoded linguistic knowledge and 20 downstream tasks solved by 600 word embedding models as observed variables. As a result, we find that our PLSPM models can prove most causal assumptions of our causal diagrams with a variety of reliability indexes for validating the estimated PLSPM model. Comparing to previous studies, our PLSPM models provide more informative explanations for accuracies of downstream tasks involving multiple linguistic knowledge and the effect of hyperparameters on language models.

In addition, we also apply our proposed framework to more complicated language models and downstream tasks to prove that our proposed framework is also helpful in the practical setting. We conduct another PLSPM analysis involving 24 BERT models, two probing tasks, and four datasets of simple factoid question answering (SFQA), a subtask of question answering over a knowledge base. Since this task requires external resources and a modularized structured system to be solved, we select SFQA as a more complicated and practical target downstream task. The BERT-based system achieves the upper bound accuracy of SimpleQuestions, the benchmark dataset of SFQA. However, our PLSPM framework reports that this system depends on the surface and syntactic information for solving simple factoid questions without understanding semantic information. It indicates the possibility that the upper bound accuracy of existing SFQA systems for SimpleQuestions may rely on the specific characteristic of the dataset itself.

We conduct an empirical analysis involving five SFQA systems, which have reported the upper bound accuracy of SimpleQuestions, and four SFQA datasets to examine whether those systems have the robustness and transferability for SFQA. We find that all existing SFQA systems can not reach upper bound accuracies for other datasets like SimpleQuestions, and they show significantly low accuracy when changing test data. According to our analysis, the size and the upper bound accuracy of each dataset do not cause this phenomenon. We reveal that existing SFQA systems report similar problems related to semantic understanding, such as disambiguation of the entity and paraphrasing of the relation. Moreover, we suggest that the source of each dataset and the evaluation method for SFQA make existing SFQA systems depend on surface and syntactic information with the additional analysis.

In this thesis, we proposed a novel statistical framework to explain the accuracy and inner working of language models as the causal relationship with encoded linguistic knowledge. We also proved that our proposed framework could provide valuable information for understanding and resolving the encountered issue of an existing NLP system. We hope that our study can suggest a systematical and practical way to interpret the inner working of language models.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

For a long time, researchers have tried to develop a system that can understand human language. This research field, Natural Language Processing (NLP), has been evaluated with NLP tasks, which consist of carefully designed questions to assess the linguistic abilities of proposed systems. If a proposed system can reach high accuracy for given questions of the target task, it means that a proposed system should have the linguistic ability required for the target task. For evaluation of the proposed system, understanding how the proposed system solves the target NLP task is essential, as well as reaching high accuracy for the target NLP task. In other words, the evaluation of the proposed system is related to examine whether the proposed system understands linguistic knowledge like human being. However, the explanation for the accuracy of the proposed system based on linguistic knowledge is not conducted sufficiently as much as its performance.

In recent years, word and contextual embeddings are the most well-known method for representing human language into low dimensional vector space in the NLP field. Because of its versatility for various NLP tasks and inspiration from original paper proposing word embeddings, researchers often call this method as *language modeling*. Previous papers reported that language modeling seems a successful translation from human language to vector representations showing interesting examples, such as understanding the relationship between words like $King - Man + Woman = Queen$ (Mikolov et al., 2013b), discovering syntactic structures within encoded sentences (Tenney et al., 2019a), and reaching state-of-the-art accuracies for a variety of NLP tasks (Devlin et al., 2019). However, previous papers did not consider to prove the causal relationship between the existence of a linguistic pattern and a high accuracy for the NLP task.

Therefore, it is still an unresolved problem to interpret the performance of language models in the aspect of encoded linguistic knowledge on language models.

This study suggests a novel framework to describe the causal relationship between encoded linguistic knowledge and the accuracy of downstream tasks on the target language model. Linguistic knowledge encoded on language models, especially vector space representations, is hard to be observed directly. Researchers have proposed a variety of tasks to evaluate encoded linguistic knowledge in language models (Baroni et al., 2014, Conneau and Kiela, 2018, Gladkova et al., 2016, Hill et al., 2015, Wang et al., 2019a). We explore how encoded linguistic knowledge can explain the accuracies of downstream tasks in a statistical way. In the rest of this chapter, we introduce a brief history of evaluating language models and our proposal to interpret language models.

1.1 Probing encoded linguistic knowledge on language models

What language model is a good language model? To answer this question, researchers usually have consulted with the accuracy of downstream NLP tasks using the language model they want to evaluate. Despite its simplicity and easiness of understanding, this evaluation does not explain why one language model can obtain higher accuracy than other language models. To overcome this limitation, researchers have depended on one natural intuition that a good language model should embed linguistic knowledge of human language (Baroni et al., 2014, Chiu et al., 2016, Gladkova and Drozd, 2016, Rogers et al., 2018). It precipitated proposals of the intrinsic evaluation (or sometimes called the probing task) (Conneau and Kiela, 2018, Gladkova et al., 2016, Hill et al., 2015, Wang et al., 2019a) designed for examining whether linguistic knowledge is encoded on the target language model or not. Thanks to previous studies which examine encoded linguistic knowledge involving various intrinsic evaluations, we have found that a variety of linguistic patterns exist on language models (Baroni et al., 2014, Lin et al., 2019, Liu et al., 2019a, Rogers et al., 2018, 2020, Tenney et al., 2019a,b).

This way is very similar to how human being evaluates the ability of language fluency. Like TOEIC or TOEFL, a famous language fluency test, researchers have evaluated basic linguistic abilities of target language models such as understanding lexical similarity (Hill et al., 2015), the relationship between words (Gladkova et al., 2016), and syntactic constituents of the given sentence (Conneau and Kiela, 2018). This evaluation is based on another natural assumption that encoded linguistic knowledge on one language model should affect accuracies of downstream tasks of NLP solved by that language model. This assumption can be translated for human beings that those who receive an appropriate education for linguistic knowledge achieve good accuracy for the language fluency test. If those assumptions are valid, we can conclude that one language model is good or bad with its evaluation results for intrinsic evaluations.

1.2 Necessity of statistical explanation for the causal relationship between linguistic knowledge and the accuracy

However, this evaluation method also has problems in examining the quality of a language model. First, previous studies often reported conflicting results even for the same probing task when previous studies take different methods or samples of the language model. For example, Htut et al. (2019) shows that the change of linguistic formalism, such as universal dependency (Schuster and Manning, 2016) and Stanford dependency (de Marneffe and Manning, 2008), affects the result of probing task on the same language model. Second, this evaluation still can not explain how encoded linguistic knowledge is used for solving given downstream tasks. In other words, the natural assumption mentioned above, "encoded linguistic knowledge on one language model should affect accuracies of downstream tasks of NLP solved by that language model", has not been tested statistically yet. Those limitations can cause a lack of robustness and generality for understanding the relationship between encoded linguistic knowledge and the accuracy of language models.

One reason for this phenomenon is how previous studies derive their conclusions from evaluation results. Since they usually depended on observation for one or a few samples to conclude, slight changes in experimental settings can cause the disagreement of interpretations among studies (Rogers et al., 2020). Some previous studies tried to investigate accuracy for intrinsic evaluations and accuracy for downstream tasks by simple correlation analysis (Chiu et al., 2016, Rogers et al., 2018, Wang et al., 2019b). However, correlation analysis is still not enough to explain how encoded linguistic knowledge is used for solving downstream tasks like a traditional quote "correlation does not imply causality" (Pearl, 2009). Therefore, it requires another approach to ensure the robustness of the interpretation and explain the causal relationships between encoded linguistic knowledge and the accuracy of downstream tasks.

We focus on that previous studies usually have relied on many assumptions about linguistic knowledge for language model. Researchers have proposed a variety of methods proving given assumptions statistically with observed variables in the statistical field. One of their proposed methods, Structural Equation Modeling (SEM) (Wright, 1921), tackles hypothesis test problems by verifying a causal diagram with given observed variables. This method examines whether the causal diagram, which consists of causal hypotheses between variables suggested by the researcher, can produce statistically acceptable covariance matrices comparing to observed variables. Our study employs SEM as a framework to figure out the inner working of language models involving encoded linguistic knowledge as the index of quality of language models. This framework explains which and how encoded linguistic knowledge causally affects the accuracy

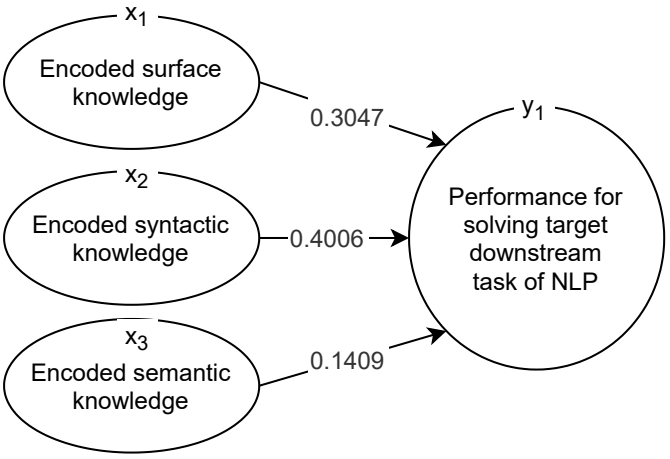

FIGURE 1.1: A sample of our proposed analysis on the target language model.

of the target downstream task with statistical verification.

1.3 How statistical explanation for language models can help to improve NLP applications?

Statistically explaining the accuracy of language models with linguistic knowledge is essential to evaluate what language model is good, as we mentioned above. Moreover, it also can provide valuable suggestions for improving existing systems designed for downstream tasks of NLP depending on a strong pretrained language model in recent days. Since a recent language model, which consists of complex neural network based structures, is hard to interpret the inner working, many studies have been conducted to investigate how the language model is working for target downstream tasks (Rogers et al., 2020). They usually have tackled this problem from the engineering approach, such as assessing the effect of hyperparameters (Turc et al., 2019) or an observation on each layer (Tenney et al., 2019a,b). On the contrary, we explain the accuracy of the target language model by how encoded linguistic knowledge works for a given downstream task.

Because of statistical verifications conducted in the SEM framework, we can report the effectiveness of encoded linguistic knowledge on the evaluated language model for the target downstream task as the concrete coefficient value. Figure 1.1 shows one sample of our proposed analysis. In this sample case, we can explain that the accuracy of the target downstream task, y_1, would be predicted by the structural equation, $0.3047 * x_1 + 0.4006 * x_2 + 0.1409 * x_3$. This explanation has a variety of advantages comparing to previous studies for the interpretability of a language model. First of all, it is intuitive and easy to understand even for the people who are not the expert of the NLP field. Since our explanation depends on the concept of linguistic knowledge evaluated by intrinsic evaluations, it does not require any prior knowledge for machine learning or neural network engineering. Furthermore, this explanation also provides clues to find limitations or

unintended behaviors of the evaluated target language model. For example, the lower coefficient of x_3 in Figure 1.1 indicates that the evaluated language model may not depend on encoded semantic knowledge well for solving the target downstream task. In this case, a researcher aiming to improve this language model would start to investigate the issue related to encoding semantic knowledge on the language model.

1.4 Outline of this study

This study suggests a novel framework for evaluating and explaining the inner working of language models, which means the causal relationship between encoded linguistic knowledge and the accuracy of language models in this study, employing a statistical method, SEM. Also, we present a concrete application of our suggested framework for an existing downstream task as a case study. We select simple factoid question answering (SFQA), a subtask of question answering over a knowledge base, with the following reasons. First, researchers have reported the upper bound accuracy of the benchmark dataset in this task, SimpleQuestions (Huang et al., 2019, Lukovnikov et al., 2019, Mohammed et al., 2018, Petrochuk and Zettlemoyer, 2018, Yu et al., 2017). However, the robustness and transferability of their proposed systems have not been examined with other datasets. Second, we have interested in proving that we can apply our proposed framework to a practical downstream task requiring external resources and a complicated structure. Since existing systems for SFQA (Huang et al., 2019, Lukovnikov et al., 2019, Mohammed et al., 2018, Petrochuk and Zettlemoyer, 2018, Yu et al., 2017) consists of modularized structures and depends on external knowledge bases, this task is suitable for our purpose.

To sum up, we should address the following problems in this study.

- Designing a suitable causal diagram representing causal relationships between encoded linguistic knowledge and the performance for downstream tasks.
- Validating whether our proposed framework can produce reasonable and acceptable results compared with previous studies.
- Applying our proposed framework into a concrete downstream task, SFQA, for understanding and verifying the inner working of existing systems.
- Proving that the result of our proposed framework is helpful to understand and resolve problems at hand for SFQA.

In the rest of this study, we address those problems in order.

- In Chapter 2, we explain fundamental and background knowledge of our proposed PLSPM framework and SFQA. This chapter includes a brief introduction of language models, SEM, question answering over a knowledge base (QAKB), SFQA, and the evaluation of language models and SFQA systems.
- In Chapter 3, we present a novel framework employing Partial Least Squares Path Modeling (PLSPM), one method of SEM, to explain the inner working of word embedding models word2vec and fasttext. We suggest causal diagrams following traditional assumptions for linguistic knowledge encoded in language models suggested by previous studies. We then prove that our proposed framework reports comparable results with previous studies, including a novel finding of the structural problems.
- In Chapter 4, we apply our proposed PLSPM framework to investigate and understand the inner working of the BERT-based system, BertQA, for SFQA. Our evaluation results reveal that the existing system solves simple factoid questions mainly depending on surface and syntactic knowledge. We also explain how this phenomenon is related to problems at the hand of SFQA and why this phenomenon appears by additional analysis.
- In Chapter 5, we examines whether other SFQA systems can succeed in solving simple factoid questions generally, which BertQA can not. Despite the success of the benchmark dataset for SFQA, the robustness and transferability of existing systems for this task have not been examined yet. We conduct an empirical analysis for this issue involving four datasets and five existing SFQA systems. As a result, we reveal that existing systems do not show the robustness and transferability for simple factoid questions outside of SimpleQuestions. We also discuss this phenomenon in the aspect of linguistic knowledge considering the PLSPM analysis of the previous chapter.
- In Chapter 6, we give conclusions and future works of our study.

Chapter 2

Fundamentals and related work

This study involves the evaluation of the language model, statistical methods for hypothesis testing, and Question Answering over a Knowledge Base (QAKB). Hence we present an overview of those research fields in this chapter as fundamentals of this study. First, we suggest a brief history of language modeling in Chapter 2.1. In the next part, we introduce statistical methods for hypothesis testing in Chapter 2.2 since we address the evaluation of language modelings as the hypothesis testing problem. We introduce simple factoid question answering (SFQA) in Chapter 2.3, which is our target downstream task to be analyzed in this study. We explain this task in the order of an introduction of a knowledge base, a brief review for QAKB and its datasets, and a survey of a more concrete subtask of QAKB, SFQA. Finally, we move to Chapter 2.4 explaining how language models have been evaluated, which is a core part of this study. We also introduce previous studies for SFQA related to the evaluation and the source of existing SFQA datasets.

2.1 Language modeling

In the natural language processing field, representing an utterance for the machine learning system is an important issue. One popular and successful method is to represent words and documents as vectors. This approach have been called by several names, such as *vector space model* (Salton et al., 1975), *distributed representations* (Hinton et al., 1986), *neural network language models* (Bengio et al., 2003), *word representations* (Turian et al., 2010), *vector space*

representation (Turney and Pantel, 2010), *word embeddings* (Levy and Goldberg, 2014), and *contextual embeddings* (Liu et al., 2020). We use the name *language modeling* for mentioning this method generally in our paper, referring to Bengio et al. (2003) which suggested the modern concept and task definition of this approach. In this chapter, we present a brief history and survey of language models.

2.1.1 Word embeddings

Early ideas that encoding words to vectors had developed many representations such as one-hot vector and n-grams. However, the turning point of language modelings for NLP was suggested by Bengio et al. (2003). The primary motivation of distributed representation is the curse of dimensionality. Since earlier representations need the same length as the number of vocabulary in the training corpus, those representations tend to become much longer sequences. Furthermore, the length of earlier representations usually changes if the training corpus changes. Bengio et al. (2003) tackled this problem by suggesting a new task, called language modeling, to train a fixed-length vector of each word for a probability of semantically neighboring. Their idea was based on *distributional hypothesis* (Harris, 1954), "words that occur in the same contexts tend to have similar meanings." If the trained distributed representations can predict context terms for a given input term, it indicates that those representations contain semantic understanding for the vocabulary in the training corpus.

Many researchers had proposed advanced versions of language modelings for words after Bengio et al. (2003). During this period, Turian et al. (2010) reported that language modelings for words could help to increase the downstream tasks of NLP, such as POS tagging and chunking. It became one motivation to encourage researchers to participate in this field. Finally, Word2vec (Mikolov et al., 2013b), which is the most popular word embedding model in this field, was proposed. Mikolov et al. (2013b) suggested the skipgram model for training word embeddings that exchanges the input data and output data of language modeling task. They proved that the skipgram model could dramatically decrease training time and machine cost than the previous study (Bengio et al., 2003), without any loss in the performance of the trained distributed representations. In addition, Mikolov et al. (2013b) also reported that word2vec could learn syntactic and semantic relationships like $King - Man + Woman = Queen$. It is the reason why various downstream tasks of NLP started to employ language models as pretrained standalone features.

We introduce the formal definition of the skipgram model here. Figure 2.1 shows the overview of the skipgram model. The skipgram model produces word embeddings for each term by training the objective function to predict context terms of the given input term. Mikolov et al. (2013b)

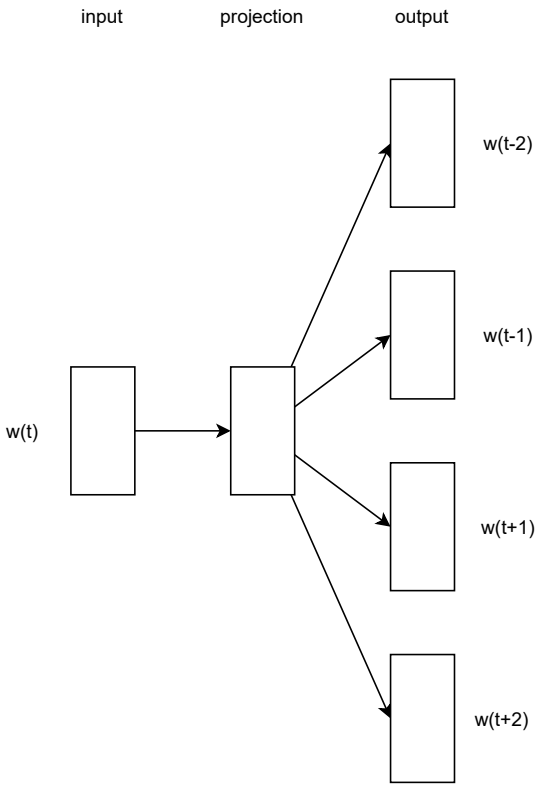

FIGURE 2.1: The architecture of the skipgram model suggested by Mikolov et al. (2013b). With the given term $w(t)$, the skipgram model predicts context terms of $w(t)$. In this figure, the number of context t is 5.

suggested that the objective function for training the skipgram model is to maximize the average log probability of the following equation:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c, j\neq 0}\log p(w_{t+j}|w_t) \tag{2.1}$$

where a sequence of given words is $w_1, w_2, w_3, ...w_t$, and c is the size of the context window as a hyperparameter. In the original definition of skipgram, $p(w_{t+j}|w_t)$ is defined like the following equation:

$$p(w_O|w_I) = \frac{\exp(v'^{\top}_{w_O} v_{w_I})}{\sum_{w=1}^{W}\exp(v'^{\top}_{w} v_{w_I})} \tag{2.2}$$

where W is the number of terms in the training corpus, v_w is the input word embedding, and v'_w is the output word embedding. Since this equation causes a computing cost in proportion to W, Mikolov et al. (2013b) employed other approaches including hierarchical softmax (Morin and Bengio, 2005) and negative sampling (Gutmann and Hyvärinen, 2012). Hierarchical softmax defines $p(w_{t+j}|w_t)$ as the following equation:

$$p(w|w_I) = \prod_{j=1}^{L(w)-1} \sigma([[n(w, j+1) = ch(n(w, j))]] \cdot v'^{\top}_{n(w,j)} v_{w_I}) \tag{2.3}$$

with below conditions.

- The output layer for all W is represented as a binary tree.
- $n(w, j)$ is the j-th node which we go through during the way from the root to the node for w.
- $L(w)$ is the length of the path from the root to the node for w.
- $ch(n)$ is an arbitrary child node of n.
- $[[x]]$ is a binary function which becomes 1 if x is true and -1 if x is false.

Negative sampling defines $p(w_{t+j}|w_t)$ as the following equation:

$$\log \sigma(v'^{\top}_{w_O} v_{w_I}) + \sum_{i=1}^{k} \mathbb{E}_{w_i \sim P_n(w)} [\log \sigma(-v'^{\top}_{w_i} v_{w_I})] \tag{2.4}$$

where $P_n(w)$ is the noise distribution and k is the number of negative samples.

Bojanowski et al. (2017) extended the skipgram model considering n-grams of each term. The fastText model proposed by Bojanowski et al. (2017) considers each term as a bag of n-grams of that term. Where G is the number of given a bag of n-grams, they define that $G_w \subset \{1, ..., G\}$ is the bag of n-grams for the given term w. The fastText model trains a vector representation z_g for each n-gram g like the skipgram model. However, a vector representation for a term w is represented as $\sum_{g=1}^{G} z_g$ in the fastText model. Therefore, we can write the scoring function in the fastText model below:

$$s(w, c) = \sum_{g \in G_w} z_g^{\top} v_c. \tag{2.5}$$

Table 2.1 shows a summary of word embedding models.

2.1.2 Contextual embeddings

While language models for words have been an indispensable tool for various downstream tasks of NLP, they have lots of limitations. Those limitations are mainly caused by assigning one

Model	length	Descriptions
One-hot vector	indefinite	a binary vector which index represent each word in the given document
Distributed Representation (Bengio et al., 2003)	fixed	the output of the model which predict the next word for a given sequence
word2vec (Mikolov et al., 2013b)	fixed	the output of the model which predict contextual words for a given word
fastText (Bojanowski et al., 2017)	fixed	the output of the model which is an extended word2vec model using a bag of character n-grams

TABLE 2.1: List of popular word embedding models. Length means the length of a vector representation for each term.

vector representation for each word. For example, the word "book" is used as different meanings in "I read my book" and "I booked this room". Since word embedding models only assign one vector representation for "book", it is hard to distinguish between two different meanings with only one vector representation. Moreover, it is also hard to handle compound terms, including the case of a named entity. It is the reason why new models considering the context information of given words have been proposed recently.

The early idea of contextual embeddings can be found from Dai and Le (2015) and Ramachandran et al. (2017) since they employ a sequence encoder into language modeling. Though they did not aim to develop embeddings themselves, their idea affects a new model, ELMo (Peters et al., 2018). ELMo employed a bidirectional LSTM encoder to extract context and dependency information from given sentences. Another advantage of ELMo is that it is adequate to involve a pre-trained ELMo model into a neural network based system for downstream tasks of NLP. Peters et al. (2018) proved it by applying ELMo into their baseline models with six downstream tasks of NLP and reporting state-of-the-art performances. GPT (Radford et al., 2018, 2019) is another important contextual embeddings, since it employ transformers model (Vaswani et al., 2017) into language modeling task. Transformers have the advantage of learning context information for given sentences than previous models such as LSTM. Radford et al. (2018) reported the state-of-the-art performance with nine downstream tasks of NLP when applying the GPT model.

In recent days, BERT (Devlin et al., 2019) is the most popular contextual embedding. Here we introduce details of BERT, which is the most representative model for contextual embedding. BERT employs bi-directional transformers encoders (Vaswani et al., 2017), comparing with GPT, which only uses a left-to-right encoder. Figure 2.2 shows an overview of the transformer model. The transformer model consists of stacked self-attention and position-wise feed-forward layers for both encoder and decoder stacks. The encoder of the transformer model converts a given input sequence to a continuous vector representation. With the result of encoder stacks, the decoder makes an output sequence. This process occurs auto-regressively when applying the output of decoder stacks to an additional input sequence for the next step.

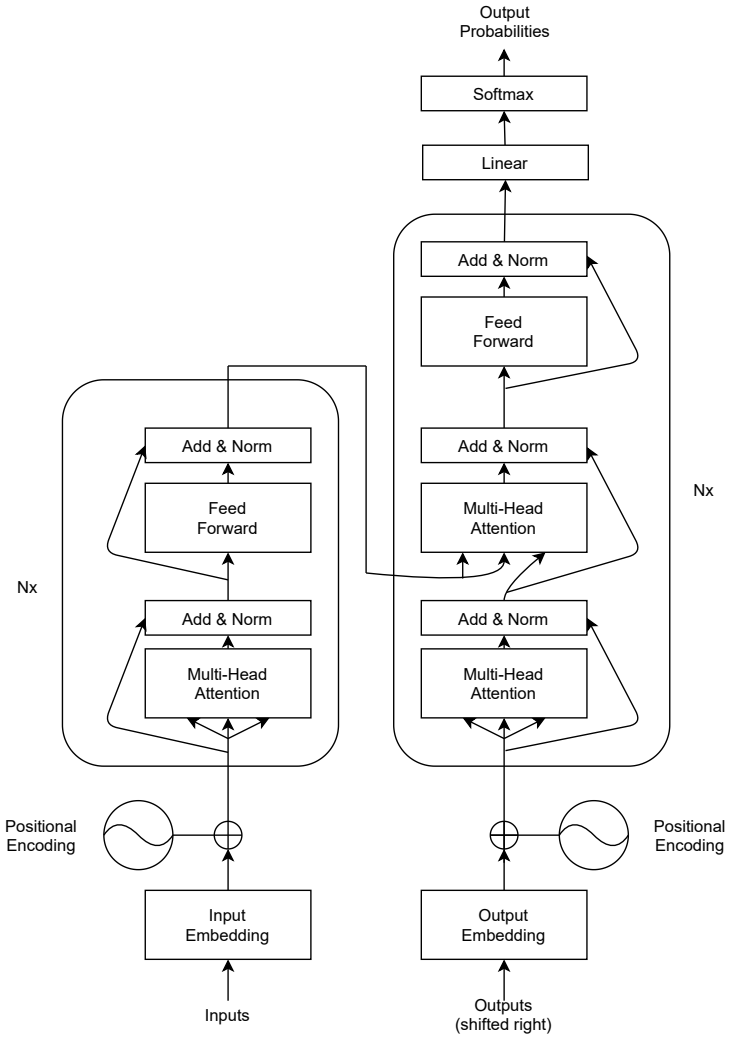

FIGURE 2.2: The architecture of the transformer model suggested by Vaswani et al. (2017). The left part represents encoder stacks, and the right part represents decoder stacks. N means the number of layer.

Since the transformer model does not consider a recurrence or convolution network, Vaswani et al. (2017) suggested positional encodings to encode information for the position of each token. Positional encodings, PE, are calculated with the following equations:

$$\begin{aligned} PE_{(pos,2i)} &= \sin(pos/10000^{2i/d_{model}}) \\ PE_{(pos,2i+1)} &= \cos(pos/10000^{2i/d_{model}}) \end{aligned} \tag{2.6}$$

where pos is the position, i is the dimension, and d_{model} is the size of the dimension for the given $model$.

In the transformer model, multi-head attention is designed for a self-attention function that mapping an input query and a set of key-value pairs to the output of the attention part. Note that each query q, key k, value v, and output o is a vector representation here. The attention

part of the transformer model, called scaled dot-product attention, computes the output with the following equation:

$$Attention(Q, K, V) = \text{softmax}(\frac{QK^\top}{\sqrt{d_k}})V \tag{2.7}$$

where Q, K, V are packed matrices from a set of query, key, and value, respectively. A computed output becomes one head, and those heads are jointed with the following equation:

$$\begin{aligned} MultiHead(Q, K, V) &= Concat(head_1, ..., head_h)W^O \\ where\ head_i &= Attention(QW_i^Q, KW_i^K, VW_I^V) \end{aligned} \tag{2.8}$$

where we denote $W_i^Q \in \mathbb{R}^d_{model} \times d_k$, $W_i^K \in \mathbb{R}^d_{model} \times d_k$, $W_i^V \in \mathbb{R}^d_{model} \times d_v$, and $W_i^O \in \mathbb{R}^h d_v \times d_{model}$.

The result of multi-head attention becomes the input of position-wise feed-forward layers. It consists of two fully connected layers with a ReLU activation. Therefore, the position-wise feed-forward layer can be defined like below:

$$FFN(x) = \max(0, xW_1 + b_1)W_2 + b_2 \tag{2.9}$$

The main advantage of the transformer model is allowing the system to encode information for global dependencies of input and output sequences. Therefore, BERT, the language model employing bi-directional transformers encoders, can understand context information more correctly than previous language models.

Moreover, BERT suggested a new objective task for training, masked language modeling. Since masked language modeling demands to predict randomly masked tokens in a given sentence, it requires more complex information for context and dependency than previous language modeling. Furthermore, BERT also proposed another objective task, next sentence prediction. They aimed to train the ability to understand the relationship between sentences. Thanks to the above improvements, Devlin et al. (2019) reported that applying BERT is effective for eleven downstream tasks of NLP with state-of-the-art performance. In this way, contextual embeddings such as BERT have become an indispensable tool for the natural language processing field in recent years.

2.2 Statistical method for causal analysis

In the NLP field, correlation analysis has been usually used to investigate the relationship between two variables, such as the accuracy of two NLP tasks. However, correlation analysis has limitations in proving the causal relation between given variables. At first, like traditional quote *correlation does not imply causality* (Koller and Friedman, 2009, Pearl, 2009), the result of correlation analysis can not indicate the causal relationship between variables. Secondly, correlation analysis of previous studies, which considers the relationship between variables as one by one relationship, does not consider external factors for calculating the correlation coefficient between variables. Therefore, sometimes previous studies conducting correlation analysis reported conflicting results for the same problem (Chiu et al., 2016, Rogers et al., 2018, Schnabel et al., 2015, Wang et al., 2019b). This chapter introduces another statistical way for testing hypotheses, Structural Equation Modeling (SEM) (Wright, 1921). Moreover, we also provide an introduction of partial least squares path modeling (PLSPM) (Wold, 1982), the method we employ in this study, which is a widely accepted SEM method across the social science field.

2.2.1 Structural Equation Modeling

Structural Equation Modeling (SEM) was first proposed to model and investigate complex relationships between variables by Wright (1921). Compared with correlation analysis, SEM suggests the causal diagram, which consists of causal hypotheses among variables by the researcher. In the causal diagram, each relationship is expressed as the structural equation. For example, a structural equation is $variable_1 = x_1 * variable_2 + x_2 * variable_3 + y$, then conditional changes of $variable_1$ can be estimated following this equation. It is hard to mathematically prove the causal relationship with only an equation Pearl (2009). SEM tackles this problem by suggesting the causal diagram and modeling causal assumptions from observed variables. Based on prior and theoretical knowledge, the causal assumption can help an interpretation of the structural equation as the causal relationship. Moreover, SEM aims to estimate a model consisting of statistically fitted structural equations from observed variables and validate an estimated model with statistical tests. For example, if an estimated model can produce similar population covariance matrices with covariance matrices based on observed variables, the researcher can accept suggested causal hypotheses within the proposed causal diagram and observed variables.

Figure 2.3 shows a sample of the causal diagram. It expresses the relationship between encoded semantic information and the performance for the QAKB task on the same contextual embedding. Note that encoded semantic information and the performance for the QAKB task are not directly measured since they are abstract concepts. We call them as *latent variables*, as opposed to *observed variables*. Each x_{1n} is the result of tasks for evaluating encoded semantic information on contextual embeddings, and each y_{1n} is the result of subtasks on the QAKB. It is a natural

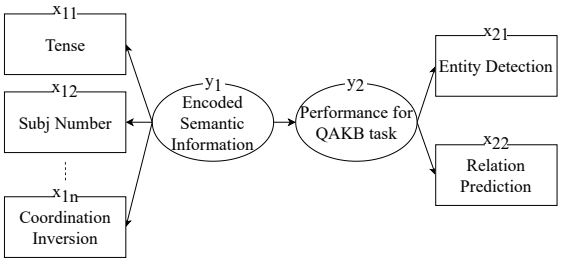

FIGURE 2.3: A sample of the causal diagram. circles represent latent variables, rectangles represent observed variables, and edge arrows represent causal relationships between variables.

assumption that encoded semantic information on the contextual embedding affects each x_{1n} of the same contextual embedding, thus by utilizing each x_{1n} we can estimate a score for y_1, a latent variable for encoded semantic information. The same is also true of the case between the performance for the QAKB task and its subtasks. In Figure 2.3, edge arrows between latent variables and observed variables represent causal assumptions. The edge arrow between latent variables represents another causal assumption that encoded semantic information on a contextual embedding affects the performance of the QAKB task. Therefore, if we prepare each x_{1n}, y_{1n} data as observed variables, SEM tries to prove that a given causal diagram can explain covariance matrices produced by observed variables.

In general, structural equation modeling is separated into two submodels: (1) the *measurement model* has relationships between the observed and latent variables, while (2) the *structural model* consists of the relationships between latent variables. Any causal relationship can be expressed by a linear regression equation, also called a *structural equation*. The measurement model for the diagram in Figure 2.3 thus consists of the following equations:

$$\begin{aligned} x_{11} &= \lambda_{11} y_1 + \varepsilon_{11} \quad & x_{21} &= \lambda_{21} y_2 + \varepsilon_{21} \\ x_{12} &= \lambda_{12} y_1 + \varepsilon_{12} \quad & x_{22} &= \lambda_{22} y_2 + \varepsilon_{22} \\ &\vdots \\ x_{1n} &= \lambda_{1n} y_1 + \varepsilon_{1n} \end{aligned} \tag{2.10}$$

where the x are observed variables, the y are latent variables, the λ denote weights for each factor, and the ε represent error terms. The structural model also has the following linear equation:

$$y_2 = \beta_{11} y_1 + \zeta_1 \tag{2.11}$$

where β is a weight and ζ is an error term. Given a causal diagram and the values of observed variables as input, we need to fit parameters, such as weights and error terms, of multiple regression equations and latent variables to the input data. After fitting the model, we can interpret the strength of a causal relation from the path coefficient and decide whether to accept a tested hypothesis appropriately according to how well it fits the data.

An SEM model includes parameters to be estimated, such as variances of latent variables and path coefficients of structural equations. In the statistic field, a variety of methods have been proposed depending on how to estimate those parameters and produce covariance matrices. One popular method is fitting parameters for maximum likelihood estimation with given observed variables (Jöreskog, 1970). However, this method requires heavy distributional assumptions and large sample sizes for observed variables. Especially, experimental data, such as accuracies of downstream tasks, usually do not follow normal distributions. Therefore, we introduce another SEM method, PLSPM, which has fewer requirements for observed variables.

2.2.2 Partial least squares path modeling

Another approach for fitting the model in structural equation modeling is partial least square path modeling (PLSPM), proposed by Wold (1982). It is often called a component-based approach because it estimates the scores of latent variables from linear combinations of observed variables. PLSPM does not require strict assumptions for observed variables, such as normal distribution and independence (Tenenhaus et al., 2005). Because of the relaxed requirements for observed variables, PLSPM has been accepted in various social science disciplines as a helpful tool for exploratory research (Henseler et al., 2014).

Here, we explain the details of the algorithm for the PLSPM estimation procedure, following Tenenhaus et al. (2005) and Sanchez (2013). To estimate parameters, PLSPM first aims to calculate the scores of latent variables. The scores of the latent variables in Figure 2.3 are thus written as below.

$$y_j = \sum_k \omega_{jk} x_{jk} + \sigma_{jk} \tag{2.12}$$

Note that PLSPM does not use or estimate any λ and β before the estimation of y finishes. Because the x is already given as observed variables, we need to estimate the parameter ω. PLSPM thus conducts an iterative procedure for updating ω. First, it initializes all ω to an arbitrary number that allows the calculated scores of the latent variables to have unit variance. For example, if all ω are initialized as 1, then all latent variables in Equation 2.12 can be estimated as sums of observed variables, as below.

$$\begin{aligned} y_1 &= \textstyle\sum_k x_{1k} + \sigma_{1k} \\ y_2 &= \textstyle\sum_k x_{2k} + \sigma_{2k} \end{aligned} \tag{2.13}$$

In the next step, PLSPM tries to obtain the weights in the structural model, e.g., β_{11} in Equation 2.11. Note that we do not use the weights of the measurement model, ω, in this step.

Instead, β is only estimated from the scores of latent variables by using correlation coefficients between adjacent latent variables. PLSPM has various options for how to obtain the weights in the structural model. For example, the centroid scheme option receives β by the following formula:

$$\beta_{ji} = \begin{cases} sign[cor(y_j, y_i)] & \text{if } y_j, y_i \text{ are adjacent} \\ 0 & \text{otherwise} \end{cases}, \tag{2.14}$$

where $sign[a]$ is the sign direction of a, taking a value of ± 1, and $\text{cor}(a, b)$ is the correlation coefficient between a and b. With the obtained β, PLSPM estimates other scores for the latent variables, y', as below:

$$y'_j = \sum_{i \leftrightarrow j} \beta_i y_i + \zeta_i \tag{2.15}$$

where $\leftrightarrow$ means that y_i and y_j are connected in the structural model. With these new scores for the latent variables, y', PLSPM can update the weights of the measurement model, ω. In general, it calculates ω as a coefficient of ordinary least squares regression on ω and y'. The estimation formula for ω depends on which variables are the cause; for example, when Equation 2.12 is given, ω will be estimated as below.

$$\omega_{jk} = ({y'}_j^\top {y'}_j)^{-1} {y'}_j^\top x_{jk} \tag{2.16}$$

PLSPM then continues the above procedures until ω convergences, usually via $|\omega_{jk}^{(e-1)} - \omega_{jk}^{(e)} < 10^{-5}|$, where e is an epoch number. When the iterative process is complete, PLSPM has already finished estimating all weights and the scores of the latent variables. Therefore, it can estimate the path coefficients in the structural model and correlation coefficients in the measurement model, which indicate the prediction strength of each path in the PLSPM model. Here, *path coefficients* in the PLSPM model are estimated by ordinary least squares regression.

$$\text{Path coefficient}_{ji} = (y_i^\top y_i)^{-1} y_i^\top y_j \tag{2.17}$$

A *loading* is usually calculated as the correlation coefficient between an observed variable and a latent variable. During the estimation of the path coefficients and loadings, the weights in the measurement model, λ, and the weights in the structural model, β, are also fitted at once. Therefore, this is the end of the PLSPM fitting process. We summary the fitting algorithm for PLSPM in the following Algorithm 1.

Algorithm 1 PLSPM estimation algorithm

Input: $X = [X_1, \ldots, X_j, \ldots, X_J]$
Output: $y_j, \omega_j, \sigma_j, \beta_j, \zeta_j$
1: **for** $j = 1, \ldots, J$ **do**
2: **Initialize:** ω_j
3: $y_j \propto \pm \sum_{k=1}^{K_j} \omega_{jk} x_{jk} + \sigma_{jk} = \pm X_j \omega_j + \sigma_j$
4: $\beta_{ji} = sign[cor(y_j, y_i)]$
5: $y'_j = \sum_{i \leftrightarrow j} \beta_i y_i + \zeta_i$
6: **Update:** $\omega_{jk} = ({y'}_j{}^\top y'_j)^{-1} {y'}_j{}^\top x_{jk}$
7: **end for**
8: Repeat 1-7 until the convergence on the ω is achieved
9: Upon the convergence:
Path coefficient$_{ji} = (y_i^\top y_i)^{-1} y_i^\top y_j$

2.2.3 Validation of an estimated PLSPM model

To assess a PLSPM result whether it is reliable and reasonable to explain observed variables, researchers use a variety of reliability indexes for the measurement model and the structural model respectively. In this chapter, we explain reliability indexes commonly used in the statistic discipline for assessing the PLSPM model.

First, the design of the measurement model can be examined with *Cronbach's* α (Cronbach, 1951) and *Dillon–Goldstein's* ρ (Dillon and Goldstein, 1984) for internal consistency. The purpose of employing Cronbach's α and Dillon–Goldstein's ρ is to examine whether observed variables belonging to the same latent variable have a significant mutual association in the measurement model (Sanchez, 2013). Cronbach's α can be interpreted as an average value of inter-variable correlation coefficients. When m is the number of observed variables in the target block and $\mathrm{cor}(x, y)$ is the correlation efficient between x and y variables, Cronbach's α is calculated as below.

$$\alpha = \frac{\sum_{p \neq p'} \mathrm{cor}\left(x_p, x_{p'}\right)}{m + \sum_{p \neq p'} \mathrm{cor}\left(x_p, x_{p'}\right)} \times \frac{m}{m-1}. \tag{2.18}$$

Dillon–Goldstein's ρ is used to examine the composite reliability of the measurement model. When λ_p is the correlation between a latent variable and p-th observed variable, it is estimated as below.

$$\rho = \frac{\left(\sum_{p=1}^{m} \lambda_p\right)^2}{\left(\sum_{p=1}^{m} \lambda_p\right)^2 + \sum_{p=1}^{m} \left(1 - \lambda_p^2\right)}. \tag{2.19}$$

Generally, the value of both metrics should be larger than 0.7 for the unidimensionality of the proposed measurement model. In other words, Cronbach's α and Dillon–Goldstein's ρ over 0.7 indicate that validated observed variables, which are linked with the same latent variable, can be represented in one-dimensional space for the latent variable.

The validation for the structural model depends on structural equations among latent variables. Since these equations are estimated by ordinary least squares regression, we can validate each equation with p values and coefficients. The path representing a solid causal relationship is acceptable by $p < 0.05$ and has a significantly high path coefficient. Moreover, the determination coefficient R^2 and *Goodness-of-Fit* (GoF) are usually employed to assess the quality of the structural model. R^2 is defined as the proportion of how many independent variables can predict the variance of dependent variables. It is calculated following the below equation:

$$R^2 = 1 - \frac{\sum_i (y_i - \bar{y})^2}{\sum_i (y_i - f_i)^2} \tag{2.20}$$

where f is a predicted value, and $\bar{y}$ is the mean of y.

In evaluating a PLSPM model, we define GoF as the geometric mean of the average both for the squared loading and R^2. GoF indicates the whole explainability of a fitted PLSPM model considering both the measurement model and the structural model. This evaluation method is similar to other multiple regression analysis methods. In particular, a latent variable with $R^2 > 0.6$ is considered highly explained, and a PLSPM model is deemed to be strong when it achieves a GoF value over 0.7 (Sanchez, 2013).

The result of validations for an estimated PLSPM model also indicates the necessity to revise the suggested causal diagram sometimes. For example, if the Cronbach's alpha value for one latent variable is lower than 0.7, the researcher should suspect that observed variables for that latent variable do not measure the same thing. In this case, loadings can provide a solid clue to find which observed variable is not suitable for presenting target latent variable. Also, the researcher should consider a revision of the structural model when most structural equations are rejected by $p < 0.05$ or the GoF value is too low. Therefore, we can utilize the validation of an estimated PLSPM model to understand correct relationships among variables in the causal diagram suggested by the researcher.

2.3 Question answering over a knowledge base

Question answering over a knowledge base is one important NLP task that has been researched for a long time. This task can be said one particular type of question answering which depends on external knowledge bases, such as DBpedia (Lehmann et al., 2015) and Freebase (Bollacker et al.,

Name	Entities	Relations	Triples
DBpedia	4,806,150	2,813	176,043,129
YAGO	4,595,906	77	25,946,870
Freebase	49,947,845	37,781	3,041,722,635
Wikidata	15,602,060	1,673	65,993,797
OpenCyc	118,499	18,526	2,413,894
Google's Knowledge Graph	570,000,000	35,000	18,000,000,000

TABLE 2.2: The statistics for popular knowledge bases (Paulheim, 2017).

2008). Since a knowledge base is organized following a predefined formal structure, question answering over a knowledge base demands a different approach from other question answering tasks. In this chapter, we start from a brief introduction of a knowledge base for an introduction of question answering over a knowledge base. Moreover, we also introduce SFQA, which is our main target task in this paper. While many systems have reported the upper bound accuracy of the benchmark dataset for this task, the robustness and transferability of those systems are not examined. We are interested that existing systems can solve general simple factoid questions and how they can solve them.

2.3.1 Knowledge base

A knowledge base is a set of structured knowledge representing facts. While this concept was suggested for the expert system initially, it is widely used for various NLP tasks, such as data integration, named entity recognition, topic detection, and document ranking (Lehmann et al., 2014). The representative knowledge base is a knowledge graph, which stores knowledge with a graphical structure. A node means the entity in the real world in a knowledge graph, and an edge implies the relation between entities in a knowledge graph. It is formally defined as $KG = (E, P)$ for a knowledge graph KG, where E is a set of entities and P is a set of predicates.

A fact in the knowledge graph can be expressed as the Resource Description Framework (RDF) format, which is proposed by W3C (Brickley et al., 1999). In RDF format, a fact is represented as a triple consisting of a subject, a predicate, and an object. Both a subject and an object can be regarded as the entity, the node in a knowledge graph. A predicate can be regarded as the relation, the edge in a knowledge graph. In other words, a knowledge graph can be represented as a set of triples for facts. Where E is the set of entities $\{e_1, e_2, ...e_n\}$ and P is the set of predicates $\{p_1, p_2, ...p_m\}$, we can formally define a knowledge base K as a set of triples $(e_1, p_1, e_2) \in E \times P \times E$. For example, a RDF triple, (`Michael_Jordan`, `people.person.nationality`, `United_States`) can be written in a plain sentence "The nationality of Michael Jordan is United States".

The problem is that popular knowledge bases, such as Freebase, DBpedia, and Wikidata, contain a massive size of knowledge triples. It makes the cost and difficulty of the search for a concrete entity or relation high. Table 2.2 shows the statistics of popular knowledge bases.

While structured query language (SQL) and relational database are representative methods to manage structured data, they can not be applied to graphical data such as RDF. It is why SPARQL Protocol and RDF Query Language (SPARQL), a semantic query language, are introduced for retrieving and managing triples. SPARQL, proposed by W3C, allows humans to make queries for searching the specific triple automatically. According to the official document of SPARQL (Consortium et al., 2013), a simple SPARQL query consists of two parts; `SELECT` and `WHERE`. A `SELECT` part shows what variable will appear in the result of the given query. A `WHERE` part includes a pattern of triple what the user wants to match with a knowledge base. For example, if the given SPARQL query is like below, this query will print `United_States`.

```
SELECT ?nation WHERE { Michael_Jordan people.person.nationality ?nation . }
```
(2.21)

If a user executes Equation 2.21, then the system finds a node that is linked to the node `Michael_Jordan` with the edge `people.person.nationality`. Using SPARQL, a user can retrieve any entity or relation if that user knows the pattern of the correct triple, including the answer for that user. This feature is strongly related to the specific NLP task, question answering over a knowledge base.

2.3.2 Question answering over a knowledge base

Question answering over a knowledge base (QAKB) is a task of natural language processing evaluating an ability to find the correct answer from a knowledge base from natural language questions. Since this task can provide the natural language interface of a knowledge base to users, it has become an essential task for bridging between users and a knowledge base (Chakraborty et al., 2019, Lopez et al., 2011). For example, if a question is "Which country is Michael Jordan from?", then one SPARQL query for getting the answer to the given question is Equation 2.21. QAKB task helps users reach the correct answer "Unite States" without any expert knowledge for SPARQL.

Formally, if a given natural language question is q, the QAKB task is defined as returning a correct answer a from all possible A in K for q. One of the traditional methods to solve the QAKB task is semantic parsing, which aims to translate q into an executable representation f (Berant et al., 2013, Reddy et al., 2014, Unger et al., 2014). f should return the correct answer a from a knowledge base, and f also can be understood as the meaning representation for q.

Many formal query languages, such as SPARQL, λ-DCS (Liang, 2013), and FunQL (Kate and Mooney, 2006) are employed for generating f. However, all of them need the information of entities and relations in q for guaranteeing f to be the same semantic representation with q. It is why parsing entities and relations from a given question is important in this task.

Thus, we can split question answering over a knowledge base into various subtasks in practice. The first subtask is to find which entities in K appear in a given question q, and this subtask is usually called *entity linking*. For example, if "Which country is Michael Jordan from?" is a given question, then the predicted entity should be `Michael_Jordan` in a knowledge base. One challenging problem for entity linking is that multiple entities can be written as the same utterance. For example, let us assume that a user uses abbreviations for his question, such as "Which country is MJ from?". In this case, MJ can be interpreted as both "Michael Jordan" and "Michael Jackson". Furthermore, it is hard to train questions for all entities because of the massive size of entities in a knowledge base. While many researchers have tried to solve those problems (Mendes et al., 2011, Yang and Chang, 2016), it is still a challenging subtask in question answering over a knowledge base.

Another subtask is to find which relations in K appear in a given question q, and this subtask is usually called *relation prediction*. In our example of "Which country is Michael Jordan from?", one proper relation for this question in Freebase can be `people.person.nationality`. However, we can not find the term "nationality" in "Which country is Michael Jordan from?". Moreover, we can paraphrase this question to various utterances without the term "nationality", such as "Where is Michael Jordan from?" and "Where is the birthplace of Michael Jordan?". It is why this subtask is a challenging problem.

While we can parse correct entities and relations from a given question, we need to generate an executable representation with entities and relations for getting the right answer of q. In the SPARQL case, an entity `Michael_Jordan` and a relation `people.person.nationality` should be composed like Equation 2.21 for answering a given question "Which country is Michael Jordan from?". In practice, this subtask, we call it *evidence integration*, involves deciding the structure of logical form and selecting suitable operators for logical form. Since a generated logical form can become unexecutable with only one trivial miss, such as an exchange between subject and object, evidence integration is critically important for question answering over a knowledge base.

Many researchers have proposed benchmark datasets for question answering over a knowledge base using various knowledge bases. In this chapter, we introduce popular datasets for Freebase. Free917 (Cai and Yates, 2013) is the first dataset for machine learning-based semantic parsing over Freebase. It contains 917 questions on a subset of Freebase, called *Freebase Commons*, covering 81 domains. Berant et al. (2013) find that each question tends to contain words directly

Dataset	train	valdiation	test
Free917	512	129	276
WebQuestions	3,778	-	2,032
WebQSP	2,478	620	1,639
SimpleQuestions	75,910	10,845	21,687
GraphQuestions	2,608	-	2,608
ComplexWebQuestions	27,734	3,480	3,475
FreebaseQA	20,358	3,994	3,996

TABLE 2.3: The statistic of popular datasets for QAKB and SFQA.

related to the target Freebase relation. For example, "What genre of music is B12?" requires the gold relation `music.artist.genre` to be answered.

Since 917 is too small for training and testing the machine learning model, WebQuestions (Berant et al., 2013) was proposed with 5,810 questions. Aiming at creating more natural questions than Free917, each question is derived from the Google Suggest API, followed by filtering by crowd workers. Consequently, the authors observe a larger divergence between the question words and relations, such as "What music did Beethoven compose?" for the fore-mentioned relation `music.artist.genre`. One limitation of WebQuestions is a lack of formal queries as gold data. Yih et al. (2016) suggested WebQSP, a subset of WebQuestions including annotated formal queries additionally.

FreebaseQA (Jiang et al., 2019) is the latest dataset aiming at more difficult factoid questions than SimpleQuestions while maintaining the scale of data size. Specifically, the questions in this dataset are first sampled from TriviaQA (Joshi et al., 2017) and then filtered by heuristics to collect factoid questions answerable on Freebase. Other datasets have been proposed continuously, such as GraphQuestions (Su et al., 2016), ComplexWebQuestions (Talmor and Berant, 2018b). Table 2.3 shows the list of popular datasets for QAKB.

2.3.3 Simple factoid question answering

While many datasets have been proposed for QAKB, QKAB is still a challenging and unsolved task yet. Bordes et al. (2015) suggested a more straightforward task definition for QAKB that questions only require one fact to be answered. We call this task as simple factoid question answering (SFQA) after this. Formally, if a triple t for a given question q is (e_1, p_1, e_2), this task demands to predict e_1 and p_1 from a given question q. In other words, this task takes only one template for the SPARQL query, "`SELECT` ?answer `WHERE` { e_1 p_1 ?answer . }". Therefore, the evaluation method of this task can be simplified as matching the predicted subject and relation with the gold subject and relation, without considering to generate an executable logical form.

Bordes et al. (2015) also proposed the benchmark dataset for SFQA in their paper. This dataset, called SimpleQuestions, has an important difference with other datasets in that this dataset only contains answerable questions by one fact. Furthermore, SimpleQuestions (Bordes et al., 2015) is the most largest dataset containing with 108,442 questions. Like previous datasets for QAKB, such as Free917 and WebQuestions, SimpleQuestions is also based on Freebase. In particular, each question is created from a sampled fact in Freebase, which is then verbalized and paraphrased by a crowd worker. Possibly due to this procedure starting from a KB fact, we find that, as in Free917, this dataset also tends to verbalize a relation with directly related terms, such as "What type of music . . .?" for `music.artist.genre`.[1] Since this approach eases data collection, it is popular in data creation for semantic parsing (Talmor and Berant, 2018a, Trivedi et al., 2017, Wang et al., 2015). The authors also define a subset of Freebase called FB2M covering 2M entities and 5K relations, including all entities appearing in WebQuestions, and create all questions from this subset.

In Bordes et al. (2015), the proposed baseline system for SimpleQuestions reported 63.9% accuracy for SimpleQuestions. Subsequently, neural network based systems have been proposed for SimpleQuestions. Ture and Jojic (2017) suggested the most successful system for SimpleQuestions with 86.8% accuracy. However, it is not confirmed since their system and result were not reproducible. Except for this system, many systems have reported around 75% accuracies for SimpleQuestions in recent days (Huang et al., 2019, Lukovnikov et al., 2019, Mohammed et al., 2018, Petrochuk and Zettlemoyer, 2018, Yin et al., 2016, Yu et al., 2017).

Here, we introduce four systems proposed for SFQA, considering their high accuracy on SimpleQuestions and the reproducibility of the system. [2] The basic assumption for those systems is that all questions can be answered by correctly predicting a subject entity e and a relation r on the knowledge base. For predicting the best pair, all systems employ a pipeline consisting of three different submodules below:

1. entity linking, which outputs a set of candidate subject entities $\{e\}$;
2. relation prediction, which outputs a set of candidate relations $\{r\}$; and
3. evidence integration, which finds the best $(\hat{e}, \hat{r})$ pair by reranking the candidate pairs.

First, we introduce BuboQA (Mohammed et al., 2018). In this system[3], both entity linking and relation prediction are modeled with simple classifiers. Despite its simplicity, this approach

[1] Cai and Yates (2013) only mentions that questions are written by two native English speakers and do not state whether they access a relation when writing questions, but we find two datasets are similar in this respect.

[2] When searching for open software, we often found that many systems along with a paper are not self-contained; in particular, they often are missing an entity linking module. This is especially the case for systems targeting WebQustions, for which many systems rely on the outputs of the entity linker used in (Yih et al., 2015) and found in `https://github.com/scottyih/STAGG`, while the entity linker itself is not available.

[3] `https://github.com/castorini/BuboQA`

outperforms several more complex architectures (Bordes et al., 2015, Yin et al., 2016). Specifically, for entity linking, a trained LSTM first detects the entity spans. This procedure is called entity detection in BuboQA. The predicted spans then heuristically mapped to the candidate KB entities and scored with the Levenshtein distance to the canonical entity label in the entity linking submodule in BuboQA.

Relation prediction is performed independently by another classifier on top of a different LSTM. Finally, the best combination of $(\hat{e}, \hat{r})$ is found according to a weighted sum of these two module scores.[4] This is an extension of an even simpler baseline of Ture and Jojic (2017), and a similar approach is employed in Petrochuk and Zettlemoyer (2018). In addition, Lukovnikov et al. (2019) proposed an extended system of BuboQA employing BERT. While they suggest minor changes in the structure of BuboQA, such as unifying entity detection and relation prediction into the same encoder, most parts of the proposed system are similar to BuboQA.

Note that BuboQA treats relation prediction as to the classification problem among relations appearing in the training data. It means that it cannot solve zero-shot relation prediction, which occurs to some extent, especially in the dataset transfer experiment. On the other hand, the other three systems theoretically can handle them, as described in the following.

Next, we introduce Hierarchical Residual BiLSTM (HR-BiLSTM) (Yu et al., 2017). In this system (and the next, KBQA-Adapter), relation prediction is performed differently, not by classification on a fixed set of relations, but by mapping a shared embedding space for KB relations and texts. This model encodes both question tokens and relation tokens (e.g., "music artist genre" for `music.artist.genre`) by different encoders. Relation candidates are then ranked by cosine similarity between the outputs of two encoders. This method allows us to calculate the score of an unseen relation.

KBQA-Adapter (Wu et al., 2019) [5] is an improvement to HR-BiLSTM with an additional adversarial adapter coupled with the relation encoder. The motivation of this adapter is to improve the zero-shot relation prediction performance. To this end, the adapter receives a relation embedding for r provided by KG embeddings, which is JointNRE (Han et al., 2018), transforming it to an embedding space where unseen relations can be handled properly.

Knowledge Embedding-based QA (KEQA) (Huang et al., 2019) [6] also builds on an external knowledge graph embedding, TransE (Bordes et al., 2013), which is used as the more direct and central part in the system. Given a knowledge graph embedding, which is fixed, this model tries to map each question into an entity embedding $\hat{\mathbf{e}}$ and relation embedding $\hat{\mathbf{r}}$, using separate LSTMs. We expect $\hat{\mathbf{e}}$ to be close to the gold node embedding in the graph and $\hat{\mathbf{r}}$ to the gold

[4] Although the paper mentions that the two scores are multiplied, they are summed with fixed weights in their official implementation.

[5] https://github.com/wudapeng268/KBQA-Adapter

[6] https://github.com/xhuang31/KEQA_WSDM19

relation embedding. Also, we would expect that the transition defined by the embedding model (e.g., addition for TransE), $f(\hat{\mathbf{e}}, \hat{\mathbf{r}})$, will get close to the answer node embedding. The query generation step of the system selects the ($\hat{e}$, $\hat{r}$, $\hat{o}$) triple based on this intuition, by minimizing the summed distances from embeddings corresponding to $\hat{e}$, $\hat{r}$, and $\hat{o}$ to the obtained encoded embeddings.

2.4 Related works

One purpose of this study is to explain the inner working of language models on downstream tasks of NLP. Though word and contextual embeddings have been widely used in the natural language processing field, their inner working is not clearly explained yet. This chapter presents an overview of previous studies for understanding and evaluating the inner working of language models, primarily focusing on encoded linguistic knowledge on language models. Furthermore, we also introduce essential issues for the evaluation and source for SFQA datasets. Those issues are related to the application of our PLSPM framework for SFQA systems.

2.4.1 Measuring encoded linguistic knowledge on language models

Since various language models have been proposed, it has become a natural question for researchers which language model is good. The most basic way to evaluate language models is to compare the accuracy of downstream tasks using various language models. This approach is based on a natural intuition; if one language model is used as a sound input feature of a supervised model for a downstream task of NLP, then that model should be a good model. Researchers have employed various NLP tasks for the evaluation, such as part-of-speech tagging, named entity recognition, sentiment analysis, and so on (Schnabel et al., 2015, Turian et al., 2010). Furthermore, researchers have proposed a package containing various NLP tasks and a straightforward script for evaluating them to conduct this evaluation conveniently (Conneau and Kiela, 2018, Nayak et al., 2016, Wang et al., 2019a).

This evaluation, called *extrinsic evaluation*, has limitations in evaluating the goodness and usefulness of language models. First, we can not experiment with all downstream tasks of NLP for language models for assessing language models. Second, a superior result for one downstream task is sometimes not transferred to other tasks. Third, extrinsic evaluation can not explain why a specific language model is more or less helpful for the target downstream task. However, it also has a substantial advantage that the result of this evaluation is easy to interpret whether one language model is helpful for the evaluated downstream task. In other words, it indicates that this evaluation can be applied to researchers who want to find helpful language

models for their interested downstream tasks. Therefore, extrinsic evaluation has been widely employed for examining and evaluating language models despite many limitations.

Meanwhile, researchers assumed that a good language model should understand semantics like human beings (Bakarov, 2018, Baroni et al., 2014, Batchkarov et al., 2016a, Gladkova and Drozd, 2016). It implies that we can evaluate the general quality of language models by examining encode linguistic knowledge on language models. Baroni et al. (2014) is one of earlier studies suggesting the concept of *intrinsic evaluation*, which compares human judgments and the result calculated by language models for evaluating language models. They employed semantic relatedness (Agirre et al., 2009, Bruni et al., 2014, Finkelstein et al., 2001, Rubenstein and Goodenough, 1965), synonym detection (Landauer and Dumais, 1997), concept categorization (Almuhareb, 2006, Baroni et al., 2008, 2010), selectional preferences (McRae et al., 1998, Padó and Lapata, 2007), and word analogy (Mikolov et al., 2013a) as the intrinsic evaluation for evaluating word embedding models.

While many intrinsic evaluations have been proposed, word similarity (Hill et al., 2015, Miller and Charles, 1991) had been a representatively employed task for the intrinsic evaluation. In this task, a pair of words with a similarity score annotated by a human is given. The researcher examines whether the distance of given words calculated with the target language models is similar to the given annotated similarity score or not. For example, if a given input is "Cup, Mug", then the researcher calculates the cosine similarity between "Cup" and "Mug" on the target language model. Calculated cosine similarity is compared with the human-annotated similarity score to examine whether the target language model can understand the similarity between given terms. Because of its intuitiveness and simplicity, researchers had employed word similarity as the intrinsic task, especially word similarity, for evaluating language models (Baroni et al., 2014, Chiu et al., 2016, Schnabel et al., 2015).

However, more recent studies argued that the result of word similarity does not report high correlation coefficients with accuracies of downstream tasks of NLP on language models (Batchkarov et al., 2016b, Chiu et al., 2016, Faruqui et al., 2016, Gladkova and Drozd, 2016, Schnabel et al., 2015). For some datasets of the word similarity task, the similarity score of some word pair means relatedness, while the similarity score of other word pair means semantic similarity. Human annotators often confused those definitions in one dataset, which causes the low reliability of the word similarity task itself. Also, the size of many datasets for the word similarity task is usually smaller than one thousand, which causes scaling problems for evaluating target language models. Therefore, other intrinsic tasks have been proposed to overcome limitations of word similarity (Camacho-Collados and Navigli, 2016b, Ettinger and Linzen, 2016, Gladkova et al., 2016, Søgaard, 2016). Table 2.4 lists popular tasks for the intrinsic evaluation and their brief descriptions.

Task	Purpose	Sample input	Expected answer	Evaluation
Word similarity (Hill et al., 2015, Miller and Charles, 1991)	Predict the similarity score for the given word pair	$\{Cup, Mug\}$	0.7	Pearson coefficient
Word analogy (Gladkova et al., 2016, Mikolov et al., 2013b)	Predict a proper word for the given relationship	$\{Man, Woman\}$, $King$	$Queen$	Accuracy
Outlier word detection (Camacho-Collados and Navigli, 2016b)	Find a word of the different category from given words	$\{banana, lemon, book, orange\}$	$book$	Outlier position percentage, Accuracy
Semantic priming (Ettinger and Linzen, 2016)	Predict a psycho-linguistically related word for the given word	$presume$	$assume$	r^2 value of regression model
Functional Magnetic Resonance Imaging (fMRI) (Søgaard, 2016)	Predict a neuro-linguistically similar word for the given word	$rolling$	pig	human observation

TABLE 2.4: List of employed or proposed tasks for the intrinsic evaluation.

One popular task as an alternative for word similarity is word analogy, proposed by Mikolov et al. (2013b). In word analogy, one question consists of two pairs of words that share the same relationship. For example, two word pairs "day, days" and "year, years" share the same relationship: the singular and plural representation for one noun. In the practical evaluation, one word in the given question is masked and needed to be predicted by the target language model. If a masked term is "years", then the system needs to predict "years" with other terms in the given question, "day, days, and year". Since the relationship to be predicted is not ambiguous, unlike word similarity, word analogy does not need to consider the limitation of word similarity mentioned above. Also, word analogy usually does not need human annotators for creating questions. We can automatically generate pairs of the word sharing the same relationship using external knowledge sources, such as a dictionary and thesaurus. It allows the dataset of word analogy to be a large-scale dataset, such as BATS (Gladkova et al., 2016) consisting of 98,000 questions.

Intrinsic evaluation is based on the assumption that encoded linguistic knowledge in language models should help to solve downstream tasks in NLP (Batchkarov et al., 2016a, Chiu et al., 2016). Word similarity is not employed for the intrinsic evaluation anymore since the accuracy of word similarity does not correlate with the accuracy of downstream tasks. In this way, proving correlations or causal relationships between intrinsic evaluation and downstream tasks is essential when evaluating language models with intrinsic evaluations. Previous studies, such as Chiu et al. (2016), Rogers et al. (2018) and Wang et al. (2019b), tried to prove this traditional assumption using correlation analysis between accuracies of intrinsic tasks and downstream tasks of NLP by existing language models. Chiu et al. (2016) conducted correlation analysis involving ten word similarity datasets, one pos tagging dataset, one chunking dataset, one named entity recognition dataset, and 30 word embedding models. They reported that accuracies of most word similarity datasets do not correlate with accuracies of downstream tasks. Rogers et al. (2018) examined correlations among seven word similarity datasets, one word analogy dataset, their proposed linguistic diagnostics toolkit, seven downstream tasks of NLP, and 60 word embedding models. Also, Wang et al. (2019b) reported the result of correlation analysis employing 13 word similarity datasets, two word analogy datasets, three concept categorization datasets, two outlier detection datasets, one subspace alignment dataset, five downstream tasks, and six word embedding models.

However, we find some conflicting results among previous studies. For example, Rogers et al. (2018) and Wang et al. (2019b) suggested the conflicting conclusion about the effectiveness of word analogy for the accuracy of downstream tasks. One important reason for this phenomenon may be the difference between employed language models and downstream tasks in their studies. It means that their results are not generally robust. We suppose that correlation analysis, the primary method in their studies, caused their conflicting conclusions. The traditional quote, *correlation does not imply causality*, indicates that depending on only the result of correlation

analysis may derive the false interpretation for the relationship between two variables. Many other possibilities are available for the high correlation coefficient, such as the existence of an external factor, bidirectional causation, and a just coincidental case. In this study, we aim to suggest a more robust way to investigate the causal relationship between the accuracy of intrinsic evaluation and downstream tasks.

2.4.2 Probing the inner working of language models

Since contextual embeddings have become a new indispensable tool for downstream tasks of NLP replacing word embeddings in recent days, researchers have tried to examine the quality of contextual embeddings. In earlier studies, probing tasks, which were designed to examine encoded linguistic knowledge for sentence embeddings, had been proposed (Adi et al., 2017, Shi et al., 2016). SentEval (Conneau and Kiela, 2018) is an advanced package consisting of ten probing tasks evaluating surface, syntactic, and semantic knowledge. They are binary classification tasks with one dense layer to minimize the effect of the neural network based model and focus on encoded knowledge in the result of the target sentence encoder. However, the interest of researchers has moved to use downstream tasks for examining encoded linguistic knowledge because of low correlation coefficients between accuracies of some probing tasks on SentEval and other downstream tasks (Conneau and Kiela, 2018).

Many researchers have involved various downstream tasks of NLP as well as probing tasks for evaluating contextual embeddings. We list some examples of previous studies involving downstream tasks for evaluating contextual embeddings in Table 2.5. For example, SentEval also provides 18 downstream tasks of NLP for extrinsic evaluations. General language understanding evaluation (GLUE) dataset (Wang et al., 2019a) is a more recent toolkit for evaluating contextual embeddings. It contains 11 downstream tasks of NLP across sentence-level classification tasks, similarity and paraphrase tasks, and inference tasks. While those toolkits have been used to examine contextual embeddings (Devlin et al., 2019, Kovaleva et al., 2019, Sanh et al., 2019), researchers have also employed other downstream tasks for their research purpose not depending on those toolkits (Liu et al., 2019a, Tenney et al., 2019a,b). Since most studies of them aimed to reveal and understand the inner working of BERT (Devlin et al., 2019), they are named as BERTology (Rogers et al., 2020).

Devlin et al. (2019) proved that a pretrained BERT could work an end-to-end system by adding one output layer. BERTologies usually have investigated the end-to-end system using BERT compared with previous studies focusing on language models themselves. They have tried to examine encoded linguistic knowledge on BERT by various approaches, such as attention analysis (Liu et al., 2019a), edge probing (Tenney et al., 2019a,b), and layerwise analysis with diagnostic classifiers (Lin et al., 2019). As a result, they have proved that BERT contains a

variety of linguistic knowledge including syntactic and semantic information. For example, Lin et al. (2019) showed that each layer of BERT encoded different linguistic knowledge. They proved that the lower layers tends to contain the information of word order, while the higher layer tends to contain the information of hierarchical structured order. Liu et al. (2019a) suggested that the middle layers of BERT is the best for syntactic knowledge, and the final layers of BERT usually became to be the most specific for the target task.

Though they revealed that BERT encodes what linguistic knowledge in where, Rogers et al. (2020) mentioned their limitations that observing linguistic knowledge from BERT does not explain how observed linguistic knowledge is used for solving downstream tasks. Furthermore, Rogers et al. (2020), Warstadt et al. (2019) also commented that a single observation, which most BERTology studies conducted, can cause conflicting conclusions among BERTology studies. Htut et al. (2019) showed that a change of the probing method could make a different conclusion for examining contextual embeddings. In addition, Tenney et al. (2019a) and Liu et al. (2019a) reported conflicting results which layer of BERT encoded syntactic knowledge mainly. Hence, we need the robust and causal explanation for the inner working of BERT between encoded linguistic knowledge the performance of BERT.

2.4.3 Evaluation of SFQA systems and datasets

Recently, Petrochuk and Zettlemoyer (2018) argued that existing SFQA systems nearly solve SimpleQuestions. They reported that some questions in SimpleQuestions have unresolved ambiguity for entities and relations. In the case of the question "who wrote gulliver's travels?", Freebase contains a variety of entities for "gulliver's travels", such as the name of book, film, and TV series. Since all those entities can be linked with a relation for "who wrote?", the correct entity for this question can not be a unique one. Petrochuk and Zettlemoyer (2018) calculated the upper bound accuracy of SimpleQuestions by eliminating all ambiguous questions. As a result, they concluded that existing SFQA systems already had nearly reached the upper bound accuracy of SimpleQuestions, 83.4%.

This problem is related to the evaluation method of SFQA, to match the predicted subject and relation with gold data (Bordes et al., 2015). The traditional evaluation method for QAKB datasets is to match the predicted object with gold data (Berant et al., 2013). The predicted object can be obtained by executing a SPARQL query consisting of predicted entities, relations, and the predicted template of the SPARQL query. However, SFQA only handles one static SPARQL query, "`SELECT` ?answer `WHERE` { subject relation ?answer .}". If the predicted subject and relation are correct, we do not need to execute a SPARQL query to evaluate a predicted object. It is the reason why SFQA takes a simplified evaluation method matching the predicted subject and relation with gold data without executing a SPARQL query.

Study	Task	Employed subtasks/datasets
Conneau and Kiela (2018)	classification	movie review, product review, subjectivity status, opinion-polarity (Wang and Manning, 2012), binary sentiment analysis, fine-grained sentiment analysis (Socher et al., 2013), question classification (Li and Roth, 2006)
	natural language inference	natural language inference (Marelli et al., 2014), SICK-E (Marelli et al., 2014)
	semantic textual similarity	STS 2012 (Agirre et al., 2012), STS 2013 (Agirre et al., 2013), STS 2014 (Agirre et al., 2014), STS 2015 (Agirre et al., 2015), STS 2016 (Agirre et al., 2016), STS benchmark (Cer et al., 2017), SICK-R (Marelli et al., 2014)
	paraphrase detection	paraphrase detection (Dolan et al., 2004)
	image caption retrieval	COCO (Conneau et al., 2018)
Wang et al. (2019a)	single-sentence	CoLA (Warstadt et al., 2018), SST-2 (Socher et al., 2013)
	similarity & paraphrase	MRPC (Dolan and Brockett, 2005), STS-B (Cer et al., 2017), QQP[7]
	inference	MNLI (Williams et al., 2018), QNLI (Rajpurkar et al., 2016), RTE (Bar Haim et al., 2006, Bentivogli et al., 2009, Dagan et al., 2006, Giampiccolo et al., 2007), WNLI (Levesque et al., 2011)
Liu et al. (2019a)	token labeling	part-of-speech taggging (Marcus et al., 1993, Silveira et al., 2014), CCG supertagging (Hockenmaier and Steedman, 2007), syntactic constituency ancestor tagging, semantic tagging (Bjerva et al., 2016), preposition supersense disambiguation (Schneider et al., 2018), event factuality (Rudinger et al., 2018)
	segmentation	syntactic chunking (Sang and Buchholz, 2000), named entity recognition (Sang and Erik, 2002), grammatical error detection (Yannakoudakis et al., 2011), conjunct identification (Ficler and Goldberg, 2016)
	pairwise relations	syntactic dependency arc prediction, e syntactic dependency arc classification, semantic dependency arc prediction, semantic dependency arc classification (Oepen et al., 2015), coreference arc prediction (Pradhan et al., 2012)
Tenney et al. (2019a,b)	labeling tasks	part-of-speech, constituents, named entities, semantic roles, coreference (Weischedel et al., 2013), dependencies (Silveira et al., 2014), semantic proto-roles (Rudinger et al., 2018, Teichert et al., 2017), relation classification (Hendrickx et al., 2010)

TABLE 2.5: Sample studies which employed downstream tasks for evaluating or investigating the inner work of contextual embeddings.

This simplified evaluation method makes the computing cost of the evaluation for SFQA low by omitting the working process with the external knowledge base. However, it also caused the problem of ambiguity for evaluating predicted entities and relations. If the final accuracy of the SFQA system is evaluated by the predicted object like other QAKB datasets, then we need only to consider whether the predicted object is `Jonathan_swift` for the given question "who wrote gulliver's travels?". In practice, we need to resolve disambiguation problems both for the predicted subject and relation. While "who worte?" can be represented in Freebase by a variety way, such as `film.film.story_by`, `film.film.written_by`, and `book.written_work.author`, most of them are treat as a negative weight during training. Petrochuk and Zettlemoyer (2018) just excluded questions containing ambiguous expressions as unanswerable questions for calculating the upper bound accuracy. Therefore, the effect of those ambiguous questions for the trained SFQA system has not been investigated yet.

Serban et al. (2016) and Jiang et al. (2019) commented another issue that questions in SimpleQuestions tend to contain labels of the gold subject and relation directly. It is another difference between SimpleQuestions and other QAKB datasets. Other QAKB datasets, such as WebQuestions, tend to express the relation with paraphrasing. For example, a relation `people.person.profession` is written like "what job does ... have?" or "who is ...?" in questions of WebQuestions. On the contrary, the same relation `people.person.profession` is written like "name the profession of ..." in questions of SimpleQuestions. According to Serban et al. (2016) and Jiang et al. (2019), the creating process for questions is one reason for this phenomenon. When creating questions of SimpleQuestions, crowd workers wrote a question with the suggested Freebase fact (Bordes et al., 2015). As a result, SimpleQuestions becomes a more simple dataset for predicting entities and relations than other QAKB datasets, literally.

In this way, we found some issues which make us consider whether the success of SimpleQuestions indicates the success of this task itself in general. This problem, which is called the robustness of a model, has been studied in other natural language processing fields (Jia and Liang, 2017, McCoy et al., 2019, Naik et al., 2018, Ribeiro et al., 2020). It is an important problem since a practical system needs to be robust on outliers in the training data for actual user queries. However, few studies have examined the robustness of a trained system on question answering over a knowledge base. In this study, we aim to investigate whether the success of SimpleQuestions also indicates the success of this task itself in general in the aspect of the inner working of existing systems.

Chapter 3

Validating causal relationships between linguistic knowledge and downstream tasks

This section presents a novel framework for explaining the inner working of language models by employing PLSPM, a statistical method for testing causal hypotheses between variables. First, we introduce the limitation of previous studies and our motivation for employing PLSPM in evaluating language models. Since we need to examine whether employing PLSPM can report comparable and reasonable results with previous studies, we conduct the experiment examining causal hypothesis between linguistic knowledge and accuracies of the downstream task of NLP, which was proposed in earlier studies. We also introduce causal diagrams representing causal assumptions of previous studies. As a result of experiments, we find that employing PLSPM can provide more robust and informative analysis results for the target language model in understanding relationships between encoded linguistic knowledge and accuracy of downstream tasks of NLP. Furthermore, PLSPM also reveals that word analogy may contain structural issues for categorizing questions into linguistic knowledge, such as inflectional morphology.

3.1 Motivation for employing the statistical method

Previous studies for evaluating language models had been conducted with an observation for few samples (Liu et al., 2019a, Schnabel et al., 2015, Tenney et al., 2019a,b) or a simple correlation analysis (Chiu et al., 2016, Rogers et al., 2018, Wang et al., 2019b). While they had contributed to exploring what linguistic knowledge is encoded on language models, their methodologies have limitations for explaining the inner working of language models, as we discussed in Section 2.2.3. For example, Chiu et al. (2016), Rogers et al. (2018), Schnabel et al. (2015), Wang et al. (2019b) reported conflicting results for the traditional assumption, *the accuracy of intrinsic evaluations are correlated with the accuracy of downstream tasks of NLP*, because of small sample sizes of language models and a change of intrinsic tasks. Rogers et al. (2020) also mentioned that finding a linguistic pattern on language models can not explain how linguistic knowledge on language models is used. It indicates that a statistical method is required for more robust analysis and causal explanation.

We try to explain the inner working of language models as one problem of hypothesis testing. Previous studies are based on many intuitive assumptions for language models, such as *accuracies of intrinsic evaluations represent how well linguistic knowledge is encoded*, *encoded linguistic knowledge on language models should affect accuracies of downstream tasks of NLP*. Those assumptions can be statistically tested by PLSPM, which we employ in this thesis if proper causal diagrams and observed variables are prepared. PLSPM has many advantages in understanding the inner working of language models compared with an observation or correlation analysis. For example, a PLSPM model provides statistical indexes for verifying whether the suggested causal diagram is acceptable or not, such as path coefficients for the strength of causal relations, determination coefficients (R^2) for the explanatory power of endogenous variables, and Goodness of Fit (GoF) for the explanatory power of whole PLSPM model. Also, PLSPM can incorporate multiple variables in one causal assumption, unlike simple correlation analysis in previous studies.

In this chapter, we aim to investigate traditional assumptions about the relationship between linguistic knowledge and the accuracy of downstream tasks on language models. For this purpose, we suggest a statistical method that can provide a robust and causal explanation for the inner working of language models.

3.2 Causal diagram

dataset-category (latent variable)	tasks (observed variables)
BATS-Inflectional Morphology (INF)	regular plurals, plurals (orthographic changes), comparative degree, superlative degree, infinitive:3ps.sg, infinitive:participle, infinitive:past, participle:3ps.sg, participle:past, 3ps.sg:past
BATS-Derivational Morphology (DER)	noun+less, un+adj., adj.+ly, over+adj./ved, adj.+ness, re+verb, verb+able, verb+er, verb+ation, verb+ment
BATS-Lexicography Knowledge (LEX)	hypernyms (animals), hypernyms (miscellaneous), hyponyms (miscellaneous), meronyms (substance), meronyms (member), meronyms (part-whole), synonyms (intensity), synonyms (exact), antonyms (gradable), antonyms (binary)
BATS-Encyclopedia Knowledge (ENC)	geography (capitals), geography (country:language), geography (uk city:county), people (nationalities), people (occupation), animals (the young), animals (sounds), animals (shelter), other (thing:color), other (male:female)
VecEval-Syntactic Properties (SYN)	POS-tagging (Toutanova et al., 2003), Chunking (Sang and Buchholz, 2000)
VecEval-Semantic Properties (SEM)	Named Entity Recognition (Sang and Erik, 2002), Sentiment Classification (Socher et al., 2013), Question Classification (Li and Roth, 2006), Natural Language Inference (Ganitkevitch et al., 2013)
SentEval-Classification (CLA)	Movie Review, Product Review, Subjectivity Status, Opinion-polarity (Wang and Manning, 2012), Binary Sentiment Analysis, Fine-grained Sentiment Analysis (Socher et al., 2013), Question Classification (Li and Roth, 2006)
SentEval-Natural Language Inference (NLI)	Natural Language Inference (Marelli et al., 2014)
SentEval-Semantic Textual Similarity (STS)	STS 2012 (Agirre et al., 2012), STS 2013 (Agirre et al., 2013), STS 2014 (Agirre et al., 2014), STS 2015 (Agirre et al., 2015), STS 2016 (Agirre et al., 2016), STS Benchmark (Cer et al., 2017), SICK-R (Marelli et al., 2014)
SentEval-Paraphrase Detection (PD)	Paraphrase Detection (Dolan et al., 2004)

TABLE 3.1: Details of the datasets used for our PLSPM models.

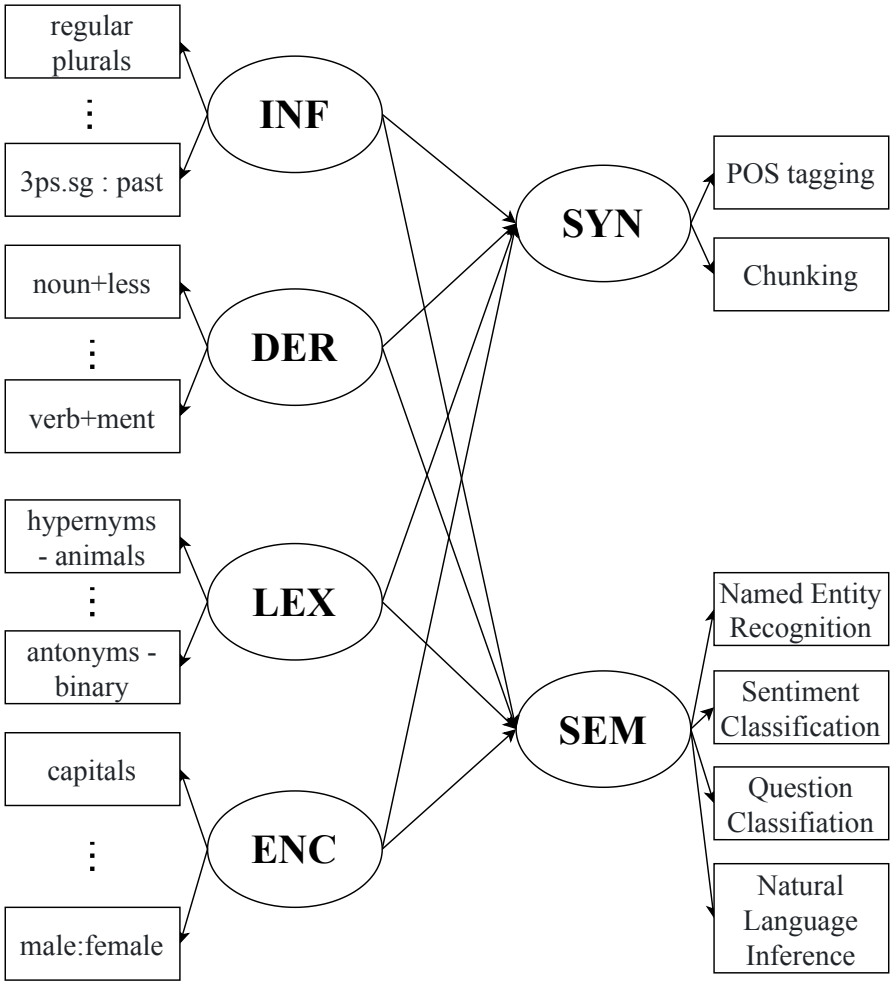

FIGURE 3.1: Causal diagrams for BATS-VecEval. All abbreviations are defined in Table 3.1.

When starting PLSPM, defining the causal diagram to be validated is required at first. In our case, the causal diagram should consist of causal relationships between encoded linguistic knowledge and accuracies of downstream tasks on the same language model. Our main causal hypothesis is that the accuracies of downstream tasks can be explained by the accuracy of intrinsic evaluation with causal relations. This hypothesis also implies that the intrinsic evaluation can measure encoded linguistic knowledge on the language model, another assumption from previous studies (Baroni et al., 2014, Batchkarov et al., 2016a). Therefore, observed variables for our causal diagram should be accuracies of intrinsic evaluations and downstream tasks.

We prepare BATS (Gladkova et al., 2016), one dataset of word analogy task, and VecEval (Nayak et al., 2016) and SentEval (Conneau and Kiela, 2018), toolkits containing downstream tasks of NLP, following experimental settings of previous studies (Chiu et al., 2016, Rogers et al., 2018, Wang et al., 2019b). Note that we do not use word similarity datasets like previous studies (Baroni et al., 2014, Chiu et al., 2016, Schnabel et al., 2015), because of the ambiguous definition of similarity and the problem of inter-annotator agreement on the dataset (Batchkarov et al., 2016a). The BATS dataset consists of four linguistic categories containing ten subcategories, such as

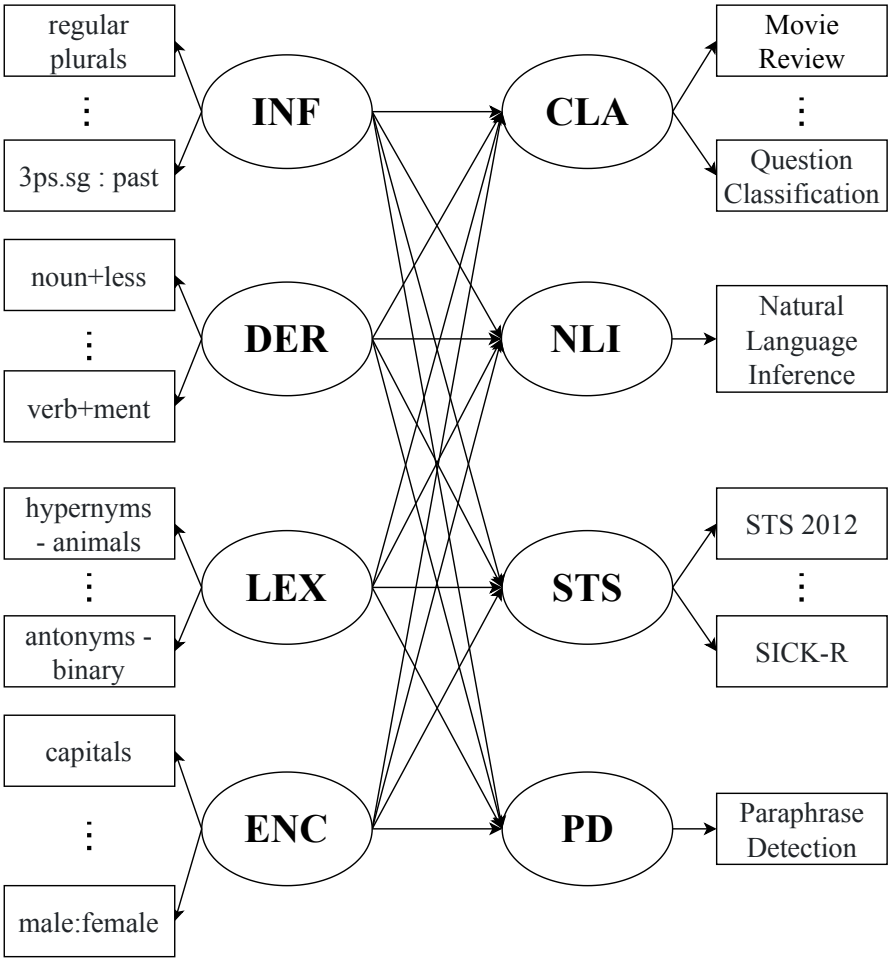

FIGURE 3.2: Causal diagrams for BATS-SentEval. All abbreviations are defined in Table 3.1.

inflectional morphology, derivational morphology, lexicography knowledge, and *encyclopedia knowledge*. Following Gladkova et al. (2016), we assume that each linguistic category is one latent variable that reflects the accuracies of its ten subcategories for the measurement model in our causal diagrams.

It has a variety of advantages in estimating PLSPM models to bind subcategories of BATS with one latent variable. First, measuring one latent variable by various observed variables makes the quality of measured one, encoded linguistic knowledge in this case, more robust and reliable. We can examine whether the structure of linguistic knowledge on the BATS dataset can be applied to word embedding by investigating the reliability of the measurement model. Moreover, we can reduce the number of parameters in the PLSPM model, which allows us to fit the model with fewer samples.

Note that we use the vector offset method (Mikolov et al., 2013b) to solve the BATS dataset, well known as the $Man + King = Woman + ?$. While Gladkova et al. (2016) suggested a new method LRCos, applying supervised learning to predict the answer term, we do not use it in

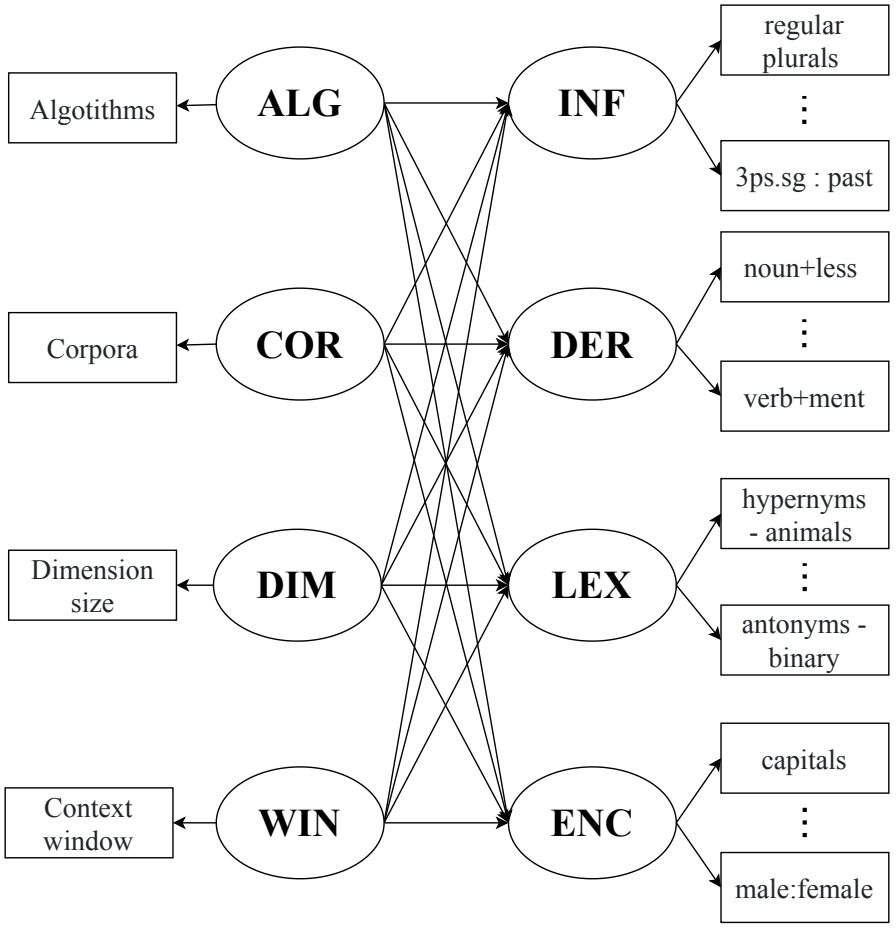

FIGURE 3.3: Causal diagram for hyperparam-BATS. All abbreviations are defined in Table 3.1 and Table 3.2.

our experiments. Because we intend to avoid the effectiveness of machine learning methods for evaluating how well linguistic knowledge is embedded.

For downstream tasks, we employ the VecEval (Nayak et al., 2016) and SentEval (Conneau and Kiela, 2018) datasets. They classified their employed downstream tasks into NLP research areas, more concretely *syntactic* and *semantic properties* in VecEval, and *classification, natural language inference, semantic textual similarity*, and *paraphrase detection* in SentEval. We design latent variables for downstream tasks with VecEval and SentEval in the same way as for BATS. For example, the latent variable for syntactic properties has POS tagging accuracy and chunking as observed variables. Table 3.1 lists details for the latent and observed variables from BATS, VecEval, and SentEval. Hereafter, we refer to the PLSPM model using the BATS and VecEval datasets as BATS-VecEval, and to the one using the BATS and SentEval datasets as BATS-SentEval. Figure 3.1 and Figure 3.2 show our causal diagrams for BATS-VecEval and BATS-SentEval.

In addition, we investigate the effectiveness of the hyperparameters of training language models for encoding linguistic knowledge. While Levy et al. (2015) and Lai et al. (2016) already reported

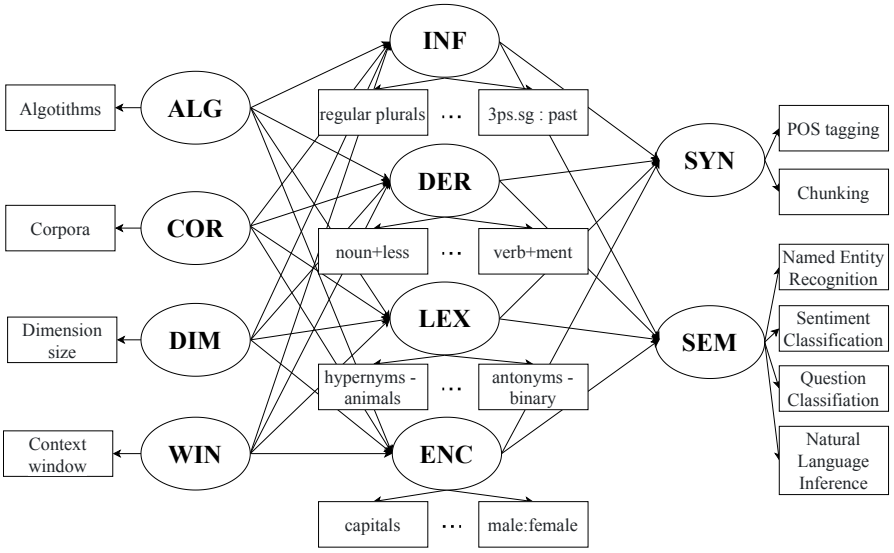

FIGURE 3.4: Causal diagrams for hyperparam-BATS-VecEval. All abbreviations are defined in Tables 3.1 and Table 3.2.

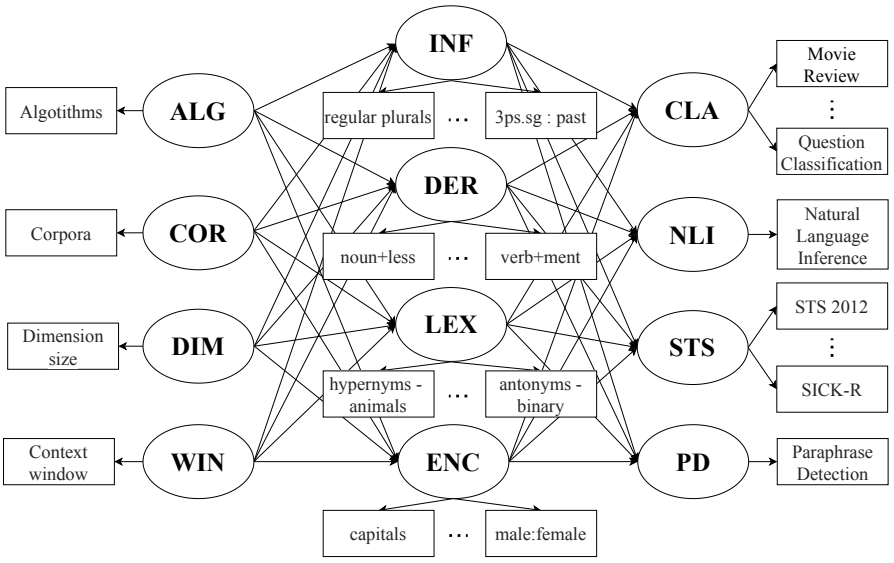

FIGURE 3.5: Causal diagrams for hyperparam-BATS-SentEval. All abbreviations are defined in Table 3.1 and Table 3.2.

that the hyperparameters of word embedding have a critical role in the accuracies of intrinsic evaluation and downstream tasks, they only conducted correlation analysis. We examine a causal diagram consisting of hyperparameters and intrinsic evaluation using BATS. We prepare four variables for the hyperparameter, including training algorithm, corpus, dimension, and context window, as shown in Table 3.2. Figure 3.3 presents our causal diagram involving hyperparameters and BATS. Note that the hyperparameters are independent variables with respect to each other, and we do not bind them as one latent variable. Moreover, because they include non-metric variables such as the algorithm and corpus, we use transformed scores of the hyperparameters during PLSPM estimation, following Russolillo (2012). We refer to the PLSPM model for the above causal diagram as hyperparam-BATS.

Hyperparameter	List
algorithm (ALG)	CBOW, skipgram, fastText
corpus (COR)	Wikipedia, New York Times
dimension size (DIM)	50, 100, 150, 200, 250, 300, 350, 400, 450, 500
context window (WIN)	1, 3, 5, 7, 9, 11, 13, 15, 17, 19

TABLE 3.2: List of hyperparameters for training word embeddings.

Furthermore, we incorporate hyperparameters into BATS-VecEval and BATS-SentEval as illustrated in Figure 3.4 and Figure 3.5. Note that we do not directly connect the latent variables of hyperparameters with the latent variables of downstream tasks in our causal diagram. In other words, we assume that the effectiveness of hyperparameters for downstream tasks can be explained only through the accuracies of intrinsic evaluation, which implies the ability of linguistic knowledge. Our causal diagram follows the ideal assumption that intrinsic evaluation, namely, that intrinsic evaluation examines the general quality of word embedding; therefore, it should also predict the accuracy of downstream tasks. We aim to examine this hypothesis with our PLSPM models using the above causal diagrams.

3.3 Experimental settings

When fitting a PLSPM model, both the causal diagrams and the observed variables are required as input. According to previous studies, we train a number of word embeddings with various sets of hyperparameters, according to previous studies (Chiu et al., 2016, Levy et al., 2015, Rogers et al., 2018). Table 3.2 lists the hyperparameters used for increasing the number of word embeddings. For other hyperparameters, we use the same values for all word embeddings, and we use the fixed training seed to prevent random effects of initialization. As a result, we obtain 600 word embeddings. Hyperparameters of word embedding have already been reported to affect the accuracies of intrinsic evaluation and downstream tasks significantly (Lai et al., 2016, Levy et al., 2015). We thus regard the result of one task with one word embedding as one data sample. Since we prepare results of solving each dataset by each word embedding model, our observed variable is a 600-dimension vector consisting of the results of BATS, VecEval, and SentEval on 600 word embeddings.

Note that the downstream tasks in VecEval and SentEval use various performance indicators, such as the accuracy, F1 score, and Pearson's r. However, we do not unify or transform them because we need its performance indicator of each dataset as suggested by the original papers. Therefore, we do not change the values of indicators except through normalization. Furthermore, we distinguish the two causal diagrams for VecEval and SentEval and do not merge them. The main reason is that they use different neural network models for solving downstream tasks. We should avoid model effects for observed variables because we do not consider any impact of a

	INF	DER	LEX	ENC	R^2
SYN	-	0.773	1.310	-	0.656
SEM	-	-0.189	-	0.771	0.546

TABLE 3.3: Path coefficients for each path and R^2 for the endogenous latent variables on BATS-VecEval. Paths with $p > 0.05$ are omitted.

machine learning model in our PLSPM models. To minimize the effect of a neural network model, we turned off fine-tuning option when running the evaluation script for VecEval. Moreover, we used a logistic regression model when running the evaluation script for SentEval.

We use the R package `plspm`[1] for our experiments, and for reproducibility we share our experimental scripts and all observed variable data[2].

3.4 Experiments

3.4.1 Relationship between accuracies of intrinsic evaluation and downstream tasks

First, we examine the reliability of the measurement model in BATS-VecEval. Cronbach's α and Dillon–Goldstein's ρ, for validating the measurement model of BATS-VecEval, are both larger than 0.7, indicating that the measurement model of BATS-VecEval is acceptable. The GoF of BATS-VecEval is 0.6484, which is also considered an acceptable value (Akter et al., 2011). Therefore, we can accept the causal hypothesis of the measurement model in BATS-VecEval that questions in subcategories of BATS and VecEval can represent the same latent variable.

We can interpret the effectiveness of a path between latent variables with the path coefficient for validating the structural model. In BATS-VecEval, there are eight paths between latent variables for BATS and latent variables for VecEval. Table 3.3 lists their coefficients and the R^2 values for SYN (Syntactic properties tasks in VecEval) and SEM (Semantic properties tasks in VecEval). Four paths, namely, DER (Derivational morphology questions in BATS)-SYN, DER-SEM, LEX (Lexicography knowledge questions in BATS)-SYN, and ENC (Encyclopedia knowledge questions in BATS)-SEM, have $p < 0.05$, indicating significant causal relations. The high path coefficients for DER-SYN and ENC-SEM are intuitively understandable because knowledge of derivational morphology helps syntactic analysis tasks such as POS tagging, and encyclopedia knowledge is indispensable in semantic analysis. The relation between lexicography and syntax is not trivial, but it has already been reported that accuracy on SimLex-999 (Hill et al., 2015), a dataset of word similarity to distinguish lexicographical relations, is correlated with POS tagging and chunking (Chiu et al., 2016). Our result is consistent with that observation.

[1] https://github.com/gastonstat/plspm
[2] https://github.com/mynlp/embedding-evaluation-plspm

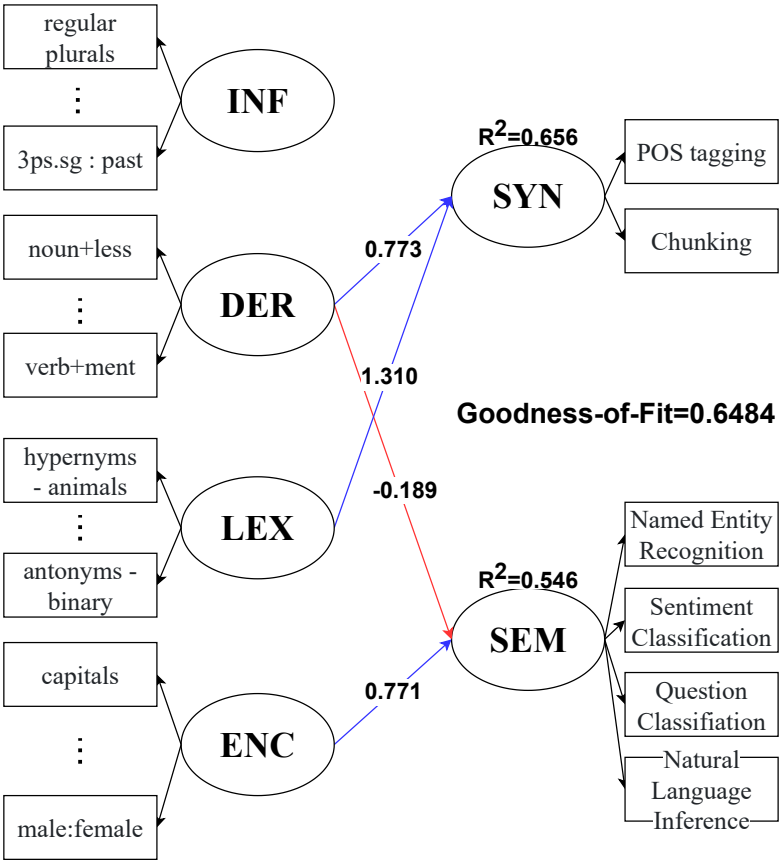

FIGURE 3.6: The estimated PLSPM model by the accuracy of BATS and VecEval

Another interesting observation is that INF, the latent variable for inflectional morphology, does not significantly affect the downstream tasks in VecEval. However, we further discuss inflectional morphology in Chapter 3.4.3. Among the rejected paths, the rejection of the path between lexicography knowledge and tasks of semantic properties seems counter-intuitive. We hypothesize that the main reason derives from the components of SEM; named entity recognition, sentiment classification, question classification, and natural language inference. Understandably, lexicography knowledge may not have enough explanatory power for some tasks for the SEM latent variable, such as named entity recognition. Figure 3.6 represents the estimated PLSPM model based on the suggested causal diagram for BATS and VecEval.

Next, we investigate BATS-SentEval in the same way. For the measurement model, both Cronbach's α and Dillon–Goldstein's ρ are larger than 0.7, indicating that the assumption of the causal diagram between the observed and latent variables is acceptable. The GoF of BATS-SentEval is 0.711, which is higher than that of BATS-SentEval. It implies that the accuracies of BATS can better explain the accuracies of SentEval than those of VecEval. Therefore, we conclude that BATS-SentEval is also acceptable.

As listed in Table 3.4, in the structural model of BATS-SentEval, all paths are accepted with $p < 0.05$, except INF (Inflectional morphology questions in BATS) -NLI (Natural language

	INF	DER	LEX	ENC	R^2
CLA	-0.565	1.140	1.490	0.716	0.619
NLI	-	0.368	0.640	0.647	0.807
STS	-	-0.397	-0.216	0.837	0.874
PD	-0.358	-0.812	-0.321	0.448	0.482

TABLE 3.4: Path coefficients for each path and R^2 for the endogenous latent variables on BATS-SentEval. Paths with $p > 0.05$ are omitted.

	ALG	COR	DIM	WIN	R^2
INF	-0.312	-0.213	0.580	-0.249	0.541
DER	0.969	-0.031	0.136	-0.068	0.963
LEX	-0.937	-0.106	0.150	-0.060	0.915
ENC	-0.861	0.268	0.218	0.072	0.865

TABLE 3.5: Path coefficients for each path and R^2 for the endogenous latent variables on hyperparam-BATS.

inference tasks in SentEval) and INF-STS (Semantic textual similarity tasks in SentEval). The results show that ENC, for encyclopedia knowledge, shows high path coefficients with all latent variables for the SentEval dataset, as SEM shows for VecEval. Among the latent variables of SentEval, classification tasks are well explained with derivational morphology, lexicography knowledge, and encyclopedia knowledge. Because most of the CLA (Classification tasks in SentEval) latent variable consists of sentiment analysis, this may indicate that such linguistic knowledge is helpful for sentiment analysis tasks. However, the results also show that NLI and STS are the best explained latent variables by the accuracy of BATS, according to the R^2 values. When $R^2 > 0.8$, it indicates an endogenous latent variable is excellently explained by its independent latent variables. Therefore, we argue that encyclopedia knowledge is strong enough to explain the evaluation results of semantic textual similarity, while the path coefficients of DER-STS and LEX-STS are low.

In contrast, PD (Paraphrase detection tasks in SentEval) shows the lowest R^2 value in BATS-SentEval. Although the value is not under the cut-off for rejecting this latent variable, it may indicate that the accuracy of BATS does not sufficiently explain the paraphrase detection task. Figure 3.7 represents the estimated PLSPM model based on the suggested causal diagram for BATS and SentEval.

3.4.2 Impact of hyperparameters

As Levy et al. (2015) and Lai et al. (2016) reported, hyperparameters for the training of word embedding affect the performance on solving downstream tasks. We thus analyze the effect of hyperparameters by adding new latent variables for hyperparameter values to the causal diagrams, as shown in Figure 3.3, Figure 3.4 and Figure 3.5.

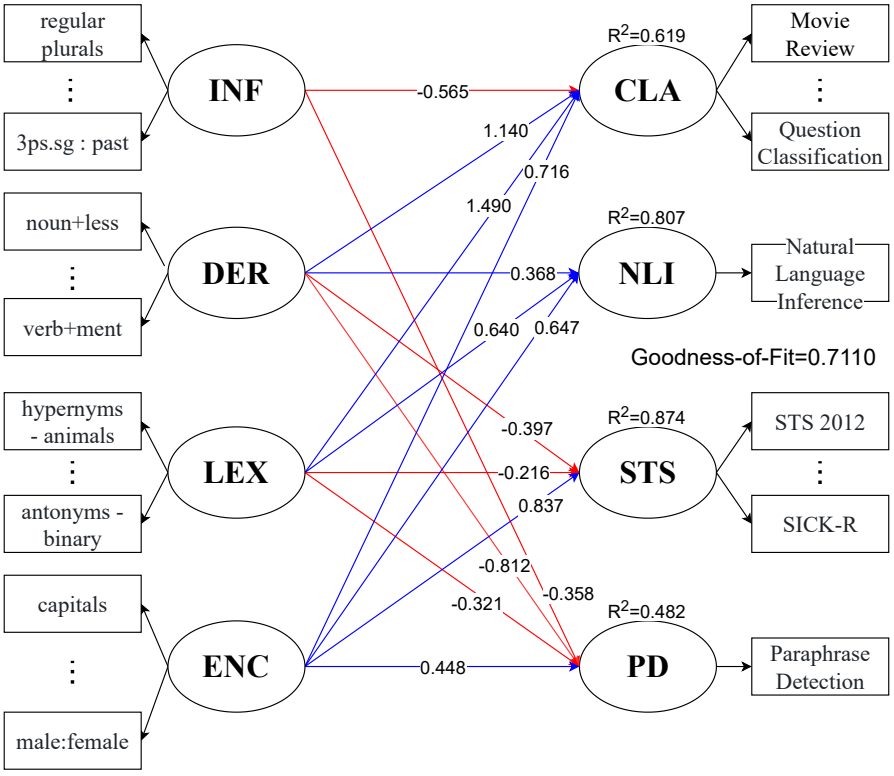

FIGURE 3.7: The estimated PLSPM model by the accuracy of BATS and SentEval

First, we examine hyperparam-BATS. Note that ALG, COR, DIM, and WIN consist of one observed variable; therefore, we do not need to validate the measurement model of hyperparam-BATS. Other latent variables, such as INF, DER, LEX, and ENC, show higher Cronbach's α and Dillon–Goldstein's ρ values than 0.7, as with BATS-VecEval and BATS-SentEval. Moreover, the GoF of hyperparam-BATS is 0.7521, the best value among our PLSPM models. Therefore, it is evident for hyperparam-BATS that the hyperparameters of word embedding are enormously influential for the accuracy of intrinsic evaluation.

Table 3.5 lists that the path coefficients and R^2 values for the structural model on hyperparam-BATS. There is no rejected path with $p > 0.05$, which indicates that all the hyperparameters impact the tasks in the BATS dataset. Training algorithms have especially strong relations with all categories of the BATS dataset, as indicated in the table. All hyperparameter values are processed with the nominal scaling (Russolillo, 2012). It means that we can not use the sign of path coefficients for interpretation. Therefore, we can conclude that the training algorithm is the most substantial factor for explaining the accuracies of intrinsic evaluation on hyperparam-BATS because of the high intensity of its path coefficient. Figure 3.8 represents the estimated PLSPM model based on the suggested causal diagram for hyperparameters and BATS.

Other hyperparameters are much weaker for predicting latent variables in path coefficients than the training algorithms. For encyclopedia knowledge, the path coefficients of the corpus and dimension are relatively high. It implies that the accuracies of encyclopedia knowledge are

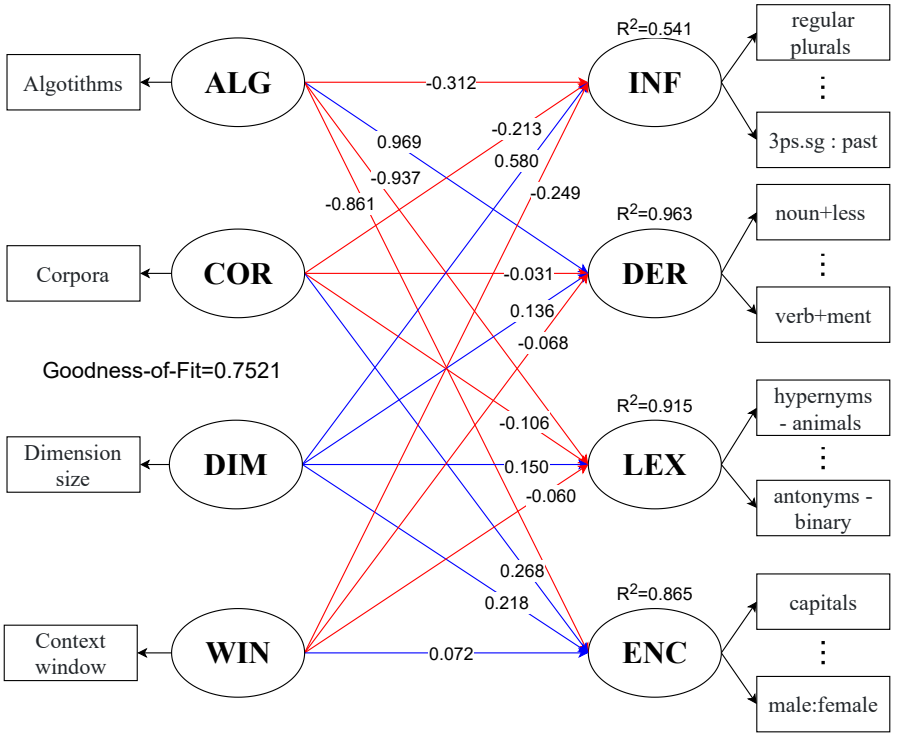

FIGURE 3.8: The estimated PLSPM model by the accuracy of BATS and hyperparameters

	ALG	COR	DIM	WIN	R^2
INF	-0.687	0.281	0.353	-0.127	0.691
DER	0.974	-	0.119	-0.051	0.966
LEX	-0.941	-0.061	0.153	-0.050	0.916
ENC	-0.878	0.226	0.212	0.062	0.871
	INF	**DER**	**LEX**	**ENC**	R^2
SYN	-0.335	0.953	1.390	0.497	0.688
SEM	-0.447	-	0.183	0.992	0.578

TABLE 3.6: Path coefficients for each path and R^2 for the endogenous latent variables on hyperparam-BATS-VecEval. Paths with $p > 0.05$ are omitted.

more related to the training corpus and dimension than those of other linguistic knowledge. Meanwhile, most latent variables of intrinsic evaluation have salient R^2 values greater than 0.85, with only the R^2 value for inflectional morphology being low, at 0.541. This problem is investigated in the next chapter.

Next, we investigate hyperparam-BATS-VecEval and hyperparam-BATS-SentEval to incorporate hyperparameters into the analysis of relationships between accuracies of BATS and downstream tasks. The main causal hypothesis of both hyperparam-BATS-VecEval and hyperparam-BATS-SentEval is that the effectiveness of hyperparameters on downstream tasks can be explained through the accuracy of intrinsic evaluation. To validate this hypothesis, we focus on the R^2 and GoF values of hyperparam-BATS-VecEval and hyperparam-BATS-SentEval. If our causal hypothesis helps explain the accuracy of downstream tasks, we should find that the R^2 and GoF

	ALG	COR	DIM	WIN	R^2
INF	-0.456	-	0.540	-0.210	0.545
DER	0.976	-	0.113	-0.043	0.967
LEX	-0.943	-0.059	0.147	-0.045	0.917
ENC	-0.888	0.196	0.207	0.063	0.874
	INF	DER	LEX	ENC	R^2
CLA	-	0.991	1.300	0.204	0.579
NLI	-	0.232	0.516	0.545	0.810
STS	-	-0.455	-0.190	0.689	0.871
PD	-0.555	-	0.430	0.282	0.522

TABLE 3.7: Path coefficients for each path and R^2 for the endogenous latent variables on hyperparam-BATS-SentEval. Paths with $p > 0.05$ are omitted.

values of hyperparam-BATS-VecEval and hyperparam-BATS-SentEval are higher than those of BATS-VecEval and BATS-SentEval.

Tables 3.6 and 3.7 list the path coefficients of hyperparam-BATS-VecEval and hyperpara m-BATS-SentEval. For both models, the results show that the R^2 values of most latent variables increase. Specifically, both SYN and SEM in hyperparam-BATS-VecEval have better R^2 values than they do in BATS-VecEval. Moreover, Table 3.8 lists the GoF values for all the PLSPM models. The GoF of hyperparam-BATS-VecEval is 0.7445, showing salient improvement over the value for BATS-VecEval, 0.6484. Therefore, we can conclude that downstream tasks in VecEval are more explainable with our causal hypothesis on hyperparam-BATS-VecEval.

On the other hand, it may not be easy to accept the same conclusion as that for hyperparam-BATS-VecEval on hyperparam-BATS-SentEval. As listed in Table 3.7, the R^2 values of CLA and STS on hyperparam-BATS-SentEval decrease below those on BATS-SentEval. The R^2 value of PD increases but is still the lowest R^2 value for hyperparam-BATS-SentEval. It indicates that the structure of hyperparam-BATS-SentEval is not suitable in explaining many tasks in SentEval. Though the GoF of hyperparam-BATS-SentEval is higher than that of BATS-SentEval, this result depends on the structural equations between the hyperparameters and BATS, not on those between BATS and SentEval.

As a result, our causal hypothesis, the effectiveness of hyperparameters for downstream tasks can be explained only through the accuracy of intrinsic evaluation, is not proper for explaining the accuracy of SentEval. This result implies two possible interpretations: that the accuracies of downstream tasks can be explained directly by the hyperparameters or that the tasks of intrinsic evaluation in BATS are not sufficient to explain the accuracy of SentEval.

PLSPM model	Goodness-of-Fit
BATS-VecEval	0.6484
BATS-SentEval	0.7110
hyperparam-BATS	0.7521
hyperparam-VecEval	0.7445
hyperparam-SentEval	0.7495

TABLE 3.8: GoF values for our PLSPM models.

3.4.3 Discussion with respect to previous studies

Our analysis using PLSPM reveals that the accuracy of intrinsic evaluation in BATS can explain the accuracy of downstream tasks both in VecEval and SentEval. Some of these relations were already reported in previous literature using correlation analysis. For example, the accuracy of POS tagging and chunking can be explained by derivational morphology and lexicography knowledge as reported in Chiu et al. (2016), Rogers et al. (2018), and Wang et al. (2019b). Similarly, classification and natural language inference tasks require derivational morphology, lexicography knowledge, and encyclopedia knowledge, which was also reported in Rogers et al. (2018) and Wang et al. (2019b).

Meanwhile, our PLSPM models also suggest some counter-intuitive relations between accuracies of intrinsic evaluation and downstream tasks. We already explained the reasons for some results that conflict with previous studies, such as the lexicography knowledge and NLP tasks for semantic properties in VecEval. The most significant problem is that, in this paper, the latent variable of inflectional morphology shows many rejected structural equations with $p > 0.05$, negative path coefficients on accepted structural equations, and relatively low R^2 values in the overall PLSPM models. It indicates that the accuracy of inflectional morphology in BATS may not have sufficient explanatory power for downstream tasks. This result conflicts with the results of previous studies, which reported that the accuracies of inflectional morphology correlate with the accuracy of downstream tasks (Rogers et al., 2018, Wang et al., 2019b).

The following reasons can explain this issue. First, we suppose that differences in the experimental setting for word embedding lead to conflicting results on inflectional morphology. For example, the accuracy of inflectional morphology in previous studies was calculated using the LRCos method (Gladkova et al., 2016), which differs from our experimental setup. In addition, the sample space of word embedding also differs, especially the conditions of the training algorithm and corpus. We leave further analysis on the effectiveness of those differences in our PLSPM models for future work.

Finally, we also investigate the relationships between the subcategories of inflectional morphology and the estimated score of INF in our PLSPM models. The left side of Figure 3.9 shows a plot with the loading of the observed variables, which is a correlation coefficient between the

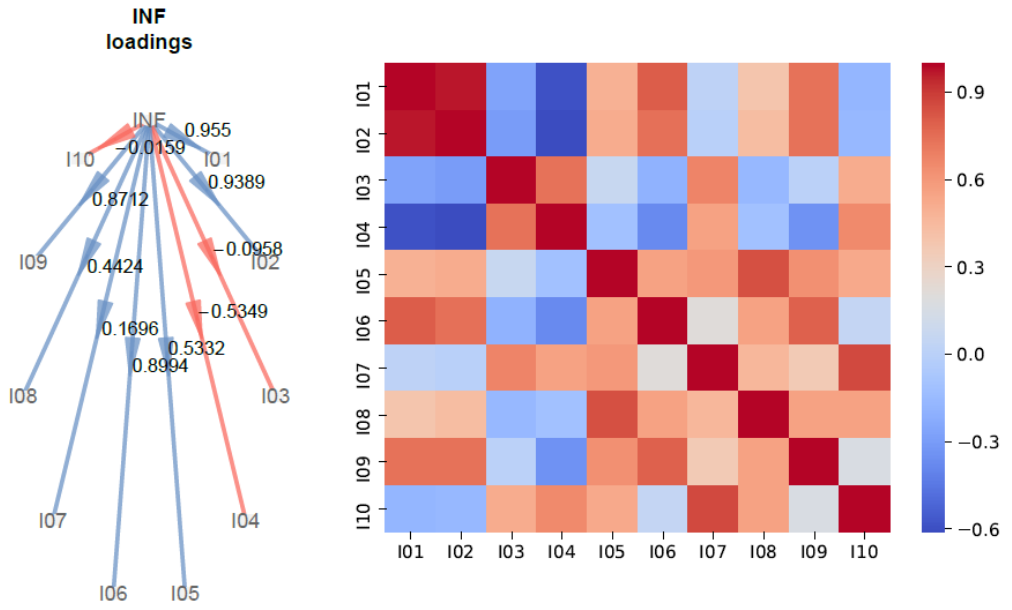

FIGURE 3.9: Left one presents loading plot of the observed variables for the INF latent variable. A red arrow indicates a negative loading. Right one presents spearman correlation heatmap for the INF questions in BATS. Here, I01 and I02 are noun plural questions, I03 and I04 are degrees of adjective inflection, and the other questions are about verbs.

scores of latent and observed variables. The results show that some observed variables have negative loadings, which indicates that subcategories of inflectional morphology on BATS may not correlate well with each other. We can find the same problem in correlation analysis among the observed variables of INF, as shown on the right side of Figure 3.9. The accuracies for noun plural questions and degrees of adjective inflection do not correlate well. It implies that word embedding may encode the inflectional morphology for nouns and adjectives in different ways, unlike the structure of the BATS dataset. Therefore, we assume that it is the main reason why the INF latent variable is not estimated well in our PLSPM models.

3.5 Summary

In this chapter, we employ the PLSPM method to explain the causal relationship between encoded linguistic knowledge and the accuracy of downstream tasks on word embedding models. The PLSPM method has an advantage in investigating comprehensive relations with causal diagrams suggested by the researcher. We have found that our suggested PLSPM models enable statistical analysis that is hard for correlation analysis, such as verifying the existence of causal relations between intrinsic evaluation and downstream tasks, the explanatory power of intrinsic evaluation for downstream tasks, and the effectiveness of hyperparameters on intrinsic evaluation and downstream tasks. As a result, we have proven causal hypotheses in previous studies that the accuracy of intrinsic evaluation can explain the accuracy of downstream tasks. For example, the

accuracy of downstream tasks about syntactic properties, such as POS-tagging and chunking, can be explained by the accuracy of linguistic knowledge for derivational morphology and lexicography. In this way, we explain the accuracy of 20 downstream tasks with the accuracy of one word analogy dataset representing four linguistic knowledge. Furthermore, Our PLSPM models also provided additional valuable findings, such as the effectiveness of hyperparameters to the accuracy of downstream tasks and the structural problem of inflection knowledge in the BATS dataset.

Camacho-Collados and Navigli (2016a) argued that previous studies on relations between intrinsic evaluation and downstream tasks have salient limitations in terms of generality. We believe that our contribution is to employ a statistical methodology to investigate causal relationships between intrinsic evaluation and downstream tasks to prove them with more generality. However, we only handle basic experimental settings on this issue, such as word embeddings and simple downstream tasks, which can be solved by one dense layer. In practice, downstream tasks of NLP require a more complicated system consisting of multiple modules or involving external resources. Furthermore, word embeddings have been replaced with contextual embeddings, a more strong pretrained language model such as BERT. In the next chapter, we start an empirical analysis as the preparation involving practical settings for downstream tasks into our PLSPM framework.

Chapter 4

Probing the causal relationship between linguistic knowledge and the accuracy of a SFQA system

In this chapter, we investigate the inner working of the existing system for SFQA and examine its robustness. We are interested in whether our PLSPM framework is also applicable both for contextual embeddings and complicated systems for practical downstream tasks. For this purpose, we select SFQA as the target downstream task since this task requires a modularized system and external resources, such as the knowledge base, to be solved. Moreover, we employ the BERT-based system for SFQA proposed by Lukovnikov et al. (2019). Involving 24 BERT models, two evaluation toolkits for BERT, and three SFQA datasets, we conduct PLSPM analysis to investigate the inner working of the BERT-based SFQA system when solving simple factoid questions. As a result, our PLSPM models show that accuracy of the BERT-based system has a significant coefficient with latent variables for surface and syntactic features. It indicates that the BERT-based system for SFQA strongly depends on the surface and syntactic features of the dataset for solving given questions.

4.1 Why apply PLSPM to SFQA systems?

In the previous chapter, we prove that our PLSPM framework can explain the accuracy of 20 downstream tasks by encoded linguistic knowledge on language models. However, analyzed downstream tasks in previous chapters are too simple tasks that can usually be solved by one dense end-to-end layer. Also, we do not consider contextual embeddings, which have become an indispensable tool for NLP recently, such as BERT (Devlin et al., 2019). We thus expand the target of our PLSPM framework involving the more complex downstream task, SFQA.

SFQA is one sub-task of QAKB since this task only handles questions that can be solved with a single fact (Bordes et al., 2015). While it is a simplified version of QAKB, existing systems proposed for SFQA also consist of submodules including entity linking, relation prediction, and evidence integration (Huang et al., 2019, Lukovnikov et al., 2019, Mohammed et al., 2018, Petrochuk and Zettlemoyer, 2018). In addition, submodules of SFQA, including entity linking and relation prediction, should involve external information in Freebase since they need to predict entities or relations in Freebase from a given question. Therefore, SFQA is a more complicated and practical downstream task than 20 downstream tasks we employed in the previous chapter.

Another reason for selecting SFQA is recent arguments for the benchmark dataset of SFQA. Lukovnikov et al. (2019) contended that existing systems for SimpleQuestions already reach the upper bound accuracy of SimpleQuestions. However, the upper bound accuracy of SimpleQuestions is calculated by excluding ambiguous questions for predicting the gold subject and relation. Furthermore, Serban et al. (2016) and Jiang et al. (2019) suggested that questions in SimpleQuestions tend to contain labels of the gold subject and relation compared with other QAKB datasets. It is the reason why we have an interest in investigating the inner working of existing SFQA systems for solving simple factoid questions.

In this chapter, we examine whether we can apply our PLSPM framework to more complicated language models and downstream tasks. By employing our PLSPM framework, we aim to explain the causal relationship between encoded linguistic knowledge and the accuracy of SFQA on BERT. We then inspect whether PLSPM models can provide an informative explanation for the inner working of BERT like our previous study.

4.2 Causal diagram

In this chapter, we investigate the inner working of the existing SFQA system by our proposed PLSPM framework. Our PLSPM analysis assumes that the same language model solves intrinsic evaluations measuring encoded linguistic knowledge on the language model and downstream tasks. Following this assumption, we employ a BERT-based SFQA system (Lukovnikov et al.,

Dataset-Category as a latent variable	Tasks as an observed variable
SentEval-Surface Information (SUR)	Length, Word Content
SentEval-Syntactic Information (SYN)	Tree Depth, Bigram Shift, Top-constituent
SentEval-Semantic Information (SEM)	Tense, Subj Number, Obj Number, Odd Man Out, Coordination Inversion
GLUE-Single-Sentence (SS)	CoLA, SST-2
GLUE-Similarity and Paraphrase (SP)	MRPC, STS-B, ~~QQP~~
GLUE-Inference (IF)	MNLI, QNLI, RTE, ~~WNLI~~
SFQA-Entity Detection (ED)	Entity Detection
SFQA-Entity Linking (EL)	Entity Linking
SFQA-Relation Prediction (RP)	Relation Prediction
SFQA-Evidence Integration (EI)	Evidence Integration

TABLE 4.1: List of tasks used for PLSPM models. Tasks with the strikethrough line are not used in our experiments because of low correlation coefficients.

2019) for our PLSPM framework. Other SFQA systems, such as Mohammed et al. (2018), Petrochuk and Zettlemoyer (2018) and Huang et al. (2019), are not easy to divide the effect of their encoder networks and language models. Meanwhile, the system proposed by (Lukovnikov et al., 2019) consists of a BERT model and a decoder of one dense layer. Therefore, we can follow our assumption for the PLSPM analysis focusing encoded linguistic knowledge on BERT models.

For the PLSPM analysis, causal assumptions for target variables should be expressed as a causal diagram. We follow the traditional assumption that the accuracy of intrinsic evaluation can explain accuracies of downstream tasks like Chapter 3. Referring to previous studies (Hao et al., 2019, Jawahar et al., 2019, Kovaleva et al., 2019, Schneider et al., 2020), we employ SentEval (Conneau and Kiela, 2018) and GLUE (Wang et al., 2019a) as intrinsic evaluations to examine accuracies of encoded linguistic knowledge in BERT. Note that previous studies usually called SentEval and GLUE as "probing tasks". However, we treat them as one toolkit for the intrinsic evaluation in this thesis since we examine encoded linguistic knowledge on language models by employing SentEval and GLUE. Like BATS, SentEval and GLUE also classify their subtasks into linguistic categories. We follow their classification when suggesting latent variables in our causal diagram.

The system proposed by Lukovnikov et al. (2019) consists of four submodules, including entity detection, entity linking, relation prediction, and evidence integration, similar to other QAKB and SFQA systems. In our suggested causal diagrams, we treat each submodule as one latent and observed variable. In other words, the accuracy of each submodule represents the performance for each submodule in causal diagrams. Since each submodule on the system proposed by Lukovnikov et al. (2019) has different characteristics distinguishing each other, we do not bind them together.

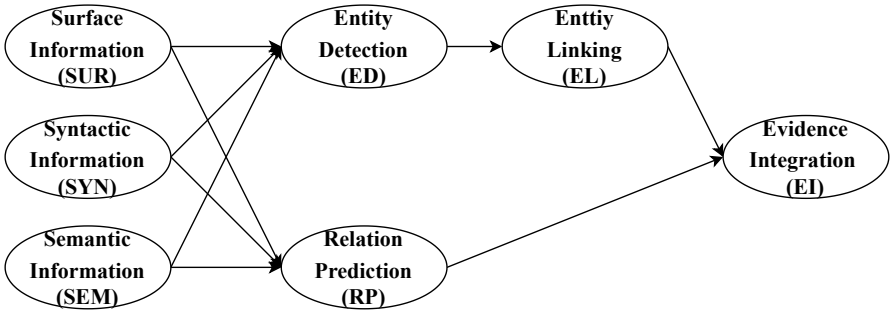

FIGURE 4.1: Causal diagrams for probing the inner working of the BERT-based system involving SentEval. Observed variables are omitted.

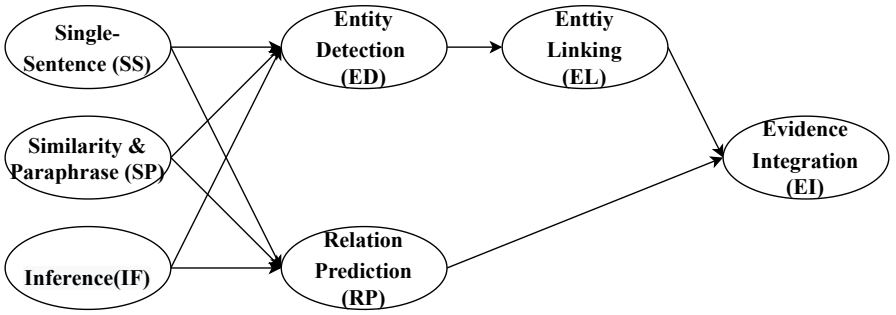

FIGURE 4.2: Causal diagrams for probing the inner working of the BERT-based system involving GLUE. Observed variables are omitted.

Among submodules, only two submodules depend on the machine learning algorithm; entity detection, predicting the span of entity for a given question, and relation prediction, predicting the gold relation for a given question. We thus link encoded linguistic knowledge on language models with only entity detection and relation prediction in our causal diagram. Since the purpose of entity linking is to link the predicted span of a given question with the predicted entity in Freebase, we set the causal relationship between the accuracy of entity detection and the accuracy of entity linking. Evidence integration predicts the most suitable pair of the predicted entity and relation from entity linking and relation prediction results. Therefore, we set causal paths both from accuracies of entity linking and relation prediction to the accuracy of evidence integration. Table 4.1 shows details of employed intrinsic evaluations and submodules of the BERT-based SFQA system.

In this way, we suggest causal diagrams as Figure 4.1 and Figure 4.2, which consists of causal hypotheses among encoded linguistic knowledge and SFQA. Our PLSPM models based on Figure 4.1 and Figure 4.2 aim to estimate structural equations for the below hypotheses.

- Accuracies of intrinsic evaluations, SentEval and GLUE, can explain the accuracy of entity detection and the accuracy of relation prediction.
- The accuracy of entity detection can explain the accuracy of entity linking.

- The accuracy of entity linking and the accuracy of relation prediction can explain the accuracy of evidence integration.

While previous studies have reported the significant effect of training parameters on BERT (Karthikeyan et al., 2020, Turc et al., 2019), we do not include the latent variable for representing training parameters in this experiment. As we will explain in the 4.3, we prepare 24 BERT models for fitting PLSPM models based on suggested causal diagrams. Since the sample size is not large, we do no include the latent variable for training parameters to decrease the parameters of the PLSPM model to be trained.

4.3 Experimental settings

We employ three QA datasets over a knowledge base as our target datasets to investigate the inner working of the BERT-based system. These datasets were selected because they share a common knowledge base, Freebase, and a large portion of each dataset consists of factoid questions, which are the main focus of this chapter. In addition, we intend to avoid that our PLSPM analysis only focuses on a specific dataset, SimpleQuestions. We aim to the inner working of the BERT-based system when solving simple factoid questions generally.

In this chapter, we prepare FreebaseQA (Jiang et al., 2019), SimpleQuestions (Bordes et al., 2015), and WebQSP (Yih et al., 2016). While all of them were proposed for QAKB, they have a variety of differences, such as the size of each dataset, how to create questions, and the required number of facts to be solved. Because we aim to evaluate the behavior of a single model across these three datasets, we perform some preprocessing on each dataset to eliminate those factors. Specifically, from all datasets, we filter questions that do not match the domain of SimpleQuestions; that is, we remove the questions that involve a multi-hop path or multi constraints, such as "What character did Natalie Portman play in Star Wars?" in WebQSP, and questions with entities or relations that are outside of FB2M.

Table 4.2 shows the resulting statistics of each dataset.[1] Following Berant et al. (2013), we take 20% of the training split as the validation split for WebQSP. We abbreviate these three datasets as FBQ, WQ, and SQ, respectively. Table 4.3 summarizes how much of the relations in one dataset (validation split) are unseen (i.e., zero-shot) in another dataset (training split). Since zero-shot prediction is hard for this task (Wu et al., 2019), we use these as a rough estimate of the difficulty of each dataset.

SentEval, GLUE, and the BERT-based system requires BERT models for our causal diagrams. However, training BERT models from pre-training needs high computational costs. Therefore, we

[1]The reason for the decrease in the first step for FreebaseQA is that it contains two-hop questions involving a mediator node in Freebase, which we exclude from the target.

Dataset	Original			Answerable by FB2M		
	Training	Valid	Test	Training	Valid	Test
FreebaseQA	20,358	3,994	3,996	10,427	2,048	2,102
SimpleQuestions	75,910	10,845	21,687	75,895	10,843	21,680
WebQSP	2,478	620	1,639	1,292	323	861

TABLE 4.2: Data statistics after preprocessing (number of examples). We use "Answerable by FB2M" subset in this study.

Training	Validation	# of questions
FBQ	FBQ	71 (3.47%)
	SQ	2,582 (23.87%)
	WQ	52 (16.15%)
SQ	FBQ	137 (6.69%)
	SQ	71 (0.66%)
	WQ	19 (5.90%)
WQ	FBQ	1,068 (52.15%)
	SQ	6,862 (63.45%)
	WQ	26 (8.07%)

TABLE 4.3: Numbers of examples with unseen relations across one training set and one validation set. The number in a bracket denotes a ratio in the validation split. For example, 71 (3.47%) examples in the valid set of FBQ contain relations not appearing in the training set of FBQ.

employ 24 BERT models from Turc et al. (2019) for this experiment. Even though PLSPM allows a lower sample size compared with other structural equation modeling methods (Sanchez, 2013, Tenenhaus et al., 2005), we do not combine SentEval and GLUE into one PLSPM model at once to decrease the parameters of PLSPM models. Furthermore, we respectively estimate PLSPM models for FBQ, SQ, and WQ. Therefore, we prepare 6 PLSPM models in total. Hereinafter, we use the following naming convention for a PLSPM model estimated with accuracies of the specific probing dataset and the BERT-based system for SFQA dataset: "the name of probing dataset"-"the name of the dataset for SFQA". For example, SentEval-FBQ means that this PLSPM model is estimated with the accuracy of SentEval and the accuracy of the BERT-based system for the FBQ dataset.

Note that we implement the BERT-based system by ourselves because the official github repository of Lukovnikov et al. (2019) is expired. For implementation, we follow instructions in the original paper as much as possible, except for the network design of entity detection and relation prediction. In the original paper, they combined two submodules, entity detection and relation prediction, into one classifier for improving accuracies of the proposed system. However, we divide those submodules like other SFQA systems when we implement this system. We intend to investigate respectively inner workings of the BERT-based system predicting the span of the predicted entity and the predicted relation. Hereinafter, we call this system BertQA since we distinguish this system from the suggested system by Lukovnikov et al. (2019).

The classifier for entity detection, relation prediction, SentEval, and GLUE consists of one dense layer in this experiment. We use the fixed seed to avoid random effects of initialization in the training phase. Meanwhile, we allow fine-tuning when solving those tasks since the accuracy of them drops significantly without fine-tuning. Previous studies have reported that encoded linguistic information of the first layer is related to the received information at first (Lin et al., 2019) and encoded linguistic information on each layer has changed gradually when training (Liu et al., 2019a). Following previous studies, we suppose that fine-tuning for only one dense layer may not hurt encoded linguistic knowledge on BERT significantly. This issue is still controversial, for example Merchant et al. (2020) supported our assumption while Singh et al. (2020) and Mosbach et al. (2020) argued that fine-tuning may hurt encoded linguistic knowledge on language models. We leave further examination for this issue as one of our future works.

For observed variables in our PLSPM models, we prepare the results of SentEval, GLUE, and submodules of BertQA with 24 BERT models. Note that we do not conduct any preprocessing for observed variables except for normalization, while they use many kinds of performance indicators, such as Top-n accuracy, F1-score, Matthew's Corr. For our causal diagram, we need the original indicator defined for measuring the performance on each task. Furthermore, we also follow the structure of tasks written by their original paper to composite latent variables, only except for QQP and WNLI in the GLUE dataset. They show low correlation coefficients with other tasks in the same category. We already discuss how the low correlation coefficient among observed variables for the same latent variable affects the explainability of a whole estimated PLSPM model. Therefore, we do not use QQP and WNLI for our experiment.

Why the accuracy of QQP and WNLI shows a low correlation coefficient with the accuracy of other datasets? For this issue, we have the below hypotheses. First, questions on QQP tend to be noisy and duplicated following to previous study and our investigation. Sharma et al. (2019) reported that preprocessing for eliminating non-ASCII character, punctuation, and numbers make the number of vocabulary in QQP about half. They also reported that about 80% of questions on QQP appear more than once, including 158 appearances for one sentence. The above features may be harmful when BERT tries to understand semantic information of a given utterance. Furthermore, we found that sometimes a question on QQP shows too similar sentences for judging their similarity. For example, one sentence pair in QQP is "What are the best books on cosmology?" and "Which is the best book for cosmology?". Since their difference in the utterance is only "What are ... s on" and "Which is ... for", the system can easily judge that they have the same meaning. Therefore, we suppose that the characteristic of QQP makes QQP have a low correlation coefficient with other similarity and paraphrase datasets.

Second, WNLI may be a more complicated and difficult dataset than other datasets in the inference category. According to Levesque et al. (2012) and Kocijan et al. (2020), this dataset requires understanding a highly ambiguous pronoun written in two sentences pair. Since the

Dataset	Final Accuracy	Comparison with upper bound
FBQ	42.29	−41.71
SQ	74.07	−08.93
WQ	64.84	−22.16

TABLE 4.4: Results of BertQA for datasets. The upper bound accuracy of each dataset is calculated referring to Petrochuk and Zettlemoyer (2018).

referent of this pronoun changes by differing in only one or two words, this dataset is also related to coreference resolution. In contrast, other datasets in the inference category, such as MNLI, tend to require only semantic understand and inference to the system. Moreover, the size of WNLI, which contains only 780 questions both for training and test splits, is much smaller than other datasets. Therefore, we suppose that the difficulty and complexity of WNLI make WNLI have a low correlation coefficient with other inference datasets.

4.4 Experiments and PLSPM analysis

Before PLSPM analysis, we test the performance of BertQA for three datasets. Table 4.4 shows the end-to-end accuracy of BertQA for FBQ, SQ, and WQ. While Lukovnikov et al. (2019) reported 77.3% end-to-end accuracy for SimpleQuestions, 74.07% end-to-end accuracy of our implemented BertQA is also acceptable considering the filtered dataset and the change of decoder design in implementation.

The interesting point is that BertQA shows lower accuracies for FBQ and WQ than SQ. Petrochuk and Zettlemoyer (2018) find that the upper bound accuracy of SQ is around 83% due to the inherent ambiguity in the data; e.g., given a question "who wrote gulliver's travels?", there is more than one equally plausible interpretation since there are multiple entities for "guliver's travels" such as the book, TV miniseries, and films, all of which could be compatible with "who wrote . . .?". To test the possibility that lower accuracies on WQ and FBQ are due to even more severe ambiguity in the data, we perform the same analysis on FBQ and WQ, finding that the upper bound accuracy is 86.85% for WQ and 84.16% for FBQ, respectively, which are comparable to SQ. This result rejects the possibility that the upper bound accuracy for these two datasets is low. As a result, BertQA does not reach near the upper bound accuracy of FBQ and WQ. We will investigate this issue further in the next chapter.

Based on the above result, we examine how much accuracies of SentEval can explain accuracies of BertQA by PLSPM analysis. Table 4.5 shows path coefficients of each path in SentEval-FBQ, SentEval-SQ, and SentEval-WQ. When interpreting the PLSPM model in the statistical field, the path coefficient is considered as the explainability of the target path. We do not list the path where the p-value is larger than 0.05. It is considered a meaningful index even though the path coefficient is minus value since a meaningless path is rejected by p-value. As a result, we

FBQ	SUR	SYN	SEM	R^2
Entity detection (ED)	−0.544	+1.060	-	0.787
Relation prediction (RP)	+0.327	+0.405	-	0.880
SQ	SUR	SYN	SEM	R^2
Entity detection (ED)	−0.674	+1.190	-	0.820
Relation prediction (RP)	+0.462	-	-	0.851
WQ	SUR	SYN	SEM	R^2
Entity detection (ED)	−0.590	+0.976	-	0.376
Relation prediction (RP)	+0.418	-	-	0.738

TABLE 4.5: Path coefficient for PLSPM models with SentEval. If p-value of path equation is higher than 0.05, we rejected that path.

FBQ	SS	SP	IF	R^2
Entity detection (ED)	-	-	+0.756	0.890
Relation prediction (RP)	+1.460	-	−0.780	0.868
SQ	SS	SP	IF	R^2
Entity detection (ED)	-	-	+1.090	0.884
Relation prediction (RP)	+1.340	-	−0.918	0.748
WQ	SS	SP	IF	R^2
Entity detection (ED)	-	-	+1.910	0.578
Relation prediction (RP)	+1.400	-	-	0.660

TABLE 4.6: Path coefficient for PLSPM models with GLUE. If p-value of path equation is higher than 0.05, we rejected that path.

can conclude that accuracies of SentEval can explain accuracies of BertQA meaningfully, only except semantic tasks of SentEval. It indicates why BertQA can not overcome the gap between different datasets since the gap of distribution for questions between each dataset is related to surface and syntactic knowledge.

PLSPM models using the GLUE dataset report more difficult results to be interpreted as in Table 4.6. While only the accuracy of inference tasks can explain the accuracy of entity detection with the p-value < 0.05 among GLUE tasks, they also report negative coefficients for explaining the accuracy of relation prediction. For relation prediction, the accuracy of single-sentence tasks, such as CoLA, mainly explain it. One interesting point is that the accuracy of similarity and paraphrase tasks are rejected for explaining any accuracy of BertQA with the p-value > 0.05. Those tasks, such as MRPC, demand to understand semantic knowledge of given sentences for solving questions. It means that encoded semantic knowledge and the ability to understand given sentences are not so helpful to explain accuracies of BertQA, even in our PLSPM model with the GLUE dataset.

For the overall results of our PLSPM models, we found another problem about the gap between each SFQA dataset. Table 4.7 lists GoF indexes of each PLSPM model. The GoF value of the PLSPM model indicates how much this model can explain observed and latent variables.

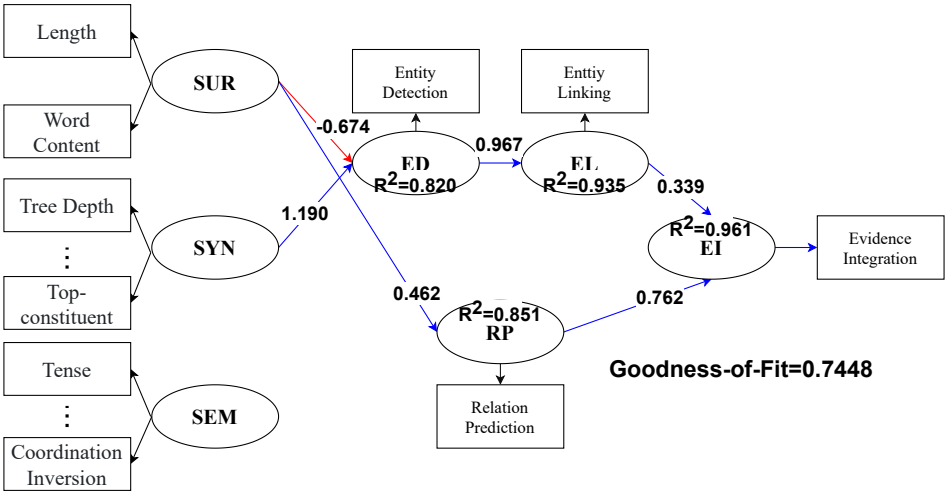

FIGURE 4.3: The estimated PLSPM model by the accuracy of SentEval and SQ

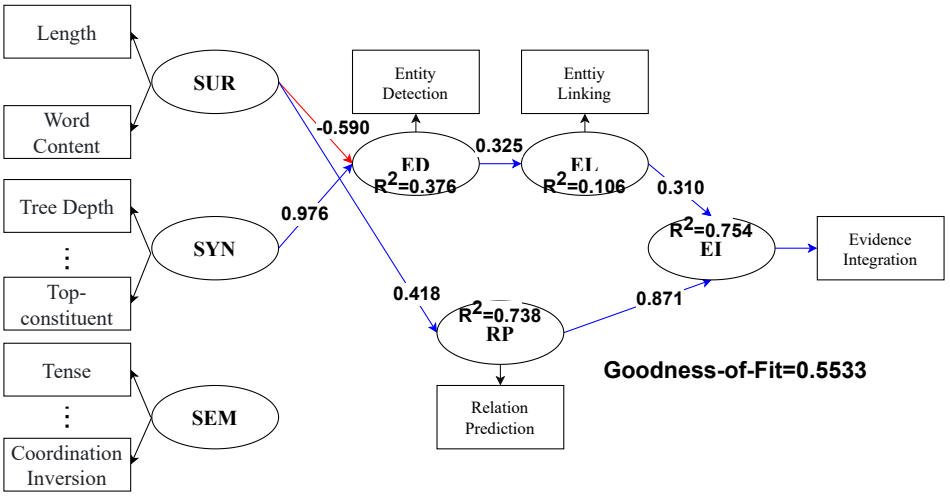

FIGURE 4.4: The estimated PLSPM model by the accuracy of SentEval and WQ

PLSPM model	Goodness-of-Fit
SentEval-FBQ	0.6826
SentEval-SQ	0.7448
SentEval-WQ	0.5533
GLUE-FBQ	0.8218
GLUE-SQ	0.8783
GLUE-WQ	0.6769

TABLE 4.7: Goodness-of-Fit index for each PLSPM models.

Though all PLSPM models share the same causal diagram, the PLSPM models for the WQ dataset reported lower GoF indexes than the PLSPM models for other datasets. In particular, SentEval-WQ shows a lower GoF index than 0.6, which indicates that this model is not well-explained by a given causal diagram. This result implies that WQ is solved with different encoded linguistic knowledge comparing to FBQ and SQ. Figure 4.3 and Figure 4.4 show the significant difference of R^2 and GoF indexes between SentEval-SQ and SentEval-WQ.

Dataset	% of Questions
FBQ	22.17
SQ	46.25
WQ	12.73

TABLE 4.8: Percentage for how many questions contain a term appeared in the label of the gold relation. Note that we examine each validation split of datasets.

4.5 Discussion

4.5.1 The effect of semantic understanding on BertQA

Table 4.5 shows that BertQA heavily depends on the encoded surface and syntactic knowledge. On the other hand, the accuracies of semantic information tasks can not explain the accuracies of BertQA at all. Semantic understanding is demanded to solve questions containing ambiguous expressions, especially for relations such as `people.person.profession`. It indicates that BertQA may be weak for predicting paraphrases or synonyms of the gold relation.

Depending on the surface and syntactic information for solving datasets also indicates that the surface and syntactic features of each dataset affect the accuracy of BertQA. In other words, the high final accuracy of BertQA from SimpleQuestions may be mainly caused by the surface and syntactic features of SimpleQuestions. As we mentioned in Chapter 2.4.3, Serban et al. (2016) and Jiang et al. (2019) mentioned that questions in SimpleQuestions tend to contain the label of the gold subject and relation directly. For example, a given question "name the profession of ..." includes "profession", a piece of the gold relation `people.person.profession`. Table 4.8 shows the result of our investigation for this issue. Following Table 4.8, we find that questions in SQ tend to provide a direct clue for predicting the gold relation than FBQ and WQ. It is one reason why BertQA, the system depending on the surface and syntactic information for solving datasets, reported the better accuracy for SQ than FBQ and WQ.

4.5.2 The effect of specific characteristics on each dataset

As shown in Table 4.7, the Goodness-of-Fit values of PLSPM models for WQ are lower than those of PLSPM models for other datasets. It indicates that the same causal diagram, which assumes that encoded linguistic knowledge can explain accuracies of BertQA, does not match for WQ, unlike other datasets. Table 4.5 and Table 4.6 lists R^2 value of each latent variable in our PLSPM models. R^2 means the explanatory power for each variable. For example, the 0.787 R^2 value for ED means that about 79% of the ED variable can be explained in the PLSPM model. Following R^2 values, we find that accuracies of entity detection (ED) and relation prediction (RP) for WQ are not explained well by encoded linguistic knowledge measured by SentEval and GLUE than ED and RP for other datasets.

Dataset	Average length of entity spans
FBQ	2.10(±1.29)
SQ	2.49(±1.56)
WQ	1.71(±0.77)

TABLE 4.9: Average length of the entity spans for each question of datasets. The value in the bracket means the standard deviation. Note that we examine each validation split of datasets.

As Table 4.9 reports, we find that the length of the gold span is different among each dataset. The longer span of the gold subject means that the gold subject consists of more tokens. It indicates that the surface and syntactic features can be utilized more when solving FBQ and SQ than WQ. Since BertQA does not depend on encoded semantic information following our PLSPM analysis, the length of the gold span directly affects the R^2 value of ED in our PLSPM models. In other words, specific characteristics on each dataset can strongly affect the explanation for accuracies of BertQA.

4.6 Summary

In this chapter, we examine the inner working of the BERT-based system for SFQA. BERT-based systems have reported state-of-the-arts accuracies for a variety of downstream tasks (Devlin et al., 2019), and they also have shown robustness in NLP areas (Devlin et al., 2019, Talmor and Berant, 2019). However, we find that BertQA fails to reach upper bound accuracies of FBQ and WQ like SQ. We conduct PLSPM analysis to investigate the inner working of BertQA when solving FBQ, SQ, and WQ involving 24 BERT models, two intrinsic evaluations. As a result, our experiment reveals that even BERT depends on the surface and syntactic features on each dataset, not the semantic understanding required for general SFQA. It indicates that BertQA is not enough to generalize the gap of distribution among FBQ, SQ, and WQ. Also, we discuss the characteristic of each dataset which may affect accuracies and explainability of BertQA for each dataset.

According to our PLSPM analysis, BertQA depends on the surface and syntactic characteristics intrinsic to each dataset. It indicates that BertQA may lack the robustness and transferability for solving simple factoid questions generally. It should be an interesting question whether other language models, such as GPT series or T5, can overcome the limitation of BERT in solving simple factoid questions. However, it seems a bit difficult because of the below reasons. First, other language models usually share the same structure, transformers, and similar objective functions, such as masked language modeling, with BERT. It indicates that encoded linguistic knowledge on those language models may not be too different. Second, the problem of BERT for solving simple factoid questions is also strongly related to the feature of datasets and the

method used to evaluate this task. Therefore, only changing the language model may not be a good solution for this problem.

Furthermore, our findings in this chapter allows us another question about other existing state-the-of-arts SFQA systems, such as Mohammed et al. (2018) and Huang et al. (2019). We will also explore the above problems with empirical analyses involving five SFQA systems and four SFQA datasets.

Chapter 5

Empirical evaluation of SFQA systems for the robustness and transferability considering linguistic knowledge

In the previous chapter, we investigated the inner working of BertQA, the BERT-based SFQA system, by our proposed PLSPM framework. While BertQA reported near the upper bound accuracy for SimpleQuestions, we find that BertQA depends on the surface and syntactic information for solving simple factoid questions. It indicates the possibility of low robustness and transferability on BertQA. According to our previous study, we examine the robustness and transferability of existing SoTA systems for SFQA involving five systems, which all reported near the upper bound accuracy for SimpleQuestions, and four datasets. We can not employ our PLSPM framework since five systems are based on different language models and hard to distinguish the accuracy for encoded linguistic knowledge from the effect of each encoder network. However, we found that the characteristic of each dataset affects the robustness of each submodule, such as entity detection and relation prediction in the previous chapter. We thus conduct empirical analysis for evaluating the robustness and transferability considering the characteristic of each dataset and submodule. As a result, the success of one dataset, SimpleQuestions, does not transfer to other datasets by all five SFQA systems. Furthermore, we discuss the evaluation method and the source of SFQA datasets considering linguistic knowledge based on the result of the previous chapter.

5.1 Robustness and transferability for SFQA

Sometimes the success for one dataset of a downstream task does not indicate a more general success of that task overall. According to the result of the previous chapter, the success of BertQA, which is the implemented the BERT-based system proposed by Lukovnikov et al. (2019) with minor changes, for SimpleQuestions is the same case. BertQA does not reach upper bound accuracies of FBQ and WQ, filtered datasets of FreebaseQA (Jiang et al., 2019) and WebQSP (Yih et al., 2016). Our PLSPM framework reveals that BertQA mainly depends on the surface and syntactic knowledge for solving simple factoid questions. It indicates that BertQA is a weak system in the robustness, the ability to solve problems generally, and transferability, the ability to apply the success of one dataset to another dataset. Those abilities require semantic understanding to overcome a gap between the distribution of each dataset.

The problem is that BertQA is an extended version of BuboQA (Mohammed et al., 2018), one of the baseline systems for SFQA. Many other SFQA systems (Huang et al., 2019, Petrochuk and Zettlemoyer, 2018, Ture and Jojic, 2017) share a similar structure and approach of submodules with BuboQA and BertQA. While they reported near the upper bound accuracy for SimpleQuestions, the robustness and transferability of those systems have not been evaluated yet. Such robustness evaluation is recently actively studied in other language understanding tasks (Jia and Liang, 2017, McCoy et al., 2019, Naik et al., 2018, Ribeiro et al., 2020) while little effort has been made on question answering over a knowledge base, though, in practice, it would be critical because a practical system has to be robust on actual user queries, which may be outliers in the training data. When Bordes et al. (2015) proposed SimpleQuestions, they expressed their motivation to cover the larger variations for question types, syntactic and lexical distributions for the robustness and transferability of systems. However, now existing SFQA systems may only focus on solving SimpleQuestions.

In this chapter, we examine the robustness and transferability of five SFQA systems involving four datasets. We assume that other existing SoTA systems for SimpleQuestions also have the same limitation: a lack of robustness and transferability since they only depend on the surface and syntactic information. Unfortunately, it is hard to conduct the PLSPM analysis for five SFQA systems. For suggesting a causal diagram handling the accuracy of intrinsic evaluations and downstream tasks like our previous study, we should calculate those accuracies using the same language model. However, existing SoTA systems employed different language models, respectively. In addition, they usually include additional encoder networks that make encoded linguistic knowledge on the employed language model change by the effect of the encoder network. Therefore, we do not conduct PLSPM analysis for examining existing SoTA systems for SFQA.

Instead, we rigorously evaluate existing SFQA systems using different datasets, such as shifting training and test datasets and training on a union of the datasets. We assume that if one system can solve various SFQA datasets with the same level of accuracy for SimpleQuestions, that system has robustness for SFQA. In addition, if one system has transferability for SFQA, then that system training with one dataset should solve a different test split. We examine existing SoTA systems for SimpleQuestions based on the above assumptions.

Meanwhile, previous studies mentioned that questions in SimpleQuestions tend to contain the label of the gold subject and relation directly (Jiang et al., 2019, Serban et al., 2016). In the previous chapter, we also found that SQ has other characteristics to make BertQA depend on surface and syntactic information, such as the longer length of the gold entity span and the more frequent appearance of the label for gold relation than WQ. It indicates that each dataset has different distributions for the gold entity and relation that is the objective of each submodule on the existing SoTA system for SimpleQuestions. Therefore, we also analyze the characteristic of each dataset and submodule in the SFQA system in this chapter.

As we mentioned above, existing SoTA systems share a similar structure and approach of submodules with BertQA. If existing SoTA systems also have limitations in robustness and transferability like BertQA, we need to examine what feature of each dataset and submodule makes existing systems depend on the surface and syntactic information. Our assumption is that; Suppose existing SoTA systems, including BertQA, have the same problem in the robustness and transferability for solving SFQA datasets, and we also can find similar tendencies in the accuracy of their submodules related to the characteristic of each dataset. In that case, we can conclude that existing SoTA systems have identical limitations depending on the surface and syntactic information like BertQA.

5.2 Experimental settings

Comparing to the experiment of the previous chapter, we prepare one additional SFQA dataset and four other SFQA systems. First, we introduce one additional dataset, Free917 (Cai and Yates, 2013). It is selected because this dataset is also based on Freebase, and a large portion of each dataset consists of factoid questions. Since we are interested in the transferability of existing SFQA systems, we also intend to handle another dataset independent of FBQ, SQ, and WQ. We perform the filtering process on Free917 to make the domain and difficulty of Free917 the same with FBQ, SQ, and WQ. This preprocessing is the same as what we conducted in the previous chapter. Because this procedure makes Free917 too small, we use the entire dataset as the test set. Table 5.1 shows the statistics of each dataset, including F917, the filtered Free917.

Dataset	Original			Answerable by FB2M		
	Training	Valid	Test	Training	Valid	Test
Free917	512	129	276	0	0	347
WebQSP	2,478	620	1,639	1,292	323	861
SimpleQuestions	75,910	10,845	21,687	75,895	10,843	21,680
FreebaseQA	20,358	3,994	3,996	10,427	2,048	2,102

TABLE 5.1: Data statistics after preprocessing (number of examples). We use "Answerable by FB2M" subset in this paper. Since Free917 is small, we use the entire dataset as the test set.

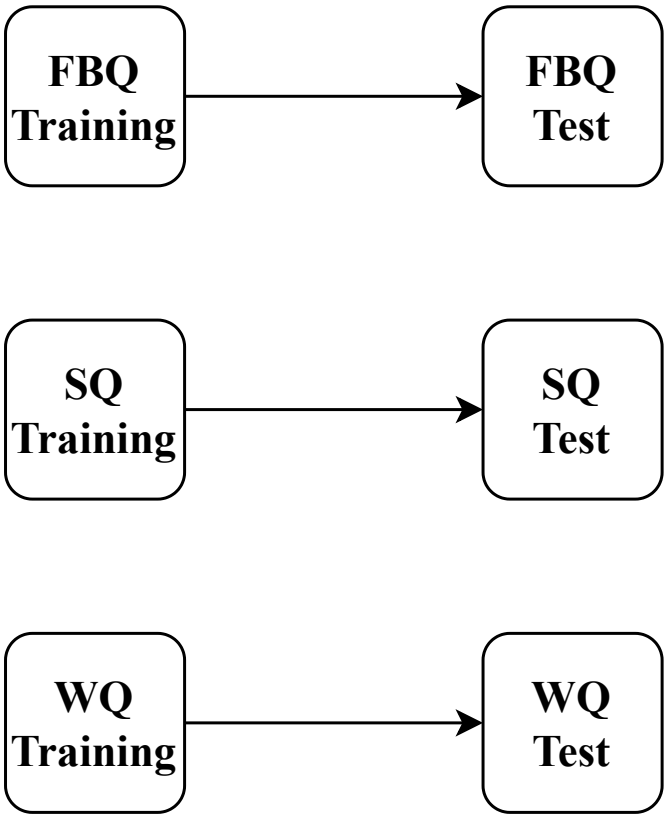

FIGURE 5.1: The experiment with the single dataset setting.

For SFQA systems, we employ BuboQA (Mohammed et al., 2018), HR-BiLSTM (Yu et al., 2017), KBQA-Adpater (Wu et al., 2019), and KEQA (Huang et al., 2019) in addition to BertQA (Lukovnikov et al., 2019). We select those systems considering the state-of-the-arts accuracy for SimpleQuestions, the availability and reproducibility of the system, and a significant difference in submodules, including entity linking and relation prediction. First, BuboQA is employed as the baseline system of SFQA. HR-BiLSTM and KBQA-Adapter are selected for the difference of relation prediction with BuboQA and BertQA. While HR-BiLSTM and KBQA-Adapter depend on external modules for entity linking and evidence integration, they suggested a mapping-based relation prediction different from other systems. Since BuboQA and BertQA treat relation prediction as a classification problem, they can not handle unseen relations in the training split. HR-BiLSTM and KBQA-Adapter have an advantage in predicting unseen relations, which is related to robustness and transferability. KEQA is distinguished from other systems since this

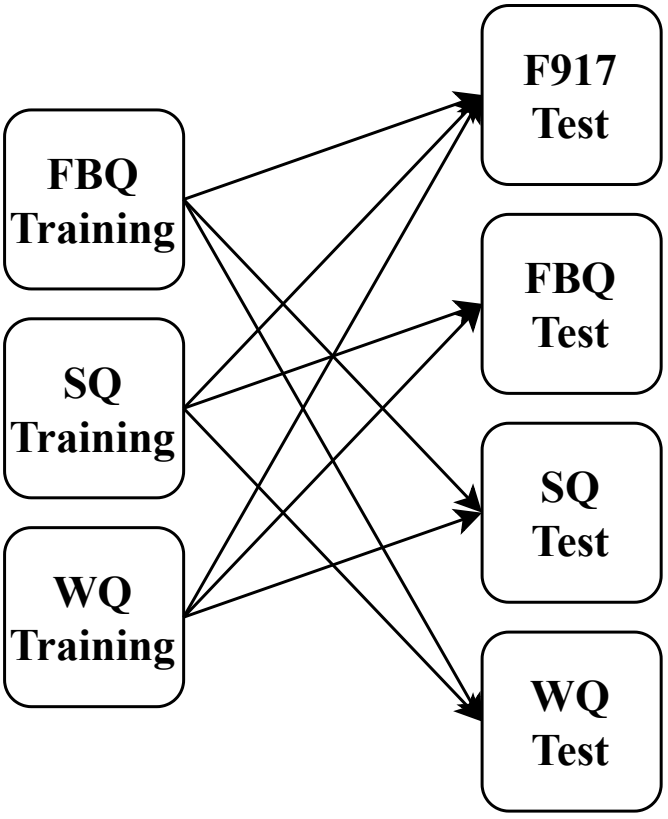

FIGURE 5.2: The experiment with the shifted dataset setting.

system involves both word embeddings and knowledge graph embedding for mapping entities and relations. We investigate the robustness and transferability of employed SFQA systems considering the difference mentioned above.

We employ the best architectures and hyperparameters reported in the paper or a related document for all systems. We set the number of entity linking outputs as 50 and that of relation prediction as 5, which are the default settings for BuboQA. Note that BuboQA, HR-BiLSTM, and KBQA-Adapter share the same entity linking and evidence integration modules of BuboQA. For evaluation, following the standard practice of SimpleQuestions, we evaluate the accuracy of predicted (subject, relation) pairs.

In this chapter, we evaluate different SFQA systems primarily suggested to solve SimpleQuestions across the normalized datasets. First, we experiment with the standard setting of SFQA training and testing the same dataset involving SFQA systems as shown in Figure 5.1. In this experiment, we will examine the robustness of SFQA systems that whether existing SFQA systems can reach near the upper bound accuracy of FBQ and WQ like SQ. Second, we also provide an experiment across two datasets, where we train a model on one dataset and test on another like Figure 5.2. We are interested in this setting because, in a practical scenario, there might be a gap between the distribution of training data, which depends on the way the data was created, and that of test data, which would be actual user queries. For example, the data creation of SimpleQuestions

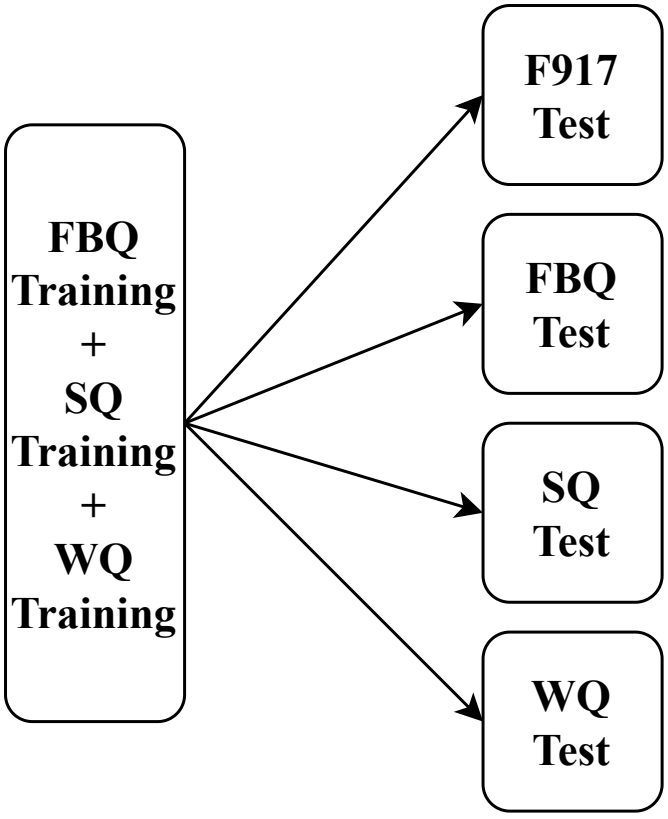

FIGURE 5.3: The experiment with the combined dataset setting.

allows collecting a lot of data efficiently, while the data distribution of WebQuestions may match the distribution in the wild. Hence, this setting is related to evaluate the transferability of existing SFQA systems. Finally, we test the effect of combining datasets with the experiment like Figure 5.3. This setting is inspired by Talmor and Berant (2019), which reported that combining different datasets is helpful for the robustness of the system in reading comprehension. We examine whether it is also applicable for SFQA.

5.3 Experimental results

Table 5.2 summarizes experimental results on the test data. The grey rows correspond to the single dataset settings. Comparing these three rows, the accuracies on FBQ and WQ are consistently lower than SQ for all SFQA systems, suggesting that FBQ and WQ have some data characteristics that cause difficulties for the current models. We inspect in detail further. When evaluated on a different dataset, which we call dataset transfer in the following, the accuracies degrade even more. Note that F917, used as a test-only dataset, reaches relatively high accuracy when trained on SQ. As we discussed in Chapter 2.3.3, SQ and F917 are somewhat similar. It suggests that, as can be expected, the accuracies on this transfer setting are affected by some notion of distance among datasets, and the current models are pretty sensitive to it.

Training	Test	BuboQA	HR	KBQA	KEQA	BertQA
FBQ	F917	17.29	**36.31**	35.73	36.02	23.63
	FBQ	38.25	28.40	28.78	28.73	**42.29**
	SQ	23.77	38.55	39.19	**42.97**	32.07
	WQ	29.10	30.27	31.43	33.18	**38.88**
SQ	F917	40.92	56.20	**59.37**	45.24	47.84
	FBQ	20.08	17.84	18.13	14.03	**24.41**
	SQ	74.81	72.30	72.01	**75.35**	74.07
	WQ	41.79	35.27	36.32	40.40	**44.59**
WQ	F917	12.68	29.97	29.39	**32.85**	14.70
	FBQ	7.94	7.61	8.37	8.90	**10.28**
	SQ	16.46	33.18	35.32	**38.01**	19.61
	WQ	61.23	49.94	49.36	**65.19**	64.84

TABLE 5.2: Comparison of top-1 accuracies across datasets. The bold value denotes the highest accuracy in each row. The grey row correspond to the single dataset setting. The abbreviation HR is HR-BiLSTM, and KBQA is KBQA-Adapter.

Test	BuboQA	HR	KBQA	KEQA	BertQA
F917	43.80(+1.98)	55.33(−0.87)	59.08(−0.29)	46.69(+1.45)	54.47(+6.63)
FBQ	36.01(−2.24)	28.64(+0.24)	26.55(−2.23)	27.07(−1.66)	42.15(−0.14)
SQ	74.18(−0.63)	71.87(−0.43)	71.56(−0.45)	74.89(−0.46)	74.48(+0.41)
WQ	60.65(−0.58)	45.05(−4.89)	46.33(−3.03)	61.35(−3.84)	61.46(−3.37)

TABLE 5.3: The final top-1 accuracies by a single model trained on a union of FBQ, SQ, WQ training set. The number in brackets denotes the difference from the model trained on a single target dataset (in Table 5.2). F917 is compared with the best model (best training data) for each system.

We also experiment with another experimental setting to see the performance of a model trained on the union of the target datasets. It is inspired by the recent success of MultiQA (Talmor and Berant, 2019), which, on reading comprehension, shows that a single model trained on the union of multiple datasets outperforms a model trained specifically on each single dataset. We combine training data of FBQ, SQ, and WQ and train a model on it. We are particularly interested in whether the accuracy of FBQ and WQ improves with the help of statistical cues from other datasets, although we have seen that the transfer from SQ only is complicated. Table 5.3 is the result along with the amount of increase/decrease from a model trained on the single dataset (corresponding to the test data). We can see that SFQA systems can handle each dataset well on average, but in most cases, the scores do not improve from the single dataset baselines.

The above results might be reasonable from our detailed analysis so far. In Chapter 5.4, we will show that FBQ contains more unfaithful questions than other datasets. We suppose that the quality of the dataset mainly affects our experimental results in the case of FBQ. In Chapter 5.5, we will inspect that the main challenge on remaining errors of WQ is in ambiguous and challenging cases of entity linking and relation prediction.

Label	F917	FBQ	SQ	WQ
impossible	1	13	1	4
notsimple	0	15	5	1
badgold	0	12	4	5
multisubj	1	9	0	1
multirel	1	4	2	1
other	0	1	3	0
okay	97	46	85	88

TABLE 5.4: Labeling results on random 100 questions from the validation split for each dataset.

Label	Example	Details
impossible	*In the musical Annie, what is Orphan Annie's dog called?*	There is no identifier for Annie's dog in FB2M.
notsimple	*What is the highest peak on Dartmoor?*	The highest cannot be evaluated in a single triple.
badgold	*Where was Princess Leia raised?*	The gold relation is `place_of_birth`, but Leia was raised elsewhere since infancy.
multisubj	*Who wrote the novels "Berlin Game", "Mexico Set" and "London Match"?*	*Berlin Game*, *Mexico Set*, and *London Match* can all derive the correct answer.
multirel	*Where is South Salt Lake, Utah located?*	Both *location.hud_county_place.county* and *location.location.containedby* can be the correct relation.
other	*What operating system uses ssh file transfer protocol?*	Not operating systems, but sshftp programs use sshftp.

TABLE 5.5: Examples for the labels used in Table 5.4.

5.4 Analysis for SFQA datasets

According to Table 5.2, WQ and FBQ are more challenging than SQ. Understanding the cause of this difficulty is essential because it directly relates to the remaining challenges in solving factoid questions in general. In the previous chapter 4.5, we investigate the characteristic of SQ and WQ to interpret estimated PLSPM models. Since other existing SoTA systems for SimpleQuestions show a similar tendency for robustness and transferability like BertQA, we can assume that the characteristic of each dataset may be one important reason for the cause of this difficulty. However, other factors like the quality or size of each dataset can also be an important reason. Therefore, we test several possibilities to reach an accurate answer.

First, we focus on the quality and size of the dataset. Inspecting datasets, we find that some questions in FBQ are not a factoid question, such as "What is the highest volcano in Africa?", which requires an aggregate operation but the gold subject and relation are just `Africa` and `location.contains`. We suspect that these questions remain in FBQ due to noisy filtering from unrestricted questions, which only assesses the path from a subject to an object with

Dataset	BuboQA	KEQA
SQ (valid)	75.79	76.69
Small-sized SQ (valid)	73.47±1.69	76.27±2.39
WQ (valid)	59.32	66.15

TABLE 5.6: Comparison of end-to-end accuracies (on the validation split) across SQ, small-sized SQ, and WQ. The scores for small-sized SQ are averaged across 10 cases (see body).

little care for additional constraints. The overall quality might be exacerbated by a reliance on non-experts (crowds) for the final assessment.

To quantify how much of the examples are problematic, we randomly sample 100 questions from the validation split on each dataset and categorize them with the labels defined in Table 5.5. Table 5.4 is the result. For this labeling, *impossible*, *notsimple*, and *badgold* labels indicate non-faithful (question, gold label) pairs as in the above example. In contrast, *multisubj* and *multirel* are rather the problems due to the evaluation method because they mean that there are multiple correct labels while the current evaluation only allows a gold one. From Table 5.4, we can see that 40% of questions in FBQ are non-faithful, much higher than the other datasets. From this result, we argue that lower accuracies on FBQ are not due to the actual difficulty of factoid questions but rather due to the undesirable complexity incurred by an inaccurate data creation process. Considering this problem, we will pay little attention to this dataset in the following analysis.

One significant difference between SQ and WQ is the training data size (Table 5.1), with SQ being roughly 60 times larger. Is this data size the primary source of the performance gap seen in Table 5.2? Or, is it due to the inherent complexity of WQ compared with SQ? To answer this question, we compare SQ and WQ eliminating the data size effects by preparing a smaller SQ dataset, which has an equal size as WQ. When sampling data from SQ, we only sample examples with relations that appear in the corresponding split of WQ. Referring to Table 4.3, we also keep the ratio of unseen relations in the validation split as roughly 8%, the same as WQ. We create ten different subsets of SQ and report the average accuracies on them. We evaluate the systems on the validation splits.

In Table 5.6, we summarize the scores of BuboQA and KEQA, which perform better on original SQ and WQ in Table 5.2. We omit BertQA in this examination since BuboQA and BertQA share the same submodules, including entity linking and evidence integration. Interestingly, the accuracies on small-sized SQ are the same level as those of the original dataset. It indicates that the main factor causing the performance gap between WQ and SQ may not be the data size but the complexity or the inherent difficulty of the dataset, which we inspected in the previous chapter.

Dataset	BuboQA-Final	BuboQA-EL	BuboQA-RP
FBQ	38.25	58.28	81.21
SQ	74.81	90.40	95.64
WQ	61.23	78.23	90.92
Dataset	KEQA-Final	KEQA-EL	KEQA-RP
FBQ	28.73	47.62	55.42
SQ	75.35	90.74	94.38
WQ	65.19	82.75	84.97

TABLE 5.7: Comparison of module-level accuracies (R@50 for entity linking (EL) and R@5 for relation prediction (RP)) for BuboQA and KEQA. "Final" denotes end-to-end top-1 accuracies.

5.5 Analysis for submodules of SFQA systems

As we explained in the chapter 2.3.3, existing SoTA systems consist of various submodules, including entity linking, relation prediction, and evidence integration. While we only have compared the final top-1 accuracy, which results from evidence integration, results of entity linking and relation prediction are also important to understand the performance of each SFQA system. We hypothesize that relation prediction is the main bottleneck on WQ since relations tend to be nontrivially verbalized compared with SQ, since we found that SQ tends to contain the label of gold relation directly than WQ in the previous chapter. To test our hypothesis, we need to examine the accuracy of entity linking and relation prediction. Table 5.7 shows in particular for BuboQA that this is not the case. Here, we evaluate the component-wise performance of entity linking (EL) and relation prediction (RP). We evaluate R@50 for EL and R@5 for RP, which are the sizes of candidates in two components of BuboQA.[1] We omit HR-BiLSTM, KBQA-Adapter, and BertQA for this analysis since they share the same entity linking module with BuboQA.

In Table 5.7, we can see that for both systems, EL scores degrade about 10 points from SQ to WQ, which is roughly the same level as decreases in final accuracies. Accuracy of entity linking is critical for both systems because, at the final query generation step, relation candidates are restricted to ones connected to the selected entities. It indicates that if the entity linking performs poorly, that can be a bottleneck of the entire system. KEQA suffers from a more considerable decrease of RP (94.38→84.97) than BuboQA (95.64→90.92), but we conjecture that this can be mainly attributed to the dependence of RP on EL for KEQA (footnote 1).

Inspecting the errors of entity linking by BuboQA, we find a particularly challenging case, specific to WQ, is the superficially ambiguous entities. For example, `Mexico` matches to more than 1,000 different entities in Freebase, according to the inverted index by BuboQA. In the top candidates, we notice that many entities are song and album names. The handling of these

[1] For KEQA, we get the same numbers of candidates for EL and RP closest to the predicted embeddings in the vector space. In this process, we restrict the candidates for RP as ones that are connected to one of the entity candidates, mimicking the final process of the system.

Training	Test	BuboQA-Final	BuboQA-EL	BuboQA-RP
FBQ	F917	17.29	70.32	29.11
	SQ	23.77(−51.04)	71.96(−18.44)	39.79(−55.85)
	WQ	29.10(−32.13)	69.85(−8.38)	59.95(−30.97)
SQ	F917	40.92	85.30	55.04
	FBQ	20.08(−18.17)	48.62(−9.66)	49.00(−32.21)
	WQ	41.79(−19.44)	75.90(−2.33)	78.11(−12.81)
WQ	F917	12.68	65.99	18.44
	FBQ	07.94(−30.31)	35.25(−23.03)	24.79(−56.42)
	SQ	16.46(−58.35)	66.71(−23.69)	25.00(−70.64)
Training	Test	KEQA-Final	KEQA-EL	KEQA-RP
FBQ	F917	36.02	60.81	60.23
	SQ	41.83(−33.52)	69.94(−20.80)	71.79(−22.59)
	WQ	33.18(−32.01)	75.32(−7.43)	62.86(−22.11)
SQ	F917	45.24	69.45	69.45
	FBQ	14.03(−14.70)	34.06(−13.56)	37.73(−17.69)
	WQ	40.40(−24.79)	74.62(−08.13)	67.05(−17.92)
WQ	F917	32.85	59.08	54.47
	FBQ	08.90(−19.83)	36.20(−11.42)	26.07(−29.35)
	SQ	38.01(−37.34)	68.49(−22.25)	65.00(−29.38)

TABLE 5.8: Comparison of module-level accuracies in the dataset transfer setting. Final: end-to-end accuracy; EL: R@50; and RP: R@5. The number in brackets denotes the difference from the non-transfer baseline (Table 5.7). The cells for FBQ are represente d in gray considering the issues in the dataset.

ambiguous entities is challenging for BuboQA since it does not rely on statistical techniques for disambiguation (only the Levinstein distance). In other words, entity linking of BuboQA is lexical pattern-based, not statistical, indicating that additional statistical cues from SQ are not very helpful for saving the problematic cases. It suggests that we need a more sophisticated entity linker exploiting a context for disambiguation. KEQA's approach is promising, but the current system has an opposite problem.

So far, we have seen that the system's performance gaps between two datasets, SQ and WQ, largely come from the gaps in entity linking performance. Can the same explanation hold for the significant gaps with the dataset transfer setting in Table 5.2? To answer this question, Table 5.8 summarizes the submodule accuracies for the transfer setting, on which the numbers in parentheses are degradations from the *non-transfer* setting. For example, R@50 of entity linking on BuboQA drops 2.33 points on WQ when changing training data from WQ to SQ. From the table, we can see that score drops are more severe in relation prediction. We conjecture that entity linking is less affected by transfer because expressions of entities (e.g., the name of a person) are relatively fixed compared with relations across datasets.

Label	Description	Example	Number
WrongString	Entity label in Freebase and written string are different	who created the chinese communist party, communist party of china	11
WrongKB	Freebase has too many or no entities for one label	what continent is mexico located on, mexico	6
WrongDetection	Entity detection is evaluated as correct, but actually it fails	what is the northeast of the united states, united states	3
Total			20

TABLE 5.9: 20 error cases in entity linking for WQ.

5.6 Discussion considering linguistic knowledge

According to the analysis of previous chapters, we find that existing SFQA systems all reported a lack of robustness and transferability for simple factoid questions. We reveal that FBQ contains unfaithful simple factoid questions, which make FBQ unreasonably difficult. However, our analysis investigating the effect of upper bound accuracy of each dataset, faithfulness, and size can not suggest the reason why existing SFQA systems fail with WQ. By the additional analysis for submodules of SFQA systems, we find two tendencies for entity linking and relation prediction. For entity linking, SFQA systems tend to fail more with WQ than SQ in the single dataset setting. Since the drop of the accuracy in entity linking is similar to the drop of the accuracy in the whole SFQA system, we suppose that the bottleneck for lack of robustness is entity linking. Meanwhile, we find that the drop of the accuracy in relation prediction is significantly high in the shifted dataset setting. It indicates that predicting the relation of an unseen or paraphrased dataset may be the main reason for the lack of transferability.

Since no SFQA system success to prove the robustness and transferability for solving simple factoid questions, we need to recall our PLSPM analysis for BertQA in the previous chapter. In Chapter 4.4, we showed that the accuracy of entity detection for WQ is not explained by encoded linguistic knowledge compared with FBQ and SQ. We explained this phenomenon with the following discussion in Chapter 4.5.1; while BertQA heavily depends on the surface and syntactic information, the entity spans containing the surface and syntactic information in WQ are shorter than those in FBQ and SQ. In Chapter 5.5, we already reported a similar issue on entity linking for WQ, the case of `Mexico`. BuboQA fails to predict the correct entity for the span "Mexico" since Freebase contains too many noisy entities for the label "Mexico". We investigated error cases of BertQA, sharing the same entity linking module with BuboQA, for the validation split of WQ to figure out how similar the problem in entity linking is to the situation in entity detection.

We manually labeled 20 error questions, which are evaluated as correct in entity detection but not correct in entity linking, as in Table 5.9. As a result, we find that mainly two problems occur for

Label	BuboQA	HR	KBQA	KEQA	BertQA
relnotfound	**8**	**3**	**7**	**9**	**7**
wrongent	14	13	12	35	5
wrongrel	**23**	**23**	**21**	**31**	**21**
ambient	2	1	1	-	5
ambirel	**29**	**27**	**18**	**24**	**24**
unknown	7	-	-	-	13
other	-	-	-	1	-
Total	83	67	59	100	75

TABLE 5.10: Labeling of errors on examples (in the validation set of WQ), which are missed by changing the training data from WQ to SQ. Bold font denotes the errors on relation prediction.

Label	Example	Datails
relnotfound	*Who was vice president under Lincoln?*	Gold relation `us_president.vice_president` is an unseen relation (not appear in the SQ training split).
wrongent	*What to do with kids in phx az?*	The systems finds a different entity than the correct entity `Phoenix, Arizona`.
wrongrel	*What money is used in England?*	The systems finds a different relation than the correct `location.country.currency_used`.
ambient	*Where were the Chickasaw Indians located?*	Predicted entity is `Chickasaw Nation` while gold entity is `Chickasaw`. Both are OK on Freebase.
ambirel	*Who is Aidan Davis?*	Gold answer is `people.person.profession`, but prediction is `common.topic.notable_types`.
unknown	*Where was the battle of Antietam creek?*	The system outputs nothing by failing to bridge predicted entities and relations.
other	*What is the actual current local time now in uk?*	Freebase cannot answer the current time.

TABLE 5.11: Examples for the labels used in Table 5.10

linking the entity in given questions. First, Freebase links too many entities with a single entity label. For example, when we find an entity with the label "mexico" in the preprocessed index by BuboQA, we obtain 2,830 results. Since the entity linking of BuboQA (and also BertQA) does not have any scoring process to sort ambiguous results, they sometimes can not find the correct entity for a given question within top-*n* results. The second problem is that the string label for the entity in Freebase and an utterance for the gold subject are not identical. For example, the given question in WQ has this string, "communist party of china", but the string label in Freebase is "Chinese communist party" for the same entity. We find that the entity linking of BuboQA fails with the disambiguation for polysemy and paraphrase, which requires semantic understanding, according to Table 5.9. Therefore, we suppose that the problem for entity linking is also related to a lack of semantic understanding in SFQA systems.

To confirm what kinds of questions become hard for relation prediction, we manually analyze

errors on examples from SQ→WQ case in Table 5.8. Not that this analysis is on the validation split. We select up to 100 examples for each system, which are originally solved, but failed when trained on WQ. We categorize the errors according to Table 5.11. If multiple labels would apply, we choose the highest one from the table. Since an entity linking error often accompanies a relation prediction error, we prioritize errors related to entity linking (under the same category). The top priority for *relnotfound* (zero-shot relation prediction) is under the assumption that they are particularly hard for models.

Table 5.10 shows the result. Note that the total numbers are not 100 for some systems because we only consider examples that original models (trained on WQ) answer correctly. We can see that errors related to relation prediction are dominant across systems, which is consistent with Table 5.8. We distinguish two types of relation errors: *wrongrel* means a totally wrong prediction while *ambirel* is a spurious error, for which the predicted relation leads to the correct answer on Freebase, but the current label-based metric penalizes it.

We find that most of this latter case occurs by ambiguities of `people.person.profession` and `common.topic.notable_types`, which are often aliases. For a question "who is . . .?", the gold relation of SQ is often `common.topic.notable_types`, but that is often `people.person.profession` in WQ. It can be seen as a kind of dataset bias, and one way to resolve it is to change the evaluation metric to evaluate the answers, not labels. Under the current metric, this can be seen as an inherent limitation of solving all questions under the dataset transfer setting. While these are spurious, the other half of relation prediction errors are *wrongrel*. We find that these are essentially due to different paraphrasing patterns of a relation across datasets, as we discussed in Chapter 2.3.2, and this result suggests such variation for a relation is the main challenge for the transfer.

Finally, we notice that KEQA contains more entity linking errors (*wrongent*), and in many cases, these errors are distinguished in that they are entirely irrelevant to the target entity. It suggests that the KEQA entity linker would be more affected by a dataset bias, possibly due to not relying on a string match when linking. In other words, although entity linking in KEQA is statistical, KEQA does not exploit useful features from SQ examples to handle WQ, at least regarding entity linking. A better model or a learning method could utilize the data with different distribution in a clever way, but our analysis suggests that current practices do not have such an ability. An interesting future direction is an extension with additional features to consider the surface similarities as in BuboQA, which would lead to more robust generalization.

We also examine the relation `people.person.profession` as the sample relation to investigate the difference of writing pattern between datasets. If the term "profession" appears in the question directly, we annotate the label *directly specify relation*. If a term like "job" or "work" similar to "profession" appears in the question, we use the label *indirectly specify relation*, and if there is no term similar to "profession" in the question, *paraphrasing* is used for annotation. Table 5.12

Dataset	Label	Example	Number
SQ	directly specify relation	name the **profession** of peter heller.	84
	indirectly specify relation	what **job** does jamie hewlett have?	14
	paraphrasing	what does dan osborn do for a living?	25
	total		123
WQ	directly specify relation	-	-
	indirectly specify relation	what **job** does bill rancic have?	3
	paraphrasing	who is henry david thoreau?	20
	total		23

TABLE 5.12: Question patterns for relation *people.person.profession* in SQ and WQ. We note that all questions in this table are sampled from the validation split of each dataset.

shows the result of our annotation on the validation split of SQ and WQ. In SQ, about 80% of questions for `people.person.profession` contains the term indicating the relation directly or indirectly, while only 13% of questions do in WQ. This result indicates that relation prediction of WQ demands more complex linguistic knowledge than the surface and syntactic understanding, unlike SQ.

As we discuss, WQ has different characteristics for entities and relations comparing to other datasets, especially SQ. One main reason for this phenomenon is the difference in the method of generating questions in the dataset. For WebQuestions, the source of WQ, the question was generated from Google Suggest API, naturally written user queries (Berant et al., 2013). On the other hand, the question in SimpleQuestions, the source of SQ, was written artificially by crowd-workers with the suggested fact (Bordes et al., 2015). It indicates that the difference in creating each dataset is the reason for the gap of distribution among datasets. Moreover, it also suggests that a lack of semantic understanding in BertQA and possibly existing SFQA systems should be an essential factor of the failure for relation prediction in the shifted dataset setting.

According to our analysis, the bottleneck both in entity linking and relation prediction of existing SFQA systems is related to a lack of semantic understanding. We suppose that the evaluation method of simple factoid question answering task, matching the predicted subject and relation with gold data (Bordes et al., 2015), is one reason for this problem. Especially in SimpleQuestions, the subject and the relation can be extracted from the question without semantic understanding since questions usually contain labels of the subject and the relation (Serban et al., 2016). It means that the evaluation method makes existing SFQA models concentrate on the surface and syntactic features.

Furthermore, the possibility of multiple correct facts for given questions can be another problem with this evaluation method. For example, when existing SFQA system predicted `people.person.profession` for a given question and it can find correct answer from Freebase, but traditional evaluation method may reject this prediction if the gold relation in dataset is

Train	Test	Match Accuracy	Reachability Accuracy
FBQ	F917	23.63	27.38
	FBQ	42.29	51.47
	SQ	32.07	36.37
	WQ	38.88	40.75
SQ	F917	47.84	51.30
	FBQ	24.41	32.57
	SQ	74.07	78.05
	WQ	44.59	44.70
WQ	F917	14.70	15.85
	FBQ	10.28	12.89
	SQ	19.61	21.97
	WQ	64.84	62.40
Average		36.43	39.64

TABLE 5.13: Result of BertQA for QAKB datasets. Match accuracy is calculated by checking whether predicted subject and relation are same with gold data. Reachability accuracy is calculated by checking whether predicted subject and relation can reach to the gold object.

`common.topic.notable_type`. After this, we call the traditional evaluation method *match accuracy* since the criteria of this evaluation is based on matching the correct subject and relation from a given question.

To examine how much the evaluation method affects the accuracy, we conduct additional experiments. We employ an older evaluation method (Berant and Liang, 2014, Berant et al., 2013) considering an object, the answer to a given question in the QAKB task, to overcome the limitation of the match accuracy. Originally, evidence integration combines predicted subjects and predicted relations for a given input question and evaluates its result with the gold fact provided by the dataset. Here, our employed evaluation method compares the predicted object derived by a predicted subject and relation with the gold object in the dataset. Moreover, we extend this method to entity linking and relation prediction. In particular, we aggregate all facts from FB2M to automatically examine whether each submodule's predicted result can reach the gold object or not. We call this employed evaluation method *reachability accuracy* since the criteria of this evaluation are based on the reachability to the correct answer in a knowledge base.

Table 5.13 shows the comparison between match and reachability accuracy for the experimental results. We employ BertQA in this experiment since we also conduct PLSPM analysis further. As a result, BertQA still does not reach the upper bound accuracies of each dataset, even evaluating the reachability accuracy. However, BertQA achieves higher accuracies on average with the reachability accuracy. It means that many entities and relations, which can reach the correct answer in a knowledge base, had been scored as wrong predictions with the previous evaluation method. Therefore, it indicates that the previous evaluation method may make BertQA not consider semantic information, such as understanding paraphrasing or synonyms.

	GoF using match accuracy	GoF using reachability accuracy
SentEval-FBQ	0.6826	0.7095
SentEval-SQ	0.7448	0.7323
SentEval-WQ	0.5533	0.5891
GLUE-FBQ	0.8218	0.8673
GLUE-SQ	0.8783	0.8721
GLUE-WQ	0.6769	0.7127

TABLE 5.14: Comparison of Goodness-of-Fit (GoF) indexes between evaluation methods.

Model	Variable	Using match acc.	Using reachability acc.
SentEval-FBQ	ED	0.787	0.797
	EL	0.345	0.758
	RP	0.880	0.713
	QA	0.988	0.979
GLUE-FBQ	ED	0.890	0.887
	EL	0.345	0.758
	RP	0.868	0.816
	QA	0.988	0.979
SentEval-SQ	ED	0.820	0.820
	EL	0.935	0.836
	RP	0.851	0.841
	QA	0.961	0.953
GLUE-SQ	ED	0.884	0.885
	EL	0.935	0.836
	RP	0.748	0.805
	QA	0.961	0.953
SentEval-WQ	ED	0.376	0.406
	EL	0.106	0.041
	RP	0.738	0.875
	QA	0.754	0.923
GLUE-WQ	ED	0.578	0.581
	EL	0.106	0.041
	RP	0.660	0.778
	QA	0.754	0.923

TABLE 5.15: R^2 for each variable in our PLSPM models. The higher R^2 value means higher explainablilty for target variable.

We also estimate PLSPM models with reachability accuracies of BertQA. Note that we did not change our causal diagrams suggested in Chapter 4.2 since we only employ the reachability accuracy of BertQA as observed variables. New PLSPM models still reject structural equations between probing tasks for semantic information and BertQA. However, new PLSPM models reported a higher Goodness-of-Fit value for all datasets on average than PLSPM models with match accuracies as in Table 5.14. In particular, Table 5.15 shows that accuracies of relation prediction and evidence integration for WQ are explained better in the new PLSPM model. Since WQ demands more semantic understanding than other datasets, as we discussed, we suppose that the reachability accuracy is better to reflect the semantic understanding of the SFQA system.

5.7 Summary

Through several experiments, we have shown that although the system performance on SimpleQuestions dataset is getting better and close to the upper bound, that does not indicate a more general success of SFQA overall. The leading cause of this mismatch is that, as we have seen, there is often an inverse relationship between the *ease* of data collection and *naturalness* of collected questions. Including BertQA, the SFQA system employing a strong pretrained language model, existing SFQA systems all fail to be robust on the questions outside of the distribution of the training data. We examine this problem in the aspect of the dataset and submodule considering linguistic knowledge according to the result of our previous study. Also, we discuss the source of each dataset and the evaluation method of simple factoid question answering, which are essential to understand why existing SFQA systems tend to depend on the surface and syntactic features.

We suppose there are many directions toward general simple question answering or question answering over a knowledge base. For example, we can refer to the decision of the robustness in other disciplines tackling question answering, such as information retrieval. With the preprocessing way, improving the quality of the dataset and distributionally robust optimization can generalize biased datasets (Delage and Ye, 2010, Oren et al., 2019). A recent study (Gu et al., 2020) has suggested new datasets considering distributions and difficulties of questions. Although those approaches are promising, we suggest reconsideration for the evaluation method on simple factoid question answering is also demanded. For instance, changing an objective function from subject and relation to object may improve semantic understanding of the QA system for given questions. We hope that our findings can suggest a more robust system for simple factoid question answering in future work.

Chapter 6

Conclusion

In this thesis, we mainly suggest the method and usefulness of the analysis for explaining the causal relationship between encoded linguistic knowledge and the accuracy of downstream tasks on language models. First, we presented a statistical framework for evaluating and understanding the causal relationship between encoded linguistic knowledge and the accuracy of downstream tasks on language models by employing PLSPM, one method of SEM. Our experiments found that various linguistic knowledge can causally explain the performance of downstream tasks, such as morphology and semantics. We then investigated a variety of language models and downstream tasks to ensure that our suggested framework can produce acceptable and valuable results for improving existing systems for downstream tasks of NLP. Our study is broadly divided into two parts; a suggestion of the PLSPM framework and an application of the PLSPM framework.

For the first part, we tested our PLSPM framework involving 600 language models, one intrinsic evaluation, and 20 downstream tasks. Following traditional assumptions for the causal relationship between encoded linguistic knowledge and the performance for solving downstream tasks, we drew causal diagrams involving intrinsic evaluations and downstream tasks. We then estimated PLSPM models based on 600 samples and our causal diagrams. Our PLSPM models reported valuable results with more informative indexes than previous studies, such as R^2, structural coefficient, and Goodness-of-Fit values. For example, we found that linguistic knowledge for morphology and semantics can affect the accuracy of downstream tasks, consistent with the assumption of previous studies. Our PLSPM models also suggested applicable insights for improving the quality of intrinsic evaluations, such as the structural design of intrinsic evaluations.

However, we only considered downstream tasks that can be solved by only one dense end-to-end layer without any external knowledge. Since practical downstream tasks of NLP tend to require more complex networks or external resources, we extend our study to apply our PLSPM framework to a more complicated downstream task, SFQA. We involve BERT, a representative pretrained language model, three SFQA datasets, and two intrinsic evaluations in our PLSPM framework. As a result, our PLSPM framework finds that the BertQA strongly depends on the surface and syntactic features for solving simple factoid questions. It indicates that the success of BertQA for SimpleQuestions, the benchmark dataset of SFQA, may owe to the characteristics of SimpleQuestions. Based on our findings, we empirically examine whether other SFQA systems can generally solve simple factoid questions. As a result of our experiments involving four datasets and five existing systems, we find that existing systems, which reported state-of-the-art accuracy for the benchmark dataset of this task, show a lack of robustness and transferability of other datasets. We suggest other responsible factors for this problem, such as the source and evlauation method of each dataset, by additional examinations considering the result of our PLSPM analysis.

In the next part, we describe our contributions and future works of this thesis.

6.1 Contributions

Previous approaches for explaining the accuracy of language models tend to depend on the observation with few samples or simple correlation analysis using intrinsic evaluations. Consequently, they lack statistical verifications for robustness. In other words, they have not considered external factors for their conclusion, such as the case of different models, hyperparameters for training models, and the compound effect of multiple linguistic knowledge. Those limitations are the main reasons why we explore whether our proposed evaluation framework can overcome them by employing the statistical method, PLSPM. In Chapter 3 and Chapter 4, we prove that our PLSPM framework can explain the causal relationship between encoded linguistic knowledge and the accuracy of downstream tasks on language models considering a lot of samples and external factors, including non-metric variables. In Chapter 3, we show that linguistic knowledge about morphology and semantics can explain the accuracy of various downstream tasks. In Chapter 4, we reveal that the accuracy for SFQA datasets can be explained by the accuracy of intrinsic evaluations for surface and syntactic knowledge. Note that we can not conclude that our PLSPM framework produces a completely general conclusion yet. Since the causal diagram of each PLSPM model is only validated by prepared samples by the researcher. However, our PLSPM framework suggests a more general insight for understanding and explaining the accuracy of language models involving forgotten variables in previous approaches.

One reason for evaluating language models should be to find the problem of the evaluated language model and a way to resolve that problem. We also examine whether our PLSPM framework can produce informative and valuable information to improve target language models by evaluating them. In Chapter 4 and Chapter 5, we show that PLSPM can provide valuable clues for finding a reason for problems that a downstream task encounters. The empirical analysis, such as error analyses conducted in Chapter 5, is sometimes hard to figure out the causal reason why the problem occurs when an existing system solves this task. Since our proposed framework explains the accuracy of a downstream task with encoded linguistic knowledge on language models, it can suggest whether encoded linguistic knowledge is utilized for the target downstream task on language models or not. With the PLSPM analysis, a researcher can start from "why our system does not depend on one specific linguistic knowledge?" when improving existing systems for the target downstream task. We present a concrete example of how we can incorporate the result of PLSPM models and further suggestions for making existing systems better in Chapter 5. Therefore, we believe that our proposed framework has some contributions for both academics and applications.

6.2 Future works

We present that our proposed PLSPM framework works successfully with a variety of language models and downstream tasks in this thesis. This study is the first step to suggest a statistical framework for explaining the accuracy of language models based on linguistic knowledge to the best of our knowledge. Therefore, we can extend our study to a variety of future works to examine the inner working of language models, existing systems for downstream tasks, and open questions for interpreting language models.

For example, many other language models and downstream tasks which we have not applied to our proposed framework exist in the NLP field. The easiest way to extend this study is to apply the PLSPM evaluation to other language models or downstream tasks. More advanced language models than BERT, such as RoBERTa (Liu et al., 2019b) or GPT-3 (Brown et al., 2020), can be a good target. Comparing a variety of proposed systems for solving the same downstream task is also a promising topic for the interpretability of existing NLP systems. Also, we note that linguistic knowledge handled in this study is limited in existing intrinsic evaluations. Existing datasets for the intrinsic evaluation are also limited in language selection since most of them have been proposed only for English. Fortunately, researchers have proposed new datasets for the cross-linguistic evaluations such as jBATS (Karpinska et al., 2018) and LINSPECTOR (Eichler et al., 2019) in recent. We are also interested in the cross-linguistic evaluation for the same system and task involving those datasets.

In this study, we only discuss one downstream task of NLP, simple factoid question answering. However, we can apply our PLSPM evaluation to other tasks of NLP to explain any result of language models on those tasks. In Chapter 3 and Chapter 4, we use the accuracy as the effect to be predicted in our PLSPM models. If we select machine translation as a downstream task and choose the BLEU score as the effect to be predicted in our PLSPM model, our PLSPM models will explore the causal relationship between encoded linguistic knowledge and the BLEU score. In this way, we can expand our PLSPM evaluation to various downstream tasks in NLP, if the result or performance of a downstream task can be measured in a mathematical way.

Neural network based language models, an indispensable tool for recent NLP field, still have unresolved questions for understanding them, such as how encoded linguistic knowledge changes during pre-training and fine-tuning (Merchant et al., 2020, Mosbach et al., 2020, Singh et al., 2020) and what does the language model train before and after the double decent phenomenon (Belkin et al., 2019, Nakkiran et al., 2019). Our suggested framework can be one way to investigate those questions by preparing enough samples of language models and intrinsic evaluations. Including previous discussions, we believe that we have many rooms on NLP systems employing our proposal for further analysis. We hope that our study will contribute to helping people to understand the inner working of language models regardless of various factors, such as inner structures, downstream tasks, and target languages.

Bibliography

Y. Adi, E. Kermany, Y. Belinkov, O. Lavi, and Y. Goldberg. Fine-grained analysis of sentence embeddings using auxiliary prediction tasks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017. URL https://openreview.net/forum?id=BJh6Ztuxl.

E. Agirre, E. Alfonseca, K. Hall, J. Kravalová, M. Pasca, and A. Soroa. A study on similarity and relatedness using distributional and wordnet-based approaches. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 19–27, 2009.

E. Agirre, M. Diab, D. Cer, and A. Gonzalez-Agirre. Semeval-2012 task 6: A pilot on semantic textual similarity. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, pages 385–393. Association for Computational Linguistics, 2012.

E. Agirre, D. Cer, M. Diab, A. Gonzalez-Agirre, and W. Guo. * sem 2013 shared task: Semantic textual similarity. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*, volume 1, pages 32–43, 2013.

E. Agirre, C. Banea, C. Cardie, D. Cer, M. Diab, A. Gonzalez-Agirre, W. Guo, R. Mihalcea, G. Rigau, and J. Wiebe. Semeval-2014 task 10: Multilingual semantic textual similarity. In *Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014)*, pages 81–91, 2014.

E. Agirre, C. Banea, C. Cardie, D. Cer, M. Diab, A. Gonzalez-Agirre, W. Guo, I. Lopez-Gazpio, M. Maritxalar, R. Mihalcea, et al. Semeval-2015 task 2: Semantic textual similarity, english, spanish and pilot on interpretability. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pages 252–263, 2015.

E. Agirre, C. Banea, D. Cer, M. Diab, A. Gonzalez-Agirre, R. Mihalcea, G. Rigau, and J. Wiebe. Semeval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation.

In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 497–511, 2016.

S. Akter, J. D'Ambra, and P. Ray. An evaluation of PLS based complex models: the roles of power analysis, predictive relevance and gof index. In *A Renaissance of Information Technology for Sustainability and Global Competitiveness. 17th Americas Conference on Information Systems, AMCIS 2011, Detroit, Michigan, USA, August 4-8 2011*, 2011. URL http://aisel.aisnet.org/amcis2011_submissions/151.

A. Almuhareb. *Attributes in lexical acquisition*. PhD thesis, University of Essex, 2006.

A. Bakarov. A survey of word embeddings evaluation methods. *arXiv preprint arXiv:1801.09536*, 2018.

R. Bar Haim, I. Dagan, B. Dolan, L. Ferro, D. Giampiccolo, B. Magnini, and I. Szpektor. The second PASCAL recognising textual entailment challenge. 2006.

M. Baroni, S. Evert, and A. Lenci. Bridging the gap between semantic theory and computational simulations: Proceedings of the esslli workshop on distributional lexical semantics. *Hamburg, Germany: FOLLI*, 2008.

M. Baroni, B. Murphy, E. Barbu, and M. Poesio. Strudel: A distributional semantic model based on property and types. *Cognitive Science*, 34(2):222–254, 2010.

M. Baroni, G. Dinu, and G. Kruszewski. Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 238–247, Baltimore, Maryland, June 2014. Association for Computational Linguistics. doi: 10.3115/v1/P14-1023. URL https://www.aclweb.org/anthology/P14-1023.

M. Batchkarov, T. Kober, J. Reffin, J. Weeds, and D. Weir. A critique of word similarity as a method for evaluating distributional semantic models. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 7–12. Association for Computational Linguistics, 2016a.

M. Batchkarov, T. Kober, J. Reffin, J. Weeds, and D. Weir. A critique of word similarity as a method for evaluating distributional semantic models. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 7–12, 2016b.

M. Belkin, D. Hsu, S. Ma, and S. Mandal. Reconciling modern machine learning practice and the bias-variance trade-off, 2019.

Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin. A neural probabilistic language model. *The journal of machine learning research*, 3:1137–1155, 2003.

L. Bentivogli, I. Dagan, H. T. Dang, D. Giampiccolo, and B. Magnini. The fifth PASCAL recognizing textual entailment challenge. 2009.

J. Berant and P. Liang. Semantic parsing via paraphrasing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1415–1425, Baltimore, Maryland, June 2014. Association for Computational Linguistics. doi: 10.3115/v1/P14-1133. URL https://www.aclweb.org/anthology/P14-1133.

J. Berant, A. Chou, R. Frostig, and P. Liang. Semantic parsing on Freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1533–1544, Seattle, Washington, USA, Oct. 2013. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D13-1160.

J. Bjerva, B. Plank, and J. Bos. Semantic tagging with deep residual networks. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3531–3541, 2016.

P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017. ISSN 2307-387X.

K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250, 2008.

A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko. Translating embeddings for modeling multi-relational data. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 2787–2795. Curran Associates, Inc., 2013. URL http://papers.nips.cc/paper/5071-translating-embeddings-for-modeling-multi-relational-data.pdf.

A. Bordes, N. Usunier, S. Chopra, and J. Weston. Large-scale simple question answering with memory networks. *ArXiv*, abs/1506.02075, 2015.

D. Brickley, R. V. Guha, and A. Layman. Resource description framework (rdf) schema specification. 1999.

T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

E. Bruni, N.-K. Tran, and M. Baroni. Multimodal distributional semantics. *Journal of artificial intelligence research*, 49:1–47, 2014.

Q. Cai and A. Yates. Large-scale semantic parsing via schema matching and lexicon extension. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 423–433, Sofia, Bulgaria, Aug. 2013. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P13-1042.

J. Camacho-Collados and R. Navigli. Find the word that does not belong: A framework for an intrinsic evaluation of word vector representations. In *Proceedings of the ACL Workshop on Evaluating Vector Space Representations for NLP*, pages 43–50, Berlin, Germany, 2016a.

J. Camacho-Collados and R. Navigli. Find the word that does not belong: A framework for an intrinsic evaluation of word vector representations. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 43–50, 2016b.

D. Cer, M. Diab, E. Agirre, I. Lopez-Gazpio, and L. Specia. Semeval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation. *arXiv preprint arXiv:1708.00055*, 2017.

N. Chakraborty, D. Lukovnikov, G. Maheshwari, P. Trivedi, J. Lehmann, and A. Fischer. Introduction to neural network based approaches for question answering over knowledge graphs. *arXiv preprint arXiv:1907.09361*, 2019.

B. Chiu, A. Korhonen, and S. Pyysalo. Intrinsic evaluation of word vectors fails to predict extrinsic performance. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 1–6, 2016.

A. Conneau and D. Kiela. Senteval: An evaluation toolkit for universal sentence representations. *arXiv preprint arXiv:1803.05449*, 2018.

A. Conneau, D. Kiela, H. Schwenk, L. Barrault, and A. Bordes. Supervised learning of universal sentence representations from natural language inference data, 2018.

W. W. W. Consortium et al. Sparql 1.1 overview. 2013.

L. J. Cronbach. Coefficient alpha and the internal structure of tests. *psychometrika*, 16(3): 297–334, 1951.

I. Dagan, O. Glickman, and B. Magnini. The PASCAL recognising textual entailment challenge. In *Machine learning challenges. evaluating predictive uncertainty, visual object classification, and recognising tectual entailment*, pages 177–190. Springer, 2006.

A. M. Dai and Q. V. Le. Semi-supervised sequence learning. In *Proceedings of the 28th International Conference on Neural Information Processing Systems-Volume 2*, pages 3079–3087, 2015.

M.-C. de Marneffe and C. D. Manning. The Stanford typed dependencies representation. In *Coling 2008: Proceedings of the workshop on Cross-Framework and Cross-Domain Parser Evaluation*, pages 1–8, Manchester, UK, Aug. 2008. Coling 2008 Organizing Committee. URL https://www.aclweb.org/anthology/W08-1301.

E. Delage and Y. Ye. Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Operations research*, 58(3):595–612, 2010.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. 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, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://www.aclweb.org/anthology/N19-1423.

W. R. Dillon and M. Goldstein. *Multivariate analysis: methods and applications*. J. Wiley, 1984.

B. Dolan, C. Quirk, and C. Brockett. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceedings of the 20th international conference on Computational Linguistics*, page 350. Association for Computational Linguistics, 2004.

W. B. Dolan and C. Brockett. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the International Workshop on Paraphrasing*, 2005.

M. Eichler, G. G. Şahin, and I. Gurevych. LINSPECTOR WEB: A multilingual probing suite for word representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*, pages 127–132, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-3022. URL https://www.aclweb.org/anthology/D19-3022.

A. Ettinger and T. Linzen. Evaluating vector space models using human semantic priming results. In *Proceedings of the 1st workshop on evaluating vector-space representations for NLP*, pages 72–77, 2016.

M. Faruqui, Y. Tsvetkov, P. Rastogi, and C. Dyer. Problems with evaluation of word embeddings using word similarity tasks. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 30–35, 2016.

J. Ficler and Y. Goldberg. Coordination annotation extension in the penn tree bank. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 834–842, 2016.

L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin. Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*, pages 406–414, 2001.

J. Ganitkevitch, B. Van Durme, and C. Callison-Burch. Ppdb: The paraphrase database. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 758–764, 2013.

D. Giampiccolo, B. Magnini, I. Dagan, and B. Dolan. The third PASCAL recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL workshop on textual entailment and paraphrasing*, pages 1–9. Association for Computational Linguistics, 2007.

A. Gladkova and A. Drozd. Intrinsic evaluations of word embeddings: What can we do better? In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 36–42, 2016.

A. Gladkova, A. Drozd, and S. Matsuoka. Analogy-based detection of morphological and semantic relations with word embeddings: What works and what doesn't. In *Proceedings of the NAACL-HLT SRW*, pages 47–54, San Diego, California, June 12-17, 2016, 2016. ACL. doi: 10.18653/v1/N16-2002. URL https://www.aclweb.org/anthology/N/N16/N16-2002.pdf.

Y. Gu, S. Kase, M. Vanni, B. Sadler, P. Liang, X. Yan, and Y. Su. Beyond i.i.d.: Three levels of generalization for question answering on knowledge bases, 2020.

M. U. Gutmann and A. Hyvärinen. Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. *Journal of Machine Learning Research*, 13(2), 2012.

X. Han, Z. Liu, and M. Sun. Neural knowledge acquisition via mutual attention between knowledge graph and text. In S. A. McIlraith and K. Q. Weinberger, editors, *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 4832–4839. AAAI Press, 2018. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16691.

Y. Hao, L. Dong, F. Wei, and K. Xu. Visualizing and understanding the effectiveness of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4143–4152, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1424. URL https://www.aclweb.org/anthology/D19-1424.

Z. S. Harris. Distributional structure. *Word*, 10(2-3):146–162, 1954.

I. Hendrickx, S. N. Kim, Z. Kozareva, P. Nakov, D. Ó. Séaghdha, S. Padó, M. Pennacchiotti, L. Romano, and S. Szpakowicz. Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 33–38, 2010.

J. Henseler, T. Dijkstra, M. Sarstedt, C. Ringle, A. Diamantopoulos, D. Straub, D. Jr, J. Hair, T. Hult, and R. Calantone. Common beliefs and reality about pls: Comments on rönkkö & evermann (2013). *Organizational Research Methods*, 17:182–209, 04 2014. doi: 10.1177/1094428114526928.

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

G. E. Hinton et al. Learning distributed representations of concepts. In *Proceedings of the eighth annual conference of the cognitive science society*, volume 1, page 12. Amherst, MA, 1986.

J. Hockenmaier and M. Steedman. Ccgbank: a corpus of ccg derivations and dependency structures extracted from the penn treebank. *Computational Linguistics*, 33(3):355–396, 2007.

P. M. Htut, J. Phang, S. Bordia, and S. R. Bowman. Do attention heads in bert track syntactic dependencies?, 2019.

X. Huang, J. Zhang, D. Li, and P. Li. Knowledge graph embedding based question answering. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, WSDM '19, pages 105–113, New York, NY, USA, 2019. ACM. ISBN 978-1-4503-5940-5. doi: 10.1145/3289600.3290956. URL http://doi.acm.org/10.1145/3289600.3290956.

G. Jawahar, B. Sagot, and D. Seddah. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1356. URL https://www.aclweb.org/anthology/P19-1356.

R. Jia and P. Liang. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1215. URL https://www.aclweb.org/anthology/D17-1215.

K. Jiang, D. Wu, and H. Jiang. FreebaseQA: A new factoid QA data set matching trivia-style question-answer pairs with Freebase. 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 318–323, Minneapolis, Minnesota,

June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1028. URL https://www.aclweb.org/anthology/N19-1028.

K. G. Jöreskog. A general method for analysis of covariance structures. *Biometrika*, 57(2): 239–251, 1970.

M. Joshi, E. Choi, D. S. Weld, and L. Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada, July 2017. Association for Computational Linguistics.

M. Karpinska, B. Li, A. Rogers, and A. Drozd. Subcharacter information in Japanese embeddings: When is it worth it? In *Proceedings of the Workshop on the Relevance of Linguistic Structure in Neural Architectures for NLP*, pages 28–37, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-2905. URL https://www.aclweb.org/anthology/W18-2905.

K. Karthikeyan, Z. Wang, S. Mayhew, and D. Roth. Cross-lingual ability of multilingual bert: An empirical study. *ArXiv*, abs/1912.07840, 2020.

R. Kate and R. Mooney. Using string-kernels for learning semantic parsers. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 913–920, 2006.

V. Kocijan, T. Lukasiewicz, E. Davis, G. Marcus, and L. Morgenstern. A review of winograd schema challenge datasets and approaches. *arXiv preprint arXiv:2004.13831*, 2020.

D. Koller and N. Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

O. Kovaleva, A. Romanov, A. Rogers, and A. Rumshisky. Revealing the dark secrets of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4365–4374, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1445. URL https://www.aclweb.org/anthology/D19-1445.

S. Lai, K. Liu, S. He, and J. Zhao. How to generate a good word embedding. *IEEE Intelligent Systems*, 31(6):5–14, 2016.

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

J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. Mendes, S. Hellmann, M. Morsey, P. Van Kleef, S. Auer, and C. Bizer. Dbpedia - a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web Journal*, 6, 01 2014. doi: 10.3233/SW-140134.

J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. Van Kleef, S. Auer, et al. Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic web*, 6(2):167–195, 2015.

H. Levesque, E. Davis, and L. Morgenstern. The winograd schema challenge. In *Thirteenth International Conference on the Principles of Knowledge Representation and Reasoning*. Citeseer, 2012.

H. J. Levesque, E. Davis, and L. Morgenstern. The Winograd schema challenge. In *AAAI Spring Symposium: Logical Formalizations of Commonsense Reasoning*, volume 46, page 47, 2011.

O. Levy and Y. Goldberg. Linguistic regularities in sparse and explicit word representations. In *Proceedings of the eighteenth conference on computational natural language learning*, pages 171–180, 2014.

O. Levy, Y. Goldberg, and I. Dagan. Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3: 211–225, 2015.

X. Li and D. Roth. Learning question classifiers: the role of semantic information. *Natural Language Engineering*, 12(3):229–249, 2006.

P. Liang. Lambda dependency-based compositional semantics. *arXiv preprint arXiv:1309.4408*, 2013.

Y. Lin, Y. C. Tan, and R. Frank. Open sesame: Getting inside BERT's linguistic knowledge. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 241–253, Florence, Italy, Aug. 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-4825. URL https://www.aclweb.org/anthology/W19-4825.

N. F. Liu, M. Gardner, Y. Belinkov, M. E. Peters, and N. A. Smith. Linguistic knowledge and transferability of contextual representations. 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 1073–1094, Minneapolis, Minnesota, June 2019a. Association for Computational Linguistics. doi: 10.18653/v1/N19-1112. URL https://www.aclweb.org/anthology/N19-1112.

Q. Liu, M. J. Kusner, and P. Blunsom. A survey on contextual embeddings. *arXiv preprint arXiv:2003.07278*, 2020.

Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019b.

V. Lopez, V. Uren, M. Sabou, and E. Motta. Is question answering fit for the semantic web?: a survey. *Semantic web*, 2(2):125–155, 2011.

D. Lukovnikov, A. Fischer, and J. Lehmann. Pretrained transformers for simple question answering over knowledge graphs. In *The Semantic Web - ISWC 2019 - 18th International Semantic Web Conference, Auckland, New Zealand, October 26-30, 2019, Proceedings, Part I*, pages 470–486, 2019. doi: 10.1007/978-3-030-30793-6_27. URL https://doi.org/10.1007/978-3-030-30793-6_27.

M. Marcus, B. Santorini, and M. A. Marcinkiewicz. Building a large annotated corpus of english: The penn treebank. *Computational Linguistics*, 19(2):313–330, 1993.

M. Marelli, L. Bentivogli, M. Baroni, R. Bernardi, S. Menini, and R. Zamparelli. Semeval-2014 task 1: Evaluation of compositional distributional semantic models on full sentences through semantic relatedness and textual entailment. In *Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014)*, pages 1–8, 2014.

T. McCoy, E. Pavlick, and T. Linzen. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1334. URL https://www.aclweb.org/anthology/P19-1334.

K. McRae, M. J. Spivey-Knowlton, and M. K. Tanenhaus. Modeling the influence of thematic fit (and other constraints) in on-line sentence comprehension. *Journal of Memory and Language*, 38(3):283–312, 1998.

P. N. Mendes, M. Jakob, A. García-Silva, and C. Bizer. Dbpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th international conference on semantic systems*, pages 1–8, 2011.

A. Merchant, E. Rahimtoroghi, E. Pavlick, and I. Tenney. What happens to BERT embeddings during fine-tuning? In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 33–44, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.blackboxnlp-1.4. URL https://www.aclweb.org/anthology/2020.blackboxnlp-1.4.

T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013a.

T. Mikolov, W.-t. Yih, and G. Zweig. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751, 2013b.

G. A. Miller and W. G. Charles. Contextual correlates of semantic similarity. *Language and cognitive processes*, 6(1):1–28, 1991.

S. Mohammed, P. Shi, and J. Lin. Strong baselines for simple question answering over knowledge graphs with and without neural networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 291–296, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-2047. URL https://www.aclweb.org/anthology/N18-2047.

F. Morin and Y. Bengio. Hierarchical probabilistic neural network language model. In *Aistats*, volume 5, pages 246–252. Citeseer, 2005.

M. Mosbach, A. Khokhlova, M. A. Hedderich, and D. Klakow. On the interplay between fine-tuning and sentence-level probing for linguistic knowledge in pre-trained transformers. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 68–82, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.blackboxnlp-1.7. URL https://www.aclweb.org/anthology/2020.blackboxnlp-1.7.

A. Naik, A. Ravichander, N. Sadeh, C. Rose, and G. Neubig. Stress test evaluation for natural language inference. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2340–2353, Santa Fe, New Mexico, USA, Aug. 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/C18-1198.

P. Nakkiran, G. Kaplun, Y. Bansal, T. Yang, B. Barak, and I. Sutskever. Deep double descent: Where bigger models and more data hurt, 2019.

N. Nayak, G. Angeli, and C. D. Manning. Evaluating word embeddings using a representative suite of practical tasks. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 19–23, 2016.

S. Oepen, M. Kuhlmann, Y. Miyao, D. Zeman, S. Cinková, D. Flickinger, J. Hajic, and Z. Uresova. Semeval 2015 task 18: Broad-coverage semantic dependency parsing. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 915–926, 2015.

Y. Oren, S. Sagawa, T. Hashimoto, and P. Liang. Distributionally robust language modeling. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4227–4237, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1432. URL https://www.aclweb.org/anthology/D19-1432.

S. Padó and M. Lapata. Dependency-based construction of semantic space models. *Computational Linguistics*, 33(2):161–199, 2007.

H. Paulheim. Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic web*, 8(3):489–508, 2017.

J. Pearl. *Causality*. Cambridge university press, 2009.

M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*, 2018.

M. Petrochuk and L. Zettlemoyer. SimpleQuestions nearly solved: A new upperbound and baseline approach. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 554–558, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1051. URL https://www.aclweb.org/anthology/D18-1051.

S. Pradhan, A. Moschitti, N. Xue, O. Uryupina, and Y. Zhang. Conll-2012 shared task: Modeling multilingual unrestricted coreference in ontonotes. In *Joint Conference on EMNLP and CoNLL-Shared Task*, pages 1–40, 2012.

A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving language understanding by generative pre-training. *URL https://s3-us-west-2. amazonaws. com/openai-assets/research-covers/languageunsupervised/language understanding paper. pdf*, 2018.

A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 1:8, 2019.

P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of EMNLP*, pages 2383–2392. Association for Computational Linguistics, 2016.

P. Ramachandran, P. J. Liu, and Q. Le. Unsupervised pretraining for sequence to sequence learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 383–391, 2017.

S. Reddy, M. Lapata, and M. Steedman. Large-scale semantic parsing without question-answer pairs. *Transactions of the Association for Computational Linguistics*, 2:377–392, 2014.

M. T. Ribeiro, T. Wu, C. Guestrin, and S. Singh. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.442. URL https://www.aclweb.org/anthology/2020.acl-main.442.

A. Rogers, S. Hosur Ananthakrishna, and A. Rumshisky. What's in your embedding, and how it predicts task performance. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2690–2703, Santa Fe, New Mexico, USA, Aug. 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/C18-1228.

A. Rogers, O. Kovaleva, and A. Rumshisky. A primer in bertology: What we know about how bert works. *arXiv preprint arXiv:2002.12327*, 2020.

H. Rubenstein and J. B. Goodenough. Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633, 1965.

R. Rudinger, A. S. White, and B. Van Durme. Neural models of factuality. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 731–744, 2018.

G. Russolillo. Non-metric partial least squares. *Electronic Journal of Statistics*, 6:1641–1669, 2012.

G. Salton, A. Wong, and C.-S. Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

G. Sanchez. Pls path modeling with r. *Berkeley: Trowchez Editions*, 383:2013, 2013.

E. F. T. K. Sang and S. Buchholz. Introduction to the conll-2000 shared task chunking. In *Fourth Conference on Computational Natural Language Learning and the Second Learning Language in Logic Workshop*, 2000.

T. K. Sang and F. Erik. Introduction to the conll-2002 shared task: language-independent named entity recognition. In *proceedings of the 6th conference on Natural language learning-Volume 20*, pages 1–4. Association for Computational Linguistics, 2002.

V. Sanh, L. Debut, J. Chaumond, and T. Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

T. Schnabel, I. Labutov, D. Mimno, and T. Joachims. Evaluation methods for unsupervised word embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 298–307, 2015.

N. Schneider, J. D. Hwang, V. Srikumar, J. Prange, A. Blodgett, S. Moeller, A. Stern, A. Shalev, and O. Abend. Comprehensive supersense disambiguation of english prepositions and possessives. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 185–196, 2018.

R. Schneider, T. Oberhauser, P. Grundmann, F. A. Gers, A. Loeser, and S. Staab. Is language modeling enough? evaluating effective embedding combinations. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4739–4748, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL https://www.aclweb.org/anthology/2020.lrec-1.583.

S. Schuster and C. D. Manning. Enhanced English Universal Dependencies: An improved representation for natural language understanding tasks. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2371–2378, Portorož, Slovenia, May 2016. European Language Resources Association (ELRA). URL https://www.aclweb.org/anthology/L16-1376.

I. V. Serban, A. García-Durán, C. Gulcehre, S. Ahn, S. Chandar, A. Courville, and Y. Bengio. Generating factoid questions with recurrent neural networks: The 30M factoid question-answer corpus. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 588–598, Berlin, Germany, Aug. 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1056. URL https://www.aclweb.org/anthology/P16-1056.

L. Sharma, L. Graesser, N. Nangia, and U. Evci. Natural language understanding with the quora question pairs dataset, 2019.

X. Shi, I. Padhi, and K. Knight. Does string-based neural mt learn source syntax? In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1526–1534, 2016.

N. Silveira, T. Dozat, M.-C. de Marneffe, S. Bowman, M. Connor, J. Bauer, and C. D. Manning. A gold standard dependency corpus for english. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2897–2904, 2014.

J. Singh, J. Wallat, and A. Anand. BERTnesia: Investigating the capture and forgetting of knowledge in BERT. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 174–183, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.blackboxnlp-1.17. URL https://www.aclweb.org/anthology/2020.blackboxnlp-1.17.

R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Ng, and C. Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642, 2013.

A. Søgaard. Evaluating word embeddings with fmri and eye-tracking. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 116–121, 2016.

Y. Su, H. Sun, B. Sadler, M. Srivatsa, I. Gür, Z. Yan, and X. Yan. On generating characteristic-rich question sets for qa evaluation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 562–572, 2016.

A. Talmor and J. Berant. The web as a knowledge-base for answering complex questions. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 641–651, New Orleans, Louisiana, June 2018a. Association for Computational Linguistics. doi: 10.18653/v1/N18-1059. URL https://www.aclweb.org/anthology/N18-1059.

A. Talmor and J. Berant. The web as a knowledge-base for answering complex questions. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 641–651, 2018b.

A. Talmor and J. Berant. MultiQA: An empirical investigation of generalization and transfer in reading comprehension. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4911–4921, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1485. URL https://www.aclweb.org/anthology/P19-1485.

A. Teichert, A. Poliak, B. Van Durme, and M. Gormley. Semantic proto-role labeling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

M. Tenenhaus, V. E. Vinzi, Y.-M. Chatelin, and C. Lauro. Pls path modeling. *Computational statistics & data analysis*, 48(1):159–205, 2005.

I. Tenney, D. Das, and E. Pavlick. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy, July 2019a. Association for Computational Linguistics. doi: 10.18653/v1/P19-1452. URL https://www.aclweb.org/anthology/P19-1452.

I. Tenney, P. Xia, B. Chen, A. Wang, A. Poliak, R. T. McCoy, N. Kim, B. Van Durme, S. Bowman, D. Das, et al. What do you learn from context? probing for sentence structure in contextualized word representations. In *7th International Conference on Learning Representations, ICLR 2019*, 2019b.

K. Toutanova, D. Klein, C. D. Manning, and Y. Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 conference of the North American chapter of the association for computational linguistics on human language technology-volume 1*, pages 173–180. Association for computational Linguistics, 2003.

P. Trivedi, G. Maheshwari, M. Dubey, and J. Lehmann. Lc-quad: A corpus for complex question answering over knowledge graphs. In C. d'Amato, M. Fernandez, V. Tamma, F. Lecue, P. Cudré-Mauroux, J. Sequeda, C. Lange, and J. Heflin, editors, *The Semantic Web – ISWC 2017*, pages 210–218, Cham, 2017. Springer International Publishing. ISBN 978-3-319-68204-4.

I. Turc, M.-W. Chang, K. Lee, and K. Toutanova. Well-read students learn better: On the importance of pre-training compact models. *arXiv preprint arXiv:1908.08962*, 2019.

F. Ture and O. Jojic. No need to pay attention: Simple recurrent neural networks work! In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2866–2872, Copenhagen, Denmark, Sept. 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1307. URL https://www.aclweb.org/anthology/D17-1307.

J. Turian, L. Ratinov, and Y. Bengio. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 384–394, 2010.

P. D. Turney and P. Pantel. From frequency to meaning: Vector space models of semantics. *Journal of artificial intelligence research*, 37:141–188, 2010.

C. Unger, C. Forascu, V. Lopez, A.-C. N. Ngomo, E. Cabrio, P. Cimiano, and S. Walter. Question answering over linked data (qald-4). In *Working Notes for CLEF 2014 Conference*, 2014.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 6000–6010, 2017.

A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. 2019a. In the Proceedings of ICLR.

B. Wang, A. Wang, F. Chen, Y. Wang, and C.-C. J. Kuo. Evaluating word embedding models: methods and experimental results. *APSIPA Transactions on Signal and Information Processing*, 8:e19, 2019b. doi: 10.1017/ATSIP.2019.12.

S. Wang and C. D. Manning. Baselines and bigrams: Simple, good sentiment and topic classification. In *Proceedings of the 50th annual meeting of the association for computational*

linguistics: Short papers-volume 2, pages 90–94. Association for Computational Linguistics, 2012.

Y. Wang, J. Berant, and P. Liang. Building a semantic parser overnight. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1332–1342, Beijing, China, July 2015. Association for Computational Linguistics. doi: 10.3115/v1/P15-1129. URL https://www.aclweb.org/anthology/P15-1129.

A. Warstadt, A. Singh, and S. R. Bowman. Neural network acceptability judgments. *arXiv preprint 1805.12471*, 2018.

A. Warstadt, Y. Cao, I. Grosu, W. Peng, H. Blix, Y. Nie, A. Alsop, S. Bordia, H. Liu, A. Parrish, S.-F. Wang, J. Phang, A. Mohananey, P. M. Htut, P. Jeretic, and S. R. Bowman. Investigating BERT's knowledge of language: Five analysis methods with NPIs. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2877–2887, Hong Kong, China, Nov. 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1286. URL https://www.aclweb.org/anthology/D19-1286.

R. Weischedel, M. Palmer, M. Marcus, E. Hovy, S. Pradhan, L. Ramshaw, N. Xue, A. Taylor, J. Kaufman, M. Franchini, et al. Ontonotes release 5.0 ldc2013t19. *Linguistic Data Consortium, Philadelphia, PA*, 23, 2013.

A. Williams, N. Nangia, and S. R. Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of NAACL-HLT*, 2018.

H. Wold. Soft modeling: the basic design and some extensions. *Systems under indirect observation*, 2:343, 1982.

S. Wright. Correlation and causation. *Journal of agricultural research*, 20(7):557–585, 1921.

P. Wu, S. Huang, R. Weng, Z. Zheng, J. Zhang, X. Yan, and J. Chen. Learning representation mapping for relation detection in knowledge base question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6130–6139, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1616. URL https://www.aclweb.org/anthology/P19-1616.

Y. Yang and M.-W. Chang. S-mart: Novel tree-based structured learning algorithms applied to tweet entity linking. *arXiv preprint arXiv:1609.08075*, 2016.

H. Yannakoudakis, T. Briscoe, and B. Medlock. A new dataset and method for automatically grading esol texts. In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*, pages 180–189, 2011.

W.-t. Yih, M.-W. Chang, X. He, and J. Gao. Semantic parsing via staged query graph generation: Question answering with knowledge base. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1321–1331, Beijing, China, July 2015. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P15-1128.

W.-t. Yih, M. Richardson, C. Meek, M.-W. Chang, and J. Suh. The value of semantic parse labeling for knowledge base question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 201–206, Berlin, Germany, Aug. 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-2033. URL https://www.aclweb.org/anthology/P16-2033.

W. Yin, M. Yu, B. Xiang, B. Zhou, and H. Schütze. Simple question answering by attentive convolutional neural network. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1746–1756, Osaka, Japan, Dec. 2016. The COLING 2016 Organizing Committee. URL https://www.aclweb.org/anthology/C16-1164.

M. Yu, W. Yin, K. S. Hasan, C. dos Santos, B. Xiang, and B. Zhou. Improved neural relation detection for knowledge base question answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 571–581, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1053. URL https://www.aclweb.org/anthology/P17-1053.