事前学習済み Transformer モデルのための注意教師付き Few-shot データの蒸留

前川 在　小林 尚輝　船越 孝太郎　奥村 学
東京工業大学
{maekawa, kobayasi, funakoshi, oku}@lr.pi.titech.ac.jp

概要

本稿では，データセットに含まれる知識を蒸留することで，ニューラルネットワークを効率的に学習可能な少量の合成サンプルを構築するデータセット蒸留に取り組む．我々は，事前学習済み Transformer モデルを効果的にファインチューニング可能な Few-shot データの獲得のために，Transformer の注意機構における注意確率の教師ラベルを導入する．実験では，4 つの異なる言語処理タスクのデータセットに対して，BERT を高い性能でファインチューニングする注意教師付き Few-shot データが構築可能であることを示した．

1　はじめに

深層学習モデルは，大規模なニューラルネットワークを大量のデータで学習することで，自然言語処理を含む様々な分野で高い性能を達成している．しかし，深層学習モデルの学習には，学習時間，計算資源，消費エネルギーなどを含む膨大なコストがかかる．そこで，この学習コストを削減するために，学習後のモデルの性能を維持したまま，学習用データセットを縮小する試みが行われている．

それらの従来研究の多くは，データ選択に基づく手法を用いている．データ選択では，クラスタ中心 [1]，多様性 [2]，モデルの尤度 [3] などのヒューリスティックスに基づいて，学習効果の高いサンプルの部分集合を抽出する．データ選択は，効率的かつ安定した高い学習性能を実現しており，能動学習 [1] や継続学習 [2] などにも応用されている．しかし，データ選択による手法は，元のデータセットに学習効果の高い代表的なサンプルが存在するという仮定に基づいているため，その性能は明らかに制限される．

近年，学習用データセットを縮小するための新しいアプローチとして，データセット蒸留 [4] が提案されている．データセット蒸留は，モデルを効率的に学習するように最適化された合成サンプルからなる蒸留データの獲得を目的とする．データセット蒸留は，学習サンプルを直接最適化することで，データ選択よりもはるかに少量のデータに圧縮する．データセット蒸留は，特に画像分野を中心に近年関心が高まっており [4, 5, 6, 7, 8, 9]，それらは理論的な興味からだけでなく，ニューラルアーキテクチャ/ハイパーパラメータ探索 [10]，継続学習 [11, 12]，連合学習 [13, 14]，データのプライバシー保護 [15, 16] など，応用の観点でも注目されている．

一方で，データセット蒸留に関する既存研究のほとんどが画像データセットを対象としており，言語処理タスクを扱った研究はごくわずかである．Sucholutsky ら [6] と Li ら [17] は，テキストの代わりに単語埋め込みのレベルで蒸留データを構築することで，データセット蒸留を離散データであるテキストデータセットに適用した．しかし，これらの研究は，CNN や RNN ベースのニューラルネットワークを対象としており，近年の言語処理タスクにおいて標準となっている事前学習済み Transformer モデルを対象とした研究は，我々が知る限りまだない．そこで本稿では，言語処理タスクに対して，事前学習済み Transformer モデルを効果的にファインチューニング可能な Few-shot データの獲得に取り組む．

この目的のために，我々は Transformer の核である注意機構に着目する [18]．従来研究 [19, 20, 21] では，注意確率の教師を利用することで，モデルを効果的に学習する手法が提案されている．本稿では，これらの手法に着想を得て，蒸留データの各サンプルに対して注意教師ラベルを付与することで，蒸留データによる Transformer の学習効果の向上を図る．

実験では，AGNews，SST-2，QNLI，MRPC の 4 つのテキスト分類タスクに対して，BERT [22] をファ

インチューニングするための注意教師付き Few-shot データセットを構築した．その結果，注意教師ラベルの利用による大幅な性能改善が見られ，BERT を高い性能でファインチューニング可能であることを示した．具体例として，4 クラスのニュース記事分類タスクである AGNews において，我々が構築した各クラス 1 サンプルのみで構成される蒸留データは，BERT を 1 回の勾配更新のみでファインチューニングし，元の学習データセット全体で学習した場合の 98.5%の性能である最大 93.2%の分類精度を達成した．

2 提案手法

2.1 データセット蒸留

データセット蒸留のアルゴリズムとして，Wang ら [4] が提案した最適化手法を適用する．この手法は，メタ学習 [23] で用いられている手法と同様に，2 次勾配を計算することで蒸留データを勾配法により直接最適化する．

元の学習データセットを $D = \{(x_i, y_i)\}_{i=1}^{N}$ とする．ただし (x_i, y_i) は入力と出力ラベルのペアである．このとき，データセット蒸留の目標は，はじめにランダムに初期化される蒸留データ $\tilde{D} = \{(\tilde{x}_i, \tilde{y}_i)\}_{i=1}^{M}$ $(M \ll N)$ を最適化することである．

モデルパラメータ θ は，蒸留データのミニバッチ $(\tilde{\boldsymbol{x}}_t, \tilde{\boldsymbol{y}}_t)$ を用いて以下の式で更新される．

$$\begin{aligned} \theta_{t+1} &= \theta_t - \tilde{\eta} \nabla_{\theta_t} L_{\text{task}} \\ \text{s.t.} \quad L_{\text{task}} &= l(\tilde{\boldsymbol{x}}_t, \tilde{\boldsymbol{y}}_t, \theta_t) \end{aligned} \tag{1}$$

ただし，$l()$ は二階微分可能な損失関数であり，$\tilde{\eta}$ は $\tilde{D}$ とともに最適化されるモデルの学習率である．θ の初期値を θ_0 とすると，蒸留データによる学習後のモデルは以下のように書ける．

$$\theta_T = F(\theta_0; \tilde{D}, \tilde{\eta}, T) \tag{2}$$

ただし，$F()$ は T ステップの勾配更新 (式 1) からなるモデルの学習過程を表す．

データセット蒸留の目的は，蒸留データで学習したこの θ_T が真のデータに対して高い性能を発揮することであると考えられるので，蒸留データ $\tilde{D}$ の最適化における目的関数 L_{distill} は以下のように計算できる．

$$L_{\text{distill}}(\tilde{D}, \tilde{\eta}; \theta_0) \coloneqq l(\boldsymbol{x}_s, \boldsymbol{y}_s, \theta_T) \tag{3}$$

$$= l(\boldsymbol{x}_s, \boldsymbol{y}_s, F(\theta_0; \tilde{D}, \tilde{\eta}, T)) \tag{4}$$

ただし，$(\boldsymbol{x}_s, \boldsymbol{y}_s)$ は元の学習データセットのミニバッチである．

したがって，データセット蒸留における最適化は以下のように定式化できる．

$$\tilde{D}^*, \tilde{\eta}^* = \arg\min_{\tilde{D}, \tilde{\eta}} \mathbb{E}_{\theta_0 \sim p(\theta_0)} \left[L_{\text{distill}}(\tilde{D}, \tilde{\eta}; \theta_0) \right] \tag{5}$$

ただし，$p(\theta_0)$ はモデルパラメータ θ の初期値の分布である．

この目的関数 L_{distill} に対して，勾配降下法を用いて蒸留データ $\tilde{D}$ を最適化する．しかし，テキストデータは離散的であるため，勾配降下法による最適化アルゴリズムを直接適用することは困難である．そこで本研究では，従来研究 [6, 17] に着想を得て，テキストの代わりに，単語埋め込みベクトルの系列を蒸留データの入力として利用する．単語埋め込みベクトルを利用することで，L_{distill} は蒸留データに関して微分可能となり，勾配降下法を用いた最適化が適用できる．

2.2 ソフトラベル

一般的に真のデータセットの出力ラベルには，単一クラスを示す one-hot ベクトルで表される，離散的なハードラベルが用いられる．しかし，蒸留データでは，出力ラベルをソフトなラベルとして，入力とともに最適化することが可能である．ソフトラベルを利用することで，ハードラベルを用いる場合よりも，蒸留データの各サンプルに多くの情報を持たせることが可能となる．本研究では，従来研究 [6, 7] に従い，ソフトラベルを one-hot ベクトルの値で初期化し，任意の実数値を取れるようにすることで，入力の単語埋め込みベクトルとともに勾配法を用いて最適化する．

2.3 注意教師ラベル

蒸留データによる Transformer の効果的な学習のために，本研究では，注意教師ラベルを導入する．注意教師ラベルは，データセットの知識を入力系列の各単語に対する注意確率として蒸留し，Transformer の自己注意機構を直接誘導することで効率的なモデルの学習を実現する．

従来研究 [21] に着想を得て，蒸留データの各サンプルに対して，Transformer の各層の注意機構の各ヘッドに対応する注意教師ラベルを付与し，モデルの注意確率との間の Kullback-Leibler (KL) ダイバージェンス (D_{KL}) を計算する．これより，注意確率に

表 1 データセットの概要．C はクラス数を示す．

Dataset	Task	Metric	C	# Train
AGNews	news classification	acc.	4	120k
SST-2	sentiment	acc.	2	67k
QNLI	QA/NLI	acc.	2	105k
MRPC	paraphrase	acc./F1	2	3.7k

関する損失 L_{attn} は次のように計算される．

$$L_{\text{attn}} = \frac{1}{K}\sum_{k=1}^{K}\frac{1}{H}\sum_{h=1}^{H} D_{\text{KL}}(\tilde{A}_{k,h}||A_{k,h}(\theta)) \quad (6)$$

ただし，$\tilde{A}_{k,h}$ と $A_{k,h}(\theta)$ は k 層目の注意機構の h 番目のヘッドに対応する，注意教師ラベルとモデルの注意確率をそれぞれ表す．本研究では，蒸留データのデータサイズの観点から，注意教師ラベルを文頭の `[CLS]` トークンにのみ適用した．

モデルは，タスクの損失 L_{task} と注意確率に関する損失 L_{attn} を同時に最小化するように学習される．これより，モデルパラメータ θ の勾配更新は，式 1 から次のようになる．

$$\theta_{t+1} = \theta_t - \tilde{\eta}\nabla_{\theta_t}(L_{\text{task}} + \lambda L_{\text{attn}}) \quad (7)$$

ただし，λ は L_{attn} のバランスを調整するためのハイパーパラメータである．

注意教師ラベル $\tilde{A}$ は，任意の実数値を取るベクトルとして，単語埋め込みベクトルとソフトラベルとともに最適化され，Softmax 関数を適用することで確率分布に変換して KL ダイバージェンスを計算する．

3 実験

3.1 実験設定

データセット データセット蒸留の性能を評価するために，ニュース記事分類タスクである AGNews [24] と，GLUE [25] から 3 つの多様な分類タスク (SST-2，QNLI，MRPC) を用いた．各データセットにおける評価指標については，AGNews は accuracy，その他のデータセットは GLUE の設定に従った．GLUE のデータセットについては，test セットが利用できないため development セットで評価を行った．各データセットの概要は表 1 に示す．

モデル 各データセットに対して，$\text{BERT}_{\text{BASE}}$ [22] をファインチューニングするための蒸留データセットを構築した．[22] の設定に従い，文頭の `[CLS]` の埋め込みに対して，埋め込み次元からクラス数次元に線形変換するための分類層と Softmax 関数を適用することで，各クラスの確率を計算する．Wang ら [4] に従い，蒸留データセットの学習時と評価時の両方においてモデルの Dropout は無効化した．これはモデルの学習における不確実性を避けることで学習の安定化を図るためである．

表 2 1-shot / 1-step の設定における実験結果．‘HL’，‘SL’，‘AL’ はそれぞれハードラベル，ソフトラベル，注意教師ラベルを示す．‘Majority’ は最頻クラスを予測するベースラインである．*のついたデータセット全体で学習したモデルの性能は，[22] から引用した．

	AGNews	SST-2	QNLI	MRPC
Majority	25.0	50.9	50.5	74.8
HL	$87.4_{\pm 1.8}$	$81.6_{\pm 2.4}$	$68.6_{\pm 2.5}$	$74.8_{\pm 0.0}$
SL	$88.4_{\pm 0.9}$	$82.5_{\pm 1.6}$	$76.4_{\pm 0.8}$	$74.8_{\pm 0.0}$
HL + AL	$\mathbf{93.2}_{\pm 0.1}$	$\mathbf{90.1}_{\pm 0.3}$	$85.9_{\pm 0.1}$	$76.4_{\pm 0.8}$
SL + AL	$93.0_{\pm 0.1}$	$89.0_{\pm 0.2}$	$\mathbf{86.4}_{\pm 0.1}$	$\mathbf{78.8}_{\pm 0.7}$
Full dataset	94.6	92.7*	91.8*	88.6*

ハイパーパラメータ 全ての蒸留データセットを，学習率 $\alpha \in \{1e^{-3}, 1e^{-2}, 1e^{-1}\}$ の Adam [26] を用いて 30 epoch ずつ学習した．蒸留データとともに最適化されるモデルの学習率 $\tilde{\eta}$ は $\{1e^{-2}, 1e^{-1}\}$ のいずれかの値で初期化した．α と $\tilde{\eta}$ の組み合わせのうち最も良い性能の結果を報告する．ただし，探索する α と $\tilde{\eta}$ の値の粒度は粗く，評価データに対する過適合の心配はない．式 7 における λ については，事前実験により 1.0 に設定した．

評価方法 蒸留した Few-shot データセットを用いて，BERT を 100 回ファインチューニングし，その性能の平均と標準偏差を報告する．ただし，追加した最後の分類層の重みパラメータは毎回異なる値で初期化した．

3.2 実験結果

3.2.1 1-shot / 1-step の実験結果

まず，各クラス 1 サンプルのみかつ 1 回の勾配更新のみで BERT をファインチューニングする設定において，ハードラベルとソフトラベル，及び注意教師ラベルの有無について蒸留データセットの性能を比較した結果を表 2 に示す．ハードラベルのみ，すなわち単語埋め込みベクトルのみを最適化した蒸留データセットでも，AGNews，SST-2，QNLI において，元のデータセット全体で学習した場合のそれぞれ 92.4%，88.0%，74.7%の性能を達成した．ま

た，これらの性能はソフトラベルを利用することでさらに改善され，特に QNLI では 8 ポイント近い性能向上を示した．一方，MRPC については，ソフトラベルの適用に関わらず，蒸留データセットは最頻クラスを予測した場合と同等の性能しか得られなかった．

注意教師ラベルを適用した場合，蒸留データセットの性能は 4 つの全てのタスクにおいて大幅に向上し，その効果はソフトラベルよりもはるかに大きいことが確認された．注意教師ラベルを適用した我々の蒸留データセットは，AGNews，SST-2，QNLI，MRPC のそれぞれに対して，元の学習データセット全体で学習した場合の最大 98.5，97.2，94.1，88.9%の性能を達成した．これより，注意教師ラベルを用いて元のデータセットの知識を注意確率として抽出することで，それらを Transformer モデルに効率的に伝達させることが可能であると考えられる．

また，今回使用した 4 つのデータセット間で性能を比較すると，データセット蒸留は，AGNews や SST-2 のような比較的単純な分類タスクでは非常に良好な性能を発揮する一方で，QNLI や MRPC のような 2 文間の関係の理解を必要とするようなタスクでは，性能がある程度制限された．特に MRPC では，注意教師ラベルを適用することである程度性能改善が見られるものの，その他の 3 つのタスクと比較して，元のデータセット全体で学習した場合との性能差が見られた．この原因として，MRPC データセットにおけるクラス間のサンプル数の不均衡が蒸留データセットの最適化を困難にしている可能性が考えられる．従って，アップサンプリングやダウンサンプリングなどにより，元のデータの不均衡に対処することで，性能改善の余地があると考える．

3.2.2 Multiple-step の実験結果

次に，複数回の勾配更新を用いて BERT をファインチューニングする設定において，蒸留データセットの性能を比較した．我々は，全ての勾配更新で同じ蒸留データを使用する場合と，各勾配更新ごとに異なる蒸留データを使用する場合のそれぞれの設定について実験を行った．ただし，いずれの設定においても，各勾配更新には各クラス 1 サンプルずつの蒸留データを用いる．本実験では，全ての蒸留データセットにおいてソフトラベルと注意教師ラベルを適用した．

実験結果を表 3 に示す．まず全ての勾配更新で同じ蒸留データを用いた場合，勾配更新 1 回のみでファインチューニングする場合よりも性能が劣化した．一方で，各勾配更新ごとに異なる蒸留データを用いることでその性能は改善され，勾配更新 1 回のみの設定における性能も上回った．これは，複数回の勾配更新において，各勾配更新ごとに蒸留データに求められる役割が異なっており，全ての勾配更新に対して有効であるような汎用的なデータを獲得することが困難であることを示唆している．

表 3 Multiple-step の実験結果．“**# shot**” は蒸留データに含まれる各クラスのサンプル数，“**# step**” は勾配更新回数を示す．ただし，すべての設定において，各勾配更新には各クラス 1 サンプルずつの蒸留データが使用される．

# shot	# step	AGNews	SST-2	QNLI	MRPC
Single-step setting					
1	1	$93.0_{\pm 0.1}$	$89.0_{\pm 0.2}$	$86.4_{\pm 0.1}$	$78.8_{\pm 0.7}$
Same distilled data for each step					
1	3	$93.0_{\pm 0.1}$	$89.8_{\pm 0.4}$	$84.2_{\pm 0.4}$	$74.8_{\pm 0.0}$
1	5	$92.1_{\pm 0.1}$	$85.8_{\pm 0.4}$	$85.9_{\pm 0.1}$	$74.8_{\pm 0.0}$
Different distilled data for each step					
3	3	$92.5_{\pm 0.1}$	$90.4_{\pm 0.2}$	$\mathbf{87.0}_{\pm 0.1}$	$\mathbf{80.3}_{\pm 0.8}$
5	5	$\mathbf{93.1}_{\pm 0.1}$	$\mathbf{90.7}_{\pm 0.2}$	$86.1_{\pm 0.1}$	$76.5_{\pm 0.8}$

本研究で利用したデータセット蒸留のための最適化手法では，全ての勾配更新を通した誤差逆伝播によって 2 次勾配を計算する必要があるため，勾配更新回数 T に応じて必要となるメモリや計算コストが線形に増加する．そのため，我々の実験では勾配更新回数を 5 以上に増加させることが困難であった．これはデータセット蒸留における明らかな課題であり，より複雑で困難なタスクのデータセットや，ファインチューニングではなくゼロからのモデルの学習に対してデータセット蒸留を適用するためには解決する必要がある．

4 おわりに

本稿では，事前学習済み Transformer モデルを効果的に学習するための，言語処理タスクのデータセットの蒸留に取り組んだ．我々は，Transformer の注意機構における注意確率の教師ラベルを蒸留データに適用することで，Transformer モデルに対する蒸留データセットの性能向上を図った．複数の言語処理タスクのデータセットを対象とした実験の結果，我々の蒸留データセットは各クラス 1 サンプルのみでも良好な性能を達成した．さらに我々が提案した注意教師ラベルを利用することで，その性能は全てのタスクにおいて大幅な性能改善を示した．

参考文献

[1] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In **6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings**. OpenReview.net, 2018.

[2] Rahaf Aljundi, Min Lin, Baptiste Goujaud, and Yoshua Bengio. Gradient based sample selection for online continual learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, **Advances in Neural Information Processing Systems**, Vol. 32. Curran Associates, Inc., 2019.

[3] Robert C. Moore and William Lewis. Intelligent selection of language model training data. In **Proceedings of the ACL 2010 Conference Short Papers**, pp. 220–224, Uppsala, Sweden, July 2010. Association for Computational Linguistics.

[4] Tongzhou Wang, Jun-Yan Zhu, Antonio Torralba, and Alexei A. Efros. Dataset distillation. **CoRR**, Vol. abs/1811.10959, , 2018.

[5] Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. Dataset condensation with gradient matching. In **9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021**. OpenReview.net, 2021.

[6] Ilia Sucholutsky and Matthias Schonlau. Soft-label dataset distillation and text dataset distillation. In **2021 International Joint Conference on Neural Networks (IJCNN)**, pp. 1–8, 2021.

[7] Ondrej Bohdal, Yongxin Yang, and Timothy M. Hospedales. Flexible dataset distillation: Learn labels instead of images. **CoRR**, Vol. abs/2006.08572, , 2020.

[8] Kai Wang, Bo Zhao, Xiangyu Peng, Zheng Zhu, Shuo Yang, Shuo Wang, Guan Huang, Hakan Bilen, Xinchao Wang, and Yang You. Cafe: Learning to condense dataset by aligning features. In **2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)**, pp. 12186–12195, 2022.

[9] George Cazenavette, Tongzhou Wang, Antonio Torralba, Alexei A. Efros, and Jun-Yan Zhu. Dataset distillation by matching training trajectories. In **IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2022, New Orleans, LA, USA, June 19-20, 2022**, pp. 4749–4758. IEEE, 2022.

[10] Felipe Petroski Such, Aditya Rawal, Joel Lehman, Kenneth Stanley, and Jeffrey Clune. Generative teaching networks: Accelerating neural architecture search by learning to generate synthetic training data. In Hal Daumé III and Aarti Singh, editors, **Proceedings of the 37th International Conference on Machine Learning**, Vol. 119 of **Proceedings of Machine Learning Research**, pp. 9206–9216. PMLR, 13–18 Jul 2020.

[11] Wojciech Masarczyk and Ivona Tautkute. Reducing catastrophic forgetting with learning on synthetic data. In **2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2020, Seattle, WA, USA, June 14-19, 2020**, pp. 1019–1024. Computer Vision Foundation / IEEE, 2020.

[12] Andrea Rosasco, Antonio Carta, Andrea Cossu, Vincenzo Lomonaco, and Davide Bacciu. Distilled replay: Overcoming forgetting through synthetic samples. In Fabio Cuzzolin, Kevin Cannons, and Vincenzo Lomonaco, editors, **Continual Semi-Supervised Learning**, pp. 104–117, Cham, 2022. Springer International Publishing.

[13] Jack Goetz and Ambuj Tewari. Federated learning via synthetic data. **CoRR**, Vol. abs/2008.04489, , 2020.

[14] Yanlin Zhou, George Pu, Xiyao Ma, Xiaolin Li, and Dapeng Wu. Distilled one-shot federated learning. **CoRR**, Vol. abs/2009.07999, , 2020.

[15] Guang Li, Ren Togo, Takahiro Ogawa, and Miki Haseyama. Soft-label anonymous gastric x-ray image distillation. In **2020 IEEE International Conference on Image Processing (ICIP)**, pp. 305–309, 2020.

[16] Tian Dong, Bo Zhao, and Lingjuan Lyu. Privacy for free: How does dataset condensation help privacy? In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, **Proceedings of the 39th International Conference on Machine Learning**, Vol. 162 of **Proceedings of Machine Learning Research**, pp. 5378–5396. PMLR, 17–23 Jul 2022.

[17] Yongqi Li and Wenjie Li. Data distillation for text classification. **CoRR**, Vol. abs/2104.08448, , 2021.

[18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, L ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, **Advances in Neural Information Processing Systems**, Vol. 30. Curran Associates, Inc., 2017.

[19] Lemao Liu, Masao Utiyama, Andrew Finch, and Eiichiro Sumita. Neural machine translation with supervised attention. In **Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers**, pp. 3093–3102, Osaka, Japan, December 2016. The COLING 2016 Organizing Committee.

[20] Haitao Mi, Zhiguo Wang, and Abe Ittycheriah. Supervised attentions for neural machine translation. In **Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing**, pp. 2283–2288, Austin, Texas, November 2016. Association for Computational Linguistics.

[21] Gustavo Aguilar, Yuan Ling, Yu Zhang, Benjamin Yao, Xing Fan, and Chenlei Guo. Knowledge distillation from internal representations. **Proceedings of the AAAI Conference on Artificial Intelligence**, Vol. 34, No. 05, pp. 7350–7357, Apr. 2020.

[22] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)**, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[23] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In Doina Precup and Yee Whye Teh, editors, **Proceedings of the 34th International Conference on Machine Learning**, Vol. 70 of **Proceedings of Machine Learning Research**, pp. 1126–1135. PMLR, 06–11 Aug 2017.

[24] Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett, editors, **Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada**, pp. 649–657, 2015.

[25] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In **Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP**, pp. 353–355, Brussels, Belgium, November 2018. Association for Computational Linguistics.

[26] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, **3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings**, 2015.