Bi-encoder と kNN の組み合わせによる 職務記述書に書かれた文のスキルマッピング

牧野拓哉
株式会社リクルート Megagon Labs, Tokyo, Japan
makino@megagon.ai

概要

スキルマッピングは職務記述書や履歴書中の文で言及されるオントロジーで定義されたスキルを特定する作業であり労働市場の分析に不可欠である．詳細な分析に必要な細分化されたスキルが付与された学習データを人手で構築するのは高いコストなため既存研究は Large Language Model (LLM) が生成した合成データが bi-encoder の学習に用いる．合成データを活用したさらなる精度改善のため，提案手法 (kNNBE) は推論時に学習に利用したラベル付き合成文を k-nearest neighbor (kNN) によって取得し入力文との類似度を bi-encoder のスコアに加える．実験により kNNBE は bi-encoder の精度改善，さらに既存の最高精度を示す LLM でスキルをリランキングする手法と比較して高いスループットを維持しながら精度改善を確認した．

1 はじめに

労働市場におけるスキルの需要と供給の分析は企業の戦略的な採用や個人のキャリア向上を支援するために重要である [1]．このような分析にはオントロジーで定義されたスキルを履歴書や職務記述書に自動的に付与するスキルマッピングが不可欠である．詳細な分析には ESCO [2] のような細分化されたスキルオントロジーを活用する必要があるが，スキル数の多さ（例えば ESCO は 13,000 以上）のため学習データを手作業で構築することは高コストである．高いマッピング精度を得るために LLM でスキル候補リストをリランキングする方法 [3, 4, 5] が提案されているが，日々増え続ける職務記述書や履歴書を効率的に処理する用途には適さない[1)]．Decorte ら [4] は GPT-3.5 によって生成された 138,000 件以上の合成文とスキルのペアを使用して bi-encoder を学習した．bi-encoder は BERT に基づく文埋め込みモデル [7] によって個別に得られた文埋め込みとスキル埋め込みの類似度を計算する．bi-encoder は高いスループットを実現できる一方で，LLM でリランキングする手法 [3, 5] と比較して精度は低い．精度の低い原因の一つとして細分化されたスキルセットの区別が難しいことが挙げられる．例えば “Develop interfaces to ERP and POS software providers and work on ETL pipelines to integrate our customers data bases.” に対して bi-encoder はスキル “transportation software related to an ERP system” を上位にランクする．このスキルは輸送に関する ERP に関するスキルであり文意とは一致しない．

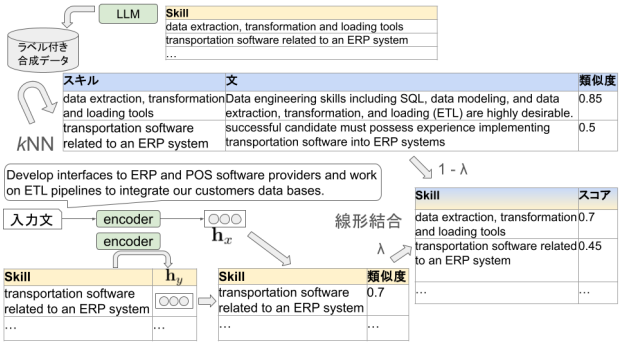


図 1: 提案手法 (kNNBE) 概要．ラベル付き合成データを kNN で取得し，bi-encoder のスコアと足し合わせる．

本稿は bi-encoder の精度をさらに向上させるために kNNBE を提案する．kNNBE は推論時に学習事例を取得しスコアを調整する（図 1）．LLM で合成データ生成することで bi-encoder の目的タスクにおける精度改善は報告されているが [8]，kNN の検索事例に合成データを利用する場合の有効性は明らかではない．本稿ではスキルマッピングの 3 つのベンチマークデータでの実験により kNNBE は合成データを kNN の検索事例として利用することで bi-encoder と比較して精度が改善することを検証し

1) 例えば米国労働統計局は 1 ヶ月の求人件数が 1,200 万件に達することがあると報告している [6].

た．またこれまでの最高精度である LLM でスキル候補をリランキングする既存手法 [5] よりもスループット 50 倍以上を維持しながら，2 から 3 ポイント高い精度となることを確認した．

2 提案手法

2.1 アーキテクチャ

本稿は LLM2Vec [9] を埋め込みの計算に用いる．この手法は decoder として学習した LLM を encoder に変換する．入力文 x の先頭に指示文「`Given a job ad sentence, retrieve skills mentioned in that sentence:`」を加え，候補スキル y を与えたとき，encoder$E$ は文埋め込み $\mathbf{h}_x$ とスキル埋め込み $\mathbf{h}_y$ を独立に計算する．文およびスキルの埋め込みは平均プーリングとした．ただし文は指示文のトークンを除外した上で計算した．bi-encoder は $\mathbf{h}_x$ と $\mathbf{h}_y$ のコサイン類似度を計算し，これを 文 x に対するスキル y のスコアとして用いる：

$$s(\mathbf{h}_x, \mathbf{h}_y) = \frac{\mathbf{h}_x \mathbf{h}_y}{|\mathbf{h}_x||\mathbf{h}_y|}. \tag{1}$$

2.2 スコア計算

スコア計算は言語モデルと kNN の組み合わせである kNN-LM [10] を参考に bi-encoder によるスコアと kNN によるスコアの線形結合を用いる．

合成データで bi-encoder を訓練した後，ラベル付き事例の集合 $\mathcal{D}$ からメモリ $\mathcal{M} = \{(\mathbf{g}_{x_i}, y_i) | (x_i, y_i) \in \mathcal{D}\}$ を事前に構築する．$\mathbf{g}$ は encoder の最終層の self-attention の出力に layer normalization を適用したベクトルである [10]．入力 x に対して，メモリ $\mathcal{M}$ から取得された k 個の最近傍の集合を $\mathcal{N}$ とする．x に対するラベル y のスコアは以下のように定義される：

$$\begin{aligned}\bar{s}(\mathbf{h}_x, \mathbf{h}_y) = (1-\lambda) \quad & s(\mathbf{h}_x, \mathbf{h}_y) \\ + \lambda & \sum_{(\mathbf{h}_{x_k}, y_k) \in \mathcal{N}} \mathbb{1}_{y_k = y} \frac{s(\mathbf{g}_x, \mathbf{g}_{x_k})}{k},\end{aligned} \tag{2}$$

出力は上位 K 個のスコアが最も高いスキルである．λ は bi-encoder と kNN の重みを制御するハイパーパラメータである．初項は文とスキルの埋め込みの類似度，第二項は文とラベル付き事例の文の埋め込みの類似度を表す．

3 実験

3.1 実験データ

実験はスキルマッピングデータセットである HOUSE，TECH，および TECHWOLF を使用する [11, 4]．開発データの文数はそれぞれ 61，75，0 であり，テストデータは 338，262，326 である．TECH と TECHWOLF はソフトウェアエンジニアが主な職務を対象としており，HOUSE はより一般的な職務を含む．多くの文は 1 つのスキルが付与される一方で複数のスキルが付与される文も存在する．[2] これらのデータセットを構築するために文はあらかじめ作成されたショートリストからスキルを選択するか，ESCO を通じて関連するスキルを検索してラベル付けされた．詳細は Decorte ら [11] の付録 A を参照されたい．

3.2 比較手法

IReRa [5] は bi-encoder が選択した候補スキルリストを GPT-4 で並び替える手法であり，ベンチマークデータで最高精度を示す．精度改善のため Llama-2-7b-chat-hf[3] を用いて文を書き換え，MPNet [12] に基づく bi-encoder への入力とする．またクエリ抽出とスキルマッピングのプロンプトへ文脈内学習のための事例を追加する．これらの事例は開発データを用いた交差検定で RP@10 を最大化するように選択する．IReRa$_{M-7b}$ は文の書き換えおよび候補スキルリストの並び替えに Mistral-7B-Instruct-v0.2[4] を用いた手法である．いずれも少数事例の選択には各データセットでおよそ 0.8〜1 時間かかった．

BE は bi-encoder を表す．埋め込みの計算には Mistral-7b-Instruct-v0.2 に対して LLM2Vec [9] を適用して作成された encoder[5] を使用する．

kNN は式 2 の第 2 項のみを用いて計算する．kNN で使用する encoder は BE と同じものである．

kNNBE（提案手法）は，合成データで学習した BE と同じ bi-encoder を利用する．推論時に kNN によって取得されたラベル付き文を 2.2 節に記載された方法で組み込む．

2） スキル数の分布は参考情報の図 2 を参照されたい
3） https://huggingface.co/meta-llama/Llama-2-7b-chat-hf
4） https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.2
5） https://huggingface.co/McGill-NLP/LLM2Vec-Mistral-7B-Instruct-v2-mntp-supervised

表 1: RP@K およびスループット (事例数/秒)．括弧中の数字は IReRa に対するスループット比を表す．下線は 1，2 行目のグループ，3，4，5 行目のグループ中でそれぞれの最大値を示す．**太字** は列で最大の値を示す．IreRa の RP@K は論文 [5] の値を引用した．† は推論時に開発データの事例を利用していることを示す．

	HOUSE		TECH		TECHWOLF		Average		
	RP@5	RP@10	RP@5	RP@10	RP@5	RP@10	RP@5	RP@10	スループット
IReRa†	**56.60**	65.76	59.61	70.23	57.04	65.17	57.75	67.05	2.49 (×1.00)
IReRa†$_{M-7b}$	40.38	53.85	48.11	60.01	37.62	44.97	42.04	52.94	2.93 (×1.18)
BE	53.60	65.07	65.00	74.28	56.98	65.23	58.53	68.20	**145.56** (×58.5)
kNN	48.73	59.92	57.71	68.02	54.84	62.78	53.76	63.57	129.84 (×52.1)
kNNBE (ours)	55.79	**66.03**	**65.23**	**75.63**	**59.43**	**68.80**	**60.16**	**70.15**	130.11 (×52.3)

3.3 評価尺度

先行研究 [4, 5] に従いスキルマッピングの精度指標として Rank Precision@K (RP@K) を使用する：

$$\mathrm{RP@K} = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{\min(K, R_n)}\sum_{k=1}^{K}\mathrm{Rel}(n,k). \tag{3}$$

kNNBE を既存研究と比較するため，RP@5 および RP@10 を使用する．テストデータにおけるスキル数の最大値は 10 である（図 2）．したがって RP@10 は再現率に相当し，モデルが取得した上位 10 件のスキルに含まれる適切なスキル数を文中の全適切スキル数で割った値となる．多くの文は 5 つ以下のスキルを持つため RP@5 はおおむね適合率に相当し，上位 5 件のスキルに含まれる適切なスキル数を 5 で割った値となる．また 1 秒あたりに処理できる事例数をスループットとして比較する．

3.4 学習・推論の設定

学習および推論には単一の A100-80GB GPU を使用した．学習には既存研究 [4] の合成データ[6)]を利用した．このデータは与えられたスキルに対して GPT-3.5 を用いて 10 の合成文を生成することで構築されている．学習時には存在しないスキルについても精度を比較するため学習に用いる合成データから開発およびテストデータに出現するスキルをランダムに 5 割削除した．その結果，学習に用いる事例数は 134,410 件となった．損失関数は InfoNCE [13] とした．温度は [14] に基づき 0.05 とした．パラメータは学習率 $5e-6$，バッチサイズ 512，1 epoch で AdamW [15] を用いて更新した．GPU メモリを効率的に活用するため勾配チェックポイントと LoRA [16] を使用した．bi-encoder の学習には約 2.5 時間かかった．bi-encoder に用いた学習データと異なり，kNN の検索対象は評価データに出現するスキルも含む学習データ 138,260 件とした．kNN の実装には faiss [17] を用いた．インデックスは検索時間を効率化するため検索対象の事例を事前にクラスタリングする Inverted file index とした．クラスタ数は 128，推論時の検索対象のクラスタ数は 16 とした．

kNNBE の式 (2) における λ および k を決定するため，開発データにおける RP@5 と RP@10 の和に基づいて値を決定した．具体的には $\lambda = 1.0$ と固定して kNN が最も高くなる値を示す k を $\{2^n\}_{n=1}^{10}$ から選び，次に λ を $\{0.1, 0.2, ..., 0.9\}$ の中から選んだ．その結果，$\lambda = 0.7$，$k = 64$ とした．その他の詳細は A.3 を参照されたい．

3.5 実験結果

表 1 は kNNBE とベースライン手法の RP@K およびスループットを表す．kNNBE は BE と kNN と比較して平均して高い RP@5，RP@10 となった．このことから kNNBE は 2 つのモジュールが補完しあい，高い評価値となっていることがわかる．IReRa$_{M-7b}$ はしばしばオントロジーに存在しないスキルを出力するため，他の手法に比べて RP@K が大幅に低かった．ローカル LLM を使用する手法が低い評価値となったことは Clavie ら [3] の報告とも一致する．TECH および TECHWOLF において BE は GPT-4 でリランキングする IReRa と同等かそれ以上の評価値を示した．GPT-4 を使用する手法は事前に候補スキルのセットを選択するために別のモデルを使用する必要があり，これが評価値を制限している可能性を示唆している．これまでの最高精度を示す IReRa と比較して，kNNBE はスループット 50 倍以上を保持

6) https://huggingface.co/datasets/jensjorisdecorte/Synthetic-ESCO-skill-sentences/tree/main

表 2: 学習データに存在しないスキルに対する RP の平均．括弧の数字は表 1 の RP Average との差を表す

	RP5	RP10
BE	55.73 (-2.80)	65.87 (-2.33)
*k*NN	52.14 (-1.61)	61.91 (-1.66)
*k*NNBE (ours)	57.41 (-2.75)	68.09 (-2.06)

表 3: 上から文，正解のスキル，bi-encoder と *k*NNBE の出力の 1 位 (左) から 5 位 (右) までのスキル．

文	Contribution to the administrative responsibilities of the Department and to CBS-wide tasks.
正解	“execute administration”
bi-encoder 出力	“assess administrative burden”, “manage administrative systems”, “office administration”, “ensure cross-department cooperation”, “assist with personal administration issues”
*k*NNBE 出力	“manage administrative systems”, “execute administration”, “office administration”, “manage university department”, “maintain professional administration”

しながら，RP@5 および RP@10 が 2 から 3 ポイント改善した．*k*NNBE のスループットは BE よりも低いもののその低下は 10%にとどまった．

3.6 分析

表 2 は評価データのうち学習データに出現しない未知のスキルのみを対象とした RP@k の比較結果を表す．*k*NNBE は BE と比較して表 1 の結果からの低下が小さい．このことから *k*NNBE は BE よりも未知のスキルに対してより頑健であることがわかる．

表 3 に実例と bi-encoder および *k*NNBE の出力結果を示す．bi-encoder は “assess administrative burden” や “office administration” など正解スキルと類似する文字列を持つ誤ったスキルを上位にする一方で，*k*NNBE は正解スキルを 2 位にランクした．*k*NN で取得したラベル付き合成文 64 件のうち 7 件が “execute administration” であった．近傍事例数の 64 件のうちで最も頻度が高く bi-encoder と比較してより上位にランクされることに貢献した．

4 関連研究

4.1 人事分野における自然言語処理

人事分野では職務記述書と求職者のマッチング [18, 19]，職務記述書の作成支援 [20]，キャリアパス予測 [21]，スキルマッピングなど様々な観点で研究が取り組まれている．スキルマッピングは人手による学習データ構築の難しさから多くの既存研究は学習データを必要としない手法に基づいている [22, 11, 4, 3, 5]．初期の研究は遠距離学習を活用した [22, 11]．ただし，遠距離学習で生成した学習データに含まれるノイズによってモデルの精度が低下するため近年の研究は LLM を使用している．Decorte ら [4] は bi-encoder の学習用に合成データを LLM を用いて生成した．*k*NNBE は Decorte ら [4] と類似するが推論時にラベル付き文を組み込むことが可能である点が異なる．また LLM でスキルをリランキングする手法も提案されている [3, 5]．この手法はプロンプト長制約のため bi-encoder によって候補スキルリストを事前に作成するためこのリストの作成方法が精度に影響を与える．

4.2 大規模マルチラベル学習

スキルマッピングは多くのラベルセットから関連性の高いラベルの部分集合の予測を目的とした，大規模マルチラベル学習（XMLC; Extream Multi-label Classification）タスクとみなせる．XMLC は分類モデルや検索モデルで解くことができる．*k*NNBE は検索モデルの一種である．そのため，あらかじめ定義されたクラスに分類する分類モデルとは異なり，*k*NNBE はモデルの学習後に新たに追加されたクラスに対しても予測可能である．さらに，新しいクラスとしてラベル付けされた事例は推論時に予測結果に影響するように使用できる．XMLC はウェブ検索クエリの分類 [23]，法的文書の分類 [24]，製品マッチング [25]，有害薬物事象の抽出 [26] など，幅広い分野で応用されている．本稿は *k*NNBE をスキルマッピングで評価したが，この手法はこれら他のタスクにも適用可能である．

5 おわりに

bi-encoder と *k*NN に基づく *k*NNBE を提案した．実験により *k*NNBE は bi-encoder よりも高い精度を示した．このことから合成データを *k*NN の検索対象として推論時に利用することで精度改善に貢献することがわかった．また *k*NNBE は文脈内学習を使った LLM に基づく既存手法よりも 50 倍以上のスループットを維持しながら 2 から 3 ポイント高い精度となることを確認した．

参考文献

[1] World Economic Forum. Strategies for the new economy: Skills as the currency of the labour market. **Centre for the New Economy and Society**, 2019.

[2] Johannes De Smedt, Martin le Vrang, and Agis Papantoniou. Esco: Towards a semantic web for the european labor market. In **LDOW@WWW**, 2015.

[3] Benjamin Clavié and Guillaume Soulié. Large language models as batteries-included zero-shot ESCO skills matchers. In **Proceedings of RecSys in HR**, Vol. 3490. CEUR-WS.org, 2023.

[4] Jens-Joris Decorte, Severine Verlinden, Jeroen Van Hautte, Johannes Deleu, Chris Develder, and Thomas Demeester. Extreme multi-label skill extraction training using large language models, 2023.

[5] Karel D'Oosterlinck, Omar Khattab, François Remy, Thomas Demeester, Chris Develder, and Christopher Potts. In-context learning for extreme multi-label classification, 2024.

[6] Rick Penn and Victor Huang. Job openings reach record highs in 2022 as the labor market recovery continues, 2023. Accessed: 2024-07-07.

[7] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In **Proceedings of EMNLP-IJCNLP**, pp. 3982–3992. Association for Computational Linguistics, 2019.

[8] Jinhyuk Lee, Zhuyun Dai, Xiaoqi Ren, Blair Chen, Daniel Cer, Jeremy R. Cole, Kai Hui, Michael Boratko, Rajvi Kapadia, Wen Ding, Yi Luan, Sai Meher Karthik Duddu, Gustavo Hernandez Abrego, Weiqiang Shi, Nithi Gupta, Aditya Kusupati, Prateek Jain, Siddhartha Reddy Jonnalagadda, Ming-Wei Chang, and Iftekhar Naim. Gecko: Versatile text embeddings distilled from large language models, 2024.

[9] Parishad BehnamGhader, Vaibhav Adlakha, Marius Mosbach, Dzmitry Bahdanau, Nicolas Chapados, and Siva Reddy. Llm2vec: Large language models are secretly powerful text encoders, 2024.

[10] Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Generalization through memorization: Nearest neighbor language models, 2020.

[11] Jens-Joris Decorte, Jeroen Van Hautte, Johannes Deleu, Chris Develder, and Thomas Demeester. Design of negative sampling strategies for distantly supervised skill extraction, 2022.

[12] Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. Mpnet: masked and permuted pre-training for language understanding. In **Proceedings of NIPS**. Curran Associates Inc., 2020.

[13] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. **CoRR**, Vol. abs/1807.03748, , 2018.

[14] Tianyu Gao, Xingcheng Yao, and Danqi Chen. SimCSE: Simple contrastive learning of sentence embeddings. In **Proceedings of EMNLP**, pp. 6894–6910. Association for Computational Linguistics, 2021.

[15] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In **Proceedings of ICLR**, 2019.

[16] Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In **Proceedings of ICLR**, 2022.

[17] Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. The faiss library, 2024.

[18] Shuqing Bian, Wayne Xin Zhao, Yang Song, Tao Zhang, and Ji-Rong Wen. Domain adaptation for person-job fit with transferable deep global match network. In **Proceedings of EMNLP-IJCNLP**, pp. 4810–4820. Association for Computational Linguistics, 2019.

[19] Changmao Li, Elaine Fisher, Rebecca Thomas, Steve Pittard, Vicki Hertzberg, and Jinho D. Choi. Competence-level prediction and resume & job description matching using context-aware transformer models. In **Proceedings of EMNLP**, pp. 8456–8466. Association for Computational Linguistics, 2020.

[20] Liting Liu, Jie Liu, Wenzheng Zhang, Ziming Chi, Wenxuan Shi, and Yalou Huang. Hiring now: A skill-aware multi-attention model for job posting generation. In **Proceedings of ACL**, pp. 3096–3104. Association for Computational Linguistics, 2020.

[21] Jun Zhu and Celine Hudelot. Towards job-transition-tag graph for a better job title representation learning. In **Findings of NAACL**, pp. 2133–2140. Association for Computational Linguistics, 2022.

[22] Mike Zhang, Kristian Nørgaard Jensen, and Barbara Plank. Kompetencer: Fine-grained skill classification in Danish job postings via distant supervision and transfer learning. In **Proceedings of LREC**, pp. 436–447. European Language Resources Association, 2022.

[23] Himanshu Jain, Venkatesh Balasubramanian, Bhanu Chunduri, and Manik Varma. Slice: Scalable linear extreme classifiers trained on 100 million labels for related searches. In **Proceedings of WSDM**, p. 528–536. Association for Computing Machinery, 2019.

[24] Ilias Chalkidis, Manos Fergadiotis, and Ion Androutsopoulos. MultiEURLEX - a multi-lingual and multi-label legal document classification dataset for zero-shot cross-lingual transfer. In **Proceedings of EMNLP**, pp. 6974–6996. Association for Computational Linguistics, 2021.

[25] Justin Chiu. Retrieval-enhanced dual encoder training for product matching. In **Proceedings of the EMNLP: Industry Track**, pp. 216–222. Association for Computational Linguistics, 2023.

[26] Karel D'Oosterlinck, François Remy, Johannes Deleu, Thomas Demeester, Chris Develder, Klim Zaporojets, Aneiss Ghodsi, Simon Ellershaw, Jack Collins, and Christopher Potts. BioDEX: Large-scale biomedical adverse drug event extraction for real-world pharmacovigilance. In **Findings of EMNLP**, pp. 13425–13454. Association for Computational Linguistics, 2023.

A　参考情報

A.1　ESCO

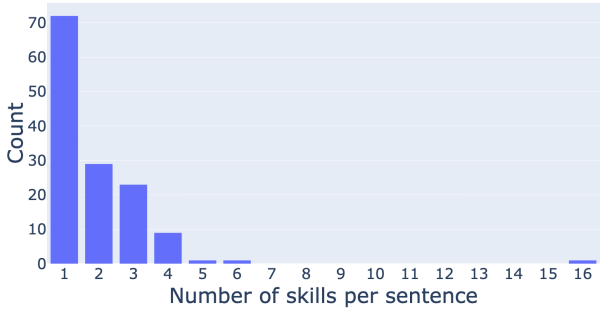
ESCO で定義されたスキルの一例を表 4 に示す．

表 4: ESCO で定義されたラベル名 (preferredLabel) の例

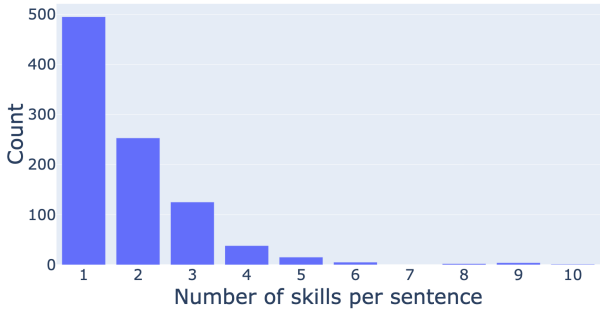
preferredLabel
manipulate dental material
compare alternative vehicles
follow nuclear plant safety precautions
preparation for parenthood
apply road transport environmental measures

A.2　正解スキル数の分布

開発データと評価データにおける文に対して付与された正解スキル数の分布を図 2 に示す．横軸が文に対して付与されたスキルの数，縦軸が文の数を表す．



(a) validation set



(b) test set

図 2: 正解スキル数の分布

A.3　実験設定

学習ステップの最初の 5%は学習率スケジューリングのために線形 warm up を適用した．推論時はバッチサイズを 64 に設定した．ミニバッチを作成する前に文をトークン数に基づいてソートし，トークン数が類似した文集合を 1 つのミニバッチにまとめた．