Proceedings of the 13th Conference on Language Resources and Evaluation (LREC 2022), pages 1408–1417
Marseille, 20-25 June 2022

Probing Pre-trained Auto-regressive Language Models for Named Entity Typing and Recognition

Elena V. Epure, Romain Hennequin
Deezer Research
22-26 rue de Calais, 75009 Paris, France
research@deezer.com

Abstract

Multiple works have proposed to probe language models (LMs) for generalization in named entity (NE) typing (NET) and recognition (NER). However, little has been done in this direction for auto-regressive models despite their popularity and potential to express a wide variety of NLP tasks in the same unified format. We propose a new methodology to probe auto-regressive LMs for NET and NER generalization, which draws inspiration from human linguistic behavior, by resorting to meta-learning. We study NEs of various types individually by designing a zero-shot transfer strategy for NET. Then, we probe the model for NER by providing a few examples at inference. We introduce a novel procedure to assess the model's memorization of NEs and report the memorization's impact on the results. Our findings show that: 1) GPT2, a common pre-trained auto-regressive LM, without any fine-tuning for NET or NER, performs the tasks fairly well; 2) name irregularity when common for a NE type could be an effective exploitable cue; 3) the model seems to rely more on NE than contextual cues in few-shot NER; 4) NEs with words absent during LM pre-training are very challenging for both NET and NER.

Keywords: auto-regressive language model, probing, zero-shot NET, few-shot NER, memorization testing

1. Introduction

Before transformer LMs (Devlin et al., 2019), the state-of-the-art NER was based on training recurrent neural networks, such as bidirectional LSTM (BiLSTM) with a Conditional Random Field (CRF) layer, from scratch (Yadav and Bethard, 2018). The widely adopted approach with transformers has been to fine-tune them for the desired task, thus specializing their general linguistic knowledge, acquired during pre-training. While LMs fine-tuned for NER have achieved impressive results on standard benchmarks (Akbik et al., 2019), multiple works have emphasized their limitations with regard to their generalization capacity to new textual genres (e.g. clean versus noisy text), NE type sets (e.g. NE types belonging to new domains such as music or e-commerce) and new NEs, unseen during training (Lin et al., 2020b).

To gain more insights into the NER generalization ability of LMs, multiple studies have been conducted. Probing has been designed for BiLSTM-CRF LMs (Augenstein et al., 2017; Taillé et al., 2020; Fu et al., 2020) or masked LMs such as BERT (Petroni et al., 2019; Jiang et al., 2020). Yet, little has been done for auto-regressive models such as GPT2 despite their popularity and potential to express a wide variety of NLP tasks in the same unified format (Raffel et al., 2020).

Additionally, although the past probing studies have broadened the knowledge about how LMs generalize in the NER context, multiple improvements could be brought to existing methodologies. First, the impact of pre-training LMs on the results has never been assessed. Second, the proposed setups test generalization by relying on large annotated datasets which are manipulated in different ways to create test and train splits. However, when assessing generalization in relation to *human linguistic behavior* (Levesque, 2014), which we claim as more realistic, these datasets are insufficient and different testing conditions should exist. Humans can easily recognize NEs based on prior domain and common sense linguistic knowledge, or by leveraging contextual cues in text (Lin et al., 2020a). Humans can perform new linguistic tasks quite well even when exposed to a few examples or very simple instructions (Brown et al., 2020). When it comes to technology creation, human linguistic behavior could lead to infinite examples, many of them new to everyone, including to the systems' designers (Webber et al., 2020).

Hence, datasets used in past studies cannot capture this variability for a realistic testing unless continuously updated. However, like humans, LMs have gained diverse domain and linguistic knowledge, and developed general pattern recognition abilities from experience, during pre-training (Brown et al., 2020). Given these, the research question we investigate is:

Can the knowledge gained during pre-training be leveraged by auto-regressive LMs at inference to adapt to diverse NE-related tasks, when queried with a few examples at most and simple natural language instructions?

Contributions. Inspired by testing conditions related to human linguistic behavior, we design a probing methodology centered on meta- or "in-context" learning. It entails the task specification via the text input used to prompt the model, without performing any gradient updates (Brown et al., 2020). First we study NEs of various types individually by defining a zero-shot transfer strategy for NET. We design *a novel method to assess NE memorization* by the model and report the memorization's impact on the results. Our memorization method could be used to sample (un)popular NEs (Shwartz et al., 2020), but also, beyond the NE context, with other types of n-grams. Second, we model NER as a machine reading comprehension (MRC) task and probe the model by providing a few examples at inference (e.g., the model should extract spans of text from input, as answers to simple queries). We also test NER with (un)memorized

NEs and gain insights on the role of context, i.e. text around NEs. We use four datasets: CoNLL-2003 (Tjong Kim Sang and De Meulder, 2003), WNUT2017 (Liu, 2014), MIT Movie (Liu, 2014) and extensive lists of NEs from DBpedia (Auer et al., 2007). These datasets contain clean and noisy text, and regular NEs such as people names and irregular NEs such as creative work titles.

Our study[1] joins other efforts that looked into NET and NER generalization but that, compared to us, achieved this by manipulating datasets during fine-tuning / testing or targeted other types of LMs. To our knowledge, we are the first to extensively probe *pre-trained auto-regressive LMs as they are* for these tasks and ensure testing conditions related to human linguistic behavior.

Findings. Pre-trained GPT2, a common auto-regressive LM, appears to perform the tasks fairly well without any fine-tuning for NET or NER, especially on regular NEs or memorized during pre-training. These models, as they are, already know quite a lot about NEs and encode NER patterns. Our finding is particularly important given that past works study NET and NER generalization of existing LMs without explicitly considering the impact of model pre-training. Then, compared to other studies that claim named entity irregularity to be problematic (Augenstein et al., 2017), we show that when frequently present for a certain NE type it can become, in fact, an effective exploitable cue. We also show that the model seems to rely more on NE cues than on context cues in few-shot NER, and that the model's exposure to the NE's words weighs much more than the exposure to the exact NE in zero-shot NET.

2. Background and Related Work

2.1. NE Generalization in Current Models

The common way to perform NER nowadays relies on training or fine-tuning a deep neural network using a relatively large annotated dataset and often aims at extracting a few regular NE types such as person, location and organisation (Yadav and Bethard, 2018; Akbik et al., 2019; Lison et al., 2020). Although recent LMs have yielded impressive results, NER generalization to all types of textual genres is still an issue, in particular, in informal text, frequently found on social media or in chat-bot interactions. This type of text can often lack proper formatting, e.g. word capitalization, and contain unusual grammatical structures or jargon (Aguilar et al., 2018; Guerini et al., 2018).

Another challenge is NER generalization to diverse and growing NE type sets, belonging to new domains such as movies, music or e-commerce (Ma et al., 2016; Guerini et al., 2018; Lin et al., 2020b). These types are often more heterogeneous (e.g. *groups* in WNUT includes sport teams and music bands (Aguilar et al., 2018)); lack name regularity (e.g. *creative work* titles are not necessarily noun phrases (Lin et al., 2020b)); can be composed of common words or of words which are typically from other languages (e.g. the film "Demolition Man" (Derczynski et al., 2017)). Then, NER generalization to new NEs, unseen during training is another challenge. This is common in the real-world where a system learns from a limited number of examples per type while NE mentions are expected to shift in time (Augenstein et al., 2017).

These challenges have been addressed by relying on new training datasets with each new case. However, collecting thousands of human annotations for new genres, NE types or NE mentions is expensive and time-consuming (Augenstein et al., 2017; Lin et al., 2020a). Other recent works rely on existing NE resources, such as gazetteers and dictionaries, to either perform NER in a distant or weak supervision setup (Lison et al., 2020; Shang et al., 2018), or to train NET classifiers adaptable to unseen NEs (Guerini et al., 2018). Constraining the model to rely more on context than on NEs have also appeared promising to achieve generalization (Mengge et al., 2020; Lin et al., 2020a).

Other efforts towards NET and NER generalization share the same rationale as us—the challenges in collecting annotations with each new case and the human linguistic behavior, and design zero- or few-shot learners to perform the tasks (Zhou et al., 2018; Zhang et al., 2020; Yang and Katiyar, 2020; Ding et al., 2021; Aly et al., 2021). There are several major differences with our study. First, our goal is not to propose a new NET or NER model, but to probe pre-trained LMs as meta-learners without modifying their weights or leveraging external knowledge. Second, assuming that our probing methodology is exploited as a basic NET and NER model, its input is much more constricted (lists of NEs and NE types in NET, a few NEs in context for NER) compared to the other works which make use of a wide range of resources such as: knowledge bases, definitions of entity types in a taxonomic and/or natural language forms, NEs in context even for NET, or datasets of seen NE types with many examples.

2.2. Probing Studies for NER Generalization

Lin et al. (2020b) propose an extensive use of randomization tests to study the extent to which a fine-tuned LM relies on: name regularity—regular (e.g. persons) versus irregular names (e.g. creative works); on mention coverage—the ratio of overlapping NEs in train and test data; and on context diversity—unique sentences for each NE type. Fu et al. (2020) investigate the popular NER architecture, LSTM-CRF, from various views including NE and contextual coverage. Also, they study the impact of the relations among NE types on model learning. A BiLSTM-CRF is also studied in (Taillé et al., 2020), but the focus is on bench-marking different contextualized or static embeddings for generalization to new NE mentions and domains.

Compared to these, we focus on pre-trained auto-regressive LMs as-is and not on fine-tuning / training them and, implicitly, on the impact of train / test datasets. To our knowledge, this is the first detailed study designed for pre-trained LMs as NET and NER meta-learners without modifying them or using external resources. We study many generalization angles: seen versus unseen NEs–referred also as *memorized versus unmemorized NEs* in the paper, regular and irregular NEs, diverse genres including noisy text, and reliance on context versus NE cues.

[1] Code is available at https://github.com/deezer/net-ner-probing.

3. Proposed Probing Methodology

We divide our study in NET in a zero-shot transfer followed by NER in a few-shot settings. This allows us to acquire knowledge first about names and then about NEs in context.

3.1. NET in a Zero-shot Setup

Auto-regressive LMs estimate the empirical distribution from the training data, where each training example x is a sequence of tokens $x = (s_1, s_2, ..., s_n)$. Given the sequential nature of the language, it is common to factorize the distribution $p(x)$ with the Bayes' rule and express it as a product of conditional probabilities of each sequence's token s_i given the previous tokens:

$$p(x) = \prod_{i=1}^{n} p(s_i|s_1, ..., s_{i-1}) \tag{1}$$

On a new task, the model infers $p(output|input)$ or more completely written $p(output|input, task)$. Brown et al. (2020) merge the $input$ and $task$ in a single natural language query and express $output$ as the predicted next sequence of tokens. For instance, we could write a query for NET as "Sentence: is Italy a person, location or organisation? Answer:". The predicted NE type is then the token among "person", "location", or "organisation" which the model estimates as most likely to follow.

Alternatively, we could frame NET as the most likely statement among multiple competing ones such as "Anne is a person", "Anne is a location". In this case, the sequence with the lowest perplexity is the one that the model is less surprised to see, hence describing the most likely NE type. The perplexity of a sequence x, using a model θ, is:

$$\text{PPL}_\theta(x) = \exp\{-\frac{1}{n}\sum_{i=1}^{n} \log p_\theta(s_i|s_{<i})\} \tag{2}$$

We adopt this latter approach as it provides a simple task framing in zero-shot settings and perplexity can be efficiently computed by relying on a single model call. Thus, given a NE mention e and a NE type set T, the most likely type $t_e \in T$ for e is:

$$t_e = \arg\min_{t \in T} \text{PPL}_\theta(\text{query}(e, t)) \tag{3}$$

where query(e, t) is the template "e is a t" (e.g. "Cinderella is a city" or "Cinderella is a character").

Assessing generalization. The model's generalization to NEs unseen or rare during training is essential for a human-centered setup. Thus, assessing how the model performs on (un)memorized NEs could provide a more realistic understanding of its performance. Previous works that led such investigation, trained models from scratch (Lin et al., 2020b; Taillé et al., 2020), so could keep track of (un)seen NEs during training. As we focus on pre-trained LMs and have no access to their training data, we devise a method to assess if the model has memorized a NE or not.

Carlini et al. (2019) propose a test for unintended memorization of rare sequences based on perplexity. Given all possible sequences for a matter at hand (or a very large sample, S) prefixed by the same query (e.g. prefix "the random number is " and $S = \{281265011, 281265017...\}$), rank them by PPL_θ and use ranks to compute exposure:

$$\text{exposure}_\theta(x) = \log_2 |\mathcal{S}| - \log_2 \text{rank}_\theta(x) \tag{4}$$

For $x \in \mathcal{S}$, the exposure metric is negatively correlated with the rank, i.e. the lower the rank the higher the exposure, thus likely memorization.

This test is a helpful point of departure, but less applicable to our task as-is. Without a very large set of NEs for each type, the estimates could be inaccurate, especially when only few sequences have lower perplexity than a target one (Carlini et al., 2019). Also, we noticed experimentally (see Figure 1) that the mean perplexity tended to decrease with the number of tokens per NE[2], a phenomenon most likely related to the open-vocabulary language modeling over sub-word units[3]. Thus, with the method of Carlini et al. (2019), NEs would have a higher chance to be flagged as memorized when they are tokenized in more tokens.

As originally stated, our goal is to evaluate the model's behavior with (un)memorized NEs. Thus, we want to be able to assign NEs to two groups when we are confident of their (non-)memorization, while ignoring NEs in the gray area. We changed the previously shown test to rely directly on probabilities of NE's tokens, obtained when calling the model with NEs as input, prefixed by a fixed string. The test we propose is further summarized:

If NE *words* are known (e.g. "Great" and "Britain" are in the model's vocabulary) and their sequential *transitions* are unsurprising (e.g. p(Britain|Great) is large), then the model has likely seen the NE during training. Formally, we define two exposure metrics for these two aspects as follows:

$$\begin{aligned} \text{exposure}^{word}_\theta(x) &= \prod_{(i,j) \in W_x} \text{test}^{word}_\theta(x, i, j) \\ \text{test}^{word}_\theta(x, i, j) &= \begin{cases} 1 & \text{if } i = j \\ p_\theta(s_j|s_{<j}) & \text{if } i < j \end{cases} \end{aligned} \tag{5}$$

$$\text{exposure}^{trans}_\theta(x) = \min_{i \in T_x} p_\theta(s_i|s_{<i}) \tag{6}$$

where W_x consists of tuples marking the start and end indices of each word in x (a word can have multiple tokens) and T_x has indices marking the transitions (the index of each new word). In Equation 5, we identify two cases when a word can be considered known by the LM. It is directly mapped on a token in the LM's vocabulary V or, when it is split in multiple tokens, the last token becomes an indicator of its memorization. In Equation 6, to test whether the sequential association of words is unsurprising to the model, we take the minimum probability of the tokens marking the start of each new word.

[2]The trend was similar per number of words or characters.

[3]Among the NEs with large number of tokens and lower perplexity, we often noticed NEs from other languages than English and with more words (e.g. *L'Hospitalet-près-l'Andorre*). Given the previous ones, the probability of many of these sub-tokens is quite high (e.g. p(s|L'Hospitalet-prè) $= 0.993$), which results in low perplexity. This could be an effect of the model memorizing some rare or foreign words, but not necessarily the NE.

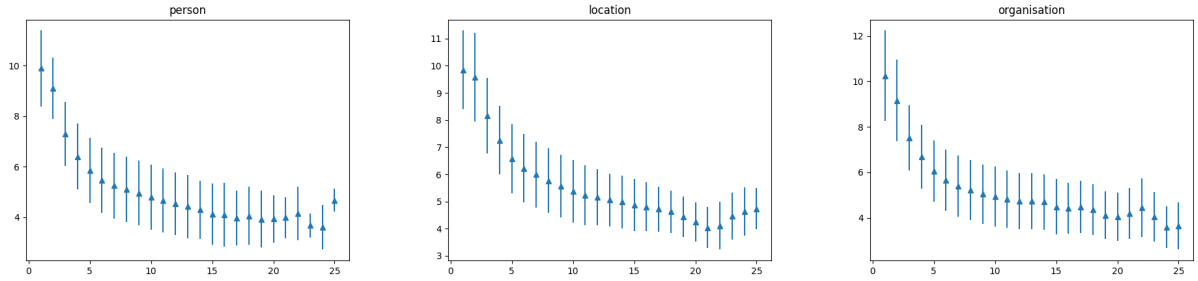

Figure 1: Mean and standard deviation of log perplexity computed with GPT2 for large lists of persons (left), locations (middle) and organisations (right) from DBpedia. Values are grouped by the number of tokens per NE.

For the final decision, NEs with exposure values higher or lower than some established thresholds could be assigned to the memorized / unmemorized NE groups. These thresholds could be defined considering the NE set and the model's vocabulary size (more details in Section 4). An advantage of our method over (Carlini et al., 2019) is that we do not need access to a very large set for each NE type, the token probabilities being sufficient to establish the NE exposure / degree of memorization.

3.2. NER in a Few-shot Setup

We frame NER as a MRC task (Mengge et al., 2020; Li et al., 2020), but, instead of fine-tuning / training a pre-trained LM for MRC, we exploit it in a few-shot setup. As detailed in Section 1, the few-shot setup has been considered closer to human linguistic behavior and has shown competitive results in other NLP tasks such as question answering, translation, and classification (Brown et al., 2020). Zhao et al. (2021) have also tested information extraction for slot-filling with some slots targeting NEs (e.g. the director of a movie). Yet, they assume that each sentence contains that type of slot, without assessing the case when no NEs exist in the sentence.

Similar to past works, we use a query to formulate the task and insert examples, which are used only at inference, without triggering updates of the pre-trained model weights. We show in Figure 2 a query generated from WNUT2017 dataset for the entity type *product*. The query has two parts: a prefix and a test sentence. The *prefix* is appended to each sentence and introduces the examples (0-7). We provide 4 examples, two with NEs and two without. We use the token "none" to mark the absence of a NE of the target type. The second part with the last two lines (8-9) introduces the *test sentence*, for which NER is performed.

Previous works have shown that the query choice has a significant impact on the task's accuracy (Li et al., 2020). In the few-shot learning case, the set of examples and their order can lead to different results too (Zhao et al., 2021). For instance, the model might tend to predict the majority token or the one nearest to the end of the query. To overcome this, Zhao et al. (2021) propose a procedure to calibrate the LM's output probabilities by taking into account the LM's bias towards certain outputs. Specifically, in the generation task, an affine tranformation and softmax is applied to $\hat{\mathbf{p}}$, the set of probabilities of the first token: $\hat{\mathbf{p}}_{cal} = \text{softmax}(\mathbf{W}\hat{\mathbf{p}})$. $\mathbf{W}$ is estimated from $\hat{\mathbf{p}}_{cf}$, the probabilities obtained when feeding in the model a "content-free" input such as "N/A", as $\mathbf{W} = \text{diag}(\hat{\mathbf{p}}_{cf})^{-1}$. We use the same calibration procedure and run each experiment multiple times, with varied examples as demonstration. As for the query format, we stick to the one shown in Figure 2 and leave for future work the exploration of other formats.

0:	Sentence:	I don't like to be stuck at home
1:	product:	none
2:	Sentence:	Where is Gelato Gilberto?
3:	product:	none
4:	Sentence:	Well, I was gonna buy a Zune HD
5:	product:	Zune HD
6:	Sentence:	BEAUTY TIPS: SK-II UV Cream
7:	product:	SK-II UV Cream
8:	Sentence:	CVS sells their own epipen
9:	product:	
	True Answer:	epipen

Figure 2: NER query and the expected generated answer. Lines 0-7 are examples (two negatives, two positives) and lines 8-9 are the test sentence.

Assessing generalization. Assessing if the LM can generalize to unmemorized NEs is a more humanlike setup for evaluation. We could select two sets of sentences with our memorization test, and report performance on each separately. However, splitting smaller datasets as WNUT2017 would not allow for reliable conclusions, nor to analyse the role of the context (i.e. the other parts of the sentence) because sentences would be different in the dataset splits. As Lin et al. (2020b) highlighted, we should aim at NER models that rely more on context for generalization, rather than memorizing NEs, in particular for irregular NE types such as creative work titles. Thus, it is relevant to allow for the study of context.

For these reasons, the experiment design we propose is to fix the examples in the query prefix (0-7 in Figure 2), and compute performance on three variations of the test sentence (8-9 in Figure 2): *test as-is* (the original sentence), *test seen* and *test unseen*. To obtain *test seen*, we randomly sample NEs to replace the existing ones in the original sentence from a list of memorized NEs identified with our memorization test. For *test unseen*, we replace NEs by choosing among random lowercase strings, which do not exist in the English language, thus are not memorized. The hypotheses are: 1) if the model relies more on context

Dataset	Type	NE types	NET	NER
CoNLL-2003	clean	person, location, organisation	✓	✓
MIT Movie	noisy	person, creative work	✓	✓
WNUT2017	noisy	person, location, corporation, group, product, creative work	✓	✓
DBpedia	clean	person, location, organisation, creative work	✓	

Table 1: Overview of the datasets used in each task. For NET, we consider NE mentions from all dataset (train, test, and dev if available). NER is evaluated on test sets while the train sets are used only to sample examples for the query.

for NER then the performance on *test as-is* and *test unseen* should be similar; 2) if the model relies more on NE cues then the performance on *test seen* should be much larger than on *test as-is*.

4. Experiments

We apply the proposed methodology to a medium-sized GPT2. A larger model like GPT3 yielded better results as a meta-learner in past experiments (Zhao et al., 2021). However, we have decided to use the proposed probing methodology with a model that was easily accessible and had lower memory requirements, leaving the extension to other autoregressive models as future work.

Datasets. CoNLL-2003 (Tjong Kim Sang and De Meulder, 2003) and WNUT2017 (Derczynski et al., 2017), commonly found in NER benchmarks, are kept as they are. The MIT Movie dataset (Liu, 2014), originally created for slot filling, is modified by ignoring some slot types (e.g. *genre*, *rating*) and merging others (e.g. *director* and *actor* in *person*, and *song* and *movie title* in *title*) in order to keep consistent NE types across all datasets. MIT movie dataset contains only lowercase text, sometimes with typos, thus falling under the noisy text genre as WNUT2017. For NET, we consider the NE mentions from each dataset in its entirety (train, test, and dev if available). We also collected large lists of different NE types from DBpedia (Auer et al., 2007). These are particularly interesting because Wikipedia has not been included in the GPT2 training corpora (Radford et al., 2019). The NER experiments are run only on test sets while the train sets are used for sampling examples for the query. A summary of the datasets and the tasks in which they are used is presented in Table 1.

Probing NET. For NET, we create prompts starting from NE types and choose as predicted value the type which leads to the lowest perplexity as presented in Section 3.1. In practice, we use multiple keywords for each NE type starting from their definition. We also include *character* for *person*; *company*, *group*, *institution*, *club*, and *corporation* for *organization*; *place*, *city*, and *country* for *location*. As the perplexity decreases with the number of tokens as shown in Figure 1, we choose all keywords such that they are part of the model vocabulary. Thus, *creative work* is replaced by *work*, *title*, *movie*, *song*, and *book*. We do not include other keywords for *product*, *corporation* and *group* in WNUT2017.

For the exposure computation, we prefix NEs with the default unknown token when retrieving probabilities. The thresholds for word and transition exposures are established per dataset. For the lower limit, we consider the size of the GPT2 vocabulary ($\approx 50K$); thus, assuming a uniform word distribution[4], each token would have a $2e$-05 probability to be generated next. CoNLL-2003 has many one-word NEs with rare transitions. For this reason, we focus only on exposure$_\theta^{word}$ to establish if a NE is memorized ($\geq .8$) or not ($\leq 1e$-04). The rest of NEs are not classified. In contrast, in MIT Movies, NEs are often composed of multiple words common in English-language, thus present in the model's vocabulary. In this case, exposure$_\theta^{tran}$ is more informative for selecting memorized NEs ($\geq .001$) and unmemorized NEs ($\leq 1e$-05).

We sample the two groups from the DBpedia lists using either exposure$_\theta^{word}$ or exposure$_\theta^{tran}$. In this way, we investigate the impact of knowing words vs. recognizing word transitions on a much larger sample. We ignore one-word NEs from MIT Movies and DBpedia because they are rare or often spurious. Finally, we only run the NET experiment on the complete WNUT2017 dataset because the number of NEs for each entity type is too small to allow reliable memorized vs. unmemorized split.

Probing NER. We opt for a maximum of training examples in the query that can be kept in memory, in our case 16. Out of these, 9 contain NEs of the targeted type and 7 are randomly chosen from the rest of the dataset. We run each experiment three times with different random seeds to compute variance. The test set is slightly modified too: for each NE type, we keep all positive sentences and sample negative sentences such that the ratio positive-negative is about 2:1. The maximum number of tokens asked when querying the model is set to 15. The calibration we apply follows the steps described in (Zhao et al., 2021).

We design the NER meta-learner to extract one NE of the prompted type at a time, leaving the case of multiple NEs per text as future work. Because a test sentence can mention multiple NEs of the same type, we consider a generated answer to be correct if it matches one of the existing NEs. In computing scores, we rely mostly on exact NE matching with some exceptions. The evaluation is insensitive to the letter case (e.g. 'none' and 'None' are considered equivalent). Also, we noticed that the model tends to add spaces for NEs written together such as in social media mentions. To cover these cases, we consider that the prediction is equal to the ground-truth, if their Levenshtein distance divided by the true NE length is lower than 0.2. When no NEs should be extracted but the model generates another string that does not have any words in common with the input, we consider it a correct prediction even if it's not explicitly "none"[5].

[4]This assumption is strong, but used only to establish an order of magnitude for the unmemorized exposure$_\theta^{trans}$.

[5]The model can generate strings such as *null* or "."

Dataset	NE Type	All			Memorized		Unmemorized	
		F1	F1 (ZOE)	Count	F1	Count	F1	Count
CoNLL-2003	person	0.90	0.90	3613	0.93	695	0.86	619
	location	0.66	0.80	1331	0.74	546	0.37	80
	organisation	0.70	0.74	2401	0.74	770	0.63	289
	macro-average	0.75	0.81	7345	0.81	2011	0.62	988
MIT Movie	person	0.80	-	2866	0.82	605	0.81	369
	creative work	0.60	-	2122	0.64	402	0.58	256
	macro-average	0.70	-	4988	0.73	1007	0.69	625

Table 2: Zero-shot NET F1-scores. Results obtained with ZOE on CoNLL-2003 are also shown.

Metric	NE Type	M	UM	Count
$\text{exposure}_\theta^{word}$	person	0.88	0.64	10000
	location	0.81	0.63	10000
	organisation	0.76	0.67	10000
	creative work	0.69	0.36	10000
	macro-average	0.78	0.58	40000
$\text{exposure}_\theta^{tran}$	person	0.83	0.78	10000
	location	0.83	0.75	10000
	organisation	0.78	0.71	7014
	creative work	0.63	0.55	6012
	macro-average	0.77	0.70	33026

Table 3: Zero-shot NET results on DBpedia NEs. M stands for Memorized and UM for Unmemorized. The **Metric** column reports the exposure metric used to select memorized and unmemorized NE lists.

Validation. As previously mentioned, out goal is not to propose a new NET or NER model, but to probe pre-trained LMs as meta-learners without modifying their weights or leveraging external knowledge. However, as a sanity check to understand if the model used in this way actually works, we position the model's performance with respect to the performance of other baselines.

For NET on CoNLL-2003, we include a zero-shot baseline, ZOE (Zhou et al., 2018), which derives NE types by having as input the NE mention in a sentence and a taxonomy of NE types with the corresponding definition for each type. ZOE is designed for clean text and uses Wikipedia as the taxonomy of NE types. For these reasons, we cannot apply it to the other noisy datasets or to DBpedia, which is extracted from Wikipedia. Nonetheless, we also report the performance of a weighted random guess NET classifier for WNUT2017.

For NER, we report the results of the best supervised baseline of the shared WNUT2017 task, UH-RiTUAL (Aguilar et al., 2017). UH-RiTUAL relies on a neural network to extract feature representations that are further fed in a CRF classifier. The feature extractor model is trained in a multi-task fashion on two objectives, NE segmentation and NE classification, and leverages as input character embeddings, Part-of-Speech tag embeddings, word embeddings and gazetteers.

A recent comprehensive study on few-shot NER (Huang et al., 2021) benchmarks multiple methods entailing training (prototype-based, self-training). We also report their score range for each dataset (CoNLL-2003, MIT Movie and WNUT2017) and compare them to our results.

NE Type	F1	F1 (random guess)	Count
person	0.79	0.40	1317
location	0.63	0.19	616
corporation	0.16	0.07	231
group	0.44	0.13	412
product	0.46	0.11	353
creative work	0.46	0.11	361
macro-average	0.49	0.17	3290

Table 4: Zero-shot NET F1-scores on WNUT2017.

5. Results and Discussion

5.1. NET in a Zero-shot Setup

Tables 2 and 3 show that GPT2 without relying on any NE context or other resources can perform NET quite well on most NE lists. On CoNLL-2003, the results are even close to those obtained by ZOE (Zhou et al., 2018), a much more complex system. Higher scores are obtained for regular types such as *person* or clean NEs (e.g. DBpedia NEs). We see lower scores for *creative work* in MIT Movies and *location* in CoNLL-2003, these being often confused with *person* (in 53% of the cases) and *organisation* (in 29% of the cases) respectively. The first confusion is not surprising given that movie titles could contain character names while *character* is included in *person*. The second confusion, *location-organisation*, is already mentioned as a common issue (Derczynski et al., 2017). Thus, most likely including context would help to disambiguate such NEs that belong to multiple types.

The NET performance on memorized NEs is, as expected, larger than on unmemorized NEs. However, Table 2 shows a much smaller drop on MIT Movie than on CoNLL-2003. The difference between these NE lists lies in the criterion we used in the memorization test, either focused on knowing NE's individual words in CoNLL-2003 or transitions between words in MIT Movie. This suggests that for a better NET performance, *the model's exposure to the NE's words weighs much more than the exposure to the exact NE, i.e. its word transitions*. In other words, even if a specific NE was not seen during pre-training, but its composing words were present as part of other NEs, then the model could still leverage this exposure in order to correctly classify the unseen NE in the proposed setup. This is further confirmed on the larger DBpedia NE lists (Table 3).

Table 4 shows that the model in a zero-shot setup yields significantly higher results than the random baseline on WNUT2017 NEs; though, overall lower for this textual genre, except for *person* and *location*. The Twitter-style

NE type	Test as-is	UH-RiTUAL	Test seen	Test unseen	Count
person	0.68±0.04	0.68	0.81±0.02	0.63±0.15	490
location	0.67±0.01	0.71	0.81±0.04	0.61±0.07	187
corporation	0.66±0.03	0.36	0.82±0.03	0.51±0.07	93
group	0.57±0.04	0.33	0.68±0.04	0.65±0.03	180
product	0.45±0.12	0.20	0.53±0.07	0.44±0.16	144
creative-work	0.63±0.03	0.16	0.75±0.02	0.55±0.02	184

Table 5: 16-shot NER F1-scores and standard deviations on the WNUT2017 dataset. The third column shows the results obtained with the baseline UH-RiTUAL.

Dataset	NE Type	F1	Count
CoNLL-2003	person	0.74±0.09	1537
	location	0.79±0.01	1899
	organisation	0.73±0.01	1843
MIT Movie	person	0.80±0.04	1908
	creative work	0.43±0.08	906

Table 6: 16-shot NER F1-scores and the standard deviations on CoNLL-2003 and MIT Movie datasets.

NEs may contain many words unseen by the model during pre-training. Also, we noticed similar confusion patterns as before: *corporation* or *group* (associated with *organisation*) with *location*, and *creative work* with *person*. Thus, context seems again promising with both the typing of NEs with new / rare words and the disambiguation for related NE types.

5.2. NER in a Few-shot Setup

As presented in Tables 5 and 6, the pre-trained LM, without any further fine-tuning or training can perform NER surprisingly well in the designed few-shot settings, even on noisy data. On WNUT2017, the noisiest dataset, we can see that the model outperforms the supervised baseline for all NE types apart from *location*. Similar to the baseline, *location* and *person* are among the easiest to extract NE types, while *product* is quite hard. In contrast, *corporation* and *creative work* types are recognized rather well. A qualitative analysis of the predicted NEs for CoNLL-2003 shows that the model has more challenges with false positives. This suggests that more negative examples may be needed at inference. For MIT Movie, the model often predicts "none" for *creative work*, an issue that might be overcome with better chosen positive examples.

Test (un)seen in Table 5 shows F1-scores when all NEs are replaced by random strings (lists available in Appendix), while fixing the context and the query examples. The scores for **Test as-is** are lower than for **Test seen** and larger than for **Test unseen**. Also, the score differences between **Test seen** and **Test as-is** are larger than the ones between **Test unseen** and **Test as-is**. These results lead to the rejection of hypothesis 1 and confirmation of hypothesis 2 introduced at the end of Section 3.2 and show that *the model appears to prioritize NEs cues more than context cues in few-shot settings*. Thus, when choosing query examples, one may favour to focus more on providing diverse NE patterns for an entity type than diverse context patterns.

As for NET, the impact of *NE (un)memorization during pre-training* is significant, with some exceptions such as *product*, for which F1-scores on **Test as-is** are almost the same to F1-scores on **Test unseen**. We checked the NEs sampled for the query and noticed frequent irregular names for *product*. Previously, Lin et al. (2020b) showed that name regularity is critical for generalization to new NEs. However, while the placeholders we used for unmemorized NEs were highly irregular, *this irregularity when often present for a certain NE type appears an exploitable cue*.

The best F1-scores reported by Huang et al. (2021) are in the range 0.65-0.90 on the original CoNLL-2003, 0.56-0.67 on the MIT Movie with the original NE types and 0.38-0.51 on the original WNUT2017, depending if 5 examples or 10% of the data are used as shots. Our average F1-scores with 16 shots appear competitive on our subsampled test sets: 0.75 for CoNLL-2003, 0.62 for MIT Movie and 0.61 for WNUT2017, which appears to validate the effectiveness of the pre-trained GPT2 probed for NER under the proposed conditions.

6. Conclusion

We proposed a NET and NER probing methodology designed for pre-trained auto-regressive LMs in zero- or few-shot settings. Our goal was to investigate if such a model, without any fine-tuning, could handle the tasks well while generalising to noisy text, diverse NE types, and new NEs. For this, we also defined a novel procedure to assess the exposure of the model to various NEs to create sets of (un)memorized NEs. Overall, we deemed our setup under more realistic conditions inspired by human linguistic behavior.

The results showed that a medium-size GPT2 in the proposed settings was quite good at NET and NER and we revealed multiple new insights, impactful for future work.

With pre-trained encoders, the exposure of the LM to NEs should not be investigated in fine-tuning only while neglecting the memorization during pre-training. We have proposed an effective method to support future studies with this. Also, a LM already pre-trained on a general task and a large corpus could effectively bootstrap NER for new applications, especially when NEs are common constructs in a language. Frequent name irregularity for a type in context can become a regularity effectively exploited by the LM in a few-shot NER. Context is important but with limited impact in our studied setup. Finally, choosing good query examples of NE patterns in context for few-shot NER and extending the study to other auto-regressive or masked LMs are still matters of investigation.

Appendix: Additional experiment details

In experiments, we used an NVIDIA GTX 1080 with 11GB RAM. We show the running time for each experiment in

Table 7.

Dataset	NE type	Time(s)	
		NET	NER
CoNLL-2003	person	210	6371
	location	85	6906
	organisation	144	6202
MIT Movie	person	131	4850
	creative work	100	2713
WNUT2017	person	86	2759
	location	49	1081
	corporation	29	730
	group	40	1372
	product	38	1127
	creative work	39	1502

Table 7: Running time in seconds for each experiment.

Lists of memorized and unmemorized NEs used to create the *test seen* and *test unseen* datasets for assessing NER generalization are presented further.
Memorized:

- *person*: Mary, Steve, Davis, Sam, Robert, Alex, Michelle, James, Danny, Rose, Edward, Rob, Harry, Tom, Paul
- *location*: Florida, Toronto, Syria, India, Houston, America, France, Australia, Turkey, NEW YORK, Chicago, Germany, Scotland, Washington, Ukraine
- *corporation*: Reuters, CNN, NBA, Uber, YouTube, CBC, Netflix, Microsoft, Twitter, Facebook, Apple, MAC, Tesla, Disney, Reddit
- *group*: Army, Chicago Blackhawks, Real Madrid, CIA, Senate, ART, NBA, The Black Keys, Crystal Palace, European Union, green day, Labor, Chelsea, the warriors, Democrats
- *product*: Air Music Jump, Android, Linux OS, iOS, Windows 7, Tesla, Google Music, SQL, Amazon Prime, Nintendo plus, google pixel, iPhone, Xbox 360, Legendary Skin, Bio Spot
- *creative work*: Black Swan, Iron Man 2, Finding Bigfoot, Good Morning Britain, Teen Titans, Pac- Man, Game of Thrones, La La Land, Last Christmas, Star Wars, Doctor Who, the Twilight Zone, Pokémon, Star Trek, Minecraft

Unmemorized:

- all: xgwqicng, kiooaiql, wpvqymid, rrmihdcg, owblmgbx, tiybjelq, ytlbllnh, ybwifxxv, svlsskxx, jdtqyoov, tzrtffbu, jvwywjhy, hzhwhahw, gjrmquke, gmenqwpb

The thresholds set for pruning the exposure metrics in order to select *memorized NEs* and *unmemorized NEs* are presented in Tables 8 and 9.

Dataset	$\text{exposure}_\theta^{word}$	$\text{exposure}_\theta^{trans}$
DBpedia	1	-
	-	0.01
CoNLL	0.8	-
MIT Movie	-	0.001

Table 8: Thresholds used for pruning the exposure metrics in order to select *memorized NEs*.

Dataset	$\text{exposure}_\theta^{word}$	$\text{exposure}_\theta^{trans}$
DBpedia	1e-06	-
	-	1e-06
CoNLL	1e-04	-
MIT Movie	-	1e-05

Table 9: Thresholds used for pruning the exposure metrics in order to select *unmemorized NEs*.

A. Bibliographical References

Aguilar, G., Maharjan, S., López-Monroy, A. P., and Solorio, T. (2017). A multi-task approach for named entity recognition in social media data. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 148–153, Copenhagen, Denmark, September. Association for Computational Linguistics.

Aguilar, G., López-Monroy, A. P., González, F., and Solorio, T. (2018). Modeling noisiness to recognize named entities using multitask neural networks on social media. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1401–1412, New Orleans, Louisiana, June. Association for Computational Linguistics.

Akbik, A., Bergmann, T., and Vollgraf, R. (2019). Pooled contextualized embeddings for named entity recognition. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 724–728, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Aly, R., Vlachos, A., and McDonald, R. (2021). Leveraging type descriptions for zero-shot named entity recognition and classification. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1516–1528, Online, August. Association for Computational Linguistics.

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z. (2007). Dbpedia: A nucleus for a web of open data. In *The Semantic Web*, pages 722–735, Berlin, Heidelberg. Springer.

Augenstein, I., Derczynski, L., and Bontcheva, K. (2017). Generalisation in named entity recognition. *Comput. Speech Lang.*, 44(C):61–83, July.

Brown, T., Mann, B., Ryder, N., and Subbiah, M. e. a. (2020). Language models are few-shot learners. In

H. Larochelle, et al., editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Carlini, N., Liu, C., Erlingsson, U., Kos, J., and Song, D. (2019). The secret sharer: Evaluating and testing unintended memorization in neural networks. In *Proceedings of the 28th USENIX Conference on Security Symposium*, SEC'19, page 267–284, USA. USENIX Association.

Derczynski, L., Nichols, E., van Erp, M., and Limsopatham, N. (2017). Results of the WNUT2017 shared task on novel and emerging entity recognition. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 140–147, Copenhagen, Denmark, September. Association for Computational Linguistics.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Ding, N., Xu, G., Chen, Y., Wang, X., Han, X., Xie, P., Zheng, H., and Liu, Z. (2021). Few-NERD: A few-shot named entity recognition dataset. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3198–3213, Online, August. Association for Computational Linguistics.

Fu, J., Liu, P., and Zhang, Q. (2020). Rethinking generalization of neural models: A named entity recognition case study. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):7732–7739, Apr.

Guerini, M., Magnolini, S., Balaraman, V., and Magnini, B. (2018). Toward zero-shot entity recognition in task-oriented conversational agents. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue*, pages 317–326, Melbourne, Australia, July. Association for Computational Linguistics.

Huang, J., Li, C., Subudhi, K., Jose, D., Balakrishnan, S., Chen, W., Peng, B., Gao, J., and Han, J. (2021). Few-shot named entity recognition: An empirical baseline study. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10408–10423, Online and Punta Cana, Dominican Republic, November. Association for Computational Linguistics.

Jiang, Z., Xu, F. F., Araki, J., and Neubig, G. (2020). How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

Levesque, H. J. (2014). On our best behaviour. *Artificial Intelligence*, 212:27–35.

Li, X., Feng, J., Meng, Y., Han, Q., Wu, F., and Li, J. (2020). A unified MRC framework for named entity recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5849–5859, Online, July. Association for Computational Linguistics.

Lin, B. Y., Lee, D.-H., Shen, M., Moreno, R., Huang, X., Shiralkar, P., and Ren, X. (2020a). TriggerNER: Learning with entity triggers as explanations for named entity recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8503–8511, Online, July. Association for Computational Linguistics.

Lin, H., Lu, Y., Tang, J., Han, X., Sun, L., Wei, Z., and Yuan, N. J. (2020b). A rigorous study on named entity recognition: Can fine-tuning pretrained model lead to the promised land? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7291–7300, Online, November. Association for Computational Linguistics.

Lison, P., Barnes, J., Hubin, A., and Touileb, S. (2020). Named entity recognition without labelled data: A weak supervision approach. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1518–1533, Online, July. Association for Computational Linguistics.

Liu, J. J. (2014). A conversational movie search system based on conditional random fields. In *Interspeech 2012*, October.

Ma, Y., Cambria, E., and Gao, S. (2016). Label embedding for zero-shot fine-grained named entity typing. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 171–180, Osaka, Japan, December. The COLING 2016 Organizing Committee.

Mengge, X., Yu, B., Zhang, Z., Liu, T., Zhang, Y., and Wang, B. (2020). Coarse-to-Fine Pre-training for Named Entity Recognition. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6345–6354, Online, November. Association for Computational Linguistics.

Petroni, F., Rocktäschel, T., Riedel, S., Lewis, P., Bakhtin, A., Wu, Y., and 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)*, pages 2463–2473, Hong Kong, China, November. Association for Computational Linguistics.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. (2019). Language models are unsupervised multitask learners. *OpenAI blog*.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Shang, J., Liu, L., Gu, X., Ren, X., Ren, T., and Han, J. (2018). Learning named entity tagger using domain-specific dictionary. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2054–2064, Brussels, Belgium, October-November. Association for Computational Linguistics.

Shwartz, V., Rudinger, R., and Tafjord, O. (2020). "you

are grounded!": Latent name artifacts in pre-trained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6850–6861, Online, November. Association for Computational Linguistics.

Taillé, B., Guigue, V., and Gallinari, P. (2020). Contextualized embeddings in named-entity recognition: An empirical study on generalization. In Joemon M. Jose, et al., editors, *Advances in Information Retrieval*, pages 383–391, Cham. Springer International Publishing.

Tjong Kim Sang, E. F. and De Meulder, F. (2003). Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Bonnie Webber, et al., editors. (2020). *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Online, November. Association for Computational Linguistics.

Yadav, V. and Bethard, S. (2018). A survey on recent advances in named entity recognition from deep learning models. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2145–2158, Santa Fe, New Mexico, USA, August. Association for Computational Linguistics.

Yang, Y. and Katiyar, A. (2020). Simple and effective few-shot named entity recognition with structured nearest neighbor learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6365–6375, Online, November. Association for Computational Linguistics.

Zhang, T., Xia, C., Lu, C.-T., and Yu, P. (2020). MZET: Memory augmented zero-shot fine-grained named entity typing. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 77–87, Barcelona, Spain (Online), December. International Committee on Computational Linguistics.

Zhao, Z., Wallace, E., Feng, S., Klein, D., and Singh, S. (2021). Calibrate before use: Improving few-shot performance of language models. In *ICML*, pages 12697–12706.

Zhou, B., Khashabi, D., Tsai, C.-T., and Roth, D. (2018). Zero-shot open entity typing as type-compatible grounding. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2065–2076, Brussels, Belgium, October-November. Association for Computational Linguistics.