言語横断的情報検索の大規模データセットとパラメータ共有モデル

佐々木 翔大 [1]　Shuo Sun[2]　Shigehiko Schamoni[3]　Kevin Duh[2]　乾 健太郎 [1,4]
[1] 東北大学　[2]Johns Hopkins University　[3]Heidelberg University　[4] 理研 AIP

{sasaki.shota,inui}@ecei.tohoku.ac.jp　ssun32@jhu.edu
schamoni@cl.uni-heidelberg.de　kevinduh@cs.jhu.edu

1　はじめに

言語横断的情報検索（Cross-lingual Information Retrieval，以下 CLIR）とは文書を記述する言語とユーザの検索クエリを記述する言語が異なる状況下で，クエリに関連する文書を検索するタスクである．昨今，Web 上には多様な言語で記述された文書が多く存在しているため，CLIR は重要なアプリケーションである．一例として，Twitter 上の多様な言語の会話から国際ブランド製品の消費者の感情極性を観察したい投資家（英語話者）がいるとする．投資家は検索クエリを英語で記述して，言語の違いにかかわらず，関連する全ての文書を検索したいと考えるだろう．

CLIR システムの構築に関しては，主に 2 つアプローチがある．1 つ目は，翻訳と単言語情報検索の 2 つの構成要素から成る**モジュール型アプローチ**である [9]．これは検索クエリと文書の言語が異なるという問題を事前に翻訳によって解決することで，CLIR を単言語情報検索の問題として捉えようとするアプローチである．

モジュール型アプローチと明確に異なるもう 1 つのアプローチが**直接モデリングアプローチ**である [1, 14]．直接モデリングアプローチは，英語で記述された検索クエリを q，英語以外で記述された文書を d，q に対して d がどれほど関連しているかを表すスコアを r としたとき，訓練データとして (q, d, r) の 3 つ組を用いて，未知の q, d に対してその関連度のスコア $S(q, d)$ を予測するモデルを学習する．モデルは，異なる言語の q と d からなる訓練データから，翻訳と情報検索のためのスコアリングの両方を直接学習すると言える．モジュール型アプローチと比べて，直接モデリングアプローチは以下の 2 点のメリットがある：(1) 翻訳元の文での意味や構造を翻訳後においても保持しようとする，一般的な機械翻訳ではなく，より検索に対して有効な翻訳を学習することができる．(2) CLIR を end-to-end で行うことができる．

しかしながら現状では，多様な言語における直接モデリングのための大規模なデータセットは存在しない．検索クエリと文書の関連度スコアを得ようとしたとき，典型的な手法の 1 つとして，検索クエリと文書のどちらの言語も読むことができるバイリンガル話者にその関連度スコアを付与してもらう方法があるが，これは非常に大きなコストがかかる．

そこで我々は CLIR システムの訓練及び評価のための大規模データセットを Wikipedia から自動構築する．このデータは英語クエリと英語以外の 25 言語で記述された文書で構成されており，直接モデリングアプローチにおける訓練に必要なデータ量を十分満たすばかりでなく，モジュール型アプローチの評価データとしても用いることができる[*1]．

さらにこのデータの有用性を示すために我々は，多くのデータが使用できる言語（高資源言語）ペアのデータセットを用いて，使用できるデータの少ない言語（低資源言語）ペアのデータセット上での検索性能を改善するパラメータ共有手法を提案する．これによって，例えば日本語–英語の訓練データを利用して，スワヒリ語–英語の検索システムの Mean Average Precision の結果を 4-7 ポイント改善したことを報告する．

2　CLIR のための大規模データセット作成

我々は CLIR のための大規模なデータセットを Wikipedia から作成した[*2]．主な作成手順は，ある英語記事から 1 文をクエリとして抽出し，さらにクエリに関連がある他言語の文書を **inter-language リンク**を利用することで取得する（図 1）．このデータ作成手順は英語–ドイツ語のデータセットを作成した Schamoni ら [11] に則しているが，我々はより大規模なデータの作成を行った．

より具体的には，はじめに英語 Wikipedia の記事（図 1 の E1）の 1 文目を抽出したものをクエリとした．これは，「記事の 1 文目は一般的にその記事の要約になっており，さらに inter-language リンクされた記事（図 1 の F1）に対しても，主題が一貫している」という仮

[*1] データセットは http://www.cl.ecei.tohoku.ac.jp/~sasaki.shota/wikiclir にて公開予定である．
[*2] 2017 年 8 月の Wikipedia ダンプファイルを用いた．

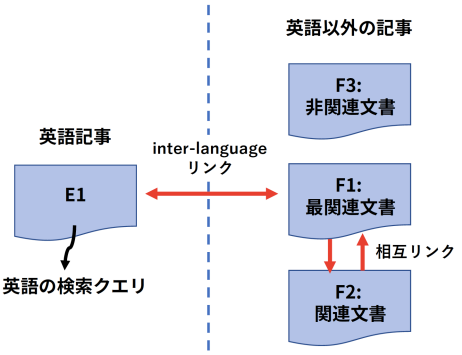

図 1　CLIR のためのデータ作成過程．データは (英語クエリ q, 英語以外の言語の文書 d, 関連度スコア $r \in \{0, 1, 2\}$) の 3 つ組で構成される．

定に基づくものである．Schamoni ら [11] と同様に，英語記事のタイトルにある単語はクエリから削除した．これは，記事のタイトルにある単語は inter-language リンク先の記事 (F1) にも存在していることが多く，タスクが簡単なキーワードマッチングのタスクになってしまうことを防ぐために行った．

次にクエリを抽出した記事 (E1) から inter-language リンクされている記事を**最関連文書**（図 1 の F1）とした．さらに F1 と相互リンクされている全ての記事を**関連文書**（図 1 の F2）とした．最後にその他全ての記事を**非関連文書**（図 1 の F3）とした．それぞれの文書は記事のはじめから 200 単語までを用い，空の記事やカテゴリ記事は除外した．最終的に，我々のデータセットは 280 万の英語クエリと英語以外の 25 の言語の文書からなる（表 1）．つまり，我々は言語の種類数及び検索クエリと関連文書の数という点で非常に大規模なデータセットを作成したといえる．

また，我々のデータセットは以下の 2 つのシナリオにおいて利用することができる．(1) 全ての言語のデータを 1 つのデータセットとして用いて，英語のクエリから多様な言語の文書を検索する．(2) 25 の独立したデータデータセットとして，英語クエリから英語以外の 1 つの言語の文書を検索する．実験の章（第 4 章）では (2) のシナリオで実験を行う[*3]．

3　CLIR のための直接モデリング

3.1　ニューラルランキングモデル

これまで，CLIR のモデルに拡張可能な多くのランキングモデルが提案されている [4, 13, 16, 8] が，これらは全て，クエリと文書から特徴を抽出し，ランキングロスを通して $S(q, d)$ を最適化するという共通の枠組みを持っている．我々はこれに則し，英語のクエリ q と英語以外の言語で記述された文書 d を与えられたとき，その関連度スコア $S(q, d)$ を計算するモデルを定義する．まずはじめに，各単語を n 次元のベクトルで表現することで，q, d をそれぞれ行列 $\mathbf{Q} \in \mathbb{R}^{n \times |q|}$, $\mathbf{D} \in \mathbb{R}^{n \times |d|}$ として表す：

$$\mathbf{Q} = [E_q(q_1); E_q(q_2); ...; E_q(q_{|q|})]$$
$$\mathbf{D} = [E_d(d_1); E_d(d_2); ...; E_d(d_{|d|})]$$

ここで $|q|$, $|d|$ はそれぞれ，q, d に含まれる単語の総数，q_i, d_i はそれぞれ q, d 内の i 番目の単語を表す．E は各単語を n 次元のベクトルに変換する埋め込み関数であり，; は連結演算子である．次にこれらの行列に畳み込み特徴マップ[*4]，続いて活性化関数としての tanh と平均プーリングを適用することで，それぞれの表現ベクトル $\hat{q}$, $\hat{d}$ を得る：

$$\hat{q} = CNN_q(\mathbf{Q}); \quad \hat{d} = CNN_d(\mathbf{D}) \qquad (1)$$

言語	文書数	クエリ数	SR 数
Arabic	535	324	194
Catalan	548	339	625
Chinese	951	463	462
Czech	386	233	720
Dutch	1908	687	1646
Finnish	418	273	665
French	1894	1089	4048
German	2091	938	4612
Italian	1347	808	2635
Japanese	1071	426	2912
Korean	394	224	343
Norwegian-Nynorsk	133	99	150
Norwegian-Bokmål	471	299	663
Polish	1234	693	1777
Portuguese	973	611	1130
Romanian	376	199	251
Russian	1413	664	1656
Simple English	127	114	135
Spanish	1302	781	2113
Swahili	37	22	35
Swedish	3785	639	1430
Turkish	295	185	195
Ukrainian	704	348	565
Vietnamese	1392	354	257
Tagalog	79	48	23

（全て単位は千）

表 1　CLIR データセットの統計情報．それぞれの言語 X に対して，言語 X で記述された文書の数と英語クエリの数を示した．最関連文書は定義よりクエリ数と同数である．関連文書数は SR 数の列に示した．

[*3] 将来的な研究の拡張の目的で，今回の実験は全体のデータセットからランダムにサンプルした半量のデータを用いた．

[*4] 畳み込み窓のサイズは $n \times 4$ でフィルターサイズは 100，ストライドサイズは 1 とした．

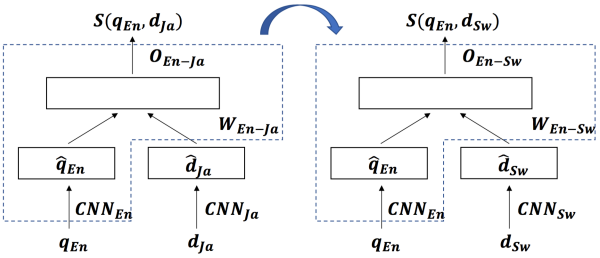

図 2　提案手法の概要図．低資源言語のデータセット（例 スワヒリ語–英語）における実験で，クエリのエンコードのための CNN のパラメータ (CNN_{En}) と全結合層のパラメータ (O_{En-Sw}, W_{En-Sw}) を高資源言語のデータセット（例 日本語–英語）で学習済みのパラメータで初期化する．

ここで $S(q,d)$ を計算する 2 種類のモデルを定義する．1 つ目は $S(q,d)$ として $\hat{q}$ と $\hat{d}$ の cosine 類似度を測る **cosine モデル**である：

$$S_{cos}(q,d) = cossim(\hat{q}, \hat{d}) \tag{2}$$

2 つ目のモデルは $\hat{q}$ と $\hat{d}$ を連結した 200 次元のベクトルを全結合層に入力する **deep モデル**である：

$$\begin{aligned} S_{deep}(q,d) &= tanh(O \cdot h_{vec}^{\mathrm{T}}) \\ &= tanh(O \cdot relu(W \cdot [\hat{q}; \hat{d}]^{\mathrm{T}})) \end{aligned} \tag{3}$$

ここで $O \in \mathbb{R}^{1\times h}$, $W \in \mathbb{R}^{h\times 200}$ は重みパラメータで h は隠れ層 $h_{vec} \in \mathbb{R}^{1\times h}$ の次元数である．また隠れ層に dropout[15] を適用する（dropout 率は 0.5）．

訓練時には，ランキング学習に広く用いられている pairwise ランキングロス [10, 3, 5, 16, 2] を最小化する．

$$L = max\left\{0, 1 - (S(q,d^+) - S(q,d^-))\right\} \tag{4}$$

ここで d^+ と d^- はそれぞれ最関連文書（図 1 の F1），非関連文書（図 1 の F3）である[*5]．最適化時には単語ベクトルは固定し，その他のパラメータはチューニングする．

3.2　パラメータ共有手法

deep モデルのような大きなネットワークを訓練するには，一般に非常に多くのデータを必要とする．このデータ量の問題に対処するために，我々は異なる言語ペアで学習した CLIR モデルのパラメータを共有するという，簡易的かつ効果的な手法を提案する．基本的にモデルの構造は deep モデル ($S_{deep}(q,d)$, 式 3) と同じものを用いる．但し，低資源な言語ペア（例えばスワヒリ語–英語）の実験の際に，高資源な言語ペア（例えば日本語–英語）のデータセットで学習されたパラメータを用いてパラメータを初期化する．

[*5] 実験の簡易化のために，関連文書は用いずに，最関連文書と非関連文書のみを用いた．

	Ja	De	Fr
$S_{cos}(q,d)$: cos	59/74	49/66	55/70
$S_{deep}(q,d)$: h=100	61/75	64/77	69/81
$S_{deep}(q,d)$: h=200	68/80	67/79	74/84
$S_{deep}(q,d)$: h=300	70/82	70/81	74/84
$S_{deep}(q,d)$: h=400	**73**/83	**71**/**82**	75/**85**
$S_{deep}(q,d)$: h=500	**73**/**84**	70/81	**76**/**85**

表 2　高資源な言語のデータセットにおける，cosine モデルと deep モデルの P@1/MAP（%）の性能比較．各列で最も良い性能の値を太字で示した．

手法の概要を図 2 に示した．具体的には，学習済みのパラメータを用いて，クエリのエンコードに用いる CNN のパラメータ (CNN_q) と 全結合層のパラメータ (O, W) を初期化する．パラメータの最適化方法は通常のモデルと同様で，単語ベクトルのみ固定して，その他のパラメータはチューニングする．

このパラメータ共有手法は，「直接モデリングアプローチにおいて $\hat{q}$ と $\hat{d}$ はクエリと文書の言語非依存な表現となっている」という仮定に基づくものである．ゆえに全結合層のパラメータ O と W に関しても，どの言語ペアのデータセットでも用いることができると考えられる．cosine モデルにおいては CNN_q のみ共有する．

4　実験

4.1　実験設定

我々は 3 つの高資源言語（日本語［Ja］，ドイツ語［De］，フランス語［Fr］）と 2 つの低資源言語（タガログ語［Tl］，スワヒリ語［Sw］）のデータセットを使用した．ただし，高資源言語と低資源言語はデータの量と言語の性質という 2 点で異なっている．言語の性質の違いによる影響を排除して，訓練データの量の影響のみを明らかにするために，ドイツ語とフランス語のデータをスワヒリ語のデータサイズと同等になるようにサブサンプリングしたデータセットも利用した．単語ベクトルのサイズは 100 次元で word2vec SGNS [7] を用いて Wikipedia コーパス上で訓練した．deep モデルの隠れ層の次元数は {100, 200, 300, 400, 500} を用いた．最適化アルゴリズムは Adam [6] を用い，20 epoch 訓練した中で，開発セットに基づいて最も良い epoch を選び，評価に用いた．パラメータ共有手法においては，日本語–英語のデータセットで訓練したパラメータを初期化に用いた．

4.2　結果と分析

高資源言語: 10 万クエリ以上の訓練クエリからなるデータセットにおける P@1（ランキング最上位における precision）と MAP（mean average precision）の結

	Tl			Sw			De（サブサンプル）			Fr（サブサンプル）		
	In	Sh	Δ	In	Sh	Δ	In	Sh	Δ	In	Sh	Δ
cos	51/68	50/67	−/−	51/67	49/65	−/−	40/59	38/56	−/−	46/63	43/60	−/−
h=100	34/50	48/62	+/+	46/62	46/62	=/=	39/55	46/62	+/+	40/57	46/62	+/+
h=200	44/58	55/67	+/+	47/63	52/67	+/+	41/57	48/63	+/+	43/60	51/66	+/+
h=300	42/57	49/63	+/+	50/65	58/70	+/+	44/60	50/65	+/+	49/65	51/66	+/+
h=400	49/63	**57/69**	+/+	51/66	**60/73**	+/+	45/61	**51/66**	+/+	47/64	**56/70**	+/+
h=500	51/64	54/67	+/+	53/68	56/69	+/+	44/60	49/65	+/+	47/63	51/66	+/+

表 3　低資源言語のデータセットにおける P@1/MAP（%）の性能比較．Δ の列は 単一の言語ペアのデータセット上で訓練したモデル（In）とパラメータ共有手法を用いたモデル（Sh）の比較である．+ は Sh の性能が In の性能を上回ることを示し，–はその逆を示している．各データセットで最も良い性能の値を太字で示した．

果を表 2 に示す．全ての条件において deep モデルの性能は cosine モデルの性能を上回った．これは deep モデルの全結合層が，大量の訓練データからより表現力の高いスコアリング関数を学習したことを示している．

低資源言語: 低資源な言語のデータセット上での 2 つの設定における結果を表 3 に示す．1 つ目の設定は，パラメータ共有手法を用いずに，単一の言語ペアの訓練データのみを利用して学習する設定（In）である．2 つ目の設定は，日本語–英語のデータセットで事前に学習されたパラメータを利用するパラメータ共有手法を用いて学習する設定（Sh）である．パラメータ共有手法を用いない設定では，cosine モデルの性能が deep モデルの性能を上回った．この結果は，高資源言語のデータセット上での結果とは対照的であり，deep モデルのような多くのパラメータを持つモデルは十分な訓練データ量がある場合でないと有効でないことを示唆している．

また，ほとんど全ての条件においてパラメータ共有手法を用いた deep モデルの性能がパラメータ共有手法を用いない deep モデルの性能を上回った．このことから，我々のパラメータ共有手法を用いることで，deep モデルを訓練するために必要な訓練データの量を抑えることができていることがわかった．さらに，パラメータ共有手法を用いることによって deep モデルの性能は cosine モデルの性能を上回り，全てのデータセットにおいて最も良い性能を達成した[*6]．

5　おわりに

CLIR のための直接モデリングの訓練及び評価に用いることができる 25 の言語からなる大規模データセットを作成した．さらにパラメータ共有手法を提案し，低資源な言語のデータセットにおける有用性を示した．

将来的には，(a) より多くの言語のデータに拡張する (b) 他のランキングモデルで実験することが考えられる．また我々のモデルを TREC [12] のようなより標準的な CLIR の評価セットにおいて評価する予定である．このことによって，自動的に生成されたデータセットから学習された知識が，一般的な CLIR の問題に適用することができるかを明らかにしたいと考えている．

[*6] cosine モデルにおいては，パラメータ共有手法を用いても性能の改善が見られなかった．このことからパラメータ共有手法は，モデルが十分な表現力を有するときに限って有効であると考えられる．

謝辞

本研究は JST CREST（課題番号: JPMJCR1513），JSPS 科研費 15H0170，短期共同研究留学プログラム COLABS の支援を受けて行った．

参考文献

[1] Bing Bai, et al. Learning to rank with (a lot of) word features. *Information Retrieval*, 2010.
[2] Mostafa Dehghani, et al. Neural ranking models with weak supervision. In *Proceedings of SIGIR*, 2017.
[3] Jiafeng Guo, et al. A deep relevance matching model for ad-hoc retrieval. In *Proceedings of CIKM*, 2016.
[4] Po-Sen Huang, et al. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of CIKM*, 2013.
[5] Kai Hui, et al. PACRR: A position-aware neural ir model for relevance matching. In *Proceedings of EMNLP*, 2017.
[6] Diederik Kingma, et al. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
[7] Tomas Mikolov, et al. Distributed representations of words and phrases and their compositionality. In *Proceedings of NIPS*, 2013.
[8] Bhaskar Mitra, et al. Learning to match using local and distributed representations of text for web search. In *Proceedings of WWW*, 2017.
[9] Jian-Yun Nie. *Cross-Language Information Retrieval*. Morgan & Claypool Publishers, 2010.
[10] Liang Pang, et al. A study of match pyramid models on ad-hoc retrieval. Neu-IR '16 SIGIR Workshop, 2016.
[11] Shigehiko Schamoni, et al. Learning translational and knowledge-based similarities from relevance rankings for cross-language retrieval. In *Proceedings of ACL*, 2014.
[12] P. Schäuble, et al. Cross-language information retrieval (CLIR) track overview. In *Proceedings of TREC Conference*, 1997.
[13] Yelong Shen, et al. A latent semantic model with convolutional-pooling structure for information retrieval. In *Proceedings of CIKM*, 2014.
[14] Artem Sokolov, et al. Boosting cross-language retrieval by learning bilingual phrase associations from relevance rankings. In *Proceedings of EMNLP*, 2013.
[15] Nitish Srivastava, et al. Dropout: A simple way to prevent neural networks from overfitting. *Journal of machine learning research*, 2014.
[16] Chenyan Xiong, et al. End-to-end neural ad-hoc ranking with kernel pooling. In *Proceedings of SIGIR*, 2017.