

Highlights

Embracing Ambiguity: Improving Similarity-oriented Tasks with Contextual Synonym Knowledge

Yangning Li, Jiaoyan Chen, Yinghui Li, Tianyu Yu, Xi Chen, Hai-Tao Zheng

- Contextual synonym knowledge is extremely effective for similarity-oriented tasks, and we are the first work to inject **contextual** synonym knowledge into the Pre-trained Language Model (PLM).
- We propose PICSO, a flexible framework equipped with our designed entity-aware Adapter. PICSO supports continuous injection of synonym knowledge from multiple domains, while the contextual semantic understanding capability of the original PLM is not undermined.
- PICSO can dramatically outperform the original PLMs and the other knowledge and synonym injection models on various similarity-oriented tasks. In addition, PICSO also benefits general natural language understanding tasks.

Embracing Ambiguity: Improving Similarity-oriented Tasks with Contextual Synonym Knowledge

Yangning Li^a, Jiaoyan Chen^b, Yinghui Li^a, Tianyu Yu^a, Xi Chen^c and Hai-Tao Zheng^{a,d,*}

^aShenzhen International Graduate School, Tsinghua University, Shenzhen, 518055, Guangdong, China

^bDepartment of Computer Science, The University of Manchester, Manchester, M13 9PL, UK

^cPlatform and Content Group, Tencent, Shenzhen, 518055, Guangdong, China

^dPeng Cheng Laboratory, Shenzhen, 518055, Guangdong, China

ARTICLE INFO

Keywords:

Natural Language Processing

Pre-trained Language Model

Similarity-oriented Tasks

Synonym Knowledge Enhancement

ABSTRACT

Contextual synonym knowledge is crucial for those similarity-oriented tasks whose core challenge lies in capturing semantic similarity between entities in their contexts, such as entity linking and entity matching. However, most Pre-trained Language Models (PLMs) lack synonym knowledge due to inherent limitations of their pre-training objectives such as masked language modeling (MLM). Existing works which inject synonym knowledge into PLMs often suffer from two severe problems: (i) Neglecting the ambiguity of synonyms, and (ii) Undermining semantic understanding of original PLMs, which is caused by inconsistency between the *exact semantic similarity* of the synonyms and the *broad conceptual relevance* learned from the original corpus. To address these issues, we propose PICSO, a flexible framework that supports the injection of contextual synonym knowledge from multiple domains into PLMs via a novel entity-aware Adapter which focuses on the semantics of the entities (synonyms) in the contexts. Meanwhile, PICSO stores the synonym knowledge in additional parameters of the Adapter structure, which prevents it from corrupting the semantic understanding of the original PLM. Extensive experiments demonstrate that PICSO can dramatically outperform the original PLMs and the other knowledge and synonym injection models on four different similarity-oriented tasks. In addition, experiments on GLUE prove that PICSO also benefits general natural language understanding tasks. Codes and data will be public.

1. Introduction

Pre-trained language models (PLMs) such as BERT [17], RoBERTa [27] and GPT [38] have achieved great success in natural language processing (NLP) due to their semantic understanding capabilities achieved by pre-training on large-scale corpora. However, most PLMs only acquire statistical word co-occurrence knowledge through their pre-training objectives such as masked language modeling (MLM) [22], which leads to limited capabilities in understanding synonyms.

Synonym knowledge facilitates models to capture fine-grained semantic relations and is crucial in NLP especially for addressing **similarity-oriented tasks**, such as entity linking [54] and entity resolution [49]. The core challenge of such tasks lies in modeling the semantic similarity of entities in complex contexts, where understanding the synonymous relationship between phrases is essential. Taking ontology alignment as an example, cross-ontology class pairs with synonymic relationships account for 51% of the total class mappings in the widely used benchmark FMA-SNOMED of OAEI LargeBio Track¹. In knowledge graph (KG) canonicalization, which aims to cluster semantically identical entities, about 30% identical entities in the Reverb45k dataset [43] appear in the synonym sets (synsets)

of UMLS². In specific domains, making sense of synonym knowledge could become even more important and challenging. As the biomedical example in Figure 1 shows, *Elephantiasis* is synonymous to *Lymphatic filariasis* but non-synonymous to *Elephantiasis graecorum*, although the latter has a closer surface form and would be regarded as synonymous by a normal PLM. To better address such similarity-oriented tasks, capturing the synonymous relationship between phrases under complex contexts is urgently required. This motivates us to inject synonym knowledge into PLMs.

Some pioneering works have explored injecting synonym knowledge into PLMs. LIBERT [19] trains BERT from scratch with an auxiliary task that binary classifies whether entity pairs are synonymous pairs. SAPBERT [25] pre-trains BERT with synsets from UMLS. A metric learning objective is used to optimize the BERT, with synonymous and non-synonymous entity pairs as positive and negative training samples, respectively. Although promising results have been achieved, these works still suffer from the following two problems:

Neglecting the ambiguity of synonyms. Synonyms are naturally context-sensitive. It is intuitive that some entities are synonyms and thus close to each other in the semantic space, but are comparably different in some specific aspects and thus far away from each other in the corresponding semantic spaces. However, pre-training objectives of the current synonym injection works completely ignore this

*Corresponding author

✉ liyn20@mails.tsinghua.edu.cn (Y. Li);

jiaoyan.chen@manchester.ac.uk (J. Chen);

zheng.haitao@sz.tsinghua.edu.cn (H. Zheng)

ORCID(s): 0000-0002-1991-6698 (Y. Li)

¹<https://www.cs.ox.ac.uk/isg/projects/SEALS/oei/>

²Unified Medical Language System (UMLS) is a comprehensive collection of biomedical terms.

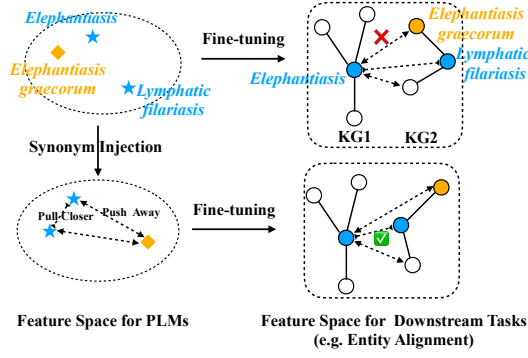


Figure 1: An example demonstrating the importance of synonym knowledge for entity alignment. Some normal PLM such as BERT tend to encode entities with common tokens to more similar spaces ignoring synonymic semantics, such as *Elephantiasis* for *Elephantiasis graecorum* and *Lymphatic filariasis* (6.62 vs 9.17 we measured in BERT feature space), which causes misalignment.

ambiguity and rigidly pull synonym pairs closer together in semantic space. Take the example in the upper right of Figure 2, it is unreasonable to always treat *Washington* as synonymous to either *George Washington* or *Washington, D.C.* since *Washington* has different meanings in different contexts. The pre-training objective of closing the distance from *Washington* to *George Washington* and *Washington, D.C.* cannot be satisfied simultaneously. Therefore, Context is imperative to disambiguate synonyms and should be considered in injecting synonym knowledge. Ignoring the synonym context will also significantly limit the generality of the synonym injected PLM, i.e., the PLM is hard to be applied to tasks out of the domain where the training synonyms are extracted.

Undermining semantic understanding of original PLMs. The *exact semantic similarity* expressed by synonyms and the *broad conceptual relevance* implied by MLM are often contradictory [19]. This is because PLMs that adopt MLM for pre-training (e.g., BERT) acquire semantic understanding capabilities based on word co-occurrence statistics. The neighboring words in the feature space of such PLMs are related words rather than synonyms, e.g., the top 10 nearest neighbors of *good* in BERT contain antonyms like *bad*. The existing methods directly inject synonym knowledge on top of the parameters of the PLMs that have established semantic understanding. This will inevitably result in semantic conflicts and weakens the PLM’s original semantic understanding capabilities. We refer to this phenomenon as **semantic forgetting**, analogous to the catastrophic forgetting [16, 18] of old samples in continual learning community. In another word, we argue that existing work utilizing synonyms for pre-training is task-specific pre-training at the expense of semantic understanding. SAPBERT, for example, is a further pre-training of PLM with *context-free* synonym knowledge for *context-free* entity linking, which sacrifices the ability to

understand semantics (as evidenced by the general degradation of performance for various downstream tasks in Section 4) and thus is not generalized.

To address these issues, we propose a **Pre-trained language model Injected with Contextual Synonyms knowledge (PICSO)**, whose input for pre-training is sentences with marked synonyms rather than synonyms without contexts. In order for PICSO to not only capture the semantics of the entire sentence, but also to focus on the semantics of the entity (synonym) in the context, we develop a new entity-aware Adapter structure with a novel masked self-attention mechanism. Equipped with Adapters of such a structure, PICSO supports continuous injection of synonym knowledge from multiple domains, while the contextual semantic understanding capability of the original PLM is not undermined. In the evaluation, we consider a general domain with 12.8 million synonym pairs extracted from Wikidata and a biomedical domain with 3.7 million synonym pairs extracted from UMLS. Extensive experiments on four similarity-oriented tasks have demonstrated that PICSO can dramatically outperform the original PLMs and the other knowledge and synonym injection methods including LIBERT and SAPBERT. In addition, experiments on GLUE have proven that PICSO also benefits general Natural Language Understanding (NLU) tasks.

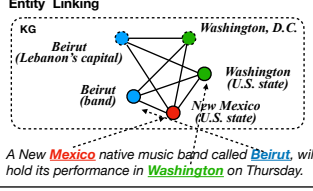
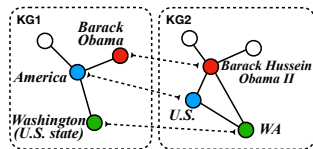
2. Related Work

2.1. Injecting Structure Knowledge into PLMs

Despite great success in many NLP tasks, some works [15, 35, 36] expose that PLMs such as BERT struggle to acquire rich knowledge during pre-training. Some efforts have been made to inject structure knowledge into PLMs, which can be divided into three main categories. The first category is **KG Injection**. With the emergence of plenty of general and domain-specific KGs, they have become one of the most important knowledge sources. A body of mainstream research [34, 41, 47] is devoted to inject KG triples (facts) in form of (*subject, relation, object*) into PLMs. ERNIE-THU [55] encodes the entities and relations in Wikidata by the KG embedding model TransE, then integrates entity representations based on the alignments between entity mentions and KG entities. K-Adapter [46] leaves the original parameters of the PLM unchanged and exports representations for structure knowledge of the KG. This is achieved via an additional compact neural model termed Adapter. Note that we also use Adapter, but our work differs from K-Adapter in two fundamental ways: (1) We focus on the gain of clean and efficient synonym knowledge for similarity-oriented tasks. (2) To inject contextual synonym knowledge more precisely, we specially designed entity-aware Adapter, which proved to be particularly effective in Section 4.4.1. The second category is **Rule Injection**. Rules exist as informal constraints or logical expressions, which can import sound explanatory [9] or precise reasoning capabilities [1] for PLMs. The last category is **Syntax-tree Injection**. Syntactic knowledge guides PLMs to understand

Table 1

Taxonomy of entity-level similarity-oriented tasks.

Source Entity	Target Entity	Typical Tasks	Examples
Unstructured Text	Unstructured Text	Entity Resolution Lexical Simplification	Entity Resolution Entity 1: sony playstation 2 dualshock 2 analog controller emerald . Entity 2: ps2 dualshock analog controller green . Same or Not Same?
Unstructured Text	Structured KG	Entity Linking Coreference Resolution	Entity Linking 
Structured KG	Structured KG	KG Canonicalization Entity Alignment Ontology Alignment	Entity Alignment 

the core constituents in sentences. Some studies [2, 56] have considered syntax-trees.

Structure knowledge may benefit similarity-oriented tasks to some extent but the synonym knowledge they contain are implicit and only take a small ratio. Meanwhile, existing works [26, 53] demonstrate the presence of redundant and irrelevant structure knowledge injected, which may instead lead to negative impact in solving downstream tasks. Compared to pure synonym knowledge, structure knowledge is inefficient and prone to introduce noisy knowledge.

2.2. Injecting Synonym Knowledge into PLMs

The injection of semantic constraints from synonyms into static word representations has been extensively studied before PLMs become popular. Numerous works [11, 30] demonstrated that synonyms can help models clearly distinguish between *exact semantic similarity* and *broader conceptual relatedness*. These works mainly fall into two categories: (1) **Joint Optimization Models** [31, 32] which introduce auxiliary objectives in pre-training to constrain the embeddings, and (2) **Post-optimization Models** [10, 44] which tune the pre-trained embeddings by adapting the pairwise distances to the semantic constraints of synonyms.

Following the popularity of PLMs, injecting synonym knowledge into PLMs has attracted wide attention. Some earlier works [7, 42] injected task/domain-specific synonym knowledge into PLMs during fine-tuning. For example, BERTMap [13] collects synonyms in ontologies to construct corpora which are used to fine-tune BERT for predicting class mappings between ontologies. Some previous works inject synonym knowledge into PLMs during pre-training. LIBERT [19] trains BERT from scratch with an auxiliary binary classification task that predicts whether entity pairs are synonymous or not. SAPBERT [25] further pre-trains BERT with massive synsets extracted from UMLS. A metric

learning objective is used to optimize the BERT, where synonymic and non-synonymic pairs are extracted as positive and negative samples, respectively. As we have stated before, these methods suffer from ambiguity of synonyms and the semantic forgetting problem. Besides, LIBERT has to combine the synonym corpus and the text corpus to pre-train a PLM from scratch, which brings a huge computational overhead, while our PISCO overcomes this weakness and supports flexible continual learning. SAPBERT is evaluated with biomedical entity linking solely with the synsets from UMLS, while PISCO is evaluated by four different tasks with synonym knowledge of both general domain and the medical domain.

2.3. Similarity-oriented Tasks

Similarity-oriented tasks refer to tasks whose core challenge is to capture entity-level or sentence-level semantic similarity, which covers an extensive range of natural language processing tasks. Modeling entity-level and sentence-level semantic similarity play a significant role for almost all AI applications, such as machine translation[48], dialogue systems [29], and recommendation systems [39]. As shown in Table 1, entity-level similarity-oriented tasks can be roughly categorized according to the source of two entities (source and target entity) in the entity pairs: (1) Both source and target entities are derived from unstructured text. Typical tasks include entity resolution [33], lexical simplification [37]. For example, the task set for entity resolution is to determine whether both entities are identical given two entities and their contextual descriptions (e.g. text, table). (2) Source and target entities are derived from unstructured text and structured KGs, respectively. Representative tasks include entity linking [54], coreference resolution [21], etc. For instance, the goal of entity linking is to link entities in sentences to the corresponding entity entries in KG. (3) Both

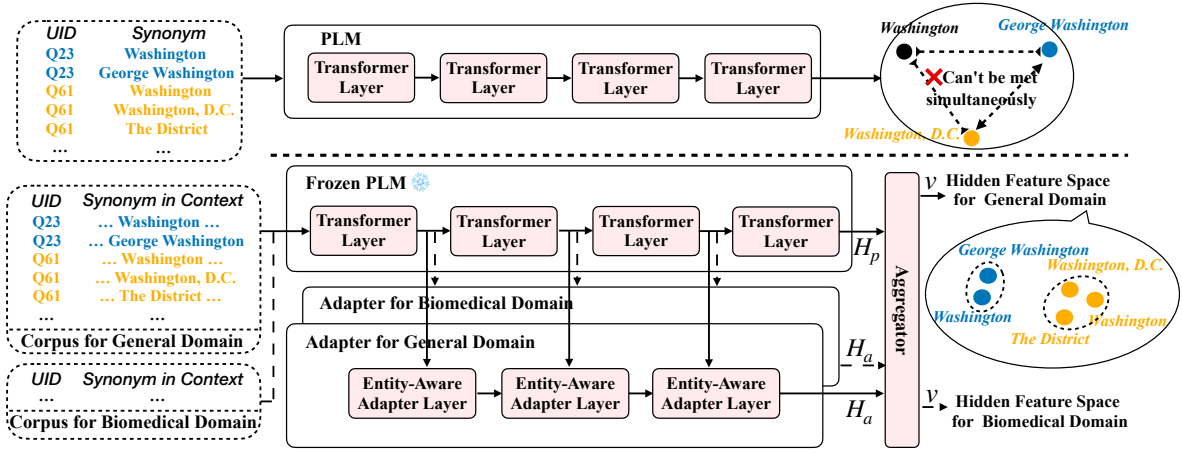


Figure 2: Frameworks of injecting synonym knowledge into PLMs. Top: the previous methods LIBERT and SAPBERT. Bottom: our method PICSO.

source and target entities come from structured knowledge graphs, and representative tasks consist of KG canonicalization [8], entity alignment [42], and ontology alignment [13]. Entity alignment associates entities in different knowledge graphs if they are semantically same. The core challenge of all the above tasks is to model semantic similarity based on entities as well as their contexts. Sentence-level similarity-oriented tasks are likewise an essential branch of general NLU tasks, including paraphrase identification [51], semantic textual similarity [5], and numerous others. For example, semantic textual similarity measures the meaning similarity of sentences, which is also included in the General Language Understanding Evaluation (GLUE) benchmark [45].

3. Methodology

3.1. Overall Framework

As shown in the top of Figure 2, the framework of the previous methods LIBERT and SAPBERT take the context-missing synonyms as input and directly update the parameters of the PLM in pre-training. Such a framework cannot handle ambiguous or context-sensitive synonyms, and will inevitably disrupt the original semantic understanding capabilities of the PLM learned from the text corpus. To address these issues, as shown in the bottom of Figure 2, our method PICSO uses sentences with marked synonyms rather than individual synonyms as input. In pre-training, the internal parameters of the PLM are frozen, while some entity-aware Adapters are attached with each one trained by synonym knowledge of one domain. We will next introduce some key data annotations and the framework modules.

We assume a collection of synsets $\{S_1, \dots, S_m\}$ are extracted from synonym knowledge sources. Each synset is a collection of synonyms (i.e., words or phrases that have the same meaning), denoted as $S_i = \{\text{UID}_i, e_i^1, \dots, e_i^n\}$, where UID_i denotes the unique identifier³ of the synset S_i . The

³UID is determined by the synonym knowledge source. For example, in Wikidata, a UID is by the letter Q and a number (e.g., Q61); in UMLS, UID is by the letter C and a number.

pre-training corpus C is composed of instances, and each instance is denoted as $x = \{\text{UID}, w, p_s, p_e\}$, where w is a sequence of tokens with a marked synonym. We define two special markers $\langle e \rangle$ and $\langle /e \rangle$ to locate the synonym, and use p_s and p_e to represent the indexes of $\langle e \rangle$ and $\langle /e \rangle$ in the sequence, respectively. An instance example is as follows: $x = \{\text{Q61}, [\dots, \langle e \rangle, \text{Washington}, \text{DC}, \langle /e \rangle, \dots], 26, 29\}$.

PICSO mainly includes three modules. The first module is a **Frozen PLM**. We select BERT_{base} as the backbone. It outputs hidden features denoted as $H_p \in \mathbb{R}^{l \times d}$, in which l and d represent the length of the input word sequence and the dimension of the last hidden features of BERT, respectively. The second module is **Entity-aware Adapter**. The input is the hidden features output by Transformer layers of the BERT. Each Adapter is plugged into the PLM as a separate module and pre-trained independently for learning synonym knowledge of a specific domain. It learns the semantics of the entities with their contexts and the entire sentence semantics via a combination of two masked self-attention mechanisms, and eventually outputs features $H_a \in \mathbb{R}^{l \times d}$. The third module is an **Aggregator**. It fuses H_p and H_a to obtain the final feature v where two different strategies are proposed. The modules are pre-trained with a contrastive learning objective, using the pre-training corpus. It is worth mentioning that the Adapters can be continuously learned with one domain by another, and can be either used together or independently in Aggregator for a downstream task.

3.2. Entity-Aware Adapter

Our Adapters are expect to capture the entity-centric semantics from multiple different domains. Each Adapter contains K layers, while each layer is a stack of one down projection layer, N Transformer layers and one up projection layer, as shown in Figure 3. The output of the intermediate Transformer layer of the frozen PLM and the output of the previous Adapter layer are summed up as the input of the current Adapter layer, where a residual connection is applied between the input and the output. To enable the Adapter to perceive the semantics of entities with their contexts, a

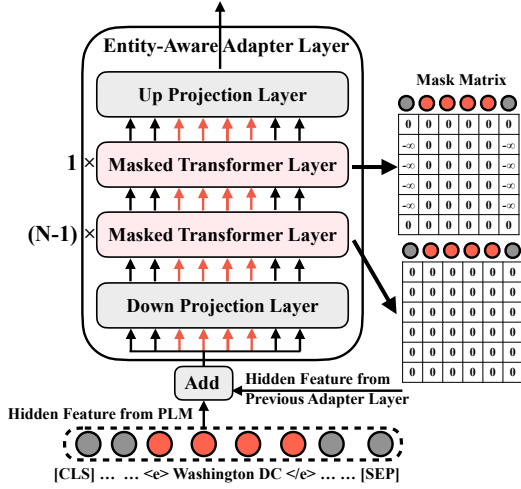


Figure 3: Structure of one layer of our entity-aware Adapter.

new extension of the masked self-attention mechanism is proposed:

$$\begin{aligned} q_i^{l+1}, k_i^{l+1}, v_i^{l+1} &= h_i^l W_q, h_i^l W_k, h_i^l W_v \\ h_i^{l+1} &= \text{Attention}(q_i^{l+1}, K^{l+1}, V^{l+1}) \\ &= \sum_{j=1}^I \frac{1}{Z} \exp \left(\frac{\langle q_i^{l+1}, k_j^{l+1} \rangle + M_{ij}^{l+1}}{\sqrt{d_k}} \right) v_j^{l+1} \end{aligned} \quad (1)$$

where $h_i^l \in \mathbb{R}^d$ represents the hidden feature corresponding to the i -th token of the l -th Transformer layer. W_q, W_k, W_v are trainable parameters. Z and $\sqrt{d_k}$ refer to the normalization factor and scale factor, respectively. M^l is the mask matrix of the l -th Transformer layer. Note that when M_{ij}^l tends to $-\infty$, the value of $\exp(*)$ tends to 0, that is, the token w_i is not concerned with the semantics of the token w_j . When $M_{ij}^l = 0$, the computation degenerates to a regular self-attention mechanism. Formally, the mask matrix M is defined as:

$$M_{ij}^l = \begin{cases} -\infty & l = N \wedge p_s \leq i \leq p_e \wedge j < p_s \\ -\infty & l = N \wedge p_s \leq i \leq p_e \wedge j > p_e \\ 0 & \text{Otherwise} \end{cases} \quad (2)$$

This means that for the first $N-1$ Transformer layers, we employ the conventional self-attention mechanism such that the hidden features model the semantic features of the whole sentence. For the last Transformer layer, the new masked self-attention mechanism is adopted to make each token of entity between $\langle e \rangle$ and $\langle /e \rangle$ focus on its own entity sense, achieving a trade-off between sentence sense and entity sense.

3.3. Aggregator

The output features of the frozen PLM and the Adapter are fed to the aggregator to obtain the final features v . During pre-training, the aggregator concatenates H_p and H_a and

takes out the vectors at indexes p_s and p_e to be concatenated again: $H = H_p \oplus H_a, h = H[p_s] \oplus H[p_e]$. Then, the resulting intermediate feature $h \in \mathbb{R}^{4d}$ is passed into a fully-connected layer followed by a normalization layer. When applied in downstream tasks, if fine-tuning is performed, we send the concatenated features H into the task-specific layer. The Adapter can be used individually. If multiple Adapters are used, H_p and multiple H_a can be concatenated together. If PICSO is directly used as a feature extractor without fine-tuning, we sum H_p and the l_2 -normalized H_a as the semantic features for the downstream task.

3.4. Pre-training Objective

We apply a contrastive objective which pulls synonymous pairs closer and non-synonymous pairs away to train the Adapters and the aggregator. To do this we first generate positive and negative instance pairs from each batch of the pre-training corpus. For an arbitrary instance x_i in the batch, it is combined with each of the instances that have the same UID as x_i (i.e., $\text{pos}(x_i) = \{x_j \mid x_i[\text{UID}] = x_j[\text{UID}]\}$) for positive instance pairs, and combined with the other instances in the batch for negative instance pairs. Inspired by [23, 24, 40], we design the contrastive objective, which concerns more on the hard negative instance pairs, i.e., non-synonymous pairs that are difficult to distinguish. Concretely, the contrast loss is calculated as follows:

$$\begin{aligned} \mathcal{L}_{cl} &= - \sum_{i=1}^B \log \frac{S_i^+}{S_i^+ + S_i^-}, \\ S_i^+ &= \sum_{j=1}^{|\text{pos}(x_i)|} e^{v_i^\top \cdot v_j / t}, \\ S_i^- &= \max \left(\frac{-(B-1-|\text{pos}(x_i)|) \cdot \tau^+ \cdot S_i^+ + \widetilde{S}_i^-}{1 - \tau^+}, e^{\frac{-1}{t}} \right), \\ \widetilde{S}_i^- &= \frac{(B-1-|\text{pos}(x_i)|) \sum_{k:k \neq \text{pos}(x_i)} e^{(1+\beta)v_i^\top \cdot v_k / t}}{\sum_{k:k \neq \text{pos}(x_i)} e^{\beta v_i^\top \cdot v_k / t}}. \end{aligned} \quad (3)$$

where S_i^+ (resp. S_i^-) reflects the similarity between training pairs from the positive (resp. negative) pairs, B is the size of a batch, τ^+ is the class-prior probability that can be estimated from data or treated as a hyper-parameter, β is the hyper-parameter controlling the level of concentration on negative samples, t is the temperature scaling factor which we set as 0.5 in all our experiments. The \widetilde{S}_i^- term awards higher weights to negative instance pairs whose instances have high similarity (i.e., hard negative samples) by reweighting. We assign a greater penalty to these hard negative samples instead of mining hard negative samples by modifying the sampling strategy as in LIBERT and SAPBERT. Hence, our proposed contrastive objective is easier to use and achieves better results, as also demonstrated in Section 4.4.2.

3.5. Construct Pre-training Corpus

Table 2
Statistics for pre-training corpus

Domain	# UID	# Synonym Pairs	# Synonym Pairs per UID	Average Sentence Len	Average Edit Distance
General	0.65M	3.75M	5.8	117.9	18.4
Biomedical	2.10M	12.79M	6.1	107.3	27.7

One entity-aware Adapter learns independently from the corpus of one domain. We consider a general domain with the corpus extracted from Wikidata and a biomedical domain with the corpus extracted from UMLS. The concrete corpus construction procedure has the following two steps: **Gathering sentences and synonyms.** We get the synsets from the two knowledge bases, Wikidata and UMLS, which already have a massive collection of high-quality synonyms (i.e., entities with identical meanings). To construct the corpus, we further collect sentences that contain entity mentions linked to knowledge base entities. For the general domain, we crawl Wikipedia articles that contain abundant entity mentions with human-annotated hyperlinks to Wikidata entities. For the biomedical domain, we use the off-the-shelf high-precision entity linking tool Medlinker [28] to link entity mentions of article abstracts from PubMed⁴ to UMLS. In addition, we include the definitions of entities from UMLS as a supplementary corpus. We also remove simple pairs with edit distances less than 10 and limit the generation of up to 50 synonym pairs per UID.

Balancing low-frequency entities. To ensure the proportion of low-frequency synonyms in the corpus, we replaced entities in some sentences with their low-frequency synonyms. For example, *California* in a sentence would be replaced by *The Golden State*.

In the end, we constructed over 3.7 million and over 12.8 million context-equipped synonym pairs for the general domain and the biomedical domain, respectively. The statistics of the pre-training corpus for each domain are shown in Table 2.

4. Experiments

4.1. Experiment Setup

4.1.1. Downstream Tasks.

We chose four extensively studied similarity-oriented downstream tasks, i.e., entity resolution, entity linking, KG canonicalization, and lexical simplification, covering the assessment of tasks consuming both structured data and unstructured text. For entity resolution and entity linking, PICSO is further fine-tuned by their samples; while for the other two tasks, PICSO is not fine-tuned, which evaluates PICSO in a zero-shot or unsupervised setting. To verify that PICSO also benefits general NLU tasks, we also conducted experiments on GLUE which has 9 NLU tasks. Each task will be separately introduced in detail below.

⁴<https://pubmed.ncbi.nlm.nih.gov/>

4.1.2. Baselines.

We comprehensively compare PICSO with three types of baselines. **Basic PLMs:** (1) **BERT** [17] is an important baseline since PICSO is based on it; (2) **Roberta** [27] is the the advanced version of BERT, which removes the next sentence prediction task and employs a larger pre-trained corpus. **PLMs with KG knowledge injected:** (3) **ERNIE-THU** [55] is the most classic KG knowledge-enhanced PLM model, which incorporates entity representations learned through TransE into BERT; (4) **K-Adapter** [46] injects KG structured knowledge by an Adapter via relational classification. Note that our entity-aware Adapter is an enhancement of the original Adapter. The original K-Adapter is based on Roberta_{large}. For a fair comparison, we re-implemented K-Adapter with Roberta_{base}. **PLMs with synonym knowledge injected:** (5) **LIBERT** [19] trains BERT from scratch by predicting synonymous entity pairs; (6) **SAPBERT** [25] continues the pre-training of a BERT with massive synonymous and non-synonymous entity pairs from UMLS. Since LIBERT and SAPBERT are the most relevant methods to this study, they will get more attention in the following result analysis. Note that PICSO is compatible for the methods designed for specific tasks, so we don't compare task-specific SOTA methods, as in much of the PLM works.

4.1.3. PICSO Setup

We use BERT_{base} which contains 12 Transformer layers and 433M parameters. The dimension of the hidden feature corresponding to each token is 768. Each entity-aware Adapter contains 3 Adapter layers plugged at layers 0, 5, 11 of BERT. Two Transformer layers identical to those in BERT are set in each Adapter layer, i.e., $N=2$. We tally the parameters for each entity-aware Adapter to be approximately 46M. Compared with LIBERT and SAPBERT, which require pre-training the entire BERT, we have fewer parameters to tune. We pre-train the PICSO for 3 epochs on 8 Tesla V100s with a batch size of 256. The time to train the general domain Adapter and the medical domain Adapter is about 0.46 h and 1.58 h per epoch, respectively. The full model using two Adapters is abbreviated as PICSO(W+U). The model using one Adapter trained on Wikidata (resp. UMLS) is denoted as PICSO(W) (resp. PICSO(U)). PICSO(w/o k) represents BERT with a randomly initialized Adapter, i.e., an Adapter without synonym knowledge injection.

4.2. Experiments on Similarity-oriented Tasks

4.2.1. Entity Linking

Datasets and Fine-tuning. Entity linking aims to match an entity mention in a textual context with an entity in the target KG. SAPBERT is also tested on the entity linking task, but the datasets used such as BC5CDR, lack the entity mentions' contexts and are less ambiguous. Hence, we adopt a more challenging and widely used dataset named AIDA CoNLL-YAGO [14], and conduct cross-domain experiments following the same setting as in [20, 50, 52]. Namely, models are trained on a training subset of AIDA (AIDA-train) and evaluated on a test subset of AIDA as well as five popular public datasets: AQUAINT, MSNBC, ACE2004, CWEB

Table 3
Results (%) on entity linking

Model	AIDA		ACE2004		AQUAINT		CWEB		MSNBC		WIKI	
	Acc@1	Acc@5	Acc@1	Acc@5	Acc@1	Acc@5	Acc@1	Acc@5	Acc@1	Acc@5	Acc@1	Acc@5
BERT	72.35	86.64	70.81	82.87	60.24	80.74	47.84	68.59	56.40	78.04	58.68	79.15
Roberta	74.89	87.32	67.91	79.64	58.28	78.93	45.22	65.54	55.25	75.54	55.27	76.13
ERNIE-THU	69.12	86.48	69.26	80.93	51.71	73.03	44.49	65.39	56.25	77.28	55.97	76.98
K-Adapter	73.72	85.72	69.84	80.41	60.27	79.86	45.49	65.37	53.11	72.93	53.56	75.70
LIBERT	69.69	83.75	66.92	78.21	63.68	80.05	48.51	66.71	53.35	74.84	57.70	76.78
SAPBERT	54.21	73.64	49.02	70.03	52.26	77.02	37.97	60.99	41.31	66.46	47.15	66.68
PICSO(W+U)	78.26	88.63	75.87	84.26	69.32	82.46	52.25	70.17	62.50	81.47	61.04	80.69
PICSO(W)	76.79	88.25	73.64	83.87	66.71	81.70	49.49	69.52	60.04	80.62	60.33	80.28
PICSO(U)	75.91	88.05	72.84	83.65	63.54	81.15	48.62	69.39	59.06	79.47	58.93	79.34
PICSO(w/o k)	73.10	87.10	71.98	83.20	63.13	80.90	48.35	68.21	52.28	72.56	57.15	76.60

and WIKI, which cover a wide range of domains such as medicine and technology & science. To fine-tune the PLMs for entity linking, as with SAPBERT, we use the multi-similarity loss, a metric learning objective that adjusts the pairwise distances of positive and negative pairs. Each positive pair consists of an entity mention and its corresponding entity in KG, and negative pairs are generated by randomly corrupting positive pairs. Acc@k is employed to evaluate the a model with the ranking of KG entities, which means the percentage of the top-k predictions that contain the ground truth KG entity. To ensure fairness, for all downstream tasks, we set their key hyperparameters (e.g., learning rate, number of training rounds), the same, and no complex tuning of hyperparameters is performed.

Results. The results for entity linking are shown in Table 3, through which we can have the following observations. (1) PICSO(W+U) achieves the best results on all six datasets, and the absolute improvement of Acc@1 on some datasets even exceeds 5%. Except for LIBERT on AQUAINT, LIBERT and SAPBERT show general performance decreasements, which confirms that their pre-training negatively impacts the original semantic understanding capabilities of PLMs. (2) Injecting KG knowledge does not necessarily benefits entity linking. K-Adapter outperforms its base PLM Roberta in some cross-domain cases, while ERNIE-THU generally shows a light degradation compared to its base PLM BERT. (3) PICSO (W) has a higher gain on performance than PICSO (U). Although these testing datasets contain medical data, medical data only take a small ratio, and the general synonym knowledge from Wikidata contributes more. (4) PICSO (w/o k) has an overall gain compared with the base PLM BERT. This is due to the fine-tuning and a larger model of PICSO.

4.2.2. Entity Resolution

Datasets and Fine-tuning. Entity resolution (a.k.a entity matching) is to find records that refer to the same real-world entity across different data sources. Following [33],

we compare the methods using three datasets: WDC LSPC, DBLP-Scholar and Company. The WDC LSPC dataset is built by product offers from e-shop, containing four categories of products: computer, camera, shoe and watch. For each product category, there are one test set and four training sets with different sizes. In other words WDC LSPC has 4×4 sub-datasets. We pick two training sets of medium (M) and large (L) sizes for each category in our experiments. For WDC LSPC, the entities in the test set all appear in the training set, while for Company and DBLP-Scholar, there is no overlap between the entities of the test set and those of the training sets. Thus, WDC LSP enables the evaluation with *seen* entities, while the other two enable the evaluation with *unseen* entities. Following [33], a linear layer and a sigmoid function are attached after each pre-trained model (on top of the hidden feature of the [CLS] token) to predict whether two entities match or not, and a binary cross-entropy loss is used to fine-tune the model. Precision (P), Recall (R) and F1 Score are adopted as the metrics.

Results. The results for entity resolution are shown in Table 4, from which we can have similar observations as in entity linking. We note that in the seen case, i.e., in the WDC LSPC, PICSO is insensitive to the size of the training set compared to other PLMs, suggesting that the synonyms contain partial knowledge that entity resolution desires. The testing set sizes of Company and DBLP-Scholar are 20 and 5 times larger than that of WDC LSPC, respectively. Thus the improvement by PICSO in the unseen case is actually more significant compared to the seen case, demonstrating the sound learning potential of PICSO. Compared to PICSO, LIBERT and SAPBERT have competitive recall but much worse precision. This indicates that LIBERT and SAPBERT confusingly treat some non-synonymous pairs as synonymous, which could be caused by the lack of contextual constraints in pre-training.

Table 4
Results (%) on entity resolution

Model	Computers-M			Computers-L			Shoes-M			Shoes-L			Watches-M			Watches-L			Cameras-M			Cameras-L			Company			DBLP-Scholar		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
BERT	-	-	89.31	-	-	92.11	-	-	79.82	-	-	87.37	-	-	89	-	-	95.23	-	-	87.02	-	-	91.02	-	-	91.7	-	-	95.27
RoBERTa	-	-	91.9	-	-	94.68	-	-	81.12	-	-	86.6	-	-	92.28	-	-	93.93	-	-	90.2	-	-	93.91	-	-	91.81	-	-	95.29
ERNIE-THU	80.00	93.33	86.15	90.9	93.33	92.1	78.41	82.6	80.45	90.23	87.33	88.75	77.59	92.33	84.32	93.53	91.66	92.59	76.98	90.33	83.12	87.61	92.00	89.75	89.29	93.81	91.5	95.14	95.32	95.23
K-Adapter	91.55	91.55	91.55	94.95	94.33	94.63	87.45	86.28	86.86	95.10	90.96	92.99	90.28	96.00	93.05	97.50	91.33	94.32	92.00	92.00	92.00	93.85	91.66	92.74	93.27	92.09	92.67	94.69	96.82	95.74
LIBERT	79.41	90.00	84.37	86.29	92.33	89.21	77.23	83.94	80.44	86.97	82.60	84.73	84.37	90.00	87.09	94.82	91.66	93.22	79.68	83.66	81.62	87.91	87.33	87.62	76.9	92.76	84.09	93.21	94.95	94.07
SAPBERT	53.61	81.66	64.72	52.60	87.66	65.75	76.87	82.27	79.48	87.01	82.94	84.93	87.37	90.00	88.66	94.36	89.33	91.78	87.26	91.33	89.25	86.16	91.33	88.67	90.74	89.89	90.31	94.10	96.91	95.48
PICSO(W+U)	93.62	93.00	93.31	94.50	95.66	95.07	91.17	92.62	92.62	94.88	92.97	93.91	93.77	94.00	93.88	98.57	95.00	96.75	94.68	93.42	94.04	95.84	92.33	94.05	92.94	92.91	92.92	94.26	97.87	96.03
PICSO(W)	90.70	94.33	92.48	94.96	94.33	94.64	91.97	91.97	91.97	95.12	91.3	93.17	93.64	93.33	93.48	97.93	94.66	96.27	93.19	91.33	92.25	94.33	94.33	94.33	92.63	92.3	92.46	94.32	97.56	95.91
PICSO(U)	92.35	92.66	92.51	93.66	93.66	93.66	91.08	92.3	91.69	94.03	89.63	91.78	92.66	92.66	92.66	98.58	93.00	95.71	93.11	94.66	93.88	94.86	92.33	93.58	92.87	92.42	92.65	94.09	97.34	95.69
PICSO(w/o k)	91.14	92.66	91.90	91.65	94.03	92.82	84.51	87.62	86.04	93.07	89.30	91.00	91.64	92.33	91.98	96.91	96.91	95.60	91.88	89.66	90.75	94.16	90.00	92.03	91.02	93.61	92.29	94.49	95.74	95.11

4.2.3. Lexical Simplification

Datasets and Task Method. Lexical simplification seeks to replace target words in contextual sentences with simpler substitutes without altering the meanings. It is an important evaluation task in LIBERT. Three public datasets — LexMTurk, BenchLS, and NNSeval are used for evaluation where LexMTurk is collected from Wikipedia, and BenchLS and NNSeval are expanded versions of LexMTurk. As the evaluation of LIBERT, an unsupervised pre-trained PLM-based method [37], which first generates a set of candidate substitutes and then ranks these substitutes, is adopted. We report scores of Precision, Recall and F1 score for candidate generation, as well as the scores of Accuracy for the final result, so as to compare different pre-trained PLMs that are adopted.

Results. The results of lexical simplification are presented in Table 5. We can observe that PICSO obtains the highest final accuracy on all three datasets. Surprisingly, SAPBERT scores almost 0 on all three datasets. On the one hand, SAPBERT severely loses its semantic understanding capability of textual context, and on the other hand, the unsupervised setting prevents it from reconstructing the semantic space by fine-tuning. Regarding LIBERT, it has a lead in precision for substitute generation on BenchLS and LexMTurk. However, this is achieved at the expense of recall, which is undesirable in the first stage of substitute generation. Meanwhile, although LIBERT authors claim that it outperforms their own implementation of BERT with a smaller pre-training corpus, it has no advantage over the original BERT implemented by Google.

4.2.4. KG Canonicalization

Datasets and Task Method. KGs constructed from unstructured data usually store redundant entities. KG canonicalization aims to identifying the equivalent entities in a KG. It is an inherently unsupervised task since we usually are not given any annotated data. As in [8], we used four datasets — Rever-base, Reverb45k, Reverb-ambiguous, and CANON-ICNELL for evaluation. The first three are homogeneous, while CANONICNELL is heterogeneous, constructed based on NELL [4]. A simple PLM-based method [8, 43] is often adopted for comparing the performance of PLMs. Briefly it first uses a PLM to build text embeddings for entities and

then uses a Hierarchical Agglomerative Clustering (HAC) algorithm for clustering. We use the macro and micro F1 scores as evaluation metrics.

Results. The results for KG canonicalization are shown in Table 6. Since KG canonicalization is an intra-KG task and excludes textual context, models injected with both synonym knowledge and KG knowledge obtain performance boosts versus their base PLMs. PICSO achieves the best result, followed by SAPBERT, which strongly confirms the superiority of synonym knowledge over generic KG knowledge for this similarity-oriented task. Meanwhile, according to our statistics, medical-related entities in the Reverb* datasets occupy 20% to 30%, which is consistent with that PICSO(U) has higher performance than PICSO(W). As expected, PICSO(w/o k) shows a serious performance decline due to feature space corruption.

4.3. Experiments on General NLU Tasks

The General Language Understanding Evaluation (GLUE) benchmark [45] covers diverse NLU tasks, which is the main benchmark used in PLMs. To explore whether synonym knowledge deteriorates performance on general NLU tasks, we evaluate PICSO on eight datasets of GLUE, and the results are shown in Table 7.

In summary, PICSO achieves competitive results on GLUE, and some interesting phenomena can be observed. (1) PICSO has the highest average score on all the 8 tasks, which proves that the synonym knowledge is beneficial. SAPBERT, on the other hand, struggles to handle the context-involved tasks, achieving the worst result. (2) These GLUE tasks mainly have two categories. The first category involves similarity prediction, which includes MRPC, STS-B and QQP. They require the model to infer whether two sentences have paraphrase/semantic equivalence. PICSO and LIBERT achieved the best results as synonym knowledge is particularly valuable. (3) The second category is natural language inference tasks including MNLI, QNLI and RTE. The goal of these tasks is to determine whether two sentences have implicative relations (i.e., whether the hypothesis can be inferred from the premises), which requires factual knowledge. PLMs with KG knowledge injected, i.e., K-Adapter, achieve better performance.

Table 5

Results (%) on lexical simplification

Model	BenchLS				LexMTurk				NNSeval			
	P	R	F1	Acc	P	R	F1	Acc	P	R	F1	Acc
BERT	24.81	32.08	27.98	57.03	32.04	23.43	27.06	71.24	18.32	23.78	20.69	37.65
Roberta	20.08	27.25	23.12	52.41	25.66	19.95	22.45	64.00	18.87	25.18	21.57	41.84
ERNIE-THU	24.19	32.83	27.86	55.97	31.60	24.57	27.64	70.00	18.66	24.90	21.33	33.05
K-Adapter	19.93	27.05	22.95	55.40	25.28	19.66	22.11	69.32	16.23	21.66	18.56	41.09
LIBERT	27.66	22.52	24.83	47.57	37.00	17.27	23.54	63.2	15.39	20.54	17.60	27.19
SAPBERT	0.05	0.07	0.06	0.10	0.06	0.04	0.05	0.2	0.04	0.05	0.05	0
PICSO(W+U)	25.18	34.18	29.00	59.63	33.08	25.72	28.94	73.20	20.15	26.38	22.84	42.76
PICSO(W)	25.15	34.13	28.96	58.66	32.58	25.33	28.50	73.59	19.03	25.40	21.76	42.65
PICSO(U)	25.10	34.06	28.90	57.91	32.60	25.35	28.52	73.0	18.99	25.34	21.71	42.10
PICSO(w/o k)	20.97	28.46	24.15	47.68	28.56	22.21	24.98	63.60	15.48	20.65	17.69	29.70

Table 6

Results (%) on KG canonicalization

Model	Reverb-base		Reverb45k		Reverb-ambiguous		CANONICNELL	
	Macro	Micro	Macro	Micro	Macro	Micro	Macro	Micro
BERT	69.83	92.35	16.92	75.35	60.10	81.62	63.19	66.93
Roberta	75.75	92.71	26.48	76.83	61.55	87.97	70.63	76.01
ERNIE-THU	72.49	92.57	19.54	76.09	61.78	87.95	65.50	69.56
K-Adapter	75.17	91.28	28.10	76.58	59.13	80.83	71.48	77.62
LIBERT	66.93	91.97	16.18	75.58	61.22	86.12	62.09	68.37
SAPBERT	82.41	93.10	41.18	79.59	61.92	87.98	74.57	80.54
PICSO(W+U)	89.21	93.56	44.69	80.67	62.67	88.11	76.39	81.98
PICSO(W)	85.31	93.21	42.62	79.97	62.51	88.10	75.81	81.69
PICSO(U)	87.14	93.26	43.97	79.79	62.41	88.09	76.22	81.85
PICSO(w/o k)	63.10	90.84	15.70	70.74	57.54	72.44	60.43	64.83

Table 7

Results on eight GLUE tasks.

Model	CoLA	SST-2	MRPC	STS-B	QQP	MNLI-(m/mm)	QNLI	RTE	Avg
BERT	56.53	92.32	88.85	88.48	87.49	83.81/84.1	90.66	65.7	81.99
Roberta	50.19	94.15	81.83	84.88	87.48	87.36/87.34	92.17	56.32	80.19
ERNIE	44.28	90.6	82.15	85.03	86.75	83.10/83.51	89.99	58.48	78.21
K-Adapter	54.70	93.69	85.62	87.62	86.72	87.29/87.02	92.7	68.95	82.70
SAPBERT	4.38	88.53	81.21	82.57	85.86	81.81/82.54	89.38	54.87	72.35
LIBERT	37.2	89.3	88.7	-	90.0	79.6/80.0	87.7	66.4	77.36
PICSO(W+U)	58.04	93.89	89.41	89.43	88.58	84.66/84.92	91.63	64.62	82.80
PICSO(W)	57.78	92.89	89.83	89.12	88.07	84.61/84.95	91.12	64.98	82.59
PICSO(U)	58.29	92.32	89.61	89.09	87.82	84.63/85.31	91.27	63.90	82.47
PICSO(w/o k)	57.78	93.00	88.93	88.98	86.92	81.85/82.27	90.74	63.18	81.51

Table 8

Ablation study on entity-aware Adapter. w/o EA means without entity-aware mechanism.

Model	Entity Linking		Lexical Simplification					GLUE	
	AIDA		BenchLS					STS-B	QQP
	Acc@1	Acc@5	P	R	F1	Acc	Pearson	F1	
BERT	72.35	86.64	24.81	32.08	27.98	57.03	88.48	87.49	
PICSO	78.26	88.63	25.18	34.18	29.00	59.63	89.43	88.58	
w/o EA	73.23	87.08	24.80	32.89	28.27	57.68	88.96	87.54	

4.4. Additional Experiments

4.4.1. Ablation Study on Entity-Aware Adapter

In the ablation experiments, for the similarity-oriented tasks, we selected entity linking and lexical simplification as representatives of the fine-tuned and unsupervised methods, respectively. For the NLU tasks, STS-B and QQP are picked.

We conducted ablation experiment for the entity-aware mechanism to demonstrate its effectiveness and necessity for injecting contextual synonym knowledge. The results in Table 8 show that although PICSO w/o EA does not show the same negative gain as the SAPBERT and LIBERT, the magnitude of the gain is much smaller compared to PICSO, especially for the unsupervised lexical simplification. We argue that with entity-aware Adapter, the model can focus more on the semantics represented by the entities and thus more accurately draw the corresponding synonyms closer without introducing noise.

4.4.2. Comparison of Pre-training Loss Functions

As illustrated in Table 9, we compare the effects of three pre-training objectives on the pre-training effect. Triplet Margin Loss [3] was designed for computer vision tasks such as image classification and can be formulated as $L = [d_{ap} - d_{an} + m]_+$, where d_{ap} and d_{an} denote the distance from the anchor to the positive and negative samples, respectively. InfoNCE Loss is a classic contrastive loss and has been widely used in self-supervision papers [6, 12]. It can be equated as $L = -\log \frac{\exp(q \cdot k_+ / \tau)}{\sum_{i=0}^K \exp(q \cdot k_i / \tau)}$, in which q , k_+ , k_i represent anchor, positive and negative samples respectively. Note that online hard negatives mining is available for both pre-training objectives. In contrast, our designed pre-training objective does not require cumbersome online mining, but only re-weights the penalty for negative samples based on similarity, thus allowing the model to focus more on non-synonymous entity pairs that are elusive to distinguish. Our proposed pre-training objective is more efficient and effective.

4.4.3. Comparison of PLM backbones

In the main experiments we adopt BERT as the backbone of PICSO. Roberta differs from BERT in that it discards next sentence prediction (NSP) as the pre-training task. In Table 10, we compare the performances of PICSO with BERT and Roberta as the backbone, and their average gains on

Table 9

Comparison of different pre-training loss functions.

Loss	Entity Linking		Lexical Simplification					GLUE	
	AIDA		BenchLS					STS-B	QQP
	Acc@1	Acc@5	P	R	F1	Acc	Pearson	F1	
Triplet Margin Loss	77.23	87.12	23.69	32.18	27.29	54.76	87.28	85.73	
InfoNCE	77.32	87.32	23.46	32.10	27.11	54.27	88.05	85.71	
Ours	78.26	88.63	25.18	34.18	29.00	59.63	89.43	88.58	

Table 10

Comparison of different PLM backbones.

Model	Entity Linking		Lexical Simplification					GLUE	
	AIDA		BenchLS					STS-B	QQP
	Acc@1	Acc@5	P	R	F1	Acc	Pearson	F1	
BERT	72.35	86.64	24.81	32.08	27.98	57.03	88.48	87.49	
Roberta	74.89	87.32	20.08	27.25	23.12	52.41	84.88	87.48	
PICSO	78.26	88.63	25.18	34.18	29.00	59.63	89.43	88.58	
Roberta-based	78.34	89.10	21.38	28.91	24.58	54.84	87.13	88.89	

downstream tasks are 2.24 and 2.14, respectively. Benefiting from the favorable performance of Roberta on the entity linking task, Roberta-based PICSO outperforms BERT-based PICSO on the entity linking task. We can conclude that the enhancement of contextual synonym knowledge for downstream tasks is model-agnostic and universal.

4.4.4. Case Study

Case study is conducted on four similarity-oriented tasks. From Table 11, we can observe that: (1) The example on the entity linking task illustrates that BERT struggles to resolve the ambiguity caused by multiple meanings of a word, while benefiting from the introduction of contextual information about synonyms, PICSO can clearly distinguish non-synonymous entity pairs with the same surface name. This is more obviously illustrated by the example on the lexical simplification task, where tender has the meaning of gentle in some contexts, and painful when describing a body part. Compared to BERT, PICSO can distinguish more explicitly between the two semantics. (2) The case on entity resolution and KG canonicalization demonstrates that BERT lacks the ability to capture synonym information. For example, *Sky Caption Blue* and *Blue* are synonyms in some sense. Whereas on the KG canonicalization task, BERT fails to infer the exact synonym pair and tended to consider *Virginia wesleyan college* and *Rider college* as synonyms, which contain the common token *college*, or consider *Palm casino* and *Flamingo hotel* as synonyms, which are both buildings in Las Vegas.

5. Conclusion and Future Work

The paper presents PICSO that can inject contextual synonym knowledge from multiple domains into the PLM without disrupting its original semantic understanding capabilities. PICSO are equipped with entity-aware Adapters, each of which constrains the visible range of the tokens of

Table 11

Case study for similarity-oriented tasks.

Task	Output of		Ground Truth
	PICSO	BERT	
Entity Linking	Mention: "or 10 continues west into <i>beaverton</i> , where it interchanges with oregon route 217, a freeway." Ranked Candidate: [" <i>beaverton, oregon</i> ", " <i>beaverton</i> ", " <i>beaverton, ontario</i> "]	Mention: "or 10 continues west into <i>beaverton</i> , where it interchanges with oregon route 217, a freeway." Ranked Candidate: [" <i>beaverton</i> ", " <i>beaverton, oregon</i> ", " <i>beaverton, ontario</i> "]	beaverton, oregon
Entity Resolution	Entity 1: "TomTom Runner 2 Cardio+Music DBL/LBL (Large) - <i>Sky Captain Blue/Scuba Blue</i> TomTom Running Accessories Blue" Entity 2: "TomTom Runner 2 Cardio GPS Watch with Music Large Strap - <i>Blue Blue</i> " Prediction: Same	Entity 1: "TomTom Runner 2 Cardio+Music DBL/LBL (Large) - <i>Sky Captain Blue/Scuba Blue</i> TomTom Running Accessories Blue" Entity 2: "TomTom Runner 2 Cardio GPS Watch with Music Large Strap - <i>Blue Blue</i> " Prediction: Not Same	Same
Lexical Simplification	Sentence: Women usually notice little change in their breasts, but if you are a man, your breasts may become slightly larger and may be <i>tender</i> . Candidate: [strong, gentle, serious, sensitive , weak, painful]	Sentence: Women usually notice little change in their breasts, but if you are a man, your breasts may become slightly larger and may be <i>tender</i> . Candidate: [strong, gentle, <i>soft</i> , special, weak, sweet]	[sore, sensitive, painful]
KG Canonicalization	{Virginia wesleyan, Virginia wesleyan college }	{Virginia wesleyan college, <i>Columbus college, Rider college</i> }	{Virginia wesleyan, Virginia wesleyan college }
	{Flamingo hotel, Flamingo la vega }	{ <i>Palm casino</i> , Flamingo hotel }	{Flamingo hotel, Flamingo la vega }

the synonyms through a masked self-attention mechanism for learning the semantics of the entity and its context. With the contextual synonym knowledge from Wikidata (general domain) and UMLS (medical domain), PICSO often dramatically outperforms the original PLMs and the other knowledge and synonym injection PLMs on four different similarity-oriented tasks, and can also benefit general NLU tasks in GLUE. In the future, we will investigate a multi-task pre-training paradigm for synonym knowledge injection to better exploit the synonyms widely available in both unstructured text and structured KGs, and will evaluate our methods on more similarity-oriented tasks, e.g. ontology alignment.

Acknowledgement

This research is supported by National Natural Science Foundation of China (Grant No.62276154 and 62011540405), Beijing Academy of Artificial Intelligence (BAAI), the Natural Science Foundation of Guangdong Province (Grant No. 2021A1515012640), Basic Research Fund of Shenzhen City (Grant No. JCYJ20210324120012033), and Overseas Cooperation Research Fund of Tsinghua Shenzhen International Graduate School (Grant No. HW2021008).

CRedit authorship contribution statement

Yangning Li: Conceptualization of this study, Methodology, Experiments. **Jiaoyan Chen:** Conceptualization of

this study, Methodology, Experiments. **Yinghui Li:** Investigation process, Experimental verification. **Tianyu Yu:** Investigation process, Experimental verification. **Xi Chen:** Revision of the paper, Funding Support. **Hai-Tao Zheng:** Revision of the paper, Funding Support.

References

- [1] Amizadeh, S., Palangi, H., Polozov, A., Huang, Y., Koishida, K., 2020. Neuro-symbolic visual reasoning: Disentangling, in: International Conference on Machine Learning, PMLR. pp. 279–290.
- [2] Bai, J., Wang, Y., Chen, Y., Yang, Y., Bai, J., Yu, J., Tong, Y., 2021. Syntax-bert: Improving pre-trained transformers with syntax trees, in: Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, pp. 3011–3020.
- [3] Balntas, V., Riba, E., Ponsa, D., Mikolajczyk, K., . Learning local feature descriptors with triplets and shallow convolutional neural networks.
- [4] Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka, E.R., Mitchell, T.M., 2010. Toward an architecture for never-ending language learning, in: Twenty-Fourth AAAI conference on artificial intelligence.
- [5] Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I., Specia, L., 2017. Semeval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation, in: Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), pp. 1–14.
- [6] Chen, T., Kornblith, S., Norouzi, M., Hinton, G., 2020. A simple framework for contrastive learning of visual representations, in: International conference on machine learning, PMLR. pp. 1597–1607.
- [7] Cui, W., Xiao, Y., Wang, H., Song, Y., Hwang, S., Wang, W., 2017. Kba: Learning question answering over qa corpora and knowledge bases. Proceedings of the VLDB Endowment 10, 565.

- [8] Dash, S., Rossiello, G., Mihindukulasooriya, N., Bagchi, S., Gliozzo, A., 2021. Open knowledge graphs canonicalization using variational autoencoders, in: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 10379–10394.
- [9] Deng, J., Ding, N., Jia, Y., Frome, A., Murphy, K., Bengio, S., Li, Y., Neven, H., Adam, H., 2014. Large-scale object classification using label relation graphs, in: European conference on computer vision, Springer. pp. 48–64.
- [10] Ferret, O., 2018. Using pseudo-senses for improving the extraction of synonyms from word embeddings, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 351–357.
- [11] 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), pp. 34–45.
- [12] He, K., Fan, H., Wu, Y., Xie, S., Girshick, R., 2020. Momentum contrast for unsupervised visual representation learning, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 9729–9738.
- [13] He, Y., Chen, J., Antonyrajah, D., Horrocks, I., 2022. Bertmap: A bert-based ontology alignment system, in: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 5684–5691.
- [14] Hoffart, J., Yosef, M.A., Bordino, I., Fürstenauf, H., Pinkal, M., Spaniol, M., Taneva, B., Thater, S., Weikum, G., 2011. Robust disambiguation of named entities in text, in: Proceedings of the 2011 conference on empirical methods in natural language processing, pp. 782–792.
- [15] Kassner, N., Schütze, H., 2020. Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 7811–7818.
- [16] Kemker, R., McClure, M., Abitino, A., Hayes, T., Kanan, C., 2018. Measuring catastrophic forgetting in neural networks, in: Proceedings of the AAAI Conference on Artificial Intelligence.
- [17] Kenton, J.D.M.W.C., Toutanova, L.K., 2019. Bert: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of NAACL-HLT, pp. 4171–4186.
- [18] Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A.A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., et al., 2017. Overcoming catastrophic forgetting in neural networks. Proceedings of the national academy of sciences 114, 3521–3526.
- [19] Lauscher, A., Vulić, I., Ponti, E.M., Korhonen, A., Glavaš, G., 2020. Specializing unsupervised pretraining models for word-level semantic similarity, in: Proceedings of the 28th International Conference on Computational Linguistics, pp. 1371–1383.
- [20] Le, P., Titov, I., 2018. Improving entity linking by modeling latent relations between mentions, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 1595–1604.
- [21] Lee, K., He, L., Lewis, M., Zettlemoyer, L., 2017. End-to-end neural coreference resolution, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 188–197.
- [22] Li, B., Zhou, H., He, J., Wang, M., Yang, Y., Li, L., 2020. On the sentence embeddings from pre-trained language models, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 9119–9130.
- [23] Li, Y., Li, Y., He, Y., Yu, T., Shen, Y., Zheng, H.T., 2022a. Contrastive learning with hard negative entities for entity set expansion, in: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Association for Computing Machinery, New York, NY, USA. p. 1077–1086. URL: <https://doi.org/10.1145/3477495.3531954>, doi:10.1145/3477495.3531954.
- [24] Li, Y., Zhou, Q., Li, Y., Li, Z., Liu, R., Sun, R., Wang, Z., Li, C., Cao, Y., Zheng, H.T., 2022b. The past mistake is the future wisdom: Error-driven contrastive probability optimization for chinese spell checking, in: Findings of the Association for Computational Linguistics: ACL 2022, pp. 3202–3213.
- [25] Liu, F., Shareghi, E., Meng, Z., Basaldella, M., Collier, N., 2021. Self-alignment pretraining for biomedical entity representations, in: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 4228–4238.
- [26] Liu, W., Zhou, P., Zhao, Z., Wang, Z., Ju, Q., Deng, H., Wang, P., 2020. K-bert: Enabling language representation with knowledge graph, in: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 2901–2908.
- [27] 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. arXiv preprint arXiv:1907.11692.
- [28] Loureiro, D., Jorge, A.M., 2020. Medlinker: Medical entity linking with neural representations and dictionary matching, in: European Conference on Information Retrieval, Springer. pp. 230–237.
- [29] Ma, Y., Nguyen, K.L., Xing, F.Z., Cambria, E., 2020. A survey on empathetic dialogue systems. Information Fusion 64, 50–70.
- [30] Mrkšić, N., Vulić, I., Séaghdha, D.Ó., Leviant, I., Reichart, R., Gasic, M., Korhonen, A., Young, S., 2017. Semantic specialization of distributional word vector spaces using monolingual and cross-lingual constraints. Transactions of the Association for Computational Linguistics 5, 309–324.
- [31] Nguyen, K.A., Köper, M., im Walde, S.S., Vu, N.T., 2017. Hierarchical embeddings for hypernymy detection and directionality, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 233–243.
- [32] Osborne, D., Narayan, S., Cohen, S.B., 2016. Encoding prior knowledge with eigenword embeddings. Transactions of the Association for Computational Linguistics 4, 417–430.
- [33] Peeters, R., Bizer, C., 2021. Dual-objective fine-tuning of bert for entity matching. Proceedings of the VLDB Endowment 14, 1913–1921.
- [34] 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), pp. 43–54.
- [35] Petroni, F., Rocktäschel, T., Riedel, S., Lewis, P., Bakhtin, A., Wu, Y., Miller, A., 2019. Language models as knowledge bases?, 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), pp. 2463–2473.
- [36] Poerner, N., Waltinger, U., Schütze, H., 2019. Bert is not a knowledge base (yet): Factual knowledge vs. name-based reasoning in unsupervised qa. arXiv preprint arXiv:1911.03681 3.
- [37] Qiang, J., Li, Y., Zhu, Y., Yuan, Y., Wu, X., 2020. Lexical simplification with pretrained encoders, in: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 8649–8656.
- [38] Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al., 2018. Improving language understanding by generative pre-training.
- [39] Riyahi, M., Sohrabi, M.K., 2020. Providing effective recommendations in discussion groups using a new hybrid recommender system based on implicit ratings and semantic similarity. Electronic Commerce Research and Applications 40, 100938.
- [40] Robinson, J., Chuang, Ching-Yao, Sra, S., Jegelka, S., 2021. Contrastive learning with hard negative samples. International Conference on Learning Representations.
- [41] Sun, T., Shao, Y., Qiu, X., Guo, Q., Hu, Y., Huang, X.J., Zhang, Z., 2020. Colake: Contextualized language and knowledge embedding, in: Proceedings of the 28th International Conference on Computational Linguistics, pp. 3660–3670.
- [42] Tang, X., Zhang, J., Chen, B., Yang, Y., Chen, H., Li, C., 2021. Bert-int: a bert-based interaction model for knowledge graph alignment, in: Proceedings of the Twenty-Ninth International Conference on

- International Joint Conferences on Artificial Intelligence, pp. 3174–3180.
- [43] Vashishth, S., Jain, P., Talukdar, P., 2018. Cesi: Canonicalizing open knowledge bases using embeddings and side information, in: Proceedings of the 2018 World Wide Web Conference, pp. 1317–1327.
- [44] Vulić, I., Mrkšić, N., Korhonen, A., 2017. Cross-lingual induction and transfer of verb classes based on word vector space specialisation, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 2546–2558.
- [45] Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., Bowman, S., 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding, in: Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, pp. 353–355.
- [46] Wang, R., Tang, D., Duan, N., Wei, Z., Huang, X.J., Ji, J., Cao, G., Jiang, D., Zhou, M., 2021a. K-adapter: Infusing knowledge into pre-trained models with adapters, in: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pp. 1405–1418.
- [47] Wang, X., Gao, T., Zhu, Z., Zhang, Z., Liu, Z., Li, J., Tang, J., 2021b. Kepler: A unified model for knowledge embedding and pre-trained language representation. Transactions of the Association for Computational Linguistics 9, 176–194.
- [48] Wieting, J., Berg-Kirkpatrick, T., Gimpel, K., Neubig, G., 2019. Beyond bleu: Training neural machine translation with semantic similarity, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 4344–4355.
- [49] Xu, P., Lu, J., 2019. Towards a unified framework for string similarity joins. Proceedings of the VLDB Endowment .
- [50] Yang, X., Gu, X., Lin, S., Tang, S., Zhuang, Y., Wu, F., Chen, Z., Hu, G., Ren, X., 2019. Learning dynamic context augmentation for global entity linking, 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), pp. 271–281.
- [51] 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, pp. 901–911.
- [52] Zhang, H., Chen, Q., Zhang, W., Nie, M., 2022. Hsie: Improving named entity disambiguation with hidden semantic information extractor, in: 2022 14th International Conference on Machine Learning and Computing (ICMLC), pp. 251–257.
- [53] Zhang, N., Deng, S., Cheng, X., Chen, X., Zhang, Y., Zhang, W., Chen, H., Center, H.I., 2021. Drop redundant, shrink irrelevant: Selective knowledge injection for language pretraining., in: IJCAI, pp. 4007–4014.
- [54] Zhang, W., Su, J., Tan, C.L., Wang, W.T., 2010. Entity linking leveraging automatically generated annotation, in: Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010), pp. 1290–1298.
- [55] Zhang, Z., Han, X., Liu, Z., Jiang, X., Sun, M., Liu, Q., 2019. Ernie: Enhanced language representation with informative entities, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 1441–1451.
- [56] Zhou, J., Zhang, Z., Zhao, H., Zhang, S., 2020. Limit-bert: Linguistics informed multi-task bert, in: Findings of the Association for Computational Linguistics: EMNLP 2020, pp. 4450–4461.



Yangning Li received the BEng degree from the Department of Computer Science and Technology, Huazhong University of Science and Technology, in 2020. He is currently working toward a Master's degree with the Tsinghua Shenzhen International Graduate School, Tsinghua University. His research interests include natural language processing and data mining.



Jiaoyan Chen is permanent lecturer at Department of Computer Science, The University of Manchester. He used to be a Senior Researcher at Department of Computer Science, University of Oxford, and got his Ph.D and Bachelor degree in Computer Science and Technology in Zhejiang University. His research interests includes knowledge graph, ontology, symbolic and sub-symbolic reasoning, neural-symbolic AI, etc.



Yinghui Li received the BEng degree from the Department of Computer Science and Technology, Tsinghua University, in 2020. He is currently working toward the PhD degree with the Tsinghua Shenzhen International Graduate School, Tsinghua University. His research interests include natural language processing and deep learning.



Tianyu Yu received the bachelor's degree in software engineering from the Beihang University, CHina. He is currently a graduate student major in computer technology in Tsinghua University. His research interest include information extraction, knowledge representation learning and cross-modal representation learning.



Xi Chen received his PhD degree in computer science from the Zhejiang University. He is currently the head of the cross-modal algorithm center of Tencent Platform and Content Group and mainly focuses on various applications of NLP.



Hai-Tao Zheng received the bachelor's and master's degrees in computer science from the Sun Yat-Sen University, China, and the PhD degree in medical informatics from Seoul National University, South Korea. He is currently an associate professor with the Shenzhen International Graduate School, Tsinghua University, and also with Peng Cheng Laboratory. His research interests include web science, semantic web, information retrieval, and machine learning.