

修 士 論 文 の 和 文 要 旨

研究科・専攻	電気通信大学 情報理工学研究科 情報・ネットワーク工学専攻 博士前期課程		
氏 名	安部文紀	学籍番号	1631005
論 文 題 目	575 の音韻的読みやすさを付与した学術論文の要約文自動生成手法		
<p>要 旨</p> <p>学術論文の投稿や検索に特化した Web サービスの台頭によって、論文を大量に入手できるようになった。これに伴い、「論文を読むべきかどうか判断する機会」と、概要を思い出すために「論文を読み返す機会」が増えた。一般的にこのような機会には論文のタイトルやアブストラクトが読まれるが、タイトルとアブストラクトよりも短く、かつ印象に残りやすい文章が論文に付与されていれば、「論文を読むべきかどうか判断する手間」と「概要を思い出す手間」が軽減することが期待される。</p> <p>そこで本研究では、論文をタイトルとアブストラクトよりも少ない文章量で、かつ印象に残りやすく要約するために、古来より親しまれてきた俳句や川柳のような 575 の持つ音韻的読みやすさを付与した要約文が望ましいと考え、学術論文の新たな要約手段として 575 形式のキャッチフレーズで論文を表現する「575 自動生成手法」を提案する。</p> <p>575 自動生成手法は、「特徴語抽出」と「575 候補生成」、候補絞り込みのための「合議制スコアリング」の 3 つのモジュールから構成される。特徴語抽出では、入力する論文における単語の出現頻度に加えて、論文の持つ意味を加味した単語スコアリングによって特徴語を決定する。575 候補生成では、ソフトウェア工学関連の学会論文集から人手で作成した 575 を基に、575 テンプレートを作成し、特徴語を当てはめることで 575 候補を生成する。合議制スコアリングでは、人手で 575 を作成する工程で得た知見を反映させた評価指標によって 575 候補をスコアリングし、最終的な出力を決定する。</p> <p>評価実験では、575 自動生成手法を学術論文に適用したときに自動生成された 575 (論文 575) を用いて、論文タイトルと比較するユーザスタディを行った結果、タイトルよりも少ない文章量で論文概要を表現できることを確認した。また、論文 575 から受ける印象を人手生成の 575 と比較したところ、手製のものより劣る結果となった。そのため、人間の印象に残りやすい論文 575 を生成する手法の探求を今後の課題とする。</p>			

平成 2 9 年度修士論文

575 の音韻的読みやすさを付与した
学術論文の要約文自動生成手法

情報・ネットワーク工学専攻
コンピュータサイエンスプログラム

1 6 3 1 0 0 5 安部 文紀

主任指導教員 寺田 実 准教授

指導教員 沼尾 雅之 教授

提出日 2 0 1 8 年 1 月 2 9 日

概要

学術論文の投稿や検索に特化した Web サービスの台頭によって、論文を大量に入手できるようになった。これに伴い、「論文を読むべきかどうか判断する機会」と、概要を思い出すために「論文を読み返す機会」が増えた。一般的にこのような機会には論文のタイトルやアブストラクトが読まれるが、タイトルとアブストラクトよりも短く、かつ印象に残りやすい文章が論文に付与されていれば、「論文を読むべきかどうか判断する手間」と「概要を思い出す手間」が軽減することが期待される。

そこで本研究では、論文をタイトルとアブストラクトよりも少ない文章量で、かつ印象に残りやすく要約するために、古来より親しまれてきた俳句や川柳のような 575 の持つ音韻的読みやすさを付与した要約文が望ましいと考え、学術論文の新たな要約手段として 575 形式のキャッチフレーズで論文を表現する「575 自動生成手法」を提案する。

575 自動生成手法は、「特徴語抽出」と「575 候補生成」、候補絞り込みのための「合議制スコアリング」の 3 つのモジュールから構成される。特徴語抽出では、入力する論文における単語の出現頻度に加えて、論文の持つ意味を加味した単語スコアリングによって特徴語を決定する。575 候補生成では、ソフトウェア工学関連の学会論文集から人手で作成した 575 を基に、575 テンプレートを作成し、特徴語を当てはめることで 575 候補を生成する。合議制スコアリングでは、人手で 575 を作成する工程で得た知見を反映させた評価指標によって 575 候補をスコアリングし、最終的な出力を決定する。

評価実験では、575 自動生成手法を学術論文に適用したときに自動生成された 575 (論文 575) を用いて、論文タイトルと比較するユーザスタディを行った結果、タイトルよりも少ない文章量で論文概要を表現できることを確認した。また、論文 575 から受ける印象を人手生成の 575 と比較したところ、手製のものより劣る結果となった。そのため、人間の印象に残りやすい論文 575 を生成する手法の探求を今後の課題とする。

キーワード

自然言語処理学, 文書要約, 自動要約, 575 生成, 特徴語抽出

目次

第 1 章	序論	7
1.1	背景	7
1.2	目的	7
1.3	本論文の構成	8
第 2 章	575	9
2.1	575 のメリット	9
2.1.1	音韻的な読みやすさ	9
2.1.2	記憶に残りやすさ	9
2.2	575 の形式が採用されている実績	10
第 3 章	先行研究	11
3.1	学術論文から要約文を生成する研究	11
3.1.1	概要	11
3.1.2	抽出型要約	11
3.1.3	生成型要約	11
3.1.4	本研究との関連や差異	12
3.2	学術論文からタイトルを生成する研究	12
3.2.1	抽出型・生成型のタイトル生成	12
3.2.2	本研究との関連や差異	12
3.3	文書から特徴語を抽出する研究	12
3.3.1	表層情報を用いる手法	12
3.3.2	潜在情報を用いる手法	13
3.3.3	本研究との関連や差異	13
第 4 章	本研究の位置付け・関連研究	14
4.1	俳句生成の視点から	14
4.1.1	俳句生成の研究	14
4.1.2	本研究との関連や差異	14
4.2	歌詞生成の視点から	15
4.2.1	歌詞生成の研究	15
4.2.2	本研究との関連や差異	15

第 5 章	予備実験	16
5.1	関連研究の追実装	16
5.1.1	目的	16
5.1.2	方法	16
5.1.3	結果と考察	17
5.2	学術論文を 575 で表現する実験	18
5.2.1	概要	18
5.2.2	準備	18
5.2.3	方法	19
5.2.4	結果と考察	19
第 6 章	提案手法	20
6.1	要件	20
6.1.1	少ない文章量で論文を表現できる	20
6.1.2	印象に残りやすい	20
6.2	概要	20
6.3	特徴語の抽出	21
6.3.1	概要	21
6.3.2	文の表層情報に基づく手法	21
6.3.3	文の潜在情報に基づく手法	22
6.3.4	unigram-rescaling	23
6.4	575 候補の生成	25
6.5	合議制スコアリングによる 575 の決定	27
6.5.1	概要	27
6.5.2	タイトルとの cos 類似度, アブストラクトとの cos 類似度	27
6.5.3	上五・中七・下五の各句の係り受けスコア	27
6.5.4	575 に含まれる単語のポジティブ・ネガティブスコア (ポジネガスコア)	28
6.5.5	合議制スコア	28
6.5.6	合議制スコアのパラメータ $\alpha, \beta, \gamma, \omega$ の重みの決定	28
第 7 章	実装	31
7.1	準備	31
7.2	提案システム: 論文要約 575	31
7.2.1	GUI	31
7.2.2	使い方	32
7.2.3	出力例	33
第 8 章	評価実験	34
8.1	実験 1: 合議制スコアリングの妥当性評価	34
8.1.1	概要	34
8.1.2	方法	34

8.1.3	結果	34
8.2	実験 2: 論文を表現できるか確認	35
8.2.1	概要	35
8.2.2	1 つ目: 論文のアブストラクトと参照 575 (手製の論文 575) を結びつける問題 5 題, 論文のアブストラクトとシステム 575 を結びつける問題 5 題	35
8.2.2.1	方法	35
8.2.2.2	結果	35
8.2.3	2 つ目: 論文のアブストラクトとタイトルを結びつける問題 5 題, 論文のアブストラクトとシステム 575 を結びつける問題 5 題	36
8.2.3.1	方法	36
8.2.3.2	結果	36
8.2.4	まとめ	36
8.3	実験 3: 論文が印象に残りやすいか確認	36
8.3.1	概要	36
8.3.2	方法	37
8.3.3	結果	37
第 9 章	結論	39
9.1	まとめ	39
9.2	今後の課題	40
9.2.1	論文 575 の質	40
9.2.2	論文の利用方法	40
9.2.3	計算量	40
9.3	展望	40
9.3.1	利用例	40
9.3.2	応用例	40
参考文献		42
付録		44
A.	提案手法に用いた 575 テンプレート一覧	44

図目次

1.1	本論文を提案システム (論文要約 575) に入力した場合に想定される 575 の出力例	7
5.1	Rafal らの特徴語抽出手法を筆者の過去の学術論文に適用したときに得られた名詞と形容詞 . .	18
6.1	学術論文の 575 を生成するまでのステップ	21
6.2	LDA によるトピック分布と単語分布の生成例	22
7.1	提案システム: 論文要約 575 の動作例 (論文入力前)	31
7.2	提案システム: 論文要約 575 の動作例 (論文入力後)	32
8.1	形容詞対によるシステム 575 と参照 575 の印象の比較 (SD 法)	38

表目次

6.1	パラメータ β の最適値決定のための $nDCG_{20}$ の結果	25
6.2	提案特徴語抽出手法で学術論文から抽出した特徴語上位 20 個	26
6.3	作成した 575 テンプレートの例	26
6.4	ランダムサーチで得られたパラメータの重み組み合わせ 5 つ	29
6.5	パラメータの重み組み合わせを他論文に適用したときに F 値の和が最も高くなった回数 . . .	30
8.1	ソート済みシステム 575 と未ソートシステム 575 の「良い」評価の数 (最大 5)	35
8.2	実験 3 に用いた形容詞対の表	37

第 1 章 序論

1.1 背景

研究をするにあたって学術論文を調査することは必要不可欠である. 近年は Google Scholar^{*1} のような論文検索エンジンや, arXiv^{*2} のような論文投稿 Web サイトの台頭によって所望の論文を大量に, かつ容易に入手できるようになった. これに伴って, 「論文を読むべきかどうか判断する機会」と概要を思い出すために「論文を読み返す機会」が増えた. このような機会では, 一般的に論文のタイトルやアブストラクトが読まれるが, タイトルとアブストラクトよりも短く, かつ印象に残りやすい文章が論文に付与されていれば, 「論文を読むべきか判断する手間」と「概要を思い出す手間」が軽減することが期待される.

1.2 目的

本研究では, 学術論文を読むべきかどうか判断する手間と, 論文の概要を思い出す手間の軽減を目的に, 古来より親しまれてきた俳句や川柳といった 575 の持つ音韻的な読みやすさを付与した要約文の生成に取り組む. これによって, 学術論文をタイトルとアブストラクトよりも少ない文章量で, かつ印象に残りやすく表現することを目指す. そして, これを実現するための論文の 575 (575 形式のキャッチフレーズ) 自動生成手法を提案する. 例えば, 本論文を提案手法に入力した場合, 図 1.1 のような要約文が出力されることを想定する.

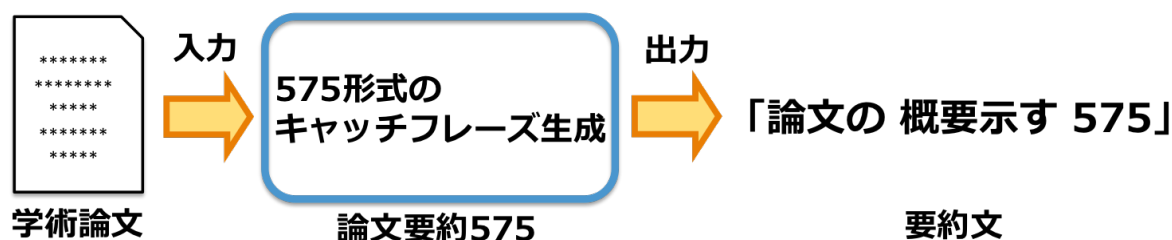


図 1.1: 本論文を提案システム (論文要約 575) に入力した場合に想定される 575 の出力例

本研究は, 学術論文を少ない文章量で, かつ印象に残りやすく要約することで論文読者に貢献する. また, 論文に付与するキャッチフレーズやキーワードの生成を代行, もしくは考案を支援することで論文著者に貢献する. 科学技術系の学術論文には研究のダイジェストとなる画像やイメージ図, もしくは研究のキーワードが論文の冒頭に付与される習慣があるため, 将来的にはそのような研究のダイジェストにおいて「論文要約 575」が併用

^{*1} <https://scholar.google.co.jp/>

^{*2} <https://arxiv.org/>

されることを目指す。

1.3 本論文の構成

本論文の章立てと内容を概説する。本章では、序論として研究の背景と目的について述べた。第2章では、本研究の要となる 575 の関連研究を利点の観点から、そして実績を交えながら説明する。第3章では、先行研究を「要約文生成の研究」、「タイトル生成の研究」、そして「特徴語抽出の研究」の3つに分け、各研究を紹介しながら本研究との関係や差異について述べる。第4章では、関連する研究を挙げながら本研究の位置付けを整理し、また、本研究の方向性を示す。入力する文書に学術論文を用いることとした理由も本節で述べる。第5章では、提案手法を裏付けるために行った予備実験について述べる。予備実験は2種類行い、ひとつは関連研究で提案された手法の追実装、もうひとつは学術論文を 575 形式のキャッチフレーズで表現することの有効性を確認する実験である。第6章では、提案手法である論文の 575 自動生成手法について述べる。学術論文を入力してから 575 を提案手法で生成するまでの過程を3つのステップに分け、それぞれに節を設けて詳細に説明する。第7章では、提案手法の実装について述べる。提案手法をシステムとして興したアプリケーションの GUI を概観しながら実際の出力例を示す。第8章では、提案手法の評価実験について述べる。本研究では、提案手法を用いることで「少ない文章量で論文を表現できる」、「印象に残りやすい」という2つの効果が期待されるため、満たしているか確認するための実験を3つ行った。第9章では、予備実験と評価実験を通して得た知見を踏まえながら本研究をまとめ、今後の課題と展望について述べる。

第 2 章 575

本研究の要となる 575 の関連研究について、利点と実績を交えながら説明する。

2.1 575 のメリット

本研究では要約文の生成に 575 の形式を用いるが、575 には二つのメリットが存在する。ひとつは「音韻的に読みやすいこと」、もうひとつは「記憶に残りやすいこと」である。

2.1.1 音韻的な読みやすさ

一つ目のメリットについて述べる。575 は、作成される際に五音節と七音節を基調とされており、言い換えると基準となる音数の規則が存在する。これを五七調・七五調と呼ぶ。坂野 [11] によると、五音と七音が心地良いと結論づけており、理由は、二音一単位の「律拍」という概念に基づいて二音、四音、八音、... (2 の冪乗) は繰り返すとリズムカルでありスムーズであるためと述べている。575 は上五、中七、下五のそれぞれが 2 の冪乗の音数にはなっていないが、五音の後に三音の休止 (空白) が入り、七音の後には一音の休止が入り、それぞれが八音となっている。つまり、575 は律拍のなかでも八音を基本の単位としており、八音に満たない部分については休止が入るようになっていると述べている。

また、渡部ら [12] は、古来より五七調・七五調のリズムが良いとされていることについて疑問を呈し、現代人でも五七調・七五調がリズムカルに感じるかどうかを実証実験した。結果は七五調、五七調、六六調、四八調、八四調の順に心地良いと被験者が評価したため、「五音七音がリズムカルと感じるのは、単になじみのある型の持つ心地よさというだけではなく、五音七音という音の構成がリズム知覚に影響している」と結論づけている。

2.1.2 記憶に残りやすさ

二つ目のメリットについて述べる。越場ら [13] は、リズムカルな五七調・七五調は記憶に残りやすいという仮説を掲げ、五七調が暗記学習に効果があるかを実証実験した。歴史の暗記問題を韻律を考慮しない語呂合わせで覚えてもらった場合と、韻律を考慮した五七調の語呂合わせで覚えてもらった場合の正解数を比較したところ、後者の正解数の方が多かったため、「五七調は暗記学習に効果があることがわかった」と結論づけている。

まとめると、575 の五七調はリズムカルで耳に残りやすい特徴があるため、文章が記憶に残りやすいメリットがあると言える。学術論文は長文で堅苦しい言い回しがあるため、論文を身近な五七調で表現することによって読みやすく、印象に残りやすくなると考える。

2.2 575 の形式が採用されている実績

575 の形式が読みやすく、印象に残りやすいことを利用して注意喚起や啓発イベントにて標語を広告する際に五七調が採用されている例をいくつか紹介する。

北海道旭川市で行われた選挙の際には、投票へ行くことを促すためのスローガンを募集し、採用されたスローガンのほとんどが五七調であった^{*1}。また、広島県広島市では毎年人権啓発イベントを行っており、その際にスローガンを募集している。採用されているほとんどが五七調である^{*2}。その他、交通安全の標語でも五七調は多く採用されている^{*3*4}。

五七調が知覚に訴えるリズムの良さと五七調の持つ印象に残りやすさは、周知の事実であることが実績を通して確認できる。

^{*1} <http://www.city.asahikawa.hokkaido.jp/kurashi/461/463/d053708.html>

^{*2} <http://www.city.hiroshima.lg.jp/www/contents/1268627303816/index.html>

^{*3} <http://www2.tadsa.or.jp/anzen2017s/>

^{*4} <http://www.mainichi.co.jp/event/aw/anzen/slogan/archive.html>

第 3 章 先行研究

本章では, 学術論文を把握しやすくすることや, 短い文章で説明することを目的に取り組まれてきた研究を 3 つの節に分けて説明する.

3.1 学術論文から要約文を生成する研究

3.1.1 概要

学術論文の概要を把握しやすくすることを目的として, 要約文を生成する研究がこれまで取り組まれてきた. 要約文生成の研究には「抽出型要約」と「生成型要約」が存在する. 抽出型要約では論文中の各文を重要度の高い順に並び替え, いくつか抽出したものを要約文としている. 生成型要約では, 論文から抽出した文を要約文として扱うことはせず, 人間が要約を作る際に行う言い換え表現や, 不要な文字列の除去のような単語列の再構成を機械学習によって行い, 生成した新しい単語列を要約文としている.

3.1.2 抽出型要約

抽出型要約の重要度の導出において, Shin らの研究 [1] では \LaTeX 文書のセグメント構造に着目し, 各セグメントに n -gram に基づく手法や TF-IDF に基づく手法などの異なる評価基準を持たせて各文の重要度を導出した. 中須賀らの研究 [2] では論文中の文や句間の関係を表す談話構造に着目し, 各文から抽出した談話構造に基づく特徴量を用いてアブストラクトとの類似度を測る評価基準を持たせて重要度を導出した.

また, Amjad らの研究 [3] では, 入力する論文を引用している論文から著者を参照している引用に関連する部分を抽出するために, Page-Rank アルゴリズムから着想を得た LexRank[4] を用いて関連する語句部分の重要度を測る手法を提案した. 他にも, Structured Abstract と呼ばれる背景, 目的, 方法, 結果, 結論の 5 つの見出しからなるアブストラクトを正解データとして用い, 各見出しに論文の各文を振り分ける 5 値分類器を機械学習した研究 [6] が存在する.

3.1.3 生成型要約

生成型要約の単語列の生成において, 機械翻訳のタスクで用いられるニューラルネットワークの Encoder-Decoder モデルを論文要約のタスクに適用し, Recurrent Neural Network (RNN) に全文を一度に入力するのではなく論文の章構造に基づき, セクションごとに文を入力する方法で要約文を生成する研究 [5] が存在する. Encoder-Decoder モデルとは, 入力される文を単語に分割して意味ベクトルで単語を表現しながら情報を蓄積する RNN (Encoder) と, 蓄積された情報 (ある単語の次に生起する単語の確率) を取り出しながら単語を生成する RNN (Decoder) を 2 つ組み合わせたモデルである.

3.1.4 本研究との関連や差異

要約文生成のどの研究においても、取り組まれてきたほとんどが Structured Abstract や Extended Abstract のような長文の要約を生成する研究であるため、論文の内容の理解支援を目的としている。本研究では、五七調で構成された 17 音程度の音韻的に読みやすい要約文を生成することで、論文内容の理解支援ではなく、内容理解の前段階で行う「論文を読むべきかどうか判断すること」の支援を目的とする。

3.2 学術論文からタイトルを生成する研究

論文執筆におけるタイトルの作成支援を目的として、学術論文からタイトルを生成する研究がこれまで行われてきた。

3.2.1 抽出型・生成型のタイトル生成

タイトル生成の研究は、一種の要約文生成とみなして取り組まれてきた。そのため、学術論文から要約文を生成する研究と同様に、本文から抽出した最重要な一文をタイトルとする抽出型の手法 [7] や、Encoder-Decoder モデルを用いてアブストラクトから新しい単語列を生成する生成型の手法 [8] が存在する。他にも、Jan らの研究 [9] では、事前に用意したテンプレートに抽出した特徴語を当てはめてタイトルを生成する抽出型と生成型の中間に位置する手法と、K-nearest Neighbor (KNN) を用いて入力する論文のアブストラクトと類似度の高い文を事前に用意したコーパスから抽出する抽出型の手法を提案した。

3.2.2 本研究との関連や差異

論文のタイトル生成研究は、3.1 節にて述べた要約文生成の研究と比較すると出力する文章の量は少ないが、本研究は五七調で構成される 17 音で要約するため、タイトルは 575 と比較すると一覽性に欠ける。また、2.1 節で述べたとおり、575 を構成する日本語の五音節と七音節は読みやすく、印象に残りやすい利点を持つため、印象に残りやすさの点においてもタイトルよりも優位性を有す。

3.3 文書から特徴語を抽出する研究

文書から特徴語を抽出する研究がこれまで取り組まれてきた。特徴語抽出の研究には文書の「表層情報を用いる手法」と「潜在情報を用いる手法」が存在する。

3.3.1 表層情報を用いる手法

表層情報とは文書における単語の出現頻度を基にした情報のことで、TF-IDF (Term Frequency – Inverse Document Frequency) や共起度 [10] などが存在する。提案手法では TF-IDF を利用するため、概念を説明する。TF-IDF とは、文書に存在する単語の重要度を計算する尺度であり、TF では、ある文書に何度も出てくる単語は重要であるという考察より、文書中の高頻度語が重要となるように設定されている。IDF では、単語の出現する文書数が多いことは、その単語が一般的であり、重要性が低いという考察より、出現する文書数が少ない単語が重要となるように設定されている。

3.3.2 潜在情報を用いる手法

潜在情報とは文書に潜在する(文書の持つ)意味などを考慮した情報のことで、確率的潜在意味解析 (Probabilistic Latent Semantic Analysis, PLSA) や 潜在的ディリクレ配分法 (Latent Dirichlet Allocation, LDA)[22] などが存在し、一般にトピックモデルと呼ばれる。提案手法ではトピックモデルを用いるため、概念を説明する。トピックモデルとは、1つの文書が複数のトピックを持つと仮定する確率モデルであり、文書集合から潜在的なトピック構造を発見する際に用いられる。トピックモデルによって、文書に潜在するトピックとその生起確率や、トピックごとの単語の生起確率を求めることができる。

3.3.3 本研究との関連や差異

取り組まれてきた特徴語抽出の研究のほとんどは、キーワードに名詞を用いる場合が多く、抽出した名詞をどのように利用するかは位置付けが不明瞭である問題点があった。そこで本研究は、抽出した名詞のキーワードを論文 575 を構築するための素材として扱う。

本研究では、提案システムに入力する文書を「575 を生成したい論文一本」としているため、その論文の特徴語抽出に 575 の品質が委ねられている。そこで、単純に単語の出現頻度のみで特徴語を抽出するのではなく、文書の潜在情報も併せて用いた特徴語の抽出を行うこととする。

科学技術系学術論文の冒頭に研究のキーワードとなるキーワード群を付与して論文の特徴を表す習慣があるが、論文 575 を用いることで論文を端的に表すことに加えて、論文を印象に残りやすくする効果が期待される。

第 4 章 本研究の位置付け・関連研究

本章では、本研究を位置付けるための研究を紹介する。

4.1 俳句生成の視点から

4.1.1 俳句生成の研究

古来より俳句は人手で書かれ詠まれ親しまれてきたが、自然言語処理技術を用いて機械的に俳句を生成しようと試みた研究がいくつか存在する。Rafal らの研究 [14] では、Web に存在する文書を言語資源として使い、Web 検索ヒット数とユーザが選択するブログ記事における単語の出現頻度を基に俳句に用いる単語を決定し、オリジナルの文法テンプレートに抽出した単語を割り当てて俳句を自動生成する手法を提案した。Daniel らの研究 [15] では、Wikipedia を言語資源として用い、単語 N-gram モデルによって俳句を自動生成する手法を提案した。Yeal らの研究 [16] は、俳句協会の Web サイトや童話、松尾芭蕉の俳句等を言語資源として用い、連想語ネットワークを利用して俳句に用いる単語を決定し、品詞 N-gram モデルによって俳句を生成する手法を提案した。神澤らの研究 [17] では Web に存在する文書を言語資源として使い、システムに入力される単語との共起を Web 検索のヒット数で評価し、入力単語およびそれと共起度の高い単語を俳句に用いる単語とし、オリジナルの読み句テンプレートに抽出した単語を割り当ててカルタの読み句を生成する手法を提案した。また、Martin らの研究 [18] では、俳句に多用される 550 単語を格納したシード単語辞書を用意し、言語資源であるブログ記事から俳句に使う文字列を検索するためのクエリとして用いた。そして、ヒットした単語を含む文をブログ記事から抽出し、Vector Space Model と TF-IDF を用いて俳句に用いる文を選択、結合する手法を提案した。他にも、Xianchao らの研究 [19] では日本語の俳句を生成するためのニューラルネットワークの RNN に基づく手法を 4 つ提案し、りんな (Microsoft 社が提供する対話 Bot) のユーザとの会話で得たクエリ履歴と Web から収集した俳句を使ってそれらを学習させた。

4.1.2 本研究との関連や差異

これまで取り組まれてきた俳句生成の研究は、機械的に生成する俳句が人手で作った俳句にどれだけ近付けられるかを目的に、そして俳句の美しさをどれだけ持てるかを課題に行われてきた。そのため、俳句の自動生成を要約文の生成と見なして取り組んだ研究が存在しない。また、本研究と同様に入力文書の存在を仮定している俳句生成の研究も存在するが、生成した俳句の候補から最終的なひとつに絞り込む手法の提案はされていない。

本研究では、入力論文から要約文と見なした 575 の候補を生成し、最終的なひとつを決定するまでの一貫した 575 自動生成手法を提案する。そして、「少ない文章量で論文を表現できるか」、「印象に残りやすいか」を評価実験で確認する。

4.2 歌詞生成の視点から

4.2.1 歌詞生成の研究

自然文から歌詞を自動生成する研究がこれまでにいくつか取り組まれてきた。渡邊らの研究 [20] では、出力する歌詞に特定のテーマを持たせる・歌詞の流れを考慮するような N-gram モデルを基本とした言語モデルを 3 種類提案し、評価実験により適切なモデルを探した。山本らの研究 [21] では、入力される自然文から五音・七音のフレーズを切り出して集合を作り、歌詞とする手法を提案した。

4.2.2 本研究との関連や差異

これらの研究は、歌詞創作の支援を目的に、音韻的に読みやすい文の生成を目指している。本研究との相違点は、入力する文の形式と出力する文の形式をあらかじめ定めている点である。

まず、入力する文の形式について説明する。歌詞生成の研究では入力する文を自然文全般としているが、以下に示す 2 つの理由により本研究では学術論文に統一している。一つは、論文であれば著者の伝えたい主張点が明確なため、特徴語を抽出する際や生成する 575 の重要度の導出において利用できる判断材料が豊富であると考えたためである。本提案手法では、入力する論文の全文をそのまま文字列として扱うこととしているが、特徴語抽出手法のパラメータの重み決めと、生成する 575 の候補から最終的な出力ひとつを決めるための指標として論文のタイトルと論文のアブストラクトを全文とは別に用いることとしている。二つ目の理由は、論文は一般の文書と比較するとあらかじめ決まった構造を持つ文章であるため、手がかりが多く存在すると考えたためである。論文の持つ構造を利用することも有力な手段であると考えながら、今後の課題としたい。

次に出力する文の形式について説明する。歌詞生成の研究では、出力する文に決まった形式を持たせていないが、本研究では出力文を 575 の形式に統一している。これは、2.1 にて述べた五七調の持つ二つのメリットに基づいているためである。

第 5 章 予備実験

5.1 関連研究の追実装

5.1.1 目的

Rafal らの研究 [14] で提案された俳句生成アルゴリズム (後述する) にて用いられている特徴語抽出手法を学術論文に適用し, 特徴語抽出手法が論文にも適用できるかどうかを確認することを目的とした. また, 結果を確認しながらどの品詞を使うか検討することも目的とした.

ここで, Rafal らの俳句生成アルゴリズムについて説明する. 彼らの手法は大きく「特徴語の抽出」と「文法テンプレートをを用いた俳句の生成」の 2 つのモジュールから構成されている. 特徴語の抽出では, 俳句に用いる単語を名詞と形容詞に限定して 5.1.2 節にて説明する手法により俳句に用いる特徴語を決定した. 文法テンプレートをを用いた俳句の生成では, Web 検索で見つかった 20000 詩から 312 個の詩をランダムに選択し, それを基に代表的な文法パターンを 7 個人手で作成した. 作成した文法パターンに抽出した特徴語を配置し, 上五・中七・下五を Web 検索にかけて完全一致の結果が 0 件でなければ文法的・意味的に成り立つと仮定して俳句を生成した.

5.1.2 方法

提案された特徴語抽出手法を学術論文に適用し, 名詞と形容詞の抽出およびランク付けを行った. ここで, Rafal らの特徴語抽出手法を交えながら本予備実験の方法を説明をする. 特徴語抽出は 4 つのステップに分かれており, 以下の通りである.

- テーマ単語の決定
- 名詞のスコアリング
- 形容詞のスコアリング
- 季語の決定

本研究の 575 は俳句ではなく川柳であり, 季語を不要とするため 4 つ目の季語の決定については実装しないこととした.

まず, テーマ単語の決定について説明する. テーマ単語 w は, 入力した文書における名詞を出現頻度の高い順に並べたときの上位から選ばれ, また, 他名詞との共起が高い名詞が選ばれる. 具体的には, 式 (5.1) によって高頻度かつ一般的でない単語ランキングを作り, 式 (5.2) によってランキングのトップ n 位 (ここでは $n = 5$ とした) までの単語同士で共起しやすい単語を共起確率を計算することで求め, テーマ単語 w とする. 一般的な単語とは, 多くの文書に広く出現する単語のことを指す. 式 (5.1) の w' は入力文書中の名詞であり, $N_o(w')$ は, 単語 w' を Web 検索したときのヒット数, $c(w')$ は入力文書における単語 w' の出現回数である. 式 (5.2) の N_1 はラン

キング内の任意の単語を N_2 としたときのそれ以外の単語である。 $Retr(N_1, N_2)$ は、 N_1 と N_2 を検索エンジンで AND 検索したときのヒット数であり、 $Retr(N_1)$ は N_1 の検索ヒット数である。

$$ThemeScore(w') = N_o(w')^{c(w')} \quad (5.1)$$

$$CooccurrenceProb(N_1, N_2) = \frac{Retr(N_1, N_2)}{Retr(N_1)} \quad (5.2)$$

式 (5.2) において、 N_2 におけるその他の N_1 との共起確率を求め、足しあわせたときに最高値となる N_2 をテーマ単語 w とする (N_2 はすべての単語について計算する)。ここで決定したテーマ単語 w は、名詞と形容詞をスコアリングする際に用いる。

次に、名詞のスコアリングについて説明する。名詞は、テーマ単語 w と入力文書のタイトル t に関連する度合いを式 (5.3) によって計算することでスコアリングされる。

$$NounScore(N_1) = \frac{CooccurrenceProb(N_1, w)}{CooccurrenceProb(N_1, t)} \quad (5.3)$$

式 (5.3) の N_1 は入力文書中の名詞であり、 $CooccurrenceProb(N_1, w)$ は式 (5.2) と同様に計算する N_1 と w の共起確率、 $CooccurrenceProb(N_1, t)$ は N_1 と t の共起確率である。導出するスコアが 1 を超えた場合、逆数をとった値をスコアとする。これによって、スコアが 1 に近い名詞ほど、入力文書のテーマとタイトルへの関連が強くなるとみなすことができる。

最後に形容詞のスコアリングについて説明する。形容詞は、名詞を Web 検索したときに得られる結果上位 20 件のスニペット内の形容詞を解析することでスコアリングする。具体的には、式 (5.4) を用いて名詞 N_1 を Web 検索したときに得られるスニペット中に存在する形容詞のスコアを計算する。

$$AdjScore(N_1, Adj(N_1)) = \frac{Retr(Adj(N_1), N_1)}{Retr(N_1)} \times C(N_1, Adj(N_1)) \quad (5.4)$$

式 (5.4) の $Adj(N_1)$ には、 N_1 を Web 検索したときの上位 20 件のスニペットに含まれる形容詞がひとつずつ該当する。 $C(N_1, Adj(N_1))$ は、 N_1 の検索 1 回における $Adj(N_1)$ の 20 個のスニペットにおける出現回数である。この式で求められるスコアの高い形容詞は、多くのスニペットに出現し、かつ名詞との共起が多い、つまり名詞とセットで使われることが多いとみなすことができる。

本予備実験では、ソフトウェア工学関連の学会 (WISS2016^{*1}) で筆者が発表したデモ発表論文^{*2}を入力文書として用いた。この論文は、スマートフォンでのブラウジング時に、タップ操作のみで調べたい検索単語を Web 検索できる仕組みを提案した内容の研究であった。

5.1.3 結果と考察

筆者の過去の学術論文を入力として、実際に出力された上位 5 つの単語を図 5.1 に示す。

名詞は入力論文のキーワードとして適切な単語が抽出できた。一方で形容詞については、提案手法が入力する文書の中に存在しない単語を Web 検索によって抽出する手法であったことから、学術論文の要約文に用いるには不適切と思われる単語が抽出された。また、Rafal らは形容詞のスコア付け方法を見直す必要があると述べて

^{*1} <https://www.wiss.org/WISS2016/>

^{*2} <http://www.wiss.org/WISS2016Proceedings/demo/3-A17.pdf>

名詞ランキング：
["検索", "単語", "スコア", "タップ", "位置"]

形容詞ランキング：
["むずかしい", "速い", "新しい", "早い", "詳しい"]

図 5.1: Rafal らの特徴語抽出手法を筆者の過去の学術論文に適用したときに得られた名詞と形容詞

いるため、本研究では形容詞を用いないこととし、名詞だけを特徴語として抽出することとした。なお、本研究では学術論文のみを扱うため、Web にある文書のような一般文書の存在を仮定している彼らの特徴語抽出手法は使わないこととした。

ここで、Rafal らの手法と後述する本研究の提案手法との関連と差異を整理して述べておく。Rafal らの手法と本研究は入力文書の存在を仮定している点で類似しているが、彼らの手法では生成する複数の俳句候補から出力するひとつを絞り込むための手法の提案はされていない。本研究では、入力する論文の要約文とみなした 575 の候補を生成し、候補から最終的なひとつを決定するまでの一貫した 575 自動生成手法を提案する。また、彼らの特徴語抽出手法では、単語の出現頻度と共起度を用いているため、入力する文書を表現する俳句の生成を今後の課題としている。本研究では、特徴語の抽出に単語の出現頻度に入力文書の持つ意味を統合した手法を提案するため、より文書を表現できることが期待される。まとめると、Rafal らは出力する 575 に俳句の良さ(美しさ)を追求したため、評価実験の結果、文書の持つ意味をより表現する必要性が課題として残った。本研究では、目的のひとつに「少ない文章量で論文を表現すること」があるため、575 自動生成手法では文書の持つ意味を考慮する。

5.2 学術論文を 575 で表現する実験

5.2.1 概要

自動生成した 575 を用いて「少ない文章量で論文を表現できるか」、「印象に残りやすいか」を評価実験で確かめる前に、まずは学術論文の要約文を 575 形式のキャッチフレーズで表現することの有効性を確認する必要があると考えた。そこで、ソフトウェア工学関連の学会論文集一冊から筆者が人手で作成した 575 形式の要約文を評価者に読んでもらう実験を 2 行なった。具体的には、学術論文から作成した 575 が、論文にあらかじめ付与されているタイトルと同じくらい、もしくはそれ以上に論文を表現できるかをユーザスタディにより確認した。

5.2.2 準備

実験に用いる学術論文とその 575 を用意するために、WISS 2016^{*3} 論文集^{*4} に掲載されている登壇発表とデモ・ポスター発表の論文(合計 82 本)を読み、筆者の主観で各論文について 575 形式のキャッチフレーズ(全 82 個)を作成した。論文タイトルと作成した 575 例を以下に示す。

^{*3} <https://www.wiss.org/WISS2016/>

^{*4} <http://www.wiss.org/WISS2016Proceedings/>

3D プリンタのための Paper User Interface ^{*5}

→ オブジェクト スキャンするだけ 二面図を

着ぐるみ非装着時の着ぐるみポージング練習システム ^{*6}

→ 着ぐるみの ポーズ練習 支援する

物理ベースサウンドデザインのための例示ベースインタフェース ^{*7}

→ 音分布 例示ベースで 生成し

5.2.3 方法

実験協力者である本学の学生 5 人に、以下 2 つの実験を行ってもらった。

1. 学術論文のアブストラクトとその 575 を結びつける実験
2. 学術論文のアブストラクトとそのタイトルを結びつける実験

各実験、ひとつのアブストラクトにつき正解 (実験 1 では正解 575, 実験 2 では正解タイトル) を 1 つ, 不正解を 4 つ含んだ解答候補を 5 つ提示し, 5 問 (5 本の論文) について回答してもらった。実験で扱うアブストラクトは WISS 2016 論文集に掲載されている 82 本の論文からランダムに抽出した 10 本の論文とし, 5 本を実験 1 に, 残り 5 本を実験 2 に用いた。

学術論文のタイトルは、一般に論文内容の主張点を的確に表したものとして扱われるため, 575 のアブストラクト正解率 (実験 1) がタイトルのアブストラクト正解率 (実験 2) と同等, もしくはそれ以上であれば学術論文の概要を 575 (タイトルよりも少ない文章量) で表現することの有効性を示せることとした。

5.2.4 結果と考察

実験 1 と実験 2 の正解率が両方とも 100 % となった。したがって, 学術論文を 575 で表現することは有効であり, また, 論文のタイトルと同等の役割 (論文内容の主張点を的確に表す) を持つことがわかった。まとめると, 575 形式のキャッチフレーズで学術論文の概要を表現することは可能である。

本予備実験の準備作業として行った, 論文から 575 を人手で作る工程で得た知見を, 提案手法により生成する 575 の候補から適切なひとつを出力する際の評価指標に適用することとした。代表的な知見を以下に示す。

- タイトルに出現する単語は 575 に採用しやすい
- アブストラクトに出現する単語は 575 に採用しやすい
- 係り受け関係が適切な 575 は論文要約 575 として採用しやすい

^{*5} <http://www.wiss.org/WISS2016Proceedings/oral/25.pdf>

^{*6} <http://www.wiss.org/WISS2016Proceedings/oral/09.pdf>

^{*7} <http://www.wiss.org/WISS2016Proceedings/oral/10.pdf>

第 6 章 提案手法

6.1 要件

論文を 575 の形式で表現するにあたって、「内容的に正しい 575」と「面白い 575」の 2 つの方向性があると考えられる。本研究では内容的に正しい 575、つまり学術論文を要約することを重視して取り組むこととする。

ここで、提案手法の要件を再掲する。提案手法の要件は、「少ない文章量で論文を表現できる」、「印象に残りやすい」である。

6.1.1 少ない文章量で論文を表現できる

提案手法は、音韻的に読みやすい形式で要約文を生成するため、手元にある論文を読むべきかどうかスクリーニングする際や、論文の概要を把握したい場面で役に立つことが期待される。そのため、タイトルやアブストラクトよりも少ない文章量で学術論文を要約する必要がある。

6.1.2 印象に残りやすい

馴染みのある形式 (五七調) で論文を表現することには、2.1 節にて述べた 575 の持つメリットによって論文を印象づけて記憶に残す効果があるため、読み返したい論文がある場面や概要を想起したい場面でも役に立つことが期待される。そのため、印象に残りやすい 575 の生成を必要とする。

6.2 概要

提案手法では、学術論文 (和文) の PDF ファイルを一本入力すると、

1. PDF ファイルを全文テキスト化
2. テキストの前処理を行わずに特徴語を抽出
3. 特徴語から 575 形式のキャッチフレーズを生成

してそれを出力する。本研究で提案する部分は、2 と 3 であり、具体的には 3 ステップを踏む (図 6.1)。

- 特徴語の抽出
- 575 テンプレートに特徴語を割り当てて 575 の候補を生成
- 575 の候補を合議制スコアリングによって評価し、ランク付け

以下、各ステップについて説明する。

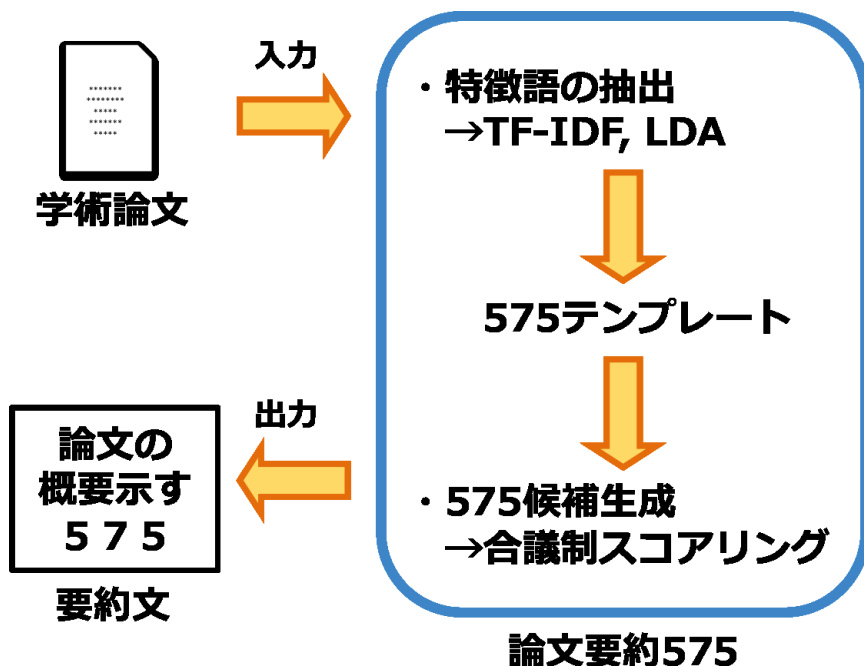


図 6.1: 学術論文の 575 を生成するまでのステップ

6.3 特徴語の抽出

6.3.1 概要

提案手法 (論文要約 575) に入力された学術論文の特徴語の決定は、以下に示す特徴語抽出手法を用いて文中の名詞をスコアの高い順にランク付けすることで行う。本研究の特徴語抽出手法では、「文の表層情報に基づく手法」と「文の潜在情報に基づく手法」を組み合わせた手法を用いた。

6.3.2 文の表層情報に基づく手法

文の表層情報に基づく手法では、単語の出現頻度に基づく単語重要度を用いた。特に TF-IDF (式 (6.3)) を用い、TF には入力する論文における単語出現頻度 (式 (6.1)) を、IDF には論文全文をピリオドで区切って抽出される文の集合を文書集合としたときの単語の文出現頻度 (式 (6.2)) を用いた。これは、予備実験にて人手で 575 を作成したところ、論文中の高頻度語が 575 に使われることが多かったためである。

式 (6.1) の $n_{w,d}$ は、入力論文 d における単語 w の出現回数、 $\sum_{s \in d} n_{s,d}$ は入力論文 d の単語総数である。式 (6.2) の N_d は、入力論文 d に存在する文の数、 $DF_{w,d}$ は論文 d において単語 w の存在する文の数である。TF-IDF の適用により、各単語の重要度がスコアとして得られ、スコアの高い単語ほど重要となる。

$$TF_{w,d} = \frac{n_{w,d}}{\sum_{s \in d} n_{s,d}} \quad (6.1)$$

$$IDF_{w,d} = \log \frac{N_d}{DF_{w,d}} \quad (6.2)$$

$$TF-IDF_{w,d} = TF_{w,d} \times IDF_{w,d} \quad (6.3)$$

6.3.3 文の潜在情報に基づく手法

文の潜在情報に基づく手法では, 文書に潜在する意味に基づく単語重要度を用いた. 特に, トピックモデリングの LDA を採用し, 入力論文の持つトピック (意味) を表現する名詞の重要度を求めるために用いた. LDA において, 潜在トピックの推定手法には変分ベイズ推定 (Variational Bayesian Estimation, VBE) [23] を用い, トピック数は階層ディリクレ過程 (Hierarchical Dirichlet Process, HDP) [24] を用いることで無限化した. 文書集合には, 言語処理学会論文誌 L^AT_EX コーパス^{*1} (565 本) を用い, これに入力する論文を加えたものを文書集合とした.

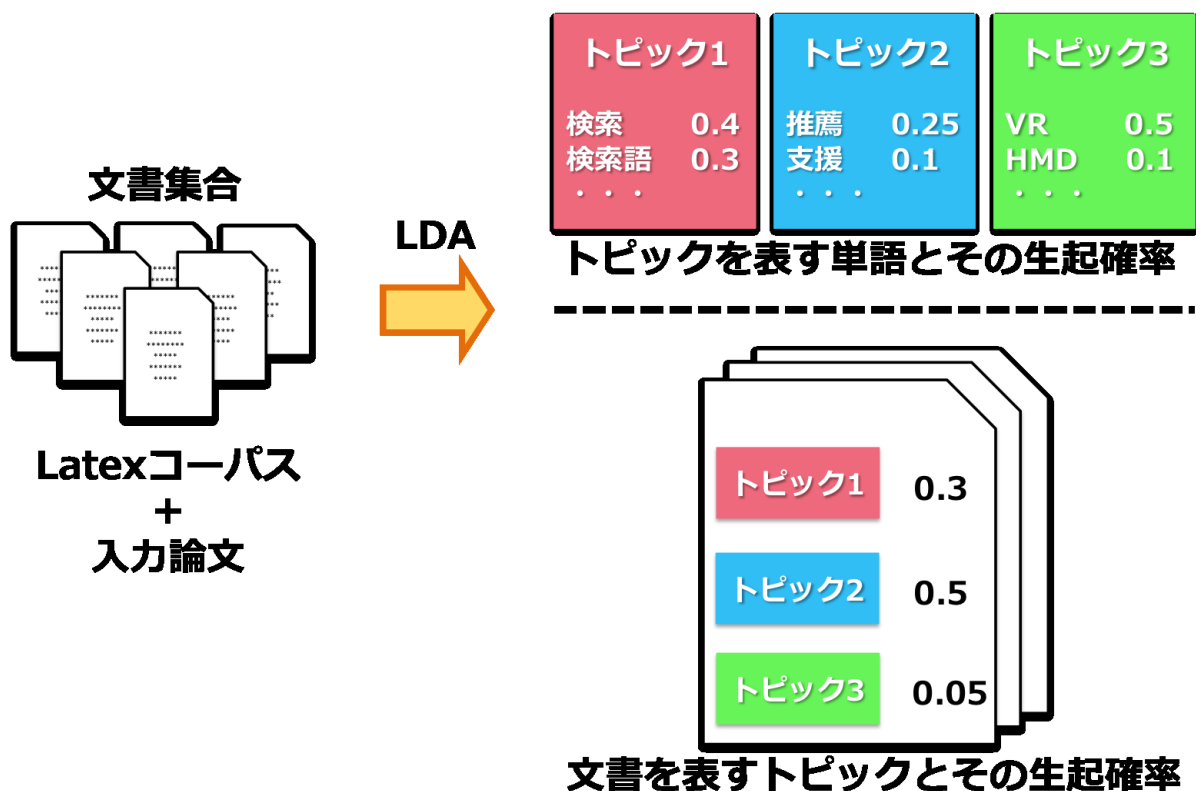


図 6.2: LDA によるトピック分布と単語分布の生成例

LDA では, 1 つの文書に複数のトピックが存在し, 1 つの単語が複数のトピックに存在しうる (図 6.2). そのため, LDA の適用によって

1. 各潜在トピックを表す単語が生起確率とともに得られる (図 6.2 の「トピックを表す単語とその生起確率」)

^{*1} http://www.anlp.jp/resource/journal_latex/

2. 各論文が持つ潜在トピックが生起確率とともに得られる (図 6.2 の「文書を表すトピックとその生起確率」)

本研究では, 2. において入力論文のトピックを占める割合が最も大きいトピックを入力論文の持つトピックとし, そのトピックにおける各単語の生起確率を重要度とした.

6.3.4 unigram-rescaling

文の表層情報と潜在情報を組み合わせた手法には, unigram-rescaling [25] を適用した. unigram-rescaling では, 式 (6.4) によって表層情報 u に基づいた単語 w の生起確率が, 同時に潜在トピック z が与えられたときの生起確率に調整される. これによって, 単語の出現頻度に入力論文の持つトピックも考慮したスコアが得られる.

$$p(w|u, z) \propto p(w|u) \left(\frac{p(w|z)}{p(w)} \right)^\beta \quad (6.4)$$

u は, 単語 w より前に現れる単語であり, $p(w|u)$ は何単語まで考慮するかによって変動する N -gram 確率である (一般に $N \geq 2$). 提案手法では, 一般的な N -gram 確率 ($N \geq 2$) を使用せずに TF-IDF のスコアをそのまま用いることとした. そのため, $N = 1$ の unigram 確率 (語順を考慮しない単純な単語の出現頻度) にしたことと同義である. $p(w)$ は単語 w の uni-gram 確率であるため, 入力する論文における出現頻度をそのまま適用した. z は潜在トピックであり, $p(w|z)$ はトピックにおける単語 w の生起確率である. 提案手法では, LDA によって得られる入力文書における最大占有率のトピックの単語分布を適用した. β はパラメータである.

ここで, パラメータ β の重み決定アルゴリズムを述べる. β の重みは, 以下の手続きによって最適値を決定した.

1. WISS2016 の学術論文からランダムに 5 本の論文を選出し, 各論文について 2. ~ 4. を繰り返す
2. 1 本の論文について, β の値を -1.0 から 1.0 の範囲で 0.1 刻みで変化させながら 3. を繰り返す
3. unigram-rescaling によって導出した特徴語の上位 20 個が論文のタイトルおよびアブストラクトに含まれている数 (最大 40 個) をポイントとする
4. ポイントの大きい順に β の値に順位をつける (最高 1 位, 最低 20 位)
5. それぞれの順位になった回数を用いて $nDCG_{20}$ (Normalized Discounted Cumulated Gain) [26] で β の各値にスコアを振る
6. スコアが 1 に最も近い値を β の最適値とする

具体的な手順を Algorithm1 に示す.

ここで, $nDCG$ について説明する. $nDCG$ とは, 情報検索システムが生成するランキングの良し悪しを評価する際に用いられる評価指標である. $nDCG$ はランキングが入力されると最大で 1 の値を返し, 値が 1 に近いほど入力したランキングが良いとされ, 式 (6.7) で表される. 式 (6.5) の R は関連度であり, 結果が上位であることの妥当性を示す. n は何位までのランキングを考慮するかを示す.

$$DCG_n = R_1 + \sum_{i=2}^n \frac{R_i}{\log_2 i} \quad (6.5)$$

$$idealDCG_n = \text{ランキングの最大 (理想) の } DCG_n \quad (6.6)$$

Algorithm 1 パラメータ β の重み決定アルゴリズム**Input:** Documents D (82 academic articles on the WISS 2016)**Output:** β

```

1:  $D \leftarrow \text{randomSelect5Papers}(D)$ 
2:  $SCORE \leftarrow \text{empty Key-Value Store}$ 
3:  $RANKING\_OF\_ \beta \leftarrow \text{empty List}$ 
4: for  $i = 0$  to  $5$  do
5:   for all  $j$  such that  $-1.0 \leq j \leq 1.0$  do
6:      $score \leftarrow \text{top20Words}(\text{unigramRescaling}(\beta \leftarrow j, \text{Document} \leftarrow D_i))$  in title and abstract
7:      $SCORE[j] \leftarrow score$ 
8:      $j \leftarrow j + 0.1$ 
9:   end for
10:   $RANKING\_OF\_ \beta \leftarrow \text{doRanking}(SCORE)$ 
11: end for
12:  $SCORES\_OF\_ \beta \leftarrow {}_nDCG_{20}(\text{relevance} \leftarrow RANKING\_OF\_ \beta)$ 
13:  $\beta \leftarrow \text{highScore}(SCORES\_OF\_ \beta)$ 
14: return  $\beta$ 

```

$${}_nDCG_n = \frac{DCG_n}{idealDCG_n} \quad (6.7)$$

例えば, ある情報検索システムで X について検索したとき, 生成されたランキングの上位 5 件のスコアが $[6, 2, 0, 3, 3]$ だったとする. このスコアは, 検索クエリ X に対する結果の関連度であり, 数値が大きいほど結果の質が良い. この検索システムの良し悪しを $nDCG$ で測ると, 式 (6.8) のようになる.

$${}_nDCG_5 = \frac{6 + \frac{2}{\log_2 2} + \frac{0}{\log_2 3} + \frac{3}{\log_2 4} + \frac{3}{\log_2 5}}{6 + \frac{3}{\log_2 2} + \frac{3}{\log_2 3} + \frac{2}{\log_2 4} + \frac{0}{\log_2 5}} \simeq 0.823 \quad (6.8)$$

上記の手続き (Algorithm1) では, β の成る順位の回数が重複を含み, $nDCG$ の計算に必要な関連度スコア (Algorithm1 の 12 行目の *relevance*) を, 1 位から 20 位のなかで成った順位の回数とみなせるため, この評価指標を用いた.

探索の結果として β の値と $nDCG$ のスコアを表 6.1 に示す.

表 6.1 より, パラメータ β の重みを -0.3 とした. 例として, 実際に決定した重みで学术论文から特徴語を抽出した結果を表 6.2 示す.

表 6.1: パラメータ β の最適値決定のための $nDCG_{20}$ の結果

β	$nDCG_{20}$
-1.0	0.443
-0.9	0.803
-0.8	0.803
-0.7	0.741
-0.6	0.894
-0.5	0.862
-0.3	0.917
-0.2	0.911
-0.1	0.762
0	0.911
0.1	0.785
0.2	0.817
0.3	0.642
0.4	0.714
0.5	0.554
0.6	0.599
0.7	0.572
0.8	0.491
0.9	0.566
1.0	0.493

6.4 575 候補の生成

Rafal らの研究 [14] では 312 個の俳句を読み, 代表的な文法テンプレートを人手で決定し, 抽出した特徴語をテンプレートに当てはめることで俳句を生成した. これを参考に, 本研究ではオリジナルの 575 テンプレートに特徴語を割りあてる方式を採用することとした.

575 テンプレートとは, 論文から抽出する特徴語を当てはめることで完成する 575 の骨組みであり, 予備実験で準備した学術論文 82 本分の 575 を用いて作成した. 具体的には以下の流れで 575 テンプレートを作成した.

1. 手製 575 のひとつひとつに形態素解析を行い, 各単語に品詞を振る
2. 名詞部分を抽出した特徴語と置換するための記号に変換する
3. 助詞や接続詞といった機能語部分についてはそのまま機能語を配置する

以上の手順で 82 個のテンプレートを自動で生成した後, 代表的なテンプレートを 20 個人手で選出した. 作成した 575 テンプレートの例をいくつか表 6.3 に示す. 提案手法で用いた全ての 575 テンプレートは付録 A に掲載する. 予備実験 5.1 において, 特徴語抽出には名詞のみを用いることとしたため, また, 575 テンプレートの内容語のほとんどが名詞だったため, 置換記号は名詞部分のみ変換する.

表 6.2: 提案特徴語抽出手法で学術論文から抽出した特徴語上位 20 個

貼るだけ探索:RFID タグの 検出履歴を利用した物探し 支援システム	マイクロ生体認証:人間の 微細生体領域を利用した生 体認証	全天球ボールカメラによる 撮影動画の視点固定手法の 提案
擬似	生体	天球
タグ	認証	カメラ
RFID	画像	回転
距離	箇	撮影
物	所	動画
Tag	利用	ボール
リーダー	肌理	姿勢
貼付	マイクロ	固定
検出	部位	高緯度
位置	要求	視点
スキャン	情報	球面
写真	ユーザ	緯度
パッシブ	撮影	フレーム
目的	テンプレート	画像
ユーザ	システム	座標
無向	マーク	特徴
電波	証	図法
用品	微細	距円筒
移動	可能	アルゴリズム
システム	マッチング	領域

表 6.3: 作成した 575 テンプレートの例

575 テンプレート		
<unk><unk>	<unk><unk> で	<unk> する
<unk><unk> で	<unk><unk>	<unk> する
<unk>	<unk><unk> の	<unk><unk>

575 テンプレートの <unk> 部分は, 自然言語処理の領域において未知語を表す際に常用されるプレースホルダーである. 575 生成過程では, <unk> が名詞に置換され, <unk> が連続する箇所には名詞が連続して配置される.

6.3 節で述べた特徴語抽出により, 論文から単語スコアの高い上位 20 個の名詞を抽出する. これを用意した 20 個の 575 テンプレートに対して割り当て, すべての組み合わせ (575 候補) を生成した.

575 候補の集合のサイズは, ${}_{20}P_6 + {}_{20}P_5 \times 12 + {}_{20}P_4 \times 6 + {}_{20}P_3 = 50937480$ である. なお, 575 テンプレート 20 個のうち, 名詞を 6 個含むテンプレートが 1 つ, 5 個含むものが 12, 4 個含むものが 6, そして 3 個含むテンプレートが 1 つ存在する.

6.5 合議制スコアリングによる 575 の決定

6.5.1 概要

生成した 575 候補の中から最終的な出力を決めるために各々に対して評価を行い, 最も適切なものを論文要約 575 の出力とした. 575 候補の評価は, 以下に示す合議制スコアリングによって行った. 本研究の合議制スコアリングでは, 次の 4 点について合議をとる.

- タイトルとの対応
- アブストラクトとの対応
- 係り受け関係
- ポジティブ度・ネガティブ度

この 4 点は, 予備実験にて学術論文から 575 を人手で作成した際, より良い 575 を決定するために筆者が重視した点を経験則として合議制スコアリングに反映させたものである. 具体的には以下の尺度で各 575 を評価する.

- タイトルとの cos 類似度
- アブストラクトとの cos 類似度
- 上五・中七・下五の各句の係り受けスコア
- 575 に含まれる単語のポジティブ・ネガティブスコア (ポジネガスコア)

6.5.2 タイトルとの cos 類似度, アブストラクトとの cos 類似度

cos 類似度におけるベクトルの計算には, 単語の出現回数を用いた. cos 類似度とは, ベクトル空間における 2 つの非ゼロベクトルの内積を計算することで類似性を測る尺度であり, 値が 1 に近いほど双方のベクトル (文書) は似ているとされる (式 (6.9))[27]. 合議制スコアリングではそれぞれの cos 類似度において, \vec{D} をタイトルもしくはアブストラクト, \vec{Q} を論文 575 としてそれぞれの類似度を計算した.

$$Sim(\vec{D}, \vec{Q}) = \sum_{t_i \in Q, D} w_{t_i Q} \cdot w_{t_i D} \quad (6.9)$$

6.5.3 上五・中七・下五の各句の係り受けスコア

係り受けスコアの導出には, 係り受け解析器の CaboCha^{*2} を用い, 575 中の係り受け関係を上五・中七・下五のそれぞれがどれくらい係りあっているかのスコアで取得し, その総和をスコアとした. 係り受け解析とは, 文章を単語に分ち書きしたあとに, 小さい文節 (名詞 + の (助詞) 等) をつくり, 各文節の修飾関係を解析する仕組みであり, 係り具合をスコアで得られる. 係り受け解析器では単語レベルで係り具合を調べているため, 同じ品詞構造でもスコアが異なる場合がある.

^{*2} <http://taku910.github.io/cabocha/>

6.5.4 575 に含まれる単語のポジティブ・ネガティブスコア (ポジネガスコア)

ポジネガスコアの導出には, 2005 年に高村らの研究によって生成された単語感情極性対応表^{*3}を用いて各単語にスコアを振り, その合計をスコアとした. 単語感情極性対応表には 55,000 語の単語が収録されており, 各単語には -1 から 1 までの数値が振られている. なお, 単語感情極性対応表に存在しない単語についてはスコア 1 を振り, 単語頻度が多いものは 1 に近く, 少ないものは 0 に近づくように出現頻度によって階段をつけるようにした.

6.5.5 合議制スコア

上記 4 つの尺度 (6.5.2 節, 6.5.3 節, 6.5.4 節) を組み入れた式 (6.10) による合議制スコア (式 (6.10)) の高い順に, 各 575 をランク付けし, 最終的なひとつを決定する.

$$\begin{aligned} \text{合議制スコア} = & \alpha * \text{タイトルとの cos 類似度} \\ & + \beta * \text{アブストラクトとの cos 類似度} \\ & + \gamma * \text{係り受けスコア} \\ & + \omega * \text{ポジネガスコア} \end{aligned} \quad (6.10)$$

式 (6.10) の $\alpha, \beta, \gamma, \omega$ はパラメータである.

6.5.6 合議制スコアのパラメータ $\alpha, \beta, \gamma, \omega$ の重みの決定

各パラメータの重み決定アルゴリズムを述べる. 合議制スコアのパラメータは, ひとつの重みが -1.0 から 1.0 までの範囲の値をとるとし, ランダムサーチによる機械学習で最適な組み合わせを見つけた. ランダムサーチのコスト関数には, 文書要約システムの要約率を測る際に用いられる ROUGE-N [28] と呼ばれる評価指標を用いて計算するシステム 575 (システムが生成する 575) と参照 575 (人手で作成した 575) の N-gram 類似度 (式 (6.11), 式 (6.12)) を F 値 (式 (6.13)) で導出した値を用い, このコストを最大化するパラメータの組み合わせを探索した. $\text{unigram}(C)$ はシステム 575 に含まれる単語集合であり, $\text{unigram}(R)$ は参照 575 に含まれる単語集合である. $\text{Count}_{\text{match}}(e)$ はシステム 575 と参照 575 で一致する単語を数える関数であり, $\text{Count}(e)$ は unigram の出現回数を数える関数である.

ROUGE-N では, 単語の N-gram 一致率を用いて 2 つの文章がどれだけ類似しているかを測り, 特に人手で作成した要約 (参照要約) とシステムで生成した要約 (システム要約) の類似度を測る. 本研究では, ROUGE-N のなかでも ROUGE-1, つまり unigram (1 単語) の一致率を測ることとしたため, 式 (6.11) では, 「システム 575 が参照 575 に含まれる単語をどれだけ当てられたか」を表しており, また, 式 (6.12) では「システム 575 の単語が参照 575 にどれだけ含まれているか」を表す. 一般的に, Recall と Precision はトレードオフの関係にあり, 適切な落とし所を探す必要があるため, 双方の調和平均を式 (6.13) により求めることでバランスのとれた所 (F 値) を求めた.

$$\text{ROUGE}_1(\text{Recall}) = \frac{\sum_{e \in \text{unigram}(C)} \text{Count}_{\text{match}}(e)}{\sum_{e \in \text{unigram}(R)} \text{Count}(e)} \quad (6.11)$$

^{*3} http://www.lr.pi.titech.ac.jp/takamura/pndic_en.html

$$ROUGE_1(Precision) = \frac{\sum_{e \in \text{unigram}(C)} \text{Count}_{\text{match}}(e)}{\sum_{e \in \text{unigram}(C)} \text{Count}(e)} \quad (6.12)$$

$$F_{\text{value}} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (6.13)$$

ランダムサーチは以下の手続きで行った.

1. WISS2016 の論文からランダムに選択した 10 個の論文をデータセットとする
2. 5 個の論文それぞれを 3. ～ 6. の処理にかける
3. 1 個の論文に対してランダムサーチとしてパラメータの重み組み合わせを 1000 個作り (iteration = 1000), 4. ～ 5. を繰り返す
4. 重み組み合わせを合議制スコアに適用し, 575 候補を合議制スコアの高い順に並び替える
5. 575 候補上位 20 件それぞれの F 値を求め, 総和を当該パラメータのコストとし, 前回のコストよりも高ければコストと重み組み合わせを更新する
6. 最高コスト (F 値の合計が最高) のパラメータの重み組み合わせを保持する

具体的な手順を Algorithm2 に示す.

パラメータ探索によって, 5 個の論文から 5 つのパラメータの重み組み合わせが得られた. 表 6.4 に示す.

表 6.4: ランダムサーチで得られたパラメータの重み組み合わせ 5 つ

	α	β	γ	ω
組み合わせ 1	0.9	-0.8	0.6	0.0
組み合わせ 2	-0.9	0.8	-0.1	0.1
組み合わせ 3	-0.8	0.1	0.2	-0.6
組み合わせ 4	-0.8	0.8	-0.1	0.6
組み合わせ 5	0.6	0.7	0.3	-0.1

得られた 5 つのパラメータの重み組み合わせのうち, どれが最適かを決定するためにバリデーションを行った. 具体的には,

1. データセットの残り 5 つの論文ひとつひとつに 5 つの重みの組み合わせを全て適用する
2. 各論文の 575 候補を合議制スコアの高い順に並び替える
3. 上位 20 件に $ROUGE_1$ を適用したときに得られる F 値の和を求める
4. 各論文において F 値の和が最も高くなった回数が最多のパラメータの重み組み合わせを最適解とする

結果を表 6.5 に示す.

表 6.5 より, F 値の和が最も高くなった回数が最も多かったのは, 組み合わせ 5 の [0.6, 0.7, 0.3, -0.1] となった. したがって, 合議制スコアリングの各パラメータの重みを $\alpha = 0.6$, $\beta = 0.7$, $\gamma = 0.3$, $\omega = -0.1$ とした.

Algorithm 2 ランダムサーチを用いたパラメータ a, β, γ, ω の重み決定アルゴリズム**Input:** Data Sets D (Random selected 10 articles from 82 academic articles on the WISS 2016)**Output:** Parameter Sets P

```

1:  $D \leftarrow \text{randomSelect5Papers}(D)$ 
2:  $P \leftarrow \text{empty List}$ 
3:  $\text{DOMAIN\_OF\_PARAMETER} \leftarrow [-1.0, 1.0]$ 
4: for  $i = 0$  to  $5$  do
5:    $\text{BEST} \leftarrow 0$ 
6:    $\text{BEST\_PARAMETER} \leftarrow \text{null}$ 
7:   for  $j = 0$  to  $1000$  do
8:      $\text{PARAMETER\_SET} \leftarrow \text{randomlyGenerate4Parameters}(\text{domain} \leftarrow \text{DOMAIN\_OF\_PARAMETER})$ 
9:      $\text{TOP\_20\_PAPER575} \leftarrow \text{top20Paper575}(\text{sort}(\text{consultationAlgorithm}(\text{document} \leftarrow D_i, \text{parameters} \leftarrow \text{PARAMETER\_SET})))$ 
10:     $\text{COST} \leftarrow 0$ 
11:    for  $k = 0$  to  $20$  do
12:       $\text{PAPER575} \leftarrow \text{TOP\_20\_PAPER575}_k$ 
13:       $\text{F\_VALUE} \leftarrow \text{calculateFValue}(\text{PAPER575})$ 
14:       $\text{COST} \leftarrow \text{COST} + \text{F\_VALUE}$ 
15:    end for
16:    if  $\text{BEST} < \text{COST}$  then
17:       $\text{BEST} \leftarrow \text{COST}$ 
18:       $\text{BEST\_PARAMETER} \leftarrow \text{PARAMETER\_SET}$ 
19:    end if
20:  end for
21:   $P[i] \leftarrow \text{BEST\_PARAMETER}$ 
22: end for
23: return  $P$ 

```

表 6.5: パラメータの重み組み合わせを他論文に適用したときに F 値の和が最も高くなった回数

	回数
組み合わせ 1	1
組み合わせ 2	0
組み合わせ 3	1
組み合わせ 4	1
組み合わせ 5	2

第 7 章 実装

7.1 準備

提案手法に入力される論文の扱いについて説明する。入力された論文は、

1. PDF ファイルをテキストファイルへ自動変換
2. テキストの前処理を行わずに全文をそのまま文字列として保持
3. 保持した文字列は特徴語を抽出するために使用し、抽出した特徴語を用いて 575 を生成

7.2 提案システム: 論文要約 575

提案手法をシステムとして実装したため、その GUI と使い方を紹介する。

7.2.1 GUI



図 7.1: 提案システム: 論文要約 575 の動作例 (論文入力前)

提案システムは、スタンドアロン デスクトップ アプリケーションとして開発した (図 7.1). 提案手法のコア部分は Python で実装し, GUI 部分は Python + JavaScript で実装した.

7.2.2 使い方

デスクトップで本アプリケーションを起動すると, 図 7.1 のような画面が立ち上がる. 図 7.1 の①部分に学术论文の PDF ファイルをドラッグ & ドロップすると, 論文を読み込んで論文要約 575 の生成処理に入る.



図 7.2: 提案システム: 論文要約 575 の動作例 (論文入力後)

575 の生成処理が終わった後の画面は図 7.2 である. ②部分には, 読み込んだ PDF ファイルをテキスト化した全文をそのまま出力している. スクロールできるため, 論文の全文を読むことができる. ③部分には, テキストから抽出した特徴語を重要度の高いものから 20 個表示している. ④部分には, 特徴語から生成した論文要約 575 をスコアの高いものから 20 個降順表示している (スクロール可).

再度, 論文から 575 を生成する場合は①部分に別の PDF ファイルをドラッグ & ドロップする.

7.2.3 出力例

学術論文の入力に対する実際の出力例をいくつか示す.

貼るだけ探索：RFID タグの検出履歴を利用した物探し支援システムの提案 ^{*1}

→ 検出タグ 貼付の位置に リードする

マイクロ生体認証：人間の微細生体領域を利用した生体認証 ^{*2}

→ 認証に 画像と要求 利用する

全天球ボールカメラによる撮影動画の視点固定手法の提案 ^{*3}

→ 動画カメラ ボールと撮影 固定する

チーム対戦型の貢献度提示によりバランスのよい参加を促すチャットシステム ^{*4}

→ 貢献の 対戦に参加 掛け合わせ

生体情報を呈示する婚活イベント支援システム Anemone の開発 ^{*5}

→ 支援の活 イベント呈示で 婚相手

出力例を見ると, 適切なものもあれば単語の不自然なつながりが見受けられるものも存在する.

^{*1} <http://www.wiss.org/WISS2016Proceedings/oral/16.pdf>

^{*2} <http://www.wiss.org/WISS2016Proceedings/oral/05.pdf>

^{*3} <http://www.wiss.org/WISS2016Proceedings/oral/21.pdf>

^{*4} <http://www.wiss.org/WISS2016Proceedings/oral/15.pdf>

^{*5} <http://www.wiss.org/WISS2016Proceedings/oral/1-B01.pdf>

第 8 章 評価実験

提案手法によって生成する 575 (システム 575) は要件 (「少ない文章量で論文を表現できる」, 「印象にのこりやすい」) を満たしているかを確認するために, 3 つの実験を行った.

8.1 実験 1: 合議制スコアリングの妥当性評価

8.1.1 概要

システム 575 が要件を満たしているかの確認を行う前に, 合議制スコアリングの妥当性を評価する必要があると考えた. そこで, 本実験では合議制スコアによって順位付けをする 575 と, 順位付けをしない 575 を比較する実験を行った. 具体的には, 実験協力者である本学の学生 8 名にシステム 575 を「良い」と「悪い」のグループに分けてもらった.

8.1.2 方法

実験は, DEIM Forum 2017 ^{*1} よりランダムに選択した論文 1 本^{*2}を用いて行った.

まず, 選択した論文に対して提案手法により 575 候補を生成した. 次に, 575 候補を合議制スコアリングによって並び替えたときに得られる上位 5 つの「ソート済みシステム 575」と, 575 候補よりランダムに抽出した 5 つの「未ソートシステム 575」を用意した. 最後に, ソート済みシステム 575 と未ソートシステム 575 を混在させ, サイズ 10 のシステム 575 集合を作成した. 作成したシステム 575 集合を実験協力者に提示し, 575 を 5 個ずつ「良い」と「悪い」の 2 つのグループに分ける試行を行ってもらった.

本実験では, 良いグループにソート済みシステム 575 がどれだけ含まれているかによって, 合議制スコアリングの適切さ, つまり 575 候補を適切に順位付けできるかを確認する.

8.1.3 結果

結果は表 8.1 のとおりとなった.

表 8.1 より, 良いグループに選ばれたシステム 575 の数は, 提案した合議制スコアリングによってソートしたシステム 575 の方が多い結果となった.

本実験では, 実験協力者に良い評価と悪い評価の数がそれぞれ 5 個ずつになるようにグループ分けをしてももらったため, 結果の表 8.1 は, 各実験者がそれぞれのシステム 575 につけた得点 (最大 5 点) と見ることができ. そこで, 5 % の有意水準で t 検定を試みたが, 有意差は見られなかった. 回答に開きが生じた理由として,

^{*1} <http://db-event.jpn.org/deim2017/proceedings.html>

^{*2} <http://db-event.jpn.org/deim2017/papers/243.pdf>

表 8.1: ソート済みシステム 575 と未ソートシステム 575 の「良い」評価の数 (最大 5)

	ソート済みシステム 575	未ソートシステム 575
1 人目	1	4
2 人目	3	2
3 人目	2	3
4 人目	3	2
5 人目	4	1
6 人目	3	2
7 人目	5	0
8 人目	1	4
合計	22	18
平均	2.75	2.25

「良い」と「悪い」の基準は個人の嗜好に依存することが考えられる。しかし、合計得点のみで比較をするとソート済みシステム 575 の方が得点が高いため、提案する合議制スコアリングによって 575 候補を適切に順位付けできることとする。

8.2 実験 2: 論文を表現できるか確認

8.2.1 概要

実験 1 より、合議制スコアリングによって適切に 575 候補を順位付けできることを確認できたため、出力するシステム 575 が論文を表現できているかを確認する実験を 2 つ行った。

2 つ目の実験は、1 つ目の実験よりも狭い専門領域において内容の近しい論文を用いている点が異なる。

8.2.2 1 つ目：論文のアブストラクトと参照 575 (手製の論文 575) を結びつける問題 5 題, 論文のアブストラクトとシステム 575 を結びつける問題 5 題

8.2.2.1 方法

本学の実験協力者 7 名に論文のアブストラクトと参照 575 を結びつける問題と、論文のアブストラクトとシステム 575 を結びつける問題合計 10 題に回答してもらった。実験に使った学術論文は、WISS 2016 よりランダムに選出した 10 本の論文である。実験では 10 題の問題を混在させた状態で提示することで、回答者にはシステム 575 と参照 575 の問題のどちらについて回答しているかが不明なようにした。各問題の選択肢は別の論文から生成したものも含めて 5 つ用意した。

本実験によってシステム 575 の正解率が参照 575 と同等、もしくはそれ以上であれば論文を表現できていることとした。

8.2.2.2 結果

回答の結果、システム 575 と参照 575 の正解率がともに 100 % となった。システム 575 と参照 575 の正解率が同等であったため、提案手法により生成するシステム 575 で論文の概要を表現できることを確認した。

8.2.3 2 目：論文のアブストラクトとタイトルを結びつける問題 5 題, 論文のアブストラクトとシステム 575 を結びつける問題 5 題

8.2.3.1 方法

1 目の実験で扱った論文よりも狭い専門領域における内容の近い論文を提案手法に入力した場合に, システム 575 で各論文を表現できるかを確認した.

本学の実験協力者 8 名を半分に分け, 論文アブストラクトとタイトルを結びつける問題 5 つを前半の 4 名に, 後半 4 名には論文アブストラクトとシステム 575 を結びつける問題 5 つに回答してもらった. 実験には, DEIM Forum 2017 のセッション「レシピ (1, 2)」にある論文からランダムに選択した 5 本の論文を用い, 2 つの問題で同じ論文アブストラクトを扱った. 本セッションでは, どの論文も「レシピ」や「料理」, 「メニュー」をトピックとして扱っているため, 抽出される特徴語が似ていると考えられる. そのため, 細かな違いを表現できているかを評価するのにふさわしい文書が揃っていると考えた.

本実験では, 2 つの課題それぞれの正解率を比較することで, 内容の似た論文において提案手法で生成する 575 は論文の細かな違いを表現できるかを確認する.

8.2.3.2 結果

実験の結果, 提示した 5 本の論文におけるアブストラクトとタイトルを結びつける問題, アブストラクトとシステム 575 を結びつける問題の両方とも正解率が 100 % となった. したがって, 同じ専門領域における内容の近い学術論文についても, 提案手法で生成するシステム 575 は細かな違いを汲み取ったうえで論文概要を表現できることを確認した.

8.2.4 まとめ

システム 575 はタイトルと同等の役割を持ち, タイトルよりも少ない文章量で論文を表現できることを確認したため, タイトルよりも一覧性が高くなる結果につながった.

8.3 実験 3: 論文が印象に残りやすいか確認

8.3.1 概要

実験 2 では提案手法によって学術論文をタイトルよりも少ない文章量で表現できることを確認したため, 3 目の実験ではシステム 575 が印象に残りやすいかどうかを確認する実験を行った. 具体的には, 実験協力者である本学の学生 7 名に以下の問題に回答してもらった.

- 学術論文のアブストラクトに適切と思う 575 をシステム 575 と参照 575 から選択する問題 1 題
- システム 575 と参照 575 のそれぞれから感じる印象を評価するアンケート 2 つ

8.3.2 方法

学術論文のアブストラクトと、それに対するシステム 575 と参照 575 の 2 択を回答者に提示し、当該アブストラクトが印象に残りやすそう (記憶しやすそう) と感じる方を選択してもらった。このとき、回答者は選択肢のどちらがシステム 575 か不明なようにした。実験に使用した論文は、WISS 2016 よりランダムに選出した 10 本の論文であり、実験 2 とは別の論文である。論文の数だけ問題を用意した (10 題) が、回答者には適当に選択した 1 題のみ回答してもらうこととした。

回答後、選択した問題におけるシステム 575 と参照 575 を形容詞対の表に基づいて評価してもらった (Semantic Differential Method, SD 法)。この方法は、事象 (ここでは印象) の評価をするための測定法であり、Rafal らの研究においても、システムが生成する俳句の詩的な価値を評価するために用いられた。実際に使用した形容詞対の表を表 8.2 に示す。

表 8.2: 実験 3 に用いた形容詞対の表

	非常に	かなり	やや	どちらでもない	やや	かなり	非常に	
楽しくない								楽しい
落ち着かない								穏やか
窮屈な								ゆったりした
醜い								美しい
劣る								優れる
ごみごみした								整った
よそよそしい								親しみのある
浅い								深い
悪い								良い
複雑								簡素
不快								快適
ぼんやりした								はっきりした
乏しい								豊か
古い								新しい
くだらない								賢い

8.3.3 結果

実験の結果、7 人中 7 人がシステム 575 よりも参照 575 の方が印象に残りやすそうと回答した。また、SD 法により回答してもらった 575 の評価を基に、システム 575 と参照 575 の印象評価を行ったところ、7 段階のリッカート尺度 (最低 1 点, 最高 7 点) でシステム 575 は平均 4.28 点, 参照 575 は平均 4.97 点となり、双方は 1 % の有意水準で有意差があった ($p < 0.01$ in t-test)。結果を図 8.1 に示す。

以上より、提案手法で生成するシステム 575 は、人手で作成する参照 575 よりも印象に残りにくい結果となった。特に図 8.1 より、システム 575 は参照 575 より「劣り、複雑である」印象が強いことが見て取れる。これは、提案手法で生成する 575 には、音韻的な読みやすさはあるものの、単語の不自然なつながりが見受けられること

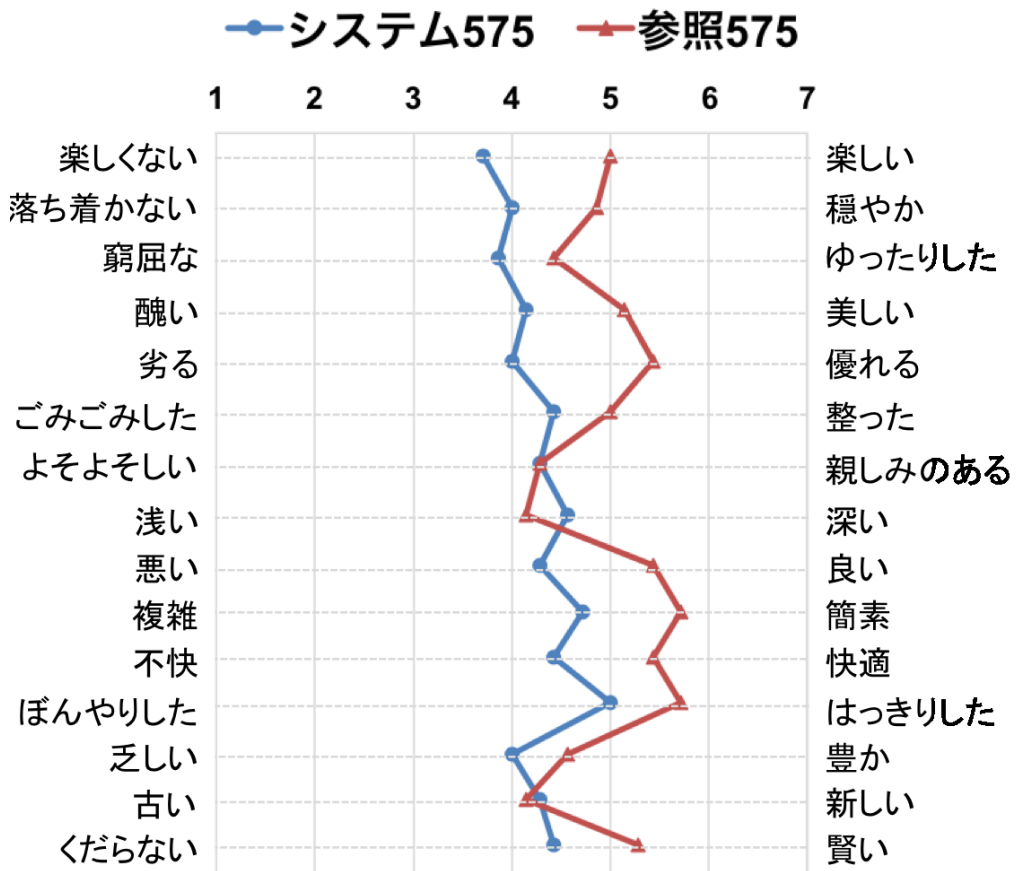


図 8.1: 形容詞対によるシステム 575 と参照 575 の印象の比較 (SD 法)

が原因ではないかと考えられる。

第9章 結論

9.1 まとめ

本研究では、学術論文を読むべきかどうか判断する手間と、論文の概要を思い出す手間を軽減することを目的に、575 形式のキャッチフレーズで論文の要約文を生成する手法を提案した。

本研究のアイデアは、古来より親しまれてきた俳句や川柳といった 575 の持つ音韻的な読みやすさを学術論文の要約文生成に適用すれば、論文を短く表現でき、かつ印象に残りやすくできるのではないかという点である。そのため、本研究は読者に対して論文を読むべきかどうか判断する手間と概要を思い出す手間を軽減することで貢献し、著者に対しては論文に付与するキャッチフレーズやキーワードの生成を代行、もしくは考案支援での貢献が期待される。

提案手法は、「特徴語抽出」と「575 候補生成」、「合議制スコアリング」の3つのモジュールから構成される。特徴語抽出では、文書の表層情報である TF-IDF と、文書の潜在情報であるトピックモデリングの LDA を組み合わせた計算尺度を提案し、情報検索システムを評価する際に用いられる手法を適用することで適切な特徴語を抽出するパラメータを探索した。575 候補生成では、ソフトウェア工学関連の学会論文集一冊分の 82 本の学術論文から人手で 575 を作り、それを解析することで論文 575 の骨組みであるテンプレートを作成した。そしてテンプレートに特徴語を配置することで候補を生成した。合議制スコアリングでは、人手で 575 を作成する工程で得た知見を基に、合議をとる形の計算尺度を設定し、機械学習を用いた最適化を行うことで適切に 575 をスコアリングできるパラメータを探索した。

提案手法を実際にシステム（論文要約 575）として実装したため、自動生成する 575 を用いた3つの評価実験によって論文 575 の持つ効果を確認した。一つ目は合議制スコアリングの妥当性評価、二つ目は少ない文章量で論文を表現できるかの確認、三つ目は印象に残りやすいかの確認である。

一つ目の実験では、生成する 575 候補から合議制スコアリングによって順位付けしたシステム 575 と、ランダムに抽出したシステム 575 を比較したところ、前者の方が適切なシステム 575 を決定・出力できることを確認した。二つ目の実験では、論文アブストラクトから参照 575 とシステム 575 を当てるユーザスタディ、および論文アブストラクトから論文タイトルとシステム 575 を当てるユーザスタディによって、提案手法で生成する論文 575 は少ない文章量で論文タイトルと同様に学術論文の概要を表現することが可能であることを確認した。三つ目の実験では、システム 575 と人手で作成した参照 575 から受ける印象を比較したところ、手製のものよりも劣る結果となり、印象に残りづらいことがわかった。

9.2 今後の課題

現在, 3つの課題が存在すると考える. それは, 論文 575 の質, 論文の利用方法, そして計算量である.

9.2.1 論文 575 の質

提案手法により出力される 575 は, 少ない文章量で論文タイトルと同様に学術論文の概要を表現することが可能であることを評価実験によって確認できたが, 印象に残りやすい効果はあまり確認できなかった. これは, システム 575 には音韻的な読みやすさはあるものの, 単語の不自然なつながりが含まれることによる影響と考える. そのため, 今後は合議制スコアリングによる評価の際に, 単語のつながりも加味した基準を考案することを課題とする.

9.2.2 論文の利用方法

現在は入力する学術論文の全文をそのまま文字列として利用しており, それとは別に, 論文のタイトルとアブストラクトを特徴語抽出のパラメータの重み決めと合議制スコアリングの際に利用している. そのため, 有力と思われる論文の文書構造を利用できていない. 今後は, 論文の持つ構造を活用した評価手法を考えることも課題とする.

9.2.3 計算量

現在, 575 候補の集合のサイズが 50937480 と膨大であるため, 計算量も膨大である問題点がある. そのため, 特徴語の上位何件を使うか検討し, 数を減らすことで特徴語と 575 テンプレートの組み合わせ数を削減させることを考えている.

9.3 展望

9.3.1 利用例

本研究のシステムが確立すれば, 手元にある学術論文を読むべきかどうかスクリーニングする際や, 読み返したい論文の概要を想起する際に役立つ. 他にも, どんな文章量の論文からでも同じ形式の文を生成するため, 学会の査読プロセスにて査読者が論文を一覧する場面や, 学会の Web ページで参加者が論文を一覧する場面でも役に立つことが期待される.

9.3.2 応用例

提案手法の応用先について述べる. 提案手法は 575 を生成するため, 17 音程度の文字列であれば空いた隙間に挿入されやすいだろうという考えより, Web 広告などで用いられるフレーズ (謳い文句) の生成に利用できることが期待される. 他にも, 論文 575 の結果から学術論文の書き方を指摘できると考える. 例えば, 綺麗な 575 が出力される論文は, 適切な文法で論文が書かれていると捉えることができるかもしれない.

謝辞

本研究は、電気通信大学 情報理工学研究科 情報・ネットワーク工学専攻 コンピュータサイエンスプログラムの寺田研究室において、寺田 実准教授のご指導のもと行われました。

寺田 実准教授には研究の仕方や論文の書き方をご教授いただいた以外にも、「物の見方」と「アイデアを文字なり絵なりソフトウェアなり形として表現し、周囲と共有することの大切さ」を教わりました。私の所属は変われどエンジニアリングを探究していくことには変わらないため、研究室生活で得たことは今後一生活かされる財産と思います。この成長は先生のご指導ご鞭撻のおかげです。厚く御礼申し上げます。

また、沼尾 雅之教授には論文構成のアドバイスをいただき、研究の主張点を整理することができました。心より御礼申し上げます。

そして、寺田研究室の学生みなさんには評価実験に協力していただいたり、研究にコメントをくださったり、輪講・輪読の円滑な進行に努めてくださったこと、本当に感謝しております。

参考文献

- [1] Shin Wonha 他. “セグメント構造を考慮した学術論文の包括的要約の自動生成の提案”. 言語処理学会第 23 回年次大会発表論文集, pp. 230–233, 2017.
- [2] 中須賀謙吾 他. “談話構造を利用した学術論文の自動要約生成”. 言語処理学会第 21 回年次大会発表論文集, pp. 569–572, 2015.
- [3] Amjad Abu-Jbara et al. “Coherent Citation Based Summarization of Scientific Papers”. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics, pp. 500–509, 2011.
- [4] GneÅ Erkan et al. “LexRank: Graph-based Lexical Centrality as Salience in Text Summarization”. Journal of the Artificial Intelligence Research 22, pp. 457–479, 2004.
- [5] 衣川和亮 他. “学術論文の章構造に基づくニューラル自動要約モデル”. 言語処理学会第 23 回年次大会発表論文集, pp. 150–153, 2017.
- [6] 麻生栄樹 他. “科学技術論文からの Structured Abstract の自動生成に関する研究”. 修士論文, 奈良先端科学技術大学院大学, 2016.
- [7] 安藤一秋 他. “論文概要からのタイトル自動生成の試み”. 言語処理学会第 10 回年次大会発表論文集, pp. A10C2–04, 2004.
- [8] 大部達也 他. “Recurrent Neural Network を用いた抽出型および生成型論文タイトル生成について”. The 31st Annual Conference of the Japanese Society for Artificial Intelligence, pp. 3A1-1, 2017.
- [9] Jan Wira Gotama Putra et al. “Automatic Title Generation in Scientific Articles for Authorship Assistance: A Summarization Approach”. Journal of the ICT Research and Applications, pp. 253–267, 2017.
- [10] Yutaka Matsuo et al. “Keyword Extraction from a Single Document using Word Co-occurrence Statistical Information”. Proceedings of the 16th International Florida Artificial Intelligence Research Society Conference, pp. 392–396, 2003.
- [11] 坂野信彦. “七五調の謎をとくー日本語リズム言原論”. 東京, 大修館書店, 2004.9, 273pp.
- [12] 渡部涼子 他. “五七調・七五調のリズム知覚に関する予備的研究”. 言語処理学会第 20 回年次大会発表論文集, pp. 574–577, 2014.
- [13] 越場千絵 他. “暗記学習のための語呂和歌作成支援システムの開発”. 卒業論文, 熊本大学, 2004.
- [14] Rafal Rzepka et al. “Haiku Generator That Reads Blogs and Illustrates Them with Sounds and Images”. Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, pp. 2496–2502, 2015.
- [15] Daniel Winterstein et al. “X575: writing rengas with web services”. Proceedings of the INLG 2016 Workshop on Computational Creativity in Natural Language Generation, pp. 27–30, 2016.
- [16] Yael Netzer et al. “Gaiku: Generating Haiku with Word Associations Norms”. Proceedings of the NAACL

- HLT Workshop on Computational Approaches to Linguistic Creativity, pp. 32–39, 2009.
- [17] 神澤義人 他. “読み句の構造に基づく郷土かるた読み句生成手法の提案”, 第9回データ工学と情報マネジメントに関するフォーラム論文集, F5–5, 2017.
- [18] Martin Tsan Wong et al. “Automatic Haiku Generation Using VSM”, Proceedings of the 7th WSEAS International Conference on APPLIED COMPUTER & APPLIED COMPUTATIONAL SCIENCE, pp. 318–323, 2008.
- [19] Xianchao Wu 他. “Haiku Generation Using Deep Neural Networks”, 言語処理学会第23回年次大会発表論文集, pp. 1133–1136, 2017.
- [20] 渡邊研斗 他. “大局的な構造を考慮した歌詞自動生成システムの提案”, 言語処理学会第20回年次大会発表論文集, pp. 694–697, 2014.
- [21] 山本貴史 他. “モーラ数と音節数を考慮した自然文から歌詞への変換”, 言語処理学会第15回年次大会発表論文集, pp. 168–171, 2009.
- [22] David M. Blei et al. “Latent Dirichlet Allocation”. Journal of the Machine Learning Research 3, pp. 993–1022, 2003.
- [23] Chong Wang et al. “Online Variational Inference for the Hierarchical Dirichlet Process”. Proceedings of the 14th International Conference on Artificial Intelligence and Statistics, pp. 752–760, 2011.
- [24] Yee Whye Teh et al. “Hierarchical Dirichlet Processes”. Journal of the American Statistical Association 101, pp. 1566–1581, 2006.
- [25] Daniel Gildea et al. “Topic-based language models using EM”. Proceedings of the 6th European Conference on Speech Communication and Technology, pp. 2167–2170, 1999.
- [26] Kalervo Jarvelin et al. “Cumulated Gain-Based Evaluation of IR Techniques”. Journal of the ACM Transactions on Information Systems, pp. 422–446, 2002.
- [27] Amit Singhal. “Modern Information Retrieval: A Brief Overview”. Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, pp. 35–43, 2001.
- [28] Chin-Yew Lin et al. “ROUGE: A Package for Automatic Evaluation of Summaries”. Proceedings of the Workshop on Text Summarization Branches Out, Post Conference Workshop of ACL, pp. 74–81, 2004.

付録

A. 提案手法に用いた 575 テンプレート一覧

575 テンプレート

<unk><unk> <unk><unk> で <unk> する
 <unk><unk> で <unk><unk> <unk> する
 <unk> <unk><unk> の <unk><unk>
 <unk><unk> <unk><unk> <unk> する
 <unk><unk> <unk><unk> で <unk> のみ
 <unk><unk> <unk><unk> に <unk>
 <unk><unk> <unk> と <unk> <unk> する
 <unk><unk> <unk> の <unk> で <unk>
 <unk><unk> <unk> の <unk> に <unk> する
 <unk><unk> に <unk><unk> <unk> する
 <unk><unk> の <unk><unk> <unk> する
 <unk> で <unk><unk> <unk><unk>
 <unk> に <unk><unk> を <unk> する
 <unk> に <unk> と <unk> <unk> する
 <unk> にて <unk><unk> <unk> する
 <unk> の <unk><unk> <unk> に
 <unk> の <unk><unk> を <unk> する
 <unk> の <unk> に <unk> 掛け合わせ
 <unk> の <unk> <unk><unk> で <unk><unk>
 <unk> を <unk><unk> で <unk> に

本研究に関する発表文献

1. 安部文紀, 寺田実. “論文要約 575: 575 の音韻的読みやすさを付与した学術論文の要約文生成”, 25th Workshop on Interactive Systems and Software (WISS 2017), 3-A01, 2017.
2. 安部文紀, 寺田実. “575 の音韻的読みやすさを付与した学術論文の要約文自動生成手法”, 第 10 回データ工学と情報マネジメントに関するフォーラム (第 16 回日本データベース学会年次大会), E3-1, 2018. (発表予定)