7V-05

依存ネットワークを用いた見出し候補生成

古川拓実†　菱田隆彰†

愛知工業大学大学院経営情報科学研究科†

1. はじめに

近年，非常に多くのウェブ小説が創造されている．我々の調査では，2022年の2月から7月にかけて，毎月約 8000 の小説が小説家になろうに投稿されていることが明らかになった．今後も日々大量のウェブ小説が創作されていくと考えられ，読み手の注目の分散は避けられない．小説家が自分の作品に注目されるにはより凝った見出しにする必要がある．

見出し生成研究界隈では見出し生成について活発に研究されているものの，目を引くという観点の見出し生成はあまり研究されていない．西原ら[1]は興味を引く論文タイトルの作成支援を行うシステムを開発しているが，独自の評価式によって見出しを評価して順位づけするものであり，生成という観点のものではない．

我々は多様な見出し候補を多数生成し，ウェブ小説家に提示することでより適切な言い回しの選択や，見出し作成時間の効率化による見出し創作の支援がしたいと考え研究に着手した．

本稿では，小説家の見出し創作支援として，依存ネットワークを用いた大量の見出し候補生成を行う．既存研究にはない自由文を依存ネットワークに変換できるライブラリ baran を開発し，また，語の重複を除いた見出し候補を多数生成するアルゴリズムを提案する．

2. 依存ネットワークとは

依存ネットワークとは，中野ら[2]によって定義された，一文を語の依存関係によってネットワーク図として表す構造である．自立語はノード，依存関係はアークとして表される．例として，図1に『エリカが多くの村人を助けた』の依存ネットワークを示す．

主述語である『助ける』がルートとなっており，ガ格の依存関係として『エリカ』，ヲ格の依存関係として『村人』が確認できる．用言である『助けた』は原型となり，代わりに活用情報として『過去形』を保持する．また，各ノードが保持する文生成順序情報は文を生成する際に利用されるものである．

中野ら[2]はまた，依存ネットワークを用いた文の言い換え手法を提案している．言い換えは指定した語が末尾になるよう依存ネットワークを変換することで実現される．指定する語を持つノードはエントリノードと呼ばれ，当該ノードがシンクになるように依存関係の逆転と，関連する格の書き換え及び活用情報の変更を次々に行なっていくことで言い換えが行われる．本研究ではこの変換を末尾語指定変換と呼ぶ．

3. 依存ネットワークライブラリ baran

任意の文を依存ネットワークに変換するツールは現時点で存在しない．中野らは基本動詞辞書に記載されている文例と文形式情報から依存ネットワークを生成している[2]．また，彼らは EDR 日本語コーパスと EDR 日本語辞書から依存ネットワークを生成している[3]．

要約文から見出しを生成するためには，自由文を依存ネットワークに変換するライブラリが必要になる．本研究では cabocha を文の依存関係解析ツールとして，自由文を依存ネットワークに変換するライブラリ，baran を開発した．

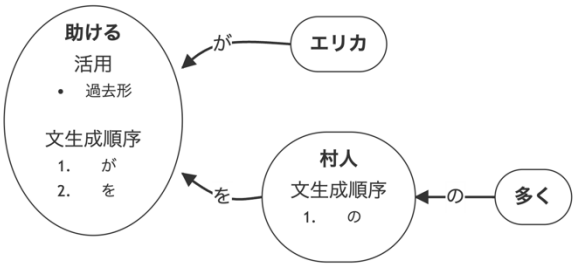

図 1:依存ネットワークの例

```
崇める () [用言ノード, 受身形, ている形] {}
  エリカ (が) [] {}
    助ける () [用言ノード, 過去形] {}
      村人 (を) [] {}
        多く (の) [] {}
  聖女 (として) [] {}
```

図 2:baran による自由文の変換例

Generating candidates of headline using dependency network
†Takumi Furukawa, Takaaki Hishida, Graduate School of Business Administration and Computer Science

Copyright ©2023 Information Processing Society of Japan.
All Rights Reserved.

baran で『多くの村人を助けたエリカが聖女として崇められている』を依存ネットワークに変換した例を図2に示す．各行はノードを表し，ノード間の依存関係はインデントの深さで表される．各種の括弧書きの情報は左から順に，依存先との関係を表す語，用言の情報，ニュアンスである．依存ネットワークへの変換が適切にできているのが確認できる．

次にbaranの末尾語指定変換機能による言い換えの実行例を示す．図2に示した依存ネットワークに対して，『助ける』ノードをエントリーノードとして末尾語指定変換すると，図3のように変換される．文に直すと『多くの村人を聖女として崇められているエリカが助けた』であり，言い換えができていることが確認できる．

4. 見出し候補生成アルゴリズム

見出し候補生成の入力は中括弧で囲まれた抽象フレーズと呼ばれる要素を持つテンプレートと，十分に要約された文の二つである．フレーズを生成し，抽象フレーズのタイプに適合するフレーズで抽象フレーズを置換することで見出しが生成される．生成の詳細な手順を次に示す．

1. テンプレートから抽象フレーズの列を取得
2. 要約文を依存ネットワークに変換
3. 各ノードをエントリノードに末尾語指定変換
4. 各変換結果からフレーズ列挙
5. 列挙したフレーズを抽象フレーズのタイプごとにグループ分け
6. 抽象フレーズのタイプに適合するフレーズで抽象フレーズを置換
7. 同一語フィルタにより冗長なフレーズ列を持つ見出し候補を除去

手順4により依存ネットワークの意味を破壊せずに多数のフレーズを生成できる．図4に手順6，手順7を示す．手順7を経ることで，一般的に好まれない冗長な見出し候補が生成されるのを防げる．また，手順7のアルゴリズムを図5に示す．各ノードにビットフラグを割り当てることで冗長の検出を行なっている．

テンプレートに『{名詞句}、{動詞句}！？』，要約文に『多くの村人を助けたエリカが聖女として崇められる』を指定した場合の出力例としては，a.多くの村人を助けたエリカ、聖女として崇められる！？，b．聖女として崇められるエリカ、多くの村人を助けた！？，c.エリカ、多くの村人を助けた！？などが得られた．

本稿の見出し候補生成アルゴリズムにより，a,b,c のような意味の通る見出しを複数確認した．フレーズの単位では入力した要約文の意味を適切に表現していた．本稿で目的にしていた多数の見出し候補生成は達成できたといえる．一部の見出し候補は入力文の意味から乖離した内容であった．これはテンプレートにフレーズを適用した際に元文での関係性が維持されなかったためであると考えられる．

```
助ける()[用言ノード,過去形]{}
  エリカ(が)[]{}
    崇める()[用言ノード,受身形,ている形]{}
      聖女(として)[]{}
  村人(を)[]{}
    多く(の)[]{}
```

図 3:『助ける』で末尾語指定変換した結果

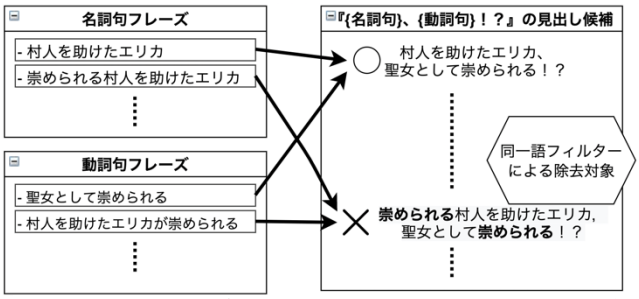

図 4:見出し候補の生成・同一語フィルタの対象

```
1. 関数 同一語フィルタ(フレーズ組みのリスト)
2. リザルト: []
3. 重複フラグ: bit[ノードの数] = 1...
4. for フレーズ組み in フレーズ組みのリスト:
5.   for フレーズ in フレーズ組み:
6.     重複フラグ &= フレーズ.使用ノードビット列
7.   if not 重複フラグ:
8.     リザルトに追加する(フレーズ組み)
9. return リザルト
```

図 5:同一語フィルタアルゴリズム

5. まとめ

本稿では見出し候補生成で小説家の見出し創作支援を目的に依存ネットワークを考慮して語の重複を除いた見出し候補生成アルゴリズムの提案を行った．

今後の展望としては，生成したフレーズ同士の元文での関係性を保持することで，元文の意味を反映させた見出し候補を生成したい．

引用文献

[1]**西原陽子他**，多くの人の興味を引く研究発表タイトル作成支援，人工知能学会全国大会論文集JSAI07巻,2H-45, 2007.

[2]**中野滋徳他**，依存構造を利用した言い換え文の自動生成，情報処理学会研究報告 2004-NL-159(26),p185-191, 2004.

[3]**中野滋徳他**,復文構造における言い換えについて,情報処理学会研究報告 2004-NL-160(10),p65-70.

Copyright ©2023 Information Processing Society of Japan.
All Rights Reserved.