ARG WI2 No.10, 2017
WI2-2017-4

遺伝的アルゴリズムを用いた文書間類似度による小説要約システムの性能評価

今野勇気　　荒木健治

北海道大学大学院情報科学研究科

y-konno@eis.hokudai.ac.jp　　*araki@ist.hokudai.ac.jp.*

概要　現代では数多くの小説が存在し，また毎年多くの小説が発表されている．莫大な量の小説の中から，人々が読みたいと感じる小説を探す手助けになるように，小説の自動要約システムを提案する．本提案システムは，遺伝的アルゴリズムを用い，要約対象の小説とシステムが出力した要約文候補との文書間類似度の最大化を図ることにより，要約文を出力するものである．本稿では，提案手法を用いた小説の自動要約のシステム構築，評価実験，考察を行った結果について述べる．

キーワード　文書要約，遺伝的アルゴリズム

1 はじめに

電子書籍等の登場により，最近ではより多くの小説に触れる機会が増えている．しかし，小説を読む行為は多くの時間を要する．読者が好みの小説を探す手間を簡略化するために，小説の内容を反映した要約文を自動で生成するシステムの作成，性能評価を行う．

昨今，自動要約の必要性が高まり，様々な手法の自動要約システムの提案が行なわれている．中でも，重要文抽出を行い文書の要約を行う手法が代表的である．本研究では，この重要文抽出を行い，文書を要約するという手法を使用し，小説の要約を行う．

遺伝的アルゴリズムを使用して重要文を抽出し，要約を行う手法として小倉らの研究[1]がある．小倉らの研究では，単語の出現頻度，文の位置，JS ダイバージェンスからなる適応度関数を設定し，重要文抽出における組み合わせ最適化を多目的最適化問題とみなし，多目的遺伝的アルゴリズムを使用して文書要約を行っている．

また文書間の類似度を使用して要約を行う手法として住田らの研究[2]がある．住田らの研究では，元の文書と要約後の文書の言語概念ベクトルを作成し，ベクトル同士の類似度が最も高い要約文を探索する手順を整数線形計画問題として解いている．

本研究では，小説の要約を重要文抽出における組み合わせ最適化問題とみなし，遺伝的アルゴリズムを用いてこれを解く．またその際に，小説の内容を網羅した要約文の生成を目指し，集合間の類似度を表現する Dice 係数，Jaccard 係数を目的関数として用いる．これにより小説の内容を網羅した要約文の生成を行う．

2 システム概要

2.1 遺伝的アルゴリズム

本研究では，Dice 係数や Jaccard 係数といった目的関数を最大化する手法として遺伝的アルゴリズムを用いる．遺伝的アルゴリズムは，いくつかの個体を用意し，交雑や突然変異等の遺伝的操作をした後，適応度の高い順に次の世代の遺伝的操作を行い，適応度の高い個体を探索するアルゴリズムである．

本研究では，個体は小説の総文数の遺伝子を持ち，i 番目の遺伝子は小説の i 文目の文 s_i に対応する．また，i 番目の遺伝子が 1 の場合は s_i が要約に含まれ，i 番目の遺伝子が 0 の場合は s_i は要約に含まれない．この小説の総文数の遺伝子を持つ個体に対し，交叉，突然変異，淘汰の遺伝的操作を行う．交叉では，ランダムに親個体を 2 体選択し，親個体の遺伝子をランダムに受け継いだ子個体を生成する．突然変異では，ランダムに選択された個体に対し，ランダムに1箇所の遺伝子の 0,1 を反転させる．淘汰では，適応度の高い順に順位付けし，順位の高い 50 個体だけを残し，交叉から遺伝的操作をやり直す．適応度を算出する目的関数は Dice 係数（式(1)），Jaccard 係数（式(2)）を使用する．

$$Dice(X,Y) = 2 \times \frac{X \bigcap Y}{X + Y}. \tag{1}$$

$$Jaccard(X,Y) = \frac{X \bigcap Y}{X \bigcup Y}. \tag{2}$$

ここで，X は小説全体の単語集合，Y は遺伝子により選択された文集合の単語集合である．また，淘汰では，

正解データとする参照要約の文字数を上限とした文字数制限を行う. 参照要約の文字数を超えない個体のみに対し, 淘汰を行う.

2.2 処理過程

本研究のシステムの処理概要を図 1 に示す. 本システムは, 小説が入力された際, 前処理を行った後, 遺伝的アルゴリズムを使用し, 最終的に一番適応度の高かった個体をシステム要約として出力する. また, このシステムでの終了条件は, 500 世代のループである.

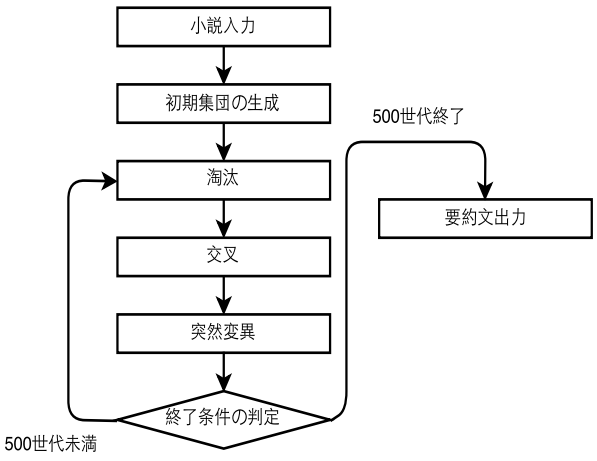

図 1 システム概要

3 評価実験

青空文庫[2]から収集した5作品に対して実験を行った. 参照要約として, Wikipedia に掲載されている各作品の「あらすじ」を使用した. 評価方法は ROUGE-1[3]を採用した. ROUGE-1 を式(3)で示す.

$$ROUGE-1(C,R)=\frac{\sum_{e\in unigram(C)} Count_{clip}(e)}{\sum_{e\in unigram(R)} Count(e)} \quad (3)$$

ここで, e∈unigram(C)は, システム要約に含まれる unigram . e ∈ unigram(R) は参照要約に含まれる unigram. Count(e)はある unigram の出現頻度を数える関数であり, $Count_{clip}(e)$は, システム要約に含まれる unigram のシステム要約における出現頻度と参照要約における出現頻度の小さい値を採用する.

表 1 評価実験の結果

	baseline	Dice	Jaccard
Text1	0.402	0.430	0.415
Text2	0.467	0.485	0.494
Text3	0.481	0.496	0.504
Text4	0.392	0.480	0.480
Text5	0.404	0.508	0.516
average	0.429	0.480	0.482

実験結果を表 1 に示す. また, 本実験では, ベースラインとして tf-idf のスコアが高い順に文を順位付けし, 参照要約の文字数を上限として, 順位が高い順に文を抜き出す手法を採用した.

ここで, 4 作品に対して, Jaccard 係数を目的関数とした遺伝的アルゴリズムを用いたシステムが優位な結果となった. また, 3 手法の 5 作品に対してのそれぞれの ROUGE 値の平均は, baseline に対し Dice 係数を用いたシステムで 0.051 ポイント, Jaccard 係数を用いたシステムで 0.053 ポイント上昇し, Jaccard 係数を用いたシステムが最も ROUGE 値が良い結果となった. baseline が tf-idf により重要語を設定し, 重要語が多く含んだ文を抽出し要約文を生成するのに対し, 提案手法では, 小説と要約文の単語の集合の類似度が最大になるような文集合をシステム要約として出力する. したがって, 要約において, 重要な単語を設定し要約文を生成する手法より, 要約元の文書と要約後の文書間類似度を最大にする手法の方が優れていることが確認された.

4 おわりに

本稿では, 小説の要約を自動で行うシステムを作成し, その性能を評価実験を行い評価を行った. その結果 Dice 係数を用いたシステムでは 0.051 ポイント, Jaccard 係数を用いたシステムでは 0.053 ポイントの性能向上を確認した. この結果から, 要約において重要文を設定し要約を行う手法より, 要約元の文書と要約後の文書の類似度の最大化を図る手法である本提案手法の優位性を確認することができた.

今後は, 目的関数を変更し, 要約に最適な目的関数を探し出す予定である. また, 本提案システムでは, 要約文の文法的な点を考慮していないので, 人間が読んで違和感のない, より実用的なシステムの実現を最終的な目標とする予定である.

参考文献

[1] 小倉由佳里, 小林一郎:多目的遺伝的アルゴリズムを用いた組み合わせ最適化による要約生成, 言語処理学会第 21 回年次大会発表論文集, pp. 585-588, 2015

[2] 住田恭平, 二宮崇:言語概念ベクトルを用いた文書間類似度に基づく複数文書自動要約, 言語処理学会第 22 回年次大会発表論文集, pp. 513-516, 2016.

[3] 青空文庫, http://www.aozora.gr.jp/.

[4] Lin, C.Y and Hovy, E.: Automatic evaluation of summaries using n-gram co-occurrence statistics, Proc. of the 4th Meeting of the North American Chapter of the Association for Computational Linguistics and Human Language Thechnology, pp. 150-157, 2003.