MINER: Improving Out-of-Vocabulary Named Entity Recognition from an Information Theoretic Perspective

Anonymous ACL submission

Abstract

NER model has achieved promising performance on standard NER benchmarks. However, recent studies show that previous approaches may over-rely on entity mention information, resulting in poor performance on out-of-vocabulary(OOV) entity recognition. In this work, we propose MINER, a novel NER learning framework, to remedy this issue from an information-theoretic perspective. The proposed approach contains two mutual information based training objectives: i) generalizing information maximization, which enhances representation via deep understanding of context and entity surface forms; ii) superfluous information minimization, which discourages representation from rotate memorizing entity names or exploiting biased cues in data. Experiments on various settings and datasets demonstrate that it achieves better performance in predicting OOV entities.

1 Introduction

Named Entity Recognition(NER) aims to identify and classify entity mentions from unstructured text, e.g., extracting location mention "Berlin" from sentence "Berlin is wonderful in the winter". NER is a key component in information retrieval (Tan et al., 2021), question answering (Min et al., 2021), dialog systems (Wang et al., 2020), etc. Traditional NER models are feature-engineering and machine learning based (Zhou and Su, 2002; Takeuchi and Collier, 2002; Agerri and Rigau, 2016). Benefiting from the development of deep learning, neural-network-based NER models have achieved state-of-the-art results on several public benchmarks (Lample et al., 2016; Peters et al., 2018; Devlin et al., 2018; Yamada et al., 2020; Yan et al., 2021).

Recent studies (Lin et al., 2020; Agarwal et al., 2021) show that, context does influence predictions of NER models, but the main factor driving high performance is learning the named tokens themselves. Consequently, NER models underperform when predicting entities that have not been seen during training (Fu et al., 2020; Lin et al., 2020), which is referred to as an Out-of-Vocabulary(OOV) problem.

	Precision			Recall		
	InDict	OutDict	**Diff**	InDict	OutDict	**Diff**
PER	88.03	75.40	14%	92.90	85.20	8%
ORG	73.51	72.77	1%	81.93	76.56	7%
GPE	79.55	78.21	2%	85.37	77.22	10%
FAC	65.91	65.67	0%	86.05	65.67	24%
ALL	83.37	71.97	**12%**	89.08	79.11	**11%**

Table 1: The comparison between the in-dictionary and out-of-dictionary parts of the CoNLL 2003 baseline (Lin et al., 2020), which was tested on Bert-CRF. It is obvious that the performance gap between InDict and OutDict is significantly large.

There are three classical strategies to alleviate the OOV problem: external knowledge, OOV word embedding, and contextualized embedding. The first one is to introduce additional features, e.g., entity lexicons (Zhang and Yang, 2018), part-of-speech tags (Li et al., 2018), which alleviates the model's dependence on word embeddings. However, the external knowledge is not always easy to obtain. The second strategy is to get a better OOV word embedding (Peng et al., 2019; Fukuda et al., 2020). The strategy is learning a static OOV embedding representation, but not directly utilize the context. Last one is fine-tune pre-trained models, e.g., ELMo (Peters et al., 2018), BERT (Devlin et al., 2018), which provide contextualized word representations. Unfortunately, Yan et al. (2021) shows that the higher performance of pre-trained models could be the results of learning the subword structure better.

How do we make the model focus on contextual information to tackle the OOV problem? Motivated by the information bottleneck principle (Tishby et al., 2000), we propose a novel learning framework - Mutual Information based Named Entity

Recognition (MINER). The proposed method provides an information-theoretic perspective to the OOV problem by training an encoder to minimize task-irrelevant nuisances while keeping predictive information.

Specifically, MINER contains two mutual information based learning objectives: i) generalizing information maximization, which aims to maximize the mutual information between representations and well-generalizing features, i.e., context and entity surface forms; ii) superfluous information minimization, which prevents the model from rote memorizing the entity names or exploiting biased cus via eliminating entity name information.

Our main contributions are summarized as follows:

1. We propose a novel learning framework, i.e., MINER, from an information theory perspective, aiming to improve the robustness of entity changes by eliminating entity-specific and maximize well-generalizing information.

2. We show its effectiveness on several settings and benchmarks, and suggest that MINER is a reliable approach to better OOV entity recognition.

2 Background

In this section, we highlight the information bottleneck principle. Subsequently, the analysis of possible issues when applying it to OOV entity recognition was provided. Furthermore, we review related techniques in deriving our framework.

Information Bottleneck (IB) principle originated in information theory, and provides a theoretical framework for analyzing deep neural networks. It formulates the goal of representation learning as an information trade-off between representation compression and predictive power. Given the input dataset (X,Y), it seeks to learn the internal representation Z of some intermediate layers by:

$$L_{IB} = -I(Z;Y) + \beta \, * \, I(Z;X),$$

where I represents the mutual information(MI), a measure of the mutual dependence between the two variables. The trade-off between the two MI terms is controlled by a Lagrange multiplier β. A low loss indicates that representation Z does not keep too much information from X while still retaining enough information to predict Y.

Section 5 suggests that directly applying IB to NER can not bring obvious improvement. We argue that IB cannot guarantee well-generalizing representation.

On the one hand, it has been shown that it is challenging to find a trade-off between high compression and high predictive power (Tishby et al., 2000; Wang et al., 2019; Piran et al., 2020). When compressing task-irrelevant nuisances, however, useful information will inevitably be left out. On the other hand, it is unclear for the IB principle which parts of features are well-generalizing and which are not, as we usually train a classifier to solely maximize accuracy. Consequently, neural networks tend to use any accessible signal to do so (Ilyas et al., 2019), which is referred to as a *shotcut learning* problem (Geirhos et al., 2020). For training sets with limited size, it may be easier for neural networks to memorize entity names rather than to classify them by context and common entity features (Agarwal et al., 2021). In Section 4, we demonstrate how we extend BN to the NER task and address these issues.

3 Model Architecture

In recent years, NER systems have undergone a paradigm shift from sequence labeling, which formulates NER as a token-level tagging task (Chiu and Nichols, 2016; Akbik et al., 2018; Yan et al., 2019), to span prediction (SpanNER), which regards NER as a span-level classification task (Mengge et al., 2020; Yamada et al., 2020; Fu et al., 2021). We choose SpanNER as base architecture for two reasons:

1) SpanNER can yield the whole span representation, which can be directly used for optimize information. 2) compared with sequence labeling, SpanNER does better in sentences with more OOV words (Fu et al., 2021).

Overall, SpanNER consists of three major modules: token representation layer, span representation layer, and span classification layer. Besides, our method inserts a bottleneck layer to the architecture for information optimization.

3.1 Token Representation Layer

Let $X = \{x_1, x_2, \cdots, x_n\}$ represents the input sentence, thus, the token representation h_i is as follows:

$$u_1, \cdots, u_n = Embedding(x_1, \cdots, x_n) \quad (1)$$

$$h_1, \cdots, h_n = Encoder(u_1, \cdots, u_n) \quad (2)$$

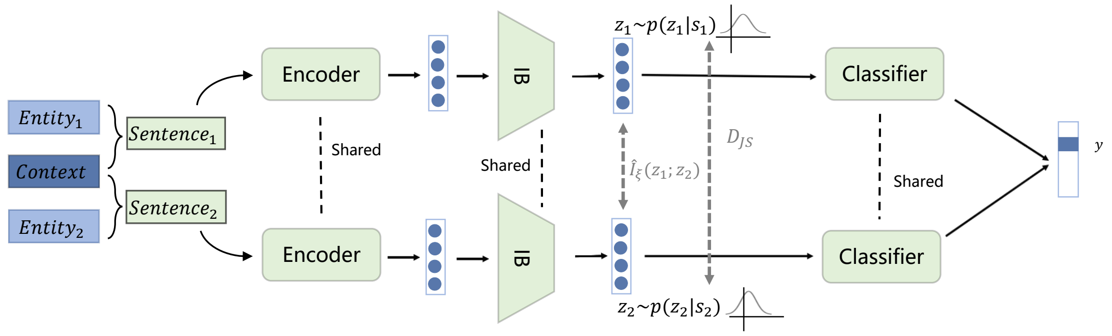

Figure 1: Visualization of MINER, where $Sentence_1$ and $Sentence_2$ share the same context and entity labels, while their entity name is different. s_1 and s_2 represents the entity representation of $Sentence_1$ and $Sentence_2$, respectively. z_1 and z_2 are compressed representations which are sampled by $p(z_1|s_1)$ and $p(z_2|s_2)$, respectively, which are implemented by information bottleneck(IB) layer. Our method add two additional learning objectives to basic architecture. The first one is to maximize the mutual information, i.e., $I_\epsilon(z_1; z_2)$, to enhance context information and entity surface form information of z_1 and z_2. The second objective is minimize the Jensen-Shannon divergence which represents the mutual information between z_1 and z_2, which aims to eliminating task irrelevant nuisances.

where $Embedding()$ is the non-contextualized word embeddings, e.g., Glove (Pennington et al., 2014) or contextualized word embeddings, e.g., ELMo (Peters et al., 2018), BERT (Devlin et al., 2018). $Encoder()$ can be any network structures with context encoding function, e.g., LSTM (Hochreiter and Schmidhuber, 1997), CNN (LeCun et al., 1995), transformer (Vaswani et al., 2017), and so on.

3.2 Span Representation Layer

For all possible spans $S = \{s_1, s_2, \cdots, s_m\}$ of sentence X, we re-assign a label $y \in Y$ for each span. Take "Berlin is wonderful" as an example, its possible spans and labels are $\{(1,1),(1,2),(1,3),(2,2),(2,3),(3,3)\}$ and $\{LOC, O, O, O, O, O\}$, respectively.

Given the start index b_i and end index e_i, the representation of span s_i can be calculated by two parts: boundary embedding and span length embedding.

Boundary embedding: This part is calculated by concatenating the start and end tokens' representation $t_i^b = [h_{b_i}; h_{e_i}]$.

Span length embedding: In order to introduce the length feature, we additionally provide the length embedding t_i^l, which can be obtained by a learnable look-up table.

Finally, the span representation can be obtained as: $t_i = [t_i^b; t_i^l]$.

3.3 Information Bottleneck Layer

In order to optimize the information in the span representation, our method additionally adds an information bottleneck layer of the form:

$$\mathcal{N}\left(z \mid f_e^{\mu}(t), f_e^{\Sigma}(x)\right) \tag{3}$$

where f_e is an MLP which outputs both the K-dimensional mean μ of z as well as the $K * K$ covariance matrix Σ.

3.3.1 Span Classification Layer

Once the information bottleneck layer is finished, z_i is fed into the classifier to obtain the probability of its label y_i. Based on the probability, the basic loss function can be calculated as follows:

$$L_{base} = -\frac{score(z_i, y_i)}{\sum_{y' \in Y} score(z_i, y')}, \tag{4}$$

where $score()$ is a function that measures the compatibility between a specified label and a span representation:

$$score(z_i, y^k) = exp(z_i^T y^k), \tag{5}$$

where y^k is a learnable representation of class k.

Heuristic Decoding A heuristic decoding solution for the flat NER is provided to avoid the prediction of over-lapped spans. For those overlapped spans, we keep the span with the highest prediction probability and drop the others.

It's worth noting that our method is flexible and can be used with any other NER model based on span classification. In next section, we will introduce two additional objectives to tackle the OOV problem of NER.

4 MI-based objectives

Motivated by IB (Tishby et al., 2000; Federici et al., 2020), we can subdividing $I(X;Z)$ into two components by using the chain rule of mutual information(MI):

$$I(X;Z) = \underbrace{I(Y;Z)}_{predictive} + \underbrace{I(X;Z|Y)}_{superfluous}, \quad (6)$$

The first term determines how much information about Y is accessible from Z. While the second term, conditional mutual information term $I(X;Z|Y)$, denotes the information in Z that is not predictive of Y.

For NER, which parts of the information retrieved from input are useful and which are redundant?

From human intuition, **text context** should be the main predictive information for NER. For example, "The CEO of X resigned", the type of X in each of these contexts should always be "ORG". Besides, **entity mentions** also provide much information for entity recognition. For example, nearly all person names capitalize the first letter and follow the "firstName lastName" or "lastName firstName" patterns. However, **entity name** is not a well-generalizing features. By simply memorizing the fact which span is an entity, it may be possible for it to fit the training set, but it is impossible to predict entities that have never been seen before.

We convert the targets of Eq. (6) into a form that is easier to solve via a contrastive strategy. Specifically, consider x_1 and x_2 are two contrastive samples of similar context, and contains different entity mentions of the same entity category, i.e., s_1 and s_2, respectively. Assuming both x_1 and x_2 are both **sufficient** for inferring label y. The mutual information between x_1 and z_1 can be factorized to two parts.

$$I(x_1;z_1) = \underbrace{I(z_1;x_2)}_{consistent} + \underbrace{I(x_1;z_1|x_2)}_{specific}, \quad (7)$$

where z_1 and z_2 are span representations of s_1 and s_2, respectively, $I(z_1;x_2)$ denotes the information that isn't entity-specific. And $I(x_1;z_1|x_2)$ represents the information in z_1 which is unique to x_1 but is not predictable by sentence x_2, i.e., entity-specific information.

Thus any representation z containing all information shared from both sentences would also contain the necessary label information, and sentence-specific information is superfluous. So Eq. (6) can be approximated by Eq. (7) by:

$$maximize\, I(z_1;y) \sim I(z_1;x_2), \quad (8)$$

$$minimize\, I(x_1;z_1|y) \sim I(x_1;z_1|x_2), \quad (9)$$

The target of Eq. (8) is defined as **generalizing information** maximization. We proved that $I(z_1;z_2)$ is a lower bound of $I(z_1;x_2)$(proof could be found in appendix 7). InfoNCE (Oord et al., 2018) was used as a lower bound on MI and can be used to approximate $I(z_1;z_2)$. Subsequently, it can be optimized by:

$$L_{gi} = -\mathbb{E}_p \left[g_w(z_1,z_2) - \mathbb{E}_{p'} \log \sum_{z'} \exp g_w(z_1,z') \right], \quad (10)$$

where $g_w(\cdot,\cdot)$ is a compatible score function approximated by a neural network, z_2 are the positive entity representations from the joint distribution p of original sample and corresponding generated sample, z' are the negative entity representations drawn from the joint distribution of original sample and other original sample.

The target of Eq. (9) is defined as **superfluous information** minimization. To restrict this term, we can minimize an upper bound of $I(x_1;z_1|x_2)$ (proofs could be found in appendix 7) as follows:

$$L_{si} = \mathbb{E}_{x_1,x_2}\mathbb{E}_{z_1,z_2}\left[D_{JS}[P_{z_1}||P_{z_2}]\right], \quad (11)$$

where D_{JS} represents Jensen-Shannon divergence. In practice, Eq. (11) encourage z to be invariant to entity changes.

4.1 Contrastive sample generation

It is difficult to obtain samples with similar contexts but different entity words. We generate contrastive samples by the mention replacement mechanism(Dai and Adel, 2020). For each mention in the sentence, we replace it by another mention from the original training set, which has the same entity type. The corresponding span label can be changed accordingly. For example, "LOC" mention "Berlin" in sentence "Berlin is wonderful in the winter" is replaced by "Iceland".

Datasets	sents	entities	OOV Rate
WNUT2017	1286	947	1.00
TwitterNER	3257	3990	0.62
BioNER	3856	4344	0.77
Conll2003-Typos	2676	4130	0.71
Conll2003-OOV	3684	5648	0.96

Table 2: Number of OOV entities in the test sets.

4.2 Training

Combine Eq. (4), (10), and (11), we can get the following objective function, which try to minimize:

$$L = L_{base} + \gamma * L_{gi} + \beta * L_{si}, \qquad (12)$$

where γ and β are the weights of the generalizing information loss and superfluous information loss, respectively.

5 Experiment

In this section, we verified the performance of the proposed method on five OOV datasets, and compared it with other methods. In addition, We tested the universality of the proposed method in various pre-trained models.

5.1 Datasets and Metrics

Datasets We performed experiments on:

1. WNUT2017 (Derczynski et al., 2017), a dataset focus on unusual, previous-unseen entities in training data, and is collected from social media.

2. TwitterNER (Zhang et al., 2018), an English NER dataset created from Tweets.

3. BioNER (Kim et al., 2004), the JNLPBA 2004 Bio-NER dataset focus on technical terms in the biology domain.

4. Conll03-Typos (Wang et al., 2021), which is generated from Conll2003 (Sang and De Meulder, 2003). The entities in the test set is replaced by typos version(character modify, insert, and delete operation).

5. Conll03-OOV (Wang et al., 2021), which is generated from Conll2003 (Sang and De Meulder, 2003). The entities in the test set is replaced by another out-of-vocabulary entity in test set.

Table 2 reports the static results of the OOV problem on the test sets of each dataset. As shown in the table, the test set of these data sets comprises a substantial amount of OOV entities.

Metrics We measured the entity-level micro average F1 score on the test set to compare the results of different models.

5.2 Baseline methods

Li et al. (2020) share the same intuition, enrich word representations with contextual, with us. However, the work is neither open source nor reported on the same data set, so this method is not compared with MINER. We compare our method with baselines as follows:

- SpanNER (Fu et al., 2021), which is trained by original SpanNER framework, means without any constraint and extra data processing.

- Vanilla information bottleneck(VaniIB), this method employs the original information bottleneck constraint to the SpanNER, which is optimized based on Alemi et al. (2016). Compared with our method, it directly compresses all the information from the input.

- Dai and Adel (2020) (DataAug) , which trains model with data augmentation strategy, while keeps the same model architecture of SpanNER. This model is trained by 1:1 original training set and entity replacement training set, which keeps the same input as the proposed method.

- Shahzad et al. (2021) (InferNER), the method focus on word-, character-, and sentence-level information for NER in short-text, without recurring to external sources. In addition, it is able to incorporate visual information and introduce an attention component which computes attention weight probabilities over textual and text-relevant visual contexts separately.

- Li et al. (2021) (MIN), which utilizes both segment-level information and word-level dependencies, and incorporates an interaction mechanism to support information sharing between boundary detection and type prediction to enhance the performance for the NER task.

- Fukuda et al. (2020) (CoFEE), which refer to pre-trained word embeddings for known

Methods	**WNUT2017**	**JNLPBA**	**TwitterNER**	**CoNLL 2003**	
				Typos	OOV
VaniIB	51.55	73.22	71.00	83.49	70.12
DataAug	52.29	75.85	73.69	81.73	69.6
InferNER	50.52	-	74.17	-	-
MIN	49.93	**77.97**	-	-	-
CoFEE	39.1	-	69.5	-	-
MAML	24.19	76.36	-	-	-
SA-NER	50.36	-	-	-	-
SpanNER (Bert large)	51.83	73.78	71.57	81.83	64.43
SpanNER (Roberta large)	51.65	74.49	71.7	82.85	64.7
SpanNER (AlBert large)	49.13	71.08	70.33	82.49	64.12
Our Method (Bert large)	54.52	77.03	75.26	87.09	78.03
Our Method (Roberta large)	**54.86**	76.43	**75.38**	**87.57**	**79.15**
Our Method (Albert large)	51.94	75.23	72.67	86.53	77.95

Table 3: Performance of the proposed method compared with state-of-the-arts.

words with similar surfaces to target OOV words.

- Nie et al. (2020) (SA-NER), which utilize semantic enhancement methods to reduce the negative impact of data sparsity problems. Specifically, the method obtains the augmented semantic information from a large-scale corpus, and propose an attentive semantic augmentation module and a gate module to encode and aggregate such information, respectively.

To verify the universality of our method, we measured its performance in various pre-trained models, i.e., Bert (Devlin et al., 2018), Roberta (Liu et al., 2019), Albert (Lan et al., 2019).

5.3 Implementation Details

Bert-large released by Devlin et al. (2018) is selected as our base encoder. The learning rate is set to 5e-5, and the dropout is set to 0.2. The output dim of information bottleneck layer is 50. In order to make a trade-off for the performance and efficiency, on the one hand, we truncate the part of the sentence whose tokens exceeds 128. On the other hand, we count the length distribution of entity length in different datasets, and finally chose 4 as the maximum enumerated entity length. The values of β and γ are different for different data sets. Empirically, 1e-5 for β and 0.01 for γ can get promised results. The model is trained in a NVIDIA GeForce RTX 2080Ti GPU. Checkpoints with top-3 performance are finally evaluated on the test set to report averaged results.

5.4 Main Results

We demonstrate the effectiveness of MINER against other state-of-the-art models. As shown in table 3, we have the following observations and analysis:

1) Our baseline model, i.e., SpanNER, does a good job at predicting OOV entities. Compared to sequence labeling, the the span classification could model the relation of entity tokens directly;2) The performance of SpanNER is further boosted with our proposed approach, which proved the effectiveness of our method. As shown in table, we almost beats all other SOTA methods without any external resource;3) Compared to *Typos* data transformation, it is more difficult for model to predict *OOV* words. To pre-trained model, typos word may not appear in training set, but they share most subwords with the original token. MOreover, the subword of OOV entity may be rare; 4) It seems that the traditional information bottleneck will not greatly improve the OOV prediction ability of the model. We argue that the traditional information bottlenecks will indiscriminately compress the information in the representation, leading to underfitting; 5Our model has significantly improved the performance of the model on the entity perturbed methods of typos and OOV, proving that our method can not only improve the generalization ability of OOV words in the field, but also significantly improve the robustness in the face of noise; 6) It is clearly

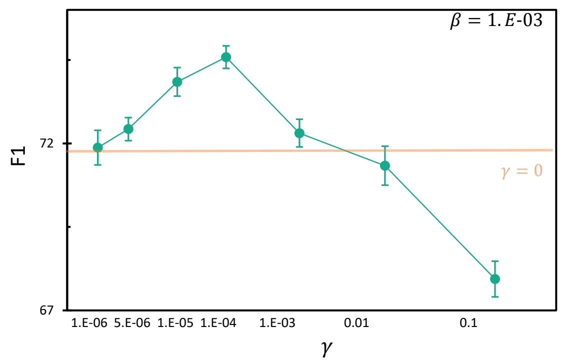

Figure 2: Illustration of f1 score in different γ values. We fix $\beta = 1e03$, and the orange line is f1 score when $\beta = 0$.

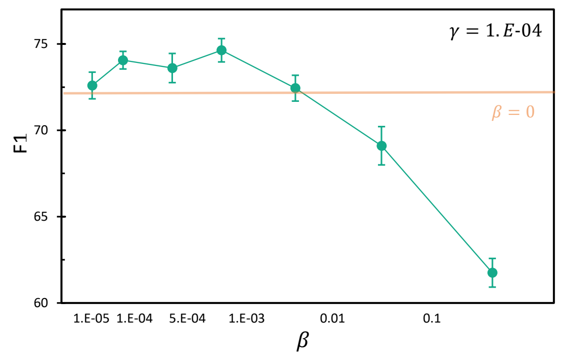

Figure 3: Illustration of f1 score in different β values. We fix $\gamma = 1e04$, and the orange line is f1 score when $\beta = 0$.

Dataset	OOV	MI	F1
WNUT 2017	-	-	51.83
	✓	-	52.57
	-	✓	53.91
	✓	✓	**54.52**
JNLPBA	-	-	73.78
	✓	-	75.23
	-	✓	74.22
	✓	✓	**77.03**
Twitter-NER	-	-	71.57
	✓	-	73.78
	-	✓	73.32
	✓	✓	**75.26**

Table 4: Ablation study results on three datasets.

that our proposed method is universal and can further improve OOV prediction performance for different embedding models. As we get stably improvements on Bert, Roberta, and Albert.

5.5 Ablation Study

We also perform ablation studies to validate the effectiveness of each part in MINER. Table 4 demonstrates the results of different settings for the proposed training strategy equipped with BERT. After only adding the Lgi loss for enhance context and entity surface form information, we find that the results are better than the original PLMs. Similar phenomenon occured in L_{si}, too. It reflects that both Lgi and L_{si} are beneficial to improve the generalizing ability on OOV entities. Moreover, the results on three dataset are significantly improved by add both Lgi and L_{si} learning objectives. It means Lgi and L_{si} can boost each over, which proves that our method enhances representation via deep understanding of context and entity surface forms and discourages representation from rotate memorizing entity names or exploiting biased cues in data.

5.6 Sensitivity Analysis of β and γ

To show the different influence of our proposed training objectives L_{gi} and L_{si}, we conduct sensitivity analysis of the coefficient β and γ. Figure 2 shows the performance change under different settings of the two coefficients. The yellow line denotes ablation results without the corresponding loss functions (with β=0 or γ=0). From Figure 2 we can observe that the performance is significantly enhanced with a small rate of β or γ, where the best performance is achieved when β=1e-3 and γ=1e-4, respectively. It probes the effectiveness of our proposed training objectives that enhances representation via deep understanding of context and entity surface forms and discourages representation from rotate memorizing entity names or exploiting biased cues in data. When the coefficient rate increases continuously, the performance shows a decline trend, which means the over-constraint of L_{gi} or L_{si} will hurt the generalizing ability of predicting the OOV entities.

5.7 Interpretable Analysis

The above experiments show the promising performance of MINER on predicting the unseen entities. To further investigate which part of the sentence MINER focuses on, we visualize the attention weights over entities and contexts. We demonstrate an example in Figure 4 , where is selected from TwitterNER. The attention score is calculated by averaging the attention weight of the 0th layer of BERT. Take the attention weights of entity "State Street" as a example, it is obvious that baseline model, i.e., SpanNER, focus on entity

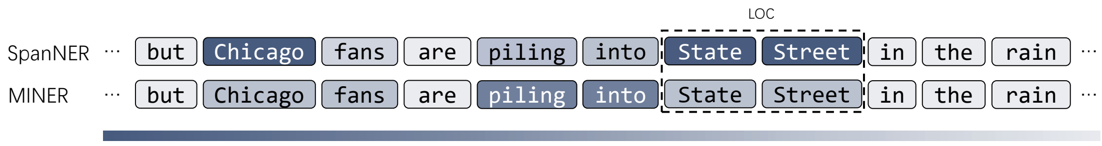

Figure 4: Visualization of attention weights over entities and context.

words themselves. While the scores of our model is more average, means that our method concern more context information.

6 Related Work

6.1 External Knowledge

This group of methods makes it easier to predict OOV entities using external knowledge. Zhang and Yang (2018) Use a dictionary to list numerous entity mentions. It is possible to get stronger "look-up" models by integrating dictionary information, but there is no guarantee that entities outside the training set and vocabulary will be correctly identified. To diminish the model's dependency on OOV embedding, Li et al. (2018) introduces part-of-speech tags. External resources are not always available, which is a limitation of this strategy.

6.2 OOV word Embedding

The OOV problem can be alleviated by improving the OOV word embedding. The character ngram of each word is used by Bojanowski et al. (2017) to represent the OOV word embedding. Pinter et al. (2017) captures morphological features using character-level RNN. Another technique is to first match the OOV words with the words that have been seen in training, then replace the OOV words' embedding with the seen words' embedding. Peng et al. (2019) trains a student network to predict the closest word representation to the OOV term. Fukuda et al. (2020) referring to pre-trained word embeddings for known words with similar surfaces to target OOV words. This kind of method is learning a static OOV embedding representation, and does not directly utilize the context.

6.3 Contextualized Embedding

Contextual information is used to enhance the representation of OOV words in this strategy. (Hu et al., 2019) formulate the OOV problem as a K-shot regression problem and learns to predict the OOV embedding by aggregating only K contexts and morphological features. Pre-trained models contextualized word embbeddings via pretraining on large background corpora. Furthermore, contextualized word embeddings can be provided by the pre-trained models which are pre-trained on large background corpora (Peters et al., 2018; Devlin et al., 2018; Liu et al., 2019). Yan et al. (2021) shows that BERT are not always better at capturing context as compared to Gloe-based BiLSTM-CRFs. Their higher performance could be the results of learning the subword structure better.

7 Conclusion

Based on the recent studies of NER, we analyzed how to improve the OOV entity recognition. In this work, we propose a novel and flexible learning framework - MINER, to tackle OOV entities recognition issue from an information-theoretic perspective. On the one hand, this method can enhance the context information of the output of the encoder. On the other hand, it can safely eliminate task-irrelevant nuisances and prevents the model from rote memorizing the entities. Specifically, the proposed approach contains two mutual information based training objectives: generalizing information maximization, and superfluous information minimization. Experiments on various datasets demonstrate that MINER achieves much better performance in predicting out-of-vocabulary entities.

References

Oshin Agarwal, Yinfei Yang, Byron C Wallace, and Ani Nenkova. 2021. Interpretability analysis for named entity recognition to understand system predictions and how they can improve. *Computational Linguistics*, 47(1):117–140.

Rodrigo Agerri and German Rigau. 2016. Robust multilingual named entity recognition with shallow semi-supervised features. *Artificial Intelligence*, 238:63–82.

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th international conference on computational linguistics*, pages 1638–1649.

Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. 2016. Deep variational information bottleneck. *arXiv preprint arXiv:1612.00410*.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Jason PC Chiu and Eric Nichols. 2016. Named entity recognition with bidirectional lstm-cnns. *Transactions of the Association for Computational Linguistics*, 4:357–370.

Xiang Dai and Heike Adel. 2020. An analysis of simple data augmentation for named entity recognition. *arXiv preprint arXiv:2010.11683*.

Leon Derczynski, Eric Nichols, Marieke van Erp, and Nut Limsopatham. 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.

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

Marco Federici, Anjan Dutta, Patrick Forré, Nate Kushman, and Zeynep Akata. 2020. Learning robust representations via multi-view information bottleneck. *arXiv preprint arXiv:2002.07017*.

Jinlan Fu, Xuanjing Huang, and Pengfei Liu. 2021. SpanNER: Named entity re-/recognition as span prediction. 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 7183–7195, Online. Association for Computational Linguistics.

Jinlan Fu, Pengfei Liu, and Qi Zhang. 2020. Rethinking generalization of neural models: A named entity recognition case study. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7732–7739.

Nobukazu Fukuda, Naoki Yoshinaga, and Masaru Kitsuregawa. 2020. Robust Backed-off Estimation of Out-of-Vocabulary Embeddings. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4827–4838, Online. Association for Computational Linguistics.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. 2020. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Ziniu Hu, Ting Chen, Kai-Wei Chang, and Yizhou Sun. 2019. Few-shot representation learning for out-of-vocabulary words. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4102–4112, Florence, Italy. Association for Computational Linguistics.

Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. 2019. Adversarial examples are not bugs, they are features. *arXiv preprint arXiv:1905.02175*.

Jin-Dong Kim, Tomoko Ohta, Yoshimasa Tsuruoka, Yuka Tateisi, and Nigel Collier. 2004. Introduction to the bio-entity recognition task at jnlpba. In *Proceedings of the international joint workshop on natural language processing in biomedicine and its applications*, pages 70–75. Citeseer.

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, California. Association for Computational Linguistics.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Yann LeCun, Yoshua Bengio, et al. 1995. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995.

Changliang Li, Liang Li, and Ji Qi. 2018. A self-attentive model with gate mechanism for spoken language understanding. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3824–3833, Brussels, Belgium. Association for Computational Linguistics.

Fei Li, Zheng Wang, Siu Cheung Hui, Lejian Liao, Dandan Song, Jing Xu, Guoxiu He, and Meihuizi Jia. 2021. Modularized interaction network for named entity recognition. 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 200–209, Online. Association for Computational Linguistics.

Yangming Li, Han Li, Kaisheng Yao, and Xiaolong Li. 2020. Handling rare entities for neural sequence labeling. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6441–6451, Online. Association for Computational Linguistics.

Hongyu Lin, Yaojie Lu, Jialong Tang, Xianpei Han, Le Sun, Zhicheng Wei, and Nicholas Jing Yuan. 2020.

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. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Xue Mengge, Bowen Yu, Zhenyu Zhang, Tingwen Liu, Yue Zhang, and Bin Wang. 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. Association for Computational Linguistics.

Sewon Min, Kenton Lee, Ming-Wei Chang, Kristina Toutanova, and Hannaneh Hajishirzi. 2021. Joint passage ranking for diverse multi-answer retrieval. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6997–7008, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yuyang Nie, Yuanhe Tian, Xiang Wan, Yan Song, and Bo Dai. 2020. Named entity recognition for social media texts with semantic augmentation. *arXiv preprint arXiv:2010.15458*.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Minlong Peng, Qi Zhang, Xiaoyu Xing, Tao Gui, Jinlan Fu, and Xuanjing Huang. 2019. Learning task-specific representation for novel words in sequence labeling. *arXiv preprint arXiv:1905.12277*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Yuval Pinter, Robert Guthrie, and Jacob Eisenstein. 2017. Mimicking word embeddings using subword RNNs. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 102–112, Copenhagen, Denmark. Association for Computational Linguistics.

Zoe Piran, Ravid Shwartz-Ziv, and Naftali Tishby. 2020. The dual information bottleneck. *arXiv preprint arXiv:2006.04641*.

Erik Tjong Kim Sang and Fien De Meulder. 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.

Moemmur Shahzad, Ayesha Amin, Diego Esteves, and Axel-Cyrille Ngonga Ngomo. 2021. Inferner: an attentive model leveraging the sentence-level information for named entity recognition in microblogs. In *The International FLAIRS Conference Proceedings*, volume 34.

Koichi Takeuchi and Nigel Collier. 2002. Use of support vector machines in extended named entity recognition. In *COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002)*.

Xingwei Tan, Gabriele Pergola, and Yulan He. 2021. Extracting event temporal relations via hyperbolic geometry. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8065–8077, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Naftali Tishby, Fernando C Pereira, and William Bialek. 2000. The information bottleneck method. *arXiv preprint physics/0004057*.

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*, pages 5998–6008.

Kai Wang, Junfeng Tian, Rui Wang, Xiaojun Quan, and Jianxing Yu. 2020. Multi-domain dialogue acts and response co-generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7125–7134, Online. Association for Computational Linguistics.

Qi Wang, Claire Boudreau, Qixing Luo, Pang-Ning Tan, and Jiayu Zhou. 2019. Deep multi-view information bottleneck. In *Proceedings of the 2019 SIAM International Conference on Data Mining*, pages 37–45. SIAM.

Xiao Wang, Qin Liu, Tao Gui, Qi Zhang, Yicheng Zou, Xin Zhou, Jiacheng Ye, Yongxin Zhang, Rui Zheng, Zexiong Pang, et al. 2021. Textflint: Unified multilingual robustness evaluation toolkit for natural language processing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 347–355.

Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. LUKE: Deep contextualized entity representations with entity-aware self-attention. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6442–6454, Online. Association for Computational Linguistics.

Hang Yan, Bocao Deng, Xiaonan Li, and Xipeng Qiu. 2019. Tener: adapting transformer encoder for named entity recognition. *arXiv preprint arXiv:1911.04474*.

Hang Yan, Tao Gui, Junqi Dai, Qipeng Guo, Zheng Zhang, and Xipeng Qiu. 2021. A unified generative framework for various NER subtasks. 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 5808–5822, Online. Association for Computational Linguistics.

Qi Zhang, Jinlan Fu, Xiaoyu Liu, and Xuanjing Huang. 2018. Adaptive co-attention network for named entity recognition in tweets. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Yue Zhang and Jie Yang. 2018. Chinese NER using lattice LSTM. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1554–1564, Melbourne, Australia. Association for Computational Linguistics.

GuoDong Zhou and Jian Su. 2002. Named entity recognition using an hmm-based chunk tagger. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 473–480.

A Appendix

This section provides the proof of generalizing information maximization, i.e., Eq. (8). Consider x_1 and x_2 are two contrastive samples of similar context, and contains different entity mentions of the same entity category, i.e., s_1 and s_2, respectively.

$$\begin{aligned} I(z_1; x_2) =& I(z_1; x_2 z_2) - I(z_1; z_2|x_2) \\ =& I(z_1; x_2 z_2) \\ =& I(z_1; z_2) + I(z_1; x_2|z_2) \\ \geq& I(z_1; z_2) \end{aligned} \tag{13}$$

B Appendix

This section provides the proof of superfluous information minimization, i.e. Eq. (9).

$$\begin{aligned} &\mathrm{I}(\mathrm{x}_1; z_1|x_2) \\ &= E_{x_1,x_2\sim p(x_1,x_2)} E_{z\sim p(z_1|v_1)} \log \frac{p(x_1,z_1|x_2)}{p(x_1|x_2)p(z_1|x_2)} \\ &= E_{x_1,x_2\sim p(x_1,x_2)} E_{z\sim p(z_1|v_1)} \log \frac{p(z_1|x_1)p(x_1|x_2)}{p(x_1|x_2)p(z_1|x_2)} \\ &= E_{x_1,x_2\sim p(x_1,x_2)} E_{z\sim p(z_1|v_1)} \log \frac{p(z_1|x_1)}{p(z_1|x_2)} \\ &= E_{x_1,x_2\sim p(x_1,x_2)} E_{z\sim p(z_1|v_1)} \log \frac{p(z_1|x_1)p(z_2|x_2)}{p(z_2|x_2)p(z_1|x_2)} \\ &= D_{KL}(p(z_1|x_1)||p(z_2|x_2)) \\ &- D_{KL}(p(z_1|x_2)||p(z_2|x_2)) \\ &\leq D_{KL}(p(z_1|x_1)||p(z_2|x_2)) (14) \end{aligned}$$