Data Annealing for Informal Language Understanding Tasks

Jing Gu
University of California, Davis
jkgu@ucdavis.edu

Zhou Yu
University of California, Davis
joyu@ucdavis.edu

Abstract

There is a huge performance gap between formal and informal language understanding tasks. The recent pre-trained models that improved formal language understanding tasks did not achieve a comparable result on informal language. We propose data annealing transfer learning procedure to bridge the performance gap on informal natural language understanding tasks. It successfully utilizes a pre-trained model such as BERT in informal language. In the data annealing procedure, the training set contains mainly formal text data at first; then, the proportion of the informal text data is gradually increased during the training process. Our data annealing procedure is model-independent and can be applied to various tasks. We validate its effectiveness in exhaustive experiments. When BERT is implemented with our learning procedure, it outperforms all the state-of-the-art models on the three common informal language tasks.

1 Introduction and Related Work

Because of the noisy nature of the informal language and the shortage of labeled data, the progress on informal language is not as promising as in formal language. Many tasks on formal data obtain a high performance due to deep neural models (Peters et al., 2018; Devlin et al., 2018). However, these state-of-the-art models' excellent performance usually fails to transfer to informal data directly. For example, when a BERT model is fine-tuned on informal data, its performance is less encouraging than on formal data. It is because of the domain discrepancy between the pre-training corpus used by BERT and the target data.

To solve the issues mentioned above, we propose a model-agnostic data annealing procedure. We set informal data as target data and set formal data as source data. The training data first contains mainly source data, when data annealing procedure takes the advantages of a proper parameter initialization from the clean nature of formal data. The proportion of source data keeps decreasing exponentially while the proportion of target data keeps increasing, which empowers the model with more freedom to explore the direction of its next update.

The philosophy behind data annealing is shared with other commonly used annealing techniques. One popular usage of annealing is learning rate annealing. A gradually decayed learning rate enhances the model with more freedom of exploration at the beginning and leads to better model performance (Zeiler, 2012; Yang and Zhang, 2018; Devlin et al., 2018). Another widespread implementation of annealing is simulated annealing (Bertsimas and Tsitsiklis, 1993). It reduces the probability of a model converging to a bad local optimal by introducing random noise in the training process. Data annealing has similar functionality with simulated annealing but replaces random noise with source data. By doing this, the model explores more space at the beginning of the training process and is guided by the knowledge learned from the source domain.

Current state-of-the-art models on informal language tasks are usually designed specifically for a particular task and cannot generalize to different tasks (Kshirsagar et al., 2018; Gui et al., 2018). Data annealing is model-independent and could be employed in various informal language 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.

When BERT is fine-tuned with data annealing procedure, it outperforms all three state-of-the-art models with the same structure. By doing this, we also set the new state-of-the-art result for the three

Findings of the Association for Computational Linguistics: EMNLP 2020, pages 3153–3159
November 16 - 20, 2020. ©2020 Association for Computational Linguistics

informal language understanding tasks. Experiments also validate our data annealing procedure's effectiveness when there are limited training resources in target data.

2 Data Annealing

A pre-trained model like BERT is suggested to avoid over-training when implemented on downstream task (Peters et al., 2019; Sun et al., 2019). In transfer learning, It is not ideal to feed too much source data, as it not only prolongs the training time but also confuses the model. Therefore, we propose data annealing, a transfer learning procedure that adjusts the ratio of the formal source data and the informal target data from large to small in the training process to solve the overfitting and the noisy initialization problems.

At the first stage of data annealing, most of the training samples are source data. Therefore the model obtains a proper initialization from the abundant clean source data. In the second stage, as we gradually increase the proportion of the target data and reduce the proportion of the source data, the model explores a larger parameter space. Besides, the labeled source dataset works as an auxiliary task. At the third stage of the training process, most of the training data is target data so that the model focus on the target information more.

We reduce the source data proportion exponentially. α represents the initial proportion of the source data. t represents the current training step, and m represents the number of batches in total. λ represents the exponential decay rate of α. r_S^t and r_T^t represent the proportion of the source data and proportion of target data at time step t.

$$r_S^t = \alpha\lambda^{t-1}, 0 < \alpha < 1, 0 < \lambda < 1 \quad (1)$$

$$r_T^t = 1 - \alpha \cdot \lambda^{t-1} \quad (2)$$

Let D_S represents the accumulated source data used to train the model, and let B represents the batch size. We have

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

After the model is updated for adequate batches, we can approximate D_S using

$$D_S = B \cdot \frac{\alpha}{1-\lambda} \quad (4)$$

D_S could be empirically decided based on the relation between source dataset and target dataset. For example, the higher the similarity between the source and the target data, the more knowledge the target task could borrow from the source task, and larger D_S is. If researchers want to simplify the hyper-parameters tuning process or constrain the influence of source data, α can be set by D_S:

$$\alpha = D_S \cdot (1-\lambda)/B \quad (5)$$

3 Experimental Design

We validate it by two popular model LSTM and BERT on three tasks: named entity recognition (NER), part-of-speech tagging (POS), and chunking. These tasks have much better performance on formal text (such as news) than informal text (such as tweets).

3.1 Datasets

We use OntoNotes-nw (Ralph Weischedel, 2013) as the source dataset, and Ritter11-NER dataset (Ritter et al., 2011) as the target dataset to validate the NER task. While we use Penn Treebank (PTB) POS tagging dataset (Mitchell P. Marcus, 1999) as the source data set, and Ritter11-POS (Ritter et al., 2011) as the target dataset in the POS tagging task. For the chunking task, we use CoNLL 2000 (Sang and Buchholz, 2000) as the source dataset, and Ritter11-CHUNK (Ritter et al., 2011) as the target dataset. Please refer to Appendix B for more details about datasets.

3.2 Model Setting

We implemented BERT and LSTM to validate the effect of data annealing on all three tasks.

BERT. We implemented both $\text{BERT}_{\text{BASE}}$ model and $\text{BERT}_{\text{LARGE}}$ model. CRF has been validated as a good classifier by many researchers (Lafferty et al., 2001; Tseng et al., 2005). We use CRF as a decoder on the top of the BERT structure. In some tasks, the source dataset and target dataset do not have the same set of labels. Therefore, we use two separate CRF decoder for source task and target task.

LSTM. We used character and word embedding as input features following previous works (Yang and Zhang, 2018; Yang et al., 2017). We use one layer bidirectional LSTM to process the input features. For the same reason as in the implementation of BERT, we use two separate CRF classifiers on the top of the LSTM structure.

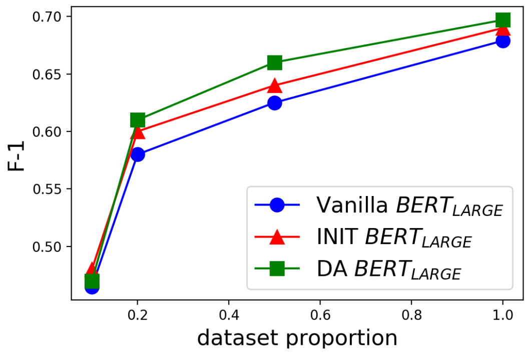

Figure 1: Performance on named entity recognition task. DA BERT$_{\text{LARGE}}$ indicates Vanilla BERT$_{\text{LARGE}}$ finetuned with data annealing.

We compare data annealing with two popular transfer learning paradigms, parameter initialization (INIT) and multi-task learning (MULT) (Weiss et al., 2016; Mou et al., 2016). Now we introduce the training procedure in experiments.

Data annealing. In all data annealing experiments, the initial source data ratio α and decay rate λ are tuned in range (0.9, 0.99). When training the BERT model, we also calculated the estimated total batches from source data D_S that fed into the model by equation 5. By avoiding a large D_S, the model has a lower probability of suffering from catastrophic forgetting as mentioned in section 2.

MULT. Multi-task transfer learning optimizes an auxiliary task to improve the performance on the target task. We implemented MULT on both LSTM-CRF and BERT-CRF structure. In all MULT experiments, following Yang et al. (2017) and Collobert and Weston (2008), we tune the ratio of source data in range (0.1, 0.9).

INIT. Parameter initialization transfer learning transfers weights from a pre-trained model to improve the performance of the target model. We implemented INIT on BERT-CRF structure. In all INIT experiments, we run three times on source data and conduct weight transferring on the model that achieves the highest performance. In INIT, the target model benefits from a good initialization with contains knowledge from source dataset.

4 Experiment Results

The result of the three tasks is shown in Table 1. Vanilla means the model is trained without transfer learning and only utilizes the target data. DA means the model is implemented with data annealing procedure. All the numbers in the tables are the average result of three runs. It is worth noting that state-of-the-art results on these three tasks are achieved by different models and complicated adaptation methods. Meanwhile, our proposed data annealing algorithm is applied to the same structure without fancy decoration across different tasks. Within our appropriate range set of (0.9, 0.99) for α and λ, we find the data annealing consistently outperforms other transfer learning methods and the state-of-the-art method. In most cases, it is a moderate annealing speed that leads to an optimal result. We noticed that the improvement in recently reported literature on these tasks is usually less than 0.5 in absolute value on either F_1 or accuracy (Gui et al., 2018; Lin and Lu, 2018). Our data annealing moves the state-of-the-art performance a big step forward. For more experiment detail such as hyper-parameters, please refer to Appendix C

Named Entity Recognition (NER). Our annealing procedure outperforms other transfer learning procedures in terms of F_1, meaning our data annealing is especially effective in striking a balance between the precision and recall in extracting named entities. Usually, a sentence contains more words that are not entities. So if the model is not sure whether a word is an entity, the model is likely to predict it as not an entity in order to reduce the training loss. The state-of-the-art models achieved high precision but low recall by using several adaptation methods. It indicates that the 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.

Part-of-speech Tagging (POS tagging). All the BERT models and LSTM models under our data annealing procedure outperform other transfer learning procedures. The improvement over the state-of-the-art model DCNN (Gui et al., 2018) is 1.37 in accuracy measure in POS tagging. It is worth noting that improvement in this task was limited before our work. For example, DCNN only improved 0.26 in accuracy comparing research works before it. Our method also outperforms a recent pre-training work BERTweet (Nguyen et al., 2020) by 2.24 in accuracy. **Chunking.** When LSTM, BERT$_{\text{BASE}}$, and BERT$_{\text{LARGE}}$ are used as the training model under our data annealing procedure, they achieve better performances compared to other transfer learning paradigms. Our best model outperforms the state-of-the-art model by 3.03 in F_1.

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**
Vanilla BERT$_{\text{BASE}}$	68.73	62.74	65.58	91.05	85.05	85.96	85.50
INIT BERT$_{\text{BASE}}$	69.28	**63.74**	66.40	90.85	85.48	86.77	86.13
MULT BERT$_{\text{BASE}}$	70.42	62.38	66.12	91.39	86.01	87.75	86.87
DA BERT$_{\text{BASE}}$	**71.09**	**63.74**	**67.21**	**91.55**	**86.16**	**87.91**	**87.03**
Vanilla BERT$_{\text{LARGE}}$	68.41	67.45	67.88	91.88	85.55	86.78	86.16
INIT BERT$_{\text{LARGE}}$	68.85	**69.20**	68.99	92.04	86.42	87.59	87.00
MULT BERT$_{\text{LARGE}}$	70.05	66.08	68.00	92.06	86.29	87.21	86.54
DA BERT$_{\text{LARGE}}$	**70.61**	68.81	**69.69**	**92.54**	**86.71**	**88.15**	**87.53**
*Over state-of-the-art	**-5.51**	**+9.71**	**+3.16**	**+1.37**	**+2.24**	**+3.61**	**+3.03**
**State-of-the-art	76.12	59.10	66.53	91.17	84.47	84.54	84.50

Table 1: Results on NER, POS tagging and chunking task. * means the difference between DA BERT$_{\text{LARGE}}$ and state-of-the-art results. ** means the state-of-the-art for these three tasks are achieved by different models. Listed state-of-the-art NER and POS tagging result came from Lin and Lu (2018), Gui et al. (2018). Since Yang et al. (2017) proposed the state-of-the-art model on informal chunking task but experimented on a different informal text dataset, we implement their model on Ritter11-Chunk dataset and report the result.

The Dataset Size Influence. To further evaluate data annealing when there is limited labeled data, we randomly sample 10%, 20%, and 50% of the training set in Ritter11-NER. Then we compare our proposed DA BERT$_{\text{LARGE}}$ with INIT BERT$_{\text{LARGE}}$ and Vanilla BERT$_{\text{LARGE}}$ baselines. We take the average performance of 5 runs for each model. The result in Figure 1 shows that our model is still better than INIT BERT$_{\text{LARGE}}$ on the condition of a limited resource and achieves a significant improvement over Vanilla BERT$_{\text{LARGE}}$ baseline.

5 Error Analysis

We did an error analysis in Ritter11-NER dataset. We randomly sampled 30 sentences that contain entities that are incorrectly predicted by DA BERT$_{\text{LARGE}}$ and attached them in Appendix A. We found that a relatively large proportion of sentences has a too strong noisy feature to be predicted correctly. This feature is embedded in the informal text, and we might need to explore more on the nature of informal language to solve it perfectly.

We also calculated the F_1 score of the ten predefined entity types. We find that compared with Vanilla BERT$_{\text{LARGE}}$ and INIT BERT$_{\text{LARGE}}$, DA BERT$_{\text{LARGE}}$ achieves higher F_1 score on two frequent entities, ”PERSON” and ”OTHER”. ”PERSON” is a frequent concept in formal data. It shows our method learns to utilize formal data knowledge to improve ”PERSON” detection. Besides, ”OTHER” means entities that are not in the ten pre-defined entity types. Higher performance on ”OTHER” suggests DA BERT$_{\text{LARGE}}$ has a better understanding of the general concept of an entity. INIT BERT$_{\text{LARGE}}$ achieves a higher F_1 score on ”GEO-LOC”. We did not find a clear difference in other entity types.

Besides, we found that if a word is of a rarely appeared entity type, all the three models are less likely to predict its entity type correctly. We suspect that a neural model implicitly learns to predict a word when it is trained to predict other words in the same entity type since these words could share a similar representation in the NER task. We plan to assign more penalty to infrequent entity types to tackle this issue in the future.

6 Conclusion

In this paper, we propose data annealing, a model-independent transfer learning procedure for informal language understanding tasks. It applies to various models such as LSTM and BERT. It has been proven as a good approach to utilizing knowledge from formal data to informal data by exhaustive experiments. When data annealing is applied with BERT, it outperforms different state-of-the-art models on different informal language understanding tasks. Since large pre-trained models have been widely used, it could also serve as an excellent finetuning method. Data annealing is also useful when there are limited labeled resources.

References

Dimitris Bertsimas and John Tsitsiklis. 1993. Simulated annealing. *Statist. Sci.*, 8(1):10–15.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pages 160–167, New York, NY, USA. ACM.

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

Tao Gui, Qi Zhang, Jingjing Gong, Minlong Peng, Di Liang, Keyu Ding, and Xuanjing Huang. 2018. 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*, pages 2540–2549.

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

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Bill Yuchen Lin and Wei Lu. 2018. Neural adaptation layers for cross-domain named entity recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2012–2022, Brussels, Belgium. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Beatrice Santorini Mitchell P. Marcus. 1999. Treebank-3 ldc99t42. In *Linguistic Data Consortium*.

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

Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. 2020. Bertweet: A pre-trained language model for english tweets.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *CoRR*, abs/1802.05365.

Matthew E. Peters, Sebastian Ruder, and Noah A. Smith. 2019. To tune or not to tune? adapting pretrained representations to diverse tasks. *CoRR*, abs/1903.05987.

Martha Palmer Ralph Weischedel. 2013. Ontonotes release 5.0 ldc2013t19. In *Linguistic Data Consortium*.

Alan Ritter, Sam Clark, Mausam, and Oren Etzioni. 2011. Named entity recognition in tweets: An experimental study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 1524–1534, Stroudsburg, PA, USA. Association for Computational Linguistics.

Erik F Sang and Sabine Buchholz. 2000. Introduction to the conll-2000 shared task: Chunking. *arXiv preprint cs/0009008*.

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

Huihsin Tseng, Pichuan Chang, Galen Andrew, Daniel Jurafsky, and Christopher Manning. 2005. A conditional random field word segmenter for sighan bakeoff 2005. In *Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing*.

Karl Weiss, Taghi M. Khoshgoftaar, and DingDing Wang. 2016. A survey of transfer learning. *Journal of Big Data*, 3(1):9.

Jie Yang and Yue Zhang. 2018. Ncrf++: An open-source neural sequence labeling toolkit. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*.

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

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

A Mispredicted Sentences Examples on Named Entity Recognition Task

1	Is making me purchase windows{**NO_ENTITY, B-PRODUCT**} , antivirus and office{**NO_ENTITY, B-PRODUCT**}
2	ellwood{**NO_ENTITY, B-PERSON**} 's sushi , a glass of pinot , &quot; strokes{**NO_ENTITY, B-OTHER**} of{**NO_ENTITY, I-OTHER**} genius{**NO_ENTITY, I-OTHER**} &quot; by john wertheim{**NO_ENTITY, I-PERSON**} , play at barksdale{**NO_ENTITY, B-FACILITY**} in a bit , lovely friday night :)
3	lalala{**B-GEO-LOC, NO_ENTITY**} south{**B-GEO-LOC, NO_ENTITY**} game tonight !!!! Go us . http://bit.ly/b351o9 RT BunBTrillOG : Okay #teamtrill time to show them our power ! #BunB106andPark needs to trend now ! RT til it hurts ! I got ya twitter{**NO_ENTITY, B-COMPANY**} jail ...
4	Chicago Weekend Events : Lebowski{**NO_ENTITY, B-OTHER**} Fest{**NO_ENTITY, I-OTHER**} , Dave{**NO_ENTITY, B-PERSON**} Matthews{**NO_ENTITY, I-PERSON**} , Latin Music And More : The lively weekend (well , Friday throu ... http://bit.ly/cLTnyl
5	RT @DonnieWahlberg : Soldiers ... Familia ... BH's...{**B-PERSON, NO_ENTITY**} NK Fam ... Homies ... Etc . Etc . Etc .... I 'm gonna need some company next Friday in NYC ...
6	tell ur dad2bring the ypp back in Hayes{**B-GEO-LOC, NO_ENTITY**} we sorted it out last time I'm like yea I'll tell him *covers eyes*wat informing am I doing #llowit
7	#aberdeen RT flook_firehose2010Polar Bear http://flook.it/c/1H1HZq Sun , 17 Oct 2010 at 10:28 am The Tunnels Carnegies{**B-GEO-LOC, NO_ENTITY**} Brae Aberdeen{**B-GEO-LOC, NO_ENTITY**} Un ...
8	&lt; 3 it RT Djcheapshot : Tonite I m DJing at Mai{**NO_ENTITY, B-FACILITY**} Tai{**NO_ENTITY, I-FACILITY**} in Long Beach{**B-GEO-LOC, I-GEO-LOC**} . I'm considering wearing MY TIE !! Get it ? My tie = Mai Tai ? No ? Sorry . Bye .
9	&quot; I gotta admit , Alex{**NO_ENTITY, B-PERSON**} sounds hot when he talks in spanish during the ' Alejandro{**NO_ENTITY, B-OTHER**} ' Cover &quot; -via someone 's tumblr{**NO_ENTITY, B-COMPANY**} I'm pleased to have introduced TheSmokingGunn to twitter{**NO_ENTITY, B-COMPANY**} . May he become as inane as me .
10	Before I proceed into the paradise , let 's not forget the Princess{**NO_ENTITY, B-MOVIE**} Lover{**NO_ENTITY, I-MOVIE**} OVA{**NO_ENTITY, I-MOVIE**} 1{**NO_ENTITY, I-MOVIE**} teaser pic , SFW{**B-GEO-LOC, NO_ENTITY**} http://yfrog.com/0fg2kfj

Table 2: Ten examples of mispredictted sentences. In each bracket, the left is the entity type predicted by model, and the right one is the correct entity type.

B Dataset Statistic

We show the statistic of all the datasets used in this paper. The three informal text datasets Ritter11-NER, Ritter11-POS and Ritter11-CHUNK are all created by Ritter et al. (2011). However, different research work has been using different name for these datasets. Here we name each dataset as the concatenation of the most used name “Ritter11” and the name of the task.

Task Type	Category	Dataset	Train Tokens	Dev Tokens	Test Tokens
NER	Formal	Ontonote-nw	848,220	144,319	49,235
	Informal	Ritter11-NER	37,098	4,461	4,730
POS Tagging	Formal	PTB 2003	912,344	131,768	129,654
	Informal	Ritter11-POS	10,857	2,242	2,291
Chunking	Formal	CoNLL 2000	211,727	-	47,377
	Informal	Ritter11-CHUNK	10,610	2,309	2,292

Table 3: Dataset statistics.

C Hyper-parameters and Training process

We introduce the detail of the experiment in this section for the reproduction of our results. Max training epoch is 20 for all LSTM models and 10 epochs for all BERT models. Adam optimizer with $\beta 1$ as 0.9, $\beta 2$ as 0.999, L2 weight decay as 0 is used for all LSTM models. The learning rate for all LSTM model is chosen between 1e-2 to 1e-4. AdamW (Loshchilov and Hutter, 2019) with $\beta 1$ as 0.9, $\beta 2$ as 0.999, L2 weight decay as 0.01 is used for all BERT models. Batch size in all LSTM and BERT models is set to be 8. The warmup ratio is set to be 0.1 for all LSTM and BERT models. For the INIT transfer learning setting, we pick the model that achieves the highest performance as a source model. For MULT transfer learning, the ratio of source data among the mixed data is in range (0.1, 0.9). In detail, the ratio 0.4 for NER task, 0.5 for Chunking task, 0.5 for POS Tagging task. For data annealing setting, within our appropriate range set of (0.9, 0.99), we find the data annealing constantly outperforms other transfer learning methods and the state-of-the-art method. We set α to be 0.95 and γ to be 0.9 for NER task, α to be 0.99 and γ to be 0.95 for Chunking task, α to be 0.95 and γ to be 0.95 for POS Tagging task. All the hyper-parameters are tuned on the development set of the corresponding dataset. The results are reported on the test set.