文書間距離尺度の特性

浅原 正幸 (国立国語研究所)*　加藤 祥 (国立国語研究所)

On the Document Distance Metric with n-gram and p-mer

Masayuki Asahara (NINJAL)　Sachi Kato (NINJAL)

要旨

本研究では文書間距離尺度（類似度・相関係数）の特性を様々なコーパスを用いて評価する。まず、代表的な文書間距離尺度として単純な 1-gram 形態素ベクトルから、n-gram、p-mer、順序尺度の数理的な構造を整理する。次に各文書間距離尺度を要約・語釈・再話課題コーパスを用いて評価する。多人数による課題間における尺度のふるまい、個人による再話の複数回間の尺度のふるまい、口述・タイプ入力・筆述など言語生成方法間の尺度のふるまいを分析する。

1. はじめに

まず、最初に語順に対する順序尺度を含めた距離空間・類似度・カーネル・相関係数により既存の自動評価指標の整理を行う。先行研究の文献では連続記号列を表す部分文字列 (substring) とギャップを許す部分列 (subsequence) との混同が見られ、定性的な議論が弱い。本稿では、大きく分けて一致部分文字列による尺度・一致部分列による尺度・ベクトル型順序尺度・編集型順序尺度の四つに分類し議論する。

次に言語生産・受容過程の多様性を 4 種類の尺度により評価する。複数人が同一課題を実施した場合の各尺度の分散や、同一人が同一課題繰り返し実施した場合の各尺度の分散などを検討する。生産過程においては口述・筆術・タイプ入力の 3 種類について評価し、課題においては要約・語釈・再話について評価する。

なお、本節で用いる用語や記号の定義は浅原ほか (2015) の付録を参照されたい。

2. 文書間距離

2.1 LCStr, LCS

2.1.1 記号列と文字列と部分文字列と部分列

最初に記号列と文字列と部分文字列と部分列の違いについて確認する。

何らかの全順序が付与されている記号集合のことを**記号列**と呼ぶ。本稿では記号列ベクトル $s = \langle s_1, \ldots, s_m \rangle, t = \langle t_1, \ldots, t_m \rangle$ などで表現する。文書は文字 (character) ベースの記号列もしくは形態素解析後の形態素 (morpheme) ベースの記号列とみなすことができる。

評価する記号列上の連続列のことを**文字列 (string)** と呼ぶ。記号列の要素が文字 (character) である場合を「文字ベースの文字列 (character-based string)」、記号列の要素が形態素

* masayu-a@ninjal.ac.jp

(morpheme) である場合を「形態素ベースの文字列 (morpheme-based)」と呼ぶこととする。

記号列に対して隣接性と順序を保持した部分的記号列のことを**部分文字列 (substring)** と呼ぶ。長さ n の部分文字列を特に n-gram 部分文字列と呼ぶ。記号列 s の i 番目の要素からはじまる n-gram 部分文字列を $s_{i,\ldots,i-n+1}$ で表現する。

記号列に対して順序を保持した部分的記号列のことを**部分列 (subsequence)** と呼ぶ。隣接性は保持しなくてよい。長さ p の部分列を特に p-mer 部分列と呼ぶ。記号列 s の p-mer 部分列を、インデックスベクトル $\vec{i} = \langle i_1, \ldots, i_p \rangle (1 \leq i_1 < i_2 < \cdots < i_p \leq |s|)$ を用いて、$s[\vec{i}]$ と表す。

2.1.2　最長共通部分文字列 (Longest Common String: LCStr) 長

最長共通部分文字列 (Longest Common String) の abbreviation は LCS だが、一般には 2.1.2 に示す最長共通部分列 (Longest Common Subsequence) のことを LCS と呼ぶことが多い。本稿では前者を LCStr, 後者を LCS と呼び、区別する。

記号列 s, t を与えた際の最長共通部分文字列を次式で定義する: $\mathrm{LCStr}(s,t) = \mathrm{argmax}_{s_{i,\ldots,i-n+1} | \exists j, s_{i,\ldots,i-n+1} = t_{j,\ldots,j-n+1}} n$. 記号列 s, t を与えた際の最長共通部分文字列長 (LCStr 長) を次式で定義する: $|\mathrm{LCStr}|(s,t) = \max_{\forall i, \forall j, s_{i,\ldots,i-n+1} = t_{j,\ldots,j-n+1}} n$. これを [0,1] 区間に正規化すると以下のようになる: $\mathrm{Score}_{\mathrm{LCStr}}(s,t) = \frac{2 \cdot |\mathrm{LCStr}|}{|s|+|t|}$.

2.1.3　最長共通部分列 (Longest Common Subsequence: LCS) 長と Levenshtein 距離

記号列 s, t を与えた際の最長共通部分列 (Longest Common Subsequence: LCS) を次式で定義する: $\mathrm{LCS}(s,t) = \mathrm{argmax}_{s[\vec{i}] \exists \vec{j}, s[\vec{i}] = t[\vec{j}]} |\vec{i}|$. 記号列 s, t を与えた際の最長共通部分列長 (LCS 長) を次式で定義する: $|\mathrm{LCS}(s,t)| = \max_{\forall \vec{i}, \forall \vec{j} : s[\vec{i}] = t[\vec{j}]} |\vec{i}|$. [0,1] 区間に正規化すると、以下のようになる: $\mathrm{Score}_{\mathrm{LCS}}(s,t) = \frac{2 \cdot |\mathrm{LCS}|}{|s|+|t|}$. なお、挿入のコストを 1、削除のコストを 1、代入のコストを 2(もしくは代入を禁止) した場合の Levenshtein 距離 (編集型) と LCS 長の関係は以下のようになる: $\mathrm{d}_{\mathrm{Levenshtein}}(s,t) = |s| + |t| - 2 \cdot |\mathrm{LCS}|$. さらに LCS は §2.2.2 で示すとおり、対称群上の編集型距離のうちの Ulam 距離と深く関連し、一種の順序尺度であるとも考えられる。

2.2　関連するカーネル・順序尺度

以下では、関連するカーネルおよび順序尺度について確認する。

2.2.1　カーネル・距離 (文字列の共有)

畳み込みカーネルのうち系列データに対するカーネル (Shawe-Taylor ほか (2010)) は、共通する可能な部分文字列・部分列を数え上げる。いずれも効率よく計数する方法が提案されている。また、適切に正規化することにより部分文字列・部分列の共有についての距離やスコアを規定することができる。

各カーネルの説明に入る前に、スコア化 ([0,1] 区間正規化) について示す。カーネルのスコア化は次式により行われる: $\mathrm{Score}_{K\_}(s,t) = \frac{K\_(s,t)}{\|K\_(s,s)\| \|K\_(t,t)\|}$.

■全部分文字列カーネルと文字列長加重全部分文字列カーネル　全部分文字列カーネル (All String Kernel or Exact Matching Kernel) は共通する全ての部分文字列の数を数える。長さ n の部分文字列 u を座標とする特徴量空間 $\Phi^*_{\mathrm{str}} : \sigma^* \to F_{\mathrm{all\_str}} \sim R^{|\sigma|^*}$ 但し $\Phi^*_{\mathrm{str}} = (\phi^*_u(s))_{u \in \sigma^*}$ を考える。$K_{\mathrm{n\text{-}gram}}(s,t) = \langle \Phi^*_{\mathrm{str}}(s), \Phi^*_{\mathrm{str}}(t) \rangle_{F_{\mathrm{all\_str}}} = \sum_{u \in \sigma^*} \phi^*_u(s) \phi^*_t(s), (\text{但し} \phi^*_u(s) = |\{i | s_{i\ldots *} = u\}|)$. カーネル関数を直接計算すると以下のよう

になる: $K_{\text{all\_seq}}(s,t) = \sum_{n=1}^{\min(|s|,|t|)} \sum_{i=1}^{|s|-n+1} \sum_{j=1}^{|t|-n+1} \delta(s_{i\ldots i+n-1}, t_{i\ldots i+n-1})$.

このカーネルは、提案された 2002 年ごろではバイオインフォマティクスなど特定の分野以外では有効な用途が提案されていない。言語処理の場合、得られる n-gram に対して加重をかけることが一般に行われている。例えば、文字列長に対して加重をかけたものを文字列長加重全部分文字列カーネル (Length Weighted All String Kernel or Length Weighted Exact Matching Kernel) と呼ぶ。$K_{\text{all\_seq}}(s,t) = \sum_{n=1}^{\min(|s|,|t|)} \sum_{i=1}^{|s|-n+1} \sum_{j=1}^{|t|-n+1} \omega_{|s|}\delta(s_{i\ldots i+n-1}, t_{i\ldots i+n-1})$. ここで ω_n は長さ n に対する重みを表す。このカーネルと次の n-スペクトラムカーネルは Suffix Tree を用いて効率よく計算する方法が提案されている。

■n-スペクトラムカーネル　n-gram スペクトラムカーネル (Spectrum Kernel) は共通する長さ n の部分文字列 (n-gram) の数を数える。長さ n の部分文字列 u を座標とする特徴量空間 $\Phi^n_{\text{str}} : \sigma^* \to F_{\text{n-gram}} \sim R^{|\sigma|^n}$(但し $\Phi^n_{\text{str}} = (\phi^n_u(s))_{u\in\sigma^n}$) を考える。$K_{\text{n-gram}}(s,t) = \langle\Phi^n_{\text{str}}(s), \Phi^n_{\text{str}}(t)\rangle_{F_{\text{n-gram}}} = \sum_{u\in\sigma^p} \phi^n_u(s)\phi^n_t(s)$(但し$\phi^n_u(s) = |\{i|s_{i\ldots i+n-1} = u\}|$) 直接計算すると以下のようになる: $K_{\text{n-gram}}(s,t) = \sum_{i=1}^{|s|-n+1} \sum_{j=1}^{|t|-n+1} \delta(s_{i\ldots i+n-1}, t_{j\ldots j+n-1})$.

■全部分列カーネル　全部分列カーネルは共通するすべての部分列の数を数える。任意の長さの部分列 v を座標とする特徴量空間 $\Psi^*_{\text{seq}} : \sigma^* \to F_{\text{all\_seq}} \sim R^{|\sigma|^\infty}$(但し$\Psi^*_{\text{seq}}(s) = (\psi^*_v(s))_{v\in\sigma^*}$) を考える。$K_{\text{all\_seq}}(s,t) = \langle\Psi^*_{\text{seq}}(s), \Psi^*_{\text{seq}}(t)\rangle_{F_{\text{all\_seq}}} = \sum_{v\in\sigma^*} \psi^*_v(s)\cdot\psi^*_v(t)$(但し$\psi^*_v(s) = |\{\vec{i}|s[\vec{i}] = v\}|$). $K_{\text{all\_seq}}(s,t)$ は以下のように再帰的に計算することにより $O(|s||t|)$ で計算することができる。ϵ を空記号列とすると $K_{\text{all\_seq}}(s,\epsilon) = K_{\text{all\_seq}}(t,\epsilon) = 1$ とし、$K_{\text{all\_seq}}(s,t)$ が求まると $K_{\text{all\_seq}}(s{\cdot}a,t) = K_{\text{all\_seq}}(s,t) + \sum_{1\le i\le|t|, j:t_j=a} K_{\text{all\_seq}}(s, t_{i\ldots j-1})$ と s 再帰的に定義できる。さらに $\tilde{K}_{\text{all\_seq}}(s{\cdot}a,t) = K_{\text{all\_seq}}(s,t_{i\ldots j-1})$ とすると、$\tilde{K}_{\text{all\_seq}}(s{\cdot}a,t{\cdot}b) = \tilde{K}_{\text{all\_seq}}(s{\cdot}a,t) + \delta(a,b)K(s,t)$ と t 再帰的に定義できる。

■固定長部分列カーネル　固定長部分列カーネルは共通する長さ p の部分列 (p-mer) の数を数えあげる。長さ p の部分文字列 v を座標とする特徴量空間 $\Psi^p_{\text{seq}} : \sigma^* \to F_{\text{p-mer}} \sim R^{|\sigma|^p}$(但し$\Psi^p_{\text{seq}}(s) = (\psi^p_v(s))_{v\in\sigma^*}$). を考える。$K_{\text{p-mer}}(s,t) = \langle\Psi^p_{\text{seq}}(s), \Psi^p_{\text{seq}}(t)\rangle_{F_{\text{p-mer}}} = \sum_{v\in\sigma^p} \psi^p_v(s)\cdot\psi^p_v(t)$. ここで $\psi^p_v(s) = |\{\vec{i}|s[\vec{i}] = v\}|$ とする。

■ギャップ加重部分列カーネル　ギャップ加重部分列カーネル: p-mer の部分列の数え上げの際に隣接性を考慮して重み λ を加重する。長さ p の部分列 v を座標とする特徴量空間 $F_{\text{p-mer}}$ を考える。$K_{\text{gap\_p-mer}}(s,t) = \langle\Psi^{gap\text{-}p}_{\text{seq}}(s), \Psi^{gap\text{-}p}_{\text{seq}}(t)\rangle_{F_{\text{p-mer}}} = \sum_{v\in\sigma^p} \psi^{gap\text{-}p}_v(s)\cdot\psi^{gap\text{-}p}_v(t)$ ここで $\psi^{gap\text{-}p}_v(s) = \sum_{\vec{i}:v=s[\vec{i}]} \lambda^{l(\vec{i})}$ とし、$l(i) = |s_{i_1,\ldots,i_{|v|}}|(\vec{i} = \langle i_1,\ldots,i_{|v|}\rangle)$ とする。

2.2.2　順序尺度

以下では順序尺度について考えるが、神嶌 (2009) が詳しい。基本的には同じ長さ m の二つの順位ベクトル $\mu,\nu \in S_m$ に対する 2 種類の距離を考える。

■順位ベクトル型距離　一つ目の距離は「順位ベクトル型」の距離で順位ベクトルを m 次元空間中の点を表すベクトルとみなし、ベクトル空間上の距離を定義する。ベクトル空間を θ-ノルム採用すると以下のようになる: $d_{\|\text{Rank}\|_\theta}(\mu,\nu) = (\sum_{i=1}^m |\mu(i) - \nu(i)|^\theta)^{1/\theta}$. ここで $\theta = 1$

表 1　指標・スコア・距離・カーネル・相関係数の関係まとめ

	スコア $[0,1]\uparrow$	距離 $[0,\infty]\downarrow$	カーネル $[0,\infty]\uparrow$	相関係数 $[-1,1]\uparrow$
部分文字列系	$\mathrm{Score}^{(\gamma)}_{K\mathrm{all\_str}}$		(加重) 全部分文字列 §2.2.1	
(n-gram)	$\mathrm{Score}^{(\gamma)}_{K\text{n-gram}}$		n-スペクトラム §2.2.1	
部分列系	$\mathrm{Score}^{(\gamma)}_{K\mathrm{all\_seq}}$		(加重) 全部分列 §2.2.1	
(p-mer)	$\mathrm{Score}^{(\gamma)}_{K\text{p-mer}}$		p-mer 部分列 §2.2.1	
	$\mathrm{Score}^{(\gamma)}_{K\mathrm{gap\_p\text{-}mer}}$		加重 p-mer 部分列 §2.2.1	
順序系 §2.2.2	$\mathrm{Score}_{\|\mathrm{rank}\|_\theta}$			
(ベクトル型)	$\mathrm{Score}_{\mathrm{footrule}}$	$d_{\mathrm{footrule}(\theta=1)}$		
	$\mathrm{Score}_{\mathrm{Spearman}}$	$(d_{\mathrm{Spearman}(\theta=2)^2})$		Spearman's ρ
	$\mathrm{Score}_{\mathrm{Hamming}}$	d_{Hamming}		
順序系 §2.2.2 (編集型)	$\mathrm{Score}_{\mathrm{Kendall}}$	d_{Kendall}		Kendall's τ
(最長一致部分列長)	$\mathrm{Score}_{\mathrm{LCS}}$	d_{Ulam}§2.1.3		
(加重最長一致部分列長)	$\mathrm{Score}^{(\gamma)}_{\mathrm{WLCS}}$			
(最長一致部分文字列長)	$\mathrm{Score}_{\mathrm{LCStr}}$			

の場合、特に Spearman footrule と呼ぶ。$d_{\mathrm{Footrule}}(\mu,\nu) = (\sum_{i=1}^{m} |\mu(i) - \nu(i)|)$. $\theta = 2$ の場合は通常の Euclid 距離だが、この Euclid 距離を 2 乗したものを特に Spearman 距離と呼ぶ。$d_{\mathrm{Spearman}}(\mu,\nu) = (\sum_{i=1}^{m} |\mu(i) - \nu(i)|^2)$. Spearman 距離は、距離の公理のうち対称性と正定値性を満たす。しかし、Euclid 距離を 2 乗したものなので三角不等式を満たさないが、慣習的として距離として扱われる。さらに [-1, 1] 区間に正規化したものは Spearman の順位相関係数 ρ として知られている。Spearman's $\rho = 1 - \frac{6 \cdot d_{\mathrm{Spearman}}(\mu,\nu)}{m^3 - m}$. この値は順序尺度に基づく二つの順位ベクトル μ, ν の Pearson 相関関係と等しい[(1)]。その他、順位ベクトルの同一順位のものが同じ要素である要素数を数えた Hamming 距離がある。$d_{\mathrm{Hamming}}(\mu,\nu) = \sum_{i=1}^{m} \delta(\mu(i), \nu(i))$. Hamming 距離は文字列上で代入 (コスト 1) のみを許した編集距離としても解釈できる。

■対称群上の編集型距離　二つ目の距離は「編集型」の距離である。

順序ベクトルを記号列とみなした場合、順位ベクトル μ をもうひとつの順位ベクトル ν に変換するために必要な最小操作数を Levenshtein 距離について述べた。以下では、順序ベクトルを対称群とみなした場合の編集型距離について述べる。編集に許される操作によっていくつかの距離のバリエーションがある。

Kendall 距離 d_{Kendall} は順序ベクトルを対称群とみなした際に隣接互換によって置換する最小回数によって定義される。言い換えると隣接する対象対を交換 (Swap) する操作の最小回数を用いたものである。Kendall 距離は、二つの順位ベクトル中の $\frac{m(m-1)}{2}$ 個の対象対のうち逆順になっている対の数に等しい。$d_{\mathrm{Kendall}} = \min(\mathrm{argmax}_q\, \delta((\Pi_{q=1}^{q} \pi_2(k_q, k_q + 1)) \cdot \mu, \nu)) = \sum_{i=1}^{m} \sum_{j=i+1}^{m} \chi(i,j)$. ここで χ は対象対 $\langle i, j \rangle$ が同順のとき 0、逆順のとき 1 を返す指示関数: $\chi = \begin{cases} 1 & \text{if } (\mu(i) - \mu(j))(\nu(i) - \nu(j)) < 0, \\ 0 & \text{if } (\mu(i) - \mu(j))(\nu(i) - \nu(j)) \geq 0 \end{cases}$ これをスコアとして使いやすくするために [0,1] 区間の範囲に正規化すると以下のようになる: $\mathrm{Score}_{\mathrm{Kendall}} = 1 - \frac{2 \cdot d_{\mathrm{Kendall}}(\mu,\nu)}{m^2 - m}$. こ

表 2　指標評価に使う言語資源

言語資源名	収集場所	生成過程	繰り返し	取得人数	摘要
BCCWJ-SUMM_C	クラウドソーシング	タイプ入力	なし	100-200	19 文書の要約
BCCWJ-SUMM_L	実験室	筆述	3 回	のべ 47	8 文書の要約
GROSS_C	クラウドソーシング	タイプ入力	なし	71,111,113	鶏・兎・象の語釈
GROSS_L	実験室	筆述	4 回	7,6,3	鶏・兎・象の語釈
RETELLING_I	実験室	口述	10 回	5	インタビュー
RETELLING_K	実験室	口述	3 回	3,3,3	怪談 3 種の再話
RETELLING_M	実験室	筆述	4 回	10	物語「桃太郎」の再話

れを [-1,1] 区間の範囲に正規化したものは Kendall の順位相関係数 τ として知られている。Kendall's $\tau = 1 - \frac{4 \cdot d_{\mathrm{Kendall}}(\mu,\nu)}{m^2-m}$.

Ulam 距離 d_{Ulam} は順序ベクトルを対称群とみなした際に連続した順序ベクトル部分列 $i, i+1, \ldots, j-1, j\rangle$ の巡回置換の操作のみによって置換する最小回数によって定義される。これは「本棚の本の入れ換え」で例えられる。順位ベクトル μ で並んでいる本棚の本を順位ベクトル ν に並び替えるために、ある要素を抜いて別の場所に挿入するということを行う。Ulam 距離は同じ要素が記号列に存在しないという前提のもと、最大共通部分列距離と以下の関係にあることが知られている。$d_{\mathrm{Ulam}}(\mu,\nu) = m - |\mathrm{LCS}(\mu,\nu)|$ これを [0,1] 区間の範囲に正規化すると以下のように正規化最大共通部分スコアと同じになる: $\mathrm{Score}_{\mathrm{Ulam}}(\mu,\nu) = 1 - \frac{d_{\mathrm{Ulam}}(\mu,\nu)}{m} = \frac{|\mathrm{LCS}(\mu,\nu)|}{m} = \mathrm{Score}_{\mathrm{LCS}}(\mu,\nu)$.

2.3　スコアの一般化

以上、指標・スコア・距離・カーネル・相関係数を議論してきた。まとめると表 1 のようになる。各スコアと人手の評価結果という観点からすると、平尾ほか (2007) のように、表 1 にあげたすべてのスコア $\mathrm{Score}_{\_} \in \{\mathrm{Score}_*\}$ の加重相乗平均 (下式) を考え、加重 $\omega_{\_}$ と各スコアに付随するパラメータを各指標の従属性や相関に注意しながら人手の評価指標との回帰により求めれば良い: $\overline{\mathrm{Score}_*} = \sum^{\omega} \sqrt{\Pi \mathrm{Score}_{\_}^{\omega_{\_}}}$. このスコアのあり方については議論すべき点がいくつかある。まず、substring(部分文字列: n-gram 系) と subsequence(部分系列: p-mer 系) との違いを踏まえる。次に最長一致部分文字列は対称群上の編集型距離である Ulam 距離と深く関連する。さらに順序に対する順位ベクトル型距離と編集型距離の間には様々な不等式が成り立つ。本稿では**スコアの一般化についてはこれ以上踏み込まない**。次節以降各スコアがさまざまな言語資源上でどのようなふるまいをするのかについてみていきたい。

3.　評価に用いる言語資源

ここでは研究室で有する言語資源のテキスト対のスコアを検証することにより、各スコアがとらえようとしているものが何なのかを分析する。表 2 に利用する言語資源について示す。まず言語生産の目的として、要約 (BCCWJ-SUMM) と語釈 (GROSS) と再話 (RETELLING) の 3 種類の言語資源を準備する。要約と語釈については、クラウドソーシングにより安価で大量にデータを得る手法 (タイプ入力) と実験室にて被験者に繰り返し同一課題を依頼してデータを得る手法 (筆述) の 2 種類の方法を用いた。再話のデータについては既存のデータを用いた。再

話については、言語生産形態として筆述による形態と口述による形態のデータを準備した。

以下各言語資源について解説する。

3.1 BCCWJ-SUMM_C

BCCWJ-SUMM_C は BCCWJ の新聞記事の要約を Yahoo! クラウドソーシング (15 歳以上の男女) により被験者実験的に作成したものである。

BCCWJ の 1 サンプルには複数の記事が含まれており、それを記事単位に分割したうえで元文書集合 19 文書を構築した。元文書集合は BCCWJ コアデータ PN サンプル (優先順位 A) から選択した。40 文字毎に改行した元文書を画像として提供し、実験協力者に 50-100 文字に要約せよという指示で収集した。実験協力者の環境は PC 環境に限定した。元文書毎に約 100〜200 人の実験協力者が要約に従事した。実験実施時期は 2014 年 9 月である。

得られたデータには、文字数制限を守っていないもの・実験の趣旨を理解していないもの・既に実験を行った実験協力者から同一回答を提供されたと考えられるものなどが含まれており、これらを排除したものを有効要約とする。統計分析においてこの有効要約のみを用いる。

3.2 BCCWJ-SUMM_L

BCCWJ-SUMM_L は BCCWJ の新聞記事の要約を実験室環境で筆述により作成したものである。BCCWJ-SUMM_C で用いた元文書を印刷紙面で提供し、実験協力者に 50-100 文字に要約せよという指示で収集した。一つの元文書に対して、3 回まで繰り返して要約文作成を行った。繰り返しに際しては、特別に「前と同じ要約文を作成してください」などといった指示は行わず、質問された場合にも「自由に要約文を作成してください」と教示した。実験協力者は原稿用紙上で筆述 (鉛筆と消しゴム利用) で要約を行い、そのデータを電子化した。現在のところデータは 8 文書のべ 47 人分に限定した。

3.3 GROSS_C

GROSS_C は語釈文を Yahoo! クラウドソーシング (15 歳以上の男女) により被験者実験的に作成したものである。

「その動物を知らない人がどのようなものかわかるように説明してください」と教示し、同意した実験協力者は兎 (単語親密度 6.6)・鶏 (6.4)・象 (同 6.0) の 3 種類から対象物を選択回答した(2) 。150 文字以上 250 文字以内で 3 文字以上の同文字連続は認めない設定とした。実験協力者 300 名を募集したところ得られた解答数は、鶏:71・兎:111・象:113(295/300) であった。

3.4 GROSS_L

GROSS_L は語釈文を実験室環境で筆述により収集したものである。

実験協力者 8 名 (20 代-50 代の男女) に、GROSS_C と同様に「その動物を全く知らない人がどのようなものかわかるように説明してください」と教示した。実験協力者は、10 分間で兎 (単語親密度 6.6)・鶏 (6.4)・象 (同 6.0) の 3 種類から 2 種類の対象物を選択回答した。目安として 5 分経過時にブザー音を鳴らした。選択した対象物について同様に記述を繰り返すことを 4 回行った。得られた解答数は、兎 7 人分× 4 回、鶏 6 人分× 4 回、象 3 人分× 4 回である。平均 145 文字 (max 227 文字, min 85 文字) を得た。

3.5 RETELLING_I

最初の再話のデータは「独話 Retelling コーパス」保田ほか (2013a,b) である。このコーパスは宮部ほか (2014) でも用いられている。

実験協力者は 5 名で、同一人が同内容をそれぞれ 10 回独話を繰り返した。就職活動を前提とした模擬面接の設定で、実験協力者は自ら予め用意した「学生生活で力を入れてきたこと (3 分間程度)」についての独話を行った。同内容を繰り返すことや何回依頼するかは知らせていない。5 人分× 10 回 (50 話分) の独話を取得した。面接官 (聴衆) は有無を交互とした。奇数回 (1・3・5・7・9 回) は聴衆なしの独話、偶数回 (2・4・6・8・10 回) は聴衆に対する独話である。聴衆には、聴いていることを表すために頷くことのみを許可しており、話者への質問や意見など、発話は一切行わなかった。収録は録音と録画を行い、音声データを書き起こした。

被験者によってインタビュー内容が異なるために、統計分析においては同一被験者の回数間のスコア (RETELLING_I(T)) のみを評価する。

3.6 RETELLING_K

次の再話のデータは怪談を繰り返し口述したものであり、保田・荒牧 (2012) によるものである。実験協力者は 3 名[(3)] で、実験は 1 名ずつ個別に行った。実験協力者は怪談を聞いたのち、その怪談について 3 回の再話を行った。怪談は 3 種類を用意したため、各人 9 回の語りを行った。語りに関しては、「怪談として他の人に伝えるよう話す」との指示をした。既存の物語では、個人の記憶による先入観の影響が予測されたため、4 分間程度の新規な怪談を 3 本作成した。実験環境はビデオカメラと録音機により、録音と録画を行った。聴衆の影響を除去するために、聴衆は設置しなかった。本稿では音声データを書き起こしたものを用いる。

3.7 RETELLING_M

最後の再話のデータは桃太郎の物語を筆述で繰り返し記述したものであり、保田 (2014) によるものである。実験協力者 10 名 (20 代-50 代の男女) に,「桃太郎の物語を全く知らない人に向けて記述してください」と教示し、実験協力者は 10 分間で記述 (筆述) した。同様に記述を繰り返すことを 4 回行った。平均延べ 284 語 (min:150 語・ max:451 語)、異なり語 107 語 (min:74 語・max:152 語) の「桃太郎」 10 人分× 4 回 (40 話分) を取得した。

4. 評価

本節では前節で述べたコーパスを用いて文書間距離がどのようにふるまうかを観察する。利用する文書間距離は以下の 30 種類である。

- n-gram スペクトラム (1,2,3,4) (char/mrph)
- n-gram 以下スペクトラム (≤2,≤3,≤4) (char/mrph)
- p-mer 部分列 (2,3,4) (char/mrph)
- p-mer 以下部分列 (≤2,≤3,≤4) (char/mrph)
- 1-gram スペクトラム +Footrule (char/mrph) (=Spearman)
- 1-gram スペクトラム +Kendall (char/mrph)

http://goo.gl/nBeMeZ にそれぞれの距離空間によるスコアの平均値 (Mean) と標準偏差 (SD) を示す。スコアについて “_char” は文字単位の記号列として評価したもの、“_mrph” は形態素単位の記号列 (MeCab-0.98+IPADIC-2.7.0 による) として評価したものである。シャピロ・ウィルク検定の結果、ほとんどの場合 p 値が 0.05 未満であり、正規分布とはいえない傾向が見られた。

unigram(n-gram(1)) を用いた場合、要約と語釈は中程度、再話はかなり高いスコアを達成している。GROSS_L(T) がほぼ再話と同程度のスコアで一方、BCCWJ-SUMM_L(T) が低いことから、要約を繰り返す際の言語生産の特殊性が見られる。要約を繰り返す際には、回数毎に文章中の重要箇所を変更するサンプル・被験者が存在し、標準偏差も高くなっている。

Bigram(n-gram(2)), skip-bigram(p-mer(2)) を用いた場合、異なる被験者間のスコアと繰り返し間のスコアとの間に差が見られるようになる。これは何らかの個々人の文体差が形態素の連接に影響を与えているのではないかと考える。

Bigram(n-gram(2)) と skip-bigram(p-mer(2)) の間の差として、語釈の場合のみ bigram のスコアが下がることがわかる。語釈という課題の都合上、物語や要約と異なり、情報の提示順が変わることも考えられる。しかし、順序尺度である Kendall のスコアでは bi-gram のスコアほど顕著な差が見られなかった。単語の隣接性が語釈のみ下がるというスコアの振る舞いについては今後検討していきたい。

クラウドソーシングと研究室内被験者実験との差 (BCCWJ-SUMM_C ⇔ BCCWJ-SUMM_L(P), GROSS_C ⇔ GROSS_L(P)) については、各スコア・各課題 (要約・語釈) で差が見られなかった。

4.1 課題間の評価

以下、課題間を比較するために、6 種類の評価軸を分析する。殆どの場合、正規分布であることも等分散であること (F 検定による) も仮定できない。ここではウィルコクソンの順位和検定 (0.05 未満で 2 群の代表値が左右にずれている) を行う。(4)

- 実験室における複数人の課題間の違いの評価
  BCCWJ-SUMM_L(P) ⇔ GROSS_L(P) ⇔ RETELLING_K(P) ⇔ RETELLING_M(P)
  - BCCWJ-SUMM_L(P) ⇔ GROSS_L(P)
    文字単位の評価の場合 n-gram(2,3,4)_char, Kendall_char に有意差が見られた。
    形態素単位の評価の場合 n-gram(2,3,4,≤2,≤3,≤4)_mrph, Footrule_mrph, Kendall_mrph に有意差が見られた。
  - BCCWJ-SUMM_L(P) ⇔ RETELLING_K(P)
    n-gram(3,4)_mrph **以外**で有意差が見られた。
  - BCCWJ-SUMM_L(P) ⇔ RETELLING_K(M), GROSS_L(P) ⇔ RETELLING_{K,M}(P)
    全てのスコアについて、有意差が見られた。
  - RETELLING_K(P) ⇔ RETELLING_M(P)
    n-gram(≤3,≤4)_mrph,p-mer(3,4,≤3,≤4) で有意差が見られた。

  要約 ⇔ 語釈間は n-gram(1) で有意差が見られなかった。同じ文字・同じ形態素を使うという観点では一致度のレベルが等しいが、語の連接や順序尺度が入ると有意差が見られることがわかった。グラフの見た目から語釈の方が語の連接や順序尺度の一致度が低い。これは語釈の目的としては情報の提示順に重要性がないことが伺える。
  要約 ⇔ 再話、語釈 ⇔ 再話の間においては有意差が見られた。再話は同じ話をするという特性から、一致度が高くなる一方、要約・語釈は目的を達成するがために同じ表現を用いなければならないという制約がなく、低くなる傾向にある。

- 実験室における単一人の回数間距離の課題間の違いの評価
  BCCWJ-SUMM_L(T) ⇔ GROSS_L(T) ⇔ RETELLING_I(T) ⇔ RETELLING_K(T) ⇔ RETELLING_M(T)
  - BCCWJ-SUMM_L(T) ⇔ GROSS_L(T)
    文字単位の評価の場合 n-gram(2,3,4)_char, Kendall_char に有意差が見られた。
    形態素単位の評価の場合 n-gram(2,3,4,≤2,≤3,≤4)_mrph, Footrule_mrph, Kendall_mrph に有意差が見られた。
  - BCCWJ-SUMM_L(T) ⇔ RETELLING_{I,K,M}(T), GROSS_L(T) ⇔ RETELLING_{I,K,M}(T)
    全てのスコアについて、有意差が見られた。
  - RETELLING_I(T) ⇔ RETELLING_K(T)
    文字単位の評価の場合 n-gram(1,4,≤2)_char, p-mer(2,≤2)_char に有意差が見られた。
    形態素単位の評価の場合、全てのスコアに有意差が見られた。
  - RETELLING_I(T) ⇔ RETELLING_M(T)
    Kendall_char **以外**について有意差が見られた。
  - RETELLING_I(T) ⇔ RETELLING_M(T)
    文字単位の評価の場合 n-gram(2,≤2,≤3,≤4)_char, p-mer(2,3,4,≤2,≤3,≤4)_char に有意差が見られた。
    形態素単位の評価の場合、n-gram(1,2,≤2,≤3,≤4)_mrph, p-mer(2,3,4,≤2,≤3,≤4)_mrph に有意差が見られた。

  複数人間の評価ではなく、複数回間の評価でも、前項と同じ傾向が見られる。
  再話課題の間については、形態素単位の評価においては、三課題のうちどの二つ組においても有意差が出る傾向にある。口述による再話 (RETELLING_{I,K}) の方が筆述による再話 (RETELLING_M) より一致度が高くなる。また口述による再話においては、自身の体験に基づく再話 (RETELLING_I) の方が、他者から聞いた話の再話 (RETELLING_K) よりも一致度が高くなることが認められた。
- クラウドソーシングにおける課題間の違いの評価
  BCCWJ-SUMM_C ⇔ GROSS_C について、全てのスコアについて、有意差が見られた。
  クラウドソーシングにおける課題間の違いについても、前項と同じ傾向が見られる。
- 要約課題においてクラウドソーシングと実験室との違いの評価 (複数人間)
  BCCWJ-SUMM_C ⇔ BCCWJ-SUMM_L(P) について、n-gram(2)_char, n-gram(3)_char, n-gram(4)_char にのみ有意差が見られた。これは、タイプ入力 (BCCWJ-SUMM_C) と筆述 (BCCWJ-SUMM_L(P)) とで、表記ゆれの統制の差が出たのではないかと考える。
- 語釈課題においてクラウドソーシングと実験室との違いを評価する (複数人間)
  GROSS_C ⇔ GROSS_L(P) について、n-gram(2,3,4)_char, n-gram(2,3,4)_mrph, Footrule_mrph, Kendall_mrph **以外**について有意差が見られた。語釈においては、クラウドソーシングの場合 wikipedia や辞書サイトからのコピーが行われる傾向にある一方、実験室の場合は参照文献なしで筆述で行うために差が出たのではないかと考える。
- 複数人間距離と単一人の回数間距離の違い
  BCCWJ-SUMM_L(P) ⇔ BCCWJ-SUMM_L(T), GROSS_L(P) ⇔ GROSS_L(T), RETELLING_K(P) ⇔ RETELLING_K(T), RETELLING_M(P) ⇔ RETELLING_M(T) について、全てのスコアについて有意差が見られた。基本的に単一人が実施したほうが一致度が高いと考えられるが、統計分析の結果からもそれが確認できる。

4.2 スコア毎の特性

前節の課題間の議論から考えられるスコア毎の特性について論じる。

文字 n-gram はタイプ入力と筆述入力の差として認められることから、表記ゆれレベルで一致度が下がる特性があると考える。形態素 n-gram は再話と繰り返しで顕著に高くなることから、個々人の言い回しや文体などを反映していると考える。

p-mer, Footrule, Kendall などは語順などを反映していると考えるが、情報の提示順が重要な

要約・再話で一致度が高い一方、語釈などにおいては低い傾向にあることがわかった。

n-gram, p-mer ともに *n*, *p* の値が高くなるにつれてスコアが低くなる。このために有意差が出にくくなる傾向にある。n-gram, p-mer ともに *n* (or *p*) 以下のスコアとして設定した場合に、より低い *n* (or *p*) の方が一致の多くなる傾向にあるために、より高い *n* (or *p*) の差異が見られなくなる傾向がある。これはスコアの自然な解釈であると考えられるが、何らかの用途で長い n-gram, p-mer を重要視する場合には加重を行う必要があるだろう。

n-gram(1)_* と Kendall_* とを比較した場合、 n-gram(1)_*では有意差が出るが、順序尺度を入れた Kendall_* では有意差が出ないスコアの組み合わせがいくつかあった。これは文字順・語順の一致度が低い場合に、順序尺度を掛けあわせたがために全体の一致度の差がなくなったことが考えられる。

5. おわりに

本稿では、文書間距離尺度の数理的構造を説明した。カーネル・距離・相関係数とどう対応しているのかを説明し、n-gram 系、p-mer 系、順序尺度の三つに抽象化した。次に様々な言語資源を用いて各指標で用いられているスコアの特性を明らかにした。要約・語釈・再話からなる 7 種類の言語資源を用いて、課題・多人数産出・複数回産出・産出手段 (口述・筆述・タイプ) の軸を用いて、どのような分散が観察されるかを確認した。

謝辞

本研究の一部は科研費基盤 (B)「言語コーパスに対する読文時間付与とその利用」、科研費若手 (B)「コーパスから取得しやすい情報と取得しにくい情報の研究」、国語研基幹型共同研究プロジェクト「コーパスアノテーションの基礎研究」および国語研「超大規模コーパス構築プロジェクト」によるものです。

参考文献

浅原正幸・加藤祥・今田水穂 (2015).「単一文書自動要約のための言語資源構築に向けて」　情報処理学会研究報告 2015-NL-220 巻.

Shawe-Taylor，John・Nello Cristianini・大北剛 (訳) (2010).『カーネル法によるパターン解析 (Kernel Methods for Pattern Analysis)』, 第 11 章　共立出版.

宮部真衣・四方朱子・久保圭・荒牧英治 (2014).「音声認識による認知症・発達障害スクリーニングは可能か？–言語能力測定システム”言秤”の提案–」　グループウェアとネットワークサービスワークショップ 2014.

神嶌敏弘 (2009).「順序の距離と確率モデル」　人工知能学会研究会資料 SIG-DMSM-A902-07.

平尾努・奥村学・安田宣仁・磯崎秀樹 (2007).「投票型回帰モデルによる要約自動評価法」　人工知能学会論文誌, 22:2, pp. 115–126.

保田祥 (2014).「同じ話を成立させる語 –「桃太郎」を「桃太郎」として成立させる語彙–」　社会言語科学会第 33 回大会発表論文集.

保田祥・荒牧英治 (2012).「人が同じ話を何度もするとどうなるか？：繰り返しによって生じる物語独話の変化」　日本認知科学会第 29 回.

保田祥・田中弥生・荒牧英治 (2013a).「繰り返しにおける独話の変化」　社会言語科学会第 31 回大会発表論文集, pp. 190–193.

保田祥・田中弥生・荒牧英治 (2013b).「同じ話であるとはどういうことか」　社会言語科学会第 32 回大会発表論文集.