Leveraging Type Descriptions for Zero-shot Named Entity Recognition and Classification

Rami Aly[1], **Andreas Vlachos**[1], **Ryan McDonald**[2]*
[1]Computer Laboratory, University of Cambridge, U.K.
[2]ASAPP
{rami.aly|andreas.vlachos}@cl.cam.ac.uk, ryanmcd@asapp.com

Abstract

A common issue in real-world applications of named entity recognition and classification (NERC) is the absence of annotated data for target entity classes during training. Zero-shot learning approaches address this issue by learning models that can transfer information from observed classes in the training data to unseen classes. This paper presents the first approach for zero-shot NERC, introducing a novel architecture that leverage the fact that textual descriptions for many entity classes occur naturally. Our architecture addresses the zero-shot NERC specific challenge that the not-an-entity class is not well defined, since different entity classes are considered in training and testing. For evaluation, we adapt two datasets, OntoNotes and MedMentions, emulating the difficulty of real-world zero-shot learning by testing models on the rarest entity classes. Our proposed approach outperforms baselines adapted from machine reading comprehension and zero-shot text classification. Furthermore, we assess the effect of different class descriptions for this task.

1 Introduction

Named entity recognition and classification (NERC) is the task of identifying spans of text corresponding to named entities and classifying these spans from a set of pre-defined entity classes. A prevalent issue for many real-world applications is that annotated data does not readily exist. This motivates the focus on the *zero-shot* setting (Xian et al., 2018; Wang et al., 2019), where annotated data is not available for the classes of interest. Instead, information available from *observed* classes must be transferred to *unseen* target classes.

Recently zero-shot approaches making use of textual representations to represent entity classes were explored for entity linking (EL) (Logeswaran et al., 2019; Wu et al., 2020) and named entity typing (NET) (Obeidat et al., 2019), which are similar to the NERC subtask of named entity classification (NEC). However, no previous work has addressed the task of zero-shot NERC, which additionally requires the detection of which tokens make up an entity in addition to its type, i.e. Named Entity Recognition (NER).

This paper is the first to study zero-shot NERC, by leveraging entity type descriptions. The task is illustrated in Figure 1. During testing, the input is a sentence and a set of target entity classes. each accompanied by its description, and the goal is to recognize and classify entities in these target classes. Descriptions contain crucial information for the task. Given as input "*Shantou Harbour, a natural river seaport, opens to the South China Sea.*" and a class *Facility* in Figure 1, using a description "*Names of human-made structures: infrastructure (streets, bridges), [...]*" a connection between *Facility* and *Shantou Harbour* can be made without having seen an annotated example in training. While using descriptions enables us to predict entity classes unseen in training, NERC poses the additional challenge of modelling the negative class (non-entity tokens) as its definition includes different entity classes and tokens in training and testing. It is possible that words observed as non-entities during training belong to one of the test classes, as seen in Figure 1: both *Huaqiao Park*, in training, and *Shantou Harbour*, during testing, are entities of the class *Facility*, however, *Huaqiao Park* is labelled as a non-entity in the former.

Based on this insight we propose several architectures for NERC based on cross-attention between the sentence and the entity type descriptions using transformers (Vaswani et al., 2017) combined with pre-training (Devlin et al., 2019). We

[*] Work done when author was working at Google.

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, pages 1516–1528
August 1–6, 2021. ©2021 Association for Computational Linguistics

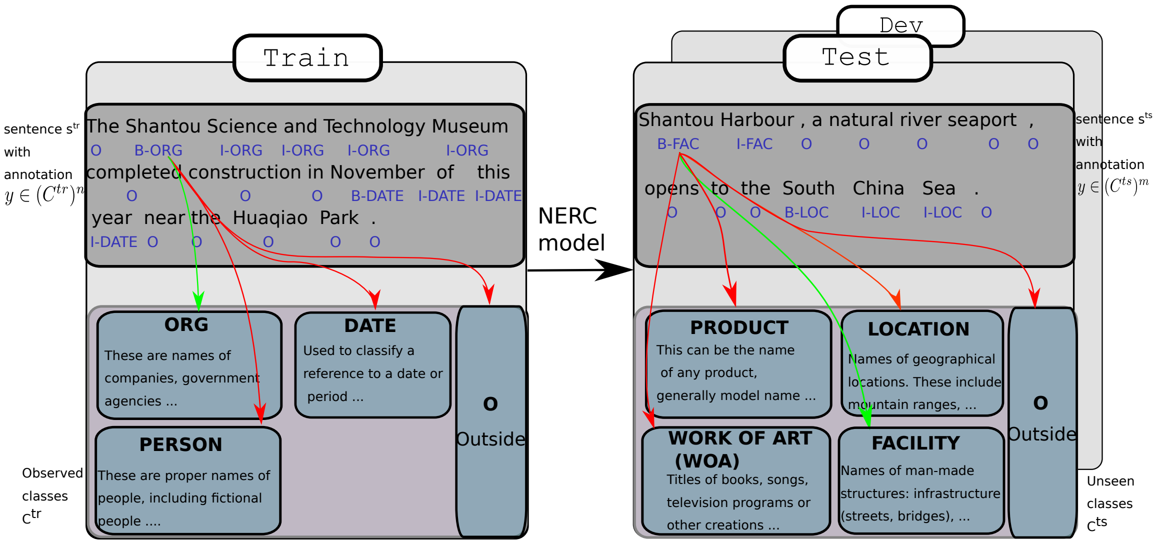
Train
sentence s^tr with annotation y ∈ (C^tr)^n
The Shantou Science and Technology Museum
O B-ORG I-ORG I-ORG I-ORG I-ORG
completed construction in November of this
O O O B-DATE I-DATE I-DATE
year near the Huaqiao Park .
I-DATE O O O O O
ORG
These are names of companies, government agencies ...
DATE
Used to classify a reference to a date or period ...
PERSON
These are proper names of people, including fictional people ....
O
Outside
Observed classes C^tr
NERC model
Dev
Test
Shantou Harbour , a natural river seaport ,
B-FAC I-FAC O O O O O
opens to the South China Sea .
O O O B-LOC I-LOC I-LOC O
PRODUCT
This can be the name of any product, generally model name ...
LOCATION
Names of geographical locations. These include mountain ranges, ...
WORK OF ART (WOA)
Titles of books, songs, television programs or other creations ...
FACILITY
Names of man-made structures: infrastructure (streets, bridges), ...
O
Outside
sentence s^ts with annotation y ∈ (C^ts)^m
Unseen classes C^ts

Figure 1: Zero-shot named entity recognition and classification.

explore modelling the negative class by (i) using a description for the negative class, (ii) modelling the negative class directly, (iii) modelling the negative class using the representations generated for the classes corresponding to types.

For evaluation we introduce zero-shot adaptations to two real-world NERC datasets with distinct properties: the OntoNotes (Pradhan et al., 2013) as well as the highly domain-specific MedMentions dataset (Mohan and Li, 2019). The adaptations adhere to recommendations to zero-shot evaluation (Xian et al., 2018) by evaluating models on the rarest classes while ensuring that all class sets are disjoint. Our best model achieves a macro F_1 of 0.45 on `OntoNotes-ZS` and 0.38 on `MedMentions-ZS`, outperforming a state-of-the-art MRC model for NERC (Li et al., 2020; Sun et al., 2020) and an adapted zero-shot text classification model (Yin et al., 2019). An analysis on the classification and recognition task in isolation highlights the importance of the description choice, finding that annotation guidelines result in higher scores than the class name itself or Wikipedia passages.

2 Zero-shot NERC

In NERC, given a sentence $s = w_1, ..., w_n$ of length n and a description d_c for each class $c \in \mathbb{C}^{ts}$ in the test set, we predict a sequence of labels $\hat{\boldsymbol{y}} \in (\mathbb{C}^{ts})^n$, with n being the length of the sentence. We model the task as multiclass classification, which despite ignoring the sequential structure of the output, it has been found to be competitive (Lample et al., 2016; Rei, 2017). Thus, we predict the correct class for each token w at position t: $\arg\max_{c \in \mathbb{C}^{ts}} F(s, w_t, d_c)$, using a suitable function F modelling the semantic affinity between w_t and d_c in the context of s. The parameters of F need to be learned without annotated data for $\mathbb{C}^{ts}$, but with annotated data and descriptions for the training C^{tr} classes.

To model F we focus on the use of cross-attention (Humeau et al., 2019; Wolf et al., 2019b) in the form of a transformer encoder (Vaswani et al., 2017). For each type description d_c, the cross-attention encoder (X-ENC) generates a vector representation $v_{t,c} \in \mathbb{R}^h$ for a token w_t in the sentence s:

$$v_{1,c}, ..., v_{n,c} = \text{X-ENC}(s, d_c). \qquad (1)$$

The vector $v_{t,c}$ of each token is then linearly transformed

$$o_{t,c} = v_{t,c} \cdot w^T + b, \qquad (2)$$

with $v_{t,c} \in \mathbb{R}^h$ and $o_{t,c} \in \mathbb{R}$. The value $o_{t,c}$ indicates how likely is that token w_t belongs to entity class c.

In order to be able to recognize entities in addition to classifying them, the scores for each token $o_{t,c_1}; ...; o_{t,c_k}$ are concatenated with a score for belonging to the negative class $o_{t,neg}$, corresponding to not belonging to any of the types considered:

$$o_t = (o_{t,c_1}; ...; o_{t,c_k}; o_{t,neg}) \qquad (3)$$

with $o_t \in \mathbb{R}^{k+1}$. Obtaining a good estimate for this score is a key challenge in performing zero shot NERC and we discuss it in the next section. We then select the class with the highest score probability after applying a softmax operation:

$$\hat{y}_t = \arg\max_{c \in \mathbb{C}^{ts}} F(s, w_t, d_c) \\ = \arg\max_{c \in \mathbb{C}^{ts}} \frac{o_{t,c}}{\sum_{c' \in \mathbb{C}^{ts}} o_{t,c'}}. \quad (4)$$

We label this model *Sequential Multiclass Cross-attention Model* (SMXM). Referring to the initial example, cross-attention enables *Shantou Harbour* to attend to *infrastructure* in the type description of the class *Facility*, generating a representation for this token based on the type description in the context of the sentence.

Cross-attention Encoder The cross-attention model is based on the pre-trained transformer encoder BERT (Devlin et al., 2019) which allows the model to capture surface-level information as well as semantic aspects between all words of the input (Jawahar et al., 2019). For X-ENC the input tuple (s, d_c) is structured in the form: $x_{\text{X-ENC}}$ = [CLS] s [SEP] d_c [SEP].

2.1 Modelling the negative class

As discussed in Section 1, the non-entity class creates a challenging setting it is possible that words observed as non-entities during training belong to one of the test classes. We explore three approaches to modelling the negative class: (i) using a (textual) description for the negative class, (ii) modelling the negative class directly, (iii) modelling the negative class using the representations generated for the classes corresponding to types.

Description-based encoding Assuming a description for the negative class d_{neg}, it is straightforward to obtain a representation $v_{t,neg}$ for each token belonging to it using the cross-attention encoder, which is then transformed to a score via a weight vector w_{neg} for this class:

$$o_{t,neg} = v_{t,neg} \cdot w_{neg}^T + b_{neg} \quad (5)$$

However, this approach requires a description to describe something that *is not* rather than is. This makes it very difficult in practice to make an informed decision on the most suitable description. Also, non-entity tokens are likely to differ between training and testing, thus a fixed description is unlikely to perform well.

Independent encoding The negative class can be directly modelled since it is observed in the training data. Thus, instead of exploring cross-attention, each token is represented for the negative class in the context of the sentence without taking any description into account:

$$v_{1,neg}, ..., v_{n,neg} = \text{ENC}(s), \quad (6)$$

with ENC being a standard transformer encoder (Vaswani et al., 2017). Similar to the description-based approach, $v_{t,neg}$ is linearly transformed to $o_{t,neg}$ using a separate vector w_{neg} (c.f. Eq. 5).

Class-aware encoding Description-based and independent encodings do not model the fact that not every entity labelled as a non-entity during training is a non-entity during testing in zero-shot NERC. Instead, we propose to model the negative class by combining the representations generated for the other classes, as generated by the cross-attention encoder (Eq. 1): $v_{t,c0}, ..., v_{t,ck}$. Each vector is then linearly transformed, using w_{neg-cl} and then concatenated to a feature map m. We then apply a max-pooling operation over this feature set and take the maximum value:

$$o_{t,neg-cl} = max\{m\}. \quad (7)$$

Finally, we compute $o_{t,neg}$ by linearly combining the representation from the independent encoding and $o_{t,neg-cl}$.

2.2 Training

To prevent the cross-attention encoder from overfitting on the few class descriptions, we use a regularizer in the form of *entity masking*, inspired by the masked language modelling objective used in BERT, to train the model on the training classes $\mathbb{C}^{tr}$. During training with a probability p (tuned as a hyperparameter) the entire entity that is to be classified is masked in the input to the model. This regularization avoids lexical memorization and encourages the model to learn entity context to class description affinities, while still learning to incorporate aspects of the entity itself (e.g. capitalization, shape, morphology) and relating them to the type description. A cross-attention model for tasks such as EL is much less likely to overfit since each entity is associated with a unique description and there is a much larger number of them than entity classes. Due to the label imbalance caused by the

	OntoNotes-ZS			MedMentions-ZS		
Statistic	Train	Dev	Test	Train	Dev	Test
# sentences	59924	8528	8262	28226	9302	9382
# words	1088503	147724	152728	721552	242358	241786
# total entities	54576	1785	1754	113095	1710	1431
# compound entities	31257	905	1628	59031	806	637
# consecutive entities	7902	49	121	30545	125	152
# consecutive entities of same class	3448	39	95	14727	120	147
# unique mentions (not in Train)	–	634	495	–	574	721

Table 1: Quantitative statistics of zero-shot dataset OntoNotes-ZS and MedMentions-ZS.

Train	PERSON(15429), GPE(15405), ORG(12820), DATE(10922)	Biologic Function(24989), Chemical(22351), Healthcare Activity(14764), Anotomical Structure(12571), Finding(9811), Spatial Concept(7511), Intellectual Product(5994), Research Activity(5443), Eukaryote(4922), Population Group(3574), Medical Device(1165)
Dev	NORP(847), MONEY†(274), ORDINAL†(232), PERCENT†(177), EVENT(143), PRODUCT(72), LAW(40)	Organization(452), Injury or Poisoning(434), Clinical Attribute(404), Virus(224), Biomedical Occupation or Discipline(196)
Test	CARDINAL†(945), TIME†(212), LOC(179), WORK OF ART(166), FAC(135), QUANTITY†(105), LANGUAGE†(22)	Bacterium(449), Professional or Occupational Group(360), Food(321), Body Substance(212), Body System(89)

Table 2: Zero-shot class splits and number of occurrences for OntoNotes-ZS and MedMentions-ZS. Trivial classes for which a rule-based system is sufficient are denoted with †.

negative class, we use class weights q_c incorporated to the cross-entropy loss:

$$\sum_{i=1}^{c} q_i \cdot p(y_{t,i}) \cdot log(p(\hat{y}_{t,i})). \qquad (8)$$

While the factor q is kept to 1 for all non-negative classes, for the negative class q is set using the underlying training dataset distributions using the ratio $\frac{\text{\# entities}}{\text{\# non-entity words}}$ and further tuned within that range as a hyperparameter.

3 Evaluation setup

3.1 Datasets for Zero-Shot NERC

We present adaptations to OntoNotes (Pradhan et al., 2013) and MedMentions (Mohan and Li, 2019) for zero-shot NERC evaluation. OntoNotes is a common benchmark dataset for NERC systems while the more recent MedMentions dataset consists of domain-specific biomedical data. The annotations in the latter are based on the Unified Medical Language System (UMLS) ontology (Bodenreider, 2004) and do not only include proper named entities but also *concepts*. For instance, in the passage "*modeling nurse-patients*", *modeling* is annotated with the concept Research Activity, thus rendering it more challenging.

The adaptations follow recommendations for zero-shot evaluation by Xian et al. (2018): **(i)** Zero-shot methods should be evaluated on the rarer classes, as in real-world scenarios annotated data is likely to be available for the more common ones, **(ii)** Evaluation metrics should focus on per-class averaged scores to account for the imbalance in terms of samples per class, thus we evaluate our models with the *macro-averaged* F_1 metric, **(iii)** Hyperparameters have to be tuned on a development set of classes disjoint from both the training and test set, **(iv)** Pre-trained neural networks used for zero-shot learning can be trained on arbitrary amount of data as long as the training data does not contain samples of the test set.

To create the zero-shot versions of both OntoNotes and MedMentions abiding by rule (i) we measure the frequencies of their respective entity types and keep the four and eleven most frequent ones in OntoNotes and MedMentions, respectively, for training. The remaining ones are split between development and test set by sorting them by frequency and then assigning them alternating between the two sets. To create the zero-shot splits we use the default data splits and remove all annotations of classes that are not associated with the respective split. Quantitative statistics of OntoNotes-ZS and MedMentions-ZS are shown in Table 1. In addition to ensuring that we evaluate on the rarer classes, we also wanted to ensure the classes considered are not trivial to recognize. For example, the class PERCENT in OntoNotes is only assigned to percentages, whose surface form follow regular patterns, while WORK OF ART or PRODUCT are more difficult to recognize. Based on the annotation guidelines of Ontonotes, seven classes were identified to be trivial to recognize (c.f. denoted with † in Table 2). To verify this, a simple rule-based system developed for these classes achieved between 0.60 and 0.89 micro F_1, only slightly worse than the fully supervised state-of-the-art NERC model of (Li et al., 2020) (see supplementary material). These classes were excluded from our experiments. We did not identify such trivial classes in MedMentions.

3.2 Entity type descriptions

Source	Avg. #tokens	Longest desc.	Shortest desc.
GL	57	129	4
WN	58	164	13
Wiki	81	160	19
SN	34	102	11
MT	67	116	14
Wiki	142	221	17

Table 3: Quantitative characteristics for different entity type description sources. Statistics measured in tokens and calculated over all classes.

A basic description is to simply use the **class name** itself. In addition, we consider three readily available type description sources for each dataset. The options for OntoNotes are:

Annotation guidelines [GL] They have been used to annotated the dataset. These descriptions are highly informative containing precise definitions accompanied by examples, as they should help a human perform the task.

WordNet [WN] Secondly, descriptions of the lexical database WordNet are employed using it's *synsets* feature.

Wikipedia [Wiki] The first one to three sentences of the most related article to a class.

For MedMentions, we use the aforementioned **Wikipedia** descriptions, as well as:

UMLS Semantic Network [SN] Since the MedMentions dataset is based on the UMLS ontology we explore the short descriptions provided by the UMLS Semantic Network Browser[1].

UMLS Metathesaurus [MT] The Metathesaurus[2] browser is a search engine that agglomerates information of different biomedical sources. For entity type not found in it, semantically similar or subordinate classes are used, e.g. Biomedical Research for Biomedical Occupation or Discipline. Quantitative characteristcs of the description types are shown in table 3.

To obtain negative type descriptions, three manually selected sentences from the training set are used that are free of *any* named entities. We also explored alternating between multiple negative descriptions that we had compiled, however, results were generally worse.

4 Experiments

4.1 Implementation details

All models are implemented using PyTorch (Paszke et al., 2017) and the huggingface implementation (Wolf et al., 2019a) of BERT, using the case-sensitive version of BERT-Large unless otherwise stated. The results reported are the averages of two runs.

All *I-* or *B-* prefixes to a label were removed for simplicity. Therefore, each entity class is defined by a single label. This simplification results in ambiguity for the NERC task in the case of two consecutive named-entities of the same class, however it reduces the model parameters by half while affecting 5.8% of the entities across the validation and test splits of both datasets (c.f. row *# consecutive entities of same class* in Table 1). Sentences without any annotations were also excluded.

The pre-training data of BERT has been compared to the development and test splits of both datasets to ensure that it has not been pre-trained on testing data (rule (iv) of Xian et al. (2018)) [3].

The hyperparameters for each model were mainly optimized on the validation split of the OntoNotes dataset considering only the non-trivial classes, and then used for the experiments with the MedMentions-ZS dataset. Only the learning rate was tuned for MedMentions-ZS separately. The best model according to development macro-averaged F_1 during training was tested in all experiments on both datasets. Further details

[1] https://uts.nlm.nih.gov/semanticnetwork

[2] https://uts.nlm.nih.gov/uts/umls

[3] The dataset has been compared only to the latest Wikipedia dump as the book corpus is not hosted anymore.

on the hyperparameter choice are in the supplementary material.

4.2 Baseline models

While a simple Tf-idf similarity baseline that measures the overlap between the sentence and entity description by computing the cosine similarity shown to be a good baseline for zero-shot entity linking (Logeswaran et al., 2019), F_1 scores on NERC were consistently below 0.04 on both datasets. Similar observation applies to similarity scores based on word2vec embeddings (Mikolov et al., 2013) as used in (Yin et al., 2019), highlighting the difficulty of this task. Our baselines thus focus on current state-of-the-art models in both NERC and related zero-shot tasks.

Binary Entailment Model (BEM) is an NERC adjusted model of the state-of-the-art approach for zero-shot text classification (Yin et al., 2019). They employ BERT, fine-tuned on an entailment dataset, to classify whether a class description (*The text is about X*) is entailed by the text. To adapt this model to NERC, we modify the description to *The word is of type X* with X being the entity class name, and classify each word instead of the entire sentence. Since their model generates a binary output for each class, the negative prediction for all classes predicts the negative class. By treating each sentence-description pair independently, the relationship between classes as well as the complexity of the negative class in zero-shot evaluation is ignored. We fine-tune BERT-Large on MNLI (Williams et al., 2018), as it performed best in the experiments of (Yin et al., 2019), before training BEM on the zero-shot datasets using adjusted class weights, which has been crucial for successful training of the model; not using it resulted in degenerated solutions in preliminary experiments. The proposed entity masking objective is not suitable for BEM's binary classification approach as it would simply learns to predict the masked token to be an entity during training.

MRC for NERC is an approach by Li et al. (2020) who construct queries for entity classes and transform NERC to a machine reading comprehension task for fully supervised flat and nested NERC. Their model generates a span by predicting start and end indices for each entity as well as a matching score for each possible start-end index. Predictions for each entity type are made independently, similar to BEM. Their model showed promising results for the transfer learning experiment when training on the CoNLL03 dataset and testing on OntoNotes, with the latter consisting of a superset of CoNLL03 entity classes, yet it was not tested on completely distinct training and test labels, i.e. zero-shot learning. However, results for our zero-shot task were too low to be considered. We hypothesise two causes: i) In our zero-shot setup the dataset is heavily imbalanced, as most token spans are not entities (typically one to three out of n^2 in a sentence of length n) ii) an incorrect prediction in either the start index, end index, or matching score results in an overall incorrect span, and the accuracy for each of these is unlikely to be high in the zero shot setup. Thus, we simplified the model by excluding the matching matrix, and we use the start and end index with greedy closest-matching to compute the entity span, similar to (Sun et al., 2020). MRC also has been trained using adjusted class weights.

Ontonotes-ZS				
Model	Dev		Test	
	Token	Span	Token	Span
BEM	0.28	0.18	0.23	0.11
MRC	0.15	0.15	0.22	0.18
SMXM	**0.35**	**0.23**	**0.45**	**0.25**
SMXM$_{\text{base}}$	0.30	0.19	0.42	0.20

MedMentions-ZS				
Model	Dev		Test	
	Token	Span	Token	Span
BEM	0.28	0.19	0.34	0.22
MRC	0.19	0.21	0.23	0.26
SMXM	**0.33**	**0.23**	**0.38**	**0.27**
SMXM$_{\text{base}}$	0.31	0.20	0.30	0.21

Table 4: Macro-averaged F_1 of NERC on `OntoNotes-ZS` and `MedMentions-ZS`, reporting token-based and span-based scores for all baselines and SMXM with class-aware encoding. Best results are highlighted in **bold**. *base* indicates a model based on the smaller BERT-Base encoder. All other models use Bert-Large encoders.

4.3 Results

NERC Results for both datasets are shown in Table 4, for both token and span-level F_1. We only report results on the best performing entity description which is the same across all models, i.e. annotation guidelines and Metathesaurus descriptions for `OntoNotes-ZS` and

MedMentions-ZS, respectively; we discuss the impact of description choice in the next section. Shown SMXM results use class-aware encoding of the negative class since it performed better than the other approaches considered (c.f. section 4.4). Statistical significance was determined using the two-tailed Monte Carlo permutation test with 5000 repetitions with $p < 0.05$.

Our proposed model, SMXM, performs significantly better than all models on both datasets, with a token-level score of 0.45 on and 0.38 for OntoNotes-ZS and MedMentions-ZS, respectively. Comparing SMXM with SMXM$_{\text{base}}$, trained on the smaller BERT-Base (335M vs 109M parameters) highlights the value of larger scale pretraining for domain-specific applications. Scores decrease on both datasets when using the smaller model, with a substantial decrease on MedMentions-ZS to only 0.30. Despite its smaller size, SMXM with Bert-Base remains competitive to both BEM and MRC which use BERT-Large. The BEM baseline achieves significantly better token-level scores than *MRC for NERC* on the development split of OntoNotes-ZS and on both splits of MedMentions. While the *MRC for NERC* model achieves poor token-level results, its span-level scores are more comparable to BEM and SMXM, even significantly outperforming BEM on the MedMentions-ZS development split despite a much lower token-level score. *MRC for NERC* has the smallest delta between the token and span-level score out of all models, yet overall scores remained low due to the difficulty of inferring the correct start and end index based only on the description in a zero-shot setup and generalizing to new, unseen types, e.g. determining whether the article *the* belongs to an entity or not (*the* is part of DATE but generally not of PRODUCT).

Per-class scores Scores for each class using SMXM are shown in Table 5. For OntoNotes, scores are comparable across the different classes, with WORK OF ART performing worse than the others. In contast, for MedMentions some classes are recognized and classified with comparably high accuracy, such as Bacterium, while Body Substance and Body System score very low. A possible explanation is the similarity (in semantics and/or description) between these classes and classes used for training. For instance, some example entities in Body System's description are also found in Anatomical Structure (e.g. *cardiovascular system*). This would further explain the very high recall but low precision, as entities belonging to the training classes are (erroneously) identified as entities of these test classes.

Class	precision	recall	f1-score
FAC	0.35	0.75	0.48
LOC	0.39	0.82	0.53
WORK OF ART	0.38	0.35	0.36
Bacterium	0.55	0.79	0.66
Body Substance	0.09	0.58	0.17
Body System	0.08	0.87	0.16
Food	0.33	0.68	0.47
Prof. or Occ. Group	0.31	0.65	0.44

Table 5: Class-based token-level macro-F$_1$ scores of SMXM on the test set of OntoNotes-ZS (top) and MedMentions-ZS (bottom).

4.4 Discussion

Analysis of entity descriptions Results on the development set using SMXM with the different entity descriptions introduced in section 3.2 are shown in Table 6. Annotation guideline descriptions performs significantly better than all other descriptions on OntoNotes-ZS. Metathesaurus descriptions work best on MedMentions-ZS, with Semantic Network descriptions performing only slightly worse. Using the class name is a surprisingly strong baseline description, performing comparably to WordNet descriptions on OntoNotes-ZS and even better than Wikipedia on MedMentions-ZS. While Wikipedia works well on general types, it performs poorly on the domain-specific types of MedMentions-ZS.

	OntoNotes-ZS		MedMentions-ZS	
Model	Token	Span	Token	Span
Class name	0.25	0.18	0.28	0.18
Wiki	0.32	0.21	0.27	0.19
WN/SN	0.26	0.20	0.29	0.21
GL/MT	**0.35**	**0.23**	**0.31**	**0.22**

Table 6: Averaged macro F$_1$ of NERC on the dev sets for SMXM with different type descriptions.

Analysing the scores, we identified three properties of descriptions with negative effect on performance: vagueness, noise, and negation. Most UMLS based type descriptions are abstract or underspecified, and require either substantial background information or expert knowledge to be useful; for instance, Eukaryote in SN description ("*One of the three domains of life, also called Eukarya. These are organisms whose cells are enclosed in membranes and possess a nucleus.*"). Furthermore, many descriptions contain noise or unrelated information (e.g. those obtained by Wikipedia). Finally, classes defined by negations or cross-references to other classes result in worse performance, as negated sentences add less information about the class in question. Cross-references cannot be processed by any of the models, as they cannot directly link parts of a class' description to another. Exploring this semi-structured knowledge is interesting future work. On the other hand, we found that explicit examples (e,g. "*infrastructure (streets, bridges)*") and mentions of syntactic and morphological cues (e.g. "*These are usually surrounded by quotation marks in the article [...]*") make the annotation guidelines perform particularly well.

To validate this qualitative analysis, we modified each dataset's best performing description with the aim to make one worse and one better. First, we worsen the annotation guidelines by removing all explicit mentions of entities and syntactic cues. The token-based macro-F_1 for NERC when using SMXM decreased by 0.05 when explicit examples are removed. Secondly, to improve the Metathesaurus descriptions we removed negations, made them less abstract, and added explicit examples.

The modifications on the UMLS descriptions improve the scores by around 0.03 on the development set. We used the modified Metathesaurus descriptions for all models in result table 4 and table 7. Only around forty minutes have been invested to modify the UMLS annotations without expertise in the biomedical domain, likely leaving much room for improvements.

Non-entity class modelling We separately analysed how well the different approaches model the negative class. Results on the development set are reported in Table 7. The token-level score of SMXM$_{\text{ca}}$ with the class-aware encoding of the negative class outperforms both the independent encoding SMXM$_{\text{ind.}}$ as well as the negative class description based encoding SMXM$_{\text{desc.}}$ approach significantly on the NERC task, confirming the motivation of this approach.

	OntoNotes-ZS		**MedMentions-ZS**	
Model	Token	Span	Token	Span
SMXM$_{\text{desc.}}$	0.24	0.18	0.29	0.19
SMXM$_{\text{ind.}}$	0.28	0.19	0.30	0.21
SMXM$_{\text{ca}}$	**0.35**	**0.23**	**0.33**	**0.23**

Table 7: Macro-averaged F_1 of NERC on the dev set of OntoNotes-ZS and MedMentions-ZS for different approaches to modelling the negative class.

Alternative Class Splits While switching classes between the development and test split resulted in overall similar results (tested on three different splits for MedMentions), reducing the number of training classes and redistributing them on the dev and test splits led to a substantial decrease in performance. Results for the extreme case where the number of training classes for MedMentions-ZS has been reduced to the four most frequent ones (with the dev and test sets having eight and nine classes, respectively) are shown in Table 8. As seen, SMXM still performs the best, however, only with a score of 0.14.

Model	Dev		Test	
	Token	Span	Token	Span
BEM	0.10	0.06	0.10	0.07
MRC	0.07	0.08	0.06	0.05
SMXM	**0.13**	**0.09**	**0.14**	**0.09**

Table 8: Macro-averaged F_1 of NERC on the modified class splits with only four training classes and nine test classes on MedMentions-ZS.

Complexity The complexity of our model's and baselines' encoding step in terms of classes is $\mathcal{O}(C)$, with C being the number of test classes (including the negative class). This is an increase in complexity over $\mathcal{O}(1)$ in the traditional scenario, however, during training the gradients are accumulated across the inputs, leading to faster convergence. With varying description lengths for different sources (c.f. table 3), the input sequence length is another important factor to consider regarding the model's efficiency, leading to an overall complexity of $\mathcal{O}(CN^2)$, with N being the length of the input sequence. In our experiments, the runtime with SMXM$_{\text{base}}$ was the shortest, followed by BEM (due to the entailment descriptions being much shorter), SMXM, and last MRC.

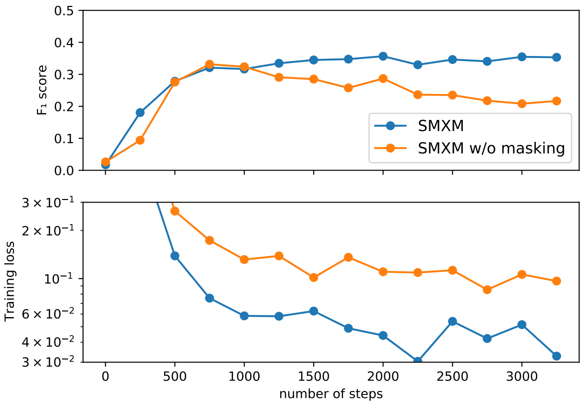

Figure 2: Learning behavior analysis of SMXM and SMXM w/o entity masking on `OntoNotes-ZS` dev.

Entity Masking Finally, we study the impact of entity masking in Figure 2. First, we plot the validation F_1 score during training for SMXM and SMXM w/o entity masking using guideline annotations. Second, the training loss of the same models in terms of cross-entropy (i.e. Eq. 8). The top plot shows that SMXM's F_1 score converges more slowly but to a higher value than SMXM's highest value w/o masking by 0.03 points. The model's validation F_1 w/o entity masking decreases in later iterations, indicating overfitting. We confirmed this by observing a higher validation loss when no masking is used. Interestingly, as seen in the loss plot (bottom), the training loss is much lower when using entity masking. This is likely due to entity masking providing additional implicit supervision to the model: masked tokens cannot be the non-entity class. For these masked tokens the model can focus on the entity classification in isolation which appears to help the model extract more useful supervision signal, as indicated by the higher validation F_1 achieved. When trained with masking, SMXM's training loss closely follows the trend of the validation F_1, indicating good transfer learning from the model's training objective to the zero-shot evaluation.

5 Related Work

State-of-the-art approaches to NERC include the bidirectional LSTM-CRF (Lample et al., 2016), and more recently models based on the pretrained transformer architectures, e.g. BERT (Devlin et al., 2019). these methods are unsuitable for zero-shot learning, with exception to the explored baselines in this paper (Li et al., 2020; Sun et al., 2020). Apart from NERC, manually defined class descriptions have also been explored for relation classification (Obamuyide and Vlachos, 2018) who pose the task as one of textual entailment. Obeidat et al. (2019) use descriptions for zero-shot NET, however, similar to a previous attempt by Ma et al. (2016), they use the underlying hierarchy to only include unseen classes in the leaves of the hierarchy to reduce the relevant unseen classes to only two or three.

The only work on zero-shot word sequence labelling (Rei and Søgaard, 2018) explores the transfer from labels on a sentence level objective (e.g. sentiment analysis) to a token or phrase-based annotation, similar to Täckström and McDonald (2011). Guerini et al. (2018) label their approach zero-shot named entity recognition, however, they focus on recognizing unseen *entities* not *entity classes*. Finally, Fritzler et al. (2019) focused on few-shot NERC using *prototypical networks* (Snell et al., 2017). They tested their model in the zero-shot setting, but concluded that their approach is not suitable for zero-shot learning as the results on OntoNotes were too low.

6 Conclusions & Future work

This paper explored the task of zero-shot NERC with entity type descriptions to transfer knowledge from observed to unseen classes. We addressed the zero-shot NERC specific challenge that the not-an-entity class is not well defined by proposing a multiclass architecture that uses class-aware encoding to model the negative class. The models were evaluated based on zero-shot adaptations of the OntoNotes and MedMentions dataset. The results show that the proposed model outperforms strong baselines and further indicate that high-quality entity descriptions (i.e. annotation guidelines) are an effective way to transfer knowledge from observed to unseen classes. Future work will aim to incorporate the dependencies between the labels predicted.

Acknowledgements

We thank the anonymous reviewers for their time and effort giving us feedback on our paper. This work was supported by the Engineering and Physical Sciences Research Council Doctoral Training Partnership (EPSRC). Andreas Vlachos is supported by the ERC grant AVeriTeC (GA 865958).

References

Olivier Bodenreider. 2004. The unified medical language system (UMLS): Integrating biomedical terminology. *Nucleic acids research*, 32:D267–D270.

Jason P.C. Chiu and Eric Nichols. 2016. Named Entity Recognition with Bidirectional LSTM-CNNs. *Transactions of the Association for Computational Linguistics*, 4:357–370.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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*, pages 4171–4186, Minneapolis, MN, USA.

Alexander Fritzler, Varvara Logacheva, and Maksim Kretov. 2019. Few-shot classification in named entity recognition task. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, SAC '19, pages 993–1000, Limassol, Cyprus.

Abbas Ghaddar and Phillippe Langlais. 2018. Robust Lexical Features for Improved Neural Network Named-Entity Recognition. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1896–1907, Santa Fe, NM, USA.

Marco Guerini, Simone Magnolini, Vevake Balaraman, and Bernardo Magnini. 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.

Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. 2019. Poly-encoders: Architectures and pre-training strategies for fast and accurate multi-sentence scoring. In *International Conference on Learning Representations*, New Orleans, LA, USA.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What Does BERT Learn about the Structure of Language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations*, CA, USA.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural Architectures for Named Entity Recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270, San Diego, CA, California.

Peng-Hsuan Li, Ruo-Ping Dong, Yu-Siang Wang, Ju-Chieh Chou, and Wei-Yun Ma. 2017. Leveraging Linguistic Structures for Named Entity Recognition with Bidirectional Recursive Neural Networks. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2664–2669, Copenhagen, Denmark.

Xiaoya Li, Jingrong Feng, Yuxian Meng, Qinghong Han, Fei Wu, and Jiwei Li. 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.

Lajanugen Logeswaran, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, Jacob Devlin, and Honglak Lee. 2019. Zero-Shot Entity Linking by Reading Entity Descriptions. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3449–3460, Florence, Italy.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations 2019*, New Orleans, LA, USA.

Yukun Ma, Erik Cambria, and Sa Gao. 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.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems 26*, pages 3111–3119, Lake Tahoe, USA.

Sunil Mohan and Donghui Li. 2019. MedMentions: A Large Biomedical Corpus Annotated with UMLS Concepts. In *Proceedings of the 2019 Conference on Automated Knowledge Base Construction (AKBC 2019)*, Amherst, MA, USA.

Abiola Obamuyide and Andreas Vlachos. 2018. Zero-shot Relation Classification as Textual Entailment. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 72–78, Brussels, Belgium.

Rasha Obeidat, Xiaoli Fern, Hamed Shahbazi, and Prasad Tadepalli. 2019. Description-Based Zero-shot Fine-Grained Entity Typing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*, pages 807–814, Minneapolis, MN, USA.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch. In *2017 Conference on Neural Information Processing Systems*, Long Beach, CA, USA.

Martin Popel and Ondřej Bojar. 2018. Training tips for the transformer model. *The Prague Bulletin of Mathematical Linguistics*, 110(1):43–70.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Hwee Tou Ng, Anders Björkelund, Olga Uryupina, Yuchen Zhang, and Zhi Zhong. 2013. Towards Robust Linguistic Analysis using OntoNotes. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 143–152, Sofia, Bulgaria.

Marek Rei. 2017. Semi-supervised Multitask Learning for Sequence Labeling. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 2121–2130, Vancouver, Canada.

Marek Rei and Anders Søgaard. 2018. Zero-Shot Sequence Labeling: Transferring Knowledge from Sentences to Tokens. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*, pages 293–302, New Orleans, LA, USA.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. In *5th Workshop on Energy Efficient Machine Learning and Cognitive Computing*.

Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems 30*, pages 4077–4087, Long Beach, CA, USA.

Cong Sun, Zhihao Yang, Lei Wang, Yin Zhang, Hongfei Lin, and Jian Wang. 2020. Biomedical named entity recognition using bert in the machine reading comprehension framework. *ArXiv*, 2009.01560.

Oscar Täckström and Ryan McDonald. 2011. Discovering Fine-Grained Sentiment with Latent Variable Structured Prediction Models. In *Proceedings of the 33rd Conference on Advances in Information Retrieval*, Lecture Notes in Computer Science, pages 368–374, Dublin, Ireland.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems 30*, pages 5998–6008, Vancouver, Canada.

Wei Wang, Vincent W. Zheng, Han Yu, and Chunyan Miao. 2019. A survey of zero-shot learning: Settings, methods, and applications. *ACM Trans. Intell. Syst. Technol.*, 10(2):1–37.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. 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 1112–1122, New Orleans, LA, USA.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019a. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

Thomas Wolf, Victor Sanh, Julien Chaumond, and Clement Delangue. 2019b. Transfertransfo: A transfer learning approach for neural network based conversational agents. *ArXiv*, abs/1901.08149.

Ledell Wu, Fabio Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. 2020. Scalable zero-shot entity linking with dense entity retrieval. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6397–6407, Online.

Yongqin Xian, Christoph H. Lampert, Bernt Schiele, and Zeynep Akata. 2018. Zero-Shot Learning - A Comprehensive Evaluation of the Good, the Bad and the Ugly. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41:2251–2265.

Wenpeng Yin, Jamaal Hay, and Dan Roth. 2019. Benchmarking Zero-shot Text Classification: Datasets, Evaluation and Entailment Approach. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 3912–3921, Hong Kong, China.

Supplementary Material

6.1 Details on the Evaluation setup

Several slightly different versions of the OntoNotes dataset have been used in papers. Our OntoNotes version aligns with the ones used in (Li et al., 2017; Ghaddar and Langlais, 2018; Chiu and Nichols, 2016).

Class	Rule
PERCENT	Any token that is % and its preceding number, as well as the preceding adverb such as 'about', 'around', or 'approximately'.
MONEY	Any token [dollars, euro, yuan, pound, ...] or its symbolic representation and preceding numbers, incl. [hundred(s)', thousand(s)', 'million(s)', 'billion(s)'].
ORDINAL	Any word that is either 'first', 'second', or 'third', or compound of 'th' and a number.
LANGUAGE	Frequent languages (English, German,...) and if preceded by [in, into, speak, write, talk, listen, ...]
TIME	'a.m', 'p.m.', 'morning', 'evening', 'night', 'minute(s)', 'hour(s)' etc. and any preceding or consecutive numerical and relevant adverb/preposition.
QUANTITY	One of ca. 20 SI units (incl. its abbreviation) and preceding number and relevant adverb/preposition.
CARDINAL	CARDINAL is only marked if it is a numerical and not a year nor `ORDINAL`, `MONEY`, `PERCENTAGE`, nor `QUANTITY`.

Table 9: Rule-based approach on non-challenging classes of `OntoNotes-ZS`.

6.2 Details on the Experimental Setup

Experiments were run on a Quadro RTX 8000. The parameter vectors/matrices w and w_{neg} have been randomly initialized from a uniform distribution $U(-\sqrt{b}, \sqrt{b})$ with $b = \frac{1}{\text{in-features}}$.

The models use the Adam optimizer (Kingma and Ba, 2015) with decoupled weight decay, called AdamW (Loshchilov and Hutter, 2019). Recommendations of related literature have been taken into account when selecting the hyperparameters and search space (Devlin et al., 2019; Sanh et al., 2020; Popel and Bojar, 2018). The tuned hyperparameters are the batch size, learning rate lr, weight decay ld, linear dropout dr, entity masking probability p and warmup steps wr. All models use $ld = 0$, $dr = 0.5$, and $wr = 0$ as they have not been very sensitive regarding these parameters. For the learning rate, the rates $lr_{BEM} = 4e^{-6}$, $lr_{MRC} = 4e^{-6}$, $lr_{SMXM} = 4e^{-6}$, were used[4]. For MedMentions, SMXM uses $lr_{SMXM} = 7e^{-6}$. Interestingly, these optimal learning rates are lower than recommended in the original paper (between $2e^{-5}$ and $5e^{-5}$) (Devlin et al., 2019). The batch size was set to 20 for BEM and MRC[5]. For SMXM we use a batch size of 8 for `OntoNotes-ZS`[6], which was the largest batch size that fitted into the GPU since SMXM accumulates the gradients when fed as input for X-ENC. For `MedMentions-ZS`, we had to further reduce the batch size to 5. The masking probability p was set to 0.7 for SMXM[7]. A model was trained for a maximum of 3 epochs. For MedMentions the class weight q for the negative class is set to 0.1 and for OntoNotes to 0.01.

The maximum sequence length to input to BERT was restricted to 300, with a maximum of 150 tokens for the description itself. Due to the restrictions to GPU memory, we used a sequence length of 200 when training SMXM on `MedMentions-ZS`, with 100 tokens being the maximum length of a type description.

For training, all models further use i) an early-stop scheduler to stop the training after no improvement on the validation F_1 score was detected for three consecutive steps, ii) a scheduler that reduces the learning rate linearly over the number of trained steps until it reaches zero with the last training step, similarly to the one described in (Devlin et al., 2019). The Bert entailment model used for BEM was trained on MNLI with the default hyperparameters used in (Devlin et al., 2019): $lr = 2e^{-5}$, epochs= 3, and we used a batch size of $= 100$.

Span-level scores are computed using the Seqeval library[8].

6.2.1 Details on the baselines

We explored the model of Li et al. (2020) by both re-implementing their paper and also by using our zero-shot dataset on their publicly available repository[9]. Several parameter settings were explored, with additional sanity checks. Training was stopped after ten epochs. Yet, in both attempts the macro F$_1$ for the best model stayed only barely above zero on `OntoNotes-ZS`. Regarding the mentioned causes for the low zero-shot NERC

[4] Exploration range: [$1e^{-7}$,$5e^{-5}$]

[5] Exploration range: [7, 25]

[6] Exploration range: [5, 8]

[7] Exploration range: [0.3 0.7]

[8] https://pypi.org/project/seqeval/

[9] https://github.com/ShannonAI/mrc-for-flat-nested-ner

scores of MRC for NERC, we have additionally noticed that in a fully supervised setting class-level scores when using the aforementioned repository are very high for very frequently observed classes, but comparably low for rare classes, indicating that substantial supervision is required to perform well, as the model is very sensitive to prediction errors as argued in the paper.

6.2.2 Example Type descriptions

Class	Description
FAC	Names of man-made structures: infrastructure (streets, bridges), buildings, monuments, etc. belong to this type. Buildings that are referred to using the name of the company or organization that uses them should be marked as FAC when they refer to the physical structure of the building itself, usually in a locative way: "I'm reporting live from right outside [Massachusetts General Hospital]"
LOC	Names of geographical locations other than GPEs. These include mountain ranges, coasts, borders, planets, geo-coordinates, bodies of water. Also included in this category are named regions such as the Middle East, areas, neighborhoods, continents and regions of continents. Do NOT mark deictics or other non-proper nouns: herea, there, everywhere, etc.
WORK OF ART	Titles of books, songs, television programs and other creations. Also includes awards. These are usually surrounded by quotation marks in the article (though the quotations are not included in the annotation). Newspaper headlines should only be marked if they are referential. In other words the headline of the article being annotated should not be marked but if in the body of the text here is a reference to an article, then it is markable as a work of art.
LAW	Any document that has been made into a law, including named treaties and sections and chapters of named legal documents.
EVENT	Named hurricanes, battles, wars, sports events, attacks. Metonymic mentions (marked with a $\sim$) of the date or location of an event, or of the organization(s) involved, are included).

Table 10: Snippet of OntoNotes NERC annotation guidelines. All rights of these descriptions belong to (Pradhan et al., 2013) and Ratheon BBN Technologies.