

論文要旨を捉えたスライド生成における 割り当て問題の適用

久保谷 善記¹ 品川政太郎² 山本晋太郎¹ 鈴木亮太³ 森島繁生⁴

¹ 早稲田大学 ² 奈良先端科学技術大学院大学

³ 産業技術総合研究所 ⁴ 早稲田大学理工学術院総合研究所

yoshikikubotani@akane.waseda.jp sei.shinagawa@is.naist.jp

s.yamamoto@fuji.waseda.jp ryota.suzuki@aist.go.jp shigeo@waseda.jp

概要

論文の要点を自身の目的に合わせてプレゼンテーションスライドにまとめることは、情報の発信や共有といった観点から重要性を増してきているが、試行錯誤を求められ労力がかかる問題がある。そこで本研究では、スライド作成者が、タイトルを指定することで、インタラクティブに論文からスライドテキストを作成できるシステムの実現を目指す。本システムは、与えられたスライドタイトルから論文の本文を検索して抽出するモジュールと、検索した文章を端的に要約するモジュールで構成される。このような要件を満たす既存手法である Document to Slide (D2S) と比べて、提案手法では検索文のスライドへの割り当てを考慮することで、スライド間でのテキストの内容の重複を抑えることに成功した。

1 はじめに

要点をまとめて情報発信を行う道具として、プレゼンテーションスライド（スライド）と呼ばれる紙芝居形式の資料を作成することが日常的に行われてきている。例えば、学会などの研究集会の場では、研究者は自身の研究成果を発表するために、論文を提出するだけでなく、スライドを用いた口頭発表により研究の要点を伝えることが求められる。また、学問分野全体において論文の数が増え続けており [1]、論文の著者ではない人々が、論文の要点をスライドにまとめ、勉強会の場や Web 上で発信することで、効率的に知見を共有し合うといったことが行われてきている。近年では、Qiita¹⁾ のスライド機能や、SlideShare²⁾ といった Web 上での共有プラットフォームが充実しており、誰でも手軽にスライドに

よる知識共有ができるようになってきている。

しかし、論文などの技術的文書のスライド作成は、経験と技能が求められる。具体的には、発信者（スライド作成者）がどのような選考・目的を持って論文の情報を発信するのか、口頭発表による説明を前提とするか否か、発表時間やスライドのページ数に制限があるかどうか、想定する読者・聴衆はどの程度その論文についての前提知識を持つか、などの条件を考慮する必要がある。これらはスライド作成に慣れていない初心者には難しい作業であり、経験を積んだ熟達者でも、一度大枠を作成した上で前述の条件を参照して微調整を繰り返したり、時には条件自体を修正して作り直すといったことを行う。つまり、スライド化の作業は少なくない労力をかけて試行錯誤を繰り返す必要のある作業だと言える。

本研究では、このようなスライド作成の難点を低減する方法として、論文をスライド化したいスライド作成者が、目的に合わせてスライドのタイトルやキーワードのみを設定するだけで、それらの条件に沿って半自動で論文をスライド化するインタラクティブなシステムを提案する。本システムの最終的な目標は、スライド作成者がこのシステムを使うことによって、論文をどのようにスライド化するか手軽に試行錯誤しながらたたき台を作成できるようになることである。

このようなインタラクティブなスライド生成システムに求められる要件には、①スライドに記載されるテキストが、スライド作成者によって指定されたタイトルに沿っていること、②そのテキストが端的にまとまっていること、③内容の重複が必要以上に起こっていないこと、の三つが挙げられる。論文の図や文章のスライドへの配置の仕方も最終的には考慮すべき重要な課題であるが、たたき台として大ま

1) <https://qiita.com/>

2) <https://www.slideshare.net/>

かな構成を決めるための試行錯誤の段階では考慮しなくとも大きな影響が無いと考え、本研究ではテキストのみに着目する。

スライドのタイトルに沿ったテキストを生成するためには、タイトルと論文本文の内容の相関を適切に測りとり、関連する文章のみを論文から抽出する検索器が必要である。また、端的にまとめたテキストを生成するには原文の重要な情報をとらえつつ冗長な部分を排除する要約モデルが求められる。スライドタイトルに相関のある文章を論文から抽出し、要約したテキストを生成する手法として Document to Slide (D2S) [2] が提案されているが、ページ間で過分に検索文が重複してしまうという問題があり、本研究の要求である、必要以上に内容の重複がないことを満たせない。そこで、本研究ではページ間の内容の重複を解消しつつ要約スライドを生成する手法を提案する。本研究では検索部分で新たに組合せ最適化の割り当て問題を解くことで、異なるスライド間で内容が被らないようにスライドを生成できることを示す。

2 関連研究

プレゼンテーションを自動で生成しようとする研究は 2000 年ごろから存在する。初期のスライド生成の研究では、文章の形態素に着目してテキストからスライド用の箇条書きテキストを生成する研究 [3] や、指定したトピックで Web サイトを検索してルールベースでスライドテキストを作成する研究 [4] など、ヒューリスティックな設計に依存するモデルが主流であった。また、後年には論文からのスライド生成を対象を絞った研究 [5-7] も登場した。これらの研究では、主に回帰モデルを用いて論文内の文章間の重要度をスコアリングし、整数線形計画法 (ILP) によってスコアが最大となるような文章を抽出してスライドのテキストとしていた。深層学習が主流となった近年では、文章の重要度を得るモデルとして畳み込みニューラルネットワークやリカレントニューラルネットワークを使用する研究 [8] が現れたほか、系列データの相関を捉えることに長けた self-attention を用いた言語モデルを利用した研究 [2,9] も存在する。

3 提案手法

はじめに、スライドタイトルに沿った内容を論文から抽出し、それらを端的にまとめたテキストを

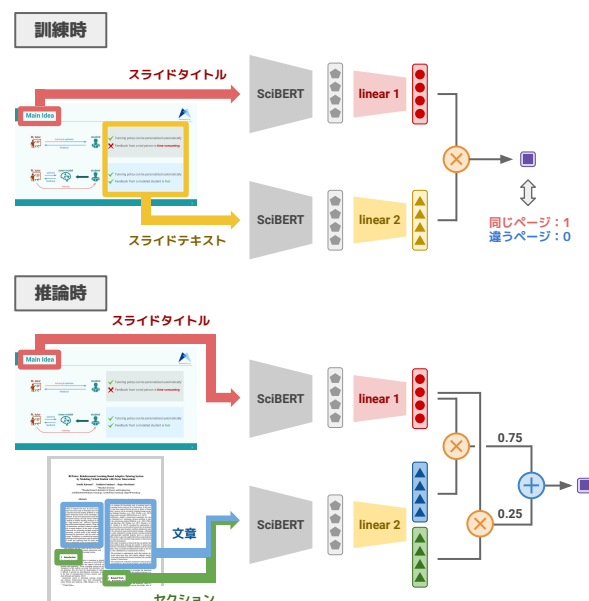


図 1: D2S の文埋め込みモデルの概要図

生成する、という目的のため本研究が参考にした Document to Slide (D2S) [2] を説明する。

D2S におけるスライド生成は、スライドページのタイトルをクエリとして論文内の関連する文章を検索する段階と、検索した文章をスライドに載せる上で適切な長さに要約する段階の 2 段階に分かれている。前半の文章検索の部分では、まずスライドのタイトルとテキストを用いて文の埋め込み表現を獲得する文埋め込みモデルを対照学習し、推論時にはスライドのタイトルと論文内の文章を入力して類似度を計測する。この文埋め込みモデルには、学術的な文書に特化して事前訓練を行なった SciBERT [10] を採用し、出力の CLS トークンを FC 層により 128 次元に圧縮することで埋め込み表現とした。図 1 に文埋め込みモデルの概要を示す。その後、各スライドタイトルごとに \cos 類似度の値が高い文章上位 10 個を抽出し、検索結果とする。後半の要約では、検索された文章からスライドに載るテキストを生成できるように BART [11] をファインチューニングして使用する。

しかし、D2S の検索のように類似度の高いものを常に 10 個取得する方法だと、同じ文章が複数のスライドタイトルと結び付けられてしまう可能性がある。これは異なるスライドページで同じ内容が何度も述べられることを意味しているため検索の仕方として不適切である。本研究ではこの検索部分に制約条件付きの割り当て問題を追加することで、スライドタイトルに関連する論文内の文章の重複を抑える

方法を提案する。この割り当て問題は、次のように定式化する。

各スライドのタイトル $q_i \in \mathbb{Q}$ に対して、論文内の文章 $s_j \in \mathcal{S}$ を割り当ててることを考える。このとき、前述の SciBERT を用いて、論文の各文・その文章の属するセクションタイトル・スライドのタイトルの分散表現 E_s, E_t, E_q を作成する。

$$E_s = \text{linear}_s(\text{SciBERT}(s_j)) \quad (1)$$

$$E_t = \text{linear}_s(\text{SciBERT}(t_j)) \quad (2)$$

$$E_q = \text{linear}_q(\text{SciBERT}(q_i)) \quad (3)$$

ここで t_j は、論文内の文章 s_j が属するセクションのタイトルを表す。式 (1)–(3) で計算した分散表現をもとに、スライドタイトルと論文の文章間の \cos 類似度は $c_{i,j}$ は次式 (4) のように計算される。

$$c_{i,j} = (0.75E_s + 0.25E_t) \cdot E_q \quad (4)$$

また、割り当てるか否かのバイナリ変数を $x_{i,j} \in \{0, 1\}$ とすると、最適化時の目的関数は以下のように定義できる。

$$\sum_j \sum_i c_{i,j} x_{i,j} \quad (5)$$

この最適化問題の目的は、重複を抑えつつスライドタイトルに関連する論文内の文章を割り当てることであるため、以下の二つの制約を加える。

$$\sum_i x_{i,j} \leq N \quad \text{for } \forall j \quad (6)$$

$$\sum_j x_{i,j} \leq M \quad \text{for } \forall i \quad (7)$$

式 (6) の制約は一つの文章が 3 個以上のスライドに割り当てられないようにする制約、式 (7) の制約は一つのスライドに割り当てられる文章の最大数を 10 個にする制約である。本研究では $N = 3, M = 10$ として実験を行った。

4 実験設定

提案手法の有効性を確かめるため、2つの実験を行った。1つ目の実験は、指定されたスライドタイトルから、論文中の関連する文章を検索して抽出するモジュールの性能を評価するための実験である。各スライドタイトルにつき、上位 10 件の文章を検索する場合と、割り当て問題を最適化した場合で抽出される文章とで比較した。評価指標には、既存手法と同様に逆ドキュメント頻度 (IDF) によって重み付けした ROUGE スコア [2] (付録 A) を用い、論文

表 1: 抽出結果の IDF 重み付け ROUGE スコア

検索手法	P	R	F	平均抽出数
上位 10 個を抽出	1.147	3.682	1.580	10.0
割り当て問題最適化	1.159	3.480	1.563	8.5

表 2: 生成テキストの ROUGE スコア

	検索手法	P	R	F
ROUGE-1	上位 10 個を抽出	19.19	29.51	21.35
	割り当て問題最適化	19.12	27.85	20.80
ROUGE-2	検索手法	P	R	F
	上位 10 個を抽出	4.23	6.52	4.59
	割り当て問題最適化	3.96	5.88	4.28
ROUGE-L	検索手法	P	R	F
	上位 10 個を抽出	16.25	25.28	18.13
	割り当て問題最適化	16.18	23.83	17.63

内でキーワードとなる単語を抽出できているかどうかを確かめた。

2つ目の実験は、両手法によって最終的に生成されるスライドの文章を評価することで、システム全体としての性能を評価するための実験である。実際に両手法によって抽出された文章を用いて要約モデルを訓練し、生成されたテキストと人手で作られたスライドテキストとの類似度を ROUGE スコアで比較した。また、生成されたテキストの内容の重複具合を調べるため、事前訓練済みの Sentence BERT [12] を用いて、生成されたスライドのテキスト同士の内容重複度を式 (8) で計算した。

$$\text{内容重複度} = \frac{1}{n} \sum_i \max_{(e_j^{(i)}, e_k^{(i)}) \in S_i, j \neq k} \cos(e_j^{(i)}, e_k^{(i)}) \quad (8)$$

内容重複度は、生成したスライド集合から算出する。 i 番目の論文から生成したスライド集合 S_i の全てのテキストを Sentence BERT でベクトル化し、この中の 2 組のベクトル $(e_j^{(i)}, e_k^{(i)})$ の中で、最も \cos 類似度が高い組合せを、 n 報の論文で平均する。内容重複度は、0 に近いほどスライド間での内容の重複が少なく、1 に近いほど重複が多いことに対応する。

5 結果

論文中の文章検索モジュールの性能を評価した結果を表 1 に示す。各スライドタイトルごとに \cos 類似度が上位 10 件の文章を抽出する手法 (D2S) では、Recall が優位であった一方で、Precision では提案手法の割り当て問題最適化による抽出が優位であった。

上位 10 件を抽出する場合に対して、割り当て問題最適化による Recall が下がったのは、割り当て問題の適用によって、論文のスニペット数の数が限ら

表 3: 提案手法により重複が抑えられた例 (Ziqiang ら [13] の論文を使用)

[ページ 2] “Seq2seq Summarization” というタイトルのスライドに対する結果	
上位 10 個を抽出	We assume similar sentences should hold similar summary patterns. Given a sentence x, we find out its analogies in the corpus and pick their summaries as candidate templates. Since the size of our dataset is quite large (over 3M), we leverage the widely-used Information Retrieval (IR) system Lucene1 to index and search efficiently.
割り当て問題最適化	We assume similar sentences should hold similar summary patterns. Given a sentence x, we find out its analogies in the corpus and pick their summaries as the candidate templates. Since the size of our dataset is quite large (over 3M), we leverage the widely-used Information Retrieve (IR) system Lucene 1 to index and search efficiently.
実際のスライドのテキスト	Require less human efforts. Achieve the state-of-the-art performance.
[ページ 4] “Template based Summarization” というタイトルのスライドに対する結果	
上位 10 個を抽出	We assume that similar sentences should hold similar summary patterns. Given a sentence x, we find out its analogies in the corpus and pick their summaries as the candidate templates. We leverage the widely-used Information Retrieval (IR) system Lucene1 to index and search efficiently. We extend the seq2seq model to jointly learn template saliency measurement (Rerank) and final summary generation (Rewrite).
割り当て問題最適化	Templates are incomplete sentences which can be filled with the input text using the manually defined rules. For instance, a concise template to conclude the stock market quotation is: [REGION] shares [open/close] [NUMBER] percent [lower/higher], e.g., hong kong shares close #.# percent lower.
実際のスライドのテキスト	A traditional approach to abstractive summarization. Fill an incomplete with the input text using the manually defined rules. Be able to produce fluent and informative summaries. Template [REGION] shares [open/close] [NUMBER] percent [lower/higher]. Source hong kong shares closed down #.# percent on friday due to an absence of buyers and fresh incentives. Summary hong kong shares close #.# percent lower.

れている場合に、スライド数あたりに割り当たるスニペット数が減少したためであると考えられる。実際に平均抽出数は D2S が 10 件であるのに対し、割り当て問題最適化の場合は 8.5 件であった。

2 つ目の実験結果を表 2 に示す。ROUGE スコアによる評価では、既存手法の方が人手のスライドテキストの内容を再現できていたことがわかる。しかし、個々の生成文の結果を見ると、提案手法によりスライド間での生成テキストの重複が削減できたケースが確認できた。表 3 にその一例を示す。この表では、ある論文のスライド 2 ページ (ページ 2 とページ 4) に対して、既存手法と提案手法が生成したテキスト、および実際に人手によって作成されたテキストを比較している。ここで、システムが提案するテキストが人手のスライドテキストの内容と一致する部分には、内容ごとに別色で色付けをしている。また、太字の部分はスライド間での内容の重複を表している。

実際に、ページ 2 とページ 4 で上位 10 件の文章を抽出した場合の生成結果を比べると、太字部分に明らかな重複が見られる。結果として、既存手法ではページ 2、ページ 4 の両方で実際のスライドテキストの内容を再生できていないことがわかる。一方で提案手法では、ページ 2 の結果には実際のスライドテキストと一致する内容は見られないものの、ペー

表 4: 生成テキストの内容重複度

検索手法	内容重複度 (↓)
上位 10 個を抽出	0.711
割り当て問題最適化	0.699

ジ 4 では高い精度で人手のスライドテキストの内容を再生できていた。

また、両手法の内容重複度の平均値を表 4 に示す。提案手法は単語・フレーズ単位では人手のスライドテキストを既存手法に比べて再現できていないが、内容の重複を抑えることに成功していることが分かる。

6 おわりに

本研究では、初心者経験者を問わず労力のかかるスライドテキストの生成を、論文から半自動的に行うシステムを目指し、生成されたスライドの内容重複を抑える手法を提案した。実験では、cos 類似度を用いたスコアリングをもとに割り当て問題の最適化を行うことで、重複の少ないスライドテキストの生成することに成功した。一方で、ROUGE スコアによる生成テキストの評価では既存手法にやや劣っており課題が残る。今後の発展としては、抽出する論文の文章の長さを情報量や文脈を考慮して調整することや、文章の重複ではなく内容の重複を排除する検索手法を提案することが考えられる。

参考文献

- [1] Lutz Bornmann and Rüdiger Mutz. Growth rates of modern science: A bibliometric analysis based on the number of publications and cited references. **Journal of the Association for Information Science and Technology**, Vol. 66, No. 11, pp. 2215–2222, 2015.
- [2] Edward Sun, Yufang Hou, Dakuo Wang, Yunfeng Zhang, and Nancy X.R. Wang. D2S: Document-to-Slide Generation Via Query-Based Text Summarization. In **Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**, 2021.
- [3] Tomohide Shibata and Sadao Kurohashi. Automatic Slide Generation Based on Discourse Structure Analysis. In **International Conference on Natural Language Processing**, Vol. 3651, pp. 754–766, 2005.
- [4] Mostafa Shaikh, Mitsuru Ishizuka, and Md Tawhidul Islam. ‘Auto-Presentation’: a multi-agent system for building automatic multi-modal presentation of a topic from World Wide Web information. In **IEEE/WIC/ACM International Conference on Intelligent Agent Technology**, pp. 246–249, 2005.
- [5] Yue Hu and Xiaojun Wan. PPSGen: Learning-Based Presentation Slides Generation for Academic Papers. **Knowledge and Data Engineering, IEEE Transactions on**, Vol. 27, pp. 1085–1097, 2015.
- [6] A. Ashray Bhandare, Chetan J. Awati, and Sonam Kharade. Automatic era: Presentation slides from Academic paper. **International Conference on Automatic Control and Dynamic Optimization Techniques**, pp. 809–814, 2016.
- [7] Sida Wang, Xiaojun Wan, and Shikang Du. Phrase-based presentation slides generation for academic papers. In **Thirty-First AAAI Conference on Artificial Intelligence**, pp. 196–202, 2017.
- [8] Athar Sefid, Jian Wu, Prasenjit Mitra, and C. Lee Giles. Automatic Slide Generation for Scientific Papers. In **SciKnow@K-CAP**, 2019.
- [9] Tsu-Jui Fu, William Yang Wang, Daniel McDuff, and Yale Song. DOC2PPT: Automatic Presentation Slides Generation from Scientific Documents. **arXiv preprint arXiv:2101.11796**, 2021.
- [10] Iz Beltagy, Kyle Lo, and Arman Cohan. SciBERT: Pre-trained Language Model for Scientific Text. In **Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing**, pp. 3615–3620, 2019.
- [11] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics**, pp. 7871–7880, 2020.
- [12] Nils Reimers and Iryna Gurevych. Making Monolingual Sentence Embeddings Multilingual using Knowledge Distillation. In **Empirical Methods in Natural Language Processing**. Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, 2020.
- [13] Ziqiang Cao, Wenjie Li, Sujian Li, and Furu Wei. Retrieve, Rerank and Rewrite: Soft Template Based Neural Summarization. In **Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics**, 2018.

A IDF 重み付け ROUGE スコア

4 で言及した、逆ドキュメント頻度 (IDF) によって重み付けされた ROUGE スコアの計算式を追記しておく。idf(w, D) をある単語 w のドキュメント D に対する IDF, $|D|$ を D に含まれる文章の総数, n_w を単語 w が D のうち何個の文章に含まれるかの回数とすると、IDF により重み付けされた ROUGE スコアは次式 (9)-(12) のように計算される。

$$\text{idf}(w, D) = \log \left(\frac{|D|}{1 + n_w} \right) + 1 \quad (9)$$

$$\text{IDF}_{\text{Recall}} = \frac{\sum_{w \in T_k^{\text{slide}} \cap T_l^{\text{picked}}} \text{idf}(w, D)}{\sum_{w \in T_k^{\text{slide}}} \text{idf}(w, D)} \quad (10)$$

$$\text{IDF}_{\text{Precision}} = \frac{\sum_{w \in T_k^{\text{slide}} \cap T_l^{\text{picked}}} \text{idf}(w, D)}{\sum_{w \in T_l^{\text{picked}}} \text{idf}(w, D)} \quad (11)$$

$$\text{IDF}_{\text{F-measure}} = \frac{2 \cdot \text{IDF}_{\text{Recall}} \cdot \text{IDF}_{\text{Precision}}}{\text{IDF}_{\text{Recall}} + \text{IDF}_{\text{Precision}}} \quad (12)$$

ここで今回の設定では、ドキュメント D は論文である。また、 T_k^{slide} は人手によって作成されたスライドのテキストに含まれる単語集合を、 T_l^{picked} は検索モジュールによって抽出された論文の文章に含まれる単語集合を表している。