CE-003

階層的知識蒸留による発話単位系列ラベリングのモデル軽量化
Model Compaction for Dialogue Sequence Labeling by Hierarchical Knowledge Distillation

折橋 翔太† 山﨑 善啓† 牧島 直輝† 庵 愛†
高島 瑛彦† 田中 智大† 増村 亮†
Shota Orihashi Yoshihiro Yamazaki Naoki Makishima Mana Ihori
Akihiko Takashima Tomohiro Tanaka Ryo Masumura

1. はじめに

近年、音声認識技術の進展に伴い、人と人との対話を理解し、得られる言語情報を活用することへの期待が高まっている。例えば、コンタクトセンタにおける電話での対話を理解することで、顧客のニーズやコンタクトセンタの課題を発見するサービスが開発されている[1]。

本稿では、対話の理解において重要な、発話の系列で構成された対話ドキュメントに対し発話単位でラベルを与える発話単位系列ラベリングを扱う。発話単位系列ラベリングは、対話ドキュメントを入力として、各発話に対するラベルを推定する教師あり学習タスクであり、対話ドキュメントのトピックセグメンテーションや対話シーンの推定に有用である[2–4]。発話単位系列ラベリングでは、誰が何をどの順序で話したかを精緻に捉えることが求められることから、発話内の短期文脈を理解するための発話内ネットワークと、発話間の長期文脈を理解するための発話間ネットワークを階層的に用いるモデリング手法が有効である[2,3]。

高い精度での発話単位系列ラベリングを実現するためには、通常、発話内ネットワークと発話間ネットワークのそれぞれに、学習可能なパラメータを多数持たせる必要がある。ただし、このような大規模なモデルを用いた推論には、潤沢な計算環境が要求される。しかし、複数の推論を並列処理する場合や、モバイル端末のように計算能力が低いデバイスで推論する場合、このような潤沢な計算環境を用意することは困難である。

我々は、大規模なモデルを用いることによる上記の課題を解決するため、大規模で高性能な教師モデルに獲得された知識を抽出し活用することで、パラメータが少なく軽量な生徒モデルを効率的に学習する知識蒸留[5–7]に着目する。近年、知識蒸留は、ニューラル機械翻訳モデルの軽量化[8]や BERT[9]のモデル軽量化[10]など、自然言語処理の分野において複数の成功例が報告されている。知識蒸留の利点は、教師モデルの動作を模倣するように生徒モデルを学習できることである。発話単位系列ラベリングは、対話ドキュメントの文脈を精緻に捉えるべきタスクであるため、性能を維持しながらモデルを軽量化するためには、生徒モデルが教師モデルの動作を忠実に模倣する必要がある。このことから、知識蒸留は発話単位系列ラベリングの軽量化に有望と考えられるが、これまでに発話単位系列ラベリングのような階層的なモデルに対する効果的な知識蒸留の手法は確立されていない。

本稿では、発話単位系列ラベリングのための新たな知識蒸留手法を提案する。提案手法では、大規模な教師モデルにより捉えられる発話内の短期文脈と発話間の長期文脈に関する知識を抽出し、それらを生徒モデルに与えることで、軽量な生徒モデルを効率的に学習することを狙う。このため、提案する階層的知識蒸留では、従来の知識蒸留のように、生徒モデルの出力するラベルの確率分布が教師モデルの出力するラベルの確率分布を模倣するように学習するだけでなく、生徒モデルの発話内ネットワークと発話間ネットワークの出力が教師モデルの各ネットワークの出力を模倣するように学習する。階層的知識蒸留により、教師モデルに獲得された発話内および発話間の文脈を捉える機能を失うことなく、軽量な生徒モデルを学習することを可能にし、分類精度が高い生徒モデルを学習する。なお、提案手法は発話単位系列ラベリングのための知識蒸留を実現する初めての手法である。コンタクトセンタの模擬的な応対データセットを用いた応対シーン推定による評価実験で、提案手法の有効性を示す。

2. 関連研究

2.1 発話単位系列ラベリング

発話単位系列ラベリングは、対話ドキュメントの各発話に対するラベルを推定する教師あり学習タスクである。発話内および発話間の文脈を効率的に捉えるため、発話内ネットワークと発話間ネットワークで構成される階層的なモデルを用いることが有効であり[2,3]、効果的な事前学習の手法も提案されている[4]。高い分類精度で発話単位系列ラベリングを実現するためには、学習可能なパラメータを多数持つ大規模なモデルを用いる必要がある。本稿では、分類精度は高いがパラメータが少ないモデルを学習するため、発話単位系列ラベリングに知識蒸留を導入する。

2.2 知識蒸留

知識蒸留は、大規模で高性能な教師モデルの知識を活用することで、性能を大幅に低下させることなく、軽量な生徒モデルを効率的に学習する手法である[5]。代表的な手法では、ソフトターゲット損失を利用して、生徒モデルの出力するラベルの確率分布が教師モデルの出力するラベルの確率分布に近づくように学習する[6]。また、生徒モデルの隠れ層による中間出力が教師モデルの隠れ層による中間出力に近づくように、生徒モデルを学習する手法も提案されている[7]。近年では、自然言語処理の分野においても知識蒸留の成功例が複数報告されている[8,10]。本稿では、発話単位系列ラベリングに対する知識蒸留を提案する。教師モデルの持つ発話内および発話間の文脈を捉える機能を維持するため、提案手法では、生徒モデルの発話内ネットワークと発話間ネットワークの出力が教師モデルの各出力を模倣するように、生徒モデルを学習する。

† 日本電信電話株式会社,
NTT メディアインテリジェンス研究所
NTT Media Intelligence Laboratories, NTT Corporation

3. 発話単位系列ラベリング

対話ドキュメントに対する発話単位系列ラベリングについて、詳述する。発話単位系列ラベリングは、入力の発話系列$\boldsymbol{X}=\{x_1,\cdots,x_T\}$から、ニューラルネットワークを用いて、発話単位のラベル系列$\boldsymbol{Y}=\{y_1,\cdots,y_T\}$を推定するタスクである。ここで、$x_t$は$t$番目の発話である。また、$y_t\in\mathcal{Y}$は$t$番目のラベルであり、$\mathcal{Y}$はラベルのセットを表す。ラベルのセットはタスクに依存し、例えばコンタクトセンタにおける応対シーン推定では、ラベルは応対シーンに相当する。

本稿における発話単位系列ラベリングでは、対話の開始からt番目までの発話$\{x_1,\cdots,x_t\}$から、逐次t番目のラベルy_tを推定する。そのため、$\boldsymbol{\Theta}$をモデルパラメータとし、条件付き確率$P(y_t|x_1,\cdots,x_t;\boldsymbol{\Theta})$をモデル化する。$t$番目の発話に対する推定ラベル$\hat{y}_t$は、(1)式により導出できる。

$$\hat{y}_t=\underset{y_t\in\mathcal{Y}}{\operatorname{argmax}}\,P(y_t|x_1,\cdots,x_t;\boldsymbol{\Theta}) \tag{1}$$

本稿では、Transformer エンコーダ[9]や LSTM を用いて、$P(y_t|x_1,\cdots,x_t;\boldsymbol{\Theta})$をモデル化するものとする。図 1 に、発話単位系列ラベリングのモデル構造を示す。

発話内ネットワークでは、まず各トークンを連続的な固定長ベクトルに埋め込む。本稿では、t番目の発話x_tを、t番目の発話のトークン数K_tを用い、トークン系列$\{w_{t,1},\cdots,w_{T,K_T}\}$と表すものとする。このとき、$t$番目の発話における$k$番目のトークンに対する埋め込みベクトル$\boldsymbol{w}_{t,\mathrm{k}}$は、(2)式により得る。

$$\boldsymbol{w}_{t,\mathrm{k}}=\mathrm{Embedding}\left(w_{t,\mathrm{k}};\boldsymbol{\theta}^{w}\right) \tag{2}$$

ここで、Embedding()はトークンをベクトルに埋め込むための線形変換関数であり、$\boldsymbol{\theta}^{w}$は学習可能なモデルパラメータである。トークンに対する埋め込みベクトル$\boldsymbol{w}_{t,\mathrm{k}}$は、(3)式に従いベクトル$\boldsymbol{q}_{t,\mathrm{k}}$に変換する。

$$\boldsymbol{q}_{t,\mathrm{k}}=\mathrm{AddPosEnc}\left(\boldsymbol{w}_{t,\mathrm{k}}\right) \tag{3}$$

ここで、AddPosEnc()は位置埋め込みのための関数である。次に、L層の Transformer エンコーダブロックを通して、$\boldsymbol{Q}_t=\{\boldsymbol{q}_{t,1},\cdots,\boldsymbol{q}_{t,K_t}\}$から$\boldsymbol{R}_t^{(L)}=\left\{\boldsymbol{r}_{t,1}^{(L)},\cdots,\boldsymbol{r}_{t,K_t}^{(L)}\right\}$に、ベクトル変換を行う。$l$層目の Transformer エンコーダブロックでは、$l-1$層目のブロックの出力である$\boldsymbol{R}_t^{(l-1)}$から、(4)式に従いベクトル系列$\boldsymbol{R}_t^{(l)}$を得る。

$$\boldsymbol{R}_t^{(l)}=\mathrm{TransformerEnc}\left(\boldsymbol{R}_t^{(l-1)};\boldsymbol{\theta}^{r}\right) \tag{4}$$

ここで、$\boldsymbol{R}_t^{(0)}=\boldsymbol{Q}_t$であり、TransformerEnc()は Transformer エンコーダブロックの機能を表す関数である[9]。また、$\boldsymbol{\theta}^{r}$は学習可能なモデルパラメータである。続いて、トークンに対する変換後のベクトル系列$\boldsymbol{R}_t^{(L)}$を、Self-Attention 機構[11]を用いて、発話の特徴を表す中間表現の埋め込みベクトルに変換する。t番目の発話に対する中間表現の埋め込みベクトルは、(5)式により得る。

$$\boldsymbol{s}_t=\mathrm{SelfAttention}\left(\boldsymbol{r}_{t,1}^{(L)},\cdots,\boldsymbol{r}_{t,K_t}^{(L)};\boldsymbol{\theta}^{s}\right) \tag{5}$$

ここで、SelfAttention()は、入力された埋め込みベクトル系列を、Self-Attention 機構により固定長ベクトルに変換する関数であり、$\boldsymbol{\theta}^{s}$は学習可能なモデルパラメータである。

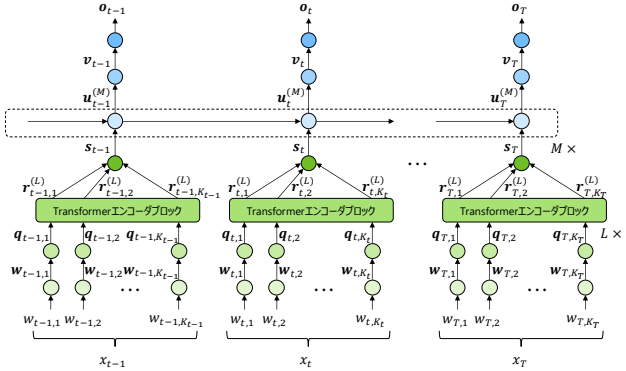

図 1　発話単位系列ラベリングのモデル構成

発話間ネットワークでは、対話の開始からt番目の発話までの対話コンテキストの情報を、中間表現としてベクトルに段階的に埋め込む。t番目の発話までの情報が埋め込まれたベクトル$\boldsymbol{u}_t^{(M)}$は、$\{\boldsymbol{s}_1,\cdots,\boldsymbol{s}_t\}$を参照し、$M$層の LSTM により導出する。$m$層目の LSTM では、$m-1$層目の LSTM の出力である$\left\{\boldsymbol{u}_1^{(m-1)},\cdots,\boldsymbol{u}_t^{(m-1)}\right\}$から、(6)式に従いベクトル系列$\boldsymbol{u}_t^{(m)}$を得る。

$$\boldsymbol{u}_t^{(m)}=\mathrm{LSTM}\left(\boldsymbol{u}_1^{(m-1)},\cdots,\boldsymbol{u}_t^{(m-1)};\boldsymbol{\theta}^{u}\right) \tag{6}$$

ここで、$\boldsymbol{u}_t^{(0)}=\boldsymbol{s}_t$であり、LSTM()は LSTM の機能を表す関数である。また、$\boldsymbol{\theta}^{u}$は学習可能なモデルパラメータである。

出力層では、t番目のラベルの推定確率$\boldsymbol{o}_t$を、ロジット$\boldsymbol{v}_t=\left[v_{t,1},\cdots,v_{t,|\mathcal{Y}|}\right]$を用いて、(7–8)式により得る。

$$\boldsymbol{v}_t=\mathrm{FeedForward}\left(\boldsymbol{u}_t^{(M)};\boldsymbol{\theta}^{v}\right) \tag{7}$$

$$\boldsymbol{o}_t=\mathrm{Softmax}(\boldsymbol{v}_t) \tag{8}$$

ここで、FeedForward()は全結合ニューラルネットワークであり、$\boldsymbol{\theta}^{v}$は学習可能なモデルパラメータである。また、Softmax()はソフトマックス関数であり、$\boldsymbol{o}_t$は$P(y_t|x_1,\cdots,x_t;\boldsymbol{\Theta})$に対応する。

モデルパラメータ$\boldsymbol{\Theta}=\{\boldsymbol{\theta}^{w},\boldsymbol{\theta}^{r},\boldsymbol{\theta}^{s},\boldsymbol{\theta}^{u},\boldsymbol{\theta}^{v}\}$は、データセット$\mathcal{D}=\{(X^1,\bar{Y}^1),\cdots,(X^N,\bar{Y}^N)\}$を用いて最適化する。ここで、$X^n$と$\bar{Y}^n$は、$n$番目の対話における、入力となる発話と正解ラベルの系列である。最適化に用いるクロスエントロピー損失は(9)式に従い、ハードターゲット損失と呼ぶ。

$$\mathcal{L}_{\mathrm{HT}}=-\frac{1}{N}\sum_{n=1}^{N}\left(\frac{1}{T_n}\sum_{t=1}^{T_n}\sum_{y\in\mathcal{Y}}\bar{o}_{t,y}^{n}\log o_{t,y}^{n}\right) \tag{9}$$

ここで、$\bar{\boldsymbol{o}}_t^{n}=\left[\bar{o}_{t,1}^{n},\cdots,\bar{o}_{t,|\mathcal{Y}|}^{n}\right]$と$\boldsymbol{o}_t^{n}=\left[o_{t,1}^{n},\cdots,o_{t,|\mathcal{Y}|}^{n}\right]$は、それぞれ$n$番目の対話における$t$番目の発話に対するラベルの正解確率と推定確率であり、T_nはn番目の対話の発話数である。なお、$\bar{\boldsymbol{o}}_t^{n}$は One-Hot ベクトルである。

4. 提案手法

発話単位系列ラベリングに対して知識蒸留を用いる提案手法について、詳細に述べる。提案手法である階層的知識蒸留は、大規模な教師モデルによって捉えられる発話内の短期文脈と発話間の長期文脈に関する知識を抽出し、それらを生徒モデルに与えることで、軽量な生徒モデルを効率的に学習することを狙う。このため提案手法は、ソフトタ

ーゲット損失[6]を用いて、生徒モデルの出力するラベルの確率分布が教師モデルの出力するラベルの確率分布に近づくように学習する。それだけでなく、提案手法は、生徒モデルの発話内ネットワークおよび発話間ネットワークの出力が教師モデルの各出力に近づくように、生徒モデルを学習する。教師モデルに獲得された発話内および発話間の文脈を捉える機能を、失うことなく生徒モデルに与えることで、提案手法は軽量で高い分類精度の生徒モデルを学習する。

図 2 に、提案手法の概要を示す。提案手法では、(9)式で定義されるハードターゲット損失と、ソフトターゲット損失、発話内文脈損失、発話間文脈損失の 4 つの損失を組み合わせた損失関数を用いて生徒モデルを最適化する。

4.1 ソフトターゲット損失

ソフトターゲット損失は、生徒モデルの出力するラベルの確率分布を教師モデルの出力するラベルの確率分布に近づけるための損失である[6]。ソフトターゲット損失の導出では、ロジット$\boldsymbol{v}_t$から確率分布$\boldsymbol{z}_t$を(10)式に従い得る。

$$\boldsymbol{z}_t = \mathrm{SoftmaxWithTemperature}(\boldsymbol{v}_t; \tau) \tag{10}$$

ここで、SoftmaxWithTemperature()は温度付きソフトマックス関数であり、τは温度を表すハイパーパラメータである[6]。このとき、ソフトターゲット損失は(11)式に従う。

$$\mathcal{L}_{\mathrm{ST}} = -\frac{\tau^2}{N}\sum_{n=1}^{N}\left(\frac{1}{T_n}\sum_{t=1}^{T_n}\sum_{y\in\mathcal{Y}}\tilde{z}_{t,y}^{n}\log z_{t,y}^{n}\right) \tag{11}$$

ここで、$\tilde{\boldsymbol{z}}_t^n = [\tilde{z}_{t,1}^n, \cdots, \tilde{z}_{t,|\mathcal{Y}|}^n]$と$\boldsymbol{z}_t^n = [z_{t,1}^n, \cdots, z_{t,|\mathcal{Y}|}^n]$は、それぞれ教師モデルと生徒モデルの出力する$n$番目の対話における$t$番目の発話に対するラベルの推定確率である。

4.2 発話内文脈損失

提案手法は、生徒モデルの発話内ネットワークが教師モデルの発話内ネットワークを模倣するよう、生徒モデルを学習する。このため、発話内文脈損失は、生徒モデルの発話内ネットワークの出力と、教師モデルの発話内ネットワークの出力の差で定義する。発話内文脈損失は、平均二乗誤差を用い(12)式に従う。

$$\mathcal{L}_{\mathrm{WU}} = \frac{1}{N}\sum_{n=1}^{N}\left(\frac{1}{T_n}\sum_{t=1}^{T_n}\|\tilde{\boldsymbol{s}}_t^n - \boldsymbol{s}_t^n\|_2^2\right) \tag{12}$$

ここで、$\tilde{\boldsymbol{s}}_t^n$と$\boldsymbol{s}_t^n$は、それぞれ教師モデルと生徒モデルの発話内ネットワークが出力するt番目の発話に対する中間表現の埋め込みベクトルである。なお、提案手法は$\tilde{\boldsymbol{s}}_t^n$と$\boldsymbol{s}_t^n$のサイズが等しいことを前提とする。

4.3 発話間文脈損失

提案手法は、生徒モデルの発話間ネットワークが教師モデルの発話間ネットワークを模倣するよう、生徒モデルを学習する。このため、発話間文脈損失は、生徒モデルの発話間ネットワークの出力と、教師モデルの発話間ネットワークの出力の差で定義する。発話間文脈損失は、平均二乗誤差を用い(13)式に従う。

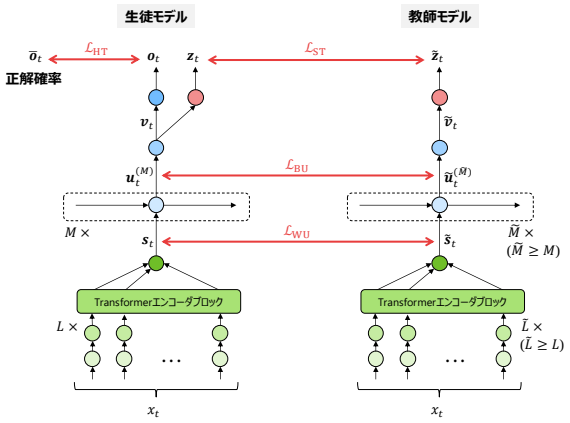

図 2　提案手法の概要

$$\mathcal{L}_{\mathrm{BU}} = \frac{1}{N}\sum_{n=1}^{N}\left(\frac{1}{T_n}\sum_{t=1}^{T_n}\left\|\tilde{\boldsymbol{u}}_t^{(\tilde{M}),n} - \boldsymbol{u}_t^{(M),n}\right\|_2^2\right) \tag{13}$$

ここで、$\tilde{\boldsymbol{u}}_t^{(\tilde{M}),n}$と$\boldsymbol{u}_t^{(M),n}$は、それぞれ教師モデルと生徒モデルの発話間ネットワークが出力するt番目の発話までの情報が埋め込まれたベクトルであり、$\tilde{M}$は教師モデルの発話間ネットワークの層数である。なお、提案手法は$\tilde{\boldsymbol{u}}_t^{(\tilde{M}),n}$と$\boldsymbol{u}_t^{(M),n}$のサイズが等しいことを前提とする。

4.4 学習

学習では、生徒モデルのパラメータ$\boldsymbol{\Theta}$を、データセット$\mathcal{D}$を用いて最適化する。損失関数は、(14)式に従う。

$$\mathcal{L} = \mathcal{L}_{\mathrm{HT}} + \lambda\mathcal{L}_{\mathrm{ST}} + \alpha\mathcal{L}_{\mathrm{WU}} + \beta\mathcal{L}_{\mathrm{BU}} \tag{14}$$

ここで、λおよびα、βはハイパーパラメータである。

5. 評価実験

5.1 データセット

コンタクトセンタの模擬的な応対データセットを用い、応対シーン推定タスクにより評価実験を行った。データセットの詳細を、表 1 に示す。応対データセットは、6 業種 361 通話で構成される。1 通話は 1 人のオペレータと 1 人のカスタマによる対話であり、すべての発話は人手で書き起こされたものである。各通話は、多様な日本語音声により学習した発話区間検出[12]を用いて発話単位に分割した。応対シーンのラベルは、オープニング、用件把握、用件対応、カスタマ情報把握、クロージングの 5 種類を、各発話に人手で一意に付与して用いた[3]。324 通話を学習用、37 通話をテスト用とした。

また、事前学習[4]を適用するため、Web から収集したラベル無しのテキストデータ約 5 億文と、様々な業種のコンタクトセンタから収集したラベル無しの通話データ約 4,000 通話を用いた。

5.2 実験条件

教師モデルは、ラベル無しの Web テキストデータとラベル無しの通話データを用い事前学習[4]を適用したうえで、ラベルありの応対データセットを用いてファインチューニングすることで学習した。提案手法を評価するため、生徒モデルは下記の 3 つの学習手法により学習した。

表 1　応対データセットの詳細

業種	通話数	発話数	単語数
金融	59	6,081	55,933
プロバイダ販売	57	3,815	47,668
地方自治体	73	5,617	48,998
通信販売	56	4,938	46,574
パソコン修理	55	6,263	55,101
携帯電話販売	61	5,738	51,061
合計	361	32,452	305,351

表 2　発話単位系列ラベリングモデルの詳細

		L	M	ユニット数	パラメータ数
教師モデル		8	2	2,048	13.11M
生徒モデル	S1	1	1	256	2.47M
	S2	2	2	512	3.65M

- **ベースライン**：ラベルありの応対データセットのみを用い、スクラッチから学習
- **事前学習**：教師モデルと同様の事前学習を適用してから、ラベルありの応対データセットによりファインチューニングすることで学習
- **知識蒸留**：ラベルありの応対データセットのみを用い、教師モデルから知識蒸留することで学習

実験では、サイズの異なる 2 種類の生徒モデル、S1 と S2 を用いた。実験に用いたモデルの詳細を表 2 に示す。表 2 において、LとMはそれぞれ発話内ネットワークと発話間ネットワークの層数を表し、ユニット数は Transformer エンコーダブロックの全結合層における中間出力次元数を表す。すべてのモデルで、単語トークンに対する埋め込みベクトルの次元数および LSTM のユニット数を 256 とした。また、Transformer エンコーダブロックにおいて、出力次元数を 256、マルチヘッド注意のヘッド数を 4 とした。教師モデルは、S1 と S2 に対し共通とした。

学習では、ミニバッチのサイズを 5 通話とし、オプティマイザとしてデフォルト設定の RAdam[13]を用いた。知識蒸留では、τ、λ、α、βを、それぞれ 5.0、0.1、0.05、0.05 とした。また、学習はモデルの初期パラメータを変更しながら 5 回ずつ行い、得られたモデルの平均分類精度により評価した。

5.3 実験結果

評価実験の結果を、表 3 に示す。表 3 において、1 行目は教師モデルによる理想的な分類精度を示す。2 行目は生徒モデルをスクラッチから学習した結果を、3 行目は事前学習を適用して生徒モデルを学習した結果を示す。1 行目と 2、3 行目の結果には差があり、これはパラメータ数を削減したことによる精度劣化である。4–7 行目は、知識蒸留による結果を示す。4 行目は、従来の知識蒸留[6]と同様のソフトターゲット損失とハードターゲット損失のみを用いた場合の結果であるが、ベースラインを下回っており、知識蒸留が機能していないことが分かる。5、6 行目は、発話内文脈損失と発話間文脈損失のいずれかを用いた場合の結果であるが、分類精度の向上は限定的である。7 行目は提案手法による結果であり、最も高い分類精度を示している。特に、S2 では教師モデルに匹敵する分類精度に達している。これは、提案手法が、教師モデルに獲得されている発話内および発話間の文脈を捉える機能を失うことなく、生徒モデルを学習しているためであると考えられる。これらの結果より、提案手法が生徒モデルの分類精度を向上させる手法として有効であることが示された。

表 3　分類精度による評価結果 (%)

		S1	S2
教師モデル		91.28	91.28
生徒モデル	ベースライン	87.42	88.24
	事前学習	89.38	89.82
	知識蒸留 w/o $\mathcal{L}_{\mathrm{WU}}, \mathcal{L}_{\mathrm{BU}}$	85.98	87.33
	知識蒸留 w/o $\mathcal{L}_{\mathrm{WU}}$	88.83	88.86
	知識蒸留 w/o $\mathcal{L}_{\mathrm{BU}}$	88.99	89.20
	知識蒸留	**90.13**	**91.10**

6. まとめ

本稿では、発話単位系列ラベリングのための新たな知識蒸留手法を提案した。提案した階層的知識蒸留では、大規模な教師モデルによって捉えられる発話内の短期文脈と発話間の長期文脈に関する知識を効率的に生徒モデルに学習させるため、発話内文脈損失と発話間文脈損失を導入し、生徒モデルが教師モデルをより精緻に模倣するよう学習した。評価実験の結果から、提案手法により生徒モデルの分類精度が向上することを確認した。

参考文献

[1] 長谷川隆明, 関口裕一郎, 山田節夫, 田本真詞, “オペレータの応対を支援する自動知識支援システム,” NTT 技術ジャーナル, vol. 31, no. 7, pp. 16–19 (2019).

[2] H. Kumar, A. Agarwal, R. Dasgupta, and S. Joshi, “Dialogue act sequence labeling using hierarchical encoder with CRF,” AAAI, pp. 3440–3447 (2018).

[3] R. Masumura, S. Yamada, T. Tanaka, A. Ando, H. Kamiyama, and Y. Aono, “Online call scene segmentation of contact center dialogues based on role aware hierarchical LSTM-RNNs,” APSIPA ASC, pp. 811–815 (2018).

[4] R. Masumura, N. Makishima, M. Ihori, A. Takashima, T. Tanaka, and S. Orihashi, “Large-context conversational representation learning: Self-supervised learning for conversational documents,” SLT, pp. 1012–1019 (2021).

[5] C. Bucilua, R. Caruana, and A. Niculescu-Mizil, “Model compression,” ACM SIGKDD, pp. 535–541 (2006).

[6] G. Hinton, O. Vinyals, and J. Dean, “Distilling the knowledge in a neural network,” NIPS Workshop (2014).

[7] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, “FitNets: Hints for thin deep nets,” ICLR (2015).

[8] X. Tan, Y. Ren, D. He, T. Qin, Z. Zhao, and T.-Y. Liu, “Multilingual neural machine translation with knowledge distillation,” ICLR (2019).

[9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” NIPS, pp. 5998–6008 (2017).

[10] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, “DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter,” NeurIPS Workshop (2019).

[11] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, “Hierarchical attention networks for document classification,” NAACL-HLT, pp. 1480–1489 (2016).

[12] F. Eyben, F. Weninger, S. Squartini, and B. Schuller, “Real-life voice activity detection with LSTM recurrent neural networks and an application to Hollywood movies,” ICASSP, pp. 483–487 (2013).

[13] L. Liu, H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and J. Han “On the variance of the adaptive learning rate and beyond,” ICLR (2020).