# 単語埋め込みの確率的等方化

横井 祥 [1,2]　下平 英寿 [3,2]
[1] 東北大学　[2] 理研 AIP　[3] 京都大学
yokoi@ecei.tohoku.ac.jp　shimo@i.kyoto-u.ac.jp

## 1　はじめに

単語埋め込みは現代の自然言語処理の必須ツールである．GloVe などの静的な単語埋め込み [1] および BERT などの文脈付き単語埋め込み [2] は広範なタスクの性能向上に貢献しており，またテキストを入出力にとるニューラルモデルのほぼすべてがその出入り口に埋め込み行列を持つ．埋め込む対象は文字やサブワードの場合もあるが，本稿では簡単のためこれらを総称して**単語埋め込み**と呼ぶ．

最近になって，単語埋め込み空間が実は少し「歪んで」いること [3, 4]，これを補正すると経験的に良好なパフォーマンスが得られること [3, 5, 6] がわかってきた．例えば，同じ方向にばかり単語ベクトルたちが固まっているよりも，**等方的**な方が（向きの意味で一様に分布している方が）埋め込みたちを「見分け」やすいと考えられる（図 1）．実際，単語埋め込みを等方的に補正することで後段タスクにおける性能が向上する [3, 7, 6]．

本項では，これらの埋め込み空間の歪みを補正手法の多くが暗黙的に単語頻度を一様だと見なしていることを指摘する．実際には単語頻度は経験的にべき分布に従っており [8]，このギャップが歪みを補正しきれない原因となる．この問題点を解決するため，単語頻度を適切に考慮させるための一貫的な指針を与える．さらに，等方性をはじめとした埋め込み空間の「歪み」の度合いも測定も同様の指針でおこなえることを示す．静的埋め込みを用いた実験では，提案する指針によって歪みをより正確に補正でき後段タスクの性能が向上すること，とりわけ白色化という簡単な処理が著しい性能向上をもたらすことを示す．また提案した等方性の尺度が埋め込みの品質をよく近似できること示す．

**■記法**　単語集合を $\{w_1, \ldots, w_n\}$，単語 $w_i$ に対応する単語埋め込みを太字 $\boldsymbol{w}_i \in \mathbb{R}^d$，対応する単語頻度を $p(w_i) \in [0, 1]$ で表す．また，一般のデータと特徴ベクトルを $x_i$，$\boldsymbol{x}_i$ で表す．

**図 1**　単語埋め込み空間の等方化

## 2　関連研究：埋込空間の歪みと補正

埋め込み空間の幾何的な歪みと補正のための手法群を概観する．いずれのタイプの埋め込みにおいても大きなトレンドは原点中心性と等方性である．

**■静的な単語埋め込み**　[9] は “良い” 単語埋め込みが満たすべき理論的仮定として等方性を挙げた．しかし実際の学習済みの単語埋め込みはこの仮定を満たさない．[3] は，学習済みの単語埋め込みを中心化（重心を原点に移動）し上位主成分方向を除去することで等方性が近似的に満たされることを示した．

**■動的な（文脈つきの）単語埋め込み**　BERT 等の動的な埋め込みは原点中心から外れており，等方性も満たさない [4]．中心化により文類似度計算における性能が改善することが報告されている [5]．

**■ニューラルモデル内の埋め込み行列**　言語モデルの埋め込み行列もやはり原点中心から外れており，等方性も満たさない [4]．これが学習がうまくいかない原因のひとつだとされており，埋め込み行列を等方的に保つ正則化が提案されている [6]．

## 3　既存手法の暗黙的な仮定と問題

前節で挙げた単語埋め込みの補正処理の多くは，特徴ベクトルの集合の前処理と似たアプローチが取られる．中心化を例としてこの一見自然な手続きを確認し，その問題点を指摘する．

### 3.1　直感的説明：単語頻度が一様と仮定

**■特徴ベクトルの前処理**　一般的なデータ解析・統計処理では，まず特徴ベクトル（観測）の集合が与

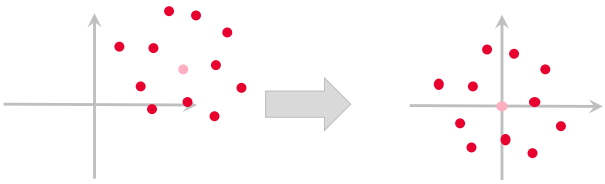

**図 2**　単語埋め込み空間の中心化.

えられる．たとえばある市の疫病の感染状況を分析したい場合には各市民の特徴量が収集される．つぎに特徴ベクトルの集合に適当な前処理がおこなわれる．たとえばもっとも基本的な前処理である**中心化**は，特徴ベクトルの集合の重心が原点と一致するよう特徴ベクトル全体を平行移動する（図 2），

**■単語埋め込みの補正**　単語埋め込みの場合も中心化による補正が盛んにおこなわれている [3, 5].

**■暗黙の仮定とその問題点**　この自然なアプローチのどこが問題なのだろうか？データ解析において各特徴ベクトル（たとえば各市民の特徴量）は平等な重みをもっていた．いずれの観測（市民）も，興味を持っている確率的な対象（たとえば市全体）からの無作為なサンプリングと見なせた．一方で単語ベクトルの場合，各単語は，興味を持っている確率的な対象（たとえばコーパス）から平等にサンプリングされたものでは**ない**．単語は実際には ['the', 'mean', 'of', 'the', 'data', 'matrix', 'is',...] のように偏りを持って出現している．データ解析と同様の処理は，単語の出現の列を単語集合 {'the', 'mean', 'of', 'data', ...} だと近似し，高頻度語も低頻度語も同様の扱いを与えることになる．

### 3.2　形式的説明：一様分布で期待値計算

既存手法の問題点を一言で述べれば「単語集合に対して期待値をとる際に一様分布を用いている」となる．以下これを**平均ベクトル**の計算を例に挙げて形式的に記述し，提案法の説明の準備としたい．

データ解析では，データの平均ベクトル $\boldsymbol{m}_{\boldsymbol{x}}$ を観測たち $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ の算術平均 $\widehat{\boldsymbol{m}}_{\boldsymbol{x}}$ で推定する．

$$\boldsymbol{m}_{\boldsymbol{x}} = \mathop{\mathbb{E}}_{x\sim p}[\boldsymbol{x}], \quad \widehat{\boldsymbol{m}}_{\boldsymbol{x}} = \sum_i \widehat{p}(x_i)\boldsymbol{x}_i = \sum_i \frac{1}{n}\boldsymbol{x}_i. \tag{1}$$

これは，$\widehat{p}(x_i) = 1/n$ と見なしている，つまり同様に観測された各データ $x_i$ を同様に扱っている点で適切である．一方で単語埋め込みの場合，

$$\boldsymbol{m}_{\boldsymbol{w}} = \mathop{\mathbb{E}}_{w\sim p}[\boldsymbol{w}], \quad \widehat{\boldsymbol{m}}_{\boldsymbol{w}}^{\textcircled{?}} = \sum_i \widehat{p}^{\textcircled{?}}(w_i)\boldsymbol{w}_i = \sum_i \frac{1}{n}\boldsymbol{w}_i \tag{2}$$

一見自然な算術平均 $\widehat{\boldsymbol{m}}_{\boldsymbol{w}}^{\textcircled{?}}$ を計算する処理は，単語頻度を $\widehat{p}^{\textcircled{?}}(w_i) = 1/n$ で見なしている，つまり各単語の頻度が一様だと仮定していることになる．

## 4　提案手法：単語頻度で期待値計算

以上の問題点を踏まえ，本稿では「**単語ベクトル集合に対して何らかの期待値をとる際は単語頻度分布を用いる**」という指針を提案する．以下では，平均ベクトルの計算，中心化に加え，等方性を確保するための前処理である**白色化**，および埋め込み空間の**等方性の度合いの推定**について，特徴ベクトルと対比させながら具体的な計算方法を述べる．$\widehat{p}(w_i)$ は，適当なコーパスにてカウントした $w_i$ の頻度，すなわち $p(w_i)$ の自然な推定量とする．

**■平均ベクトルと中心化**　単語埋め込みの**平均ベクトル**を推定する際は単語頻度で重み付き平均すれば良い．つまり $p(w_i)$ を頻度 $\widehat{p}(w_i)$ で推定すれば良い

$$\widehat{\boldsymbol{m}}_{\boldsymbol{x}} = \sum_i \frac{1}{n}\boldsymbol{x}_i, \quad \widehat{\boldsymbol{m}}_{\boldsymbol{w}} = \sum_i \widehat{p}_w(w_i)\boldsymbol{w}_i. \tag{3}$$

**中心化**する場合は各ベクトルから平均を差し引けば

$$\overline{\boldsymbol{x}}_i = \boldsymbol{x}_i - \widehat{\boldsymbol{m}}_{\boldsymbol{x}}, \quad \overline{\boldsymbol{w}}_i = \boldsymbol{w}_i - \widehat{\boldsymbol{m}}_{\boldsymbol{w}} \tag{4}$$

重心が原点に移動する（$\sum_i \frac{1}{n}\overline{\boldsymbol{x}}_i = \boldsymbol{0}$, $\sum_i \widehat{p}(w_i)\overline{\boldsymbol{w}}_i = \boldsymbol{0}$）．

**■分散共分散行列の固有値分解**　標準化，白色化，主成分分析など多くの前処理・補正処理が，固有値分解された分散共分散行列を利用する．特徴ベクトルの場合，**分散共分散行列** $S_{\boldsymbol{x}} \in \mathbb{R}^{d\times d}$ は

$$S_{\boldsymbol{x}} = \frac{1}{n-1}\overline{X}^\top\overline{X}, \quad \overline{X} := (\overline{\boldsymbol{x}}_1, \ldots, \overline{\boldsymbol{x}}_n)^\top \in \mathbb{R}^{n\times d} \tag{5}$$

で計算できる．一方単語埋め込み場合は，天下り的ではあるが，中心化し $\sqrt{\widehat{p}(w_i)}$ で重み付けした単語埋め込み行列 $\widetilde{W}$ を用いれば分散共分散行列が容易に計算できる

$$S_{\boldsymbol{w}} = \widetilde{W}^\top\widetilde{W}, \quad \widetilde{W} := (\sqrt{\widehat{p}(w_1)}\,\overline{\boldsymbol{w}}_1, \ldots)^\top \in \mathbb{R}^{n\times d}. \tag{6}$$

これで各要素 $(S_{\boldsymbol{w}})_{jk} = \sum_i \widehat{p}_w(w_i)\overline{\boldsymbol{w}}_i[j]\overline{\boldsymbol{w}}_i[k]$ が，単語頻度分布を考慮した（共）分散の推定量となる．

次に分散共分散行列を固有値固有ベクトル分解し

$$S = V\Lambda V^\top, \ \Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_d), \ V = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_d) \tag{7}$$

得られた $\Lambda, V \in \mathbb{R}^d$ で各種補正処理が実行できる．

**■白色化**　白色化は球状化とも呼ばれており，近似的にではあるが直接的に等方化を目指す処理である．上で求めた $\Lambda$ および $V$ を用いれば，

$$\widetilde{\boldsymbol{x}}_i = \Lambda^{-1/2}V^\top\overline{\boldsymbol{x}}_i, \quad \widetilde{\boldsymbol{w}}_i = \Lambda^{-1/2}V^\top\overline{\boldsymbol{w}}_i \tag{8}$$

によって適切に期待値をとった白色化 $\boldsymbol{w} \mapsto \widetilde{\boldsymbol{w}}$ が実現する．ただしこの標準的な白色化は分散の計算を伴い，結果として埋め込みの長さの影響を受ける．一方で単語埋め込みは，長さではなく向きが意味の計算に重要な役割を持つ [10]．そこで本稿では，固有値（方向毎の疎密の度合い）を単位超球上で計算してから白色化することを提案する．以下これを**方向白色化**と呼ぶ．具体的な計算方法としては，$\widetilde{W}$ の代わりに以下を用いて白色化をおこなえば良い

$$\widetilde{W}' := (\sqrt{\widehat{p}(w_1)}\,\overline{\boldsymbol{w}}_1/\|\overline{\boldsymbol{w}}_1\|, \ldots, \sqrt{\widehat{p}(w_1)}\,\overline{\boldsymbol{w}}_n/\|\overline{\boldsymbol{w}}_n\|). \quad (9)$$

**■等方性**　関連研究の節で述べたように等方性は埋め込みの歪みの尺度としてとりわけ重視されている．ここでは埋め込みの等方性の測定方法を検討する．まず，特徴ベクトル $\boldsymbol{x}$ に対する等方性は標準的には以下で定義される：$\mathbb{E}[\boldsymbol{x}\boldsymbol{x}^\top] = I$ [11]．すなわち $\frac{1}{n}\sum_i \boldsymbol{x}_i\boldsymbol{x}_i^\top$ と $I$ の間の近さをもって等方性を推定すれば良い．しかしこれを単語埋め込みに適用する際は，期待値計算における分布の補正した上で，**さらに**定義が持つ次の問題を解決する必要がある．(1) 向きが意味を保つにも関わらず，各埋め込みのノルムに影響を受ける．(2) 埋め込み全体のスケールに影響を受ける．そこで本稿では，(1) 各埋め込みを正規化した上で $\mathbb{E}[\boldsymbol{w}\boldsymbol{w}^\top]$ を計算し，(2) その固有値分布がフラットかどうか（固有値分布のエントロピーを正規化したもの）で等方性を測る．以下，この等方性の尺度を**正規化エントロピー** $H(\boldsymbol{w})$ と呼ぶ．

### 4.1　サンプリングをおこなう手法との関連

一部の手法は単語頻度に基づくサンプリングを暗におこなっており，結果，単語頻度による期待値計算が実現している [12, 5]．本稿の枠組みはこれらの手法がうまく動く正当化を与えるものでもある．

## 5　実験

学習済み単語埋め込みを歪みを各手法で補正することを試み，歪みは実際にとれるのか，また後段タスクの性能は向上するのかを確認した．

### 5.1　実験設定

本稿ではもっとも基本的と考えられる静的埋め込みを用い，また後段タスクとして，今日でも静的埋め込みが BERT 等の文脈つき埋め込みよりも有用に使われているタスクを選定した[1)]．データセット等は標準的に用いられているものを採用した．

**■単語埋め込み**　英語語彙の静的な単語埋め込み **GloVe**[1] および **word2vec**[13] を用いた．

**■埋め込み空間の補正手法**　**中心化**，**白色化**，および**方向白色化**を用いた．さらに，ベースラインとして all-but-the-top（**ABBT**）[3] を用いた．

**■補正の効果の確認のための空間の歪みの測定手法**

**原点中心性**：$Z(\boldsymbol{w}) := \|\mathbb{E}[\boldsymbol{w}]\|/\mathbb{E}[\|\boldsymbol{w}\|] \geq 0$．小さいほど重心が原点に近い．

**等方性**：$H(\boldsymbol{w}) \leq 1$ (§4)．大きいほど等方的．

以下，期待値を一様分布で（算術平均で）とった際は「**見かけの等方性**」のように書く．

**■補正の効果の確認のための後段タスク**

**単語類似度タスク**：埋め込み間のコサイン類似度と単語の意味的類似度との順位相関を確かめる．データセットとして **WordSim**[14]，**MEN**[15] を用いた．

**教師なし文類似度タスク**：単語埋め込みの和で文埋め込みを作り，単語類似度尺度と同様の手続きで人手評価とのピアソン相関[2)]を確かめる．データセットとして **STS-B**[16]，**Twitter**[17] を用いた．またベースラインとして **uSIF**[12] 用いた．

**■単語頻度**　期待値計算に必要な単語頻度として，Wikipedia 英語版のカウントデータを用いた [18][3)]．

### 5.2　実験結果

GloVe での実験結果は表 1 の通り．word2vec での実験結果は付録 A を参照されたい．まず GloVe の中心化を例に，一様な処理と重み付き処理の違いを概観する．

GloVe を**一様に**中心化すると，歪みの度合いが補正前の埋め込みに比べて悪化した．加えてすべての後段タスクにおける性能が悪化した．これまでの分野の知見を踏まえると驚くべき結果である．[9, 3] 等によれば，中心化によって等方性が一次近似の意味で満たされ，埋め込みの経験的な性能も向上するはずだからだ[4)]．しかし本稿の視点に立てば，今回の実験結果は「**一様な**中心化が**見かけの** $Z$ や $H$ を向上させた」「真の $Z$ や $H$ は悪化しており後段タスクの性能も悪化した」と自然に解釈できる．

対照的に，単語頻度を考慮して中心化した場合，$Z$，$H$ がともに向上し，またすべての後段タスクの性能が向上した．ほかの補正処理においても同様の

1)　ただし文類似度タスクの Twitter に関しては，文語体中心の STS-B と対比させる口語体のデータセットとして採用した．

2)　順位相関係数ではなく相関係数を用いるのは慣例に基づく．

3)　https://github.com/PrincetonML/SIF/

4)　[3] では中心化とそれに次いで上位主成分除去を提案している．中心化単体での経験的な効果は報告されていなかった．

**表 1**　単語埋め込み空間の補正処理とその効果.

| | | 原点中心性 ↓ | | 等方性 ↑ | | 単語類似度タスク ↑ | | 文類似度タスク ↑ | |
|---|---|---|---|---|---|---|---|---|---|
| | | みかけの | 実際の | みかけの | 実際の | MEN | WordSim | STS-B | Twitter |
| GloVe | | 0.32 | 0.49 | 0.91 | 0.66 | 80.49 | 79.57 | 45.57 | 29.35 |
| ＋中心化 | 一様 | 0.00 | 0.62 | 0.95 | 0.58 | 77.87 | 75.10 | 42.86 | 27.06 |
| | **重み付け** | 0.51 | **0.00** | 0.81 | 0.77 | 80.54 | 80.46 | 55.58 | 43.40 |
| ＋白色化 | 一様 | 0.00 | 0.30 | 1.00 | 0.78 | 83.21 | 82.46 | 48.50 | 42.61 |
| | **重み付け** | 0.11 | **0.00** | 0.97 | 0.90 | 83.78 | 81.49 | 70.70 | **49.10** |
| **＋方向白色化** | 一様 | 0.00 | 0.30 | 1.00 | 0.79 | 83.29 | **82.69** | 49.22 | 42.79 |
| | **重み付け** | 0.13 | **0.00** | 0.96 | **0.91** | **83.98** | 81.01 | **72.15** | 45.50 |
| ＋ ABBT | 一様 | 0.00 | 0.28 | 0.97 | 0.77 | 82.58 | 81.19 | 57.02 | 47.13 |
| | **重み付け** | 0.30 | 0.23 | 0.93 | 0.80 | 81.58 | 81.01 | 59.69 | 46.21 |
| ＋ uSIF | | | | | | | | 71.5 | |

**表 2**　空間の歪みと後段タスクの性能の相関係数 ×100.

| GloVe | 単語類似度タスク ↑ | | 文類似度タスク ↑ | |
|---|---|---|---|---|
| | MEN | WordSim | STS-B | Twitter |
| 見かけの $Z$ | -20.41 | 3.17 | 18.91 | 14.01 |
| $Z$ ↓ | **-59.14** | **-57.00** | **-82.94** | **-84.41** |
| 見かけの $H$ | 47.06 | 24.41 | 1.65 | 9.14 |
| $H$ ↑ | **82.43** | **74.45** | **89.53** | **90.48** |

傾向が見られた．期待値のとりかたを適切に修正することで，既存の知見を現実の言語の分布に沿った形で効果的に運用できるようになったと言える．

**■歪みの指標と後段タスクの性能の関係**　次に，実験で作成したすべての種類の埋め込みを用いて「埋め込みの歪み度合い（$Z$ および $H$）」と「後段タスクにおける性能」の相関を測ったところ，一貫して強い “正の” 相関を持つことがわかった（表 2）[5)]．このことから，適切に計算した歪みの度合いは，単語埋め込みの今後の研究に次のように役立つと考えられる．(1) 埋め込み空間の歪み（たとえば $Z$ や $H$）を見ることで，後段タスクでの実際の実験を介さずとも，経験的な性能をある程度予測できる可能性がある．(2) 埋め込み空間の歪みを補正したい場合は，単語頻度で重み付けした尺度（たとえば $Z$ や $H$）を最適化するという指針で，高性能な補正処理が手に入る可能性がある．

**■白色化に関する考察**　文類似度タスク（STS-B）において，方向白色化が uSIF と呼ばれる強力なベースラインを上回った．これは驚くべき結果と言わざるを得ない．方向白色化は簡単なごく単純な処理にすぎず，一方で uSIF は (1) 単語埋め込みの補正 (2) 単語の重要度推定 (3) 文埋め込みの補正の 3 手法併せた手法だからである．方向白色化は等方性（$H$）を近似的に向上させる提案だったことを踏まえれば，$H$ を直接的な最適化する手法の考案は魅力的な将来の研究と言える．

別名「球状化」と呼ばれる白色化は，等方性 $H$ を安定して向上させ，また後段タスクの性能を向上させた．これは，埋め込みの固有値分布の観点から見ると驚くべき結果である．単語埋め込み空間の固有値分布はまったくフラットではなく裾が減衰することが知られており [3]．単語埋め込み同士をほとんど見分けることのできない軸が多数存在する．白色化は，これらの「役立たない」軸方向を無理やり引き伸ばしてこれを使わせる処理であり，一般的な次元削減の指針に反する．ただし，自然言語処理の埋め込みの研究では，上位主成分の除去 [3, 7, 19, 18, 12] というやはり一般的な次元削減の指針に反する処理が一定の成果を収めている．これらの知見を総括する理論が待たれる．

## 6　おわりに

単語埋め込み空間の歪みの度合いを測り，また歪みを補正する際に，単語頻度の一様性が暗に仮定されることがある．本研究ではこれを修正する一貫的な指針「期待値を単語頻度でとる」を示した．また，等方性を測定する尺度として正規化エントロピー $H$ を，等方性を向上させる手法として方向白色化をそれぞれ提案した．実験の結果，提案した修正指針によって既存の補正処理がうまく動くようになり，また適切に計算した歪みの尺度は埋め込みの品質の有用な評価指標となることがわかった．

**■謝辞**　本研究は JST ACT-X JPMJAX200S の支援を受けたものです．また本研究は JSPS 科研費 20H04148 の助成を受けたものです．

5)　$Z$ は小さい方が空間の歪みが少ない（好ましい）．

# 参考文献


[1] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. GloVe: Global Vectors for Word Representation. In *EMNLP*, pp. 1532–1543, 2014.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL*, pp. 4171–4186, 2019.

[3] Jiaqi Mu and Pramod Viswanath. All-but-the-Top: Simple and Effective Postprocessing for Word Representations. In *ICLR*, 2018.

[4] Kawin Ethayarajh. How Contextual are Contextualized Word Representations? Comparing the Geometry of BERT, ELMo, and GPT-2 Embeddings. In *EMNLP*, pp. 55–65, 2019.

[5] Xi Chen, Nan Ding, Tomer Levinboim, and Radu Soricut. Improving Text Generation Evaluation with Batch Centering and Tempered Word Mover Distance. In *First Workshop on Evaluation and Comparison of NLP Systems*, pp. 51–59, Online, 11 2020. Association for Computational Linguistics.

[6] Lingxiao Wang, Jing Huang, Kevin Huang, Ziniu Hu, Guangtao Wang, and Quanquan Gu. Improving Neural Language Generation with Spectrum Control. In *ICLR*, 2020.

[7] Tianlin Liu, Lyle Ungar, and João Sedoc. Continual Learning for Sentence Representations Using Conceptors. In *NAACL*, pp. 3274–3279, 2019.

[8] George K Zipf. *Human Behaviour and the Principle of Least Effort*. Addison-Wesley, 1949.

[9] Sanjeev Arora, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. A Latent Variable Model Approach to PMI-based Word Embeddings. *TACL*, Vol. 4, pp. 385–399, 12 2016.

[10] Sho Yokoi, Ryo Takahashi, Reina Akama, Jun Suzuki, and Kentaro Inui. Word Rotator's Distance. In *EMNLP*, pp. 2944–2960, Online, 11 2020. Association for Computational Linguistics.

[11] Mark Rudelson. Random Vectors in the Isotropic Position. *Journal of Functional Analysis*, Vol. 164, No. 1, pp. 60–72, 1999.

[12] Kawin Ethayarajh. Unsupervised Random Walk Sentence Embeddings: A Strong but Simple Baseline. In *Rep4NLP*, pp. 91–100, 7 2018.

[13] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality. In C J C Burges, L Bottou, M Welling, Z Ghahramani, and K Q Weinberger, editors, *NIPS*, pp. 3111–3119, 2013.

[14] Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. Placing Search in Context: The Concept Revisited. *ACM Transactions on Information Systems*, Vol. 20, No. 1, pp. 116–131, 1 2002.

[15] Elia Bruni, Gemma Boleda, Marco Baroni, and Nam-Khanh Tran. Distributional Semantics in Technicolor. In *ACL*, pp. 136–145, 7 2012.

[16] Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. SemEval-2017 Task 1: Semantic Textual Similarity Multilingual and Crosslingual Focused Evaluation. In *SemEval*, pp. 1–14, 8 2017.

[17] Wei Xu, Chris Callison-Burch, and William B. Dolan. SemEval-2015 Task 1: Paraphrase and Semantic Similarity in Twitter (PIT). In *SemEval*, pp. 1–11, 6 2015.

[18] Sanjeev Arora, Yingyu Liang, and Tengyu Ma. A Simple but Tough-to-Beat Baseline for Sentence Embeddings. In *ICLR*, 2017.

[19] Tianlin Liu, Lyle H. Ungar, and João Sedoc. Unsupervised Post-processing of Word Vectors via Conceptor Negation. In *AAAI*, pp. 6778–6785, 2019.

# A　付録：word2vec での実験結果

**表 3**　単語埋め込み空間の補正処理とその効果．word2vec．

| | | 原点中心性 ↓ | | 等方性 ↑ | | 単語類似度タスク ↑ | | 文類似度タスク ↑ | |
|---|---|---|---|---|---|---|---|---|---|
| | | みかけの | 実際の | みかけの | 実際の | MEN | WordSim | STS-B | Twitter |
| word2vec | | 0.28 | 0.33 | 0.89 | 0.77 | 78.2 | 77.39 | 61.54 | 30.54 |
| ＋中心化 | 一様 | 0.00 | 0.32 | 0.94 | 0.76 | 79.51 | 75.10 | 60.05 | 30.78 |
| | **重み付け** | 0.51 | **0.00** | 0.81 | 0.77 | 77.78 | 77.15 | 63.17 | 32.87 |
| ＋白色化 | 一様 | 0.00 | 0.25 | 1.00 | 0.81 | 76.94 | 74.41 | 60.15 | 32.78 |
| | **重み付け** | 0.11 | **0.00** | 0.97 | 0.90 | 78.88 | 78.04 | 70.36 | 36.67 |
| **＋方向白色化** | 一様 | 0.00 | 0.23 | 1.00 | 0.82 | 76.59 | 74.81 | 61.98 | 34.31 |
| | **重み付け** | 0.10 | **0.00** | 0.96 | **0.91** | 79.14 | **78.33** | **71.46** | 33.59 |
| ＋ ABBT | 一様 | 0.00 | 0.28 | 0.96 | 0.79 | **80.22** | 77.07 | 61.23 | 31.90 |
| | **重み付け** | 0.19 | **0.00** | 0.93 | 0.84 | 78.05 | 77.47 | 64.15 | **38.11** |