語彙的連鎖とトピックモデルに基づくテキストセグメンテーション

山村崇† 嶋田和孝‡

†九州工業大学大学院 情報工学府

‡九州工業大学大学院 情報工学研究院

{t_yamamura, shimada}@pluto.ai.kyutech.ac.jp

1 はじめに

対話要約は，複数人対話の理解や分析の重要な研究課題の一つである．対話文中には様々な話題（トピック）が頻出することから，対話要約においては，トピックのまとまり毎に対話文を分割するテキストセグメンテーションが重要であると考えられる．テキストセグメンテーションは，非構造な文書をトピックなどの意味的なまとまりを表すセグメント単位に分割する処理である．対話要約を行う前に，テキストセグメンテーションによって対話中の発話群がトピック単位に分割されていれば，対話文中のトピックを適切に捉えたより良い対話要約が可能になると考えられる．

テキストセグメンテーションの手法には，語彙的連鎖を利用した Lexical Cohesion segmentation (LCseg)[1] や，トピックモデルを用いた TopicTiling[2] が存在する．LCseg では，対話文中に出現する同一単語の出現情報（語彙的連鎖）を用いて分割を行うのに対して，TopicTiling ではトピックモデルの代表である Latent Dirichlet Allocation (LDA) を用いて分割を行っている．この 2 つの手法は語彙的連鎖や LDA といったそれぞれ異なる情報を用いているため，この 2 つの情報を統合して利用することで適切な分割を行うことができると考えられる．

そこで本論文では，語彙的連鎖を用いた LCseg とトピックモデルに基づく TopicTiling を利用した分割手法を提案する．複数人対話コーパスを対象として評価実験を行い，その有効性を検証する．また，対話文の特徴を考慮したルールを適用することで，その有効性を検証する．

2 先行研究

テキストセグメンテーションは，表層的な手がかり情報を用いて各文間のセグメント境界の結束性を計算することで，セグメント境界の検出を行なう．

代表的なテキストセグメンテーションアルゴリズムの 1 つに，TextTiling[3] がある．TextTiling のアルゴリズムは，主に 2 つに分けられる．

1. 文書中のある文間を基準点としてその左右に同じ窓幅の分析窓を設け，窓間の類似度 (結束度) を計算する．類似度の計算には，単語の出現頻度などのベクトルを用いたコサイン類似度で計算される．式 (1) に 2 つの分析窓 L，R におけるコサイン類似度 $cosine(L,R)$ を示す．

$$cosine(L,R) = \frac{\sum_t w_{t,L} \cdot w_{t,R}}{\sqrt{\sum_t w_{t,L}^2 \sum_t w_{t,R}^2}} \quad (1)$$

$w_{t,L}$ と $w_{t,R}$ は，左窓 L，右窓 R における単語 t の出現頻度である．

2. 基準点を 1 文ずつ動かしながら類似度の変化に着目し，グラフにおける類似度の極小点をセグメントの境界と推定しセグメンテーションを行なう．

類似度の計算には，分析窓内に出現した単語の情報を用いて計算する．しかし，対話文など 1 文が短い文書を対象とした場合では，窓内に出現する単語が比較的少ないため上手く類似度の計算ができない．そのため，このような文書に対しては TextTiling が有効に機能しないことが知られている．この問題点を解決するために，類似度の計算に単語の出現情報を用いる代わりに，語彙的連鎖を用いた LCseg やトピックモデルを用いた TopicTiling が存在する．

2.1 LCseg

LCseg は，文書の意味的に関連の深い部分には同一の語が繰り返し出現するという語彙的連鎖を利用する．LCseg では，語彙的連鎖を構成する候補語の決定を行なうために，ストップワードなど重要でない単語を除去する．そして，文書中の決定した候補語に対して，その候補語の出現開始の文から出現終了までの文の範囲を，連鎖として生成する．このとき，連鎖を形成する単語が極端に少ない場合は連鎖としないような処理や，文書中に長い間その候補語が存在しない状態が存在する場合には，そこで連鎖を切り離して別の連鎖として扱うような処理が行なわれる．

生成された連鎖に対して，文書中のトピック構造を考慮するようにスコアリングを行なう．ある候補語 t_i から構成される連鎖 R_i のスコア $score(R_i)$ は，連鎖に含まれる単語数を $freq(t_i)$，連鎖の長さ L_i，文書全体の長さを L としたとき，式 (2) のように表される．

$$score(R_i) = freq(t_i) \cdot log(\frac{L}{L_i}) \quad (2)$$

そして，計算した連鎖スコアを用いて，発話間の語彙的結束性スコアを計算する．語彙的結束性スコアの計算

は，TextTiling の考えに非常に類似しており，2 つの隣接した固定窓幅 k の分析窓を用いて計算する．LCseg では，2 つの分析窓でオーバラップする語彙的連鎖のスコアを用いて発話間の語彙的結束性スコアを計算する．式 (3) に，2 つの分析窓 L，R における語彙的結束性スコア $cosine_{LCseg}(L,R)$ を示す．

$$cosine_{LCseg}(L,R) = \frac{\sum_i w_{i,L} \cdot w_{i,R}}{\sqrt{\sum_i w_{i,L}^2 \sum_i w_{i,R}^2}} \quad (3)$$

$where$

$$w_{i,\Gamma} = \begin{cases} score(R_i) & if\ R_i\ overlaps\ \Gamma \in \{L,R\} \\ 0 & otherwise \end{cases}$$

2.2 TopicTiling

TopicTiling は，トピックモデルの LDA を利用している．LDA は，文書が潜在的な複数のトピックを持つことを仮定して，文書内の各単語に対して単語の潜在的トピックが推定される．TopicTiling では，まず LDA にトピック数 N を与え，各単語毎に各トピックに対する確率分布を計算し，最終的に文毎の各トピックに対する確率分布（トピックベクトル）を計算する．LDA によって得られたトピックベクトルを用いて，TextTiling や LCseg と同様に文間の結束度（コサイン類似度）を計算する．式 (4) に，2 つの分析窓 L，R におけるコサイン類似度 $cosine_{topic}(L,R)$ を示す．

$$cosine_{topic}(L,R) = \frac{\sum_n w_{n,L} \cdot w_{n,R}}{\sqrt{\sum_n w_{n,L}^2 \sum_n w_{n,R}^2}} \quad (4)$$

$w_{n,L}$ と $w_{n,R}$ は，左窓 L，右窓 R におけるトピック t の確率分布の総和である．

3 提案手法

本論文では，語彙的連鎖とトピックモデルを統合した分割手法を提案する．また，それぞれの分割手法に対話文の特徴を考慮したルールを適用することで，その有効性を検証する．

3.1 LCseg と TopicTiling を統合した分割手法

本研究では，LCseg と TopicTiling によって得られた文間の結束度を利用することで，語彙的連鎖と LDA の 2 つの情報を考慮した分割手法を提案する．具体的には，式 (3) と式 (4) でそれぞれ求めた文間の結束度を，sum_ratio の比率で足し合わせ，それを新たな文間の結束度として用いる．式 (5) に，2 つの分析窓 L，R における新たなコサイン類似度 $cosine_{merge}(L,R)$ を示す．

$$\begin{aligned} cosine_{merge}(L,R) = {} & sum\_ratio \times cosine_{LCseg}(L,R) \\ & + (1 - sum\_ratio) \times cosine_{topic}(L,R) \end{aligned} \quad (5)$$

この式 (5) では，sum_ratio が 1 に近づくほど LCseg を重視し，sum_ratio が 0 に近づくほど TopicTiling を重視した式となっている．

3.2 対話文の特徴を考慮したルールの追加

対話文の特徴を考慮したルールとして，R_1，R_2，R_3 の 3 種類のルールを適用する．

R_1：相槌など短い発話に関するルール
対話文中には「うん」や「なるほど」といった多くの相槌が存在する．相槌などの短い文が出現すると窓内の類似度が下がり，それらがセグメントの先頭になる傾向がある．相槌は直前の発話に対して発言しており，直前の発話と同じトピックであると考えられるため，セグメントの先頭は不適切だと考えられる．そこで，形態素解析を利用し形態素を 1 つしかもたない短い発話の前ではセグメントを行わないルールを R_1 とする．

R_2：話者系列に関するルール
複数人対話では，リーダー的話者によって対話の議事進行がされ，話の流れ（トピック）が切り替わると考えられる．また，繰り返し出現する特定の話者の系列は発話の内容と強い結びつきがあり，そのような話者系列ではトピックは変化しないと考えられる．そこで，特に繰り返し出現する話者系列中においてセグメントを行わないルールを R_2 とする．繰り返し出現する話者系列の抽出には PrefixSpan[4] を用い，実装には系列パターンマイニングツール PrefixSpan-rel[1] を利用した．対話文中において出現頻度が 10 回以上，系列の長さが 3 話者以上の話者系列を特に繰り返し出現する話者系列と設定する．

R_3：話者の切り替わりに関するルール
特定の話者が集中的に発話する状況からその他の話者が集中的に発話する状況の切り替わりは，話の内容の切り替わりを意味し，トピックの切り替わりと関係があると考えられる．そこで，極端に話者が切り替わるような話者の切り替わりスコアを計算し，それぞれの分割手法によって求めた文間の結束度 $cosine$ に話者の切り替わりスコアを足すルールを R_3 とする．具体的には，話者の切り替わりスコア $Score_{speaker}$ は JS ダイバージェンスを用いて計算し，パラメータ $\alpha_{speaker}$ を用いて，式 (6) で表されるようなルール適用後の結束度 $cosine'$ を計算する．

$$cosine' = cosine + \alpha_{speaker} \times Score_{speaker} \quad (6)$$

4 評価実験

4.1 実験データ

評価実験は，本研究室が開発した複数人対話コーパス (Kyutech コーパス [5]) を対象として行った．Kyutech コーパスは，4 名一組によるグループディスカッションを対象としており，現在までに 9 対話収録している．具体的な議論内容は，架空の商業施設内のレストラン街に新規参入する店舗を，資料に掲載されている施設情報や地域情報を踏まえて 3 店舗の選択肢の内から決定するものとなっている．

Kyutech コーパスには，各発話に対して「何の話題（トピック）について言及されているか」を示すトピックタグの付与（トピックアノテーション）が行われており，評価実験はこのトピックタグを利用する．Kyutech コーパスの対話データでは，施設情報や店舗に関する話題といったようなトピックが対話中に存在すると考えられ，合計 28 個のトピックタグが用いられている．また，1 つの発話文が複数のトピックを持つ場合もあるため，トピック

[1] http://prefixspan-rel.osdn.jp/

表 1: Kyutech コーパスの対話データ

対話データ	発話数	正解セグメント数
対話 1	505	52
対話 2	637	77
対話 3	324	33
対話 4	504	36
対話 5	566	49
対話 6	487	50
対話 7	284	31
対話 8	445	42
対話 9（開発用）	759	48

表 2: 完全一致と前後許容の F 値

手法	完全一致	前後許容
LCseg	**0.193**	0.405
TopicTiling(10)	0.118	0.374
TopicTiling(20)	0.121	0.366
TopicTiling(30)	0.125	0.326
Merge(10,0.3)	0.126	0.363
Merge(10,0.7)	0.179	**0.412**
Merge(20,0.3)	0.149	0.374
Merge(20,0.7)	0.187	0.394
Merge(30,0.3)	0.152	0.368
Merge(30,0.7)	0.185	0.379

アノテーションは各発話に対して必ず付与される必須タグと，2 つまで追加で付与できる追加タグのアノテーションが行われている．本研究では，必須タグが切り替わる文間の境界をセグメンテーションの正解の境界とし，評価実験を行った．1 対話を開発データとし，残りの 8 対話を実験データとした．Kyutech コーパスの対話データの詳細を表 1 に示す．

4.2 評価基準

提案手法におけるパラメータは，$sum\_ratio = 0.3, 0.7$ の 2 種類について実験を行い，$\alpha_{speaker} = 0.5$ と設定した．また，分析窓幅などの各種パラメータは，文献 [1] でチューニングされたパラメータを用いた．本論文では，以下の 2 種類の評価基準を用いてテキストセグメンテーションの評価を行った．

1. **分割位置の一致に関する精度と再現率および F 値.** 対話データ D_r に対して，トピックタグが切り替わる直前の文番号の集合を B_r とする．同様に，同じ対話データを分割アルゴリズムを用いて分割したもの（仮説データ）D_h の分割直前の文番号の集合を B_h とする．B_r，B_h の一致について，精度，再現率，F 値を計算し，これを完全一致の結果とする．つまり，精度 p は，$p = |(B_r \cap B_h)|/|B_h|$，再現率 r は，$r = |(B_r \cap B_h)|/|B_r|$ で計算され，F 値はこれらの調和平均 $F = 2/(1/p + 1/r)$ である．また，$B'_r = \bigcup_{i \in B_r} i-1, i, i+1$，および $B'_h = \bigcup_{i \in B_h} i-1, i, i+1$ とし，精度 $p' = |(B'_r \cap B_h)|/|B_h|$，再現率 $r' = |(B_r \cap B'_h)|/|B_r|$，および F 値 $F' = 2/(1/p' + 1/r')$ を前後許容による結果と呼ぶ．

2. **テキスト中の 2 つの文に対する誤分類（2 文評価）.** 2 文評価 [6][7] は，異なる 2 文が正しいセグメントに分類されているかを評価する指標である．対話データ D_r に付与されている，1 つの必須タグのまとまりを段落とし，同じ対話データを分割アルゴリズムで分割した仮説データを D_h と表し，ともに長さが n 文であるとする．2 文評価は，D_r における i 番目と j 番目の文章 r_i，r_j と，D_h における i 番目と j 番目の文章 h_i，h_j について，段落への分割が一致しているか否かを測る尺度であり，式 (3) で表される $P_D(D_r, D_h)$ を求める．

$$P_D(D_r, D_h) = \sum_{1 \le i \le j \le n} D(i,j)(\delta_r(i,j) \overline{\oplus} \delta_h(i,j)) \quad (7)$$

ここで，$\delta_r(i,j)$ は，D_r において，r_i と r_j が同一の段落に含まれていれば 1，そうでなければ 0 をとる関数であり，D_h において，h_i と h_j が同一の段落に含まれていれば 1，そうでなければ 0 をとる関数である．また，$overline\oplus$ は排他的論理和の否定であるため，$\delta_r(i,j)$ と $\delta_h(i,j)$ が同一の値ときのみ 1 となり，それ以外は 0 となる．関数 $D(i,j)$ は，i 番目の文と j 番目の文の位置に対して考慮する関数であり，i と j が遠く離れている場合は低い値をとり，近い場合は高い値を返す．本論文では，以下の関数 $D_k(i,j)$ を用いた．対話データの全文数 n とすると，定数 $k(< l)$ に対して，関数 $D_k(i,j)$ は式 (8) のようになる．

$$D_k(i,j) = \begin{cases} 1/(lk - \frac{k(k-1)}{2}) & |i-j| < k \\ 0 & otherwise \end{cases} \quad (8)$$

本研究では，$k = 2, 4, 6, 8, 16$ の 5 種類について実験を行う．

4.3 分割位置の一致に関する精度評価

先行研究と提案手法の分割手法を比較して，分割位置の一致に関する精度評価実験を行った．表 2 に完全一致，前後許容の F 値を示し，各手法で最も高精度であった結果を太字で示す．

表 2 において，TopicTiling のトピック数 n を $\alpha \in \{10, 20, 30\}$ としたときの結果を TopicTiling(α) と表し，提案手法のトピック数 n を α，足し合わせ比率 sum_ratio を $\beta \in \{0.3, 0.7\}$ としたときの結果を Merge(α, β) と表している．表 2 の前後許容の尺度において，Merge(10,0.7) のみが従来手法の LCseg の F 値平均を上回ったが，それぞれの平均値間に有意な差があるか対応のある両側検定の t 検定を行ったところ，有意水準 5 %で有意差は確認されなかった (t=2.36，df=7，0.1<p)．TopicTiling の F 値は他の手法と比べて低く，提案手法において sum_ratio の値を 1 に近づけるほど精度が高いことから，TopicTiling が有効に機能していないことが確認できる．一方で LCseg 単体で分割を行なうよりも提案手法が上回っている結果が存在することから，TopicTiling によってより良い分割結果を得ることができれば，提案手法が有効に機能すると考えられる．

表 3: 2 文評価の結果

手法	K = 2	k = 4	k = 6	k = 8	k = 16
LCseg	0.907	0.782	0.709	0.668	0.646
TopicTiling(10)	0.861	0.681	0.605	0.577	0.616
TopicTiling(20)	0.876	0.714	0.638	0.606	0.629
TopicTiling(30)	0.889	0.734	0.655	0.617	0.624
Merge(10,0.3)	0.871	0.698	0.619	0.591	0.629
Merge(10,0.7)	0.903	0.771	0.700	0.663	**0.654**
Merge(20,0.3)	0.888	0.737	0.661	0.626	0.643
Merge(20,0.7)	0.910	0.785	0.710	0.668	0.648
Merge(30,0.3)	0.898	0.756	0.679	0.641	0.645
Merge(30,0.7)	**0.912**	**0.787**	**0.712**	**0.669**	0.644

表 4: ルール追加による完全一致の F 値

手法	BASELINE	B+R_1	B+R_2	B+R_3
LCseg	0.193	0.186	0.144	0.192
TopicTiling(10)	0.118	**0.124**	0.113	0.117
TopicTiling(20)	0.121	**0.127**	0.110	0.120
TopicTiling(30)	0.125	0.125	0.098	0.119
Merge(10,0.3)	0.126	**0.132**	0.115	**0.127**
Merge(10,0.7)	0.179	0.179	0.143	**0.180**
Merge(20,0.3)	0.149	**0.158**	0.121	0.149
Merge(20,0.7)	0.187	0.177	0.144	0.184
Merge(30,0.3)	0.152	0.152	0.108	0.141
Merge(30,0.7)	0.185	0.178	0.135	**0.188**

表 5: ルール追加による前後許容の F 値

手法	BASELINE	B+R_1	B+R_2	B+R_3
LCseg	0.405	0.370	0.305	0.405
TopicTiling(10)	0.374	0.349	0.328	0.373
TopicTiling(20)	0.366	0.358	0.324	**0.371**
TopicTiling(30)	0.326	0.289	0.266	**0.328**
Merge(10,0.3)	0.363	0.341	0.313	**0.366**
Merge(10,0.7)	0.412	0.377	0.320	**0.414**
Merge(20,0.3)	0.374	0.364	0.295	**0.378**
Merge(20,0.7)	0.394	0.362	0.295	**0.398**
Merge(30,0.3)	0.368	0.333	0.276	0.353
Merge(30,0.7)	0.379	0.337	0.268	**0.392**

4.4 2 文評価による精度評価

先行研究と提案手法の分割手法を比較して，2 文評価に関する評価実験を行った．表 3 に，2 文評価による 8 対話の平均値を示す．また，各手法で最も高精度であった結果を太字で表わす．

k の値が大きいほど，離れた文同士が正しい段落に分類されているかどうかを考慮するため，精度は低い値となる．表 3 より，$k = 2, 4, 6, 8$ のときに Merge(30,0.7) が最も高く，$k = 16$ のときのみ Merge(10,0.7) が最も高い結果となった．それぞれの k における提案手法が最も高くなった結果において，LCseg との有意差は確認されなかったが (t=2.36，df=7，0.1<p)，k がどの値の場合でも提案手法が上回っていたことから，一定の有効性があると考えられる．

4.5 ルール追加による精度評価

3.2 節で説明した対話文の特徴を考慮した 3 種類のルールを適用することで，精度が向上するかを検証した．ルールを適用しない場合の結果を BASELINE とし，その BASELINE に 3 種類のルールを適用したものを B+R_1，B+R_2，B+R_3 とする．表 4 に完全一致の F 値を，表 5 に前後許容の F 値の結果を示す．また，BASELINE よりも F 値が上回っていた結果を太字で示す．

ルール R_1 を適用した場合では，完全一致の結果では BASELINE を上回るものが存在したが，前後許容の結果では存在しなかった．R_1 は相槌などの短い発話の直前で分割しないようにするルールであり，無意味なセグメントを生成しないことで，相対的な精度が向上したためである．一方で，このルールの適用によって，無意味なセグメントでも前後許容によって正解と判断されていた事例がなくなったことで，結果として精度が低下した．また，ルール R_3 では完全一致と前後許容の結果どちらでも BASELINE を上回るものが存在し，Merge(10,0.7) の F 値 0.414 は前後許容の結果において最も高い値となった．今回の実験では，ルール R_2 は有効に機能しないことが確認された．

5 おわりに

本論文では，LCseg と TopicTiling に基づいたテキストセグメンテーション手法を提案し，対話文の特徴を考慮したルールの追加による有効性の検証を行った．提案手法の分割手法は，LCseg との有意差は確認できなかったが，前後許容の結果や 2 文評価では最も良い結果となった．また，対話文の特徴を考慮したルールを適用することで一部精度が上昇することが確認された．

今後の課題として，分析窓などのパラメータのチューニングや TopicTiling の効果的な利用が考えられる．

謝辞

本研究は科研費 26730176 の助成を受けたものです．

参考文献

[1] M. Galley, K. McKeown, E. Fosler-Lussier, and H. Jing, “Discourse segmentation of multi-party conversation”, Proceedings of the 41st ACL 2003, 2003.

[2] M. Riedl and C. Biemann, “TopicTiling: A Text Segmentation Algorithm based on LDA”, Proceedings of the 50th ACL 2012, 2012.

[3] Marti A. Hearst, “Multi-paragraph segmentation of expository text”, Proceedings of the 32nd ACL 1994, 1994.

[4] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal and M-C. Hsu, “PrefixSpan: Mining Sequential Patterns Efficiently by PrefixProjected Pattern Growth”, Proceedings of ICDE’01, pp. 215-224, 2001.

[5] 嶋田 和孝, 山村 崇, 河原真太郎, Giuseppe Carenini, Raymond T. Ng, “Kyutech コーパス：意思決定タスクを対象とした複数人対話コーパス”, 言語処理学会第 22 回年次大会 (NLP2016), 2016.

[6] B. Doug, B. Adam, and L. John, “Statistical Models for Text Segmentation”, Machine Learning, Vol.34, Nos.1-3, pp.177-210, 1999.

[7] 但馬康宏, “言語モデルの違いによる HMM を用いたテキストセグメンテーションの性能比較”, 情報処理学会論文誌, pp. 38-46, 2013.