対訳データを使わないマルチリンガル表現学習に必要な分布構造とは何か

李凌寒 [1]　鶴岡慶雅 [1]
[1] 東京大学大学院 情報理工学系研究科
{li0123,tsuruoka@logos.t.u-tokyo.ac.jp

概要

複数の言語にまたがって意味を表現したベクトルを獲得するマルチリンガル表現学習アルゴリズムには、対訳データを用いないものがある。これらのアルゴリズムが言語間の対応を発見するために利用しているのは、自然言語の分布構造の類似性である。この分布構造とは具体的にどのようなものだろうか？本稿では、代表的なマルチリンガル表現学習アルゴリズムとして、Skip-gram + VecMap と Masked Language Modeling (MLM) を取り上げ、それらが利用している分布構造を、局所的/大局的なものという観点から区別して分析する。

1　はじめに

人間が異なる言語間の対応関係を理解するためには、２つの言語を明示的に結びつける情報が必要である。第二言語学習者にとっては、対訳辞書や教科書の対訳文が、その役目を担うだろう。また、多言語環境で生活する人々は、現実世界の物事との対応を通して、２つの言語の対応関係を発見することができる。一方で、ニューラルネットワーク (NN) は、人間とは異なる方法で言語間の対応を発見できることが知られている。

Skip-gram などのアルゴリズムで学習した異なる言語の単語ベクトル空間は、線形変換によって対応づけられることが知られており [1]、その変換は対訳データを使わずに発見することができる [2, 3]。また、Transformer と Masked Language Modeling (MLM) を用いて文脈化単語ベクトルを学習する際に、異なる言語のコーパスを混ぜて使うだけで、各言語の対応するフレーズのベクトル表現が近づく現象が報告されており [4]、この効果は各言語に共通の語彙が存在しない場合でも観察される [4, 5]。これらの表現学習アルゴリズムが、言語間対応発見のための手がかりにできるのは、単語の分布構造の類似性である。異なる言語圏であっても、人間は共通の認知基盤を持ち [6]、またいくつかの話題を共有しているため、そうした共通性が言語の分布構造に反映されていると考えることができる。

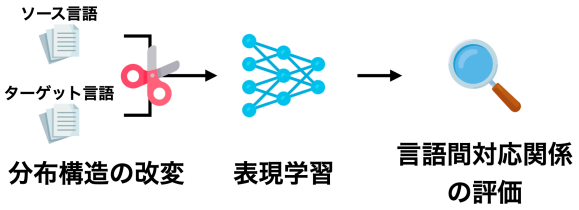

図 1　実験フレームワークの概略図。

それでは、具体的にどのような分布構造が、アルゴリズムの言語間対応発見の手がかりに使われているのだろうか。この問いを明らかにすることで、NN の自然言語の捉え方に関する理解を深め、未だに謎の多い NN という技術を上手く扱えるようになることに繋がる。本研究では、調査するアルゴリズムとして、Skip-gram + VecMap [3] および、Masked Language Modeling (MLM) を用いた手法を取り上げる。分析では、学習データとして、分布構造を改変したソース言語とターゲット言語のテキストを使い、得られたマルチリンガルベクトルを言語横断テキスト検索のタスクで評価する（図 1）。ここから、学習データの持つ分布構造の、アルゴリズムの言語間対応発見における必要性を調べることができる。

本稿では分析の観点として、ある単語の周囲数単語だけを含む局所的な分布と、周囲の文を含めた大局的な分布の、２つの分布の対比に注目する。実験の結果、Skip-gram + VecMap は、自然言語の局所的/大局的な分布のいずれかだけを手がかりにして言語間対応関係を学習できる一方で、MLM は、大局的な分布のみでは言語間対応を捉えられず、局所的な分布が必要であることが示唆された。

2 対訳データを使わないマルチリンガル表現学習

本論文の分析対象となるマルチリンガル表現学習手法は、入力として、ソース言語 S のテキストコーパス、ターゲット言語 T のコーパスを受け取り、それら言語のマルチリンガルベクトル表現モデルを得るアルゴリズムとして定式化できる。ここでは、Skip-gram + VecMap と Masked Language Modeling (MLM) の 2 つの手法を取り上げる。

2.1 Skip-gram + VecMap

Skip-gram [7] で学習されるモデルは、語彙 V 中の 1 単語に 1 つのベクトルを割り当てる静的単語ベクトルである。学習時には、コーパス中の単語列 $[w_1, ..., w_L]$ から、ターゲット単語 w_t と、その周囲 l 単語の範囲に存在する文脈語 $\{w_{t-l}, .., w_{t-1}, w_{t+1}, ..., w_l\}$ の組が取り出され、ターゲット単語のベクトルから、文脈語を予測するタスクでモデルを最適化する。本実験では、文脈窓のサイズを $l = 5$、学習するベクトルの次元を $d = 300$ と設定し、学習は gensim[1] のデフォルトハイパーパラメータで 15 エポック行った。

それぞれの言語のコーパスから別々に、Skip-gram による単語ベクトルを学習した後に、ソース言語の単語ベクトルをターゲット空間上に写像することで、マルチリンガルなベクトル空間を得る。このとき、ソース言語とターゲット言語のベクトル空間は構造が類似していることが多く、線形変換によって揃えることができることが分かっている [1]。対訳辞書 $D = \{(w_1^S, w_1^T), ..., (w_N^S, w_N^T)\}$ が存在する場合は、変換行列を $\boldsymbol{Z} \in \mathbb{R}^{d \times d}$ を、以下の最小二乗問題を解くことで得られる。

$$\arg\min_{\boldsymbol{Z}} \sum_{i=1}^{N} \left\| \boldsymbol{Z}\boldsymbol{w}_i^S - \boldsymbol{w}_i^T \right\|^2 \quad (1)$$

ここで、$\boldsymbol{w}_i^S, \boldsymbol{w}_i^T$ は、ソース言語とターゲット言語の、対応する単語のベクトルである。加えて、変換の品質を向上させるために，学習後のベクトルにはノルム正規化 [8, 9] を施し、$\boldsymbol{Z}$ には直交行列となるような制約 $\boldsymbol{Z}^\top \boldsymbol{Z} = \boldsymbol{I}$ を課している。

VecMap [3] は、このような変換行列 $\boldsymbol{Z}$ を、対訳データなしで獲得する手法である。まず、$\boldsymbol{W}^S, \boldsymbol{W}^T$ の統計情報から単語対訳辞書 D_0 を自動で構築する。ここで利用する統計情報は、単語ベクトル同士の類似度の分布であり、これは言語に関わらず似ていることが観察されている。単語ベクトルを集めた行列 $\boldsymbol{W} \in \mathbb{R}^{|V| \times d}$ の、類似度分布ベクトルは $\mathrm{sorted}(\sqrt{\boldsymbol{W}\boldsymbol{W}^\top})$ として計算される。ここで、sorted は、行ごとにベクトルの成分をソートする操作を表す。同様に、類似度分布ベクトルをソース言語とターゲット言語について求め、ソースとターゲット間で最近傍探索を行い、最も近い単語同士で単語対訳辞書 D_0 を構築する。

以後、自動で構築した対訳辞書 D_t を用いて 式 1 を解き、写像行列 $\boldsymbol{Z}_t$ を獲得するステップと、写像後のソース単語ベクトル $\boldsymbol{Z}_t\boldsymbol{W}^S$ とターゲット単語ベクトル $\boldsymbol{W}^T$ 同士で最近傍探索を行い、新たな対訳辞書 D_{t+1} を構築するステップを繰り返し、写像行列の精度を高めて最終的な写像行列 $\boldsymbol{Z}$ を獲得する。

2.2 Masked Language Modeling

MLM [10] で学習されるモデルは、単語列 $[w_1, ..., w_L]$ を受け取り、それぞれの単語に対応する文脈化ベクトル $[\boldsymbol{h}_{w_1}, ..., \boldsymbol{h}_{w_L}]$ を出力するエンコーダである。学習時には、入力単語列にランダムにノイズを適用し、ノイズを適用された単語の出力ベクトルから、元の単語を予測するタスクを解く。マスク付き言語モデリング自体には、言語間の対応を発見するような明示的な処理は存在しないにも関わらず、学習データに複数の言語を用いるだけで、各言語のフレーズの対応が取れるようなベクトルが学習され、これは言語間で語彙が共有されなくても起こる [4, 5]。本実験のエンコーダには、隠れ層の次元が 512、層の数が 6 の Transformer を用い、MLM の学習は Adam を学習率 $1e-4$ で用いて、バッチサイズ 128 で 3 エポック行った。

2.3 言語間対応が学習される条件とは何か

これらのアルゴリズムが、言語間の対応を発見するために使用しているのは、単語の分布構造の類似性だと考えられる。したがって、ソース言語とターゲット言語の分布構造にずれが生じる場合、言語間対応の学習精度は悪化する場合がある。たとえば、ソース言語とターゲット言語の言語学的な隔たりが大きい場合や、コーパスのドメインが異なる場合である [11]。この傾向は学習アルゴリズムによって異なり、MLM は静的単語ベクトルベースの手法に比べて、コーパスのドメインの違いには比較的頑健であることが知られている [4]。これは学習アルゴリ

1) https://radimrehurek.com/gensim/

ズムによって、対応関係発見に用いている分布構造が異なることが示唆しているが、その具体的な構造は明らかになっていない。本研究では、各アルゴリズムが、どのような分布構造を手がかりにして言語間の対応関係を発見しているのかを調査する。

3 分析実験

本実験では、テキストの分布構造を改変することで、アルゴリズムの構造への依存性を調べる。

3.1 学習データ

本実験は、ソース言語とターゲット言語の分布構造の差に注目するものではなく、両言語の分布構造が一致しているという前提の元で、言語間対応発見の必要条件となる分布構造を調べることが目的である。そこで、ここでは分布構造が一致しているが、共通の語彙を持たない別々の言語として、英語と、英語の語彙 ID をずらしたものを採用する。英単語の語彙を V、トークン $w \in V$ のソース言語における ID を $\mathrm{id}(w)$ としたとき、ターゲット言語における ID は $\mathrm{id}(w) + |V|$ が割り当てられる。

学習データとして、英語 Wikipedia のダンプファイルからランダムにサンプルした 100M 記事を用いた。前処理として、まず記事中のテキストに文分割・単語分割を施した後、記事をまたがずに、完全な文のみを含む 128 トークン以内のセグメントに分割する。単語分割には事前訓練済みの `bert-base-uncased`[2] のサブワードレベルのトークナイザを用いた。Skip-gram を訓練する際は、セグメントを分割前に戻し、`moses` のトークナイザ[3] を用いて単語レベルに再分割した。このセグメントが、表現学習アルゴリズムに与える系列データの 1 単位となる。この処理によって、合計約 548 万のテキストセグメントを得た。これらのデータをランダムに半分ずつに分割し、ソース言語とターゲット言語のコーパスとする。実験では、このデータをそのまま使用した学習と、分布構造を改変して学習を行った場合の結果を比較する。

3.2 言語間対応の評価

学習したマルチリンガルなベクトル表現モデルが、どれくらい正確にソース言語とターゲット言語の対応関係を発見しているかを評価するために、言語横断テキスト検索タスクを用いる。このタスクは、対訳テキストデータ $\{(s_1^{(S)}, s_1^{(T)}), ..., (s_N^{(S)}, s_N^{(T)})\}$ が与えられ、モデルは、ソース言語のテキスト $s_i^{(S)}$ から、対応するターゲット言語のデータ $s_i^{(T)}$ を全ターゲットテキスト中から探し当てるものである。

今回のベクトル表現モデルは、それぞれのテキスト s をベクトル表現 $\boldsymbol{s}$ にエンコードして、コサイン類似度に基づく最近傍探索によって検索する。Skip-gram + VecMap から獲得したモデルは、テキスト中の単語 $s = [w_1, ..., w_L]$ をそれぞれベクトルに変換し、それらを平均して得たベクトル $\frac{1}{L}\sum_i^L \boldsymbol{w}_i$ を用いて検索を行う。マスク付き言語モデリングで学習したエンコーダからも同様に、テキスト中の単語の文脈化ベクトルを平均したベクトルを用いる。マルチリンガルエンコーダのベクトル表現の対応は、最終層よりも中間層の方が良いため、結果ではエンコーダの第 4 層から抽出した結果を示す。評価スコアには平均逆順位を用いた。

3.3 大局的な分布と局所的な分布

テキスト中の単語分布は、文書のトピックや文法といった複数の要因で決まる。ここでは、大局的な分布と、局所的な分布の 2 つの区別に注目する。ここでいう大局的な分布とは、それぞれの単語が、文または周辺の文といった広い範囲を見たときに、どのような単語と共起するかの分布である。大局的な分布は、その単語がよく現れる話題を反映していると考えることができる [12]。一方で、局所的な分布は、より狭い数単語近傍にどのような単語が現れるかを指す。これは単語の話題的な意味に加えて、文法的な性質にも関連して決定される。

こうした分布に関する情報を除去する処理として、以下のデータ改変操作を導入する。

Corpus Shuffle は、コーパス中のトークンを、セグメントの境界を無視してランダムにシャッフルする。シャッフル後は、セグメントの長さと数がシャッフル前と同じになるようにコーパスを分割する。この操作により、コーパスは大局的な分布および局所的な分布のどちらも失い、アルゴリズムがアクセスできる意味のある統計情報は、トークンの頻度情報だけになる。

Sentence Shuffle はトークンを、セグメント内でランダムにシャッフルする。この操作により、データは局所的な分布を失うが、セグメント内での共起情報という大局的な分布は残る。

2) https://huggingface.co/bert-base-uncased
3) https://github.com/moses-smt/mosesdecoder/blob/master/scripts/tokenizer/tokenizer.perl

N-gram Shuffle　はコーパス中のトークンを N-gram ごとのブロックに分け、そのブロックを保ったままセグメントの境界を無視してランダムにシャッフルする。シャッフル後は、セグメントの長さと数がシャッフル前と同じになるようにコーパスを分割する。この操作により、コーパスは大局的な分布を失うが、N-gram の局所的な分布は残る。本実験では、N の大きさを 3 に設定した際の結果を示す。

以上の改変を加えたコーパスで学習した結果と、もともとのコーパスからの結果を表 1 に示す。

	Original	Corpus Shuffle	Sentence Shuffle	3-gram Shuffle
Skip-gram + VecMap	95.2	0.1	94.1	91.7
MLM	83.4	0.2	0.4	77.8

表 1　分布構造を改変したコーパスで訓練したマルチリンガルベクトルモデルを、言語横断テキスト検索タスクで評価した際の平均逆順位を 100 倍にして示している。

まず、これらのアルゴリズムは、明示されずとも言語の対応関係を発見でき、それには一定の分布構造が必要であることを確認する。改変を加えていないデータを用いた場合 (Original) は、各アルゴリズムともに 80 ポイントを上回るスコアを示しており、テキスト分布のみから高い精度で対応関係が発見されていることが分かる。一方で、コーパス全体をシャッフルして分布構造を無くした場合 (Corpus Shuffle) は、ほぼゼロに近いスコアを示し、頻度情報だけでは言語間対応を発見できないことが分かる。

次に、大局的/局所的な構造のどちらかだけでも、各アルゴリズムが対応関係を発見できるかを見ていく。大局的な構造を除去し、局所的な構造だけを残した場合 (3-gram Shuffle) は Skipgram + VecMap と MLM ともに、改変前 (Original) に近いスコアを示している。つまり、これらのアルゴリズムには言語の局所構造さえあれば、言語の対応関係が発見できるということである。セグメント中のトークンをシャッフルして、大局的な構造だけを残した場合 (Sentence Shuffle) は、Skipgram + VecMap は 94.1 ポイントと、改変前 (Original) とほぼ変わらないスコアを示している一方で、MLM は 0.4 とほぼゼロに近い値を示している。つまり、大局的な分布構造だけが与えられた場合、Skip-gram + VecMap は対応関係を発見できるが、MLM はできない。

原理的には、局所的/大局的な分布構造のいずれかの情報があれば、言語間の対応関係は発見可能であることを Skip-gram + VecMap の結果は示している。しかしなぜ MLM は、大局的な分布構造だけでは対応関係をベクトル空間上に反映しないのだろうか。

明示的な訓練信号なしで異なる言語のベクトル空間の共有が起こるメカニズムとして、タスクの学習の際にパラメータの効率的な活用をするインセンティブが働いているとする説がある。この説の根拠は、モデルのパラメータを大きくすると、モデルの発見する対応関係の精度が悪化するという観察である [13]。もしかしたら、大局的な構造だけを持つデータでの MLM は、パラメータ使用を効率化するインセンティブが働かないほどに、タスクが単純である可能性が考えられる。大局的な分布構造のみに基づいてマスクのついた単語を予測するタスクは、局所的な構造が保存されている場合に比べて、正解を絞り込むことが原理的に難しくなっている。実際に、MLM の 3-gram Shuffle と Sentence Shuffle の収束時の訓練損失の値を比べると、それぞれ 2.3 と 5.4 ポイント前後を示している。後者の収束時の損失が高いことは、タスクを解くための学習できる知識の上限が限られていることを示しており、これがベクトル空間の共有が起こらないことにつながっていると推測される。[4)]

4　おわりに

本研究では、対訳データを使わないマルチリンガル表現学習アルゴリズムが、どのような単語の分布情報を用いて、異なる 2 つの言語の対応関係を発見しているかを調査した。結果として、Skip-gram + VecMap は、大局的な分布構造さえあれば言語の対応関係を発見できるのに対して、MLM は局所的な分布構造がなければ、言語の対応関係を発見できないことが明らかとなった。

今後の研究では、暗黙的な言語の対応関係を発見するメカニズムを明らかにする必要がある。このためには多言語データ、Transformer、MLM に限る必要はなく、より広い視点から、潜在構造が類似する異なるデータで学習する際に、ニューラルネットワークの中間表現共有がどのように起こるかを考えることも有用であると思われる。

4)　MLM に用いる Transformer のエンコーダのパラメータ数を減らし、隠れ層サイズ 128、層の数 3 とした設定で評価を行ったが、対応関係発見の精度改善は見られなかった。また位置埋め込みがノイズとなり、対応関係の発見を妨げている可能性を考え、位置埋め込みを使用しないエンコーダで学習した場合も、結果は同じであった。大局的な分布のみでも、MLM で言語の対応関係が発見される条件が存在するかどうかを明らかにすることは、今後の課題としたい。

参考文献

[1] Tomas Mikolov, Quoc V. Le, and Ilya Sutskever. Exploiting similarities among languages for machine translation. **ArXiv**, Vol. abs/1309.4168, , 2013.

[2] Guillaume Lample, Alexis Conneau, Marc Aurelio Ranzato, Ludovic Denoyer, and Herve Jegou. Word Translation without Parallel Data. In **Proceedings of the 6th International Conference on Learning Representations**, Vancouver, Canada, 2018.

[3] Mikel Artetxe, Gorka Labaka, and Eneko Agirre. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In **Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 789–798, Melbourne, Australia, 2018.

[4] Alexis Conneau, Shijie Wu, Haoran Li, Luke Zettlemoyer, and Veselin Stoyanov. Emerging Cross-lingual Structure in Pretrained Language Models. In **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics**, pp. 6022–6034, Online, 2020.

[5] Stephen Mayhew Karthikeyan K, Zihan Wang and Dan Roth. Cross-Lingual Ability of Multilingual BERT: An Empirical Study. In **Proceedings of the 8th International Conference on Learning Representations**, Addis Ababa, Ethiopia, 2020.

[6] Hyejin Youn, Logan Sutton, Eric Smith, Cristopher Moore, Jon F. Wilkins, Ian Maddieson, W. Bruce Croft, and Tanmoy Bhattacharya. On the universal structure of human lexical semantics. **Proceedings of the National Academy of Sciences**, Vol. 113, pp. 1766 – 1771, 2015.

[7] Tomas Mikolov, Kai Chen, G. Corrado, and J. Dean. Efficient Estimation of Word Representations in Vector Space. In **International Conference on Learning Representations**, 2013.

[8] Mikel Artetxe, Gorka Labaka, and Eneko Agirre. Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In **Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing**, pp. 2289–2294, Austin, Texas, 2016.

[9] Chao Xing, Dong Wang, Chao Liu, and Yiye Lin. Normalized Word Embedding and Orthogonal Transform for Bilingual Word Translation. In **Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**, pp. 1006–1011, Denver, Colorado, 2015.

[10] Jacob Devlin, Mingwei 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**, pp. 4171–4186, Minneapolis, Minnesota, 2019.

[11] Anders Søgaard, Sebastian Ruder, and Ivan Vulić. On the limitations of unsupervised bilingual dictionary induction. In **Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 778–788, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[12] Omer Levy and Yoav Goldberg. Dependency-based word embeddings. In **Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)**, pp. 302–308, Baltimore, Maryland, June 2014. Association for Computational Linguistics.

[13] Philipp Dufter and Hinrich Schütze. Identifying elements essential for BERT's multilinguality. In **Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)**, pp. 4423–4437, Online, November 2020. Association for Computational Linguistics.