IELM: An Open Information Extraction Benchmark for Pre-Trained Language Models

Chenguang Wang¹, Xiao Liu², Dawn Song³

¹Washington University in St. Louis, ²Tsinghua University, ³UC Berkeley
chenguangwang@wustl.edu, liuxiao21@mails.tsinghua.edu.cn, dawnsong@berkeley.edu

Abstract

We introduce a new open information extraction (OIE) benchmark for pre-trained language models (LM). Recent studies have demonstrated that pre-trained LMs, such as BERT and GPT, may store linguistic and relational knowledge. In particular, LMs are able to answer “fill-in-the-blank” questions when given a pre-defined relation category. Instead of focusing on pre-defined relations, we create an OIE benchmark aiming to fully examine the open relational information present in the pre-trained LMs. We accomplish this by turning pre-trained LMs into zero-shot OIE systems. Surprisingly, pre-trained LMs are able to obtain competitive performance on both standard OIE datasets (CaRB and Re-OIE2016) and two new large-scale factual OIE datasets (TAC KBP-OIE and Wikidata-OIE) that we establish via distant supervision. For instance, the zero-shot pre-trained LMs outperform the F1 score of the state-of-the-art supervised OIE methods on our factual OIE datasets without needing to use any training sets.¹

1 Introduction

Pre-trained language models (LM), such as BERT (Devlin et al., 2018) and GPT-3 (Brown et al., 2020), have revolutionized NLP over the last several years and advanced the state-of-the-art results in a wide set of downstream NLP tasks. Recent studies show that a considerable amount of linguistic (Hewitt and Manning, 2019; Clark et al., 2019) and relational knowledge (Petroni et al., 2019; Talmor et al., 2019; Jiang et al., 2020; Petroni et al., 2020) has been captured by the pre-trained LMs via pre-training on large-scale textual corpora. These approaches often design “fill-in-the-blank” questions based on pre-defined relations. For example, a question “Bob Dylan was born in” is manually created for the LMs to answer the “birthplace” relation of “Bob Dylan”.

Most existing approaches that evaluate what pre-trained LMs have learned are based on benchmarks with pre-defined relation categories. Yet, the benchmarks present two limitations. First, most benchmarks only cover a limited number of pre-defined relations. Therefore, it is unclear whether the pre-trained LMs have stored general open relation information. For example, the Google-RE in LAMA benchmark (Petroni et al., 2019) includes only three relations (i.e., “birthplace”, “birthdate”, and “deathplace”), while there are hundreds of relations available in the real world scenario. Second, a majority of benchmarks evaluate LMs in a close manner. This means that the gold relation is given to the models. For example, “was born in” is given as the model’s input. Besides, the existing benchmarks often provide a single gold relation for each input sentence. However, an input sentence may indicate multiple relations, e.g., containing both “birthplace” and “birthdate” information about an argument or entity. We are curious: instead of the limited relational information, can we systematically benchmark the general information stored in the pre-trained LMs?

In this work, we set up a new open information extraction (OIE) benchmark, called IELM, towards testing the general and open relational information stored in pre-trained LMs. We refer to OIE as it is a task that is designed to extract open relations from massive corpora without requiring a pre-defined relation category. As shown in Figure 1, we successfully convert pre-trained LMs to zero-shot OIE systems. We apply them to two standard OIE datasets, including CaRB (Bhardwaj et al., 2019) and Re-OIE2016 (Stanovsky and Dagan, 2016; Zhan and Zhao, 2020), as well as two new large-scale factual OIE datasets in our IELM benchmark. We show that the zero-shot pre-trained LMs outperform the fully supervised state-of-the-arts on fac-

¹Our code and datasets are available at https://github.com/cgraywang/IELM.
DylanNP was born in MinnesotaNP, and was awarded Nobel PrizeNP.

Figure 1: Summary of our approach. The zero-shot open information extraction system takes a noun phrase (NP) chunked sentence as input, and outputs a set of triples. The approach first conducts argument extraction by encoding NPs as argument pairs, then performs predicate extraction via decoding using the parameters (i.e., attention scores) of the pre-trained language models. The output extractions are then evaluated on our IELM benchmark.

2. Language Models as Zero-Shot Information Extractors

For open information extraction (OIE), we take an input as a NP-chunked sentence and output a set of triples. Below is an example.

Input: DylanNP was born in MinnesotaNP, and was awarded Nobel PrizeNP.

Output: (Dylan; born in; Minnesota), (Dylan; awarded; Nobel Prize).

NP denotes the noun phrase.

2.1 Argument Extraction

Follow traditional linguistic OIE systems such as Stanford OpenIE (Angeli et al., 2015) and OpenIE5 (Saha et al., 2017, 2018), we treat each NP pair as an argument pair (e.g., “Dylan” and “Minnesota”). We then utilize the parameters of LMs to extract the predicates (or relations) between the arguments. To the best of our knowledge, this is the first attempt to systematically evaluate pre-trained LMs in a zero-shot OIE setting. To summarize, our key contributions are the following.

1. We benchmark the general relational information in pre-trained LMs on our IELM benchmark. Besides two standard OIE datasets (CaRB and Re-OIE2016), we also create two large-scale factual OIE datasets for our benchmark. The two new OIE datasets are called TAC KBP-OIE and Wikidata-OIE, which are constructed via distant supervision from two knowledge graphs (TAC KBP and Wikidata). Our benchmark is a general OIE benchmark, helping the development of future OIE systems.

2. We enable the zero-shot capabilities of pre-trained LMs for OIE by encoding the arguments in the input and decoding predicates using the parameters of pre-trained LMs. The pre-trained LMs are particularly good at recovering factual arguments and predicates.

3. We test the OIE performance of 6 pre-trained LMs (BERT and GPT-2 (Radford et al., 2019) families) and 14 OIE systems on IELM benchmark. The zero-shot LMs achieve state-of-the-art OIE performance on TAC KBP-OIE and Wikidata-OIE, even outperforming fully supervised OIE systems.
the average length of the candidates. In general, the beam search starts with the first argument (e.g., “Dylan”). At each step, beam search simply selects top-$k$ next tokens with the largest attention scores, and just keeps $k$ partial candidates with the highest scores, where $k$ is the beam size. When a candidate produces the second argument (e.g., “Minnesota”), the candidate is complete.

We show a running example as follows. Let’s first consider the search from left to right with beam size equal to 1. An example search process is shown in Figure 2. Given an argument pair “Dylan” and “Minnesota”, at each step, the search performs one of the following actions:

- **START** the search from first argument. The first argument is added as an initial candidate into the beam. In Figure 2(a), at step 0, “Dylan” is added into the beam. The total attention score is initialized to 0.

- **YIELD** a new partial candidate in the beam if the current candidate has not reached the second argument. This action conducts the following: The next largest attended token is appended to the end of the current candidate to yield the new candidate. The total score is increased by the associated attention score. At step 1 of Figure 2(a), “born” is appended to the current candidate to yield the partial candidate, since “born” has the largest attention score (0.2 as highlighted in Figure 2(b)) with “Dylan” in the attention matrix. The total score becomes 0.2. Note that we only consider the single head attention in this example for simplicity. “X” in Figure 2(b) marks the tokens (prior to the current token) that are not considered in the search to prevent searching backward. Step 2 takes the same action, and the score becomes 0.5.

- **STOP** the search step if the candidate has reached the second argument, then add the candidate as a valid triple into the beam. As beam size equals to 1, (Dylan; born in; Minnesota) is returned for the given pair. The final score of the triple is 0.7.

We also notice triples are often in reverse order in the sentence, thus enabling bidirectionality by running the algorithm in both directions (left to right and right to left). We merge the subwords as words, and only consider word-level attention. The beam search is implemented by the breadth-first search, which is efficient as the time complexity is $O(k \cdot d)$. $d$ is the maximum depth of the search tree.

3 The IELM Benchmark

3.1 Datasets

3.1.1 Standard OIE

We adopt two standard OIE datasets below.

CaRB CaRB (Bhardwaj et al., 2019) is a crowdsourced OIE dataset, where the input sentences are from the OIE2016 (Stanovsky and Dagan, 2016).

Re-OIE2016 Re-OIE2016 (Zhan and Zhao, 2020) is also generated based on the input sentences in the OIE2016, and is further enhanced by human annotations.

3.1.2 Factual OIE

In addition, we introduce two large-scale factual OIE datasets based on knowledge graphs (KG).
Table 1: Evaluation of unsupervised entity linking of Wikidata-OIE on AIDA benchmark. An asterisk (*) indicates a supervised method.

| Method                  | AIDA                      | dev | test |
|-------------------------|---------------------------|-----|------|
| Spitkovsky and Chang 2012 | 26.0                      | 28.2 |      |
| Kolitsas et al. 2018∗   | -                         | 82.4 |      |
| Ours                    | 63.8                      | 64.5 |      |

Table 2: Dataset statistics of the IELM benchmark.

| Dataset                  | # of triples | # of arguments | # of predicates | # of documents |
|--------------------------|--------------|----------------|-----------------|---------------|
| Re-OIE2016               | 1,508        | 3,328          | 1,506           | 595           |
| CaRB                     | 2,715        | 6,226          | 2,715           | 641           |
| TAC KBP-OIE              | 27,655       | 39,661         | 41              | 3,877,207     |
| Wikidata-OIE             | 27,368,562   | 6,047,494      | 1,156           | 6,047,494     |

3.2 Pre-Trained Language Models for OIE

Unidirectional Language Models Given an input sequence \( x = \{x_1, x_2, \ldots, x_N\} \), unidirectional LMs assign a joint probability to the sequence by factoring it as \( p(x) = \prod_t p(x_t | x_{t-1}, \ldots, x_1) \), where \( p(x_t | x_{t-1}, \ldots, x_1) = \sigma(W_h x_t + b) \). \( h_t \) is the output vector of a neural network at position \( t \).

We consider GPT-2 (Radford et al., 2019), where \( h_t \) is produced by Transformer decoders (Vaswani et al., 2017). GPT-2 is pre-trained on WebText containing 40GB of text. We explore all four pre-trained GPT-2s with different model sizes: GPT-2 (117M), GPT-2Absolutely (345M), GPT-2Absolutely (774M), and GPT-2Absolutely (1558M).

Bidirectional Language Models Different from unidirectional LMs that predict the next word given the previous words, bidirectional LMs take both left and right context of the target word into consideration, formally, \( p(x_t) = p(x_t | x_1, \ldots, x_{t-1}, x_{t+1}, \ldots, x_N) \).

We use BERT (Devlin et al., 2018) that enables bidirectional context modeling via a masked LM objective and utilizing the Transformer architecture. BERT is pre-trained on BooksCorpus and English Wikipedia. We use both its pre-trained settings: BERT\textsubscript{BASE} (109M) and BERT\textsubscript{LARGE} (335M).

3.3 Comparison Methods

We compare our method with a wide set of OIE systems including both neural and traditional linguistic OIE systems. Most OIE systems are based on supervised learning, which are indicated with asterisks (*) in Table 3. We provide details of the comparison systems in Appendix A.5.

3.4 Evaluation Method

3.4.1 Standard OIE

On CaRB and Re-OIE2016, we follow the original evaluation proposed in (Bhardwaj et al., 2019) and (Stanovsky and Dagan, 2016; Zhan and Zhao, 2020), and report precision, recall, F1, area under the curve (AUC) for compared OIE systems. AUC
is calculated from a plot of the precision and recall values for all potential confidence thresholds. The F1 is the maximum value among the precision-recall pairs. We follow the matching function proposed for each dataset, i.e., lexical match for Re-OIE2016, and tuple match for CaRB. The CaRB evaluation function is stricter as it penalizes long extractions.

3.4.2 Factual OIE

We report precision, recall, and F1 of the OIE systems on two large-scale factual OIE datasets: TAC KBP-OIE and Wikidata-OIE. We introduce exact match as the matching function for them as below.

Matching Function The matching functions for standard OIE datasets are generally flexible. For example, the lexical match of Re-OIE2016 judges an argument or predicate as correct if and only if it includes the syntactic head of the gold argument or predicate. Unlike these matching functions, our exact matching function requires both arguments and predicates are linked to the gold extractions.

For TAC KBP-OIE, we judge an argument to be correct if and only if it matches the name of the gold argument and the span position of the gold argument in the sentence. The main challenge is how to properly link a predicate, since there are often many ways to express it. We follow Stanford OpenIE (Angeli et al., 2015) to produce the predicate mapping between the OIE relations and TAC KBP predicates. A predicate is correct if the pair of OIE relation and gold predicate exists in the predicate mapping. The predicate mapping is constructed in two steps. First, a collection of predicate mappings was constructed by a single annotator in approximately a day. Second, predicate mappings were finalized through the following learning procedure. This process matches OIE relations to the TAC KBP predicates by searching for co-occurrent relations in a large distantly-labeled corpus, and decides pairs of OIE relations and TAC KBP predicates that have a high PMI. The basic idea is that the more often the argument pairs of the triples and TAC KBP triples are linked, the more likely the corresponding relations or predicates are linked to each other. Example predicate mappings are shown in Appendix A.4.

For Wikidata-OIE, we link an argument based on the entity linker used in Wikidata-OIE construction (Sec. 3.1). An argument is correct if the linked argument matches the name of the gold argument and the span position of the gold argument in the sentence. The predicate mapping is bootstrapped from TAC KBP-OIE’s mapping. In addition, we normalize each predicate phrase of the triples by lemmatization, and removing inflection, auxiliary verbs, adjectives, adverbs. One author manually filters out the bad predicate pairs. This process takes approximately a day. A predicate is correct if the OIE to gold predicate pair exists in the bootstrapped predicate mapping. An annotator randomly subsampled and checked 100 aligned triple-sentence pairs and concluded with a 93% accuracy of extracted triples.

4 Results

In this section, we show that pre-trained LMs are effective zero-shot OIE systems, and exceed the previous state-of-the-art OIE systems on our large-scale factual OIE datasets in IELM benchmark. To keep our evaluation as simple as possible, the hyperparameters and settings are shared across datasets. More experimental details are described in the appendix.
4.1 OIE Results

Table 3 shows the results. While zero-shot OIE systems synthesized by pre-trained LMs obtain notably lower scores compared to previous OIE systems on standard OIE datasets, they outperform the previous OIE systems on factual OIE datasets. We also find that larger LMs obtain improved results on all datasets. For example, BERT\textsubscript{LARGE} outperforms BERT\textsubscript{BASE} due to its larger model size. GPT-2s share similar trends. This is because larger LMs store richer relational information. This finding is consistent with previous studies (Petroni et al., 2019, 2020).

4.1.1 Standard OIE

The main reasons for the degraded performance of pre-trained LMs on standard OIE datasets are three-fold. First, the comparison methods mainly involve supervised systems that are trained on OIE datasets, which are denoted with asterisks (*) in Table 3. Besides the supervised systems, the remaining comparison systems all require human involvement, such as providing linguistic patterns for the extraction. In contrast, the pre-trained LMs are used as zero-shot OIE systems without using any training sets. Second, the zero-shot OIE modules still have room to improve. For example, approximately 30.0% of the argument extraction errors are due to the spaCy noun chunker. 16.9% of the gold extractions contain predicates outside the argument pairs. The current predicate extraction only allows searching between the arguments, and thus cannot handle such cases. Third, standard OIE benchmarks such as CaRB and Re-OIE2016 mainly examine the general information extraction capability. The zero-shot approach is not able to recall the information of interest in LMs. We might need a specific module (e.g., ranking) to locate such information. Interestingly, pre-trained LMs achieve comparable performance with supervised Stanford OpenIE. The result indicates pre-trained LMs contain informative patterns that are useful for OIE.

4.1.2 Factual OIE

As shown in Table 3, the best zero-shot OIE system based on GPT-2\textsubscript{XL} obtains a +2.6% and a +3.1% absolute F1 improvement on TAC KBP-OIE and Wikidata-OIE respectively over the previous supervised state-of-the-art. Due to the computation cost of OIE systems (Sec. 4.3), we only select several best performed OIE systems on the standard OIE datasets for the large-scale OIE experiments including: linguistic OIE systems (OpenIE4, OpenIE5, Stanford OpenIE) and neural OIE systems (RrnOIE, IMOJIE, OpenIE6).

Compared to the results on standard OIE datasets, pre-trained LMs consistently achieve state-of-the-art performance on both datasets. Both datasets emphasize measuring factual arguments and predicates in the reference KGs. Previous studies (Petroni et al., 2019, 2020) show that LMs have stored a considerable amount of factual information via pre-training on large-scale text. We draw the same conclusion. To the best of our knowledge, our IELM benchmark is the first benchmark that includes factual OIE datasets. More importantly, both
linguistic and neural OIE systems are derived from manually designed linguistic patterns or learned patterns. The result shows that the pre-trained attention weights capture a more flexible set of factual patterns. The result also suggests that our approach is capable of using such patterns. In order to scale our approach to large-scale datasets, the argument and predicate extraction are both efficient by design. In particular, the beam search for predicate extraction is efficient in exploring the relational sequences in the input sentence. Besides, the attention scores used in the beam search are produced via a single forward pass of the pre-trained LM over the input sentence without fine-tuning.

Moreover, we find that BERT LMs outperform GPT-2 LMs under similar model sizes. On both datasets, BERT_{BASE} performs better than GPT-2 in F1, and BERT_{LARGE} outperforms GPT-2_{MEDIUM} in F1. This is mainly because the recall of BERT LMs is higher than that of corresponding GPT-2 LMs. The result indicates that the Cloze-style loss function (i.e., masked LM) of BERT is more effective and flexible in recovering information than the autoregressive LM objective. We also notice that the precision of GPT-2 LMs is higher than that of BERT LMs. The reason is that the autoregressive LM objective captures more accurate information than Cloze-style loss does by preventing extra noise (e.g., masked tokens) in pre-training.

Pre-trained LMs achieve competitive precision, e.g., the precision is greater than 60% on TAC KBP-OIE. However, only moderate recalls are obtained. Therefore, improving recall is clearly the future direction. We find that both argument and predicate extraction can be further improved. For example, the main cause of the moderate recall is the incorrect arguments caused by spaCy noun chunks as summarized in Sec. 4.2. Besides, we can incorporate predicates that are not between the argument pairs into the extractions, as we observe a number of gold triples are in inverted sentences. We also notice that the F1 gain over previous state-of-the-arts on TAC KBP-OIE is smaller compared to that on Wikidata-OIE. A larger text corpus, e.g., Wikipedia, provides more information. We could improve the recall by running on larger corpora such as WebText2 and Common Crawl (Raffel et al., 2019; Brown et al., 2020) to collect more triples.

### 4.2 Error Analysis

There is still significant room to improve the results. We argue that we are measuring a lower bound for what LMs know. To further understand the shortcomings of the current method, we conduct an error analysis of the errors in precision on all datasets. We choose BERT_{LARGE} for the study. We sample 100 documents from the Wikidata-OIE dataset, and manually check the reasons for the errors. We find 33.1% of the errors are caused by incorrect arguments, while the predicate phrases are correct. The errors are due to the incorrect noun chunks detected by the spaCy. 18.3% of the errors are due to the missing pairs in predicate mapping. We also note that approximately 23.8% of the errors are actually correct triples that are not covered by Wikidata. For example, \((\text{Bob Dylan}, \text{residence}, \text{Nashville})\) does not exist in Wikidata, but it is a correct triple. The rest of the errors made by BERT_{LARGE} are incorrect predicate phrases, such as uninformative phrases. We find similar errors are made by BERT_{LARGE} on other datasets. Based on the above analysis, enhancing argument detection and predicate mapping is helpful to further improve the results.

### 4.3 Runtime Analysis

The runtime of OIE systems is crucial in practice. We test the runtime of different OIE systems on Re-OIE2016. The results are in Table 4. We find ours is competitive in terms of efficiency given the size of the models.

### 4.4 Parameter Study

We study the effects of the key parameters using BERT_{BASE} on TAC KBP-OIE as shown in Figure 3. We randomly sample 20% of the oracle query entities (provided by TAC KBP) as a hold-out
5 Related Work

Pre-trained language models (LM), e.g., BERT (Devlin et al., 2018), GPT (Radford et al., 2018, 2019), and large LMs over 100B parameters (Brown et al., 2020; Chowdhery et al., 2022; Zeng et al., 2022) contain growing amount of linguistic and factual knowledge obtained via pre-training on large-scale corpora. To evaluate their abilities, researchers have created many knowledge benchmarks. LAMA leverages manually created prompts (Petroni et al., 2019, 2020). Recent studies have also developed soft prompts (Liu et al., 2021; Zhong et al., 2021)) for fact retrieval. KILT (Petroni et al., 2021) proposes a knowledge-intensive benchmark concerning several downstream tasks to evaluate LMs’ ability in capturing knowledge. Wang et al. (2022) have utilized a set of knowledge-intensive structure prediction tasks to evaluate the knowledge in pre-trained LMs. Shen et al. (2022) have adapted KG completion as a benchmark to evaluate LMs. Besides relational knowledge, closed-book OpenQA (Roberts et al., 2020) benchmarks (in which LMs answer the open-domain questions without retrieving contexts) have also been adopted as a way to evaluate LMs’ knowledge. While the existing benchmarks evaluate LMs in an implicit way, the main difference is that our benchmark explicitly and interpretably evaluates triples from the textual corpora extracted using model parameters (e.g. attentions). In the field of neural network interpretation (Linzen et al., 2016; Adi et al., 2016; Tenney et al., 2019), in particular the pre-trained deep LM analysis, substantial recent work focuses on both visualizing and analyzing the attention (Vig, 2019; Jain and Wallace, 2019; Clark et al., 2019; Michel et al., 2019; Ramsauer et al., 2020). Instead of analyzing or visualizing, our benchmark quantitatively evaluates the relational information with respect to open information extraction.

Open information extraction systems, e.g., OLLIE (Schmitz et al., 2012), Reverb (Fader et al., 2011), Stanford OpenIE (Angeli et al., 2015), OpenIE 5 (Saha et al., 2017, 2018), RnnOIE (Stanovsky et al., 2018), and OpenIE 6 (Kolluru et al., 2020a) aim to extract triples from web corpora for open schema KGs. Besides, NELL (Carlson et al., 2010), DeepDive (Niu et al., 2012), Knowledge Vault (Dong et al., 2014) extract information based on a fixed schema or ontology, where humans help improve the accuracy of the extractions. Probase (Wu et al., 2012) produces taxonomies instead of rich typed relations in general KGs. Our benchmark first evaluates LMs’ unsupervised information extraction ability on common open information extraction datasets such as CaRB (Bhardwaj et al., 2019) and Re-OIE2016 (Zhan and Zhao, 2020), and then aligns the extracted triples to KG triples for large-scale knowledge extraction benchmark construction. Our algorithm is similar to...
the generation algorithm in DeepEx (Wang et al., 2021). The focus of this work is to benchmark the zero-shot OIE performance of pre-trained LMs on both standard and factual OIE datasets. To further improve the OIE performance, the ranking module in DeepEx can be useful. The structure pre-training proposed in (Wang et al., 2022) can also be helpful.

6 Conclusion

We benchmark the general relational information in pre-trained language models (LM) in an open information extraction (OIE) setup. We find that the pre-trained LMs contain a considerable amount of open relational information through large-scale evaluation on both standard OIE datasets and newly created large-scale factual OIE datasets in our IELM benchmark. We are able to turn pre-training LMs into zero-shot OIE systems to efficiently deliver the benchmark results. The reach of this result is broad and has potential downstream utility for deep neural network interpretation, information extraction, and knowledge graph construction. Although the results are promising, we argue that our results just indicate a lower bound about what the LMs have. We hope our results will foster further research in the LM OIE benchmark direction.

7 Limitations

For the limitations of our method, the argument extraction module of our algorithm relies on a third-party noun chunker. As reported, the noun chunker introduces the majority of the errors in our extraction results. A limitation in our benchmark is that we have not conducted a large-scale manual evaluation of our factual OIE datasets (TAC KBP-OIE and Wikidata-OIE). The main focus of our study is to provide a large-scale OIE benchmark. As a result, this makes our benchmark more challenging to be used than standard OIE datasets in terms of computation costs and infrastructure. Finally, we have only benchmarked BERT and GPT-2 on our datasets. Future work could include testing a wide range of language models on our benchmark.

8 Ethical Considerations

We hereby acknowledge that all of the co-authors of this work are aware of the provided ACM Code of Ethics and honor the code of conduct. This work is about benchmarking the zero-shot OIE capability of pre-trained language models including BERT and GPT. Our ethical considerations and the work’s underlying future impacts are discussed in the following perspectives. Language models are known to present potential risks and limitations (Brown et al., 2020), and the corpus used in pre-training (such as Wikipedia) may introduce unwanted biases and toxicity. We do not anticipate the production of harmful outputs after using our method or datasets, especially for vulnerable populations.

9 Environmental Impact

We adopt the pre-trained language models BERT (Devlin et al., 2018) and GPT-2 series (Radford et al., 2019) in our IELM benchmark. The models’ carbon footprints are estimated to be 22–28 kilograms (Gibney, 2022). Additionally, The focus of this study is to test the zero-shot OIE ability of pre-trained language models. We do not train language models on massive datasets. Instead, we only do inference on a few evaluation datasets. This cost is less than 0.1% energy than that of their pre-training. This demonstrates that developing proper zero-shot learning strategies for large language models can not only deepen our understanding of their latent mechanisms, but also further reduce the energy consumption and environmental impacts that language models with ever-growing size may cause.

Acknowledgements

We would like to thank the anonymous reviewers for their suggestions and comments. This material is in part based upon work supported by Berkeley DeepDrive and Berkeley Artificial Intelligence Research.

References

Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. 2016. Fine-grained analysis of sentence embeddings using auxiliary prediction tasks. arXiv preprint arXiv:1608.04207.

Gabor Angeli, Melvin Jose Johnson Premkumar, and Christopher D Manning. 2015. Leveraging linguistic structure for open domain information extraction. In ACL, pages 344–354.

Sangnie Bhardwaj, Samarth Aggarwal, and Mausam Mausam. 2019. Carb: A crowdsourced benchmark for open ie. In EMNLP, pages 6262–6267.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dharwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda
Askell, et al. 2020. Language models are few-shot learners. arXiv preprint arXiv:2005.14165.

Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R Hruschka Jr, and Tom M Mitchell. 2010. Toward an architecture for never-ending language learning. In AAAI, 3.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. arXiv preprint arXiv:2204.02311.

Janara Christensen, Stephen Soderland, and Oren Etzioni. 2011. An analysis of open information extraction based on semantic role labeling. In KCAP, pages 113–120.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. 2019. What does bert look at? an analysis of bert’s attention. arXiv preprint arXiv:1906.04341.

Lei Cui, Furu Wei, and Ming Zhou. 2018. Neural open information extraction. In ACL, pages 407–413.

Luciano Del Corro and Rainer Gemulla. 2013. Clausie: clause-based open information extraction. In WWW, pages 355–366.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. 2014. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In KDD, pages 601–610.

Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In EMNLP, pages 1535–1545.

Kril Gashteovski, Rainer Gemulla, and Luciano del Corro. 2017. Minie: minimizing facts in open information extraction. ACL.

Elizabeth Gibney. 2022. How to shrink ai’s ballooning carbon footprint. Nature, 607(7920):648–648.

John Hewitt and Christopher D Manning. 2019. A structural probe for finding syntax in word representations. In NAACL, pages 4129–4138.

Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenauf, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. 2011. Robust disambiguation of named entities in text. In EMNLP, pages 782–792.

Sarthak Jain and Byron C Wallace. 2019. Attention is not explanation. arXiv preprint arXiv:1902.10186.

Zhenghao Jiang, Frank F Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know? TACL, 8:423–438.

Nikolaos Kolitsas, Octavian-Eugen Ganea, and Thomas Hofmann. 2018. End-to-end neural entity linking. arXiv preprint arXiv:1808.07699.

Keshav Kolluru, Vaibhav Adlakha, Samarth Aggarwal, Soumen Chakrabarti, et al. 2020a. Openie6: Iterative grid labeling and coordination analysis for open information extraction. In EMNLP, pages 3748–3761.

Keshav Kolluru, Samarth Aggarwal, Vipul Rathore, Soumen Chakrabarti, et al. 2020b. Imojiie: Iterative memory-based joint open information extraction. In ACL, pages 5871–5886.

Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. 2016. Assessing the ability of lstms to learn syntax-sensitive dependencies. TACL, pages 521–535.

Paul Michel, Omer Levy, and Graham Neubig. 2019. Are sixteen heads really better than one? In NIPS, pages 14014–14024.

Feng Niu, Ce Zhang, Christopher Ré, and Jude Shavlik. 2012. Elementary: Large-scale knowledge-base construction via machine learning and statistical inference. International Journal on Semantic Web and Information Systems (IJSWIS), 8(3):42–73.

Fabio Petroni, Patrick Lewis, Aleksandra Piktus, Tim Rocktäschel, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. 2020. How context affects language models’ factual predictions. arXiv preprint arXiv:2005.04611.

Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Maillard, et al. 2021. Kilt: a benchmark for knowledge intensive language tasks. In NAACL, pages 2523–2544.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. 2019. Language models as knowledge bases? arXiv preprint arXiv:1909.01066.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. OpenAI Blog, (8):9.
Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. CoRR, abs/1910.10683.

Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, Lukas Gruber, Markus Holzleitner, Milena Pavlović, Geir Kjetil Sandve, Victor Greiff, et al. 2020. Hopfield networks is all you need. arXiv preprint arXiv:2008.02217.

Youngbin Ro, Yukyung Lee, and Pilsung Kang. 2020. Multi^2oie: Multilingual open information extraction based on multi-head attention with bert. In Findings of EMNLP, pages 1107–1117.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? arXiv preprint arXiv:2002.08910.

Arpita Roy, Youngja Park, Taesung Lee, and Shimei Pan. 2019. Supervising unsupervised open information extraction models. In EMNLP, pages 728–737.

Swarnadeep Saha et al. 2018. Open information extraction from conjunctive sentences. In COLING, pages 2288–2299.

Michael Schmitz, Stephen Soderland, Robert Bart, Oren Etzioni, et al. 2012. Open language learning for information extraction. In EMNLP, pages 523–534.

Jianhao Shen, Chenguang Wang, Linyuan Gong, and Dawn Song. 2022. Joint language semantic and structure embedding for knowledge graph completion. In COLING.

Valentin I Spitkovsky and Angel X Chang. 2012. A cross-lingual dictionary for english wikipedia concepts.

Gabriel Stanovsky and Ido Dagan. 2016. Creating a large benchmark for open information extraction. In EMNLP, pages 2300–2305.

Gabriel Stanovsky, Jessica Ficler, Ido Dagan, and Yoav Goldberg. 2016. Getting more out of syntax with props. arXiv preprint arXiv:1603.01648.

Gabriel Stanovsky, Julian Michael, Luke Zettlemoyer, and Ido Dagan. 2018. Supervised open information extraction. In NAACL, pages 885–895.

Mihai Surdeanu. 2013. Overview of the tac2013 knowledge base population evaluation: English slot filling and temporal slot filling. TAC, page 2.

Alon Talmor, Yanai Elazar, Yoav Goldberg, and Jonathan Berant. 2019. olympics–on what language model pre-training captures. arXiv preprint arXiv:1912.13283.
A.1.1 Entity Linking

We use an unsupervised entity linker for both Wikidata-OIE dataset construction and OIE evaluation. The entity linker is originally developed in (Spitkovsky and Chang, 2012), which is based on a mention-to-entity dictionary. We build an enhanced dictionary as follows: we add new Wikipedia anchors to the dictionary which results in 26 million entries compared to the original 21 million entries. Then a Wikipedia anchor to the Wikidata item dictionary is used to further link the entities (or arguments) to Wikidata. If an argument is a pronoun, we further use neuralcoref 2 for coreference resolution.

A.1.2 Predicate Mapping

The predicate mapping of Wikidata-OIE is constructed offline using the method in Sec. 3.4. In more detail, we randomly sampled a hold-out dataset including 2,000 documents from English Wikipedia for the bootstrapped predicate mapping construction based on the TAC KBP mapping (Angeli et al., 2015). To filter out the wrong predicate pairs, we manually check whether the top predicate phrases are true.

A.1.3 Gold Triples

For gold triples in Wikidata-OIE, we only preserve those triples describing predicates between arguments that can be linked to corresponding Wikipedia anchors. We rule out triples of attributes about arguments and triples of auxiliary predicates (such as topic’s main category).P901 and finally result in 27,368,562 gold triple extractions.

A.1.4 Evaluation

Given the large number of source sentences and gold triples in Wikidata-OIE, a MongoDB database is maintained to store the gold triples to enable an efficient evaluation.

A.2 Zero-Shot Language Model Based Open Information Extraction

In this section, we introduce additional details about how we adapt pre-trained language models (LM) as zero-shot OIE systems.

A.2.1 Argument Extraction

We use spaCy noun chunker 3 to annotate the noun phrases in the sentences.

A.2.2 Predicate Extraction

We first describe predicate extraction introduced in Sec. 2.2 in detail.

- **Beam Search.** The inputs of the search algorithm are an argument pair \((arg_0, arg_1)\), a sentence \(s\), an attention matrix \(A_s\), of \(s\). Both \(arg_0\) and \(arg_1\) are identified by the noun chunker in \(s\). \(A_s\) is the attention matrix associated with \(s\) from the forward pass of an LM without fine-tuning. The search gets started by adding the first argument \(arg_0\) as the initial candidate in the beam. While there are still new candidates waiting to be yielded, the search continues, and the top \(k\) candidates sorted by the attention scores are maintained in the beam. The details of the proposed beam search are described in Algorithm 1. In practice, we implement an action manager \(O\) to decide which action to take at each step. Given a candidate \(c\) in the beam, \(O(c) = \text{START}\) always happens at the beginning of the search. If \(c\) has not reached the second argument \(arg_1\) yet, \(O(c) = \text{YIELD}\). Otherwise, \(O(c) = \text{STOP}\).

- **Implementation Details.** For Wikidata-OIE, we randomly split the English Wikipedia data into 20 partitions, and map the data partitions to 20 distributed servers to run. Each server is configured with four Tesla K80 12Gs. We set the maximum sequence length to 256, and batch size as 32 for BERT\_LARGE and 4 for GPT-2\_XL. We use implementations of pre-trained LMs in the

---

2 https://github.com/huggingface/neuralcoref

3 https://spacy.io/usage/linguistic-features/#noun-chunks
Transformers package \(^4\). We use spaCy sentence tokenizer \(^5\) to segment the documents into sentences. \(\text{BERT}_{\text{LARGE}}\) takes approximately 48 hours, and \(\text{GPT-2}_{\text{XL}}\) costs around 96 hours. The resulting triples from the 20 servers are then reduced to a data server. The batch sizes of \(\text{BERT}_{\text{BASE}}, \text{GPT-2}, \text{GPT-2}_{\text{MEDIUM}}, \text{GPT-2}_{\text{LARGE}}\) are 64, 32, 16, 8 respectively.

### A.2.3 Parameter Settings

We then discuss the parameter setup of our OIE systems as below.

The parameter settings are shared across all OIE datasets. All the choices are based on the parameter study in Sec. 4.4. The beam size of Algorithm 1 is set to 6. The attention score threshold is set to 0.005, and the number of relation/predicate frequencies is set to 10. To generate the attention weight matrix \(A_s\) of a sentence, we reduce the weights of every attention head in the last layer of pre-trained LMs using the mean operator. We analyze the effects of various parameters below.

Figure 3(a) illustrates the effects of various beam sizes in Algorithm 1. We find that in general, the larger the beam size is, the better F1 the setting achieves. This is because our method is able to reserve more potentially correct triples when more candidates are allowed. However, F1 improvement gradually becomes subtle, while the computation costs increase more significantly. For efficiency consideration, we do not explore larger beam sizes. We set the beam size as 6.

Figure 3(b) compares the effect of different thresholds of the total score. We set the threshold as 0.005 since it achieves the best result. Note that the summed attention score is normalized by the length of the triple to penalize the cumbersome triples. The threshold is effective. This is mainly because of the relational information contained in the self-attention matrix: the score in the attention matrix is representing the chance of the triples to be the true triples based on the stored information. Figure 3(c) shows the impact of the predicate frequency threshold in identifying common predicates. The best result is achieved when it equals 10. This shows that while our method mostly identifies frequent predicates, it is also able to capture some rare predicates.

Figure 3(d) shows the comparison between the attention weights of the last layer and the mean of all layers. The attention weights of the last layer perform better. This is due to the attention weights in lower layers being low-level linguistic knowledge according to (Clark et al., 2019; Ramsauer et al., 2020), which are less relevant to the relational information. Figure 3(e) compares the impact of different attention reductions, i.e., mean, max, over the attention heads of the last layer. We find the “mean” performs better. The reason is that the token often intensively attends to several specific tokens in the sequence (Michel et al., 2019), and the “mean” operator is less sensitive to such biases.

### A.3 The Number of Predicates of Standard and Factual OIE Datasets

Note that there are more predicates in standard OIE datasets than that in factual datasets. This is because, for standard OIE, predicates are open and not attached to a certain schema. These predicates were extracted from the input sentences and are usually natural language utterances. For factual OIE, the predicates are unified into a fixed KG schema. For example, for a person’s birthplace, there are multiple natural language expressions like “was born in” or “gave birth” in standard OIE datasets, while only a single “birth_place” predicate exists in the factual OIE sets.

### A.4 Predicate Mapping Examples

We show example predicate mappings in a dictionary below.

- \(\text{per:city_of_birth}\): born in, born at, born, birth city, hometown.
- \(\text{org:founded_by}\): established by, founded by, founded, founder, co-founder of.

where the keys are KG predicates, e.g., \(\text{per:city_of_birth}\) and \(\text{org:founded_by}\). The values are the corresponding OIE relations.

### A.5 Comparison Systems

We compare our zero-shot OIE systems with the following OIE systems.

#### A.5.1 Neural OIE Systems

The following neural network based systems are selected.

- SenseOIE (Roy et al., 2019)\(^6\) learns to ensemble various previous unsupervised OIE systems’ ex-

---

\(^4\)https://github.com/huggingface/transformers
\(^5\)https://spacy.io/api/sentencizer
\(^6\)The code is not available.
tractions using supervised learning to combine their strengths.

• **SpanOIE** (Zhan and Zhao, 2020)\(^7\) presents the Re-OIE2016 datasets for a more rigorous evaluation and a span-based (instead of sequence labeling) extraction model.

• **RnnOIE** (Stanovsky et al., 2018)\(^8\) is one of the state-of-the-art OIE systems. It uses LSTM to model the OIE problem as a sequence tagging problem, and is trained on a large-scale OIE training set.

• **NeuralOIE** (Cui et al., 2018)\(^9\) is an encoder-decoder based architecture that adopts the copy mechanism to conduct OIE.

• **IMOJIE** (Kolluru et al., 2020b)\(^10\) is a sequence generation based OIE model that uses BERT at encoding time.

• **Multi\(^2\)-OIE** (Ro et al., 2020)\(^11\) models OIE as a sequence labeling problem that combines BERT with multi-head attention blocks.

• **OpenIE6** (Kolluru et al., 2020a)\(^12\) is one of the state-of-the-art OIE systems. It treats OIE as a 2-D grid labeling task, and trains a BERT family architecture for the task.

Note that while our methods are zero-shot without needing to use the specific training sets, all the neural OIE systems are supervised on corresponding training sets.

### A.5.2 Linguistic OIE Systems

We also compare our systems with the following linguistic pattern based systems developed prior to the use of neural networks.

• **MinIE** (Gashteovski et al., 2017)\(^13\) proposes to minimize facts in OIE by representing information by annotations rather than extraction and removing redundant specific information.

• **ClausIE** (Del Corro and Gemulla, 2013)\(^14\) is a clause-based approach by first identifying linguistic structure and then their information and attributes.

• **OLLIE** (Schmitz et al., 2012)\(^15\) uses contextual sentence decomposition to conduct OIE.

• **PropS** (Stanovsky et al., 2016)\(^16\) proposes proposition structure which is implied from syntax using dependency trees.

• **OpenIE4** (Christensen et al., 2011)\(^17\) is the successor to OLLIE using similar argument and relation expansion heuristics to create OIE extractions from semantic role labeling frames.

• **OpenIE5** (Saha et al., 2017, 2018)\(^18\) is one of the state-of-the-art OIE systems, which is the successor to OLLIE, and it improves extractions from noun relations, numerical sentences, and conjunctive sentences depending on the linguistic patterns.

• **Stanford OpenIE** (Angeli et al., 2015)\(^19\) leverages POS tag and dependency parser, and generates self-contained clauses from long sentences to extract the triples.

### B The TAC KBP-OIE and Wikidata-OIE Datasets

We show samples of our zero-shot OIE extractions and the gold triples on both TAC KBP-OIE and Wikidata-OIE datasets.

#### B.1 TAC KBP-OIE

**OIE Extractions and Gold Extractions**

We randomly sample 100 documents from the TAC KBP-OIE corpus, then sample sentences from those documents. The uncurated triples and the corresponding gold triples of the sampled sentences based on our best methods BERT\(_{\text{LARGE}}\) and GPT-2\(_{\text{XL}}\) are shown in Figure 4 and Figure 5 respectively. We also randomly sample sentences in which BERT\(_{\text{LARGE}}\) differs from GPT-2\(_{\text{XL}}\) in the resulting triples for comparison, which are illustrated in Figure 6. In each table, “ID” represents the document ID of a sampled sentence in the TAC KBP-OIE corpus. “Sentence” indicates the sampled sentence. “Triples to gold triples” column contains the extraction triples (on the left side of

---

\(^7\) https://github.com/zhanjunlang/Span_OIE

\(^8\) https://github.com/gabrielStanovsky/supervised-oie

\(^9\) We use the BERT implementation available at https://github.com/dair-iitd/imojie.

\(^10\) https://github.com/dair-iitd/imojie

\(^11\) https://github.com/Youngbin-Ro/Multi2OIE

\(^12\) https://github.com/dair-iitd/openie6

\(^13\) https://github.com/uma-pi1/minie

\(^14\) https://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/ambiverse-nlu/clause

\(^15\) https://github.com/knowitall/ollie

\(^16\) https://github.com/gabrielStanovsky/props

\(^17\) https://github.com/allenai/openie-standalone

\(^18\) https://github.com/dair-iitd/OpenIE-standalone

\(^19\) https://nlp.stanford.edu/software/openie.html
“→”) and their corresponding gold triples (on the right side of “→”).

B.2 Wikidata-OIE

OIE Extractions and Gold Extractions  Similar to TAC KBP-OIE, we randomly sample 100 documents from the Wikidata-OIE corpus (i.e., English Wikipedia), then sample sentences from those documents. Similar to TAC KBP-OIE, Figure 7 and Figure 8 show the uncurated triples and the corresponding gold triples of the sampled sentences based on our zero-shot systems BERT\textsubscript{LARGE} and GPT-2\textsubscript{XL} respectively. Figure 9 illustrates the randomly sampled sentences in which BERT\textsubscript{LARGE} extracts different triples compared to that from GPT-2\textsubscript{XL}. In each table, “ID” represents the Wikipedia page’s title of the sampled sentence. “Sentence” indicates the sampled sentence. “Triples to gold triples” column contains the triples (on the left side of “→”) and their corresponding gold triples (on the right side of “→”).
Bashardost left Pakistan for France in 1981.

With his wife, Cornelie, Middelhoff invested money in 2000 and 2001 with Esch in funds that were formed to ... leased back to the department store chain before Middelhoff joined the company, according to Middelhoff's spokesman.

"When you close a country, you end up causing more problems than you prevented," said Chuck Johnson, CEO of the National Council for Adoption.

The chairman of the Swiss Bankers Association, Patrick Odier, told weekly NZZ am Sonntag that Italy and ... deals like ones Switzerland signed this week with Germany and Britain.

"It's an issue for everybody in the state because peanuts are a big part of our economy," said Don Koehler, executive director of the Georgia Peanut Commission.

After several years spent largely at the Centre National de la Recherche Scientifique in Paris, Mandelbrot ... Institute of Technology, it was not until 1987 that he began to teach at Yale, where he earned tenure in 1999.

Charles Gwathmey, an architect known for his influential modernist home designs and famous clients like director Steven Spielberg, has died.

A little more than a year after Dunne died from bladder cancer, the colorful remnants of his estate have been ... by his family to Stair Galleries in Hudson, N. Y., which will auction them Nov. 20.

Hewitt was already a highly respected TV newsman.

Don Hewitt, the CBS newsman who invented the highly popular TV newsmagazine "60 Minutes" and produced it for 36 years, died Wednesday.

Koirala was born in 1925 in Bihar of India where his father Krishna Prasad Koirala and his family were living in exile.

Koirala began his political career as a union organiser and was imprisoned for seven years in 1960 after a failed uprising against the monarchy. Upon his release he went into exile in India, where he masterminded the 1973 hijacking of a Royal Nepal Airlines plane to Mediterranean island of Sardinia to fund his banned Nepali Congress party.

Blake Edwards, a writer and director who was hailed as a Hollywood master of screwball farces and rude comedies and was the first to make the "Breakfast at Tiffany's" and the "Pink Panther" movies, died Wednesday night in Santa Monica, Calif. He was 88.

In addition to his wife, Carol, Anderson is survived by his sons, Lee and Albert; his daughter, Shirlee Englebrecht; and many grandchildren.

Penner is survived by his brother, John, a copy editor at the Times, and his former wife, Times sportswriter Lisa Dillman.

Oscar-winning actress Patricia Neal has died of lung cancer at her home on Martha's Vineyard, Massachusetts, on Sunday.

Michigan native Nancy Kissel was convicted of murder and sentenced in Hong Kong's High Court in September 2005.

In addition to his wife, Wendy, Dio is survived by son Daniel, grandchildren Julie and Joey, and father Pat.
| Subject | Predicate | Object |
|---------|-----------|--------|
| when    | 0x001     | 0x002  |
| where   | 0x003     | 0x004  |
| which   | 0x005     | 0x006  |
| who     | 0x007     | 0x008  |
| what    | 0x009     | 0x010  |
| why     | 0x011     | 0x012  |
| how     | 0x013     | 0x014  |
| because | 0x015     | 0x016  |

Figure 5: GPT-2XL on TAC KBP-OIE.
Figure 7: BERT\textsc{LARGE} on Wikidata-OIE.
