共有タスクへの結果提出を通した生徒モデルの共同教育手法

中山功太 [1,2]　栗田修平 [1]　小林暁雄 [3]
馬場雪乃 [1,4]　関根聡 [1]
[1] 理化学研究所 AIP　[2] 筑波大学大学院理工情報生命学術院
[3] 農業・食品産業技術総合研究機構　[4] 筑波大学システム情報系
{kouta.nakayama, shuhei.kurita, satoshi.sekine}@riken.jp
akio.kobayashi@naro.go.jp　baba@cs.tsukuba.ac.jp

概要

本論文では、共有タスクに提出されたシステム結果を用いて新規システムを構築する手法を提案する。本手法により、共有タスクはシステム開発における技術革新のみでなく、実際に自然言語処理アプリケーションに汎用可能なシステムの公開が可能となる。提案手法は知識蒸溜から着想を得ており、共有タスクの参加システムを教師として扱い、新規深層学習モデルを生徒として学習することで、参加システムの長所を生かしたシステムを構築する。本手法を日本語 Wikipedia からの属性抽出タスクである森羅 2019 へ提出されたシステム結果に対し適用したところ、最良のシステムよりも高性能であり、最良のシステム群と同等性能のシステムを獲得できた。本実験で用いたコードは公開してあり、更なる研究へ適用可能である。[1)]

1　はじめに

多くの参加者が共通のタスクに取り組むといった共有タスクは、自然言語処理技術の発展に大きく貢献している。だが、多くのタスクが参加者にシステム出力や手法説明のみの提出を要求しており、システム自体の提出が要求されることは稀である。そのため、多くのシステムが実際の自然言語処理アプリケーションで使用されることなく放棄されている。我々は共有タスクに提出されたシステムの多くは、たとえ最良の結果を取得しておらずとも分野全体の革新のため有用なリソースであると考えている。だが、システム自体の提出は、ライセンスや実行環境の整備といった点で非常に困難である。そのため本論文では、複数の参加者が提出したシステム結果を用いて参加システムの強みを活かした新規システムを構築する手法の提案を行う。

提案手法は知識蒸溜 [1, 2] から着想を得ており、タスク参加システムを教師として扱い、提出結果を通して生徒である新規深層学習モデルを学習することで、新規システム構築を行う。共有タスクの参加者が共同で単一の生徒を教えるといった形式をとる都合上、我々は本手法を「共同教育」と呼ぶ。共同教育は、共有タスクにおいて他の自然言語処理アプリケーションに汎用可能なシステムを公開でき、タスクにおいて最良結果を残せなかったチームを含め多くの努力を活用できるといった利点がある。

我々は共同教育を、日本語 Wikipedia からの属性値抽出タスクである森羅 2019 に適用する。本タスクは全ての Wikipedia ページの構造化を目指しており、参加者は評価データ以外の範囲のラベルなしデータに対する予測結果の提出も求められる。実験の結果、提案手法は共有タスクに提出された最良のシステムよりも高性能であり、最良のシステム群と同等性能のシステムを獲得できた。

2　関連研究

2.1　知識蒸溜

知識蒸溜 (Knowledge Distillation)[1, 2] は主に深層学習モデルの圧縮に用いられる手法であり、モデル性能を低下させず総パラメーター数を減らすことを目的としている。具体的には教師と呼ばれる大規模なモデルの結果を用いて、生徒と呼ばれる比較的小規模な新規モデルを学習することでモデルの圧縮を達成する。多くの場合、教師は学習データを用いて学習され、その際利用された学習データは生徒の学習にも転用される。知識蒸留の手法は応答ベース [1, 2] と特徴ベース [3] に分けることができる。前者は教師の出力を、後者は教師の内部パラメーター

1）https://github.com/k141303/co_teaching_scheme

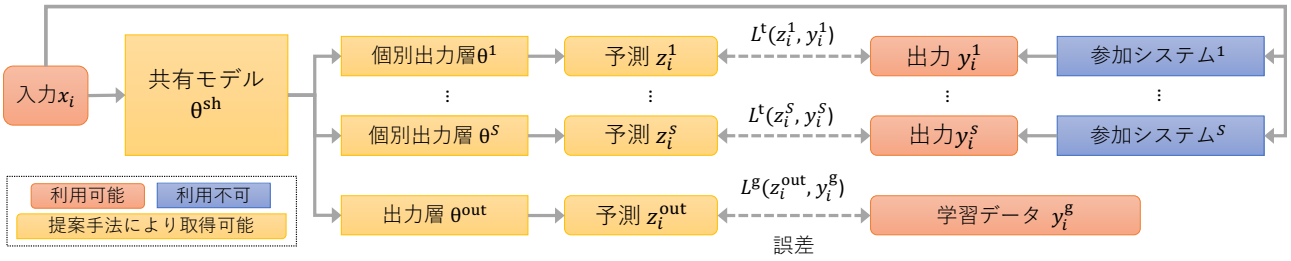

図 1　共同教育で用いる生徒モデル概要図

を生徒の学習に用いている。また、生徒の学習中に教師のパラメーターも更新するオンライン蒸留 [4] と、教師のパラメーターを固定するオフライン学習 [1, 2] に分けることもできる。本論文の設定は、教師自体にアクセスできないため、提案手法は応答ベースのオフライン蒸留手法である。

2.2　半教師あり学習

本手法は、ラベル無しデータに対する教師の予測を生徒の学習に用いる点で半教師あり学習 [5] や共学習 (Co-Training)[6] の一種である。半教師あり学習は、学習済みモデルの予測のうち確信度の高い予測を学習データに追加し、モデルを再学習することでより堅牢な機械学習モデルを構築する手法である。共学習は、半教師あり学習の応用であり、確信度を 2 つ以上のモデルにより算出する。我々の知る限り、共有タスクの結果を用いて学習データを拡張した研究はない。

3　提案手法: 共同教育

共有タスクに提出されたシステム結果から新規モデルを学習する共同教育を提案する。以下では、システムを教師、新規モデルを生徒と呼ぶ。共同教育を行うために必要なデータは、教師の予測結果と共有タスクで配布された学習データであり、これらが公開されていればタスク終了後であっても適用可能である。ただし、教師の予測データは多い方が良いため、共有タスク実施時に評価データ以外の範囲のラベル無しデータに対する予測提出が要求されることが好ましい。

提案手法で用いる生徒モデルの概要を図 1 に示す。生徒モデルは、共有モデル θ^{sh} と各教師に対応する個別出力層 θ^j ($j \in \{1, 2, \ldots, S\}$) と出力層 θ^{out} から成る。ここで S は教師システムの総数である。生徒モデルの入力を x_i とする。生徒モデルの学習は以下の誤差の最小化により行われる。

教師との誤差　教師システム j の出力結果 y_i^j に対応する生徒モデルの予測 z_i^j との誤差を $L^t(z_i^j, y_i^j)$ とし、全教師とのロスはその平均 $L^t(z_i, y_i) = \frac{1}{S}\sum_{j=1}^{S} L^t(z_i^j, y_i^j)$ とする。

学習データとの誤差　学習データ y_i^g と生徒モデルの予測 z_i^{out} との誤差を $L^g(z_i^{out}, y_i^g)$ とする。

入力 x_i に対する学習データのラベル y_i^g が存在しない場合は、教師との誤差 $L(z_i, y_i) = L^t(z_i, y_i)$ のみ最小化し、それ以外の場合は、両者の平均 $L(z_i, z_i^{out}, y_i, y_i^g) = \frac{1}{2}(L^t(z_i, y_i) + L^g(z_i^{out}, y_i^g))$ を最小化する。最終的な予測確率は、個別出力層の平均予測確率 $p_i^{mean} = \frac{1}{S}\sum_{j=1}^{S} \mathrm{softmax}(z_i^j)$ と出力層の予測確率 $p_i^{out} = \mathrm{softmax}(z_i^{out})$ の平均 $p_i = \frac{1}{2}(p_i^{mean} + p_i^{out})$ とする。

4　実験設定

4.1　森羅 2019

森羅は Wikipedia の構造化を目指すプロジェクトである。森羅 2019 は日本語 Wikipedia を対象としており、拡張固有表現 [7] にカテゴリー分けされた Wikipedia 記事 [8] から、同表現で定義された属性に対応する属性値の抽出を行う共有タスクである。例えば「人名」カテゴリーに分類された「木村拓哉」の記事から、属性「生年月日」に対する、属性値「1972 年 11 月 13 日」を抽出する。本タスクでは、属性値の表層文字でなく出現位置の抽出も行う必要がある。森羅 2019 では 33 カテゴリーを対象としており、JP-5（5 カテゴリー）、組織名（14 カテゴリー）、地名（14 カテゴリー）のサブタスクに分かれている。各カテゴリーには 269〜308,610 件の記事が割り当てられており、学習データとして 147〜1,000 件の記事に対してアノテーションラベルが付与されている。森羅タスクは Wikipedia の全記事の

構造化を目指しており、参加者は参加するカテゴリーに割り当てられた記事全てに対する予測結果を提出する必要がある。

森羅 2019 には合計 9 チームが参加し、各カテゴリーに対して 6 から 9 チームが参加している。参加者が用いた手法は、CRF や SVM などの機械学習、深層学習、DrQA などの機械読解など様々である。

森羅 2019 は、全てのシステム予測結果とタスクで共有された学習データに加え、アンサンブル研究用に「市区町村名」カテゴリーと「湖沼名」カテゴリーに対する開発データを配布している。我々はこれら開発データは学習には用いず、5.3 章の分析のみに使用する。

4.2 共同教育の適用

提案手法の有効性を示すため、4.1 節で述べた森羅 2019 の配布データを用いて生徒モデルを学習する。システム結果と学習データは、文章と文章に対する属性値の出現位置で構成されており、我々は系列ラベリングタスクとして扱うため、出現位置を IOB2 タグ [9] に変換する。本タスクでは、異なる属性において属性値が重複する場合があるため、各属性毎に IOB2 のタグを割り当てる。つまり、入力 x_i の j 番目の単語 $x_{i,j}$ に対して、ラベル $y_{i,j} \in \{I, O, B\}^c$ を割り当てる。ここで、c はカテゴリーに割り当てられた属性の数である。

共有モデル θ^{sh} には BERT-base [10] を用い、各個別出力層 θ^j、出力層 θ^{out} には 1 層の全結合層を用いる。BERT は RoBERTa [11] と同様の手順で日本語 Wikipedia を用いて学習する。不均衡なクラス分布に対応するため、損失関数は Class Balanced Loss [12] と Focal Loss [13] を用いる。

生徒モデルは各カテゴリー毎に学習し、計算コストの問題から使用する記事数を最大 2,000 件に制限する。この中には全てのラベル付きデータの範囲が含まれる。ラベル付きデータのうち 10%を開発データとして使用し、残りを学習に使用する。

生徒モデルの学習における最適化アルゴリズムに Adam を選択する。学習率は $\alpha_{lr} = 5 \times 10^{-5}$、Adam に関する残りのハイパーパラメーターは $\beta_1 = 0.9$、$\beta_2 = 0.999$, 、$\epsilon = 10^{-8}$ を使用する。また、計算効率を高めるため、混合精度計算モジュールである apex[2)] を使用する。バッチサイズは $\{8, 16, 32\}$ から、Class Balanced Loss で使用する γ は $\{0.999, 0.9999, 0.99999\}$ からグリッドサーチにより決定する。また、Class Balanced Loss で用いるクラス統計は学習データのみから取得する。

2） https://nvidia.github.io/apex/amp.html

5 実験結果

本章のスコアは指定がない場合全てマイクロ平均 F1 により算出される。これは森羅 2019 のスコア指標に準拠している。

5.1 サブクラス毎の結果

サブクラス毎の実験結果を表 1 に示す。サブクラスの全てのカテゴリーに参加したチームのみを記載している。共同教育は全体において、参加チームのうち最良スコアであるチーム 10 と比較し 2.38 の向上を獲得している。これは、様々な教師の強みを統合した結果であり、森羅 2019 はチーム 10 のシステムより優れた性能のシステムを公開できることを意味する。組織名サブクラスにおいてはチーム 10 に劣っているが差分は-0.36 と僅かである。

BERT は学習データのみで学習した場合の結果である。BERT は共同教育の結果と比較して大幅に劣っていることから、共同教育の向上が BERT の性能によるものでないことがわかる。また、自己教育は BERT の結果を教師として用いた場合の結果である。自己教育は共同教育の結果と比較して大幅に劣っていることから、共同教育の向上がラベル無しデータを用いたデータ拡張によるものではなく、教師システムからの学習によるものであることがわかる。

5.2 カテゴリー毎の結果

誌面の都合上、カテゴリー毎のスコアは付録表 2 に示す。各カテゴリー毎に最良のシステムを採用した場合のスコアのマクロ平均は 61.96 であるのに対し、共同教育のマクロ平均は 62.01 であり同等以上の性能が示された。参加チームのうちいずれかのカテゴリーで最良スコアを獲得したのは 5 チームである。もし、森羅 2019 で最良のシステム結果を公開する場合、以上の 5 チームにシステム提出を依頼する必要があるが、共同教育を用いた場合、同等の性能のシステムを公開できることとなる。これは提案手法の優位性を強く示す結果である。

カテゴリー毎に見た場合、共同教育は 18 カテゴリーにおいて参加チームの最良スコアを上回っているが、残りの 15 カテゴリーでは下回っている。図

サブタスク	参加チーム					BERT	自己教育	共同教育
	02	03	05	07	10			
JP-5	67.99	-	57.60	63.93	68.40	68.22	68.76	**69.95**
地名	59.51	57.89	49.56	53.68	58.21	57.74	58.43	**63.63**
組織名	51.33	53.68	37.52	48.70	**57.91**	51.14	52.14	57.55
全体	57.33	-	45.67	53.12	59.63	56.53	57.32	**62.01**

表 1　サブクラス毎の実験結果

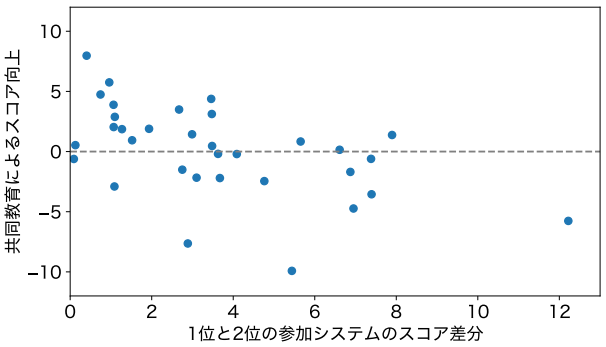

図 2　1 位と 2 位の参加システムのスコア差分と共同教育によるスコア向上の相関

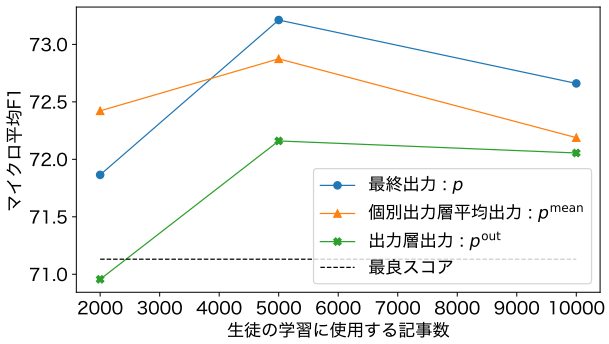

図 3　生徒の学習に使用する記事数とスコアの関係

2 に、1 位と 2 位の参加システムのスコア差分と共同教育によるスコア向上の相関を示す。両者の相関係数は $r = -0.519$ であり、負の相関が見られる。[3] つまり、最良の教師のみが優れている場合において、共同教育が効果的に機能していないことが分かる。現在の手法では、全ての教師との誤差を平均化しているため、多数決の結果は生徒モデルの勾配更新に大きな影響を及ぼす。そのため、単一の教師のみが優れているような場合では、生徒の学習に反映されないと考えられる。例えば MGDA [14] 等のような学習中に動的に誤差の重みづけする手法を用いることで、以上のような場合に対処できる可能性がある。

5.3　使用する記事数に関する分析

本実験では、生徒の学習に最大 2,000 件の記事を使用しているが、カテゴリーにおいてはより多くの記事を使用可能である。市区町村名カテゴリーにおいて生徒の学習に用いる記事数を $\{2,000, 5,000, 10,000\}$ と変化させて、森羅 2019 で配布されている開発データで評価した結果を図 3 に示す。[4] 最終出力 p を用いた場合 5,000 記事まではスコアの向上が見られるが、10,000 記事では低下している。使用する記事数が多くなると、教師と比較して学習データが生徒モデルに与える影響が小さくなるが、この場合は両者の誤差のバランスを調節する重みを導入する必要があると考えられる。また、出力層の出力 p^{out} より個別出力層の出力平均 p^{mean} が一貫して優れている。これは生徒が教師から得ている情報に非常に価値があることを示している。

6　おわりに

本論文では、共有タスクに提出されたシステム結果を用いて新規モデルを学習する手法である共同教育を提案した。本手法により、共有タスクは参加者にシステム自体の提供を要求せずとも、自然言語アプリケーションに汎用可能なシステムを公開することができる。本手法の有効性を示すため、実際の共有タスクである森羅 2019 により提供されたデータに対して共同教育を適用した。その結果、最良のシステムよりも高性能であり、最良のシステム群と同等性能のシステムを獲得できた。今後は本手法が多くの共有タスクで適用され、システム開発に注ぎ込まれた参加者の努力がより活用されることを願う。

3)　この結果 $p = 1.99 \times 10^{-3}$ は t 検定により $p < 0.01$ で統計的に有意である。

4)　この際使用するハイパーパラメーターは 4.2 章で使用したものと同じであるが、バッチサイズのみ 2,000 記事の場合 $\{8, 16, 32\}$、5,000 記事の場合 $\{20, 40, 80\}$、10,000 記事の場合 $\{40, 80, 160\}$ からグリッドサーチにより選択する。

謝辞

本研究は JST、ACT-X、JPMJAX20AI および JSPS 科研費 JP20269633、JST さきがけ JPMJPR20C2 の支援を受けたものです。

参考文献

[1] Jimmy Ba and Rich Caruana. Do deep nets really need to be deep? In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, editors, **Advances in Neural Information Processing Systems**, Vol. 27. Curran Associates, Inc., 2014.

[2] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. In **NIPS Deep Learning and Representation Learning Workshop**, 2015.

[3] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Y. Bengio. Fitnets: Hints for thin deep nets. **arXiv**, 12 2014.

[4] Ying Zhang, Tao Xiang, Timothy M. Hospedales, and Huchuan Lu. Deep mutual learning. In **2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition**, pp. 4320–4328, 2018.

[5] David Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In **33rd Annual Meeting of the Association for Computational Linguistics**, pp. 189–196, Cambridge, Massachusetts, USA, June 1995. Association for Computational Linguistics.

[6] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In **Proceedings of the Eleventh Annual Conference on Computational Learning Theory**, COLT' 98, pp. 92–100, New York, NY, USA, 1998. Association for Computing Machinery.

[7] Sekine Satoshi. Extended named entity ontology with attribute information. In **Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)**, Marrakech, Morocco, May 2008. European Language Resources Association (ELRA).

[8] Masatoshi Suzuki, Koji Matsuda, Satoshi Sekine, Naoaki Okazaki, and Kentaro Inui. A joint neural model for fine-grained named entity classification of wikipedia articles. **IEICE Transactions on Information and Systems**, Vol. E101.D, No. 1, pp. 73–81, 2018.

[9] Erik F. Tjong Kim Sang and Jorn Veenstra. Representing text chunks. In **Ninth Conference of the European Chapter of the Association for Computational Linguistics**, pp. 173–179, Bergen, Norway, June 1999. Association for Computational Linguistics.

[10] 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.

[11] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A robustly optimized bert pretraining approach, 2019.

[12] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples, 2019.

[13] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal loss for dense object detection. In **Proceedings of the IEEE International Conference on Computer Vision (ICCV)**, Oct 2017.

[14] Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, **Advances in Neural Information Processing Systems**, Vol. 31. Curran Associates, Inc., 2018.

サブタスク	カテゴリー	参加チーム									最良	共同
		01	02	03	04	05	06	07	08	10	スコア	教育
JP-5	空港名	44.15	89.55	79.92		86.03		84.74		88.03	89.55	**90.49**
	市区町村名	7.96	66.40	60.93		56.19		62.37		66.49	**66.49**	65.88
	企業名	11.92	61.95	63.25		39.59	10.48	53.82		66.13	**66.13**	58.49
	化合物名		45.67	47.98		50.32		49.35	50.72	49.04	50.72	**58.69**
	人名	3.42	76.40		34.53	55.88		69.39		72.31	**76.40**	76.19
地名	湾名	0.16	67.47	60.86		52.58		58.55		58.73	67.47	**67.62**
	大陸地域名		56.38	53.71		41.48		51.03		53.28	56.38	**59.87**
	国名		57.66	61.27		48.87		52.00		64.26	64.26	**65.70**
	国内地域名		48.55	43.09		23.91		44.71		50.49	50.49	**52.38**
	地形名__その他	2.91	57.29	58.98		43.83		46.77		62.65	**62.65**	60.45
	ＧＰＥ__その他	0.77	56.45	48.71		38.03		46.38		49.07	**56.45**	55.85
	島名		67.40	66.31		53.87		58.90		59.77	67.40	**70.29**
	湖沼名	9.67	63.09	59.63		55.42		57.01		43.43	63.09	**67.47**
	地名__その他	2.40	40.70	40.82		38.35		41.10		49.00	49.00	**50.37**
	山地名	4.18	62.73	59.24		56.62		56.27		61.46	62.73	**64.59**
	都道府県州郡名	2.10	66.18	60.45		60.42		59.39		67.25	67.25	**71.14**
	河川名	3.52	59.49	61.30		48.81		56.24		64.92	**64.92**	64.73
	海洋名		60.96	62.90		57.35		55.42		65.65	**65.65**	64.15
	温泉名	10.82	68.83	73.18		74.24		67.75		65.00	74.24	**76.28**
企業名	企業グループ名	0.57	56.42	65.03		29.72		54.32		61.55	65.03	**65.49**
	民族名__その他		51.05	50.56		39.96		47.99		56.71	56.71	**57.54**
	家系名	0.18	62.78	60.40		39.09		61.86		69.66	**69.66**	67.97
	政府組織名	2.75	50.24	51.20		43.07		47.63		47.09	51.20	**56.95**
	国際組織名	2.31	48.58	52.71		39.62		44.08		51.97	52.71	**57.45**
	軍隊名	1.94	53.14	60.12		39.55		52.67		67.52	**67.52**	63.97
	非営利団体名	3.23	46.96	47.53		39.88		40.20		59.75	**59.75**	53.99
	組織名__その他	4.06	50.46	53.95		42.22		42.74		52.87	**53.95**	51.05
	政治的組織名__その他		40.60	34.70		21.35		26.64		47.55	**47.55**	42.82
	政党名	1.35	46.65	47.48		39.78		41.49		52.24	**52.24**	49.79
	公演組織名	1.17	63.96	71.43		36.32		64.80		68.33	**71.43**	69.26
	競技連盟名	4.52	51.39	56.94		46.21		50.90		56.81	56.94	**57.47**
	競技リーグ名	2.03	45.65	47.97		24.88		58.42		63.86	**63.86**	53.95
	競技団体名	3.91	50.68	51.44		43.69		48.09		54.92	54.92	**58.04**
マクロ平均		-	57.33	-	-	45.67	-	53.12	-	59.63	61.96	**62.01**

表 2　カテゴリー毎の結果