DG Embeddings: the unsupervised definition embeddings learned from dictionary and glossary to gloss context words of Cloze task⋆

Xiaodong Liu[a,*], Rafal Rzepka[b] and Kenji Araki[b]

[a]Language Media Lab, Graduate School of Information Science and Technology, Hokkaido University, Kita 8, Nishi 5, Kita-ku, Sapporo, 060-0808, Hokkaido, Japan

[b]Language Media Lab, Faculty of Information Science and Technology, Hokkaido University, Kita 8, Nishi 5, Kita-ku, Sapporo, 060-0808, Hokkaido, Japan

ARTICLE INFO

Keywords:
Unsupervised definition embeddings
Semantic features of glosses
Context words
Auto-encoding models
Natural language processing

ABSTRACT

For both humans and machines to acquire vocabulary, it is effective to learn words from context while using dictionaries as an auxiliary tool. It has been shown in previous linguistic studies that for humans, glossing either target words to be learned or words comprising context is an effective approach. For machines, however, previous NLP studies are mainly focused on the former. In this paper, we investigate the potentiality of context words-glossed setting. During pre-training BERT, to infuse context words with semantic features of glosses, we propose DG embeddings — the unsupervised definition embeddings learned from dictionaries and glossaries. To employ unsupervised learning is inspired by a real-world scenario of dictionary use called headword search. This can also prevent a technical duplicate from happening, as learning words from context is already based on auto-encoding models with self-supervised learning. BERT-base is used for evaluation, and we refer to BERT-base with DG embeddings as DG-BERT. According to our experimental results, compared to the vanilla BERT, DG-BERT shows the following strengths: faster pre-training convergence, noticeable improvements on various downstream tasks, a better grasp of figurative semantics, more accurate self-attention for collocation of phrases, and higher sensitivity to context words for target-word predictions in psycholinguistic diagnostics.

1. Introduction

To acquire vocabulary, it is effective to learn words from context while using dictionaries as an auxiliary tool. For humans, it has been shown in the previous linguistic studies (Knight, 1994; AbuSeileek, 2011; Jung, 2016; Wege, 2018) that using the auxiliary tool on either target words to be learned from context (target words-glossed) or words comprising context (context words-glossed) is an effective approach. In Natural Language Processing (NLP) for machines, previous studies are mainly focused on the former, and thus context words-glossed explorations remain sparse. In this paper, we investigate the latter by proposing a simple yet effective approach to infusing context words with semantic features of glosses for the MLM objective of BERT (Devlin, Chang, Lee and Toutanova, 2019), which is considered as the Cloze task (Taylor, 1953) for machines to learn words from bidirectional context words.

In this section, we first concisely introduce the background of previous target words-glossed settings in Section 1.1. Then we move on to the introduction of our study regarding the linguistic motivation and proposed approach in Section 1.2. The reason why our definition embeddings are created by unsupervised learning is introduced in Section 1.3. Finally, we summarize the research goals and contributions of this paper in Section 1.4.

1.1. Target words-glossed pre-training

As suggested in applied linguistics (Knight, 1994; Wege, 2018), when encountering unknown words, people using a dictionary while guessing their meanings from their context may not only learn more words immediately but also recall them better and be able to use them.

In NLP, a previously proposed Dict-BERT (Yu, Zhu, Fang, Yu, Wang, Xu, Zeng and Jiang, 2022) matches the human learning process: to append definitions in dictionaries to input sequences of BERT (Devlin et al., 2019) for masked rare words during pre-training. Dict-BERT appears to acquire vocabulary more effectively than BERT, which is reflected by the improvements on standard downstream tasks (Wang, Singh, Michael, Hill, Levy and Bowman, 2019a) and domain-specific tasks after domain adaptive pre-training (Gururangan, Marasović, Swayamdipta, Lo, Beltagy, Downey and Smith, 2020). In other target words-glossed pre-training like the early representative ERNIE (Zhang, Han, Liu, Jiang, Sun and Liu, 2019b) and KnowBERT (Peters, Neumann, Logan, Schwartz, Joshi, Singh and Smith, 2019), instead of dictionary resources, they use Knowledge Representations (KR) (Bordes, Usunier, Garcia-Duran, Weston and Yakhnenko, 2013; Balazevic, Allen and Hospedales, 2019) learned from structured knowledge graph (Baker, Fillmore and Lowe, 1998; Speer, Chin and Havasi, 2017) to fuse the knowledge conveyed by KR into hidden states of target words. Due to different vector spaces, to address Heterogeneous Information Fusion (HIF) between

*Corresponding author
xiaodongliu@ist.hokudai.ac.jp (X. Liu); rzepka@ist.hokudai.ac.jp (R. Rzepka); araki@ist.hokudai.ac.jp (K. Araki)
ORCID(s): 0009-0004-6648-2302 (X. Liu)

KR and hidden states, separate self-attention is usually performed before fusion. Their better acquisition of vocabulary leads to the improvements on knowledge-driven tasks.

These studies belong to the category of Knowledge-enhanced Pre-trained Language Models (K-PLMs). When it comes to domain-specific corpora, target words-glossed pre-training is intrinsically advantageous, because the corpora contain denser knowledge entities than open domain corpora like Wikipedia. The sub-categorical difference is that ERNIE and KnowBERT are in the set of explicit fusion (external KR is structurally connected with BERT layers) while Dict-BERT belongs to implicit fusion (external semantic knowledge is merged into the input sequence of BERT).

There exist other K-PLMs than target words-glossed pre-training ones. For example, in K-BERT (Weijie Liu, 2020) and a recent DictBERT (Chen, Li, Xu, Yan, Zhang and Zhang, 2022), KR and definition embeddings do not participate in the MLM objective of BERT, which is only used in fine-tuning. Target words-glossed fine-tuning is also employed by some studies (Huang, Sun, Qiu and Huang, 2019; Blevins and Zettlemoyer, 2020) for word sense disambiguation tasks. For another example, in CoLAKE (Sun, Shao, Qiu, Guo, Hu, Huang and Zhang, 2020) and KEPLER (Wang, Gao, Zhu, Zhang, Liu, Li and Tang, 2021), learning KR is integrated into the pre-training process, which is more of regularization than an auxiliary tool. Using glosses for regularization is also adopted by a study called GR-BERT (Lin, An, Wu and Ma, 2022) for lexical substitution and unsupervised semantic textual similarity tasks. Despite leveraging glosses for target words in different manners, their strengths further reflect the fact that external semantic information can assist BERT in acquiring better word senses.

1.2. Context words-glossed pre-training

It is a common linguistic phenomenon that when humans try to approximate the meaning of an unknown word given its context, a good approximation is difficult to reach if some words in the context are also unknown. The phenomenon suggests that it is a vital prerequisite to understand contextual clues. An example is demonstrated in Figure 1. As shown in the previous linguistic studies (AbuSeileek, 2011; Jung, 2016), language learners tend to acquire vocabulary better after reading comprehension (incorporating the task of guessing word sense from context) if some unintelligible context words (e.g., domain-specific terminologies, idiomatic expressions, etc.) are glossed.

To the best of our knowledge as of writing, the only previous NLP study that can be barely counted as context words-glossed pre-training is SenseBERT (Levine, Lenz, Dagan, Ram, Padnos, Sharir, Shalev-Shwartz, Shashua and Shoham, 2020). It infuses both context and target words with supersense (e.g., chocolate belongs to noun.food). During pre-training, apart from the MLM objective for predicting target words, the objective of classifying supersense of target words is computed simultaneously. Stringently, supersense (i.e., semantic category) cannot be deemed as semantic

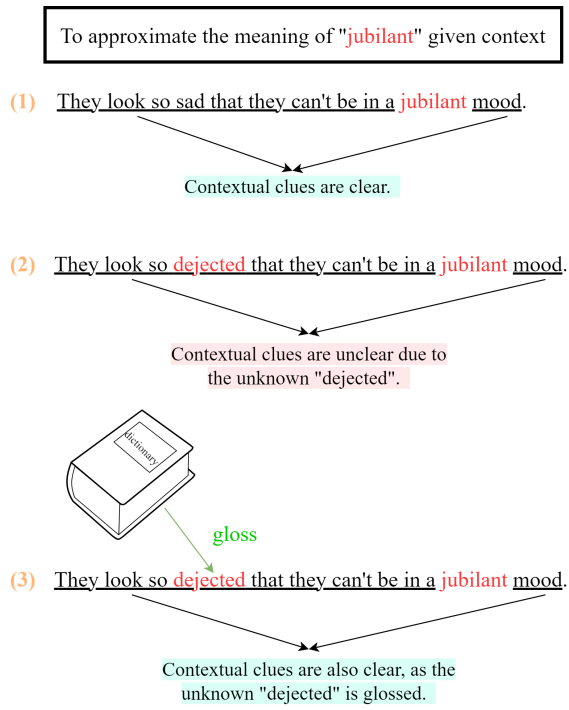

Figure 1: The approximation to the meaning of "jubilant" hinges on whether contextual clues are clear or not. If not, dictionaries are used to gloss unclear parts.

information for glosses, which are normally represented by definitions or synonyms.

Inspired by the solution to the linguistic phenomenon, we are motivated to investigate the potentiality of context words-glossed pre-training for the MLM objective of BERT (Devlin et al., 2019). Unlike the approaches for target words-glossed pre-training mentioned in the previous subsection, our strategy is focused on the embedding layer, which is illustrated in Figure 2. Except the position and segment embeddings, the roles of the embeddings related to semantic information are described as follows.

Word embeddings: They are the regular embeddings whose weight is tied to the decoder (Inan, Khosravi and Socher, 2017; Press and Wolf, 2017) for predicting target words. In HuggingFace (Wolf, Debut, Sanh, Chaumond, Delangue, Moi, Cistac, Rault, Louf, Funtowicz, Davison, Shleifer, von Platen, Ma, Jernite, Plu, Xu, Le Scao, Gugger, Drame, Lhoest and Rush, 2020) implementation of BERT that we use, its weight is initialized with normal distribution of mean 0.0 and standard deviation 0.02[1]. During pre-training[2], semantic features are optimized and stored in these embeddings by learning words from context

[1] https://github.com/HuggingFace/transformers/blob/33d033d694930110406099ce941f3f3678935b1c/src/transformers/models/bert/modeling_bert.py#L757

[2] For clarity, the NSP pre-training objective is omitted, where the quality of sentence-level predictions should also depend on the quality of acquired vocabulary.

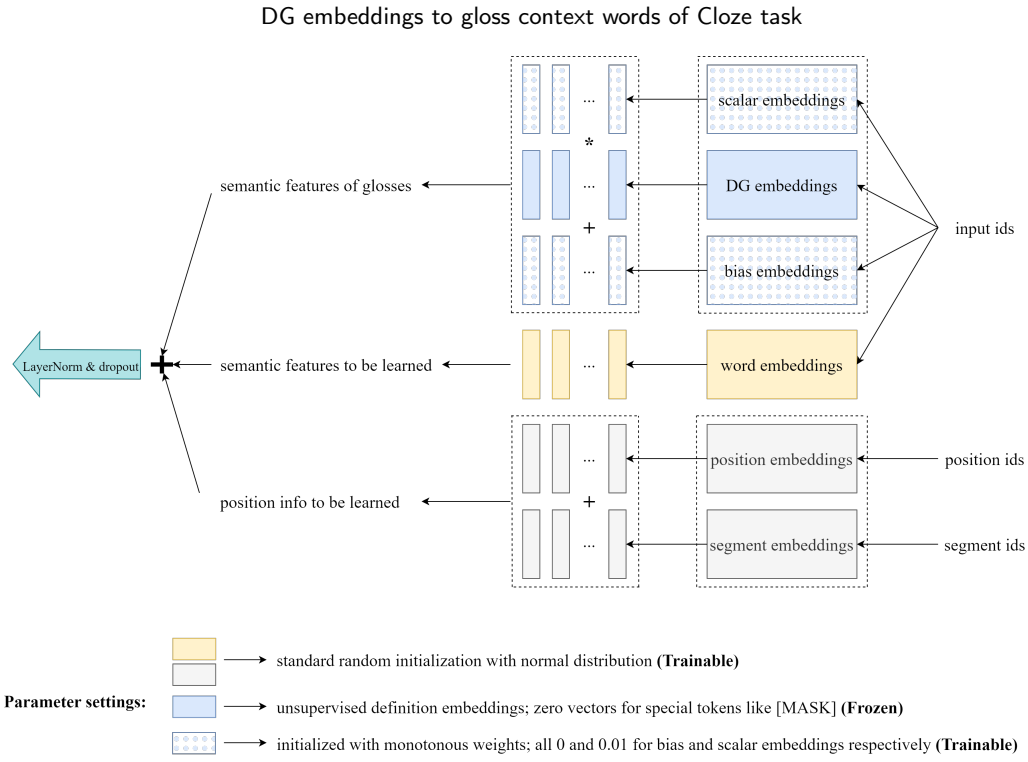

Figure 2: The embedding layer of BERT is shown for our proposed approach.

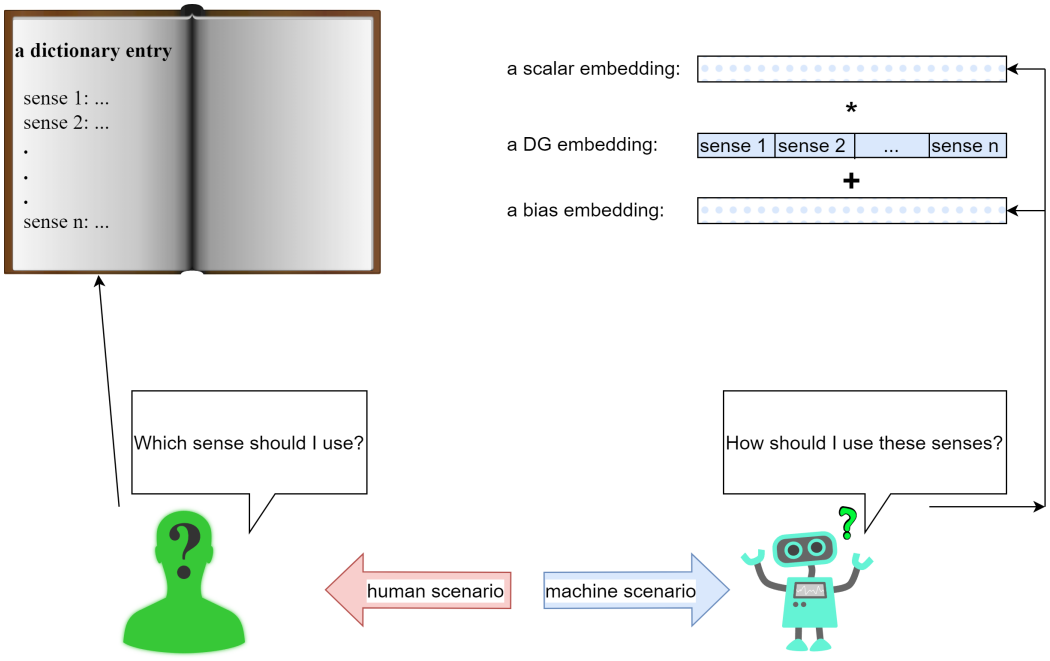

Figure 3: Let machines learn how to use the senses represented in DG embeddings.

via self-attention mechanism (Vaswani, Shazeer, Parmar, Uszkoreit, Jones, Gomez, Kaiser and Polosukhin, 2017).

DG embeddings: To infuse context words with semantic features of glosses, we propose DG embeddings to serve as auxiliary embeddings — the unsupervised definition embeddings learned from dictionaries and glossaries. Our idea is to simulate the scenario that BERT is holding a dictionary for context words while doing the Cloze task. Thus, like dictionary entries, words and tokens covered in DG embeddings should convey multiple senses. For example, the DG embedding of "cake" consists of literal meaning from the word and figurative meaning from "a piece of cake" (reflected by a certain dimensionality pattern in

its embedding vector representation). For tokens like "##ant", its DG embedding is composed of semantics from "jubilant" (ju, ##bil, ##ant), "intolerant" (into, ##ler, ##ant), and tokens in phrases if exist. Zero vectors are generated for special tokens like "[MASK]". This is desirable because in the context words-glossed setting, target words should not be glossed.

Scalar and bias embeddings: For context words or tokens in different input sequences, to allow machines to learn how to use the senses represented in DG embeddings, the scalar and bias embeddings are created for element-wise multiplication and addition (the idea is shown in Figure 3). This is consistent with the human practice of looking up words in dictionaries. When we encounter polysemy, which sense of a word is useful for us is case by case, and should be chosen by ourselves. Through pre-training, two embeddings learn a general distribution of their dimensional values. As to which value should be amplified or diminished more, it can be adjusted during fine-tuning for different tasks. Since the biggest absolute value in the weight matrix of DG embeddings created using our method (see Section 3) is 2.72, to stay similar magnitude as the weight of word embeddings, the weight initialization of scalar embeddings is filled with 0.01.

Through our approach, all context words of any input sequence can obtain DG embeddings by their input ids. Although it is viable to follow two previous works (Wu, Xing, Li, Ke, He and Liu, 2021; Yu et al., 2022) by setting a static or dynamic threshold of word frequency in corpora to highlight rare words only, any word can be potentially rare in context words-glossed pre-training. For example, almost every idiomatic expression consists of active vocabulary (e.g., "clean house", "a piece of cake", etc.) that tend to have high frequency in any corpus, but they can be regarded as rare words due to their metaphorical attribute (Choi, Lee, Choi, Park, Lee, Lee and Lee, 2021). If DG embeddings are used to gloss them only when their figurative senses occur, there will be a concern that scalar and bias embeddings are learned to completely lean towards their figurative semantics.

1.3. Unsupervised definition embeddings

To employ unsupervised learning for DG embeddings is based on three considerations.

First, based on the distributional hypothesis (Harris, 1954; Firth, 1957), by utilizing any probabilistic auto-encoding model with self-supervised learning such as early CBOW and skip-gram of word2vec (Mikolov, Chen, Corrado and Dean, 2013) or the transformer encoder (Vaswani et al., 2017) of BERT nowadays, learning words from context is applicable for machines. As DG embeddings used in our study is an auxiliary tool to assist the learning process, a technical duplicate will occur if the creation is related to those models with self-supervised learning. For example, if we use previous definition embeddings like CPAE (Bosc and Vincent, 2018) (a post-processed word2vec), dict2vec (Tissier, Gravier and Habrard, 2017) (adding regularization to the skip-gram objective of word2vec), or the recent DictBERT (Chen et al., 2022) (based on pre-trained BERT), the learning process will be "a model created for learning words from context" → "further learning word senses with definitions based on the model" → "using the learned word sense to assist a new model in learning words from context". This clearly suggests that the auxiliary tool for learning words from context originates from learning words from context. Another neural language model that can embed words with their definitions is tailored for reverse dictionaries (Hill, Cho, Korhonen and Bengio, 2016), which is not felicitous for our study. We need to underline that this consideration is based on appropriateness rather than quality.

Second, DG embeddings as an auxiliary tool are not meant for independent existence — they are not created to address downstream tasks by means of feature-based transfer (Yin and Schütze, 2015; Liu, Rzepka and Araki, 2022). Basic semantic relatedness of words and phrases is attainable via unsupervised learning based on the bag-of-words hypothesis (Salton, Wong and Yang, 1975), which is computationally inexpensive and therefore beneficial to future expansion. Creating definition embeddings based on the bag-of-words hypothesis is also in accordance with a real-world scenario of dictionary use: headword search in thesauruses like Longman Language Activator (Nesi, 2012). An example is listed below given the headword "strange".

- quirky → (of an aspect of somebody's personality or behaviour) a little strange
- grotesque → strange in a way that is unpleasant or offensive
- eerie → strange, mysterious and frightening

These dictionary entries are mainly defined by multiple underlined key features, where the headword is the key feature that they have in common while their nuance is determined by other key features. The headword recognized as active vocabulary is self-explanatory, which belongs to a linguistic property called reflexivity (McGregor, 2015). For humans, it seems that this approach is only feasible for thesauruses with conceptual categories. For machines, we devise an algorithm to extract key features (incorporating polysemy) for all dictionary entries (see Section 3.1). Extracted key features for each entry are represented by bag-of-words — for machines to process semantic relatedness of entries via key features, word order does not matter.

Third, unsupervised learning can allow phrases to have independent vector representations, as there is no need of one-hot encodings for supervisory signals. This is helpful for the following operation. Suppose the embedding for the phrase "a piece of cake" is $\vec{v}_1$, and the embedding for the word "cake" is $\vec{v}_2$. Then a combined embedding containing both literal and figurative semantics can be generated for the word "cake" by dint of $\frac{1}{2}(\frac{1}{4}\vec{v}_1+\vec{v}_2)$ (similar for the words "a", "piece", and "of"). This operation can considerably preserve

internal similarity of phrases. Further details are provided in Sections 3.2 and 3.3.

1.4. Research goals & contributions

We apply our approach to uncased BERT-case for an evaluation, and refer to our model as DG-BERT. To validate whether machines can achieve better acquisition of vocabulary via learning words from glossed context, the primary research goal is to observe if **fast pre-training convergence** for DG-BERT can be accomplished. An indication of this is the fast and effective acquisition of vocabulary for machines. Subsequently, we will further investigate the effectiveness by testing if **good understanding of literal and figurative semantics** can be achieved by DG-BERT. For literal semantics, standard downstream tasks are used for a test. For figurative semantics, we apply DG-BERT as a backbone model to MelBERT (Choi et al., 2021) for a test. In addition, as we mentioned in the previous subsection that DG embeddings can preserve internal similarities of phrases to a large extent, it will be investigated whether DG-BERT can absorb critical semantic features from DG embeddings so that **sensitive target-word predictions in the collocation of phrases** and **accurate self-attention for phrases in input sequences** are attainable. Our evaluation criterion is to compare DG-BERT with our vanilla BERT: both models are pre-trained under the exact same condition. All details and empirical evidences are provided in Section 4.

Definitions from dictionaries and glossaries are the external semantic knowledge that we draw on for our research purpose. The accomplishment of our research goals benefits from the utilization of the knowledge, which leads to the contributions of this paper enumerated as follows.

1. We discover that by glossing context words to predict target words for auto-encoding models, a better acquisition of vocabulary can be achieved by machines.
2. We show that unsupervised definition embeddings based on the bag-of-words hypothesis can help language modeling without using any cutting-edge technologies and can lead to computationally inexpensive future expansion.

2. Related works

In this section, we further extend the introduction of related works apart from the ones introduced in Sections 1.1 and 1.2. Also, we discuss the difference between our approach and the most related work.

Word meaning representations (Apidianaki, 2023) evolve as technologies advance. One invariant technical trend is to inject external semantic knowledge into the representations. The knowledge stems from diverse semantic resources: semantic networks (Miller, 1998), knowledge graphs (Baker et al., 1998; Speer et al., 2017), commonsense reasoning (Sap, Le Bras, Allaway, Bhagavatula, Lourie, Rashkin, Roof, Smith and Choi, 2019), paraphrase datasets (Dolan, Quirk and Brockett, 2004; Ganitkevitch, Van Durme and Callison-Burch, 2013), dictionaries and glossaries (He, Wang, Zhang, Huang and Caverlee, 2020). Besides the studies mentioned in Sections 1.1 and 1.2 regarding knowledge injection for language model representations nowadays, there exist three previous stages for other representations.

1st Stage. Vector Space Models (VSMs) (Turney and Pantel, 2010) such as term-document matrix use distributional counts to measure semantic similarity. Matrix factorizations like SVD (Deerwester, Dumais, Furnas, Landauer and Harshman, 1990) and LDA (Blei, Ng and Jordan, 2003) can be further used to decompose sparse count-based representations into dense latent representations. As there is no notion of antonym in VSMs (Landauer, Laham and Foltz, 1997), semantic representations of antonyms also tend to have high similarity by this approach. Yih, Zweig and Platt (2012) utilize a thesaurus to assign polarity values (1 if synonym and -1 if antonym) to each pair of words. Besides synonyms and antonyms, Chang, Yih and Meek (2013) take hypernyms into consideration by means of WordNet (Miller, 1998), and use three separate matrices to display word relations based on the three notions. We do not combine their approaches into the process of creating DG embeddings, because that will decrease the size of dictionary entries substantially, as the entries contain more than synonyms and antonyms.

2nd Stage. To incorporate semantic knowledge into word embeddings like word2vec (Mikolov et al., 2013) is reminiscent of many interesting previous works. According to Glavaš and Vulić (2018)'s work, technically there exist two categories: Joint Specialization Models (modifying pre-training objectives of embedding models by external semantic resources) (Yu and Dredze, 2014; Tissier et al., 2017) and Post-Processing Models (injecting external semantic knowledge into pre-trained embedding models) (Faruqui, Dodge, Jauhar, Dyer, Hovy and Smith, 2015; Bosc and Vincent, 2018). The latter is also known as retrofitting methods. Two representative studies that utilize dictionaries are CPAE (Bosc and Vincent, 2018) and dict2vec (Tissier et al., 2017). The CPAE model post-processes the pre-trained word2vec using an LSTM encoder, where inputs are words of concatenated definitions (considering polysemy) represented by word2vec and outputs (last hidden states) are definition embeddings. The post-processing procedure is completed by a decoder to reconstruct the bag-of-words representation of the definitions. The dict2vec method modifies the pre-training objective of word2vec skip-gram model by adding a cost for predicting related words. The relatedness is based on positive sampling, which is determined by words that are either strong or weak pairs (words co-occur in definitions of each other). The unrelatedness based on negative sampling strengthens the cost function.

3rd Stage. The transition from word embeddings to PLMs is contextualized word embeddings called ELMo (Peters, Neumann, Iyyer, Gardner, Clark, Lee and Zettlemoyer, 2018). Shi, Chen, Zhou and Chang (2019) discover that in many cases, contextualized embeddings of shared words

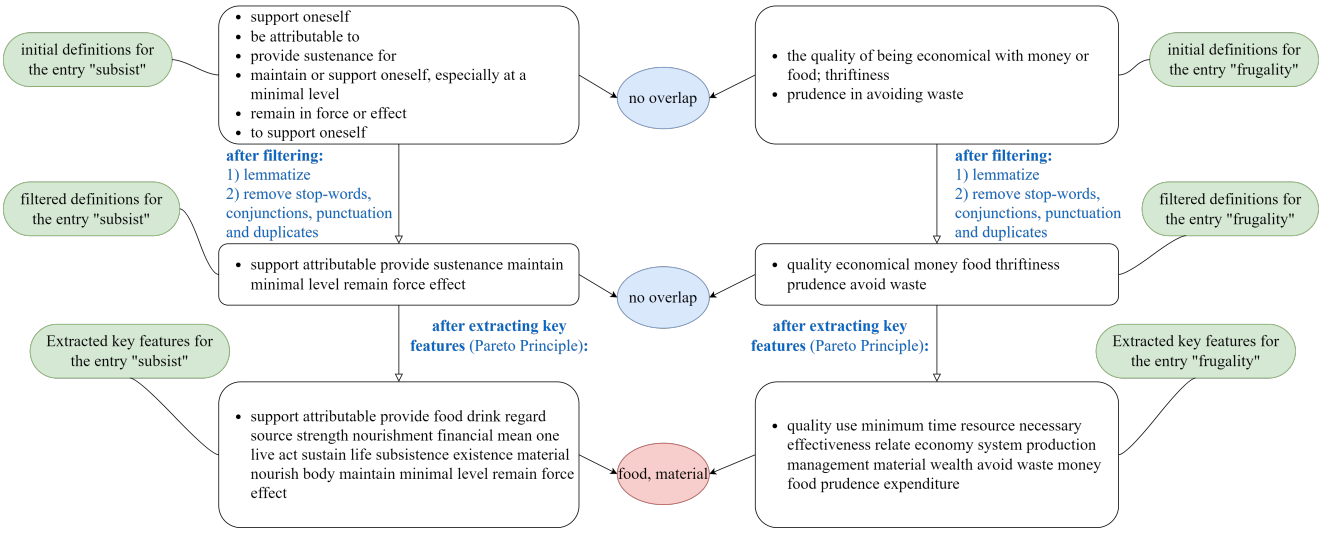

Figure 4: The flow of definition pre-processsing is exemplified by the first entry "subsist" and its semantically related entry "frugality".

in paraphrased contexts change drastically. To minimize the difference of the shared words, they propose a PAR (Paraphrase-Aware Retrofitting) method: to reshape input representations of contextualized models with an orthogonal transformation matrix. They apply the PAR method to ELMo for evaluation by utilizing paraphrase datasets. The PAR method belongs to the post-process category.

The most related work to this paper is known as Dict-BERT (Yu et al., 2022). Both approaches utilize dictionary resources to augment BERT pre-training, but the strategy is different. As mentioned in Sections 1.1 and 1.2, Dict-BERT (Yu et al., 2022) attempts to gloss target words for the MLM pre-training objective of BERT and its knowledge fusion is implicit (external semantic knowledge is merged into the input sequence of BERT), while our focus is to gloss context words for target-word predictions and the knowledge fusion can be considered as explicit (DG embeddings are structurally connected with BERT input layer). Both of them have linguistic justifications discussed at the beginning of Sections 1.1 and 1.2. However, a technical advantage for our approach is that DG-BERT can handle long-sequence tasks like question answering as usual. Contrarily, Dict-BERT has to append definition sentences to input sequences, which hinders its potentiality from long-sequence tasks. We show other intrinsic strengths of our approach in Sections 4 and 5.

3. DG embeddings

The creation of DG embeddings is not complex, which involves two critical steps: extraction of key features in Section 3.1 and adaption to the wordpiece vocabulary of BERT in Section 3.2. We present an evaluation criterion in Section 3.3, which is also beneficial for future expansion and quality improvement.

For unsupervised learning based on the bag-of-words hypothesis, the term-document matrix of VSMs (Turney and Pantel, 2010) is employed to create DG embeddings. Documents are indicated by dictionary and glossary entries (words and phrases) while terms are represented by words as features comprising definitions. The dictionary resources are released by a previous work (Ishiwatari, Hayashi, Yoshinaga, Neubig, Sato, Toyoda and Kitsuregawa, 2019), including Oxford, Wiktionary, and WordNet (Miller, 1998). We found one glossary resource provided by PARADE dataset (He et al., 2020), which contains computer science terminologies. After assembling all resources, there are 305,518 unique entries containing 189,137 phrases.

Algorithm 1 Extraction of Key Features

$X \leftarrow entry\_definition\_map$
$Y \leftarrow top\_20\%\_key\_feature\_list$
$Z \leftarrow word\_is\_identified\_as\_key\_feature$
$K \leftarrow already\_identified\_word\_list$
procedure IDENTIFY($word, X, Y, Z, K$)
  if $word \in K$ **then**
    return
  end if
  $K.add(word)$
  if $word \in Y$ **and** $word \notin Z$ **then**
    $Z.add(word)$
  else
    $definition = X[word]$
    $words = word\_tokenize(definition)$
    for $word \in words$ **do**
      IDENTIFY($word, X, Y, Z, K$)
    end for
  end if
end procedure

3.1. Extraction of key features

As exemplified in Figure 4, the initial definition for each entry are concatenated by multiple definitions (including polysemy if exists) from different dictionary and glossary resources. All words are lemmatized first, then duplicates and unnecessary words are removed after filtering. As a result, each entry consists of multiple unique words as features, which make the term-document matrix comprise the values of 1 and 0 only. Unlike information retrieval (Salton et al., 1975) relying on distributional counts to determine semantic relatedness of documents, for dictionary entries introduced in Section 1.3, it should depend on how many key features they have in common. This is consistent with two previous works (Yih et al., 2012; Chang et al., 2013) seeking to use the matrix for the generation of synonyms and antonyms. After filtering, the total unique features are 54,861; correspondingly, the matrix size is 305,518 × 54,861.

To extract key features, the concrete method is to employ Pareto Principle, also well known as 80/20 rule — roughly 80% of consequences come from 20% of causes. In our case, roughly 80% semantic relatedness are determined by top 20% words as key features. We count the frequencies of those 54,861 words based on their occurrences in all definitions, and select 10,748 words of high frequency as top 20%. For the rest 44,113 words, we devise a recursive algorithm to search their definitions for replacement (see Algorithm 1 for details). As a result shown in Figure 4, the refined definition of "subsist" contains more key features like food, drink, material, etc. More robust semantic relatedness to other entries is achievable; e.g., overlapped features are generated between the definitions of "subsist" and "frugality" after this process. Meanwhile the matrix size is scaled down to 305,518 × 10,748. With more key features added to each entry and reduced dimensionality, the matrix become less sparse.

3.2. Matrix factorization & adaption

The size of wordpiece vocabulary and length of hidden states are 30,522 and 768 respectively in the HuggingFace uncased BERT-base (Wolf et al., 2020). Correspondingly, there are two steps to convert the matrix size from 305,518 × 10,748 to 30,522 × 768. The first step is to scale down the matrix from 305,518 × 10,748 to 305,518 × 768 by matrix factorization. As our matrix does not rely on different distributional counts to distinguish semantics, such as a centered matrix for computing covariance like PCA (Maćkiewicz and Ratajczak, 1993), we directly use the basic algorithm truncated SVD for matrix factorization.

SVD decomposes a matrix X into the product of three matrices UDV^T, where U and V are in orthogonal column form with unit length: U^TU and V^TV are identity matrices and D is a diagonal matrix of singular values. If X is of rank r, then D is also of rank r. Let D_k, where k < r, be the diagonal matrix formed from the top k (k = 768 in our case) singular values, and let U_k and V_k be the matrices produced by selecting the corresponding columns from U and V. Among all matrices $\hat{X} = U_k D_k V_k^T$ of rank k, to select one that best approximates the original matrix X, the Frobenius norm (Golub and Van Loan, 1996) $||\hat{X} - X||_F$ is used to minimize the approximation errors. After obtaining best approximated $\hat{X}$, the U_k in $\hat{X}$ is the factorized matrix with the size of 305,518 × 768 for our purpose. We address the latent vector representations in this matrix for all entries as unsupervised definition embeddings.

Algorithm 2 Adaption to BERT Vocabulary

$X \leftarrow BERT\_vocabulary$
$Y \leftarrow entry\_list$
$Z \leftarrow definition\_embed$
$K \leftarrow DG\_embed$
procedure ADAPT(X, Y, Z, K)
  for $BERT\_token \in X$ **do**
    $count = 0$
    $temp\_embed = \vec{0}$
    for $entry \in Y$ **do**
      $entry\_tokens = BERT\_tokenizer(entry)$
      $length = len(entry\_tokens)$
      if $BERT\_token \in entry\_tokens$ **then**
        $temp\_embed = temp\_embed + \frac{1}{length} \cdot Z(entry)$
        $count = count + 1$
      end if
    end for
    if $count \neq 0$ **then**
      $temp\_embed = \frac{1}{count} \cdot temp\_embed$
      $K.add(temp\_embed)$
    else
      $K.add(temp\_embed)$
    end if
  end for
end procedure

The next step to scale down the factorized matrix from 305,518 × 768 to 30,522 × 768 is called matrix adaption (see Algorithm 2). We seek to allow words or tokens in BERT vocabulary to encompass multi-senses. For example, suppose there are only two definition embeddings related to the word "cake": the definition embedding $\vec{v_1}$ for "a piece of cake" and the definition embedding $\vec{v_2}$ for the word itself. Then according to Algorithm 2, the DG embedding generated for the word "cake" is $\frac{1}{2}(\frac{1}{4}\vec{v_1} + \vec{v_2})$. As each word or token contained in a phrase is an inseparable component to form the complete phrase, computationally, semantic features conveyed by the phrase should be equally shared ($\frac{1}{4}\vec{v_1}$ in the example). To avoid large dimensional values, the average is calculated for each adaption ($\frac{1}{2}$ in the example). Zero vectors are generated for 5,940 special tokens such as "[MASK]" and foreign vocabulary like Chinese characters. In this paper, we focus on a monolingual scenario, and treat foreign vocabulary as special tokens. As for "[MASK]", even if there existed a definition embedding for it, we would also modify it to a zero vector, because it represent target words and in context words-glossed pre-training, target

words should not be glossed. The matrix after adaption with the size of 30,522 × 768 is the final DG embeddings.

3.3. Evaluation of DG embeddings

DG embeddings are evaluated at word and phrase levels, plus the potentiality of their vector space to assist the one of word embeddings (see the last paragraph of this section).

For evaluation at word and phrase levels, the evaluation criterion is considered for to what degree DG embeddings can preserve semantic features after matrix adaption. In the hypothetical example mentioned in the previous subsection, the DG embedding generated for the word "cake" is $\frac{1}{2}(\frac{1}{4}\vec{v}_1 + \vec{v}_2)$. Although multi-senses exist in the DG embedding, this can decrease its cosine similarity to other literally close words (e.g., "bread"). Also for the phrase "a piece of cake", the internal similarity (see Equations 1 and 2, where "wt" denotes word or token) of this phrase before matrix adaption is 100% ($\frac{1}{4}\vec{v}_1$ for each word), but after adaption, the similarity will inevitably decrease, because the DG embeddings for each word are composed of multiple definition embeddings.

$$phrase = \{wt_0, wt_1, ..., wt_n\} \tag{1}$$

$$InternalSim(phrase) = \frac{1}{n+1}\sum_{i=0}^{n} CosineSim(embed(wt_i), \sum_{\substack{j=0 \\ j\neq i}}^{n} embed(wt_j)) \tag{2}$$

For the evaluation at the word level, we choose the TOEFL synonym selection task[3] (Landauer and Dumais, 1997) for a straightforward comparison. An example is "enormously → {appropriately, uniquely, tremendously, decidedly}", where "tremendously" is the semantically closest word among four candidates. The dataset contains 80 such questions. When a word consists of tokens, the addition of their embeddings form a complete embedding (e.g., embed(pan) + embed(##cake) to represent embed(pancake)). The highest cosine similarity between embeddings is used for decision. For the evaluation at the phrase level, the internal similarities are calculated based on Equations 1 and 2 for all 189,137 phrases. We compare DG embeddings with another two embeddings: the word embeddings from pre-trained HuggingFace uncased BERT-base (Wolf et al., 2020); the embeddings whose weight is initialized by normal distribution with mean 0 and standard deviation 0.02 (introduced in Section 1.2) as our baseline.

[3]The task dataset should be requested via a portal, which is available at https://aclweb.org/aclwiki/TOEFL_Synonym_Questions_(State_of_the_art). The samples in this dataset might change with time. For example, the sample "rug → {sofa, ottoman, carpet, hallway}" mentioned in Faruqui et al. (2015)'s work does not exist in the dataset that we have obtained.

Table 1
Three types of embeddings are evaluated at the word and phrase levels. "BERT word embeddings" is the word embeddings from pre-trained HuggingFace uncased BERT-base (Wolf et al., 2020). The weight of "Random word embeddings" is initialized by normal distribution with mean 0 and standard deviation 0.02.

Embeddings	TOEFL task (%)	Average internal similarities of phrases (%)
Random word embeddings	22.8	0.4
BERT word embeddings	76.0	39.1
DG embeddings	65.8	59.6

Table 2
The percentage in each interval of internal similarities are scored by three types of embeddings. "BERT word embeddings" is the word embeddings from pre-trained HuggingFace uncased BERT-base (Wolf et al., 2020). The weight of "Random word embeddings" is initialized by normal distribution with mean 0 and standard deviation 0.02.

Intervals	Random word embeddings (%)	BERT word embeddings (%)	DG embeddings (%)
(0.0, 0.1]	49.4	0.2	0.5
(0.1, 0.2]	1.1	2.8	2.5
(0.2, 0.3]	0.3	15.1	5.4
(0.3, 0.4]	0.1	32.4	8.9
(0.4, 0.5]	0.1	36.9	12.5
(0.5, 0.6]	0.0	12.3	16.6
(0.6, 0.7]	0.0	0.3	19.2
(0.7, 0.8]	0.0	0.0	18.8
(0.8, 0.9]	0.0	0.0	13.3
(0.9, 1.0]	0.0	0.0	2.3

The results are presented in Tables 1 and 2. There is a prerequisite information for the evaluation at the word level: when the matrix size is 305,518 × 10,748, the performance for TOEFL task is 77.2; after matrix factorization scaled down to 305,518 × 768, the performance is 74.7 due to inevitable information loss. The performance is based on total 79 samples, as there is one out-of-vocabulary sample, which are also the performance of three embeddings based on. Therefore, the upper bounds for DG embeddings at word and phrase levels are 74.7 and 100.0 (average internal similarities) respectively. As shown in both tables, although the performance naturally drops, the overall performance is comparable to BERT word embeddings, considering (65.8 + 59.6) > (76.0 + 39.1). Especially at the phrase level, DG embeddings account for bigger percentile in high intervals. Since DG embeddings are outperformed at the word level, it can be worried that DG embeddings can assist context words of BERT at the early stage of pre-training, but they might inflict side effect at the late stage. We will empirically show in Sections 4.3 and 5 that there is no side effect.

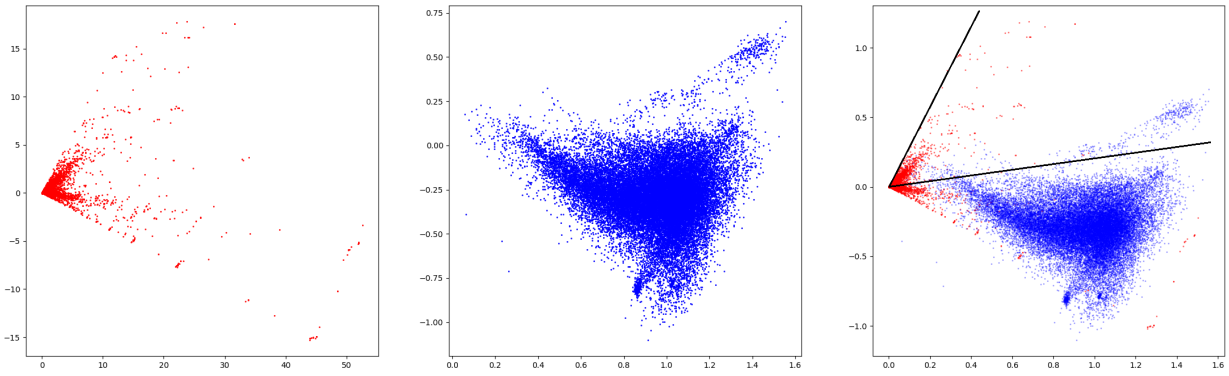

Figure 5: SVD visualization of embeddings from left to right → **1st:** DG embeddings; **2nd:** word embeddings of Hugginface BERT-base-uncased; **3rd:** comparison between **1st** and **2nd**. In **3rd**, the magnitude of DG embeddings is scaled down to a similar magnitude of the word embeddings.

We use an extreme but easy to understand example to further explain the performance at word and phrase levels. Suppose there are only five definition embeddings: $\vec{v}_1$ for "a piece of cake"; $\vec{v}_2$ for "a"; $\vec{v}_3$ for "piece"; ; $\vec{v}_4$ for "of"; ; $\vec{v}_5$ for "cake". Then according to Algorithm 2, the DG embeddings generated for these tokens are itemized as follows.

- DG_embed("a") = $\frac{1}{2}(\frac{1}{4}\vec{v}_1 + \vec{v}_2)$
- DG_embed("piece") = $\frac{1}{2}(\frac{1}{4}\vec{v}_1 + \vec{v}_3)$
- DG_embed("of") = $\frac{1}{2}(\frac{1}{4}\vec{v}_1 + \vec{v}_4)$
- DG_embed("cake") = $\frac{1}{2}(\frac{1}{4}\vec{v}_1 + \vec{v}_5)$

The internal similarity of "a piece of cake" is high because $\frac{1}{4}\vec{v}_1$ is merged into their DG embeddings. On the other hand, as multi-senses are incorporated into DG embeddings for each token, their performance at word level naturally drops. However, this is what we seek to achieve — DG embedding for each token encompasses multi-senses, and allow scalar and bias embeddings to learn which sense to use during pre-training (illustrated in Figure 3). Actually, the decrease within 10% (from 74.7 to 65.8 described in the previous paragraph) is acceptable for the trade-off between gaining multi-senses and losing performance on single sense.

We also make an attempt on the matrix without key feature extraction, namely the matrix with the size of 305,518 × 54,861 (by filtering process only in Figure 4). The performance on TOEFL task is the same (77.2), and decreases to 73.4 with the size of 305,518 × 768 after matrix factorization. However, after matrix adaption, the performance on TOEFL task increases to 69.6, while the internal similarities of phrases decrease to 43.4. As a result, the overall performance is lower than one with key feature extraction: (69.6 + 43.4) < (65.8 + 59.6). The result further validates the effectiveness of key feature extraction.

Another way to evaluate DG embeddings is to see if it can assist word embeddings in learning better semantics. The crucial criterion is to examine whether the anisotropic vector space (Gao, He, Tan, Qin, Wang and Liu, 2019; Ethayarajh, 2019) of word embeddings can be alleviated. As shown in Figure 5, although the distribution of DG embeddings is not directionally uniform, the aperture size of the cone is noticeably larger than the one of word embeddings (the comparison demonstrated in the 3rd sub-figure). As DG embeddings are structurally connected with word embeddings at the embedding layer, extrapolating from the comparison, it can be expected that the distribution of word embeddings in the enclosed area of two lines can become more dense after pre-training, which indicates the enlarged aperture size for word embeddings. The corresponding evidence will be provided in Section 5.

4. DG-BERT

In this section, we experiment with our vanilla BERT (uncased BERT-base) and DG-BERT (uncased BERT-base with DG, scalar and bias embeddings). For a fair comparison, both models are pre-trained on *the same pre-processed text files with the exact same masked tokens*[4] *and hyperparameters*.

4.1. Pre-training & experimental settings

We follow the original hyperparameter settings (Devlin et al., 2019) to pre-train our vanilla BERT and DG-BERT, which are listed in Appendix A. The loss is also computed by the addition of NSP and MLM pre-training objectives. Three settings — max-seq-length 448, no 10% random tokens to replace [MASK], and pre-training corpus — slightly vary from the origin due to three special reasons listed below.

[4]We perform three basic steps for text file pre-processing: remove HTML tags, concatenate short sentences, and convert texts to input format of BERT model. We saved the pre-processed text files (a Python dictionary of input_ids, attention_masks, segment_ids, MLM_labels and NSP_labels) to a Pytorch file. It can be loaded anytime for pre-training both models.

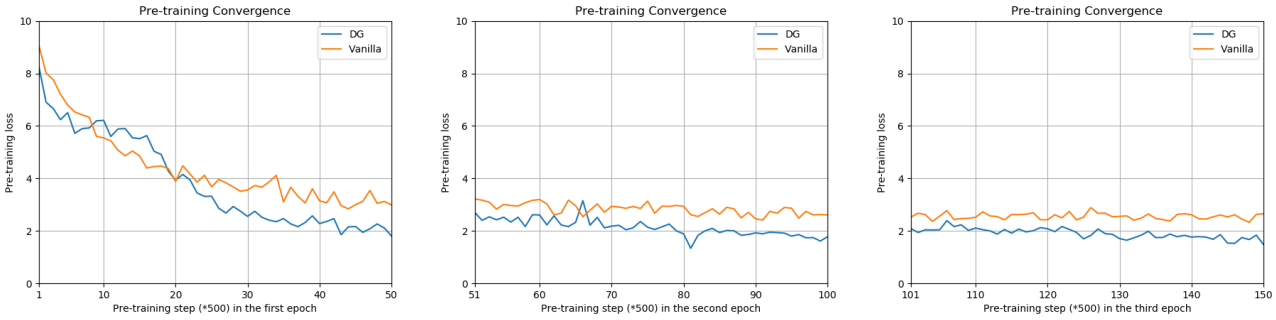

Figure 6: Left: Every 500-step batch loss is shown for the first epoch; **Middle:** Every 500-step batch loss is shown for the second epoch.; **Right:** Every 500-step batch loss is shown for the third epoch.

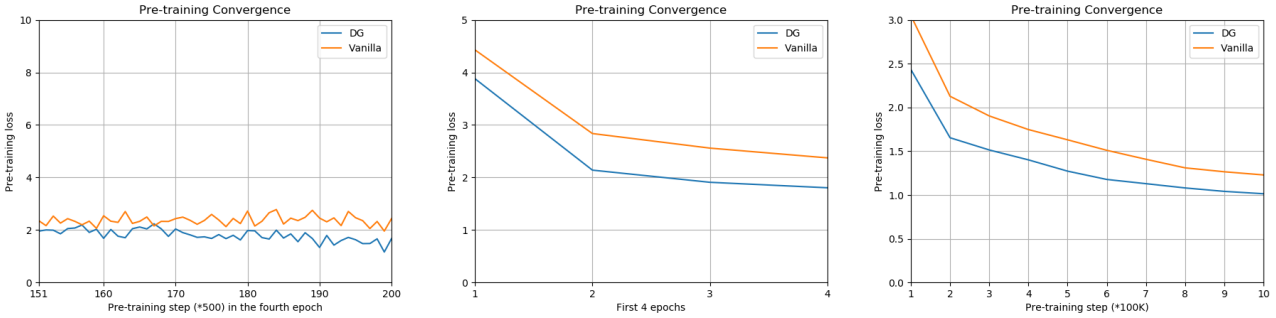

Figure 7: Left: Every 500-step batch loss is shown for the fourth epoch; **Middle:** The convergence is shown for the first 4 epochs contained in the first 100K pre-training steps. Four loss values are the average of all 500-step batch losses in each epoch; **Right:** The convergence is shown for every 100K pre-training steps. Ten loss values are the average of all 500-step batch losses in each 100K step.

1. Our pre-training server — its RAM (for loading pre-processed text files) and total GPU memory (for batch size 256) — physically cannot hold max-seq-length 512. Due to the physical constraint, we have to scale the max length from 512 down to 448.
2. Among 15% randomly-selected masked tokens, the distribution proposed by Devlin et al. (2019) is 80% for [MASK], 10% to remain original tokens, and 10% to be replaced by random tokens. In our case, the distribution is 80%, 20%, and 0%. As random tokens (in most cases except special tokens) also can obtain their non-zero-vector DG embeddings via their input ids, there is a concern that corresponding scalar and bias embeddings will be improperly optimized if we keep using random tokens.
3. As stated in the GitHub page of BERT, BookCorpus (Zhu, Kiros, Zemel, Salakhutdinov, Urtasun, Torralba and Fidler, 2015) is not available for public download, we use OpenWebText (Aaron Gokaslan*, 2019) and Wikipedia dumps. Due to the physical constraint mentioned above, text files with the size of roughly 7 GB (3.5 GB from each) are used for pre-training. The corpus size is comparatively small in modern language model pre-training, but it is an affordable size for pre-training in academic environment.

Our model implementation is based on HuggingFace transformer (Wolf et al., 2020), and coding multi-GPU pre-training with mixed precision (fp16) (Micikevicius, Narang, Alben, Diamos, Elsen, Garcia, Ginsburg, Houston, Kuchaiev, Venkatesh and Wu, 2018) is based on Accelerate (Sylvain Gugger, 2022).

In the following subsections, a variety of experiments are conducted to evaluate DG-BERT from different perspectives. For those experiments where fine-tuning is involved, two points need to be underlined: 1) existing fine-tuning techniques are not utilized; 2) each task is run with five different random seeds[5], and the average of five runs is reported based on optimal hyperparameter settings (details in Appendix A).

In this section, we consistently keep the comparison between DG-BERT and our vanilla BERT pre-trained under the exact same condition. We leave the comparison with the most related work in Section 5, where we further explain the intrinsic strengths of our approach based on this section.

[5]We leave random seed generation to the nondeterministic behavior of PyTorch (Paszke, Gross, Massa, Lerer, Bradbury, Chanan, Killeen, Lin, Gimelshein, Antiga, Desmaison, Kopf, Yang, DeVito, Raison, Tejani, Chilamkurthy, Steiner, Fang, Bai and Chintala, 2019).

Table 3
A preliminary test is performed for target-word predictions in collocations of two common idiomatic expressions. "DG-BERT w/o DG" denotes the whole package of DG, scalar and bias embeddings is removed from pre-trained DG-BERT.

Input sequences	Top 5 predictions & softmax scores	Models
[CLS] They left on the [MASK] of the moment. [SEP]	spur, heat, beat, power, scale 0.9817, 0.0052, 0.0021, 0.0007, 0.0006	DG-BERT
	spur, heat, beat, power, edge 0.9587, 0.0142, 0.0034, 0.0022, 0.0011	DG-BERT w/o DG
	spur, heat, eve, basis, face 0.5943, 0.0358, 0.0267, 0.0250, 0.0181	Vanilla BERT
[CLS] The task is a piece of [MASK]. [SEP]	cake, art, work, paper, history 0.3945, 0.2898, 0.0607, 0.0305, 0.0256	DG-BERT
	cake, art, work, paper, software 0.2623, 0.1981, 0.1141, 0.0423, 0.0314	DG-BERT w/o DG
	paper, work, cake, machinery, wood 0.1550, 0.0528, 0.0449, 0.0317, 0.0307	Vanilla BERT

4.2. Pre-training convergence

We first show that DG-BERT converges faster than our vanilla BERT in pre-training, which is the indication of acquiring vocabulary in a fast and effective manner. From our understanding, the pre-training process can be generally divided into three phases, and the corresponding pre-training losses for convergence are summarized in Figures 6 and 7.

The first phase happens in the first epoch. Until the end of this phase, as DG-BERT has not seen all input sequences, scalar and bias embeddings are still under-fitting. Accordingly, what DG embeddings can provide for context words is overall semantic features of glosses. Nevertheless, the pre-training convergence is generally faster than vanilla BERT (see the left sub-figure of Figure 6).

The second phase is the refining process of scalar and bias embeddings. After the first epoch seeing all input sequences, scalar and bias embeddings gradually refine their dimensional values to learn how to use dimensionality patterns of DG embeddings. In this phase, the fast pre-training convergence is more obvious than the first phase, because the semantic features provided by DG embeddings for context words are more rationally adjusted by scalar and bias embeddings, and further reinforced by the semantic features of word embeddings learned through predicting target words. As shown from the middle and right sub-figures of Figure 6 to the left sub-figure of Figure 7, the loss is consistently lower than the vanilla BERT (except a noticeable outlier in the second epoch). The summary of the first 4 epochs (namely the first 100K pre-training steps) is illustrated in the middle sub-figure of Figure 7. As shown in this figure, the loss gap becomes larger from the second epoch on, and keeps the trend.

The third phase is where the refinement ends and marginal adjustment begins. It can be inferred that this phase starts from the 600K pre-training step shown in the right sub-figure of Figure 7. The loss of pre-training DG-BERT drops not as significantly as previous steps, and the loss gap becomes gradually smaller. However, the gap is still noticeable, as the final loss of the vanilla BERT is somewhere between 500K and 600K of DG-BERT.

4.3. Preliminary tests

We perform a preliminary test for target-word predictions in the collocation of two common idiomatic expressions (see Table 3). Since both DG-BERT and our vanilla BERT are pre-trained on the same conditions (emphasized at the beginning of Section 4), the word embeddings of DG-BERT are optimized better than the ones of our vanilla BERT. As shown in "DG-BERT w/o DG", although the softmax scores for two target words "spur" and "cake" are lower than the ones of DG-BERT (0.9587 to 0.9817 and 0.2623 to 0.3945), they are substantially higher than the ones of our vanilla BERT (0.9587 to 0.5943 and 0.2623 to 0.0449). Later in Section 4.6, we present experimental results on a published dataset for a further evaluation.

We also perform the TOEFL test of Section 3.3 for three types of embeddings: word embeddings of DG-BERT; word embeddings plus scalar, bias and DG embeddings of DG-BERT; word embeddings of our vanilla BERT. The results are 74.7, 77.2, and 73.4 respectively, where the word embeddings of DG-BERT (74.7) is better than the one of our vanilla BERT (73.4). From 74.7 increased to 77.2, it can be concluded that although DG embeddings are moderate on the performance at the word level (65.8 for the TOEFL test), they do not stifle the optimization of word embeddings in pre-training but rather assist and improve it. This is also validated by the faster pre-traning convergence presented in this subsection. Furthermore, the better performance of DG embeddings at the phrase level shown in Section 3.3 contributes to more sensitive target-word predictions in collocations of phrases. As mentioned above, we will further validate it in Section 4.6.

4.4. Standard downstream tasks

The tasks in this subsection include sequence classification (GLUE benchmark Wang et al., 2019a), question answering (SQuAD Rajpurkar, Zhang, Lopyrev and Liang,

Table 4
The experimental results of GLUE benchmark are reported on **validation set**. The result for MNLI is reported on the average of match and mismatch.

Models	MNLI (Acc.)	QQP (Acc.)	QNLI (Acc.)	SST2 (Acc.)	COLA (Matthews)	STSB (Pearson)	MRPC (Acc.)	RTE (Acc.)	Avg.
Vanilla BERT	83.92	90.90	91.20	91.92	58.50	89.59	86.76	69.34	82.77
DG-BERT	85.09	91.13	91.40	93.00	60.14	90.07	88.09	73.94	84.11

Table 5
The experimental results are reported on **validation set** for SQuAD, CoNLL2003, and SWAG tasks. The results of CoNLL2003 are based on label-first-token.

Models	SQuAD (EM/F1)	SQuAD2.0 (EM/F1)	CoNLL2003 (F1)	SWAG (Acc.)
Vanilla BERT	80.47/87.76	72.26/75.59	94.50	72.46
DG-BERT	82.81/89.80	74.63/77.75	94.94	75.56

Table 6
The experimental results are reported on **validation set** for two paraphrase identification tasks.

Tasks	Models	Acc.	Prec.	Rec.	F1
PARADE	Vanilla BERT	73.32	74.31	69.58	71.84
	DG-BERT	75.53	76.53	72.24	74.25
PAWS-wiki	Vanilla BERT	92.60	88.79	95.31	91.93
	DG-BERT	93.61	90.31	95.85	93.00

Table 7
The experimental results are reported on test set for four metaphor detection tasks.

Tasks	Backbone Models	Prec.	Rec.	F1
VUA18	Vanilla BERT	78.07	76.82	77.43
	DG-BERT	77.36	78.68	78.01
VUA20	Vanilla BERT	72.83	68.51	70.60
	DG-BERT	72.53	70.23	71.37
MOH-X (zero-transfer)	Vanilla BERT	72.80	78.48	75.53
	DG-BERT	71.76	82.97	76.52
TroFi (zero-transfer)	Vanilla BERT	52.56	72.21	60.84
	DG-BERT	51.93	74.09	60.99

2016; Rajpurkar, Jia and Liang, 2018), token classification (CoNLL2003 Tjong Kim Sang and De Meulder, 2003), and multiple choice (SWAG Zellers, Bisk, Schwartz and Choi, 2018). The selection of those tasks follows the original BERT paper (Devlin et al., 2019). The experimental results are presented in Tables 4 and 5 respectively.

As shown in both tables, DG-BERT consistently outperforms our vanilla BERT. Besides, the increments on the two question answering tasks are noticeable. Extrapolating from the performance on the EM metric, it seems that DG-BERT can capture more accurate self-attention for collocation of phrases. An additional analysis for a further verification is described in Section 4.6.

As the definitions of computer science terminologies from PARADE (He et al., 2020) participate in the creation of DG embeddings, and some terminologies appear in definitions of other terminologies, we fine-tune PARADE paraphrase identification task to validate the effectiveness of using the glossary resource. Also to validate if position embeddings of DG-BERT is well pre-trained, we fine-tune the paraphrase identification for PAWS (Zhang, Baldridge and He, 2019a) task. This task is meant to measure model's sensitivity to word order and syntactic structure, which disables zero-transfer from other task datasets. Both tasks have high lexical overlap on non-paraphrases other than paraphrases. The results are provided in Table 6. Since the two task datasets contain fairly evenly-distributed paraphrases and non-paraphrases (46.9% & 53.1% for PARADE and 44.2% & 55.8% for PAWS-wiki), two metrics are crucial: accuracy (overall performance) and precision (implying the degree of correctly predicted non-paraphrases). As the results show, DG-BERT gains noticeable improvement. The increment on PARADE task is comparable to SciBERT (Beltagy, Lo and Cohan, 2019) reported in the original paper, and DG-BERT can retain comparatively high recall score.

4.5. Detection of figurative semantics

MelBERT (Choi et al., 2021) designs two task-specific layers above BERT encoder for detecting metaphorical expressions in input sequences. The SPV (Selectional Preference Violation) layer is designed for a theory that a metaphorical word is identified by the semantic difference from its surrounding words; the MIP (Metaphor Identification Procedure) layer is designed for a theory that a metaphorical word is identified by the gap between the contextual and literal meaning of a word. The metaphorical expressions covered in this study comprise single words with different syntactic roles. For instance, in the sentence "Finally, the debate sharpened.", "sharpend" as a verb conveys figurative semantics. To verify if DG-BERT can grasp figurative semantics better than vanilla BERT, we apply them as backbone models to MelBERT for comparison.

The experimental results are shown in Table 7 (we follow the source code to report results on test set). VUA18 (Leong, Beigman Klebanov and Shutova, 2018) and VUA20 (Leong, Beigman Klebanov, Hamill, Stemle, Ubale and Chen, 2020) are two big major task datasets, but with extremely unevenly-distributed labels (approximately 90% literal words and 10% figurative words). Thus, following the original paper, the

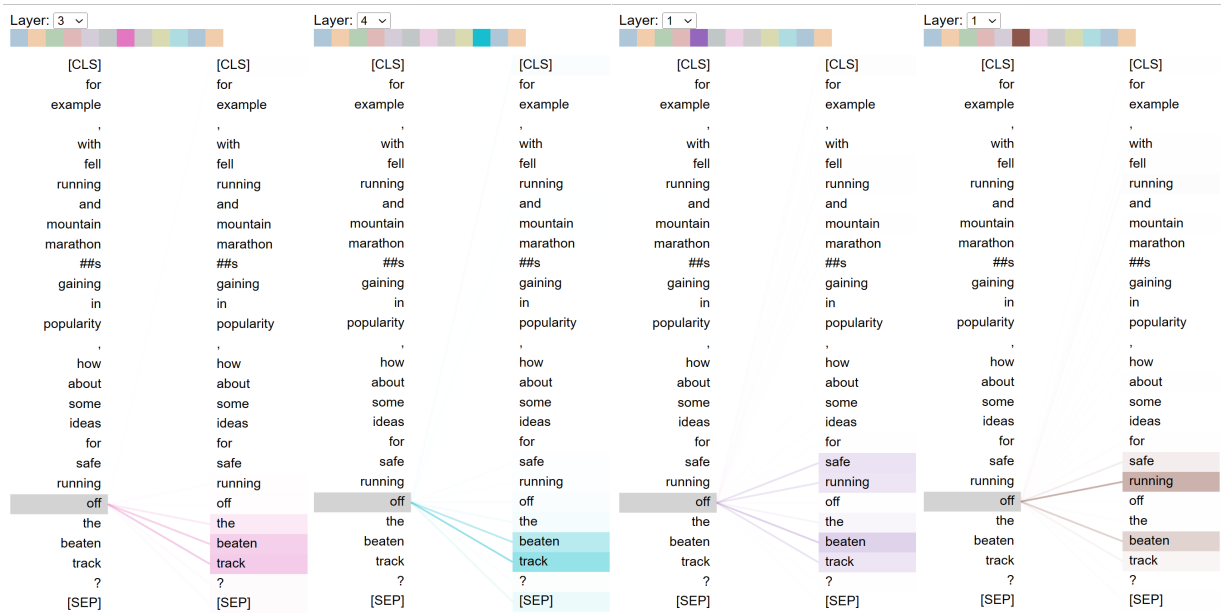

Figure 8: From left to right → **1st:** DG-BERT; **2nd:** Vanilla BERT; **3rd:** DG-BERT; **4th:** Vanilla BERT

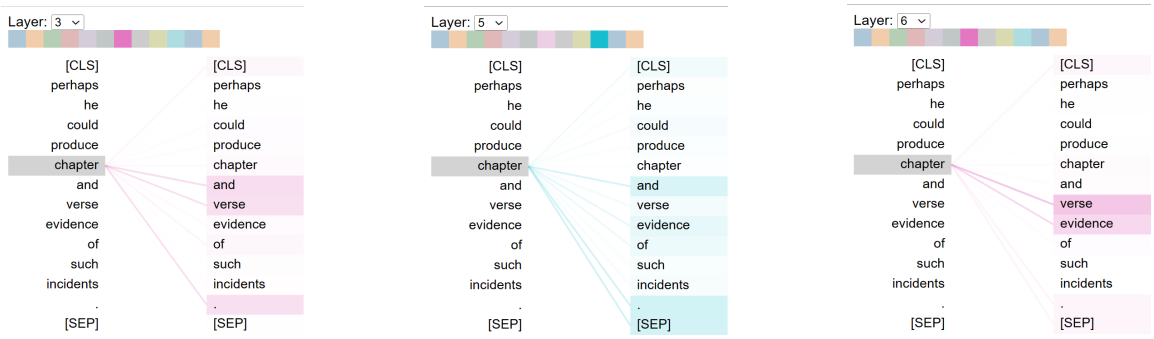

Figure 9: From left to right → **1st:** DG-BERT; **2nd:** Vanilla BERT; **3rd:** DG-BERT

accuracy metric is trivial and omitted in the table. Two tasks with small datasets MOH-X (Mohammad, Shutova and Turney, 2016) and TroFi (Birke and Sarkar, 2006) are performed by zero-transfer, namely tested by models fine-tuned on VUA20. The evaluation criterion is the same as the original paper: while the performance gap between models is much small in terms of precision, DG-BERT is better than vanilla BERT in terms of recall, which means DG-BERT as a backbone model can capture more metaphors.

4.6. Investigation on collocations of phrases

From another perspective, also as a way to observe if DG-BERT can capture more accurate self-attention for collocation of phrases, we perform analyses on raw pre-trained models without fine-tuning via visualization (Vig, 2019) on self-attention. Given a sentence containing an idiomatic expression, we traverse all 12 encoder layers and their 12 heads, to search if there exists such a head that can:

- when the start word of an expression is selected as a pivot (the dot product between its query and keys of other words), whether semantically related words in the collocation can be given proper attention to it;
- semantically unrelated words do not or marginally attend to the start token.

The first ten sentences (containing idiomatic expressions with confidence score 1.0) are selected from MAGPIE (Haagsma et al., 2020) for comparison. For idiomatic expressions contained in these ten sentences, there are four of them whose definitions participate in the creation of DG embeddings. DG-BERT outperforms our vanilla BERT by six samples, and achieve four on-par performances. We show two samples ("off the beaten track" with participation and "chapter and verse" without participation) involving all analytic factors in Figures 8, 9, and 10.

We first compare the 1st and 2nd sub-figures of Figure 8. Semantically unrelated words are not or marginally given attention in both models. When it comes to allocating proper attention to the collocation, DG-BERT outperforms the vanilla BERT by giving more attention to "beaten" and

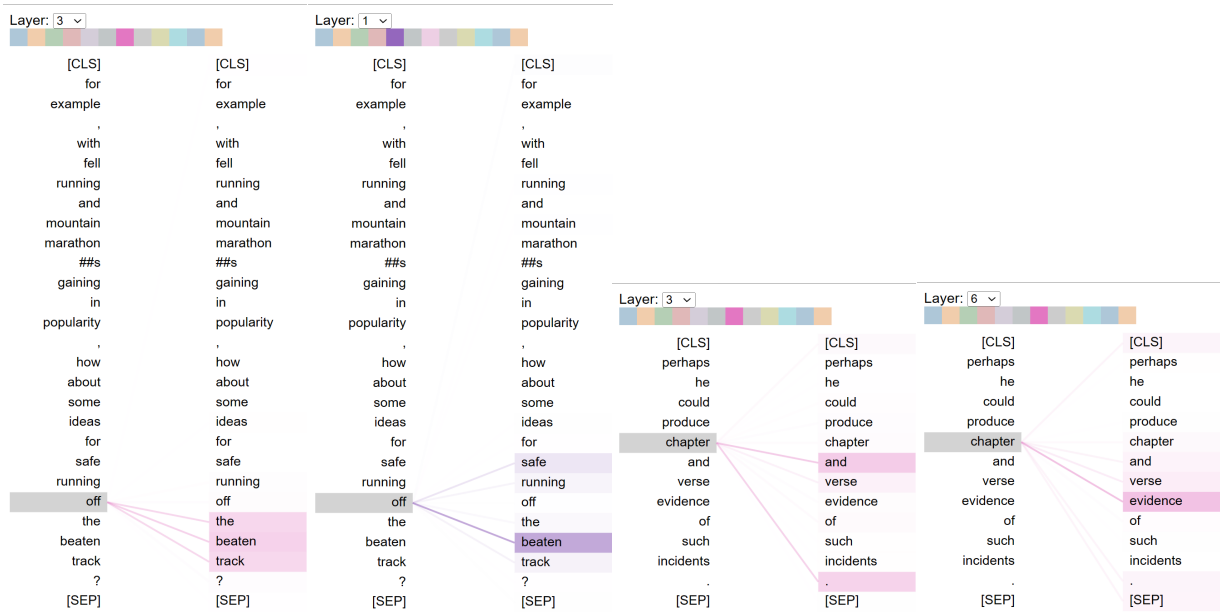

Figure 10: DG, scalar and bias embeddings are removed from pre-trained DG-BERT.

"track" than "the", because in this case, it can be distinguishable from other expressions like "off the road/track/etc.".

It can be argued that if the combination of "safe" and "running" should also be considered as semantically related words modified by the idiomatic expression. We find one head in DG-BERT (the 3rd sub-figure of Figure 8) where both words are given clear attention while proper attention to the collocation is further improved ("beaten" > "track" > "the", considering another expression "dangerous running off the track" is both syntactically and semantically justifiable), but fail to find one in our vanilla BERT ("safe" and "the" are weak in the 4th sub-figure of Figure 8).

From the 1st sub-figure of Figure 9, although DG-BERT has a flaw by giving a clear attention to ".", compared to our vanilla BERT (the 2nd sub-figure of Figure 9), it knows that "and" and "verse" are the words collocated with "chapter". In another head for DG-BERT (the 3rd sub-figure of Figure 9), "evidence" as a word modified by the idiomatic expression and "verse" as a key word collocated with "chapter" are given clear attentions. Despite the fact that the attention to "and" is marginal, considering it is a common conjunction (e.g., "chapter and paragraph" is also a legitimate expression), "verse" is comparatively crucial for the form of this collocation. There is no such a head in our vanilla BERT.

We also perform the visualization after removing DG, scalar and bias embeddings from pre-trained DG-BERT, and the results are shown in Figure 10. Compared to the ones before removing, the attentions become weaker. A proper allocation to the collocation words disappears in comparison between the 1st sub-figure of Figure 8 and the 1st sub-figure of Figure 10; the attentions to "track", "safe" and "running" in the 2nd sub-figure of Figure 10 are not as strong as the ones in the 3rd sub-figure of Figure 8; the attention to the key collocation word "verse" in the 3rd and 4th sub-figures of Figure 10 are not highlighted compared to the ones in the 1st and 3rd sub-figures of Figure 9. But they are still relatively better than our vanilla BERT; e.g., "safe" and "running" are preserved in the 2nd sub-figure of Figure 10 in comparison to the 4th sub-figure of Figure 8, and semantically unrelated words are given less attention in the 3rd sub-figure of Figure 10 than the ones in the 2nd sub-figure of Figure 9. We search other heads after removing, but do not find ones comparable to the originals. The results justify the important roles of our auxiliary embeddings. Especially for "chapter and verse" whose definition does not participate in the creation of DG embeddings, DG-BERT manages to learn the knowledge that three words form a collocation.

To delve deeper behind the scene, we make use of the ten samples for testing target-word predictions in collocation of phrases. The results are summarized in Table 8. In each sample, the word that is a comparatively critical part of a phrase is masked. Except in the second sample where "creeps" is not a single-word vocabulary of BERT, we mask "gave" instead. The results are generally consistent with the preliminary test presented in Table 3. In comparison between "DG-BERT w/o DG" and our vanilla BERT, the results show that the word embeddings of DG-BERT are better optimized than the ones of our vanilla BERT, given the same pre-training conditions. On the other hand, extrapolating from the results of first and third samples, the semantic features provided by DG embeddings might contain noise, and thus DG-BERT is outperformed by "DG-BERT w/o DG". Despite minorities, there is still room for improvement, which will be discussed in Section 6.

Table 8
The first ten sentences containing idiomatic expressions with confidence score 1.0 are selected from MAGPIE (Haagsma, Bos and Nissim, 2020) for target-word predictions in collocation of phrases. "DG-BERT w/o DG" denotes the whole package of DG, scalar and bias embeddings is removed from pre-trained DG-BERT. When the expected word is not included in top 5 predictions, - is used to indicate its ranking.

Input sequences	Ranking in top 5 predictions (softmax score)	Models
[CLS] For example, with fell running and mountain marathons gaining in popularity, how about some ideas for safe running off the [MASK] track? [SEP]	1 (0.4514)	DG-BERT
	1 (0.6255)	DG-BERT w/o DG
	1 (0.2965)	Vanilla BERT
[CLS] He [MASK] me the creeps, so I looked round, hmm hmm. [SEP]	2 (0.1604)	DG-BERT
	3 (0.0940)	DG-BERT w/o DG
	3 (0.0368)	Vanilla BERT
[CLS] 'He's done us [MASK], as well,' says Granville. [SEP]	- (0.0051)	DG-BERT
	- (0.0054)	DG-BERT w/o DG
	- (0.0009)	Vanilla BERT
[CLS] People quickly embraced formal democracy, but the tolerance and compromise that is at the heart of the democratic process took time to take [MASK]. [SEP]	2 (0.2119)	DG-BERT
	2 (0.1732)	DG-BERT w/o DG
	3 (0.1465)	Vanilla BERT
[CLS] Indeed, the Kursk crisis may provide him with an opportunity to further clean [MASK] in the military. [SEP]	4 (0.0194)	DG-BERT
	- (0.0107)	DG-BERT w/o DG
	- (0.0002)	Vanilla BERT
[CLS] A man has made medical [MASK] by having four organ transplants. [SEP]	1 (0.7121)	DG-BERT
	1 (0.6521)	DG-BERT w/o DG
	1 (0.3540)	Vanilla BERT
[CLS] So I went all the [MASK] down to the door slightly irritated, and what did I see? [SEP]	1 (0.9999)	DG-BERT
	1 (0.9998)	DG-BERT w/o DG
	1 (0.9997)	Vanilla BERT
[CLS] Perhaps he could produce chapter and [MASK] evidence of such incidents. [SEP]	1 (0.7475)	DG-BERT
	1 (0.6853)	DG-BERT w/o DG
	4 (0.0219)	Vanilla BERT
[CLS] By this time, some are up to 14 — but because the original outlay on stock didn't break the [MASK], Ken can offer them remarkably reasonably — priced. [SEP]	1 (0.6242)	DG-BERT
	1 (0.5768)	DG-BERT w/o DG
	4 (0.0384)	Vanilla BERT
[CLS] I'm still heading for the [MASK]. [SEP]	- (0.0180)	DG-BERT
	- (0.0162)	DG-BERT w/o DG
	- (0.0061)	Vanilla BERT

5. Additional comparison & further explanation on methodological advantage

We compare our results on GLUE benchmark with the most related work. The results are presented in Table 9, and we further explain the intrinsic advantage of our approach.

Although better performance is achieved, it can be argued that DG-BERT achieves the performance with the advantage of additional parameters of DG, scalar and bias embeddings. "DG-BERT w/o DG" shown in Table 9 clearly suggests that the performance is marginally diminished after removal. This is also consistent with the results regarding target-word predictions given context words presented in Tables 3, 8, and the psycholinguistic diagnostics in Appendix B, where "DG-BERT w/o DG" is slightly lower than DG-BERT, but noticeably higher than our vanilla BERT. Therefore, better performance is not caused by more parameters. Instead, the word embeddings of DG-BERT are comparatively well optimized with our approach. This can be proved by two observations and one additional visualization. First, in Section 4.2 describing pre-training convergence, DG-BERT is noticeably faster than vanilla BERT. Since the weight of decoder is tied to word embeddings, the acquisition of vocabulary for DG-BERT is more effective. Second, semantic information provided by our word embeddings contributes to better self-attention. When comparing "DG-BERT w/o DG" with our vanilla BERT in Figures 8, 9

Table 9
The experimental results of GLUE benchmark are reported on **validation set** for comparison with the most related previous work. "DG-BERT w/o DG" denotes that the whole package of DG, scalar and bias embeddings is removed from pre-trained DG-BERT. Our fine-tuning and Dict-BERT (Yu et al., 2022) are based on the average of five runs with different random seeds.

Models	MNLI (Acc.)	QQP (Acc.)	QNLI (Acc.)	SST2 (Acc.)	COLA (Matthews)	STSB (Pearson)	MRPC (Acc.)	RTE (Acc.)	Avg.	ΔAvg.
Vanilla BERT	84.12	90.75	90.69	92.52	58.59	89.39	86.17	68.67	82.65	-
Dict-BERT	84.34	90.81	91.20	92.65	**61.68**	89.68	87.21	72.89	83.80	+1.15
Vanilla BERT	83.92	90.90	91.20	91.92	58.50	89.59	86.76	69.34	82.77	-
DG-BERT w/o DG	84.85	91.00	91.22	**93.25**	58.70	90.01	87.75	**74.01**	83.88	+1.11
DG-BERT	**85.09**	**91.13**	**91.40**	93.00	60.14	**90.07**	**88.09**	73.94	**84.11**	**+1.34**

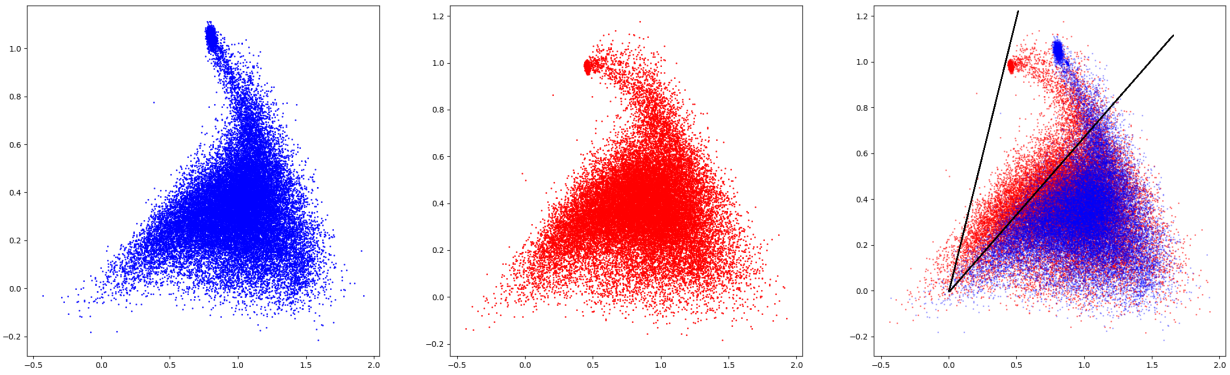

Figure 11: SVD visualization of token distribution of word embeddings from left to right → **1st:** word embeddings of our Vanilla BERT; **2nd:** word embeddings of DG-BERT; **3rd:** comparison between **1st** and **2nd**.

and 10, "DG-BERT w/o DG" achieves better self-attention than our vanilla BERT (detailed descriptions are given in Section 4.6). In language models, the input layer provides fundamental semantic information for attention layers. This can also be validated by the comparison between "DG-BERT w/o DG" and DG-BERT in those figures, as the only difference between them is in the input layer. Especially for collocations of phrases, the high phrasal internal similarities of DG embeddings shown in Tables 1 and 2 help our model to gain the advantage. Thus, the word embeddings of DG-BERT learn useful semantic information from context words glossed by DG embeddings. Third, word embeddings of language models are known to be anisotropic (Gao et al., 2019), but the problem is alleviated by our approach. As shown in Figure 11, the aperture size of the cone is comparatively increased — in the angle depicted in the **3rd** subfigure, noticeably more tokens are directionally distributed for the word embeddings of DG-BERT. This validates the assumption mentioned in Section 3.3. The distribution could be further improved if we adopt the regularization proposed by Gao et al. (2019). In this paper, we focus on the improvements made by our approach illustrated in Figure 3.

6. Discussion

Based on the experimental results, we conclude that our approach is relatively simple yet effective. If only to further improve task performance, this can be done by using more pre-training corpora and more careful pre-training schema like RoBERTa (Liu, Ott, Goyal, Du, Joshi, Chen, Levy, Lewis, Zettlemoyer and Stoyanov, 2019). This can also be achieved by augmenting self-attention mechanism such as using more encoder layers like BERT-large, or more advanced encoder structure like pre-layer normalization (Wang, Li, Xiao, Zhu, Li, Wong and Chao, 2019b; Xiong, Yang, He, Zheng, Zheng, Xing, Zhang, Lan, Wang and Liu, 2020). Additionally, ensembling methods can be considered, as our technical novelty does not conflict with any previous study. Recently, we have been planning to further pre-train DG-BERT by max sequence length 512 once the physical constraint is resolved, for future works that would like to apply DG-BERT to longer-sequence tasks, or to ensemble it with other models. We will upload a brief technical report after this research activity is completed. Our source code and current pre-trained model is publicly available[6].

[6] https://github.com/ryuliuxiaodong/DG-BERT

6.1. Further insight into research meaning

There are two contributions of our work stated in Section 1.4. Still, one might argue that the unsupervised definition embeddings mentioned in the second contribution are technically weak among modern technical trends, even though it is proved in this paper that they can assist context words for auto-encoding models. There might be other methods to create DG embeddings, as long as the creation of the embeddings is not subject to the technical duplicate mentioned in Section 1.3. However, in NLP domain, our approach offers a witness for the successful cooperation between two general AI research directions: Symbolism and Connectionism. Pre-training BERT belongs to Connectionism, which is based on deep learning of neural networks and big data. On the other hand, the creation of DG embeddings can be considered as an implementation of Symbolism. The extracted key features described in Section 3.1 are knowledge symbols. For dictionary entries having similar symbols in the term-document matrix, similar semantics are obtained in their vector representations. The outcome of their cooperation, DG-BERT, yields intrinsic advantages such as better target-word predictions given context words (especially for collocations of phrases) and alleviated anisotropic vector space of word embeddings, as well as better performance on downstream tasks than BERT.

6.2. Limitation

Following the convention of NLP community, we would also like to discuss the potential limitations of our approach, and how they can be mitigated.

DG embeddings. The method to create DG embeddings is inspired by the headword search in thesauruses, which is introduced in Section 1.3. Through our method described in Section 3.1, as shown in Figure 4, the key features extracted for each dictionary entry can generate overlap to represent similar semantics. However, some unnecessary key features are also extracted like “regard”, “relate” and so on. This could cause noise for DG embeddings, albeit there is no big negative impact inflicted on our later experimental settings. For future expansion, two solutions are conceivable for quality improvement. First, to scale down the threshold of extraction is an option, as in some other real-world scenarios, the 80/20 rule can be more effective after it is scaled down to 82/18. Namely, only top 18% features are considered as key features for extraction to avoid those unnecessary features. Second, as the implementations of Symbolism are typically rule-based (e.g., the 80/20 rule in our case), it can be investigated if there exists such a knowledge base that can be used to filter out those unnecessary key features.

Extensibility. DG embeddings are created for the MLM pre-training objective of BERT — glossing bidirectional context words for target-word prediction. Although there is no guarantee that DG embeddings can also assist the pre-training objectives of other language models, the advantages of DG embeddings ensures a pursuit worthy of exploration: 1) it is inexpensive for the creation (only single CPU device is enough); 2) DG embeddings can be removed from pre-trained models with performance slightly decreased (no occupation of additional memory resource). This is conducive to two scenarios of future expansion enumerated as follows.

1. One attempts to create DG embeddings using different dictionary resources.
2. One seeks to apply DG embeddings to different language models.

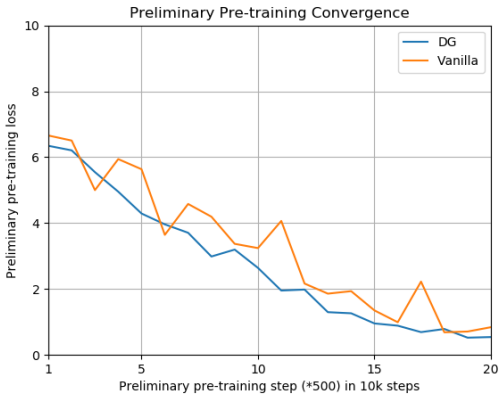

Figure 12: The convergence speed of preliminary pre-training (every 500-step batch loss in first 10k steps). This is performed to consider if full pre-training is worthy of pursuit. Compared to full pre-training, the setting is scaled down in terms of batch size 32, max-seq-length 128, approximately 20k input sequences, which can run on a single GPU device.

For the second research attempt, a preliminary pre-training (small batch size and corpora run on a single GPU) can be conducted to observe if convergence is faster than vanilla models. If so, full pre-training can be performed accordingly. This is what we had done before we performed full pre-training of DG-BERT. As shown in Figure 12, in the preliminary pre-training, DG-BERT is faster than vanilla BERT. Based on the observation, it can be expected that DG embeddings can assist BERT in acquiring vocabulary more effectively, and then full pre-training is performed. As a matter of fact, Figures 6 and 7 in Section 4.2 confirms that full pre-training meets our expectation. After full pre-training, the whole package of DG, and ancillary scalar and bias embeddings can be removed — no extra burden on memory.

7. Conclusion

In this paper, we investigated context words-glossed pre-training for the MLM pre-training objective of BERT, which is considered as the Cloze task for machines to learn words from bidirectional context words. We proposed DG embeddings to infuse context words with semantic features of glosses, and utilized scalar and bias embeddings to allow machines to learn how the semantic features represented in DG embeddings should be used. Our motivation, insights, and main claims were stated in Section 1. Some related

works were introduced in Section 1 and the rest were described in Section 2. The details regarding DG embeddings were provided in Section 3. We conducted a wide range of experiments in Section 4 to evaluate DG-BERT (vanilla BERT with DG, scalar and bias embeddings), and compared it with our vanilla BERT pre-trained based under the exact same condition. We also compared DG-BERT with the most related work in Section 5, and further explained the intrinsic advantage our approach. We discussed our further insight based on the experimental results in Section 6, along with the potential limitations of our approach and how they can be addressed.

CRediT authorship contribution statement

Xiaodong Liu: Conceptualization, Methodology, Investigation, and Writing - Original Draft. **Rafal Rzepka:** Writing - Review Editing. **Kenji Araki:** Supervision.

Declaration of competing interest

We, as the authors of this paper, declare that we have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Declaration of Generative AI and AI-assisted technologies in the writing process

We, as the authors of this paper, declare that there is no Generative AI and AI-assisted technologies involved in the writing process of this paper. From our idea to methodology and experiments, each part is completed by purely manual effort and human intelligence.

Ethical statement

We follow professional practice for NLP technology and the use of data proposed by Bender and Friedman (2018). Above all, this paper is written for purely academic demonstration, and the pre-training data that we use is limited when compared to modern language model pre-training. Nevertheless, there are potential concerns that such data could still be associated with gender bias, behavior patterns, bot masquerading as humans and so on. Although no explicit attempts have been made to identify those cases, preprocessing methods (Rae, Borgeaud, Cai, Millican, Hoffmann, Song, Aslanides, Henderson, Ring, Young et al., 2021) and manual labor can be employed to resolve the concerns if one seeks to apply our model to real-world use. In addition, the dictionary, glossary, and other evaluation datasets involved in this paper are publicly available and widely-used in NLP community. Except for the TOEFL dataset only available via portal request, we have signed an agreement with the dataset provider to use it only in not-for-profit academic researches. We will not leak it to public.

The images contained in Figures 1 and 3 are cited from https://publicdomainvectors.org/en/, where uploaders have waived all copyright to their vector images. There is no issue of using the images for non-profit academic papers.

Acknowledgement

This work was supported by JST SPRING, Grant Number JPMJSP2119. We are grateful to the valuable comments from the reviewers, which helped us to improve the quality of this paper.

References

Aaron Gokaslan*, Vanya Cohen*, E.P.S.T., 2019. Openwebtext corpus. http://Skylion007.github.io/OpenWebTextCorpus.

AbuSeileek, A.F., 2011. Hypermedia annotation presentation: The effect of location and type on the efl learners' achievement in reading comprehension and vocabulary acquisition. Computers Education 57, 1281–1291. URL: https://www.sciencedirect.com/science/article/pii/S0360131511000285, doi:https://doi.org/10.1016/j.compedu.2011.01.011.

Apidianaki, M., 2023. From Word Types to Tokens and Back: A Survey of Approaches to Word Meaning Representation and Interpretation. Computational Linguistics , 1–58URL: https://doi.org/10.1162/coli_a_00474, doi:10.1162/coli_a_00474.

Baker, C.F., Fillmore, C.J., Lowe, J.B., 1998. The Berkeley FrameNet project, in: 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1, Association for Computational Linguistics, Montreal, Quebec, Canada. pp. 86–90. URL: https://aclanthology.org/P98-1013, doi:10.3115/980845.980860.

Balazevic, I., Allen, C., Hospedales, T., 2019. TuckER: Tensor factorization for knowledge graph completion, 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), Association for Computational Linguistics, Hong Kong, China. pp. 5185–5194. URL: https://aclanthology.org/D19-1522, doi:10.18653/v1/D19-1522.

Beltagy, I., Lo, K., Cohan, A., 2019. SciBERT: A pretrained language model for scientific text, 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), Association for Computational Linguistics, Hong Kong, China. pp. 3615–3620. URL: https://aclanthology.org/D19-1371, doi:10.18653/v1/D19-1371.

Bender, E.M., Friedman, B., 2018. Data statements for natural language processing: Toward mitigating system bias and enabling better science. Transactions of the Association for Computational Linguistics 6, 587–604. URL: https://aclanthology.org/Q18-1041, doi:10.1162/tacl_a_00041.

Birke, J., Sarkar, A., 2006. A clustering approach for nearly unsupervised recognition of nonliteral language, in: 11th Conference of the European Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, Trento, Italy. pp. 329–336. URL: https://aclanthology.org/E06-1042.

Blei, D.M., Ng, A.Y., Jordan, M.I., 2003. Latent dirichlet allocation. Journal of machine Learning research 3, 993–1022.

Blevins, T., Zettlemoyer, L., 2020. Moving down the long tail of word sense disambiguation with gloss informed bi-encoders, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online. pp. 1006–1017. URL: https://aclanthology.org/2020.acl-main.95, doi:10.18653/v1/2020.acl-main.95.

Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O., 2013. Translating embeddings for modeling multi-relational data, in: Burges, C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K. (Eds.), Advances in Neural Information Processing Systems, Curran

Associates, Inc. URL: https://proceedings.neurips.cc/paper/2013/file/1cecc7a77928ca8133fa24680a88d2f9-Paper.pdf.

Bosc, T., Vincent, P., 2018. Auto-encoding dictionary definitions into consistent word embeddings, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Brussels, Belgium. pp. 1522–1532. URL: https://aclanthology.org/D18-1181, doi:10.18653/v1/D18-1181.

Chang, K.W., Yih, W.t., Meek, C., 2013. Multi-relational latent semantic analysis, in: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Seattle, Washington, USA. pp. 1602–1612. URL: https://aclanthology.org/D13-1167.

Chen, Q., Li, F.L., Xu, G., Yan, M., Zhang, J., Zhang, Y., 2022. Dictbert: Dictionary description knowledge enhanced language model pre-training via contrastive learning, in: IJCAI, pp. 4086–4092. URL: https://doi.org/10.24963/ijcai.2022/567.

Choi, M., Lee, S., Choi, E., Park, H., Lee, J., Lee, D., Lee, J., 2021. MelBERT: Metaphor detection via contextualized late interaction using metaphorical identification theories, in: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Online. pp. 1763–1773. URL: https://aclanthology.org/2021.naacl-main.141, doi:10.18653/v1/2021.naacl-main.141.

Chow, W.Y., Smith, C., Lau, E., Phillips, C., 2016. A "bag-of-arguments" mechanism for initial verb predictions. Language, Cognition and Neuroscience 31, 577–596.

Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R., 1990. Indexing by latent semantic analysis. Journal of the American Society for Information Science 41, 391–407. doi:https://doi.org/10.1002/(SICI)1097-4571(199009)41:6<391::AID-ASI1>3.0.CO;2-9.

Devlin, J., Chang, M.W., Lee, K., Toutanova, K., 2019. BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota. pp. 4171–4186. URL: https://aclanthology.org/N19-1423, doi:10.18653/v1/N19-1423.

Dolan, B., Quirk, C., Brockett, C., 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources, in: COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics, COLING, Geneva, Switzerland. pp. 350–356. URL: https://aclanthology.org/C04-1051.

Ethayarajh, K., 2019. How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings, in: Inui, K., Jiang, J., Ng, V., Wan, X. (Eds.), Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China. pp. 55–65. URL: https://aclanthology.org/D19-1006, doi:10.18653/v1/D19-1006.

Ettinger, A., 2020. What BERT is not: Lessons from a new suite of psycholinguistic diagnostics for language models. Transactions of the Association for Computational Linguistics 8, 34–48. URL: https://aclanthology.org/2020.tacl-1.3, doi:10.1162/tacl_a_00298.

Faruqui, M., Dodge, J., Jauhar, S.K., Dyer, C., Hovy, E., Smith, N.A., 2015. Retrofitting word vectors to semantic lexicons, in: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Denver, Colorado. pp. 1606–1615. URL: https://aclanthology.org/N15-1184, doi:10.3115/v1/N15-1184.

Federmeier, K.D., Kutas, M., 1999. A rose by any other name: Long-term memory structure and sentence processing. Journal of memory and Language 41, 469–495.

Firth, J.R., 1957. A synopsis of linguistic theory, 1930-1955. Studies in linguistic analysis .

Ganitkevitch, J., Van Durme, B., Callison-Burch, C., 2013. PPDB: The paraphrase database, in: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Atlanta, Georgia. pp. 758–764. URL: https://aclanthology.org/N13-1092.

Gao, J., He, D., Tan, X., Qin, T., Wang, L., Liu, T., 2019. Representation degeneration problem in training natural language generation models, in: International Conference on Learning Representations. URL: https://openreview.net/forum?id=SkEYojRqtm.

Glavaš, G., Vulić, I., 2018. Explicit retrofitting of distributional word vectors, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Melbourne, Australia. pp. 34–45. URL: https://aclanthology.org/P18-1004, doi:10.18653/v1/P18-1004.

Golub, G.H., Van Loan, C.F., 1996. Matrix Computations (3rd Ed.). Johns Hopkins University Press, USA.

Gururangan, S., Marasović, A., Swayamdipta, S., Lo, K., Beltagy, I., Downey, D., Smith, N.A., 2020. Don't stop pretraining: Adapt language models to domains and tasks, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online. pp. 8342–8360. URL: https://aclanthology.org/2020.acl-main.740, doi:10.18653/v1/2020.acl-main.740.

Haagsma, H., Bos, J., Nissim, M., 2020. MAGPIE: A large corpus of potentially idiomatic expressions, in: Proceedings of the Twelfth Language Resources and Evaluation Conference, European Language Resources Association, Marseille, France. pp. 279–287. URL: https://aclanthology.org/2020.lrec-1.35.

Harris, Z.S., 1954. Distributional structure. Word 10, 146–162.

He, Y., Wang, Z., Zhang, Y., Huang, R., Caverlee, J., 2020. PARADE: A New Dataset for Paraphrase Identification Requiring Computer Science Domain Knowledge, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Online. pp. 7572–7582. URL: https://aclanthology.org/2020.emnlp-main.611, doi:10.18653/v1/2020.emnlp-main.611.

Hill, F., Cho, K., Korhonen, A., Bengio, Y., 2016. Learning to understand phrases by embedding the dictionary. Transactions of the Association for Computational Linguistics 4, 17–30. URL: https://aclanthology.org/Q16-1002, doi:10.1162/tacl_a_00080.

Huang, L., Sun, C., Qiu, X., Huang, X., 2019. GlossBERT: BERT for word sense disambiguation with gloss knowledge, 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), Association for Computational Linguistics, Hong Kong, China. pp. 3509–3514. URL: https://aclanthology.org/D19-1355, doi:10.18653/v1/D19-1355.

Inan, H., Khosravi, K., Socher, R., 2017. Tying word vectors and word classifiers: A loss framework for language modeling, in: International Conference on Learning Representations. URL: https://openreview.net/forum?id=r1aPbsFle.

Ishiwatari, S., Hayashi, H., Yoshinaga, N., Neubig, G., Sato, S., Toyoda, M., Kitsuregawa, M., 2019. Learning to describe unknown phrases with local and global contexts, 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), Association for Computational Linguistics, Minneapolis, Minnesota. pp. 3467–3476. URL: https://aclanthology.org/N19-1350, doi:10.18653/v1/N19-1350.

Jung, J., 2016. Effects of glosses on learning of l2 grammar and vocabulary. Language Teaching Research 20, 92–112. URL: https://doi.org/10.1177/1362168815571151, doi:10.1177/1362168815571151, arXiv:https://doi.org/10.1177/1362168815571151.

Ke, G., He, D., Liu, T.Y., 2021. Rethinking positional encoding in language pre-training, in: International Conference on Learning Representations. URL: https://openreview.net/forum?id=09-528y2Fgf.

Knight, S., 1994. Dictionary use while reading: The effects on comprehension and vocabulary acquisition for students of different verbal abilities. The Modern Language Journal 78, 285–299.

Landauer, T., Laham, D., Foltz, P., 1997. Learning human-like knowledge by singular value decomposition: A progress report, in: Jordan, M., Kearns, M., Solla, S. (Eds.), Advances in Neural Information Processing Systems, MIT Press. URL: https://proceedings.neurips.cc/paper/1997/file/cec6f62cfb44b1be110b7bf70c8362d8-Paper.pdf.

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

Leong, C.W.B., Beigman Klebanov, B., Hamill, C., Stemle, E., Ubale, R., Chen, X., 2020. A report on the 2020 VUA and TOEFL metaphor detection shared task, in: Proceedings of the Second Workshop on Figurative Language Processing, Association for Computational Linguistics, Online. pp. 18–29. URL: https://aclanthology.org/2020.figlang-1.3, doi:10.18653/v1/2020.figlang-1.3.

Leong, C.W.B., Beigman Klebanov, B., Shutova, E., 2018. A report on the 2018 VUA metaphor detection shared task, in: Proceedings of the Workshop on Figurative Language Processing, Association for Computational Linguistics, New Orleans, Louisiana. pp. 56–66. URL: https://aclanthology.org/W18-0907, doi:10.18653/v1/W18-0907.

Levine, Y., Lenz, B., Dagan, O., Ram, O., Padnos, D., Sharir, O., Shalev-Shwartz, S., Shashua, A., Shoham, Y., 2020. SenseBERT: Driving some sense into BERT, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online. pp. 4656–4667. URL: https://aclanthology.org/2020.acl-main.423, doi:10.18653/v1/2020.acl-main.423.

Lin, Y., An, Z., Wu, P., Ma, Z., 2022. Improving contextual representation with gloss regularized pre-training, in: Findings of the Association for Computational Linguistics: NAACL 2022, Association for Computational Linguistics, Seattle, United States. pp. 907–920. URL: https://aclanthology.org/2022.findings-naacl.68, doi:10.18653/v1/2022.findings-naacl.68.

Liu, X., Rzepka, R., Araki, K., 2022. Parameter-efficient feature-based transfer for paraphrase identification. Natural Language Engineering , 1–31doi:10.1017/S135132492200050X.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V., 2019. Roberta: A robustly optimized BERT pretraining approach. CoRR abs/1907.11692. URL: http://arxiv.org/abs/1907.11692, arXiv:1907.11692.

Maćkiewicz, A., Ratajczak, W., 1993. Principal components analysis (pca). Computers Geosciences 19, 303–342. URL: https://www.sciencedirect.com/science/article/pii/009830049390090R, doi:https://doi.org/10.1016/0098-3004(93)90090-R.

McGregor, W.B., 2015. Linguistics: an introduction. Bloomsbury Publishing.

Micikevicius, P., Narang, S., Alben, J., Diamos, G., Elsen, E., Garcia, D., Ginsburg, B., Houston, M., Kuchaiev, O., Venkatesh, G., Wu, H., 2018. Mixed precision training, in: International Conference on Learning Representations. URL: https://openreview.net/forum?id=r1gs9JgRZ.

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

Miller, G.A., 1998. WordNet: An electronic lexical database. MIT press.

Mohammad, S., Shutova, E., Turney, P., 2016. Metaphor as a medium for emotion: An empirical study, in: Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics, Association for Computational Linguistics, Berlin, Germany. pp. 23–33. URL: https://aclanthology.org/S16-2003, doi:10.18653/v1/S16-2003.

Nesi, H., 2012. Dictionary use, in: The encyclopedia of applied linguistics. Wiley.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S., 2019. Pytorch: An imperative style, high-performance deep learning library, in: Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R. (Eds.), Advances in Neural Information Processing Systems, Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf.

Peters, M.E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Zettlemoyer, L., 2018. Deep contextualized word representations, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), Association for Computational Linguistics, New Orleans, Louisiana. pp. 2227–2237. URL: https://aclanthology.org/N18-1202, doi:10.18653/v1/N18-1202.

Peters, M.E., Neumann, M., Logan, R., Schwartz, R., Joshi, V., Singh, S., Smith, N.A., 2019. Knowledge enhanced contextual 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), Association for Computational Linguistics, Hong Kong, China. pp. 43–54. URL: https://aclanthology.org/D19-1005, doi:10.18653/v1/D19-1005.

Press, O., Wolf, L., 2017. Using the output embedding to improve language models, in: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers, Association for Computational Linguistics, Valencia, Spain. pp. 157–163. URL: https://aclanthology.org/E17-2025.

Rae, J.W., Borgeaud, S., Cai, T., Millican, K., Hoffmann, J., Song, F., Aslanides, J., Henderson, S., Ring, R., Young, S., et al., 2021. Scaling language models: Methods, analysis & insights from training gopher. arXiv preprint arXiv:2112.11446 .

Rajpurkar, P., Jia, R., Liang, P., 2018. Know what you don't know: Unanswerable questions for SQuAD, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Association for Computational Linguistics, Melbourne, Australia. pp. 784–789. URL: https://aclanthology.org/P18-2124, doi:10.18653/v1/P18-2124.

Rajpurkar, P., Zhang, J., Lopyrev, K., Liang, P., 2016. SQuAD: 100,000+ questions for machine comprehension of text, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Austin, Texas. pp. 2383–2392. URL: https://aclanthology.org/D16-1264, doi:10.18653/v1/D16-1264.

Salton, G., Wong, A., Yang, C.S., 1975. A vector space model for automatic indexing. Commun. ACM 18, 613–620. URL: https://doi.org/10.1145/361219.361220, doi:10.1145/361219.361220.

Sap, M., Le Bras, R., Allaway, E., Bhagavatula, C., Lourie, N., Rashkin, H., Roof, B., Smith, N.A., Choi, Y., 2019. Atomic: An atlas of machine commonsense for if-then reasoning, in: Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, AAAI Press. URL: https://doi.org/10.1609/aaai.v33i01.33013027, doi:10.1609/aaai.v33i01.33013027.

Shi, W., Chen, M., Zhou, P., Chang, K.W., 2019. Retrofitting contextualized word embeddings with paraphrases, 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), Association for Computational Linguistics, Hong Kong, China. pp. 1198–1203. URL: https://aclanthology.org/D19-1113, doi:10.18653/v1/D19-1113.

Speer, R., Chin, J., Havasi, C., 2017. Conceptnet 5.5: An open multilingual graph of general knowledge, in: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, AAAI Press. p. 4444–4451.

Sun, T., Shao, Y., Qiu, X., Guo, Q., Hu, Y., Huang, X., Zhang, Z., 2020. CoLAKE: Contextualized language and knowledge embedding, in: Proceedings of the 28th International Conference on Computational Linguistics, International Committee on Computational Linguistics, Barcelona, Spain (Online). pp. 3660–3670. URL: https://aclanthology.org/2020.coling-main.327, doi:10.18653/v1/2020.coling-main.327.

Sylvain Gugger, Lysandre Debut, T.W.P.S.Z.M.S.M., 2022. Accelerate: Training and inference at scale made simple, efficient and adaptable. https://github.com/huggingface/accelerate.

Taylor, W.L., 1953. "cloze procedure": A new tool for measuring readability. Journalism quarterly 30, 415–433.

Tissier, J., Gravier, C., Habrard, A., 2017. Dict2vec : Learning word embeddings using lexical dictionaries, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Copenhagen, Denmark. pp. 254–263. URL: https://aclanthology.org/D17-1024, doi:10.18653/v1/D17-1024.

Tjong Kim Sang, E.F., De Meulder, F., 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition, in: Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003, pp. 142–147. URL: https://aclanthology.org/W03-0419.

Turney, P.D., Pantel, P., 2010. From frequency to meaning: Vector space models of semantics. J. Artif. Int. Res. 37, 141–188.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need, in: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (Eds.), Advances in Neural Information Processing Systems, Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Vig, J., 2019. A multiscale visualization of attention in the transformer model, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, Association for Computational Linguistics, Florence, Italy. pp. 37–42. URL: https://www.aclweb.org/anthology/P19-3007, doi:10.18653/v1/P19-3007.

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., Bowman, S.R., 2019a. GLUE: A multi-task benchmark and analysis platform for natural language understanding, in: International Conference on Learning Representations. URL: https://openreview.net/forum?id=rJ4km2R5t7.

Wang, Q., Li, B., Xiao, T., Zhu, J., Li, C., Wong, D.F., Chao, L.S., 2019b. Learning deep transformer models for machine translation, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy. pp. 1810–1822. URL: https://aclanthology.org/P19-1176, doi:10.18653/v1/P19-1176.

Wang, X., Gao, T., Zhu, Z., Zhang, Z., Liu, Z., Li, J., Tang, J., 2021. KEPLER: A Unified Model for Knowledge Embedding and Pre-trained Language Representation. Transactions of the Association for Computational Linguistics 9, 176–194. URL: https://doi.org/10.1162/tacl_a_00360, doi:10.1162/tacl_a_00360.

Wege, M.V.D., 2018. Teaching strategies for independent vocabulary development. The TESOL Encyclopedia of English Language Teaching , 1–7.

Weijie Liu, Peng Zhou, Z.Z.Z.W.Q.J.H.D.P.W., 2020. K-BERT: Enabling language representation with knowledge graph, in: Proceedings of AAAI 2020.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., Drame, M., Lhoest, Q., Rush, A., 2020. Transformers: State-of-the-art natural language processing, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics, Online. pp. 38–45. URL: https://aclanthology.org/2020.emnlp-demos.6, doi:10.18653/v1/2020.emnlp-demos.6.

Wu, Q., Xing, C., Li, Y., Ke, G., He, D., Liu, T.Y., 2021. Taking notes on the fly helps language pre-training, in: International Conference on Learning Representations. URL: https://openreview.net/forum?id=lU5Rs_wCweN.

Xiong, R., Yang, Y., He, D., Zheng, K., Zheng, S., Xing, C., Zhang, H., Lan, Y., Wang, L., Liu, T., 2020. On layer normalization in the transformer architecture, in: Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event, PMLR. pp. 10524–10533. URL: http://proceedings.mlr.press/v119/xiong20b.html.

Yih, W.t., Zweig, G., Platt, J.C., 2012. Polarity inducing latent semantic analysis, in: Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, Association for Computational Linguistics, USA. p. 1212–1222.

Table 10
The hyperparameters for pre-training and fine-tuning DG-BERT are summarized in this table.

Hyperparameters	Pre-training	Fine-tuning
Total Steps	1M	-
Total Epochs	40	{3, 10}
Learning Rate	1e-4	{2e-5, 3e-5, 4e-5}
Batch Size	256	{12, 32}
Warm-up Ratio	0.01	-
Max Sequence Length	448	{128, 384}
Learning Rate Decay	Linear	Linear
Adam ϵ	1e-6	1e-6
Adam (β_1, β_2)	(0.9, 0.999)	(0.9, 0.999)
Dropout	0.1	0.1
Weight Decay	0.01	0.01
Mixed Precision	fp16	fp16

Yin, W., Schütze, H., 2015. Convolutional neural network for paraphrase identification, in: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Denver, Colorado. pp. 901–911. URL: https://www.aclweb.org/anthology/N15-1091, doi:10.3115/v1/N15-1091.

Yu, M., Dredze, M., 2014. Improving lexical embeddings with semantic knowledge, in: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Association for Computational Linguistics, Baltimore, Maryland. pp. 545–550. URL: https://aclanthology.org/P14-2089, doi:10.3115/v1/P14-2089.

Yu, W., Zhu, C., Fang, Y., Yu, D., Wang, S., Xu, Y., Zeng, M., Jiang, M., 2022. Dict-BERT: Enhancing language model pre-training with dictionary, in: Findings of the Association for Computational Linguistics: ACL 2022, Association for Computational Linguistics, Dublin, Ireland. pp. 1907–1918. URL: https://aclanthology.org/2022.findings-acl.150, doi:10.18653/v1/2022.findings-acl.150.

Zellers, R., Bisk, Y., Schwartz, R., Choi, Y., 2018. SWAG: A large-scale adversarial dataset for grounded commonsense inference, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Brussels, Belgium. pp. 93–104. URL: https://aclanthology.org/D18-1009, doi:10.18653/v1/D18-1009.

Zhang, Y., Baldridge, J., He, L., 2019a. PAWS: Paraphrase Adversaries from Word Scrambling, in: Proc. of NAACL.

Zhang, Z., Han, X., Liu, Z., Jiang, X., Sun, M., Liu, Q., 2019b. ERNIE: Enhanced language representation with informative entities, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy. pp. 1441–1451. URL: https://aclanthology.org/P19-1139, doi:10.18653/v1/P19-1139.

Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., Fidler, S., 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books, in: The IEEE International Conference on Computer Vision (ICCV).

A. Hyperparameters

Pre-training and fine-tuning hyperparameters are listed in Table 10. For fine-tuning, we follow previous works (Wu et al., 2021; Ke, He and Liu, 2021) to set 10 epochs for small training sets. As for the batch size used in those sets, compared to 8 and 16 for our vanilla BERT, 12 can produce more stable performance for DG-BERT. Max sequence

Table 11
Models are compared on CPRAG-102 word prediction accuracies, and corresponding sensitivity tests: prefer expected completions over inappropriate completions.

Models	Percentage of correct prediction (Top 1 / Top 5)	Percentage of preferring good	w/ 0.01 threshold
Vanilla BERT	11.8 / 38.2	64.7	35.3
DG-BERT	17.7 / 47.1	67.7	47.1
DG-BERT w/o DG	14.7 / 41.2	64.7	44.1

Table 12
Models are compared on ROLE-88 word prediction accuracies, and corresponding sensitivity tests: prefer expected completions over inappropriate completions.

Models	Percentage of correct prediction (Top 1 / Top 5)	Percentage of preferring good	w/ 0.01 threshold
Vanilla BERT	12.5 / 29.5	70.5	34.1
DG-BERT	14.8 / 36.4	81.8	40.9
DG-BERT w/o DG	14.8 / 30.7	75.0	36.4

length of 384 is used for long-sequence question answering tasks. When we apply our models as backbone models to MelBERT (Choi et al., 2021), we use their default sequence length 150.

B. Psycholinguistic diagnostics

To verify if DG-BERT is also sensitive to target-word predictions in the context without collocation, we perform two tests utilized in the psycholinguistic diagnostics proposed by Ettinger (2020). CPRAG-102 stems from an influential human study Federmeier and Kutas (1999) to test how brains respond to different types of context completions. ROLE-88 originates from a human experiment (Chow, Smith, Lau and Phillips, 2016) for testing the brain's sensitivity to role reversals. For the single-word vocabulary of BERT, 34 contexts from CPRAG-102 and 44 context pairs from ROLE-88 are selected by Ettinger (2020).

The results are presented from Tables 11 to 18. In Tables 11 and 12, the value under the column of "preferring good" is calculated based on the softmax scores where expected word is higher than inappropriate words (good completion), regardless of top 5 predictions; the "w/ 0.01 threshold" indicates that the difference of softmax scores in good completion is greater than 0.01. The overall performance is consistent with the target-word predictions in the context with collocation shown in the previous subsection: DG-BERT > "DG-BERT w/o DG" > vanilla BERT. DG-BERT achieve comparatively good predictions on the difficult samples selected by Ettinger (2020) shown from Tables 13 to 18. It can be concluded that DG-BERT is comparatively sensitive to context words for target-word predictions. The performance on ROLE-88 further explains the reason why DG-BERT can achieve better results on the PAWS-wiki task (see Table 6).

Table 13
Top word predictions are selected for the CPRAG-102 items by Ettinger (2020). The underlined predictions are the expected ones.

Context	Vanilla BERT predictions
Pablo wanted to cut the lumber he had bought to make some shelves. He asked his neighbor if he could borrow her ____.	lumber, furniture, wood, house, equipment
The snow had piled up on the drive so high that they couldn't get the car out. When Albert woke up, his father handed him a ____.	gun, key, hug, shovel, ride
At the zoo, my sister asked if they painted the black and white stripes on the animal. I explained to her that they were natural features of a ____.	zoo, animal, dog, bird, lion

Table 14
Top word predictions are selected for the CPRAG-102 items by Ettinger (2020). The underlined predictions are the expected ones.

Context	DG-BERT predictions
Pablo wanted to cut the lumber he had bought to make some shelves. He asked his neighbor if he could borrow her ____.	house, car, money, truck, tractor
The snow had piled up on the drive so high that they couldn't get the car out. When Albert woke up, his father handed him a ____.	shovel, package, bag, letter, gun
At the zoo, my sister asked if they painted the black and white stripes on the animal. I explained to her that they were natural features of a ____.	deer, cat, zebra, spider, tiger

Table 15
Top word predictions are selected for the CPRAG-102 items by Ettinger (2020). The underlined predictions are the expected ones.

Context	DG-BERT w/o DG predictions
Pablo wanted to cut the lumber he had bought to make some shelves. He asked his neighbor if he could borrow her ____.	house, car, tractor, lumber, money
The snow had piled up on the drive so high that they couldn't get the car out. When Albert woke up, his father handed him a ____.	shovel, package, gun, bag, phone
At the zoo, my sister asked if they painted the black and white stripes on the animal. I explained to her that they were natural features of a ____.	cat, spider, human, zebra, deer

Table 16
Top word predictions are selected for the ROLE-88 items by Ettinger (2020). The underlined predictions are the expected ones. * denotes inappropriate words.

Context	Vanilla BERT predictions
the camper reported which bear the girl had ____.	eaten, shot, killed, bitten, become
the camper reported which girl the bear had ____.	bitten, *, taken, attacked, kidnapped
the restaurant owner forgot which waitress the customer had ____.	ordered, asked, met, hired, been
the restaurant owner forgot which customer the waitress had ____.	been, served, met, seen, asked

Table 17
Top word predictions are selected for the ROLE-88 items by Ettinger (2020). The underlined predictions are the expected ones. * denotes inappropriate words.

Context	DG-BERT predictions
the camper reported which bear the girl had ____.	killed, eaten, seen, hunted, shot
the camper reported which girl the bear had ____.	killed, *, eaten, hit, bitten
the restaurant owner forgot which waitress the customer had ____.	hired, chosen, been, booked, requested
the restaurant owner forgot which customer the waitress had ____.	been, served, chosen, become, had

Table 18
Top word predictions are selected for the ROLE-88 items by Ettinger (2020). The underlined predictions are the expected ones. * denotes inappropriate words.

Context	DG-BERT w/o DG predictions
the camper reported which bear the girl had ____.	killed, eaten, seen, hunted, hit
the camper reported which girl the bear had ____.	killed, bitten, *, eaten, hit
the restaurant owner forgot which waitress the customer had ____.	been, hired, chosen, booked, wanted
the restaurant owner forgot which customer the waitress had ____.	been, served, chosen, become, had