DATA ANNEALING TRANSFER LEARNING PROCEDURE FOR INFORMAL LANGUAGE UNDERSTANDING TASKS

Anonymous authors
Paper under double-blind review

ABSTRACT

There are many applications for informal language understanding tasks in the real world. However, because informal language understanding tasks suffer more from data noise than formal ones, there is a huge performance gap between formal and informal language understanding tasks. The recent pre-trained models that improved the performance of formal language understanding tasks did not achieve comparable result on informal language. Although the formal tasks and informal tasks are similar in purpose, their language models significantly differ from each other. We propose a data annealing transfer learning procedure to bridge the performance gap on informal natural language understanding tasks. In the data annealing procedure, the training set contains mainly formal text data at first; then we gradually increase the proportion of the informal text data during the training process. We validate the data annealing procedure on three natural language understanding tasks: named entity recognition (NER), part-of-speech (POS) tagging, and chunking with two popular neural network models, LSTM and BERT. When BERT is fine-tuned with our learning procedure, it outperforms all the state-of-the-art models on the three informal tasks.

1 INTRODUCTION

Informal language is an important field in natural language processing (NLP). For example, tweets provide rich real-time information (Ritter et al., 2011). However, because of the noisy nature of the informal language and the shortage of labelled data, the progress on informal language is not as promising as on formal language. Many tasks on formal data obtain a high performance due to the development of deep learning methods. But usually, these state-of-the-art models' good performance can not directly transfer to informal data. For example, when a BERT model is fine-tuned on informal data, such as twitter text, its performance is less encouraging than on formal data. This is caused by the domain discrepancy between the pre-training corpus used by BERT and the target data. Towards different tasks in informal language, NLP researchers have also proposed different neural network models. Kshirsagar et al. (2018) use neural networks and pre-trained word embedding to detect hate speech in social media. Gui et al. (2018) improved the accuracy of part-of-speech tagging task in twitter by proposing hypernetwork based method to separately model contexts with different expression styles. However, these models are usually designed specifically for certain language understanding task and cannot generalize to different tasks with good performance. Researchers also explored two types of classic neural network-based transfer learning methods: parameter initialization (INIT) and multi-task transfer learning (MULT) on natural language processing. Lee et al. (2018) explored the possibility of training the Long Short-Term Memory (LSTM) model on the NER task with INIT. Yang et al. (2017) explored different MULT transfer learning strategies on sequence labelling tasks based on the relation between the source domain and the target domain. However, the performance of these existing transfer learning methods shows limited improvement in language understanding tasks Lin & Lu (2018).

To solve the issues mentioned above, we propose a data annealing procedure for transfer learning. In our proposed data annealing procedure, we set informal data as target data, and we set formal data as source data. During the training process, the training data first contains mainly source data. Our data annealing procedure takes the advantages of a good parameter initialization like INIT. Then the proportion of source data keeps decreasing while the proportion of target data keeps increasing. In the decaying process, it utilizes source data as an auxiliary task like MULT. Finally, the training set

named entity recognition example												
sentence:	Gotta	dress	up	for	london	fashion	week	and	party	in	style	!
labels:	O	O	O	O	B-activity	I-activity	I-activity	CC	VB	IN	NN	.
part-of-speech tagging example												
sentence:	Gotta	dress	up	for	london	fashion	week	and	party	in	style	!
labels:	VBP	VB	RP	IN	NNP	NN	NN	CC	VB	IN	NN	.
chunking example												
sentence:	Gotta	dress	up	for	london	fashion	week	and	party	in	style	!
labels:	b-vp	i-vp	b-prt	b-pp	b-np	i-np	i-np	o	b-vp	b-pp	b-np	o

Figure 1: A named entity recognition, part-of-speech tagging and chunking example of informal language.

is mainly target data. We suspect the data annealing process enables the model to explore a larger fraction of parameter space and keep the knowledge learnt from the source data.

In our data annealing transfer learning procedure, the BERT model is fine-tuned and achieves good performance in informal language understanding tasks. Besides, different from previously proposed models which lack generalization ability, our data annealing could be employed on different tasks. We validate our learning procedure with two popular neural network models in NLP, LSTM and BERT, on three popular natural language understanding tasks, i.e., named entity recognition (NER), part-of-speech (POS) tagging and chunking on twitter. Figure 1 shows one example for each tasks. When LSTM is trained with our proposed learning procedure, the result shows that our procedure outperforms the state-of-the-art models on the NER task and the chunking task. When BERT is fine-tuned with our data annealing procedure, it outperforms all three state-of-the-art models. By doing this, we also set the new state-of-the-art result for the three informal language understanding tasks. Last, we simulate a low resources environment in the NER task. Results show that our data annealing procedure is also effective when there are limited training resources in target data.

2 Related Work

Due to the importance of informal language, many researchers have been working on the various tasks on informal language. Aguilar et al. (2019) proposed using phonetic and phonological features to obtain a better representation in a social media text. Owoputi et al. (2013) built a POS tagging system for twitter tasks by large-scale unsupervised word clustering and new lexical features. Ma et al. (2016) proposed using recurrent neural networks to detect rumor in microblog. Researchers also have been investigating the possibility of applying transfer learning on the informal domain. Transfer learning aims to solve a problem on a domain using the knowledge gained from a related domain (Weiss et al., 2016). Parameter initialization (INIT) and multi-task learning (MULT) are two frequently used transfer learning techniques in NLP area. Lee et al. (2018) show the effect of transfer learning on named entity recognition task on informal language. Yang et al. (2017) examined different transfer learning methods based on the relation between the source domain and target domain. Lin & Lu (2018) proposed using domain adaptation in named entity recognition task on twitter text.

The core idea of annealing is to let the model have more freedom to explore its update direction or speed at the beginning of the training process. When the training process goes on, we gradually reduce the freedom of the model exploration. The philosophy of annealing has been frequently implemented in different machine learning areas. One of the popular usages of annealing is setting the neural model learning rate. Robbins & Monro (1951) and Zeiler (2012) show that a gradually decayed learning rate leads to better model performance. Moreover, recent state-of-the-art NLP models such as NCRF++ (Yang & Zhang, 2018) and BERT (Devlin et al., 2018) also empirically validate the effect of decaying learning rate. The idea of annealing is also applied to batch size adjustment in neural network model training. Smith et al. (2017) show that annealing the dataset batch size in training leads to a faster model convergence Bertsimas & Tsitsiklis (1993). Simulated

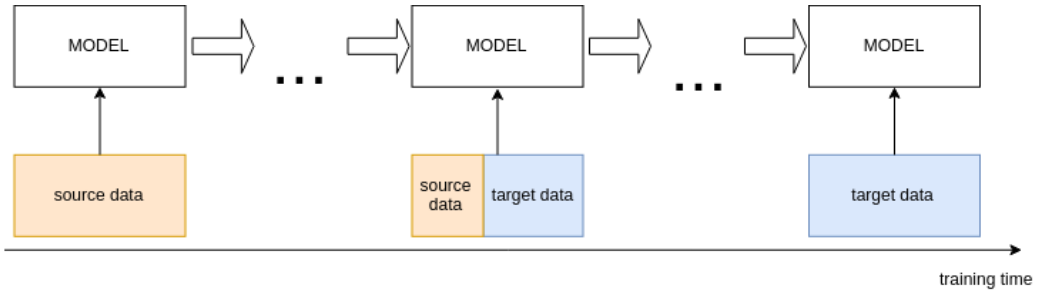

Figure 2: Data annealing procedure.

annealing algorithm is another application of the annealing concept. Simulated annealing reduces the probability of a model converging to a bad local optimal by introducing random noise in the training process.

Inspired the wide application of annealing, we propose the concept of data annealing to improve the performance on informal data in the transfer learning setting. Data annealing gradually reduces the amount of source data and increases the target data during the training process. By adjusting the data ratio, the model explores a larger parameter space at first and focuses on the targeted data later. Compared to simulated annealing, our proposed data annealing replaces random noise with source data. By including source data, the model is able to explore more space at the beginning of the training process, but also the model is guided by the knowledge learned from the source domain.

Figure 3 compare our data annealing procedure with INIT and MULT. It shows the connection among these three transfer learning technique from the variation of training set component during the training process. In INIT, before a time point T_0, the model is trained on source dataset. After T_0, the model is trained on target dataset. For MULT, the training data consists of target data and source data, and the proportion of source data is fixed to λ_0. In our proposed data annealing procedure, the proportion of the source dataset keeps decreasing during the annealing process.

3 METHOD

Data annealing is a transfer learning procedure that changes the proportional of formal source data and informal target data in the training process. At the beginning of the training, we let most of the training samples to be source data. So the model obtains a good initialization from the abundant clean source data, similar to a popular transfer learning technique in natural language processing (NLP), parameter initialization (INIT). We then gradually increase the proportion of target data and reduce the proportion of source data. As outlined in section 2, the model explores a larger parameter space. Besides, the labelled source dataset works as an auxiliary task like in multi-task transfer learning (MULT). At the end of the training process, most of the training data will be target data so that the model can focus on the target information. We illustrate the data annealing process in Figure 2.

The proposed data annealing process is guided by a unified function. We reduce the source data proportion with an exponential decay function. We let α represent the initial proportion of source data in the training set, let t represent the current training step and let m represent the number of total update batches. We let λ represent the exponential decay rate α. r_S^t and r_T^t represent the proportion of the source data and proportion of target data at time step t.

$$r_S^t = \alpha * \lambda^{t-1}, 0 < \alpha < 1, 0 < \lambda < 1 \tag{1}$$

$$r_T^t = 1 - \alpha * \lambda^{t-1} \tag{2}$$

We let D_S represent the size of source data, and B represent the batch size. We have

$$D_S = B * \sum_{i=1}^{m} r_S^t = B * \sum_{i=1}^{m} \alpha * \lambda^{i-1} = B * \frac{\alpha * (1 - \lambda^m)}{1 - \lambda} \tag{3}$$

Task Type	Category	Dataset	Train Tokens	Dev Tokens	Test Tokens
NER	Formal	Ontonote-nw	848,220	144,319	49,235
	Informal	RitterNER	37,098	4,461	4,730
POS Tagging	Formal	PTB 2003	912,344	131,768	129,654
	Informal	RitterPOS	10,857	2,242	2,291
Chunking	Formal	CoNLL 2000	211,727	-	47,377
	Informal	RitterCHUNK	10,610	2,309	2,292

Table 1: Dataset statistics.

When the model has been updated for adequate time, m will be large, so at the end of the training process, we can approximate D_S to

$$D_S = B * \frac{\alpha}{1-\lambda} \tag{4}$$

When we fine-tune large pre-trained model like BERT on a target task, it is suggested to avoid over-training the model on the target training data (Sun et al., 2019). Because most neural models have the catastrophic forgetting property, models will forget the knowledge from the pre-trained model if they are overfitted on target data (Kirkpatrick et al., 2016). However, we do not want to feed too much source data as well, as it not only prolongs the training time but also confuse the model. We empirically decide the size of source data to have at the beginning based on different tasks. So based on Equation 4, we set α as:

$$\alpha = D_S * (1-\lambda)/B \tag{5}$$

Linear decay is popular in learning rate annealing (Zeiler, 2012; Reimers & Gurevych, 2019). But linear annealing is not suitable for our proposed data annealing process. If we set a large value for α, the decay speed will be large, and α can decay to 0 in a short time. Such rapid change prevents the annealing process from taking effect in training. While if α is small, the model would not have a good parameter initialization from source data. Therefore we propose to use exponential decay in data annealing, as it decays smoothly and gradually. The proposed data annealing procedure can be used on any neural models. In this paper, we used LSTM and BERT as two examples.

4 Experiment Design

Because our proposed data annealing procedure could utilize knowledge from source domain without being binding with a specific neural network, it is applicable to all informal natural language understanding tasks. We validate it on three tasks: named entity recognition (NER), part-of-speech tagging (POS) and chunking. We choose these three tasks for two reasons. First, they are important tasks in understanding and have wide applications. Second, there exists a huge performance gap between formal text and informal text on these three tasks. These tasks are the perfect test bed for different transfer learning procedures. We test data annealing's effect on two types of popular neural models, LSTM and BERT.

4.1 Tasks and Datasets

NER locates and classifies named entity mentions in unstructured text into predefined categories. POS tagging marks a word in a text with its corresponding part-of-speech. Chunking is to separate a sentence into a group of separate noun phrases. The majority of available models for NER, POS tagging and chunking were designed for formal data like news. Because informal data such as tweets, usually include numerous nonstandard spellings, abbreviations, these models usually perform poorly on informal text genres (Strauss et al., 2016).

For NER task, we use OntoNotes-nw NER (Ralph Weischedel, 2013) as the source dataset, and use RitterNER dataset (Ritter et al., 2011) as the target dataset. While for POS tagging, we use Penn Treebank (PTB) POS tagging dataset as source data, and use RitterPOS (Ritter et al., 2011) as the target dataset. For the chunking task, we use CoNLL 2000 as the source dataset, and use RitterCHUNK (Ritter et al., 2011) as the target dataset. The statistic of these datasets is shown in Table 1.

4.2 MODEL IMPLEMENTATION

We implement two commonly used neural network models, LSTM and BERT to test the data annealing transfer learning procedure. Since parameter initialization (INIT) and multi-task learning (MULT) are two frequently used transfer learning technique in NLP (Weiss et al., 2016; Mou et al., 2016), we also implement them with the same LSTM and BERT for comparison.

4.2.1 LSTM MODEL IMPLEMENTATION

Long Short-Term Memory (LSTM) (Hochreiter & Schmidhuber, 1997) is frequently used in natural language processing tasks. It is well-suited to classify, predict and process time signal given time lags of unknown duration. Many state-of-the-art models on informal language tasks are based on LSTM. Following Yang & Zhang (2018) and Yang et al. (2017), we used character and word embedding as input features. We use one layer bidirectional LSTM to process the input features and use conditional random fields (CRFs) as the classifier. Since the source data and the target data has a different number of labels, we concatenate two separate conditional random fields (CRFs), one for the source data and one for the target data as the classifier. We use exponential data annealing as mentioned in section 3. The source data and target data flow to their corresponding CRF classifier. The parameters in CRF classifiers are updated by formal data and informal data separately, and the parameters in the LSTM or the BERT are updated by both formal data and informal data.

4.2.2 BERT MODEL IMPLEMENTATION

There are mainly two versions of BERT: $\text{BERT}_{\text{BASE}}$ and $\text{BERT}_{\text{LARGE}}$. Usually, $\text{BERT}_{\text{LARGE}}$ obtains a better performance compared with $\text{BERT}_{\text{BASE}}$. We implement both $\text{BERT}_{\text{BASE}}$ model and $\text{BERT}_{\text{LARGE}}$ as baselines. We use the pre-trained model parameter released by Google. Besides, as suggested in the original BERT paper, we utilize BERT by concatenating the token representations from the top four hidden layers of the pre-trained Transformer. Similar to the implementation in LSTM, we concatenate CRF classifiers at the top of the BERT structure. It is different from the original paper of BERT, which use the softmax function to classify the token type. There are mainly two reasons that we replace the softmax with the CRF. First, CRF has been validated as a good classifier by many researchers (Lafferty et al., 2001; Tseng et al., 2005). Besides, since all the LSTM baselines use CRF, in order to have a fair comparison with LSTM, it is necessary that we also use the CRF classifiers in BERT baselines.

4.3 BASELINE TRANSFER LEARNING PROCEDURES

As mentioned in section 2, parameter initialization (INIT) and multi-task learning (MULT) are two commonly used transfer learning technique in NLP. We also apply these transfer learning methods on both LSTM and BERT. Then we compare the effect of our proposed data annealing procedure with INIT and MULT.

4.3.1 PARAMETER INITIALIZATION (INIT)

INIT contains three steps. First, a source model is trained on source domain by supervised learning or unsupervised learning. Then some or all of the parameters in the source model is used to initialize the target model. Finally, the target model is fine-tuned on the target domain.

Lee et al. (2018) and Mou et al. (2016) validate the effect of INIT in natural language understanding (NLU). We apply INIT on both $\text{BERT}_{\text{BASE}}$ and $\text{BERT}_{\text{LARGE}}$. We set formal data as source data and informal data as target data. When training source BERT model with source data, in order to avoid catastrophic forgetting, we set the maximum training epochs on the source dataset to be 5. We first train the source model on source data. We choose the model that achieves the best performance on the validation set of source data for named entity recognition task (NER) and part-of-speech tagging task. For the chunking task, since the source dataset CoNLL 2000 does not contain a validation set, we choose the model that achieves the highest performance on the test set. Since source dataset and target dataset do not have the same number of labels, the CRF classifier for source data can not be used to initialize the parameter in the CRF classifier for target data. Therefore, we initialize the target model by parameters in the source model except for the parameters in the CRF classifier. Finally, we fine-tune the model on target dataset.

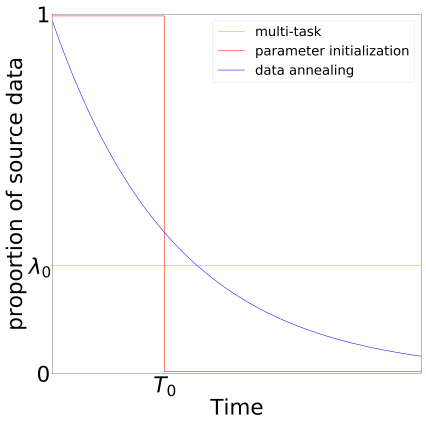

Figure 3: Comparison between data annealing transfer learning procedure, parameter initialization and multi-task transfer learning.

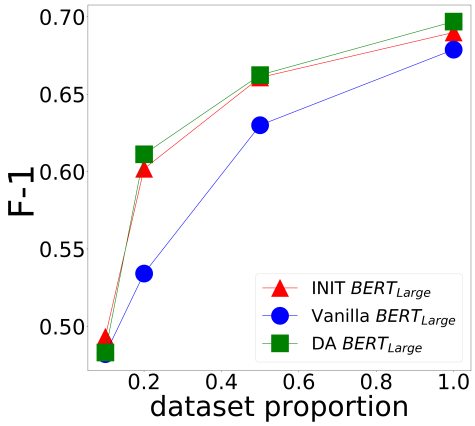

Figure 4: Performance on named entity recognition task. DA $\text{BERT}_{\text{LARGE}}$ indicates Vanilla $\text{BERT}_{\text{LARGE}}$ finetuned with data annealing.

4.3.2 Multi-Tasks Transfer Learning (MULT)

Multi-task learning is another popular transfer learning technique. In MULT, a model is trained by both source and target data at the same time, and some or all of the parameters in the model are shared between source and target domain during the learning process.

We apply MULT on both LSTM and BERT. The training set consists of informal data and formal data. The proportion of source data λ_{MULT} in each batch is fixed during the training process. We concatenate the LSTM or BERT with two separate CRF classifiers because the source dataset and target dataset have a different number of labels. Source data and target data train all the parameters in the model jointly except for the parameters in the CRF classifiers.

5 Experiment Results

We use different metrics to measure each task's performance following previous work. We use precision, recall and F_1 in entity level to evaluate named entity recognition (NER) and chunking. We use accuracy in token level to evaluate part-of-speech (POS) tagging task. We compared our data annealing procedure with three types of baselines and state-of-the-art models in Table 2. Vanilla means the model is trained without transfer learning, in other words, the model does not utilize the source data. A model with MULT means we apply multi-task transfer learning to train the model. A model with INIT means we apply the parameter initialization technique to train the model. A model with DA means the model is fine-tuned with our proposed data annealing procedure. While previous state-of-the-art results on the informal NER task, POS tagging task and chunking task are achieved by different models, we use the same model across different tasks to achieve the best result. We report the average result of three runs.

5.1 Named Entity Recognition

The results on named entity recognition task (NER) are in Table 2. When LSTM is used as the learning model, our data annealing procedure achieves the highest recall and $F-1$ score compared to INIT and MULT. When $\text{BERT}_{\text{BASE}}$ is used as the learning model, our learning procedure achieves the highest precision and F-1 score among all transfer learning methods, and it also achieves the highest recall with INIT. When $\text{BERT}_{\text{LARGE}}$ is used as the learning model, our data annealing procedure achieves the best precision and F-1 score compared with INIT and MULT. It outperforms state-of-the-art model by 3.16 in F_1 scores.

model	NER			POS	Chunking		
	P	R	F_1	A	P	R	F_1
Vanilla LSTM	**75.55**	55.75	64.05	88.65	83.76	83.78	83.77
MULT LSTM	74.51	58.48	65.49	88.81	**83.92**	84.48	84.20
DA LSTM	75.51	**61.01**	**67.45**	**89.16**	83.81	**85.37**	**84.58**
$\text{BERT}_{\text{BASE}}$	68.73	62.74	65.58	91.05	85.05	85.96	85.50
INIT $\text{BERT}_{\text{BASE}}$	69.28	**63.74**	66.40	90.85	85.48	86.77	86.13
MULT $\text{BERT}_{\text{BASE}}$	70.42	62.38	66.12	**9**1.39	86.01	87.75	86.87
DA $\text{BERT}_{\text{BASE}}$	**71.09**	**63.74**	**67.21**	**91.55**	**86.16**	**87.91**	**87.03**
Vanilla $\text{BERT}_{\text{LARGE}}$	68.41	67.45	67.88	91.88	85.55	86.78	86.16
INIT $\text{BERT}_{\text{LARGE}}$	68.85	**69.20**	68.99	92.04	86.42	87.59	87.00
MULT $\text{BERT}_{\text{LARGE}}$	70.05	66.08	68.00	92.06	86.29	87.21	86.54
DA $\text{BERT}_{\text{LARGE}}$	**70.61**	68.81	**69.69**	**92.54**	**86.71**	**88.15**	**87.53**
previous state-of-the-art*	76.12	59.10	66.53	91.17	84.47	84.54	84.50

Table 2: Results on NER, POS tagging and chunking task. *The previous state-of-the-art methods are achieved by different models. Yang et al. (2019) proposed the state-of-the-art model in NER task; Gui et al. (2018) proposed the state-of-the-art model in POS tagging task; Yang et al. (2017) proposed the state-of-the-art model in chunking task.

We notice that when LSTM or BERT is implemented with data annealing, the recall and F_1 are always higher than the previous state-of-the-art method, meaning our data annealing is especially good at finding entities in text.

Usually, a sentence contains more non-entity words than entity words. If the model is not sure whether a word belongs to a certain entity, the model is likely to predict it as non-entity in order to reduce the training loss. We observe that the previous state-of-the-art model achieved the highest precision, but its recall is lower than all versions of $\text{BERT}_{\text{BASE}}$ and $\text{BERT}_{\text{LARGE}}$ models. This indicates that the previous state-of-the-art methods achieve high performance by predicting fewer entities, while BERT models receive high performance by both covering more entities and predicting them correctly.

5.2 Part-of-speech Tagging

Different from the NER task and the chunking task which aim to identify text spans, POS tagging predicts every word's POS tag. So we use word-level accuracy to evaluate POS tagging. LSTM applied with our proposed data annealing procedure achieves higher accuracy than using the other two transfer learning procedures. Both $\text{BERT}_{\text{BASE}}$ and $\text{BERT}_{\text{LARGE}}$ fine-tuned using the data annealing procedure outperforms the same setting with other transfer learning procedures. The improvement over the previous state-of-the-art method is 1.49 in accuracy measure. We notice that the performance improvement is not as large as in the NER task. This could be partially explained by the discrepancy between source data and target data. The target dataset in POS task, i.e., RitterPOS, contains more labels than the labels in the source dataset. Besides, the labels in RiiterPOS contains several Twitter-specific tags: retweets, @usernames, hashtags, and URLs. These label types never occurred in the source dataset. So transfer learning can't benefit the recognition on tokens with these part-of-speech types. This suggests that the similarity between source data and target data is an important factor to consider in the transfer learning process.

5.3 Chunking

When LSTM is used as the training model, our data annealing procedure achieves the best results among these transfer learning methods. When $\text{BERT}_{\text{BASE}}$ is applied as the training model, our proposed methods outperform all other transfer learning procedure. When $\text{BERT}_{\text{LARGE}}$ is applied as the learning model, our proposed data annealing procedure outperforms all other transfer learning procedure and outperforms the state-of-the-art model by 3.03 in F_1. On chunking result, the performance meets our expectation. It further validates that our annealing procedure could improve the performance of informal language chunking task.

5.4 THE INFLUENCE OF THE DATASET SIZE

To further evaluate the performance of our methods when there is limited labelled data, we randomly sample 10%, 20% and 50% of the training set in RitterNER. Then we compare our proposed DA $\text{BERT}_{\text{LARGE}}$ with INIT $\text{BERT}_{\text{LARGE}}$ and Vanilla $\text{BERT}_{\text{LARGE}}$ baselines on those three sampled datasets and the original RitterNER dataset. The result in Figure 4 shows that our model is slightly better than INIT $\text{BERT}_{\text{LARGE}}$ on limited resources condition and achieves a significant improvement over Vanilla $\text{BERT}_{\text{LARGE}}$ baseline.

6 ERROR ANALYSIS

We performed error analysis on the named entity recognition task to understand each method better. We first calculated the F_1 score of the ten predefined entity types. We find that compared with Vanilla $\text{BERT}_{\text{LARGE}}$ and INIT $\text{BERT}_{\text{LARGE}}$, DA $\text{BERT}_{\text{LARGE}}$ achieves higher F_1 score on two frequent entities, "PERSON" and "OTHER". INIT $\text{BERT}_{\text{LARGE}}$ achieves the higher F_1 score on another frequent entity type, "GEO-LOC". In the comparison of performance of other entities, we did not notice clear differences. "PERSON" is a frequent concept in formal data, which is the source data in our transfer learning setting. It implies that DA $\text{BERT}_{\text{LARGE}}$ takes better use of knowledge from the source domain compared with INIT $\text{BERT}_{\text{LARGE}}$ and Vanilla $\text{BERT}_{\text{LARGE}}$. Besides, "OTHER" means entities that are not in the ten predefined entity types. Higher performance on "OTHER" suggests DA $\text{BERT}_{\text{LARGE}}$ has a better understanding of the entity than INIT $\text{BERT}_{\text{LARGE}}$ and Vanilla $\text{BERT}_{\text{LARGE}}$. Besides, we found there is a connection between the frequency of entity type and F_1 score of that entity. If the entity type of a word is infrequent in the dataset, all the three models are less likely to predict the word correctly. For example, since "TVSHOW" is one of the most infrequent entity types in the training set of RitterNER, none of these three models predicts a word with an entity type of "TVSHOW" correctly. Meanwhile, "PERSON" is the most frequent entity type in the training set, and the F_1 of "PERSON" achieves the second-highest F_1. We suspect that if a word is a frequent entity type, then even if it never appeared in training set, another word that has a similar representation may be in the training data. Therefore, the model implicitly learns to predict a word by learning from other words that belong to the same type. Assigning more penalty to a less common word may be a solution to such issue.

Besides, we randomly sampled 30 sentences which contain at least one word whose entity type is mispredicted from the test set of RitterNER. We found all of the model have a difficulty in labelling noisy sentence. For example, all models fails to recognize *JUTHTIN BEAVERRRR* as entity type "PERSON" in sentence *@darynjones imma tell you one time ! JUTHTIN BEAVERRRR !!!!*. We would like to mention that there is a fairly large proportion of such noisy sentences in total sampled sentences. This fact suggests transfer learning has limited effect when the informal data contains too much noise.

7 CONCLUSION

In this paper, we propose the data annealing transfer learning procedure for informal language understanding tasks, such as named entity recognition (NER), part-of-speech (POS) tagging and chunking. It is applicable to various models such as LSTM and BERT. In the experiment, we show that when data annealing is applied with LSTM or BERT, it outperforms different state-of-the-art models on different informal language understanding tasks. Our proposed annealing technique is also effective when there is limited labelled resources. Moreover, compared with popular transfer learning methods, our data annealing procedure achieves better results on informal language understanding tasks. In the future, we will explore a generalization version of data annealing to general domain transfer instead of from the formal domain to informal domain. Besides, we also think about an automatic annealing transfer learning procedure without setting the initial source data proportion and decaying rate. We think training loss and gradient may be a good reference.

REFERENCES

Gustavo Aguilar, Adrián Pastor López-Monroy, Fabio A. González, and Thamar Solorio. Modeling noisiness to recognize named entities using multitask neural networks on social media. *CoRR*,

abs/1906.04129, 2019. URL http://arxiv.org/abs/1906.04129.

Dimitris Bertsimas and John Tsitsiklis. Simulated annealing. *Statist. Sci.*, 8(1):10–15, 02 1993. doi: 10.1214/ss/1177011077. URL https://doi.org/10.1214/ss/1177011077.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. URL http://arxiv.org/abs/1810.04805.

Tao Gui, Qi Zhang, Jingjing Gong, Minlong Peng, Di Liang, Keyu Ding, and Xuanjing Huang. Transferring from formal newswire domain with hypernet for twitter POS tagging. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pp. 2540–2549, 2018. URL https://aclanthology.info/papers/D18-1275/d18-1275.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November 1997. ISSN 0899-7667. doi: 10.1162/neco.1997.9.8.1735. URL http://dx.doi.org/10.1162/neco.1997.9.8.1735.

James Kirkpatrick, Razvan Pascanu, Neil C. Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *CoRR*, abs/1612.00796, 2016. URL http://arxiv.org/abs/1612.00796.

Rohan Kshirsagar, Tyus Cukuvac, Kathleen R. McKeown, and Susan McGregor. Predictive embeddings for hate speech detection on twitter. *CoRR*, abs/1809.10644, 2018. URL http://arxiv.org/abs/1809.10644.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pp. 282–289, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc. ISBN 1-55860-778-1. URL http://dl.acm.org/citation.cfm?id=645530.655813.

Ji Young Lee, Franck Dernoncourt, and Peter Szolovits. Transfer learning for named-entity recognition with neural networks. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, Miyazaki, Japan, May 2018. European Languages Resources Association (ELRA). URL https://www.aclweb.org/anthology/L18-1708.

Bill Yuchen Lin and Wei Lu. Neural adaptation layers for cross-domain named entity recognition. *CoRR*, abs/1810.06368, 2018. URL http://arxiv.org/abs/1810.06368.

Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard J. Jansen, Kam-Fai Wong, and Meeyoung Cha. Detecting rumors from microblogs with recurrent neural networks. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, IJCAI'16, pp. 3818–3824. AAAI Press, 2016. ISBN 978-1-57735-770-4. URL http://dl.acm.org/citation.cfm?id=3061053.3061153.

Lili Mou, Zhao Meng, Rui Yan, Ge Li, Yan Xu, Lu Zhang, and Zhi Jin. How transferable are neural networks in NLP applications? *CoRR*, abs/1603.06111, 2016. URL http://arxiv.org/abs/1603.06111.

O. Owoputi, B. O'Connor, C. Dyer, Kevin Gimpel, N. Schneider, and N.A. Smith. Improved part-of-speech tagging for online conversational text with word clusters. *Proceedings of NAACL-HLT*, 2013:380–390, 01 2013.

Mitchell Marcus Eduard Hovy Sameer Pradhan Lance Ramshaw Nianwen Xue Ann Taylor Jeff Kaufman Michelle Franchini Mohammed El-Bachouti Robert Belvin Ann Houston. Ralph Weischedel, Martha Palmer. Ontonotes release 5.0 ldc2013t19. In *Linguistic Data Consortium*, 2013.

Nils Reimers and Iryna Gurevych. Alternative weighting schemes for elmo embeddings. *CoRR*, abs/1904.02954, 2019. URL http://arxiv.org/abs/1904.02954.

Alan Ritter, Sam Clark, Mausam, and Oren Etzioni. Named entity recognition in tweets: An experimental study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pp. 1524–1534, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics. ISBN 978-1-937284-11-4. URL http://dl.acm.org/citation.cfm?id=2145432.2145595.

Herbert Robbins and Sutton Monro. A stochastic approximation method. *Ann. Math. Statist.*, 22(3): 400–407, 09 1951. doi: 10.1214/aoms/1177729586. URL https://doi.org/10.1214/aoms/1177729586.

Samuel L. Smith, Pieter-Jan Kindermans, and Quoc V. Le. Don't decay the learning rate, increase the batch size. *CoRR*, abs/1711.00489, 2017. URL http://arxiv.org/abs/1711.00489.

Benjamin Strauss, Bethany Toma, Alan Ritter, Marie-Catherine de Marneffe, and Wei Xu. Results of the WNUT16 named entity recognition shared task. In *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)*, pp. 138–144, Osaka, Japan, December 2016. The COLING 2016 Organizing Committee. URL https://www.aclweb.org/anthology/W16-3919.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. How to fine-tune BERT for text classification? *CoRR*, abs/1905.05583, 2019. URL http://arxiv.org/abs/1905.05583.

Huihsin Tseng, Pichuan Chang, Galen Andrew, Daniel Jurafsky, and Christopher Manning. A conditional random field word segmenter for sighan bakeoff 2005. In *Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing*, 2005. URL https://www.aclweb.org/anthology/I05-3027.

Karl Weiss, Taghi M. Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big Data*, 3(1):9, May 2016. ISSN 2196-1115. doi: 10.1186/s40537-016-0043-6. URL https://doi.org/10.1186/s40537-016-0043-6.

Jie Yang and Yue Zhang. NCRF++: an open-source neural sequence labeling toolkit. *CoRR*, abs/1806.05626, 2018. URL http://arxiv.org/abs/1806.05626.

Wei Yang, Wei Lu, and Vincent W. Zheng. A simple regularization-based algorithm for learning cross-domain word embeddings. *CoRR*, abs/1902.00184, 2019. URL http://arxiv.org/abs/1902.00184.

Zhilin Yang, Ruslan Salakhutdinov, and William W. Cohen. Transfer learning for sequence tagging with hierarchical recurrent networks. *CoRR*, abs/1703.06345, 2017. URL http://arxiv.org/abs/1703.06345.

Matthew D. Zeiler. ADADELTA: an adaptive learning rate method. *CoRR*, abs/1212.5701, 2012. URL http://arxiv.org/abs/1212.5701.

A Appendix

You may include other additional sections here.