JAIST Repository
https://dspace.jaist.ac.jp/

Title	観点を反映した深層学習および強化学習による学術論文の自動要約生成
Author(s)	LI, Jinghong
Citation	
Issue Date	2021-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/17148
Rights	
Description	Supervisor:長谷川　忍，先端科学技術研究科，修士（情報科学）

JAIST
JAPAN
ADVANCED INSTITUTE OF
SCIENCE AND TECHNOLOGY

Japan Advanced Institute of Science and Technology

修士論文

観点を反映した深層学習および強化学習による学術論文の自動要約生成

LI JINGHONG

主指導教員 長谷川　忍

北陸先端科学技術大学院大学
先端科学技術研究科
情報科学

令和３年３月

Abstract

When researchers and students start a new research at the first step, they find some surveys and concentrate on the novelty of the research they want to work on. It is necessary to understand the elements and contents of research through a large amount of related-work surveys to get the information of the state-of-the-art technology. With the spread of the Internet, automatic summarization is a technology for grasping important information from massive data. Automatic summarization is a research project that automatically generates a short document that briefly describes the contents of a given document.

In recent years, a vast amount of academic papers have been to online and open resources. Collecting the essential information from them becomes an essential step in the initial stage of research activities. The contents in an academic paper reflect the viewpoint such as the background, purpose, method, experiment, evaluation, and conclusion. Catching the contents that reflects the viewpoints and recognize the critical sentences in the contents of each viewpoint can improve the effectiveness of research activities.

The purpose of this research is to develop a Viewpoint Refinement in Automatic Summarization (VPRAS) system for research articles that reflects the viewpoints such as the research background and purpose to support surveys by researchers and students. Since there is a limited dataset on the summary reflecting the viewpoints, we adopt machine learning techniques to classify sentences in the japanese article into the viewpoints. In addition to supervised machine learning, we introduced reinforcement learning and Dynamic Programming(DP) to extract the important sentence in each viewpoint. The agent automatically extracts summary sentences based on the reward function, to test the potentials of improving accuracy.

Extraction Summarization is regarded as a kind of document classification task. Chapter 3 introduces the method of text classification with viewpoints in our VPRAS model based on Deep-learning technology. We use the result of classification and apply reinforcement learning and DP(Dynamic Programming) to build a sentence extraction model to generate a summary. At the first step of building the dataset, we download academic articles of the Japanese language in PDF. Next, we use 'apache-tika' to recognize the texts in each PDF and make regular expressions in these texts to extract the body of text. The expert adds mainclass-label, subclass-label, and importance-label to each sentence in the main documents. The mainclass-labels are used in text classification by deep learning. The subclass-labels and importance-labels are used in text extraction. At the step of the Deep-learning model, we adopt the two methods of pre-training called Word2vec and PV-DM(Distributed Memory Model of Paragraph Vector) to

execute word-embedding which is one of the simplest deep learning techniques to build features that represent words, sentences, and documents. After acquiring the word-embedding vector in pre-training methods, Embedded words and sentences are inputted in the neutral network.

In the neutral network, we use Word2vec which reflects the feature of words as the input a classifier called LSTM(Long short-term memory) to execute text classification and use another classifier called SVM(Support vector machine) to classify the sentence-vector which embedded by PV-DM. In order to improve the recognition accuracy of text classification, we propose a combined-method that combines the advantage of Word2vec+LSTM and PV-DM+SVM. In combined-method, we acquire the result of each classifier to get the probability of each class and optimize these probabilities to do reclassification. In the classification by deep learning, there is a possibility that the error function does not decrease during the training process because of the different fields in the training article and test article. To solve this kind of problem, we adopt a function that configures with the probability of each class and cosine similarity to reclassification once again. We use the result of the final round of classification as the target sentences in the important sentence extraction model. At the step of text extraction, we calculate the value of each sentence by two methods. One is dependent on the importance-label, another one is dependent on both importance-label and subclass-label. Then, we calculate the cosine similarity between each sentence as a penalty to reduce information redundancy when extracting summary. Finally, we input the value of the sentence, the length of the sentence, the limited length of the summary, and the penalty of similarity into the knapsack-reinforcement learning model to extract the summary.

In the experiment of chapter 4, we conducted the simulation about the deep learning model with the pre-training method Word2vec and PV-DM. We also tested the effectiveness of the combined-method and reward function based on cosine similarity to verify our model's accuracy. In the experiment of reinforcement learning and DP, we also added the comparison model which only used the ranking of the value of sentences. In the part of the evaluation, we tested the recognition accuracy in importance-label, which were added by an expert, and calculated the Rouge-score of each summary. Finally, according to the result of the experiment, we discussed the feature of each method in our model and made an error analysis of them.

In chapter.5 of conclusion, we conclude what we did in this research and give the suggestion about how to revise our model to make a better recognition in the future work.

Keywords: Academic Paper, Automatic Summarization, ViewPoint, Deep Learning, Reinforcement Learning,Dynamic Programming

目 次

第 1 章　はじめに **1**
1.1 背景 1
1.2 目的 2
1.3 本論文の構成 2

第 2 章　関連研究 **3**
2.1 学術論文に対する自動要約 3
2.2 抽出要約の最近の手法 3

第 3 章　提案手法 **5**
3.1 概要 5
3.2 前処理 5
3.2.1 正規表現 7
3.2.2 サブクラスおよび重要度ラベル 7
3.3 深層学習 (分類) 9
3.3.1 事前学習 (Pre-training model) 9
3.3.2 Word2vec 9
3.3.3 Doc2Vec の PV-DM(Distributed Memory Model of Paragraph Vector) 11
3.3.4 分類器 12
3.3.5 Combined-method 12
3.4 COS 類似度による改良 13
3.5 重要文抽出 16
3.6 強化学習の要素 16
3.6.1 方法 1-重要度ラベルに基づく文章価値の推定 18
3.6.2 方法 2-サブクラスと重要度ラベルに基づく文章価値の推定 . 19
3.6.3 文章類似度 20
3.6.4 報酬の期待値 20
3.6.5 要約長の制限 21
3.6.6 組み合わせナップサック DP(動的計画法) 21
3.6.7 強化学習による重要文抽出 23
3.7 ユーザーインターフェイス 24

3.7.1 GUI 24
3.7.2 Highlight システム 25
3.8 開発環境 26

第 4 章 実験 **27**
4.1 実験の全体像 27
4.2 データセット 27
4.3 深層学習の実験設定 28
4.3.1 深層学習のパラメータ 29
4.4 深層学習による多値分類結果 30
4.5 強化学習による重要文抽出の実験設定 31
4.5.1 ROUGE-N 33
4.6 強化学習による重要文抽出の実験結果 34
4.6.1 論文内容の影響 41
4.6.2 強化学習+動的計画法の特徴 41

第 5 章 おわりに **43**
5.1 まとめ 43
5.2 今後の課題 44
5.2.1 データセット 44
5.2.2 正規表現 44
5.2.3 深層学習の分類部分 44
5.2.4 強化学習の抽出部分 44

図 目 次

3.1 VPRAS システム 5
3.2 前処理の全体像 6
3.3 サブクラス 8
3.4 データセットの一部 8
3.5 深層学習モデル 9
3.6 Word2vec(Skip-Gram) モデル 10
3.7 PV-DM モデル 11
3.8 Combined-method 13
3.9 Cos 類似度により報酬設定モデル 14
3.10 方法 1 の全体像 17
3.11 方法 2 の全体像 18
3.12 方法 1-文章価値の算定 18
3.13 方法 2-文章価値の算定 19
3.14 要約長の制限 21
3.15 強化学習流れ 23
3.16 GUI 画面 25
3.17 Highlight システム 26

4.1 Keras LSTM Summary 29
4.2 学習曲線 30
4.3 各メインクラスの再現率 34
4.4 方法 1-重要度ラベルとの一致度 35
4.5 方法 2-重要度ラベルとの一致度 35
4.6 ROUGE スコア 36
4.7 方法 1-研究背景 36
4.8 方法 2-研究背景 37
4.9 方法 1-研究目的 37
4.10 方法 2-研究目的 37
4.11 方法 1-研究方法 37
4.12 方法 2-研究方法 38
4.13 方法 1-実験 38
4.14 方法 2-実験 38

4.15 方法 1-結果評価 38
4.16 方法 2-結果評価 39
4.17 方法 1-知見 39
4.18 方法 2-知見 39
4.19 方法 1-関連研究 39
4.20 方法 2-関連研究 40
4.21 RL 動的計画法-正解の特徴 41
4.22 文章価値ランキング-正解の特徴 41

表 目 次

3.1 テキスト正規化処理 7
3.2 報酬の設定要素 15
3.3 強化学習による重要文抽出の流れ 24
3.4 GUI 操作 24
3.5 ライブラリーと機能 26

4.1 データセット明細 28
4.2 実験設定 28
4.3 深層学習における実験結果 30
4.4 Cos 類似度における実験結果 31
4.5 Cos 類似度+深層学習 Combined-method. 正解率表 31
4.6 訓練用データの明細 31
4.7 訓練用データの明細 31
4.8 テスト用データの明細 32
4.9 メインクラス (観点) 毎の抽出率 32

第1章　はじめに

1.1　背景

研究者や学生が新たな研究を始める際には, 自身が取り組みたい研究の新規性や全体像をある程度把握しておく必要があり，大量の関連研究のサーベイを通じて研究分野の最前線を理解することが不可欠である．つまり，研究のサーベイには，多くの学術論文を読み，その内容を理解する必要がある [1]．特に近年では膨大な量の学術論文がオンライン・オープン化されており，それらの中から必要な情報を収集・整理することは研究活動の初期段階において重要な役割を果たす．また，研究の進捗状況によって関連研究で注目すべき箇所が異なることもしばしばあり，学術論文を構成する背景や目的，方法，実験，評価などといったサーベイの観点を反映する情報を収集することは，効果的に研究を進める上で重要である [2]．

一方で，インターネットの普及によって膨大なテキストデータから重要な情報を把握するための技術として，自動要約が注目されている．自動要約とは与えられた複数の文書から，その内容を簡潔に表した短い文書を自動的に生成する研究課題である [3]．近年では，自動要約の研究の発展により，冗長な文書を短縮化することで情報の入手にかかる時間を大幅に短縮できるようになっている．その中でも，学術論文を対象とした自動要約は重要な研究課題の一つである．学術論文に対するテキスト自動要約に関する研究の多くはアブストラクトの生成が対象となっている [4]．なぜなら，サーベイの対象となる論文の全文を読むよりも，論文のアブストラクトをまず読み，その内容から選別された重要論文をより詳しく読むといった方法が効率であると考えられるためである [5]．こうした方法は，関連する研究領域の全体像を把握する上で効率的であるといえる．ただし，アブストラクトのみでは，学術論文の本文中にある様々な観点を反映した情報を収集することは難しい．関連研究に基づく背景や，研究方法としての利用される技術などといった観点に基づいて要約を行う場合には，アブストラクトにそれらの情報が含まれておらず，対応する要約文が既知でないため，自動生成は容易ではない．

1.2 目的

これらの問題に対し, 本稿では観点を反映した深層学習の分類をベースとし，強化学習による抽出要約文を作成するアプローチを提案する．本研究では，学術論文の主要な構成要素に関連する文章を要約として抽出することが目的である．これを実現するために，研究者や学生の学術論文サーベイに対する観点を反映した自動抽出要約を生成する VPRAS(ViewPoint Refinement in Automatic Summarization) システムを開発することを目指す．Cheng らは抽出要約のための教師あり機械学習の手法 [6] を提案しているが，観点を反映した自動要約を実現する際には，十分な規模のデータセットを準備することが困難である．そこで本研究では少数のデータセットを用意し，深層学習による分類手法と強化学習による重要文抽出手法を利用することで，小規模な教師データによる自動抽出要約タスクにおける精度向上を目指す．なお，初学者にとってはどういった部分を注目すべきかを理解させることがサーベイにとって重要である．そこで，要約とその周辺の情報を短時間で確認できるようにするため，要約内容を描画により強調させる Highlight 処理を入れた GUI(Graphical User Interface) を開発することで，ユーザに明示的に要約を提示できる情報収集支援環境を提供する．

1.3 本論文の構成

本論文の構成を以下に述べる.2 章では，本論文に関連する先行研究について述べる. 3 章では，本論文で提案する深層学習による論文観点分類手法，強化学習による重要文抽出，Highlight 処理および GUI 開発について述べる.4 章では，提案手法の各部分に対する評価実験として，データセットの構成，実験設定および実験結果と考察を報告する．最後に，5 章では本論文のまとめと今後の課題を述べる.

第2章　関連研究

単一文書に対する自動要約の研究は，文書全体の内容を代表する文章や語を抽出する抽出要約 (Extract Summarization) と，文書全体の内容を表す文章を新たに生成する抽象要約 (Abstract Summarization) に大別される [7]．本研究では学術論文の情報量をサーベイの観点を反映しながら削減することで，研究者や学生がより多くの関連研究を把握できるようにすることを目的としているため，主に抽出要約を対象とする研究を行う．

2.1　学術論文に対する自動要約

学術論文に対する自動要約の研究としては，Contractor らによる論文構造に注目した分類器を用いた抽出要約の生成が挙げられる [4]．ここでは文章情報を分類器でクラスタリングすることで冗長性の緩和を達成している点に特徴がある．鶴岡らは談話構造を利用した特徴量を用い，学術論文要約において談話構造が精度の向上に有用であるかを検証した [5]．平井らは論文の構成要素を Conditional Random Field (CRF) により，実験の図，表，段落に分類することで，実験情報を抽出する手法を提案し，実験情報を論文構成要素を利用して抽出する有効性を検証した [8]．

2.2　抽出要約の最近の手法

神谷らは，レビューから重要語抽出し，深層学習の LSTM 時系列ニュートラルネットワークを用い，要約を生成する実験を行った [9]．文書量に頼らず重要な内容を含めた要約の生成が期待できる．しかし，本研究で対象としている学術論文とは文章構造や観点が異なっている．梁と阿部川は自動要約問題を文圧縮に関する強化学習問題として定式化し，要約長として報酬を定義し，新聞記事要約を生成できることを示した [10]．しかしながら，要約長の制御のみを扱っており，要約の内容と可読性は保証されていない．また，Kong らは，文章の内容と位置をベクトル化し，候補文の選択を行動として学習することにより，深層強化学習による自動抽出要約を提案している [16]．しかしながら，本研究で対象とするサーベイの観点は扱われておらず，アブストラクトのような Gold Standard が存在していないため，そのまま適用することは困難である．奥村らは定式化には整数計画法

を用いる重要文抽出と文圧縮の両方を同時に行う新しい抽出的要約モデルを提案し，文中の冗長な表現を削除できることを示した [11]．しかしながら，本研究で要求される観点を代表する論文本文にある重要文が無視される可能性がある．泉田らは制約付き線形強化学習の最適化問題として定式化される [12]．しかしながら，複雑な大規模の文章要約の環境で報酬関数を設定することが困難であり，自動要約にそのまま適用することが困難である．

本研究では，学術論文におけるサーベイの観点を反映した自動抽出要約というタスクに対し，まず深層学習を使用することにより論文の本文に対応したいくつかの観点に分類する．そして，観点ごとの重要文を抽出するため，熟練者が付けたサブクラス及び各文章の重要度情報に基づく，重要度情報に要約長の制御の最適化手法を組み合わせた強化学習を行う点に新規性がある．加えて，ユーザにフレンドリな GUI として要約内容を強調する Highlight システムを作成する．こうしたシステムが実現できれば，研究活動の初期段階で重要な役割を持つサーベイの効果的な支援となることが期待される．

第3章　提案手法

3.1　概要

抽出要約は一種の文書分類課題とみなすことができる．その分類を自動的に実現するため，本研究では，日本語の学術論文を対象として，前処理・深層学習・強化学習を組み合わせた自動要約手法を提案する．なお，抽出された要約の内容を明示的にユーザに示すため，Highlight 処理を行う GUI も開発した．研究の全体像を図 3.1 に示す．

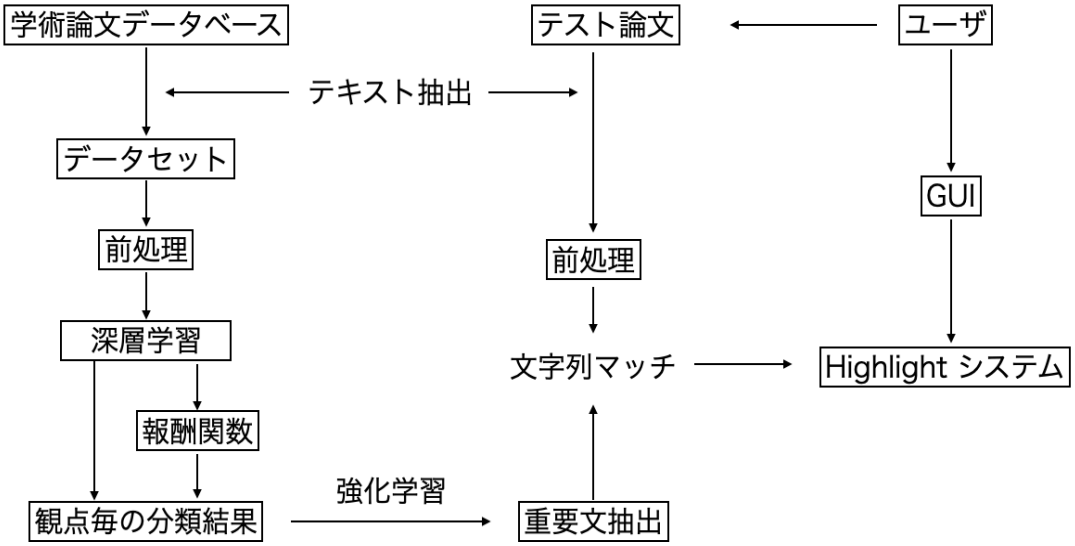


図 3.1: VPRAS システム

3.2　前処理

前処理のテキスト処理手順を図 3.2 に示す．インターネットで取得した論文の初期フォーマットは PDF である．まずは，テキストを認識するため，PDF の解析を行い，未正規化テキスト (生データ) を生成する．そして，本文だけ抽出するため，生データに正規化の作業を行う．最後は熟練者によりメインクラス，サブク

ラス，重要度といったアノテーションデータを付与して，最終的なデータセットが準備される．アノテーションデータのうちメインクラスは深層学習の教師データとして使用され，サブクラスと重要度アノテーションは強化学習の入力データとして使用される．本研究では学術論文の本文を対象とし，Apache Tika と呼ばれる Java で開発されたドキュメント分析・抽出ツールを利用して，各文章に背景，目的，方法，実験，結果評価，知見，関連研究，その他からなる論文内の観点を定義したメインクラスを教師データとして付与した．また，文章の論文内における位置情報は要約に重要であると考えられる．例えば，研究背景は一般に論文の先頭部分に出現する確率が高い．そこで，各論文の文章を m 等分にし，各文章の先頭部分に文字列フォーマットの pos1,pos2⋯,posm まで番号を付与して位置情報とした．また，各文章にメインクラス内容に基づいて細分化し，サブクラスを付与した，サブクラスの詳細は節 3.2.2 に述べる．最後に，熟練者によりメインクラス毎に各文章の重要度をランク付けした重要度ラベルを付与した．

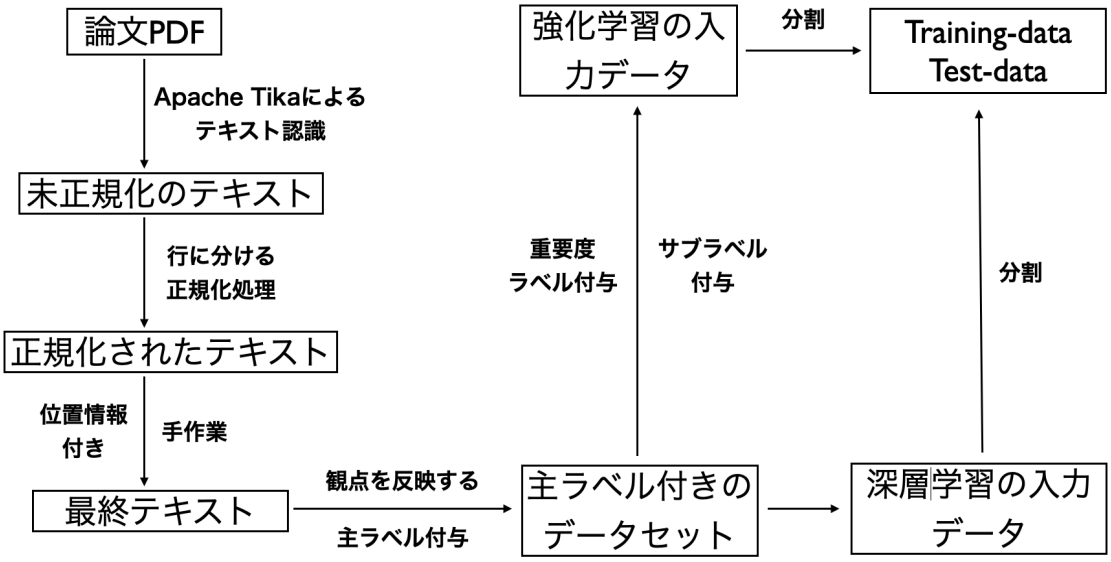


図 3.2: 前処理の全体像

3.2.1 正規表現

本研究において，テキストクリーニングはテキスト前処理の段階における重要な部分であり，深層学習による文書分類精度に関わるとともに，Highlight 処理の安定性を保証する重要な役割を果たす．深層学習の入力層に投入する文書は論文の本文のみとし，不要な固有表現は削除する必要がある．こうしたデータクリーニング作業をするため，re と呼ばれる特殊なパターンを認識できる Python のモジュールを使用する．論文 PDF にある全てのテキストを認識し，文章単位に分割してから表 3.1 で示した正規化処理を行う．完全に削除できない固有表現は手作業で処理した．

表 3.1: テキスト正規化処理

step	説明
1	はじめにの前の部分を削除
2	参考文献の後ろの部分を削除
3	章節の番号を削除
4	固有表現（メール，URL など）を削除
5	日本語を含めない行を削除
6	図と表を削除

3.2.2 サブクラスおよび重要度ラベル

3.2 節で定義したメインクラスの各観点を細分化することにより，サブクラスを構成する．図 3.3 はメインクラスとサブクラスの詳細を示した．各文章の重要度を判定するため，熟練者は以下の基準により重要度ラベルをつける．
(1) 各論文のメインクラス内の文章の中で比較して重要な文章に 1 から m の順にランクをつける．
(2) 同程度の重要度の文章には同じランクをつける．
(3) 最大で m=4 までランクをつける．
(4) 付けた個数は最大でメインクラスの文章数の 2/3 以内まで
なお，サブクラスと重要度情報は，強化学習モデルにおける環境構築に利用される．最終的に作成したデータセットの一部は図 3.4 のような形となった．

メインクラス	サブクラス1	サブクラス2	サブクラス3	サブクラス4	サブクラス5	サブクラス6	サブクラス7
1：背景	11：社会的な背景	12：関連研究に基づく背景	13：研究背景となる仮説	14：背景としての研究課題			
2：目的	21：研究目的	22：サブゴール	23：研究全体のゴール +研究における仮定	24：期待される効果			
3：方法	31：方法の全体像	32：利用する技術	33：設計・手順	34：特徴・効果	35：定義	36：データセット	37：開発
4：実験	41：実験目的	42：実験条件	43：実験方法	44：実験設定			
5：結果評価	51：実験結果	52：考察	53：実験に対する課題				
6：知見	61：やったこと	62：結論としての知見	63：研究分野への貢献	64：今後の課題			
7：関連研究	71：関連研究の紹介 +関連研究による位置づけ	72：関連研究における課題	73：自身の先行研究				
8：その他	81：論文の流れ	82：章節の説明	83：用語説明	84：ゴミ			

図 3.3: サブクラス

論文タイトル	sentence	正解ラベル	サブラベル	重要度
逆強化学習を用いた転移可能な報酬関数の推定	強化学習において学習効率の改善は重要な課題であり,そのための手法として,解決が容易な既知のタスクで得られた情報を未知のタスクに再利用する転移学習がある	1	2	1
逆強化学習を用いた転移可能な報酬関数の推定	これは過去に類似したタスク(以後,元タスク)を学習することで得た知識を,これから解こうとするタスク(以後,目標タスク)の学習に転用すべき知識を即座に獲得する,あるいはその知識を獲得するための手がかりを得ることに,	1	2	2
逆強化学習を用いた転移可能な報酬関数の推定	転移学習という語は幅広い機械学習の枠組みとして用いられている	1	2	
逆強化学習を用いた転移可能な報酬関数の推定	強化学習に応用した例として,fernardo ら[Fernando 06] は元タスクで学習した方策を,目標タスクを学習するエージェントに対して確率的なバイアスとして与え,学習エージェントが現在の方策を用いるか,新しい状態を探索	7	1	1
逆強化学習を用いた転移可能な報酬関数の推定	Matthewら [Matthew 07]は元タスクと目標タスクが大きく異なる場合(例えば迷路問題からKeepaway等)の転移を可能にするために,元タスクで得た方策をデータベースにまとめ,修正したものを目標タスクに用いるルール	7	1	2
逆強化学習を用いた転移可能な報酬関数の推定	これらの手法は元タスクで獲得した方策を転移させる手法である	7	1	
逆強化学習を用いた転移可能な報酬関数の推定	一方で方策ではなく報酬を転移させる手法として逆強化学習の分野でも研究されている	1	2	
逆強化学習を用いた転移可能な報酬関数の推定	逆強化学習は最適な行動を知るエージェント(エキスパート)の行動軌跡と状態遷移確率を用いて報酬関数を推定する枠組みである	1	2	
逆強化学習を用いた転移可能な報酬関数の推定	Monicaら [Monica 11]はそれぞれ別々の目的を持った複数のエキスパートの行動軌跡から同じ目的を持つエキスパートを EMアルゴリズムを用いてクラスタリングし,各クラスタに対して報酬関数を推定する	7	1	
逆強化学習を用いた転移可能な報酬関数の推定	次に新たな軌跡に対してベイズルールからどのクラスタに含まれるかを推定し,報酬関数を割り当てる	7	1	
逆強化学習を用いた転移可能な報酬関数の推定	Jaedeug ら[Jaedeug 12]はディリクレ過程混合モデルを利用し,あらかじめクラスタ数を設定する必要のないノンパラメトリックベイジアンアプローチによる報酬関数の推定を行った	7	1	
逆強化学習を用いた転移可能な報酬関数の推定	この手法では新しい軌跡が与えられたときに,以前の学習結果を用いて効率的に報酬関数を計算する	7	1	
逆強化学習を用いた転移可能な報酬関数の推定	逆強化学習を用いた転移学習は,所与として目標タスクにおけるエキスパートの行動軌跡が必要となるため,適用範囲が限定される	1	4	3
逆強化学習を用いた転移可能な報酬関数の推定	そこで本研究では目標タスクにおいてエキスパートの行動軌跡の代わりに,環境の情報の一部を用いた報酬関数の転移法を提案する	2	1	1
逆強化学習を用いた転移可能な報酬関数の推定	具体的には元タスクとして,内部に複数の障害物が存在する迷路問題を用意し,それぞれの報酬関数を Abbeelらの逆強化学習 [Abbeel 04]により求める	2	2	
逆強化学習を用いた転移可能な報酬関数の推定	次に各障害物の位置と報酬関数を入出力に設定した教師有学習を行うことにより,目標タスクに対する報酬関数を求める	2	2	
逆強化学習を用いた転移可能な報酬関数の推定	最後に計算機実験により本手法の有用性の検証を行う	2	2	
逆強化学習を用いた転移可能な報酬関数の推定	各状態で最適な行動をとるエージェントをエキスパートと定義する	3	5	
逆強化学習を用いた転移可能な報酬関数の推定	Abbeelの逆強化学習ではエキスパートの行動軌跡を所与とし,エキスパートと同じような行動軌跡が得られる報酬関数 Rを推定する	3	5	
逆強化学習を用いた転移可能な報酬関数の推定	具体的には,状態を S → [0, 1]k で表される特徴量 ϕ(st)で定義し,各特徴量の出現頻度によって計算される特徴期待値 μ(π) = E[∑∞t=0γtϕ(st)|π] ∈ Rk を用いて行動軌跡を数値化する	3	5	
逆強化学習を用いた転移可能な報酬関数の推定	エキスパートの特徴期待値との差が ε以下になるような報酬関数を推定することにより,エキスパートに近似した行動軌跡を獲得することができる	3	4	
逆強化学習を用いた転移可能な報酬関数の推定	Abbeelの逆強化学習では,エキスパートの特徴期待値との差が τ 以下になる特徴期待値を学習を通して得ることができる報酬関数を推定する	3	5	
逆強化学習を用いた転移可能な報酬関数の推定	なお,文献 [Abbeel 04]ではmax-margin法と projection法が提案されている	7	1	
逆強化学習を用いた転移可能な報酬関数の推定	それぞれ QPsolver,正射影ベクトルの計算を用いて報酬関数を推定する	7	1	
逆強化学習を用いた転移可能な報酬関数の推定	実験では projection法が収束率の点でわずかに良い性能を示していたため,本研究では projection法を用いる	3	2	
逆強化学習を用いた転移可能な報酬関数の推定	projection法のアルゴリズムを図1に示す	3	3	
逆強化学習を用いた転移可能な報酬関数の推定	パラメータ τ はエキスパートの特徴期待値と推定した報酬による特徴期待値の差が十分近づいたことを判定するものであり,終了判定を行うための閾値である	3	5	
逆強化学習を用いた転移可能な報酬関数の推定	なお,エキスパートの特徴期待値 μE は,エキスパートのm試行の行動軌跡 {s(j)0 , s(j)1 ,=1 から式 (1)によって推定する	3	5	
逆強化学習を用いた転移可能な報酬関数の推定	μ̂E =1mm∑i=1∞∑t=0γtϕ(s(i)t) (1)	3	5	
逆強化学習を用いた転移可能な報酬関数の推定	強化学習の学習効率を向上させるために,転移学習と逆強化学習を組み合わせ,環境の一部の情報から報酬関数を推定する手法を提案する	3	1	1
逆強化学習を用いた転移可能な報酬関数の推定	提案手法は,ステップ1:元タスクにおける強化学習,ステップ 2:逆強化学習による報酬関数の推定,ステップ 3:ニューラルネットワークによる目標タスクへの報酬の転移の三つのステップで構成される	3	3	2
逆強化学習を用いた転移可能な報酬関数の推定	各ステップについて迷路問題を例に挙げ,詳細に述べる	3	4	
逆強化学習を用いた転移可能な報酬関数の推定	まず,ステップ 1では元タスクを用意する	3	3	
逆強化学習を用いた転移可能な報酬関数の推定	ここではスタート,ゴール,障害物の座標を所与として,ゴールに対して報酬を与え,各迷路問題に対する最適方策を求め,最適方策からエキスパートの行動軌跡を求める	3	3	
逆強化学習を用いた転移可能な報酬関数の推定	次にステップ 2では,ステップ 1で求めたエキスパートの行動軌跡を所与として,Abbeelの逆強化学習を行い,各状態に関する詳細な報酬関数を推定する	3	3	
逆強化学習を用いた転移可能な報酬関数の推定	ステップ 3では,元タスクの環境の情報を入力,ステップ2で求めた報酬関数を出力としたニューラルネットワークを作成し,バックプロパゲーションを用いてニューラルネットワークの各重みを計算する	3	3	
逆強化学習を用いた転移可能な報酬関数の推定	以上の 3ステップを通して作成したニューラルネットワークに対して,目標タスクの環境の情報を入力することによって,目標タスクにおける報酬関数を推定する	3	1	3

図 3.4: データセットの一部

3.3 深層学習 (分類)

本研究では，図 3.5 に示すように少数の教師データを利用した深層学習を行う．この段階で前処理したデータセットを training-data と test-data に分割し，形態素解析を行う．形態素解析とは，自然言語で書かれた文を言語上で意味を持つ最小単位に分け，それぞれの品詞や変化などを判別することである [13]．本研究では品詞つきの単語を認識するため，京都大学で開発された形態素解析ツール mecab を適用した．次に，事前学習を行う段階で，各文章を python ライブラリである gensim を利用した Word2vec(単語埋め込み) と PV-DM(文書埋め込み) によりベクトル化し，分類器の入力として利用する．単語ベースの Word2vec に対しては時系列を反映可能な LSTM を，文章ベースの PV-DM 特徴に対しては SVM を分類器として利用して，各文章に対してそれぞれの観点の確率を出力する．さらに，単語ベースと文章ベースの特徴の違いを反映させるため，双方のモデルを混合した Combined-method モデルを構築する．

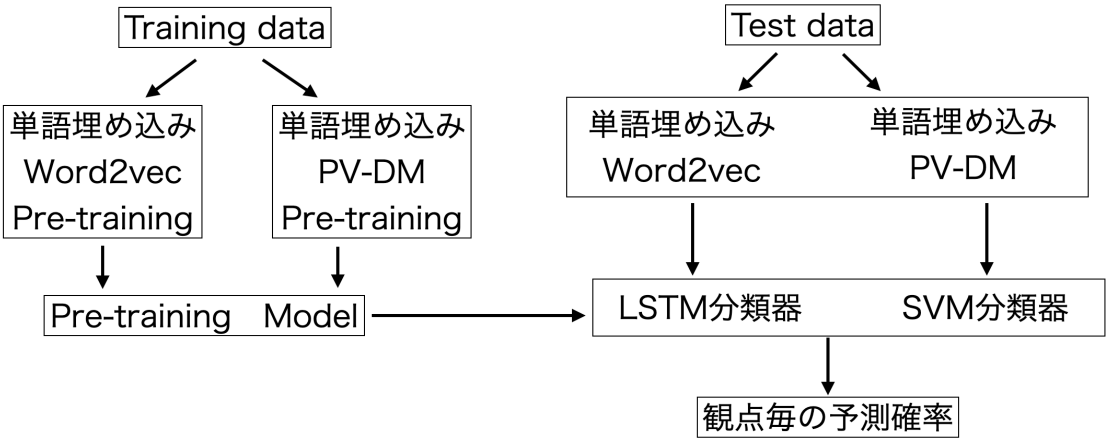


図 3.5: 深層学習モデル

3.3.1 事前学習 (Pre-training model)

Training-data のテキスト文を持っていた特徴を全体的に把握するため，事前学習は必要な手順であり，単語あるいは文の分散表現を取得することができる．本研究では，分散表現によく利用される Word2vec の Skip-Gram モデルと Doc2Vec の PV-DM モデルを検討した．

3.3.2 Word2vec

Word2vec とは，膨大な自然言語テキスト文を学習データとして単語間の物理的な近さだけを手がかりに学習を行う仕組みである [14]．本研究では，Skip-Gram

Model に基づく Word2vec を単語ベースのベクトル化に利用する．Skip-Gram とは，ある単語を対象としたときに，他の単語がどの程度共起するかを確率で予測するモデルである [15]．観点に関連する単語の特徴を得るために，各単語を対象とし，training-data から単語ごとの共起および距離をベクトル表現する．Skip-Gram model に基づくニューラルネットワークの基本構造を図 3.6 に示す．図 3.6 には”提案モデルの概要について述べる”という例文が挙げられる．”概要”は中心単語とし，隣の単語”の”と”について”が中心単語を近所に来る確率を予測することが目標である．まずは，各単語を one-hot 化し，ニュートラルネットワークに入力する．次に，中間層処理と活性化関数を通じて，出力層には中心単語を対象とした他の単語との共起程度を確率の形式で出力される．こうした得られた確率を正解ラベルと比較し，損失関数の計算を行いながら逆伝播法を実行することにより，ネットワークの重みを更新するのは最後の手順である．そこで，更新された重みを持つニュートラルネットワークモデルにより各単語の単語埋め込み (Word-Embedding) を生成できる．つまり，各単語のベクトルを得られることになる．それらの単語ベクトルが分類器に入力される．

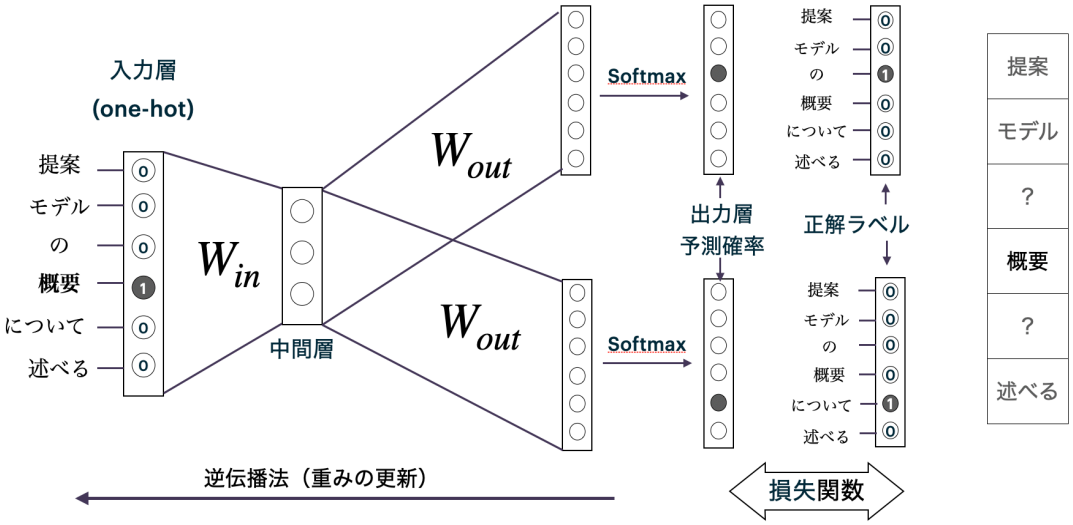


図 3.6: Word2vec(Skip-Gram) モデル

3.3.3 Doc2Vec の PV-DM(Distributed Memory Model of Paragraph Vector)

PV-DM とは，Doc2Vec 手法の一つであり，文章 ID といくつかの単語を入力として，次の単語を予測するモデルを学習する方法である [15]．Word2vec との本質的な違いは，ある単語の予測に，文章の分散表現も使用するという点である．つまり，文章ごとに単語の埋め込みベクトルを学習するのではなく，文書全体を一度に学習する．これにより，単語の特徴を考慮するだけでなく，文章の特徴も反映することが可能となる．PV-DM model に基づくニューラルネットワークの基本構造を図 3.7 に示す．入力層には文章ベクトルと単語ベクトル一斉に入力し，サンプルした単語に続く次の単語を予測することを通じて，文章ベクトルおよび中間層と出力層の重みを更新する．

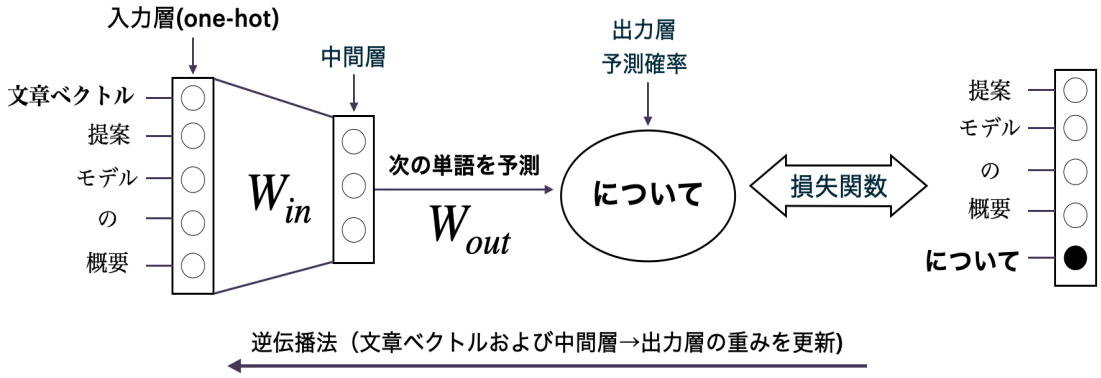


図 3.7: PV-DM モデル

3.3.4 分類器

Word2vec+LSTM：Word2vec で埋め込まれた特徴量は単語の順序に従う一種の時系列データである [14]．そこで本研究では時系列データの予測によく利用される深層学習手法である LSTM モデルを利用する．LSTM は過去の単語の情報を予測に活用することができるモデルであり，その出力は各文章に対して観点 (Class) の確率を推定したものとなる．そのため，あらかじめ指定した観点の数を N とすると，文章数× N の行列が出力される．

PV-DM +SVM ：文章の特徴を反映した特徴量を利用するため，汎化性能が高い SVM[17] による分類を行う．scikit-learn の線形 SVM は多クラス分類を行うとき，one-versus-the-rest で分類する．つまり，K クラス分類問題において，ある特定のクラスに入るか，他の K − 1 個のクラスのどれかに入るかの 2 クラス分類問題を解く分類器を K 個利用する．SVM から得られる出力フォーマットも文章数× N の行列である．

3.3.5 Combined-method

Word2vec が単語の特徴を重視している一方，PV-DM は文章の特徴を活用している．これらのベクトル化手法を組み合わせた Combined-method モデルを構築する．ベクトル化手法を式 3.1 に基づいた Combined-method モデルとして構築する．式 3.1 の proba は各文章において深層学習で得られた観点に対する確率表現である．

$$\arg max[proba_{(Word2vec)} + proba_{(PV-DM)})] \tag{3.1}$$

具体的なモデルは図 3.8 に示すように，各方法の分類結果は 1xN 行列になり，この中身の数値は各観点の予測確率と呼ぶ．各観点において 2 つの方法で分類された結果を確率で加算し，大きな値は観点の内容に近いとみなす．これらの結果から最大値を取得し再分類をすることにより，どの観点に含まれるべきかを推定する．なお，その分類結果は後述する Cos 類似度モデルにおける報酬関数の入力の一つとして取り扱う．

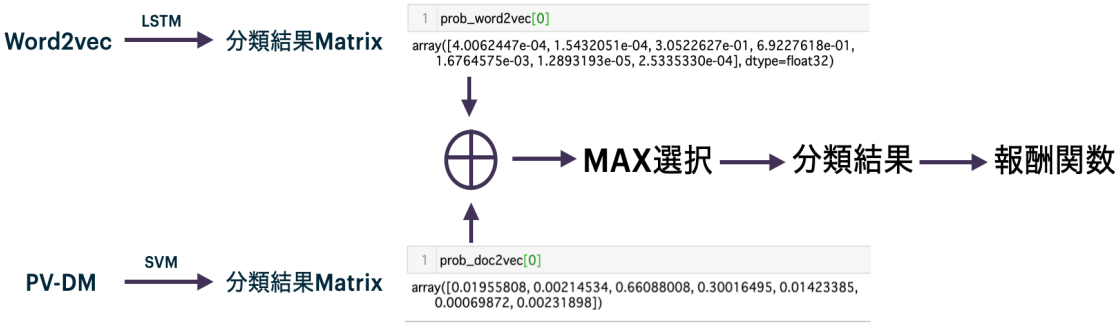


図 3.8: Combined-method

3.4 COS 類似度による改良

深層学習による分類手法においては，訓練用の論文とテスト用の論文で違う分野の文章があるケースで，トレーニングにより誤差関数が減らない可能性がある．このような場合に得られた分類結果の精度は不十分なものになると考えられる．本研究では，深層学習の分類結果を改善するため，全てのテスト文章を対象とした文章類似度の計算手法を追加する．

Kong ら [16] の強化学習モデルを参考として，文章類似度を報酬に設定する．モデルは図 3.9 で示すように，文章選択器と三つのメモリを使用する．エージェントは文章選択器により候補文を選択し，類似度が高い文章を選出する役割を担当する．候補文のベクトル化の方法は，深層学習で用いた Word2vec, PV-DM, Combined-method をそれぞれ利用する．また，候補文の生成は以下の三つのメモリのルールに従う．

(1) 類似度メモリ：各文章間の類似度の計算結果を保存するメモリ．類似度の計算には式 3.2 の Cos 類似度を用いる．a と b は二つの文章ベクトルであり，計算結果 sim は 1 に近づくと類似度が高くなると考えられる．

$$sim(\mathbf{a}, \mathbf{b}) = \cos(\mathbf{a}, \mathbf{b}) = \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{a}|\,|\mathbf{b}|} \tag{3.2}$$

(2) 抽出メモリ：現時点までに抽出された文章の集合．

(3) 候補文メモリ：まだ候補として抽出されていない文章の集合．このうち一つの文章を候補文として選択する．

現在の抽選状態 (state) で一つの文章を選択すると，抽出メモリに挿入する操作 (action) を行って，候補文メモリ (環境) からその文章を削除し，新たな候補文メモリが生成される．本研究において報酬を用いる意義は，観点を反映した文章群に新たな候補文を挿入 (action) した状態でどの程度観点の要素と類似度があるかということである．報酬の設定要素は表 3.2 にまとめた通りであり，類似度メモリ

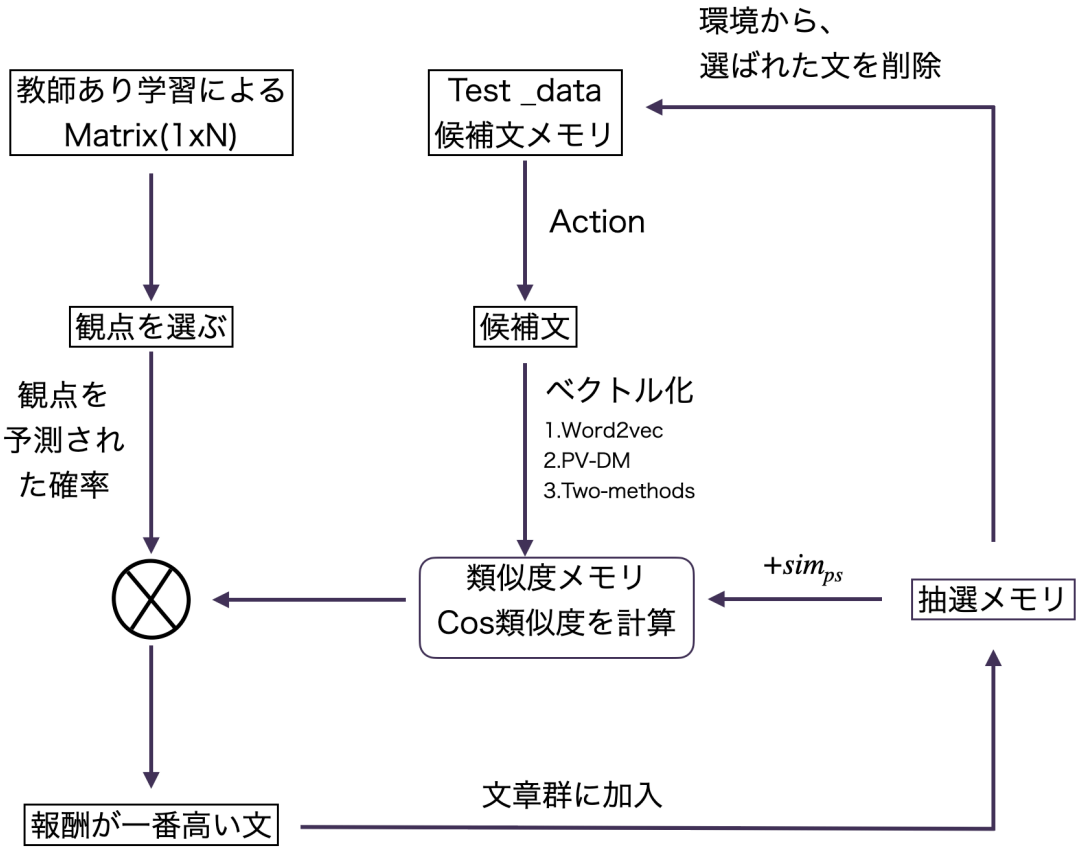


図 3.9: Cos 類似度により報酬設定モデル

表 3.2: 報酬の設定要素

要素	説明
sim	式 2 に従う計算結果
sim_{ps}	t 時刻まで (現在の文章群) の類似度情報
state	t 時刻で抽出された文
action	次の状態 t+1 の時刻で候補文メモリから一つずつ文を選出する動作
proba	候補文が深層学習の分類に当たる確率
環境	候補文メモリ
view	反映したい観点

を参照し，深層学習で得られた確率も利用することで，式 3.3 と式 3.4 に従い報酬を算出する．

$$avg_{sim} = avg(sim_{ps} + sim(state, action)) \tag{3.3}$$

$$Reward = avg_{sim} \cdot proba(action, view) \tag{3.4}$$

候補メモリからすべての文の報酬を算出したら，式 3.5 で報酬値が一番高い文を選出する．そして，式 3.5 で選出された文及び一時刻前に抽出された文との類似度を sim_{ps} に加算するのは式 3.6 になる．深層学習により分類された文章の個数を上限として上記の流れを繰り返す．分類結果に対する改善効果については 4 章で述べる．

$$a_{choice} = \arg max_{select} R(s_1, s_2.....s_n) \tag{3.5}$$

$$sim_{ps} += sim(state, a_{choice}) \tag{3.6}$$

3.5 重要文抽出

深層学習により分類されたメインクラスの内容を反映した文章群を重要文抽出要約の対象として扱う．以降では，それらの文章の重要度を測定するために提案した二つの文章価値の設定手法をと強化学習および DP 動的計画法を利用した重要文抽出手法について紹介する．

3.6 強化学習の要素

強化学習とは，最初に知識を持っていないエージェントが報酬設定のある環境からの情報を学習し，自分の知識を豊かにする半教師学習である．つまり，エージェントは一歩ずつ環境を探索しながら報酬を得ることで最適な行動を獲得する．強化学習の数理モデルを構築するため，環境におけるエージェントの状態遷移方策と報酬期待値は不可欠である [18]．しかし現実の問題においては，この報酬を設計することが困難な場合がある．特に本研究においては，単にメインクラスの情報を使うだけでは，重要文の判定と報酬関数の設計を行うことは容易ではない．そこで本研究では，節 3.2.2 で紹介した熟練者により付与されたアノテーションを利用し，テスト論文の文章毎に重要度を推定し，文章毎の価値を算出することで，強化学習の環境を構築する．

提案する強化学習モデルでは，四つのメモリを下記のように設定する．
(1) 候補文メモリ:深層学習から分類された文章群.
(2) 要約メモリ:状態 s の時刻で抽出された要約.
(3) 文章価値メモリ:状態 s の時刻で各文章の価値を保存する場所．文章価値の算定手法は節 3.6.1 と 3.6.2 で紹介する．
(4) ペナルティメモリ:情報の冗長性を解消する手法として，TF-IDF+Cos 類似度の計算により生成された文章遷移のペナルティ情報を保存する場所．ペナルティの計算手法は節 3.6.3 で紹介する．
加えて，(5)Q メモリとして，以下の二つの方法で文章価値の設定を行う．
(5-1) 方法 1 は熟練者が付与した重要度ラベルの情報に基づく方法であり，そのモデルを図 3.10 に示す．方法 1 ではまず，熟練者が付けた文章毎の重要度ラベル (重要度 1-重要度 4) を入力したトレーニングデータにおいて，重要度ラベルを付けた文章の各単語を方法 1 の重要語とみなす．テスト論文の文章は重要語の情報を利用して，重要度ラベルの確率を推定することで，文章毎の価値を推定することが可能となる．推定する手法は節 3.6.1 で紹介する．これにより，強化学習の各状態の報酬期待値を生成できる．エージェントの状態遷移については，節 3.6.3 で紹介する情報冗長性のペナルティ設定と節 3.6.7 で紹介するナップサック問題を組み合わせる．節 3.6.6 で紹介する要約長の設定方法を用い，最終的な要約を生成する．

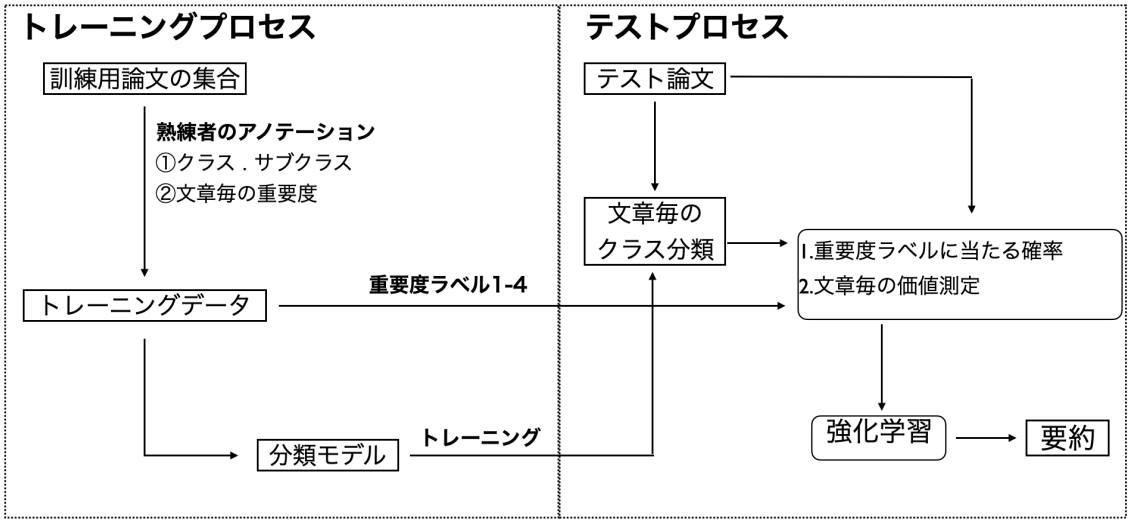


図 3.10: 方法 1 の全体像

(5-2) 方法 2 は，重要度ラベル情報に加えてサブクラスの情報を利用して文章価値を推定する．具体的な処理の流れを図 3.11 に示す．方法 2 の文章価値を設定する手順で方法 1 との違いは，サブクラスの情報を使用することである．まず，訓練用データにおいて熟練者のアノテーションに基づいてサブクラスの重要度を推定する．詳しくは節 3.6.1 に説明する．そして，テスト論文を深層学習の分類モデルを利用して，文章毎のクラス分類を行い，選ばれる観点に分類された文章群は要約の対象文章群として扱う．サブクラスの重要度を推定した結果は要約の対象とする文章群の各文章の価値を測定するため使用される．文章の価値を測定するためにもう一つの入力は，トレーニングデータにおいてサブクラスの重要語の出現回数を計算したものであり，要約対象文はサブクラスに含まれる確率を推定する．

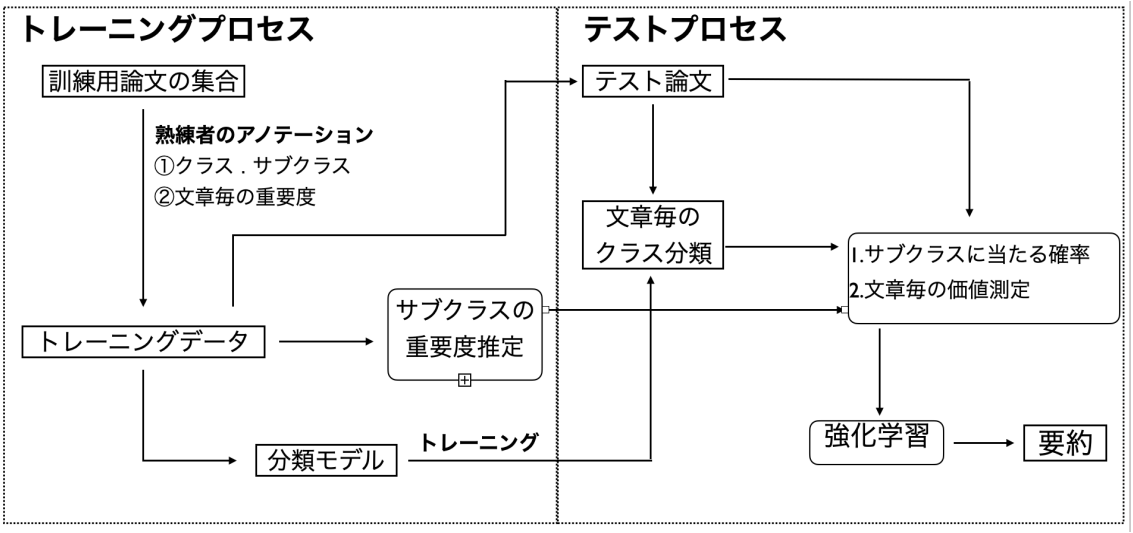


図 3.11: 方法 2 の全体像

3.6.1 方法 1-重要度ラベルに基づく文章価値の推定

方法 1 の設定は以下の四つの要素から構成される．
(1) 訓練用データのメインクラスで重要度ラベルが付与された文章の単語を重要語とみなす．
(2) 重要度ラベルのランキングによりスコアを付与する．
(3) 要約の候補文メモリにある各文章の重要語の出現回数をカウントする．
(4) 要約の候補文メモリにある各文章の単語数をカウントする．
方法 1 により文章価値を計算する流れを図 3.12 に示す．

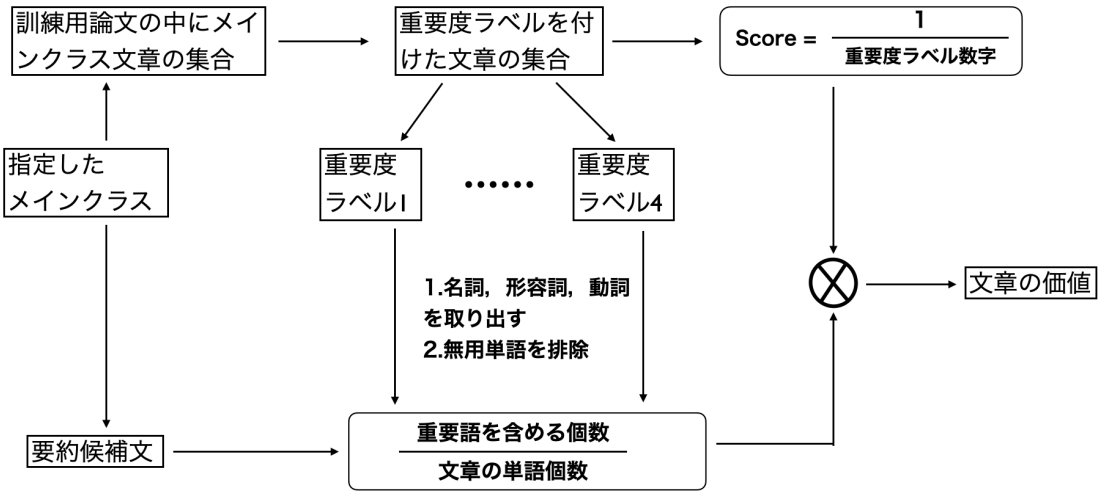


図 3.12: 方法 1-文章価値の算定

まずは，訓練用論文で指定されたメインクラスの文章群を集める．その中で重要度ラベル (1-4) が付与された文章を取り出し，それぞれランキング 1 から 4 に分けて各自の文章を保存する．次に，各重要度ラベルが付与された文章群に形態素解析を行い，名詞，動詞，形容詞を取り出す．そして，「する」「れる」「なる」「こと」などの固有表現を削除し，残った単語を方法 1 の重要語として扱う．要約の候補文毎に，重要度を含める単語数をカウントし，候補文自身の単語数を割り算して算出した各重要度ラベルに含まれる確率を文章価値とし，一つの入力として取り扱う．もう一つの入力は各重要度ラベルのスコアを設定する．具体的には，各重要度ラベルの値の逆数を取り，重要度ランキングをスコアの形式に変換する．そうして得られた二つの入力を乗算し，最終の文章価値を算出する．

3.6.2 方法 2-サブクラスと重要度ラベルに基づく文章価値の推定

方法 2 による文章価値の計算の流れを図 3.13 に示す．方法 1 に，サブクラスの情報を追加したものとなっている．方法 2 は方法 1 と似たようなモデルであるが，重要語の設定はサブクラスの重要語に変換し，サブクラスに含まれる確率を算出する．もう一つの異なる点は，重要度スコアはその重要度ラベルに対応したサブクラスの文章数をカウントする点である．これをサブクラスの重要度とみなし，サブクラスに含まれる確率と乗算することにより，最終的の文章価値を算出する．

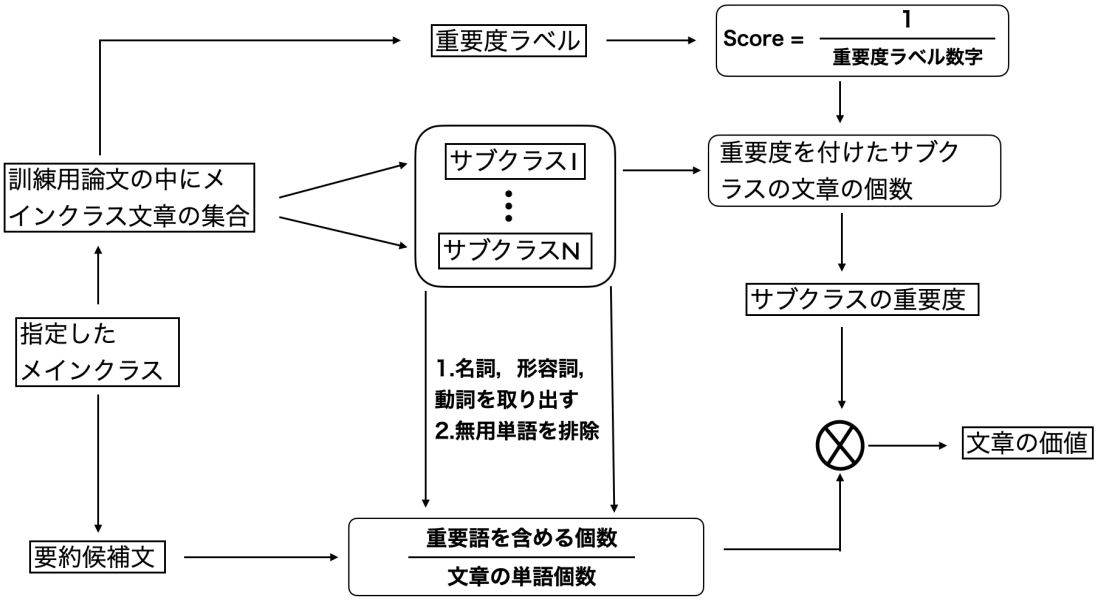


図 3.13: 方法 2-文章価値の算定

3.6.3 文章類似度

TF-IDF とは文書に含まれる単語の重要度から文書の特徴を判別する手法である．TF(Term Frequency) は単語の出現頻度，つまり，その文書においてどの単語がどれくらい出現したかを意味する．IDF(Inverse Document Frequency) は各単語のレア度を示す，逆文書頻度と呼ばれる [19].TF と IDF は式 3.7 により乗算することで，TF-IDF ベクトルを生成できる．

$$TF - IDF = TF * IDF \tag{3.7}$$

TF-IDF ベクトルを利用して Cos 類似度計算することにより，文章間の類似度を算出できる [18]．節 3.4 で紹介した方法は類似度が高い文章を集合させるといった考え方と異なり，抽出要約を行う際には要約文の中に同じ情報を多数含むと，要約の情報が冗長になり，読者の効率が低くなる．そこで本研究では，冗長な要約が生成されるのを防止するため，要約候補文の中の類似度を計算し，類似度が高ければ高いほど強化学習にはそれを対応したペナルティを設定する．まずは，候補文メモリにある文章毎に TF-IDF の encoder を行い，ベクトルを生成する．そして，状態 s と遷移先 s' の間で，式 3.2 の Cos 類似度を計算し，生成された類似度行列を強化学習の報酬期待値のペナルティとして設定する．各ペナルティの値はペナルティメモリに保存する．

3.6.4 報酬の期待値

強化学習を行う前提として，ある時刻の状態 s はその時刻の動作 a により遷移先 s′ に到達し報酬 R を獲得するといった流れで，三つの要素が不可欠である．本研究での状態 s は t 時刻での文章，動作 a は候補文メモリから文章を選択する動き，遷移先 s′ は選択された文章と考える．本節ではこうして得られた即時報酬 Rass' の計算方法を議論する．t 時刻での状態 s と行動 a を選択するときに，次の状態 s′ に遷移する流れを通じて得られた報酬の期待値は，以下の式 3.8 のようになる [20].

$$R_{ass'} = E\left\{r_{t+1} \mid s_t = s, a_t = a, s_{t+1} = s'\right\} \tag{3.8}$$

一般的な設定は遷移先 s' の価値をそのまま適用することになるが，本研究の環境構築においては，節 3.6.3 で紹介した情報冗長性の条件を加えるため，報酬期待値は以下の式 3.9 で挙げられる．

$$R_{ass'} = V_{ass'} - \gamma * avg(V) * sim_{ass} \tag{3.9}$$

式 3.8 のうちには Vass' は遷移先 s' の文章価値，avg(V) は t 時刻で文章価値の平均値，γ は t 時刻の割引率 (初期値は 0.9 と設定する)，sim_{ass}' は遷移先に移動したペナルティである．式 3.8 により各時刻での報酬期待値が得られる．

3.6.5　要約長の制限

抽出要約の文字量及び文章数を制御するため，トレーニングデータから重要度が付与された文章数とメインクラスの文章数との割合を計算し，要約の抽出率を決める．具体的な流れを図 3.14 に示す．具体的には，トレーニングデータにある i 番目の論文を対象とし，指定した観点を反映するメインクラスの文章群から重要度ラベルが付与された文章数÷文章総数により計算する．トレーニングデータにある全ての論文を上記の流れを繰り返し，加重平均を計算する．こうして得られたメインクラス毎の平均抽出率を要約対象文の文章数と掛け算し，マージンを設定することで，文章数の許容範囲という制約条件を設定した．文字列長さの許容範囲の設定は上記の流れと同じであり，単に文章数を文字列の長さに変更して設定したものである．

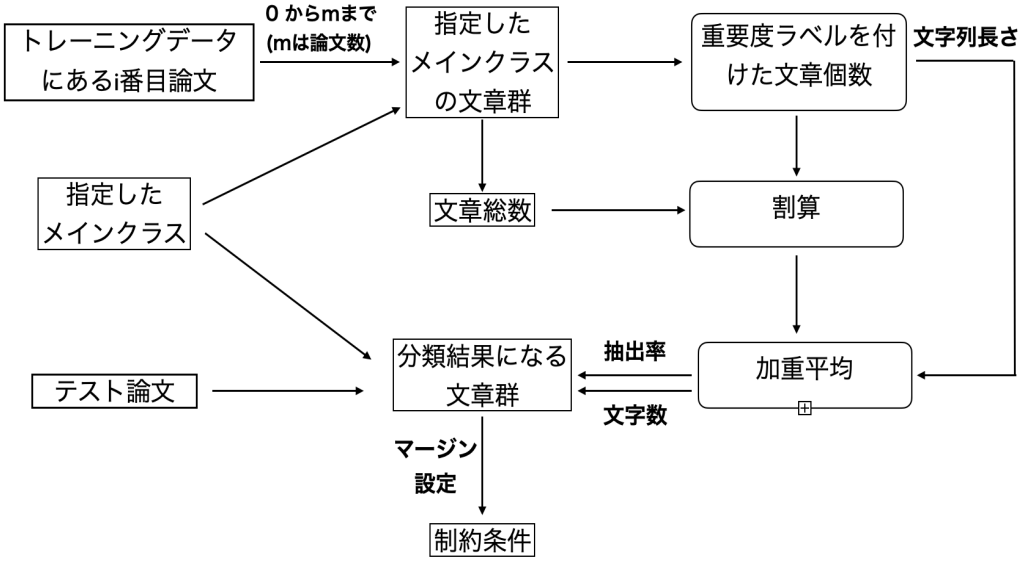


図 3.14: 要約長の制限

3.6.6　組み合わせナップサック DP(動的計画法)

組み合わせ最適化とは，与えられた条件を満たすような要素を順番を選び，組み合わせた集合の中で選択できる組み合わせの中から一番良いものを探し出すといった課題である．要約の対象となる価値を付与した文章群は組み合わせ最適化の要素群とみなすことができ，一定な要約長の範囲を超えないという制約条件を加え，最も適切な組み合わせ結果を要約として抽出する．ナップサック DP(Dynamic Programming) は動的な計画の一種であり，代表的な組合せ最適化問題である [21]．ナップサック問題の基本要素を以下に示す．

(1) 複数の荷物と各荷物の価値 (value) と重み (weight)

(2) 組み合わせ荷物の最大重量制限
(3) 荷物の個数をある範囲内で制限
重要文抽出モデルに適用すると，’文章群’は荷物の集合，’文章’は荷物，3.6.1 と 3.6.2 節で算出した’文章の価値’は荷物の価値，’文章の文字列長さ’は荷物の重みとみなすことができる．本研究では図 3.12 に示すように，要約の短縮化をしながら，重要な情報を保留するといった目的として，強化学習と組み合わせ最適化ナップサック DP を結合した動的計画法を提案する．n 個の文章 x[0],x[1],x[2]⋯x[N] が与えられ，それぞれの価値 が v[0],v[1],v[2]...v[N] であるとき，これらの文章から何個かの文章を選択して総価値の最大値を求めることが目的となる．各文章の重み (文字列長さ) をそれぞれ w[0],w[1],w[2]⋯w[N] とし，制約条件は式 3.12 に従い，length を超えない範囲で文章を選択する．

$$Max. \qquad f(x) = \sum_{i=0}^{n} v_i x_i \tag{3.10}$$

$$Subject \quad to. \qquad \sum_{i=1}^{n} w_i x_i <= length \tag{3.11}$$

$$Subject \quad to. \qquad \sum_{i=1}^{n} x_i <= m \tag{3.12}$$

$$x \in [0,1] \qquad (i = 0, 1, 2..., n) \tag{3.13}$$

式 3.12 では，現在の状態において要約の最大文章数 m を超えない制限を付ける．m の値は強化学習の実行を通じて変化する．具体的な手続きは，節 3.6.7 で紹介する．式 3.13 において，文章 x[i] が選ばれる場合は状態’1’となり，選ばれない場合は状態’0’になると設定する．それらの制約条件を付与する上で，選択した文章の価値の総合を目的関数として最大化する線形計画法を構築する．毎回ナップサック DP の実行結果は強化学習の方策 π として使用される．

3.6.7　強化学習による重要文抽出

節 3.6.4 から節 3.6.6 まで紹介した要素を用い，強化学習の流れは図 3.15 に示す．各ステップは表 3.3 に示す．

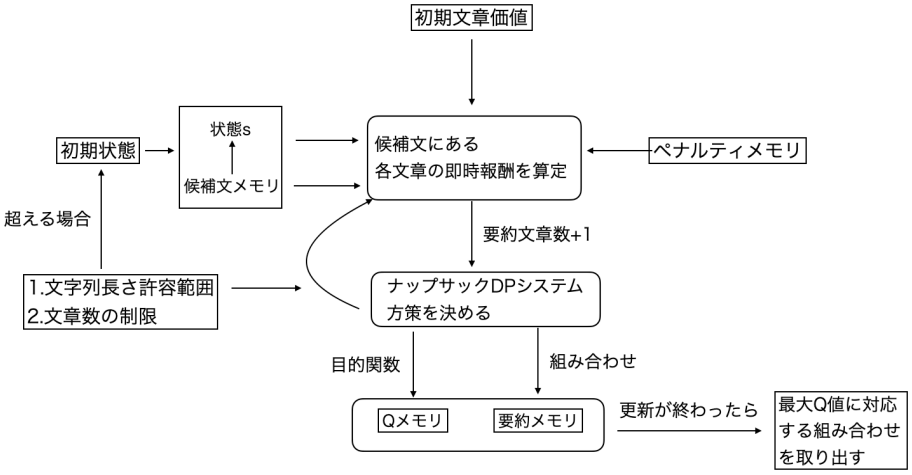


図 3.15: 強化学習流れ

本研究では，まず文章価値が一番高い候補文は強化学習に含めず，必ず抽出するものとする．次は候補文のうちに i 番目文章を選出し，式 3.8 により，各行動の報酬期待値を更新する．そして，ナップサック DP システムに一つの文章を追加して，ナップサックシステムの最適化調整により得られた目的関数を Q メモリに保存し，組み合わせの結果を方策 π として，要約メモリに保存する．文章数と文字列長さの許容範囲に到達までナップサックシステムを繰り返す．文章数と文字列長さの許容範囲を超える場合は，文章価値を初期状態に戻り，状態 i+1 番目から次のループになる．候補文メモリにある文章を全て走査したら，Q メモリと要約メモリの状態更新が終わり，Q メモリにある最大 Q 値に対応する組み合わせ結果を最終要約の抽出番号とする．

表 3.3: 強化学習による重要文抽出の流れ

流れ	説明
Step1	一番高い価値を持っている文章を選択する
Step2	残りの候補文のうち，i 番目の文章を最初状態 s とする
Step3	式 3.8 により各行動の報酬期待値を更新する
Step4	文章 (荷物) を一つ追加して，ナップサック DP を実行する
Step5	ナップサックの結果から目的関数の値を取得し，Q メモリを更新する
Step6	組み合わせの結果を方策 π として，要約メモリに保存する
Step7	割引率 γ は半分とする
Step8	Step4-Step7 は文章数と文字列長さの許容範囲に到達までに繰り返す
Step9	文章の価値と γ を初期状態に戻り，i=i+1，i=候補文数まで，Step2-Step9 を繰り返す
Step10	Q メモリにある一番高い Q 値に対応する組み合わせを最終要約の抽出番号とする

3.7 ユーザーインターフェイス

3.7.1 GUI

GUI はグラフィカルユーザーインターフェイスの略であり，ユーザーの操作に基づいて要約を自動生成させることを目的として GUI を開発することとした．本研究ではライブラリー Tkinter を使い，Python アプリが動作するユーザーインターフェイスを構築した．画面を図 3.11 に示す．ユーザーの操作は表 3.3 のような流れとなる．

表 3.4: GUI 操作

流れ	説明
Step1	ユーザーは参考したい論文をアップロードし，論文 title を取得
Step2	観点をいくつかを選択し，それぞれチェックボタンをクリック
Step3	要約生成ボタンをクリックすると，要約を要約内容欄に生成
Step4	Highlight ボタンをクリックし，描画された論文を呼び出す
Step5	選択解除ボタンをクリックし，最初の手順に戻る

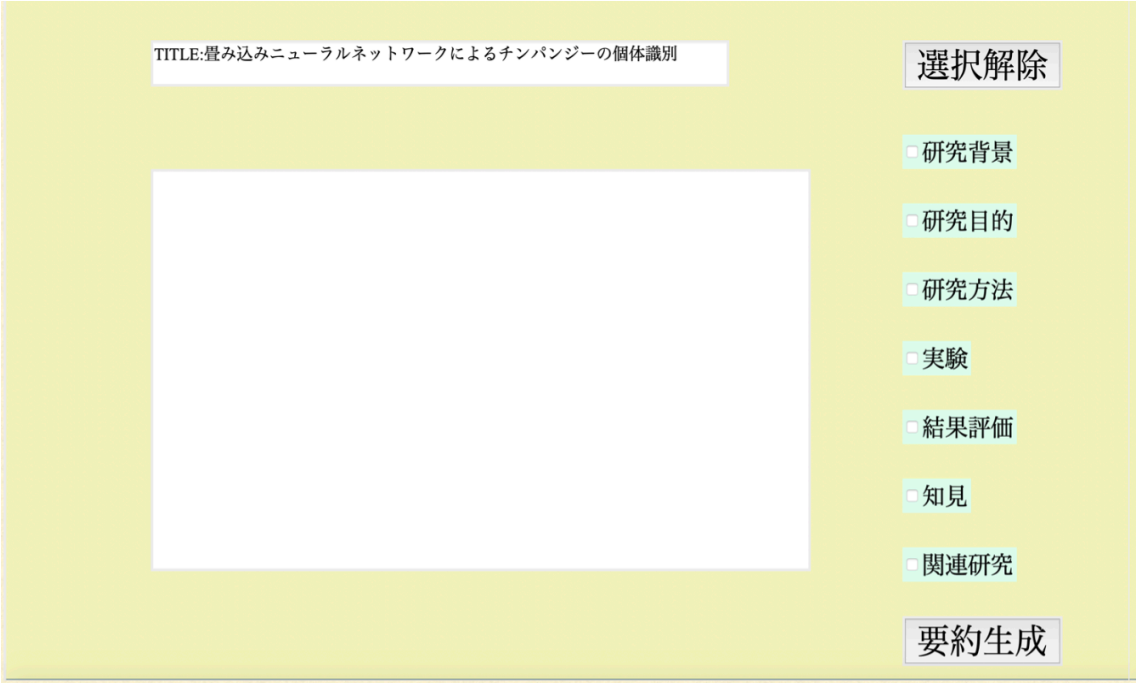


図 3.16: GUI 画面

3.7.2 Highlight システム

読者により直感的な要約を提供するため，本研究では，抽出された重要文を元の文章内でマークし，重要文の位置を論文内で可視化する Highlight 処理を行う．本研究の Highlight システムでは図 3.11 に示すように，前処理により取り出したテキストと抽出要約 s[0],s[1]…s[n] を一文ずつマッチし，ライブラリー PyMuPDF を使い，それぞれ文章内の座標情報 (rect[point[upper-left],point[lower-right]) を取得することにより，描画処理を行う．これにより，要約を文字列のフォーマットだけではなく，論文内で強調される位置情報が可視化された画像のフォーマットに変換することが可能となる．

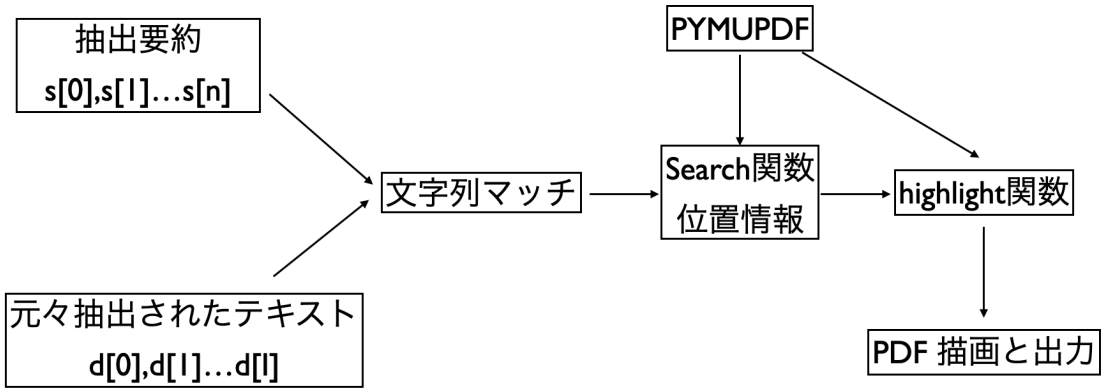


図 3.17: Highlight システム

3.8 開発環境

本研究では，計算機に macbook pro 2020，アプリケーションに Anaconda Navigator にインストールされた Python3.8 と jupyter notebook を利用した環境で開発した．本研究で使用したライブラリー及びそれらの機能を表 4.1 に示す．

表 3.5: ライブラリーと機能

ライブラリー	機能
Numpy	行列計算など
Tika	PDF テキスト認識
Pandas	データ前処理
Re	正規表現
MeCab	形態素解析
Gensim	Word2vec と PV-DM 事前学習
Sklearn	SVM 分類器，深層学習の評価，TF-IDF ＋ Cos 類似度計算
Keras	LSTM 分類器
Matplotlib	学習曲線の作成
Pulp	動的計画法
Fitz，PdfMupdf	highligh 認識

第4章　実験

4.1　実験の全体像

本研究の実験は，深層学習の評価，強化学習の評価の二つから構成される．
(1) 深層学習の評価については，まず，単に PV-DM と Word2vec で事前学習した時の，SVM と LSTM の分類結果を報告する．さらに，それらの精度をベースラインとして，節 3.3.5 に紹介した Combined-method の精度向上を検証する．また，Combined-method をベースとし，PV-DM と Word2vec それぞれの単語埋め込みによる Cos 類似度関数を用い，メインクラスの文章を抽出する．その結果を Combined-method と比較することで，有効性を検証する．深層学習部分の実験では，機械学習でよく使用される学習曲線，再現率，適合率，F1 スコアといった基準により評価を行う．
(2) 強化学習による重要文抽出実験では，まず，節 3.6.1 で紹介した文章価値に基づき，高価値文章の抽出結果を比較する．さらに，強化学習を導入した要約の抽出結果を報告する．重要文抽出の実験評価は 2 つの基準がある．
1. 熟練者が付与した重要度ラベルとの一致度を基準として評価する．
2. 要約評価で一般に使用される ROUGE-N と ROUGE-L に基づき評価する．
最後に抽出された文章のうち間違った文章の特徴を分析し，深層学習の誤認識と強化学習の誤認識に分けて考察する．

4.2　データセット

本研究で準備したデータセットは，google scholar で収集した日本語で記述された学会論文であり，強化学習を対象とするもの 25 編，音声認識 10 編，画像認識 20 編，言語処理 15 編，深層学習 28 編，教育工学 2 編であった．前処理の手続きに従って論文をそれぞれ文章に分解しラベルを付与した．各ラベルの文章数は表 4.1 に示す通りである．
なお，本研究では 8 つの観点を設定したが，関連研究については論文毎に位置が大きく異なっていることが観測された．そこで，分割数 m=6 と設定し，各論文を 6 等分して文章の位置番号を割り振った．

表 4.1: データセット明細

観点	センテンス数
背景	820
目的	347
方法	3856
実験	1755
結果評価	1798
知見	553
関連研究	1021
その他	161

4.3 深層学習の実験設定

論文の観点として，研究背景，研究目的，研究方法，実験，結果と評価，知見，関連研究の 7 つを設定した．より安定性の高い分類結果を得るために，100 編の論文の各文章をランダムに 8 割を訓練データ，2 割をテストデータとして利用した．訓練データで学習した深層学習の結果をテストデータに適用して二値分類と多値分類を実行した上で，報酬関数を行うことにより結果を改善する．提案手法の有効性を検証するため，3.3.5 および 3.4 節で説明した方法に加えて，深層学習で観点の可能性が最も高い文章を抽出し，指定した要約長まで Cos 類似度の高い文章を追加する手法をベースラインとして設定した．また，それぞれのベクトル化手法とし て Word2vec, PV-DM, Combined-method をそれぞれ採用した結果，3 種類の分類実験× 3 種類の単語埋め込みによる表 4.2 のような実験設定となった．

表 4.2: 実験設定

実験項目	説明	埋め込み手法
1.Cos 類似度	深層学習で観点の 可能性が最も高い文章を抽出し， 指定した要約長まで Cos 類似度 の高い文章を追加する手法	1. Word2vec 文章ベクトル 2. PV-DM 文章ベクトル 3. 1+2(Combined-method)
2. 深層学習	節 3.3 の方法	1. Word2vec 単語ベクトル 2. PV-DM 文章ベクトル 3. 式 1 に従う Combined-method
3.Cos 類似度 +深層学習	節 3.4 の方法	Cos 類似度埋め込む手法 ＋深層学習の Combined-method 手法

なお，最終的には新しい論文を対象として抽出精度の検証を行った．結果の安定性を保証するため，データを 10 回ランダムに取得した結果の平均値を取る．なお，抽出精度は式 4.1 に示す通りである．

$$Acc = \frac{True \quad positive}{Total \quad positive} \tag{4.1}$$

4.3.1 深層学習のパラメータ

Word2vec と SVM 事前学習の単語埋め込み次元数は 300 次元と設定する．分類器のパラメータは以下のように示す．

(1)SVM:

1:単語埋め込みの次元数は 300 と設定する．

2:kernel は linear を選択する．

3:C は誤分類の許容程度を決定する [22]，C が大きくなると，過学習になりやすいため，C=0.1 と設定した．

4:シンプルなモデルを構築するため，境界線の複雑性を表す gamma は 1 と設定する．

(2)LSTM のパラメータのまとめは図 4.1 に示す．

1:過学習を抑制するパラメータ dropout は 0.3 と設定する．

```
Model: "sequential_18"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_18 (Embedding)     (None, None, 300)         4097100
_________________________________________________________________
lstm_18 (LSTM)               (None, 32)                42624
_________________________________________________________________
dense_18 (Dense)             (None, 8)                 264
=================================================================
Total params: 4,139,988
Trainable params: 4,139,988
Non-trainable params: 0
```

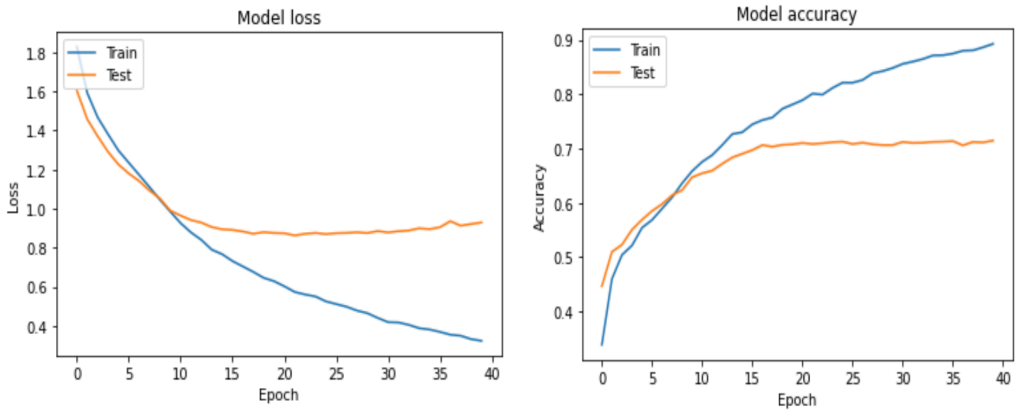
図 4.1: Keras LSTM Summary

2:出力層の活性化関数は softmax を使用する．

3:batch size は 512 にし，epochs は 40 とする．

4.4 深層学習による多値分類結果

LSTM による Word2vec 埋め込み手法を利用した学習曲線を図 4.2 に示す．Word2vec-LSTM と PV-DM 及び Combined-method の学習結果の再現率，適合率，F1 スコアは表 4.3 に示す．



学習曲線 loss　　学習曲線 acc

図 4.2: 学習曲線

学習曲線の loss については，20epoch あたりで値が 0.71 まで下がり，それ以降のテストデータの認識精度は変化しないことがわかった．単独に事前学習の手法を行う場合は，Word2vec がより良い結果を得たが，提案した Combined-method の F1 スコアはさらに 0.07 向上した．

表 4.3: 深層学習における実験結果

方法	適合率	再現率	F1 スコア
Word2vec	0.709	0.714	0.710
PV-DM	0.666	0.671	0.669
Combined-method	0.713	0.723	0.717

Cos 類似度だけのモデルを使用した表 4.4 では，PV-DM がもっとも良い結果であったが，全体的に分類精度は低かった．表 4.5 の深層学習 (Combined-method) の結果と Cos 類似度関数を併用した場合については，ベクトル化を PV-DM で行ったものが最も精度が良く，深層学習と比較して精度が約 0.9%改善した．

一番高い認識精度を持った PV-DM 埋め込みの Cos 類似度+深層学習 Combined-method を併用した分類結果は重要文抽出実験の要約対象文として使用する．

表 4.4: Cos 類似度における実験結果

Word2vec	PV-DM	Combined-method
32.75%	47.44%	42.16%

表 4.5: Cos 類似度+深層学習 Combined-method. 正解率表

Word2vec	PV-DM	Combined-method
72.05%	72.60%	71.20%

4.5 強化学習による重要文抽出の実験設定

重要文抽出を始める前に，まず深層学習を実行する．データセットにある 100 編の論文のうち，90 編を訓練用データとし，テスト用データの 10 編に対して 1 編ずつ分類を実行する．それぞれの分類結果は要約の対象文として強化学習による重要文抽出を行う．最終的に，10 編の要約抽出結果に対して，熟練者が付与した重要度ラベルを参照し，認識精度を評価する．

(1) 訓練用データの明細を表 4.6 に示す．論文分野の情報と編数は表 4.7 に示す．

表 4.6: 訓練用データの明細

観点	センテンス数
背景	721
目的	309
方法	3476
実験	1640
結果評価	1646
知見	501
関連研究	952
その他	145

表 4.7: 訓練用データの明細

分野	枚数
強化学習	22
音声認識	8
画像認識	19
言語処理	12
深層学習	28
教育工学	1

(2) テスト用データの明細は表 4.8 に示す通りである．

テスト論文を対象とした深層学習による再現率の結果から見ると，訓練用データ編数が少ない分野 (音声認識，教育工学) では再現率が 0.7 以下となった一方，訓練用データ編数が多い分野 (強化学習，画像認識) では高い再現率が得られた．したがって，強化学習の実験評価は，深層学習の誤認識を考慮しない状態で，重要

表 4.8: テスト用データの明細

	分野	文章数	深層学習の再現率
Test1	音声認識	70	0.652
Test2	音声認識	129	0.682
Test3	画像認識	87	0.811
Test4	教育工学	82	0.625
Test5	言語処理	180	0.715
Test6	言語処理	45	0.822
Test7	言語処理	56	0.767
Test8	強化学習	49	0.795
Test9	強化学習	65	0.676
Test10	強化学習	78.8	0.788

度ラベルに当たる確率を計算する実験を追加する．

(3) 各メインクラスの文章個数抽出率と文字列抽出率を表 4.9 に示す．この抽出率を利用し，動的計画法において要約長の許容範囲を設定する．

表 4.9: メインクラス (観点) 毎の抽出率

観点	文章個数の抽出率	文字列の抽出率
背景	0.314	0.341
目的	0.440	0.494
方法	0.084	0.101
実験	0.142	0.174
結果評価	0.143	0.169
知見	0.425	0.457
関連研究	0.200	0.224

上記の抽出率に従い，得られた文章数制限については，実際の状況によりマージンを設定する．研究方法と実験の候補文章数が多いため，2 個以上抽出することを必要条件として加えて，「抽出率*候補文章数」とする．他のメインクラスでは，それほどの候補文章数ではないが，重要度ラベルは複数存在する．できる限り重要度ラベルを推定するため，「(抽出率*候補文章個数)+1」と設定する．上記の抽出率をベースとして設定したが，強化学習を実行する際には候補文の個数が 2 個より小さい場合，強化学習の計算を行わず全て抽出とする．

4.5.1 ROUGE-N

Chin-Yew Lin らは，機械翻訳で使われていた BLEU と呼ばれる指標を参考にして, N-gram 単位での要約の一致を測る手法 ROUGE-N を提案した [23]．ROUGE-N スコアの計算方法を式 4.2 に示す.

$$ROUGE_N = \frac{\sum_{S \in references} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in references} \sum_{gram_n \in S} Count(gram_n)} \tag{4.2}$$

本研究で使用される要約評価手法は，ROUGE-1 と ROUGE-2 と ROUGE-L である.

(1)N=1 の ROUGE-1 手法は，参考要約 (references) に生成された要約 (summary) の単語を含む単語数に基づいた計算手法である．つまり，unigram(1-gram=単語単位) で，参考要約に一致した単語をカウントする.

(2)ROUGE-2(bi-gram) の手法では，隣り合った二つの単語をグループにし，参考要約に一致したグループを探す.

(3)ROUGE-L は, 生成した要約と参考要約とで' 一致する最大のシーケンス'(longest common subsequence=LCS) を評価するものである.

上記の方法で評価を行うが，特定のメインクラスには重要度ラベルが付与されていない場合があり，参照要約がなくて，ROUGE スコアを測定するのが不可能なケースがある．その場合はスキップ処理とする.

4.6 強化学習による重要文抽出の実験結果

(1) 各テスト論文を対象とする深層学習のメインクラス分類の再現率は図 4.3 に示す通りである．

各メインクラスの再現率

	研究背景	研究目的	研究方法	実験	結果評価	知見	関連研究
Test_1	66.6%	50%	71.4%	0%	100%	50%	25%
Test_2	100%	0%	86%	87.5%	53.8%	33.3%	30%
Test_3	57.1%	33.3%	77.4%	100%	100%	83.3%	NaN
Test_4	81.8%	20%	59.5%	NaN	NaN	80%	57.1%
Test_5	28.5%	33.3%	83.8%	67.8%	82.6%	60%	31.2%
Test_6	33%	100%	100%	100%	80%	100%	0%
Test_7	100%	40%	76.9%	70%	100%	80%	NaN
Test_8	100%	100%	70%	66.7%	100%	100%	100%
Test_9	33%	50%	83.3%	76.9%	63.6%	75%	55.5%
Test_10	55.5%	100%	95.4%	50%	80%	100%	100%
AVG	65.55%	52.66%	80.37%	68.77%	84.44%	76.16%	49.85%

図 4.3: 各メインクラスの再現率

(2) 方法 1 と 2 の重要度ラベルとの一致度は図 4.4 と図 4.5 に示す通りである．実験結果を全体的に見ると，
1:方法による差は大きくないが，方法 1 ＋ RL 動的計画法が一番良い認識率を得た．方法 1 の文章価値ランキングが二番目に良い結果となった．方法 2 の設定は全体的な正解率は 3%ほど下がるが，方法 1 で当たらないラベルが当たった場合もある．
2:深層学習により間違った認識をされた重要度ラベルは 29%を占めている．深層学習の影響を考えず，強化学習と動的計画法により正しく分類されるラベルの認識率は平均 40%を越えた．
3:Test1,Test2,Test4 の認識率が平均より遥かに低い結果を得た．三つのテスト論文の分野は訓練データに少なかった音声認識と教育工学である．

重要度ラベルとの一致度　方法1

	重要度ラベル個数	RL動的計画法正解個数	文章価値ランキング正解個数	認識率→RL動的計画法正解個数	認識率→文章価値ランキング正解個数	深層学習誤認識個数	深層学習の影響を考慮しない→RL動的計画法正解個数	深層学習の影響を考慮しない→文章価値ランキング正解個数
Test_1	11	2	2	2/11	2/11	7	0.5	0.5
Test_2	17	2	3	2/17	3/17	6	0.181	0.272
Test_3	14	4	5	4/14	5/14	1	0.307	0.384
Test_4	15	2	2	2/15	2/15	5	0.2	0.2
Test_5	19	4	3	4/19	3/19	8	0.36	0.272
Test_6	11	3	3	3/11	3/11	3	0.375	0.375
Test_7	14	8	6	4/7	3/7	3	0.727	0.54
Test_8	11	6	5	6/11	5/11	1	0.6	0.5
Test_9	16	7	8	7/16	1/2	4	0.58	0.67
Test_10	13	5	4	5/13	4/13	3	0.5	0.4
AVG	14.1	4.3	4.1	0.304	0.290	4.1	0.43	0.41

図 4.4: 方法 1-重要度ラベルとの一致度

重要度ラベルとの一致度　方法2

	重要度ラベル個数	RL動的計画法正解個数	文章価値ランキング正解個数	認識率→RL動的計画法正解個数	認識率→文章価値ランキング正解個数	深層学習誤認識個数	深層学習の影響を考慮しない→RL動的計画法正解個数	深層学習の影響を考慮しない→文章価値ランキング正解個数
Test_1	11	1	2	2/11	2/11	7	0.25	0.5
Test_2	17	2	1	2/17	1/17	6	0.09	0.181
Test_3	14	4	5	4/14	5/14	1	0.307	0.384
Test_4	15	3	3	3/15	3/15	5	0.3	0.3
Test_5	19	3	2	3/19	2/19	8	0.272	0.181
Test_6	11	3	4	3/11	4/11	3	0.375	0.5
Test_7	14	6	6	6/14	6/14	3	0.54	0.54
Test_8	11	5	5	5/11	5/11	1	0.5	0.5
Test_9	16	7	8	7/16	8/16	4	0.58	0.67
Test_10	13	5	4	5/13	4/13	3	0.5	0.4
AVG	14.1	3.9	4	0.276	0.283	4.1	0.39	0.4

図 4.5: 方法 2-重要度ラベルとの一致度

(3)10 編のテスト用データを要約対象とする ROUGE スコア結果の平均値は図 4.6 に示す通りである．

実験結果　Rougeスコア

	ROUGE-1	ROUGE-2	ROUGE-L
方法1-RL動的計画法	0.505	0.327	0.389
方法1-文章価値ランキング	0.505	0.318	0.388
方法2-RL動的計画法	0.500	0.319	0.388
方法2-文章価値ランキング	0.505	0.325	0.397

図 4.6: ROUGE スコア

各方法の ROUGE スコアの結果から，それぞれの精度はかなり近い結果であったといえる．次に，各メインクラスの実験結果を報告する．

(4) 方法 1 と 2 を利用した時の各メインクラスの実験結果は図 4.7-図 4.20 に示す通りである．

方法1-研究背景

	深層学習の再現率	分類ー>文章数	要約長制限	抽出文章数	正解個数	正解個数強化学習なし	熟練者つけた重要度ラベル個数	深層学習分類が間違った	ROUGE-1	ROUGE-1強化学習なし	ROUGE-2	ROUGE-2強化学習なし	ROUGE-L	ROUGE-L強化学習なし
Test_1	0.66	15	5	5	0	0	2	1	0.259	0.233	0.03	0.04	0.132	0.144
Test_2	1	14	5	5	1	0	2	0	0.463	0.296	0.292	0.046	0.347	0.157
Test_3	0.571	8	3	3	1	1	3	0	0.503	0.503	0.408	0.408	0.474	0.474
Test_4	0.818	14	5	5	0	0	4	1	0.506	0.488	0.140	0.146	0.220	0.224
Test_5	0.285	3	2	2	2	2	3	1	0.781	0.781	0.768	0.768	0.781	0.781
Test_6	0.33	2	2	2	1	1	2	1	0.448	0.448	0.298	0.298	0.293	0.293
Test_7	1.0	8	3	3	1	1	2	0	0.590	0.590	0.448	0.448	0.416	0.416
Test_8	1.0	3	2	2	2	1	2	0	1	0.677	1	0.508	1	0.564
Test_9	0.3	2	2	2	1	1	3	2	0.550	0.550	0.449	0.449	0.507	0.507
Test_10	0.55	5	3	3	2	1	3	1	0.699	0.554	0.613	0.341	0.640	0.453
AVG	0.6514	7.4	3.2	3.2	1.1	0.8	2.6	0.7	0.5799	0.512	0.4446	0.3452	0.481	0.4013

図 4.7: 方法 1-研究背景

方法2-研究背景

	深層学習の再現率	分類ー>文章数	文章個数制限	抽出文章数	正解個数	正解個数強化学習なし	熟練者つけた重要度ラベル個数	深層学習分類が間違った	ROUGE-1	ROUGE-1強化学習なし	ROUGE-2	ROUGE-2強化学習なし	ROUGE-L	ROUGE-L強化学習なし
Test_1	0.66	15	5	5	0	0	2	1	0.277	0.233	0.058	0.04	0.141	0.144
Test_2	1	14	5	5	1	0	2	0	0.463	0.247	0.292	0.056	0.347	0.144
Test_3	0.571	8	3	3	1	1	3	0	0.591	0.613	0.407	0.409	0.520	0.527
Test_4	0.818	14	5	5	1	0	4	1	0.602	0.451	0.324	0.128	0.356	0.233
Test_5	0.285	3	2	2	2	2	3	1	0.781	0.781	0.768	0.768	0.781	0.781
Test_6	0.33	2	2	2	1	1	2	1	0.448	0.448	0.298	0.298	0.293	0.293
Test_7	1.0	8	3	3	1	1	2	0	0.590	0.590	0.448	0.448	0.416	0.416
Test_8	1.0	3	2	2	1	1	2	0	0.688	0.688	0.450	0.450	0.590	0.590
Test_9	0.3	2	2	2	1	1	3	2	0.550	0.550	0.449	0.449	0.507	0.507
Test_10	0.55	5	3	3	2	1	3	1	0.702	0.516	0.605	0.327	0.666	0.419
AVG	0.6514	7.4	3.2	3.2	1.1	0.8	2.6	0.7	0.5692	0.5117	0.4099	0.3373	0.4617	0.4054

図 4.8: 方法 2-研究背景

方法1-研究目的

	深層学習の再現率	分類ー>文章数	要約長制限	抽出文章数	正解個数	正解個数強化学習なし	熟練者つけた重要度ラベル個数	深層学習分類が間違った	ROUGE-1	ROUGE-1強化学習なし	ROUGE-2	ROUGE-2強化学習なし	ROUGE-L	ROUGE-L強化学習なし
Test_1	0.5	3	2	2	1	1	1	0	0.679	0.679	0.673	0.673	0.679	0.679
Test_2	0	2	2	2	0	0	2	2	0.517	0.517	0.220	0.220	0.344	0.344
Test_3	0.333	2	2	2	1	1	2	1	0.764	0.764	0.712	0.712	0.764	0.764
Test_4	0.2	2	2	2	1	1	2	1	0.542	0.542	0.377	0.377	0.480	0.480
Test_5	0.33	6	3	3	0	0	2	2	0.380	0.329	0.064	0.10	0.211	0.223
Test_6	1	2	2	2	1	1	1	0	0.897	0.897	0.896	0.896	0.897	0.897
Test_7	0.4	2	2	2	2	2	3	1	0.809	0.809	0.807	0.807	0.809	0.809
Test_8	1	3	2	2	1	1	1	0	0.713	0.752	0.707	0.747	0.713	0.752
Test_9	0.5	2	2	2	1	1	1	0	0.821	0.821	0.816	0.816	0.821	0.821
Test_10	1	2	2	2	1	1	1	0	0.649	0.649	0.639	0.639	0.649	0.649
AVG	0.5263	2.6	2.1	2.1	0.9	0.9	1.6	0.7	0.6771	0.6759	0.5911	0.5987	0.6367	0.6418

図 4.9: 方法 1-研究目的

方法2-研究目的

	深層学習の再現率	分類ー>文章数	文章個数制限	抽出文章数	正解個数	正解個数強化学習なし	熟練者つけた重要度ラベル個数	深層学習分類が間違った	ROUGE-1	ROUGE-1強化学習なし	ROUGE-2	ROUGE-2強化学習なし	ROUGE-L	ROUGE-L強化学習なし
Test_1	0.5	3	2	2	1	1	1	0	0.679	0.679	0.673	0.673	0.679	0.679
Test_2	0	2	2	2	0	0	2	2	0.517	0.517	0.220	0.220	0.344	0.344
Test_3	0.333	2	2	2	1	1	2	1	0.764	0.764	0.712	0.712	0.764	0.764
Test_4	0.2	2	2	2	1	1	2	1	0.542	0.542	0.377	0.377	0.480	0.480
Test_5	0.33	6	3	3	0	0	2	2	0.391	0.297	0.091	0.136	0.231	0.202
Test_6	1	2	2	2	1	1	1	0	0.897	0.897	0.896	0.896	0.897	0.897
Test_7	0.4	2	2	2	2	2	3	1	0.809	0.809	0.807	0.807	0.809	0.809
Test_8	1	3	2	2	1	1	1	0	0.713	0.752	0.707	0.747	0.713	0.752
Test_9	0.5	2	2	2	1	1	1	0	0.821	0.821	0.816	0.816	0.821	0.821
Test_10	1	2	2	2	1	1	1	0	0.649	0.649	0.639	0.639	0.649	0.649
AVG	0.5263	2.6	2.1	2.1	0.9	0.9	1.6	0.7	0.6782	0.6727	0.5938	0.6023	0.6387	0.6397

図 4.10: 方法 2-研究目的

方法1-研究方法

	深層学習の再現率	分類ー>文章数	文章個数制限	抽出文章数	正解個数	正解個数強化学習なし	熟練者つけた重要度ラベル個数	深層学習分類が間違った	ROUGE-1	ROUGE-1強化学習なし	ROUGE-2	ROUGE-2強化学習なし	ROUGE-L	ROUGE-L強化学習なし
Test_1	0.714	37	4	4	0	0	3	3	0.379	0.397	0.104	0.165	0.218	0.210
Test_2	0.86	56	5	5	0	0	3	1	0.492	0.544	0.232	0.243	0.276	0.268
Test_3	0.774	31	4	4	0	0	3	0	0.378	0.467	0.02	0.102	0.189	0.222
Test_4	0.595	39	4	3	0	0	3	1	0.245	0.304	0.03	0.07	0.163	0.173
Test_5	0.838	78	8	7	0	0	3	0	0.441	0.430	0.137	0.120	0.207	0.223
Test_6	1	22	4	4	0	0	2	0	0.395	0.395	0.082	0.082	0.192	0.192
Test_7	0.769	24	4	4	2	1	3	1	0.675	0.546	0.539	0.440	0.532	0.472
Test_8	0.7	22	4	3	1	1	3	0	0.559	0.559	0.402	0.402	0.509	0.509
Test_9	0.83	26	4	4	2	2	3	0	0.724	0.644	0.634	0.566	0.611	0.524
Test_10	0.954	31	4	4	0	0	3	0	0.429	0.429	0.217	0.217	0.281	0.281
AVG	0.8034	36.6	4.5	4.2	0.5	0.4	2.9	0.6	0.4717	0.4715	0.2397	0.2407	0.3178	0.3074

図 4.11: 方法 1-研究方法

方法2-研究方法

	深層学習の再現率	分類ー>文章数	文章個数制限	抽出文章数	正解個数	正解個数強化学習なし	熟練者つけた重要度ラベル個数	深層学習分類が間違った	ROUGE-1	ROUGE-1強化学習なし	ROUGE-2	ROUGE-2強化学習なし	ROUGE-L	ROUGE-L強化学習なし
Test_1	0.714	37	4	4	0	0	3	3	0.376	0.397	0.07	0.165	0.223	0.210
Test_2	0.86	56	5	5	0	0	3	1	0.520	0.554	0.214	0.275	0.278	0.280
Test_3	0.774	31	4	4	0	0	3	0	0.358	0.471	0.07	0.102	0.220	0.235
Test_4	0.595	39	4	4	0	0	3	1	0.356	0.09	0.388	0.10	0.201	0.214
Test_5	0.838	78	8	8	0	0	3	0	0.451	0.457	0.134	0.126	0.213	0.221
Test_6	1	22	4	4	0	0	2	0	0.415	0.456	0.110	0.163	0.218	0.255
Test_7	0.769	24	4	4	1	1	3	1	0.578	0.611	0.433	0.478	0.477	0.416
Test_8	0.7	22	4	4	1	1	3	0	0.554	0.5	0.384	0.341	0.478	0.389
Test_9	0.83	26	4	4	2	2	3	0	0.674	0.700	0.600	0.629	0.549	0.598
Test_10	0.954	31	4	4	0	0	3	0	0.429	0.382	0.217	0.187	0.281	0.226
AVG	0.8034	36.6	4.5	4.5	0.4	0.4	2.9	0.6	0.4711	0.4618	0.262	0.2566	0.3138	0.3044

図 4.12: 方法 2-研究方法

方法1-実験

	深層学習の再現率	分類ー>文章数	文章個数制限	抽出文章数	正解個数	正解個数強化学習なし	熟練者つけた重要度ラベル個数	深層学習分類が間違った	ROUGE-1	ROUGE-1強化学習なし	ROUGE-2	ROUGE-2強化学習なし	ROUGE-L	ROUGE-L強化学習なし
Test_1	0	3	2	2	0	0	0	0	0	0	0	0	0	0
Test_2	0.875	33	6	5	1	1	3	0	0.449	0.444	0.287	0.242	0.311	0.342
Test_3	1	22	5	5	1	1	3	0	0.546	0.554	0.309	0.313	0.393	0.425
Test_4	0	7	2	2	0	0	0	0	0	0	0	0	0	0
Test_5	0.678	33	6	6	1	0	3	2	0.429	0.339	0.234	0.10	0.276	0.187
Test_6	1	9	3	3	1	0	2	0	0.623	0.319	0.573	0.085	0.608	0.235
Test_7	0.7	13	3	3	1	0	2	1	0.558	0.260	0.425	0.01	0.527	0.173
Test_8	0.67	9	3	3	0	0	1	1	0.315	0.315	0.152	0.152	0.246	0.246
Test_9	0.769	17	4	3	1	2	3	0	0.537	0.749	0.348	0.627	0.402	0.653
Test_10	0.5	5	2	2	1	1	2	1	0.647	0.647	0.575	0.575	0.588	0.588
AVG	0.6192	15.1	3.6	3.4	0.7	0.5	1.9	0.5	0.513	0.453375	0.362875	0.263	0.418875	0.356125

図 4.13: 方法 1-実験

方法2-実験

	深層学習の再現率	分類ー>文章数	文章個数制限	抽出文章数	正解個数	正解個数強化学習なし	熟練者つけた重要度ラベル個数	深層学習分類が間違った	ROUGE-1	ROUGE-1強化学習なし	ROUGE-2	ROUGE-2強化学習なし	ROUGE-L	ROUGE-L強化学習なし
Test_1	0	3	2	2	0	0	0	0	0	0	0	0	0	0
Test_2	0.875	33	6	5	1	0	3	0	0.458	0.365	0.262	0.119	0.311	0.237
Test_3	1	22	5	4	1	1	3	0	0.550	0.546	0.339	0.350	0.419	0.446
Test_4	0	7	2	2	0	0	0	0	0	0	0	0	0	0
Test_5	0.678	33	6	6	1	0	3	2	0.425	0.340	0.235	0.08	0.269	0.2
Test_6	1	9	3	2	1	1	2	0	0.823	0.823	0.78	0.78	0.823	0.823
Test_7	0.7	13	3	3	1	0	2	1	0.558	0.260	0.425	0.01	0.527	0.173
Test_8	0.67	9	3	3	0	0	1	1	0.315	0.315	0.152	0.152	0.246	0.246
Test_9	0.769	17	4	4	1	2	3	0	0.447	0.610	0.289	0.496	0.347	0.519
Test_10	0.5	5	2	2	1	1	2	1	0.647	0.647	0.575	0.575	0.588	0.588
AVG	0.6192	15.1	3.6	3.3	0.7	0.5	1.9	0.5	0.527875	0.48825	0.382125	0.32025	0.44125	0.404

図 4.14: 方法 2-実験

方法1-結果評価

	深層学習の再現率	分類ー>文章数	文章個数制限	抽出文章数	正解個数	正解個数強化学習なし	熟練者つけた重要度ラベル個数	深層学習分類が間違った	ROUGE-1	ROUGE-1強化学習なし	ROUGE-2	ROUGE-2強化学習なし	ROUGE-L	ROUGE-L強化学習なし
Test_1	1	7	3	3	0	0	1	0	0.268	0.268	0	0	0.151	0.151
Test_2	0.538	18	4	3	0	0	3	1	0.436	0.475	0.147	0.135	0.230	0.231
Test_3	1	18	4	3	0	0	0	0	0	0	0	0	0	0
Test_4	0	5	2	2	0	0	0	0	0	0	0	0	0	0
Test_5	0.826	47	8	8	1	1	4	0	0.577	0.6	0.330	0.352	0.362	0.4
Test_6	0.8	9	3	3	0	0	2	1	0.110	0.110	0.016	0.016	0.110	0.110
Test_7	1	6	3	3	1	1	2	0	0.455	0.431	0.141	0.162	0.158	0.227
Test_8	1	11	3	3	0	0	2	0	0.168	0.303	0.02	0.123	0.09	0.227
Test_9	0.636	9	3	3	1	1	2	0	0.463	0.422	0.4	0.202	0.463	0.253
Test_10	0.8	7	3	2	1	1	3	1	0.421	0.618	0.365	0.437	0.374	0.546
AVG	0.76	13.7	3.6	3.3	0.4	0.4	1.9	0.3	0.36225	0.403375	0.177375	0.178375	0.24225	0.268125

図 4.15: 方法 1-結果評価

方法2-結果評価

	深層学習の再現率	分類ー>文章数	文章個数制限	抽出文章数	正解個数	正解個数強化学習なし	熟練者つけた重要度ラベル個数	深層学習分類が間違った	ROUGE-1	ROUGE-1強化学習なし	ROUGE-2	ROUGE-2強化学習なし	ROUGE-L	ROUGE-L強化学習なし
Test_1	1	7	3	3	0	0	1	0	0.268	0.229	0	0	0.151	0.147
Test_2	0.538	18	4	3	0	0	3	1	0.436	0.402	0.147	0.076	0.230	0.217
Test_3	1	18	4	3	0	0	0	0	0	0	0	0	0	0
Test_4	0	5	2	2	0	0	0	0	0	0	0	0	0	0
Test_5	0.826	47	8	8	0	0	4	0	0.497	0.488	0.153	0.166	0.243	0.228
Test_6	0.8	9	3	3	0	0	2	1	0.110	0.19	0.016	0.03	0.110	0.121
Test_7	1	6	3	3	0	1	2	0	0.384	0.431	0.05	0.162	0.17	0.227
Test_8	1	11	3	3	0	0	2	0	0.146	0.267	0.01	0.09	0.078	0.186
Test_9	0.636	9	3	3	1	1	2	0	0.463	0.422	0.4	0.202	0.463	0.253
Test_10	0.8	7	3	2	1	1	3	1	0.421	0.618	0.365	0.437	0.374	0.546
AVG	0.76	13.7	3.6	3.3	0.2	0.3	1.9	0.3	0.340625	0.380875	0.142625	0.145375	0.227375	0.240625

図 4.16: 方法 2-結果評価

方法1-知見

	深層学習の再現率	分類ー>文章数	文章個数制限	抽出文章数	正解個数	正解個数強化学習なし	熟練者つけた重要度ラベル個数	深層学習分類が間違った	ROUGE-1	ROUGE-1強化学習なし	ROUGE-2	ROUGE-2強化学習なし	ROUGE-L	ROUGE-L強化学習なし
Test_1	0.5	6	3	3	1	1	2	1	0.546	0.574	0.408	0.424	0.360	0.395
Test_2	0.33	4	3	2	0	1	2	1	0.517	0.75	0.174	0.645	0.284	0.671
Test_3	0.833	6	3	3	1	2	3	0	0.632	0.791	0.408	0.687	0.559	0.781
Test_4	0.8	12	6	6	0	0	3	0	0.397	0.4	0.114	0.123	0.198	0.219
Test_5	0.6	8	4	4	0	0	2	1	0.495	0.418	0.144	0.123	0.209	0.234
Test_6	1	4	3	3	0	1	1	0	0.337	0.666	0.04	0.658	0.192	0.666
Test_7	0.8	4	3	2	1	1	2	0	0.646	0.646	0.536	0.536	0.606	0.606
Test_8	1	3	3	3	2	2	2	0	0.813	0.813	0.811	0.811	0.813	0.813
Test_9	0.75	5	3	3	1	1	2	0	0.591	0.591	0.477	0.477	0.486	0.486
Test_10	1	4	3	2	0	0	1	0	0.377	0.377	0.173	0.173	0.283	0.283
AVG	0.7613	5.6	3.4	3.1	0.6	0.9	2	0.3	0.5351	0.6026	0.3285	0.4657	0.399	0.5154

図 4.17: 方法 1-知見

方法2-知見

	深層学習の再現率	分類ー>文章数	文章個数制限	抽出文章数	正解個数	正解個数強化学習なし	熟練者つけた重要度ラベル個数	深層学習分類が間違った	ROUGE-1	ROUGE-1強化学習なし	ROUGE-2	ROUGE-2強化学習なし	ROUGE-L	ROUGE-L強化学習なし
Test_1	0.5	6	3	3	0	1	2	1	0.386	0.578	0.089	0.424	0.232	0.395
Test_2	0.33	4	3	2	0	1	2	1	0.517	0.75	0.174	0.645	0.284	0.671
Test_3	0.833	6	3	3	1	2	3	0	0.632	0.791	0.408	0.687	0.559	0.781
Test_4	0.8	12	6	6	0	1	3	0	0.377	0.456	0.108	0.317	0.188	0.379
Test_5	0.6	8	4	4	0	0	2	1	0.495	0.418	0.144	0.123	0.209	0.234
Test_6	1	4	3	3	0	1	1	0	0.337	0.666	0.04	0.658	0.192	0.666
Test_7	0.8	4	3	2	1	1	2	0	0.646	0.646	0.536	0.536	0.606	0.606
Test_8	1	3	3	3	2	2	2	0	0.813	0.813	0.811	0.811	0.813	0.813
Test_9	0.75	5	3	3	1	1	2	0	0.591	0.591	0.477	0.477	0.486	0.486
Test_10	1	4	3	2	0	0	1	0	0.377	0.377	0.173	0.173	0.283	0.283
AVG	0.7613	5.6	3.4	3.1	0.5	1	2	0.3	0.5171	0.6086	0.296	0.4851	0.3852	0.5314

図 4.18: 方法 2-知見

方法1-関連研究

	深層学習の再現率	分類ー>文章数	文章個数制限	抽出文章数	正解個数	正解個数強化学習なし	熟練者つけた重要度ラベル個数	深層学習分類が間違った	ROUGE-1	ROUGE-1強化学習なし	ROUGE-2	ROUGE-2強化学習なし	ROUGE-L	ROUGE-L強化学習なし
Test_1	0.25	4	3	3	0	0	2	2	0.357	0.484	0.07	0.132	0.220	0.272
Test_2	0.3	9	3	2	0	0	2	1	0.345	0.345	0.084	0.084	0.167	0.167
Test_3	0	5	2	2	0	0	0	0	0	0	0	0	0	0
Test_4	0.571	11	4	3	1	1	3	2	0.567	0.552	0.409	0.372	0.367	0.404
Test_5	0.312	9	3	3	0	0	2	2	0.429	0.411	0.112	0.09	0.195	0.183
Test_6	0	2	2	2	0	0	1	1	0.371	0.371	0.147	0.147	0.226	0.226
Test_7	0	3	2	2	0	0	0	0	0	0	0	0	0	0
Test_8	1	2	2	2	0	0	0	0	0	0	0	0	0	0
Test_9	0.55	7	3	3	0	0	2	2	0.335	0.342	0.07	0.06	0.208	0.224
Test_10	1	2	2	2	0	0	0	0	0	0	0	0	0	0
AVG	0.3983	5.4	2.6	2.4	0.1	0.1	1.2	1	0.4066	0.4175	0.1486	0.1475	0.2305	0.246

図 4.19: 方法 1-関連研究

方法2-関連研究

	深層学習の再現率	分類ー>文章数	文章個数制限	抽出文章数	正解個数	正解個数強化学習なし	熟練者つけた重要度ラベル個数	深層学習分類が間違った	ROUGE-1	ROUGE-1強化学習なし	ROUGE-2	ROUGE-2強化学習なし	ROUGE-L	ROUGE-L強化学習なし
Test_1	0.25	4	3	3	0	0	2	2	0.357	0.484	0.07	0.132	0.220	0.272
Test_2	0.3	9	3	2	0	0	2	1	0.324	0.324	0.045	0.045	0.178	0.178
Test_3	0	5	2	2	0	0	0	0	0	0	0	0	0	0
Test_4	0.571	11	4	3	1	1	3	2	0.619	0.574	0.464	0.413	0.502	0.447
Test_5	0.312	9	3	3	0	0	2	2	0.44	0.429	0.112	0.10	0.195	0.195
Test_6	0	2	2	2	0	0	1	1	0.371	0.371	0.147	0.147	0.226	0.226
Test_7	0	3	2	2	0	0	0	0	0	0	0	0	0	0
Test_8	1	2	2	2	0	0	0	0	0	0	0	0	0	0
Test_9	0.55	7	3	3	0	0	2	2	0.335	0.342	0.07	0.06	0.208	0.224
Test_10	1	2	2	2	0	0	0	0	0	0	0	0	0	0
AVG	0.3983	5.4	2.6	2.4	0.1	0.1	1.2	1	0.407	0.420	0.151	0.1495	0.254	0.257

図 4.20: 方法 2-関連研究

メインクラス「研究背景」の結果から，方法 1 と 2 は近い結果になったが，強化学習動的計画法の方は良い認識率と ROUGE スコアを得た．重要度ラベルの認識率は約 42%であり，深層学習の誤認識考慮しない場合の正解率は約 57%であった．

メインクラス「研究目的」の結果から，全部の方法で近い結果になった．要約候補文の数が少ないため，強化学習なしで処理する場合が多くなり，文章価値のみによるランキングで抽出される傾向があった．重要度ラベルの認識率は約 56%であり，深層学習の誤認識を考慮しない場合には全てが正解であった．

メインクラス「研究方法」の結果については，方法 1+強化学習動的計画法がやや良い認識率を得たが，全方法の ROUGE スコアはほぼ同じであった．重要度ラベルの認識率は約 15%であり，深層学習の誤認識を考慮しない場合の正解率は約 21%であった．

メインクラス「実験」の結果では，全方法の認識率は同じであったが，方法 2 はやや良い ROUGE スコアを得た．そして，強化学習動的計画法の方が良い認識率と ROUGE スコアであった．認識率は約 37%であり，深層学習の誤認識を考慮しない場合の正解率は約 50%であった．

メインクラス「結果評価」の結果によると，方法 1 は良い認識率であり，ROUGE スコアも方法 2 より高くなった．なお，強化学習なしの手法はより高い ROUGE スコアを得た．重要度ラベルの認識率は約 21%であり，深層学習の誤認識を考慮しない場合の正解率は約 25%であった．

メインクラス「知見」の結果では，強化学習なしの手法はより高い重要度ラベルの認識率と ROUGE スコアを得た．重要度ラベルの認識率は約 50%であり，深層学習の誤認識を考慮しない場合の正解率は約 59%であった．

メインクラス「関連研究」の結果については，深層学習の高い誤認識 (83.3%) が主な原因で，低い重要度ラベル認識率となった．

4.6.1 論文内容の影響

図 4.4 と 4.5 の結果から，平均 14.1 個の重要度ラベルのうち 4.1 個の間違いが深層学習の誤認識によるものであった．特に，深層学習による誤認識率が高い論文の分野は音声認識と教育工学であり，訓練用データの数が少ないことによる学習不足が原因となったと考えられる．具体的には，文章価値を算定する際に，学習不足により重要語の判断が間違えるケースが多くなるため，文章価値をうまく推定できないものと考えられる．

4.6.2 強化学習＋動的計画法の特徴

本節では，訓練不足と深層学習の誤認識を除いた強化学習+動的計画法の特徴について分析する．強化学習を利用した実験結果を分析するため，強化学習のみで正解になった文章と文章価値ランキングのみで正解になった文章を比較した．具体的な正解の特徴は図 4.22 と図 4.23 に示す通りである．

RL動的計画法-正解の特徴

文章	分野	メインクラス	文字列長さ	対応したメインクラス文章群の平均文字列長さ
自動音声認識は,スマートフォンの普及とともにインタフェースの一つとして以前とは比較にならないほど身近なものとなってきた	音声認識	背景	58	100
提案手法の有効性をAurora-2による連続数字読み上げ認識実験により評価した	音声認識	実験	33	57
このことは,制約条件や要素技術の種類や優先度が異なる実践事例の知見を,他のケースでそのまま適用することが困難であることを意味する	教育工学	背景	62	73
そのため,評価対象音声データはCSJのCore177講演の約30時間分とSDPWS104講演の約28時間分を使用した	言語処理	実験	42	59
近年,強化学習を使用した研究が数多く行われている	強化学習	背景	23	42

図 4.21: RL 動的計画法-正解の特徴

文章価値ランキング-正解の特徴

文章	分野	メインクラス	文字列長さ	対応したメインクラス文章群の平均文字列長さ
また,ニューラルネットワークを回帰モデルとして用いるDAEと比較した場合,ニューラルネットワークのもつ非線形性によって実現される複雑なモデル化と,静音環境下における音声特徴量のモデルとして用いたガウス分布のもつ汎化性能を組み合わせることで,既知雑音条件で低い誤り率を維持しつつ,雑音環境が未知な条件で(Decimal)%の単語誤り率の削減が可能となった	音声認識	知見	162	115
また, 提案手法により獲得されるネットワークの識別精度とハイパーパラメータ最適化フレームワークであるOptunaとHyperoptにより獲得されるネットワークの識別精度を比較した	画像認識	知見	75	68
本稿では,遠隔教育プログラムにおけるデザインパターンを構築することを目的として,コミュニケーションパターンモデルに基づき,遠隔教育における制約条件及び遠隔講義システムの構成要素をUML の表記法に従って整理する枠組みを示した	教育工学	知見	106	86

図 4.22: 文章価値ランキング-正解の特徴

強化学習にはペナルティ設定を加えたため，重要度をある程度保障しながら短い文章を選択する傾向があると考えられる．本研究のテスト論文においては，メインクラス「研究背景」・「実験」の重要文はそういった特徴を強く持っていたことによって，強化学習+動的計画法の方が良い認識率と ROUGE スコアとなったと考えられる．なお，情報の冗長性と要約文章類似度を考えていない文章価値ランキングの方法はメインクラス「知見」の重要文の特徴を当てはまりやすかった．その理由としては，メインクラス「知見」は結論とまとめに関連する文章が重要文になりやすいため，テスト論文そういう特徴を持っていた長文が数多く占めし，強化学習は長文の重要度を把握にくいため，重要度ラベルを外れた場合が多いと考えられる．

第5章　おわりに

5.1　まとめ

研究者や学生にとって，学術論文の観点を反映した要約を生成することは，より効率的な研究活動を行う上で重要な課題である．そこで本論文では，学術論文を対象として観点を反映した要約を生成するため，観点に対応するメインクラスを付与した小規模な教師データを利用して，深層学習による分類タスクを行った．そして，深層学習により分類されたメインクラスの内容を反映した文章群の重要度を測定するために強化学習および DP 動的計画法に基づく重要文抽出手法を提案した．具体的な手順は以下の通りである．
(1)Google scholar から 100 編の日本語の学術論文を収集した．
(2)Apache Tika を用い，PDF の中身にあるテキストを認識した．
(3) 正規表現式と手作業を用い，学術論文における本文だけのテキストを抽出した．
(4) 文章のメインクラス，サブクラス，重要度ラベルというアノテーションを付けたデータセットを作成した．
(5)Word2vec と PV-DM の事前学習による単語埋め込みと文章埋め込みをした．
(6)LSTM と SVM の分類器を利用し，深層学習を行った．
(7) 深層学習の精度向上を実現するため，Combined-method と Cos 類似度を利用し，分類結果を改善した．
(8) 強化学習の要素における二つ文章価値の計算方法を挙げた．
(9) 強化学習とナップサック DP に組み合わせる手法を用い，抽出された重要文は要約として生成した．

実験については，まず，単に Word2vec と PV-DM での事前学習深層学習の分類結果をベースにし，熟練者から付けたアノテーションを利用する強化学習と動的計画法の手法を組み合わせる実験を行った．比較実験として文章価値ランキングの方法を加入した．そうした抽出された要約を対象とし，重要度ラベル認識精度と ROUGE スコアを検証した．その結果としては，全体的の重要度ラベル認識率は約 30%であり，深層学習の誤認識を考慮しない場合の認識率は約 40%であった．また，各方法の ROUGE スコアは約 0.5 であった．また，読者により直感的な要約を示すできる GUI 及論文内で可視化する Highlight システムを開発した．

上記の流れで構成した VPRAS システムを開発したことにより，研究者や学生の学術論文サーベイに対する観点を反映した自動抽出要約を生成することができるため，研究活動をより効率的に進行できると期待される．

5.2 今後の課題

5.2.1 データセット

本研究で使用したデータセットは分野による文章数の差が多少あった，特に音声認識と教育工学分野の訓練用データが少ないことが原因で深層学習と強化学習の精度に影響があったと思われる．今後はデータセットを準備する際に，各分野の文章数のバランスを取ることが重要であると考えられる．

5.2.2 正規表現

本論文で用いている正規表現を改善する余地がある．具体的には，正規表現が不十分であることおよび図と表の情報が混在していることから，Highlight 処理における文字列のマッチ精度が低下するという問題が見られた．こうした問題を解決することについても今後検討する必要があると考えられる．

5.2.3 深層学習の分類部分

図 4.4 と図 4.5 から 14.1 個の重要度ラベルのうち 4.1 個が深層学習の誤認識により抽出されなかった．29%の誤認識結果から考えると，本研究で使用した強化学習部分の重要文抽出精度は深層学習の分類精度に依存しているといえる．深層学習と強化学習を単独に実行する代わりに，深層学習をトレーニングしながら同時に強化学習の価値関数を設定するという深層強化学習の手法を取り入れることは今後の課題一つとなる．

5.2.4 強化学習の抽出部分

(1) 文章価値の計算方法:実験結果より，方法 1 と 2 で全てのラベルで間違ったケースが半分以上があるため，重要文としての文章価値を把握するのが不十分であり，その計算方法を改善する余地があるといえる．例えは，メインクラスの中にある文章群は訓練用データとし，サブクラスの深層学習分類を行いながら，熟練者から付けた重要度ラベルの情報に基づく強化学習の報酬設定も同時に行うという学習が一つの考え方である．

(2) 観点毎の方法設定:本研究で行った強化学習ベースと文章価値ランキングベースの実験結果 (図 4.7-4.20) によると，強化学習ありの方が研究背景と実験の重要度ラベル認識精度と ROUGE スコアは強化学習なしより良い結果が得られた．一

方で，結果評価と知見については強化学習なしの方が良い結果が得られた．今後は観点に分けて強化学習を扱うことを検討する必要がある．

謝辞

本研究では，ご指導を頂いた主指導教員長谷川忍准教授，助言を頂いた太田光一助教に心から感謝致します．また，データセットのアノテーション部分は全部長谷川准教授により付けていただき，誠に感謝いたします．

研究業績

LI JINGHONG，太田 光一，長谷川 忍. 観点を反映した深層学習及び強化学習による学術論文の自動要約生成，電子情報通信学会，2020.9.

参考文献

[1] 相田達哉，喜多村正仁．学術論文自動要約におけるオントロジーを用いた精度向上策．経営情報学会 全国研究発表大会要旨集，2017.

[2] 長谷川忍，柏原昭博．研究活動を支援するポータルサイトの構築．教育システム情報学会第 31 回全国大会講演論文集，pp.325-326,（2006）COLING ’12,pp.663-678,2012.

[3] 強化学習を用いた自動要約における基礎研究，慶留間諒大，當間愛晃．第 75 回全国大会講演論文集, 2013.

[4] Danish Contractor,Yufan Guo,Anna Korho-nen. Using argumentative zones for extractive summarization of scientic articles.The COLING 2012 Organizing Committee, pp.663–678.

[5] 中須賀謙吾，鶴岡慶雅．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 年 3 月).

[6] Jianpeng Cheng and Mirella Lapata. Neural sum-marization by extracting sentences and words. In ACL, pp. 484-494, 2016.

[7] 石垣達也，高村大也，奥村学．複数文質問を対象とした抽出型および生成型要約 自然言語処理 Vol.26 No.1, 2019.

[8] 平井久貴，新妻弘崇，太田学，高須淳宏．学術論文からの実験情報抽出の一手法．DEIM Forum 2015 F3-1.

[9] 神谷賢太郎，原寛紀，青山幹雄．深層学習によるレビュー内の重要語に着目した要約方法の提案と評価．第 82 回全国大会講演論文集，pp.463-464，2020.2.

[10] 梁成基，阿部川武．強化学習によるテキスト自動要約手法の提案．言語処理学会第 18 回年次大会発表論文集, pp.1067-1070,2012.

[11] 富田紘平，高村大也，奥村学．重要文抽出と文圧縮を組み合わせた新たな抽出的要約手法．情報処理学会研究報告. 情報学基礎研究会報告，一般社団法人情報処理学会，pp.13-20，2009.

[12] 泉田 啓, 天野 恒佑, 強化学習における線形計画法を用いた効率的解法, 計測自動制御学会論文集, 2016, 52 巻, 10 号, p. 566-572

[13] 森 信介, 自然言語処理における分野適応 (< 特集 > 知識の転移), 人工知能, 2012, 27 巻, 4 号, p.365-372.

[14] 福田清人，森直樹，松本啓之亮．LSTM を用いた文の分散表現の獲得手法に関する一考察．言語処理学会 第 24 回年次大会 発表論文集, pp.1195-1198,(2018 年 3 月).

[15] 赤石 雅典, 江澤 美保. 現場で使える，Python 自然言語処理入門，p.276.

[16] Gyoung Ho Lee, Kong Joo Lee. Automatic Text Summarization Using Reinforcement Learning with Embedding Features. Information Retrieval Knowledge Engineering Lab.Chungnam National University,Daejeon, Korea.

[17] Kazuko TAKAHASHI, Hiroya TAKAMURA, Manabu OKUMURA, Classification of Responses to Open-ended Questions with Machine Learning and Hand-Crafted Rules, Sociological Theory and Methods, 2004, Volume 19, Issue 2, Pages 177-195.

[18] 伊藤 多一，今津 義充，須藤 広大，仁ノ平 将人，川﨑 悠介，酒井 裕企，魏 崇哲．現場で使える！ Python 深層強化学習入門 強化学習と深層学習による探索と制御．p.24-25.

[19] TF-IDF Cos 類似度推定法.
https://qiita.com/nmbakfm/items/6bb91b89571dd68fcea6

[20] 小池康晴，鮫島和行．強化学習の基礎．東京工業大学 精密工学研究所，科学技術振興事業団 ERATO 川人学習動態脳プロジェクト.

[21] 中島 裕貴, 飯間 等, 強化学習の概念を導入したメタヒューリスティック法による組合せ最適化, システム制御情報学会論文誌, 2019, 32 巻, 2 号, p. 79-86.

[22] 機械学習① SVM (Support Vector Machine) まとめ
https://qiita.com/kibinag0/items/4a5725c65b5699741b3a

[23] Chin-Yew Lin and Eduard Hovy,Automatic Evaluation of Summaries Using N-gram Co-Occurrence Statistics．Information Sciences Institute University of Southern California.Proceedings of HLT-NAACL 2003,Main Papers , pp. 71-78,Edmonton, May-June 2003.