関連記事判定のためのニュース記事キーフレーズ抽出

大倉 俊平　小野 真吾
ヤフー株式会社
{sokura, shiono}@yahoo-corp.jp

1　導入

ニュース記事には，以前に報道された事柄を前提とする続報記事が多く存在する．ある記事に対して，前提となる記事やその話題の最新記事を容易に探せることは，情報摂取の効率を大きく向上させる．

しかし，記事 A が記事 B の続報記事であるか否かを判定できる判定器を，学習データを用意して直接学習するのは容易ではない．なぜなら，多岐にわたる内容の記事に対して学習データを用意することが困難であるのに加え，学習時に存在しなかった新規の話題が次々と現れ，判定器がすぐに劣化してしまうことが想定されるからである．

そこで，学習を必要としない教師なしのキーフレーズ抽出法を用いて記事の主題を抽出し，そのフレーズを関連記事抽出に応用すること考える．

キーフレーズには，文書の内容を端的に表現することと，文書の検索を容易にすることの 2 つ役割がある．関連記事抽出に利用する際には，後者の役割が重要である．特に，現在着目している記事があり，その関連記事を検索するという文脈では，関連記事が同一のキーフレーズを共有しているという性質が重要となる．この性質をキーフレーズの共有性と呼ぶことにする．

従来のキーフレーズ抽出手法は，著者などが設定したキーフレーズの再現性で評価されることが多い [3] が，著者らが付与するキーフレーズは必ずしもこの性質を重要視していない．特に，内容を端的に表現することを重視してキーフレーズをつけた場合，関連する文書同士でも選ばれるキーフレーズが異なる場合が多く，本稿の目的にはそぐわない．

本稿では，関連記事の効率的な抽出にキーフレーズを応用することを前提に，キーフレーズ共有性に着目した，明示的な正解を用いないキーフレーズ抽出法を評価尺度を提案する．また，提案する評価尺度によると従来の抽出法には課題があることを示し，それを改善する新しい抽出法を提案する．

2　キーフレーズ共有性の評価

関連記事の特定には，関連記事間でキーフレーズが共有されていることが重要なことを導入で述べた．キーフレーズの共有性を定量化し，既存のキーフレーズ抽出手法が本稿の目的に的するかどうかを考察する．

2.1　評価データ

評価用の記事セットとして，Yahoo!トピックスのデータを用意した．本データは Yahoo!ニュースに入稿される記事から，編集者が毎日 100 件前後抽出して作成されたものである．特に同じ事件や出来事について，時間の経過に伴って複数の記事が抽出された場合には，それらは一つのトピックとして同じタグが付与されている．2017 年 10 月から 2017 年 12 月までの記事のうち，2 記事以上を含む 66 トピック 1716 記事を以降の評価で用いる．

あるキーフレーズ抽出法で各記事にそれぞれキーフレーズを付与した時，同一トピックの記事が同じキーフレーズを共有していれば，その抽出法は関連記事抽出に使いやすいフレーズを出力できていると言える．

トピックの例としては，「トランプ大統領来日」や「台風 21 号」などがあり，最大のトピックは「衆議院選挙」で 241 記事を含む．平均文章長は 425 単語であった．

2.2　評価指標の定義

D を前述した評価記事の集合，C をトピックの集合，$L : D \to C$ をトピック割付の関数とする．キーフレーズ抽出システム S は，各記事 $d \in D$ に対して，キーフレーズの集合 $S(d) = \{w_{d,1}, \cdots, w_{d,K}\} \subset W_S$ を割り当てるものとする．ここで W_S はキーフレーズとして選ばれる可能性のあるワード集合で，$w_{d,k}$ が実際に d のキーフレーズとして抽出されたワードである．キーフレーズ共有性の観点では，キーフレーズ $w \in S(d)$

は，以下を満たしていることが望ましい．

$$L(d) = L(d') \quad \Rightarrow \quad w \in S(d') \tag{1}$$
$$L(d) \neq L(d') \quad \Rightarrow \quad w \notin S(d') \tag{2}$$

そこで，各 w に対して，以下の評価値を考える．

$$\mathrm{P}(d, w; S) = \frac{|\{d' \in D \setminus \{d\} | w \in S(d') \wedge L(d) = L(d')\}|}{\max(|\{d' \in D \setminus \{d\} | w \in S(d')\}|, 1)}$$
$$\mathrm{R}(d, w; S) = \frac{|\{d' \in D \setminus \{d\} | w \in S(d') \wedge L(d) = L(d')\}|}{|\{d' \in D \setminus \{d\} | L(d) = L(d')\}|}$$
$$\mathrm{F}(d, w; S) = \frac{2\mathrm{P}(d, w; S)\mathrm{R}(d, w; S)}{\mathrm{P}(d, w; S) + \mathrm{R}(d, w; S)}$$

これは，「w をキーフレーズとしてもつか否か」を「$L(d)$ と同一トピックの記事であるか否か」の 2 値分類器であるとみなしたときの，Precision, Recall, F 値にそれぞれ対応する．w がトピック $L(d)$ の記事においてのみキーフレーズとして抽出される (i.e. (2) を満たす) 場合に $\mathrm{P} = 1.0$ となり，w がトピック $L(d)$ の記事すべてにおいてキーフレーズとして抽出される (i.e. (1) を満たす) 場合に $\mathrm{R} = 1.0$ となる．

これを元に，キーフレーズ抽出システム S の評価値を以下のように定める．

$$\mathrm{F}(S) = \frac{1}{|C|} \sum_{c \in C} \frac{1}{|D_c|} \sum_{d \in D_c} \frac{1}{|L(d)|} \sum_{w \in L(d)} \mathrm{F}(d, w; S)$$

D_c はトピック c に属する D の部分集合を表す．また，$\mathrm{P}(S), \mathrm{R}(S)$ についても同様に定める．すなわち，全ての記事に自身の抽出法でキーフレーズを付与したときに，同一トピック内の記事が多くの共有キーフレーズを持ち，それを他トピック間では共有しないとき，この評価値は高くなる．

この評価法の利点の一つは，候補キーフレーズの集合 W_S が異なるシステム同士も比較できることである．例えば，あるシステムはキーフレーズとして単語を出力し，別のシステムは複数単語が連結されたフレーズを出力したとしても，同じ評価セット D を用いて評価可能である．また，この評価法は 2.1 節のデータに限らず，事前にクラスタリング済の記事集合であれば同様に計算できる．正解キーフレーズを用いて再現性を測る評価法では，評価セットがフレーズであれば，評価対象のシステムもフレーズを出力するように設計されていなければ正当に評価できない．

2.3 既存のフレーズ抽出法の評価実験

前章で提案した評価手法の傾向をみるために，既存の 6 つの教師なしキーフレーズ抽出法について評価を行なった．

2.3.1 評価対象手法

TF-All 全ての品詞を含む全単語から，記事中の出現頻度が高い順にキーフレーズとする．当然，助詞などの意味を持たない語が選ばれるが，このような方法で評価値が高くならないことを確認するために採用する．

TF-Noun TF-All で品詞を名詞だけに制限する．

TF-IDF-Noun 各名詞の出現頻度に逆記事頻度 $\mathrm{IDF}(w) = \log |D|/|\{w \in d | d \in D\}|$ をかけ，値が高い順にキーフレーズとする．

TF-IDF-Phrase キーフレーズ候補を連続する名詞からなるフレーズとする．フレーズを構成する各名詞毎に TF-IDF を計算しその和をスコアとする．スコアの高い順にキーフレーズとする．

PosRank グラフベースの教師なしキーフレーズ抽出法である PositionPank[1] を用いたもの．キーフレーズ候補は「[形容詞]*[名詞]+」の形をしたフレーズである．ウィンドウサイズは 5 とした．

PosRank-Noun PosRank において，キーフレーズ候補を名詞 1 単語のみとしたもの．

2.3.2 評価結果

それぞれの手法に対し，抽出キーフレーズ数 K を 5,10 の 2 通りで評価した結果は表 1 のようになった．TF-All は Recall は非常に高くなるものの，Precision が極端に低いため，F 値も他手法に比べて低くなっており想定通りの結果である．これは，全記事を通して一貫して出現頻度の高いストップワードをキーフレーズとしてしまっているためである．

それ以外の手法では，Precision はある程度高くなるものの，Recall が低い．特に，複数単語をフレーズとして出力する TF-IDF-Phrase と PosRank は単語を出力する手法に比べて Precision も低くなる．フレーズ単位だと表現が多様化し，その記事にしか出現しない固有の表現をキーフレーズとしてしまうケースが多く，この評価法では，$P = R = 0$ と評価されるためである．そのようなフレーズは，記事を特徴付けるものではあるが関連記事検索のようなタスクにおいては重要度が低い．Precision を維持しつつ Recall を上げるためには，記事内容の特徴を捉えつつも，同一トピック内では汎用的に使用されるフレーズを選ばなくてはならない．

	$K=5$			$K=10$		
手法	P	R	**F**	P	R	**F**
TF-All	.02	.60	**.03**	.04	.69	**.04**
TF-Noun	.35	.21	**.20**	.32	.22	**.20**
TF-IDF-Noun	.39	.17	**.20**	.35	.18	**.19**
TF-IDF-Phrase	.30	.08	**.11**	.33	.10	**.13**
PosRank	.34	.11	**.15**	.34	.12	**.15**
PosRank-Noun	.35	.19	**.21**	.32	.20	**.20**

表 1: 既存手法の評価結果

3 提案抽出法

3.1 類似記事を参照する方法

Recall が低くなる問題を解決するためには，関連記事を直接参照して候補となっているフレーズが他の記事でも使用されているかを見るのが単純な方法である．ただし，キーフレーズ抽出時点では，評価時に用いるトピック割付 L は未知であるから，別の方法で記事間類似度を求め，類似度上位の記事を仮の関連記事として参照する．以下のように，Neighborhood Frequency (NF) を定める．

$$\mathrm{NF}_{D,M}(d,w) = \mathbf{1}_{w\in d} \sum_{d'\in N_{D,M}(d)} \mathbf{1}_{w\in d'}$$

ここで，$N_{D,M}(d)$ は D 中で d との類似度が上位の M 件の記事群を表す．$\mathbf{1}_{w\in d}$ は，フレーズ w が記事 d 中に 1 度でも出現するとき 1 で，そうでなければ 0 を表す．フレーズの候補は，PositionRank における候補に合わせ「[形容詞]*[名詞]+」の形とする．

これにフレーズ単位の IDF を掛け合わせ，$\mathrm{NF}_{D,M}(d,w)\mathrm{IDF}(w)$ が高いものをキーフレーズとする．つまり，TF-IDF において，記事内の出現頻度 (TF) に変えて，類似記事での出現記事数を用いることに相当する．この変更によって，著者特有の表現などその記事のみにおいて使用される表現がフレーズとして抽出されることを避けることができる．

一方で，1 記事のキーフレーズを抽出するために D 全体の情報が必要になるため，フレーズ抽出時のコストが非常に高いという欠点がある．

3.2 RNN を用いて推定する方法

前節の手法の欠点である，フレーズ抽出時に D 全体が必要になる点を解消するために D を用いてモデルを構成し，抽出時には，対象記事 d のみを用いて NF-IDF を用いた抽出を再現することを試みた．具体的には，以下の手順で抽出を行う．

1. 学習用記事群 D に対して，NF-IDF を用いて記事毎に各 10 フレーズを抽出する．
2. それぞれの記事でフレーズとして抽出された箇所に，固有表現抽出の学習で用いられる BIESO タグ [5] を用いてタギングを行う．
3. 2 で得られたタグを学習データとし，入力単語列からタグ列を推定するモデルを学習する．
4. フレーズ抽出時には，学習されたモデルに文章を入力し，キーフレーズとしてタギングされる確率が高い箇所から順にキーフレーズとして出力する．

系列タギングの問題に帰着することによって，フレーズの構成要素以外にも，周辺の文脈や文章中の出現位置などの情報を使って汎化が進むことを狙っている．

4 提案手法の評価

4.1 実験設定

NF-IDF を計算するための記事群 D として，評価用の記事とは別のニュース記事 50 万件を用意した．これらにはトピックを表すタグは特に付与されておらず，タイトルと本文のみからなるデータである．NF-IDF における類似度算出には [6] を用いて，参照する記事数 M は 10 とした．タグ列推定モデルは [4] を参考に，Bidirectional GRU 2 層と CRF からなるニューラルネットワークのモデルを構築し，Adam を用いた SGD によって学習した．入力は，頻出 20 万単語に絞り，それ以外は未知語を表す 1 語で置き換えた．

4.2 結果

提案法を用いて，2.1 節の評価データにキーフレーズを付与し同様の評価を行なった結果を表 2 に示す．

NF-IDF を用いた場合，既存手法に対して Recall だけでなく Precision も大きく改善した．これは，強い特徴となるフレーズを抽出できたというよりも，その記事にしか出現しないフレーズを回避することによって，$P=0$ となるケースが減ったことが大きく貢献している．BiGRU-CRF を用いた場合，学習元の NF-IDF と同等程度の性能を期待していたが，大きく上回る結果となった．これは，NF-IDF の場合，参照する記事 $N_{D,M}(d)$ として何が選ばれるかに寄るところが大きく結果が不安定であったが，フレーズの周辺情報によ

	$K=5$			$K=10$		
手法	P	R	**F**	P	R	**F**
TF-Noun	.35	.21	**.20**	.32	.22	**.20**
PosRank-Noun	.35	.19	**.21**	.32	.20	**.20**
NF-IDF	.61	.24	**.30**	.61	.28	**.34**
BiGRU-CRF	.62	.40	**.42**	.56	.36	**.38**

表 2: 提案手法の評価結果 (再掲含む)

る汎化と，CRF の確率モデルの部分がその不安定さを吸収できたためではないかと考えられる．

実際の抽出例として，上野動物園のパンダ公開に関する記事から抽出されたキーフレーズを表 3 に示す．提案法では，新しく公開されるパンダの名前「シャンシャン」に加えて,「上野動物園」や，以前から公開されている母親の名前「シンシン」を共有キーフレーズとして抽出できている．従来法では，各記事の内容を表す語を抽出できているが，記事間でフレーズを共有できていないことがわかる．

5 関連研究

この章では提案手法と関連する研究について述べる．本稿で扱ったキーフレーズ抽出というタスクは，取り組み方にいくつかの方針がある [3]．教師あり手法か否か、外部知識を利用するか否かなど観点は複数あるが，キーフレーズ付与対象が単一文書であるか [1]，文書群であるか [2] という点でも分類ができる．

単一文書に付与する場合は文書を特徴付けるフレーズが重視されるため，DF(Document Frequency) が低い語がフレーズとして好まれる傾向がある．一方で，文書群に付与する場合は，文書群全体を代表するフレーズを抽出するため，対象文書群内での DF が高い語を好む傾向がある．本稿は，付与対象は単一文書であるが動機はトピックでまとまった文書群にフレーズを付与する場合に近い．そのため，周辺文書のみに絞った DF が高いフレーズに着目した．すなわち，入力された単一文書に対して，その文書の周辺を文書群とみなしたときの代表フレーズを抽出していると言っても良い．

また，既存のキーフレーズ抽出の多くは，先に候補フレーズを抽出し，それらののスコアを求めてランキングする問題として定式化している．[1] は，このプロセスにおいて文書中での出現位置の情報が失われることを指摘し，スコア付の際に出現位置を加味する方法を提案している．ニュース記事の場合，冒頭で 1 度出現した後，冗長性回避のため後段では繰り返されない重要フレーズが存在する場合あり，フレーズの出現位置は重要な情報である．本稿の系列タギングモデルを使った手法では，候補フレーズ抽出とスコア付けを同時に行うことで，出現位置をはじめとするコンテキスト情報が失われることを回避している．

手法	記事	抽出されたフレーズ ($K=3$)		
TF-Noun	#1	パンダ	舎	シャンシャン
	#2	パンダ	シャンシャン	知事
	#3	シャンシャン	申し込み	観覧者
PosRank-Noun	#1	シャンシャン	効果	パンダ
	#2	パンダ	和歌山	知事
	#3	シャンシャン	観覧	申し込み
BiGRU-CRF	#1	シャンシャン	シンシン	上野動物園
	#2	シャンシャン	上野動物園	パンダ
	#3	シャンシャン	シンシン	上野動物園

表 3: 抽出キーフレーズ例

6 まとめ

本稿では，関連記事検索のためにキーフレーズ共有性に注目して，キーフレーズ抽出の良さを評価する方法を提案した．その評価法で従来法を評価することで，従来法は対象記事固有の表現を抽出しやすく，結果的に，関連記事の検索のためには不向きなフレーズを抽出する場合があることを示した．それを元に，対象記事固有の表現を避ける抽出法を提案し，前述の評価法において大きく改善することを示した．

参考文献

[1] C. Florescu and C. Caragea. Positionrank: An unsupervised approach to keyphrase extraction from scholarly documents. In *Proceedings of the 55th Annual Meeting of the ACL*, volume 1, pages 1105–1115, 2017.

[2] K. M. Hammouda, D. N. Matute, and M. S. Kamel. Corephrase: Keyphrase extraction for document clustering. In *Proceedings of the 4th International Conference on MLDM*, pages 265–274, 2005.

[3] K. S. Hasan and V. Ng. Automatic keyphrase extraction: A survey of the state of the art. In *Proceedings of the 55th Annual Meeting of the ACL*, volume 1, pages 1262–1273, 2014.

[4] Z. Huang, W. Xu, and K. Yu. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*, 2015.

[5] T. Kudo and Y. Matsumoto. Chunking with support vector machines. In *Proceedings of the second meeting of the NAACL*, pages 1–8, 2001.

[6] S. Okura, Y. Tagami, and A. Tajima. Article deduplication using distributed representations. In *Proceedings of the 25th International Conference Companion on WWW*, pages 87–88, 2016.