

ネットワークコンフィグの潜在的特徴抽出手法の提案と評価

Proposal and Evaluation of Extracting Latent Feature from Network Configuration

花光 大輔[†] 水谷 后宏^{‡ §} 小林 諭[#] 福田 健介[#] 明石 修[#]
 Daisuke Hanamitsu Kimihiro Mizutani Satoru Kobayashi Kensuke Fukuda Osamu Akashi

1. はじめに

近年、スマートフォンやPCなどのデバイスの利用の急増によってネットワークの大規模化が進んでいる [1]. それに伴い、大規模なネットワークを構築する際にオペレータが膨大な設定情報（以下、ネットワークコンフィグと記述する.）を各ネットワーク機器に対して入力する必要がある. それによって、人的エラーなどが原因で各ネットワーク機器の予期せぬ動作も考えられる. 具体的な例として、平成 29 年 8 月に起きた Google がネットワークの経路情報を誤って設定したことにより大規模なインターネットの接続障害が起きた事例が挙げられる [2].

本設定をミスなく円滑に実施するため、本研究でネットワークコンフィグの自動生成技術を実現することを目指す手法の一つとして、ネットワークコンフィグの雛形を作成しておき、その雛形にパラメータ部分のみを入力させる手法が考えられる. しかしながら、この手法にはネットワークコンフィグの種類に応じて雛形の数が増大になってしまい、事前に雛形を用意するのが難しい問題点がある. さらに、ネットワークコンフィグの仕様がアップデートされるたびに雛形を作り直す必要がある問題点もある.

この問題点を解決するために、本研究では図 1 のような、一定のネットワークコンフィグを教師なし学習器に与えると、そのネットワークコンフィグに関連するテンプレートブロックを返すレコメンデーション機能の作成を目指している. このようなレコメンデーション機能を作成することにより、ネットワークコンフィグの雛形を作成する手間をかけることなくネットワークコンフィグの自動生成を実現することができる. さらに、オペレータが特定のプロトコルのネットワークコンフィグのコマンドの記述の方法を完全に知らなかったとしても、断片的なネットワークコンフィグさえ覚えていれば、そのプロトコルのネットワークコンフィグのコマンドを完全に

理解できることが期待される.

本論文では図 1 のようなレコメンデーション機能を実現する第一歩として、非負値タッカー分解 (Non-Negative Tucker Decomposition, NTD) を利用したネットワークコンフィグから自動的にテンプレートブロックを抽出する手法を提案する [3][4][5]. 実験では、提案手法がどの程度、ネットワークコンフィグのテンプレートブロック抽出に貢献しているかを評価する.

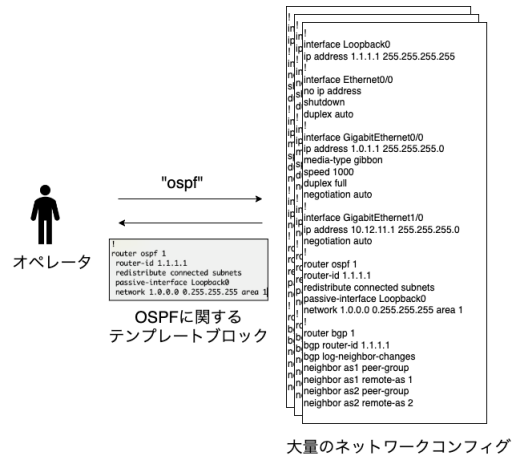


図 1: レコメンデーション機能を利用したネットワーク設定の自動化の例

2. 関連研究

ネットワークコンフィグの自動生成技術には主に 2 種類の手法がある. 一つ目はネットワークコンフィグの雛形を作成して、その雛形にパラメータを代入する手法である. しかしながら、このアプローチではネットワークコンフィグの種類が増えると同時に事前に作成するネットワークコンフィグの雛形の種類も増えることやネットワークプロトコルが更新されるたびに雛形を書き換えなければならない問題点がある. 二つ目はネットワークログのグルーピングを基に考案したネットワークコンフィグから抽出した潜在的特徴量を利用してネットワークコンフィグをグルーピングする手法である [6][7]. この手法だと一つ目のアプローチで問題になっていたネットワークコンフィグの雛形の膨大な量の作成や書き換えの手間がなくなる. しかしながら、この論文では、ネットワー

[†] 近畿大学大学院総合理工学研究科, Graduate School of Science and Engineering Research, Kindai University

[‡] 近畿大学情報学部, Faculty of Informatics (KDIX), Kindai University

[§] 近畿大学情報学研究所, Cyber Informatics Research Institute, Kindai University

[#] 国立情報学研究所, National Institute of Informatics

クコンフィグから潜在的特徴を抽出して、個々のネットワークコンフィグのコマンドに対して、手動でグルーピングを行う。そのため、ネットワークコンフィグのコマンドのグルーピングがある程度できていたとしても、そのグループ内に連続したネットワークコンフィグ (i.e., BGP など、手順を踏んで行うネットワークコンフィグが上手く動作しない) が含まれていないといった問題点がある。この問題を解決するために、本研究ではネットワークコンフィグのテンプレートブロック抽出手法を提案する。

3. 提案手法

3.1 提案手法の概要

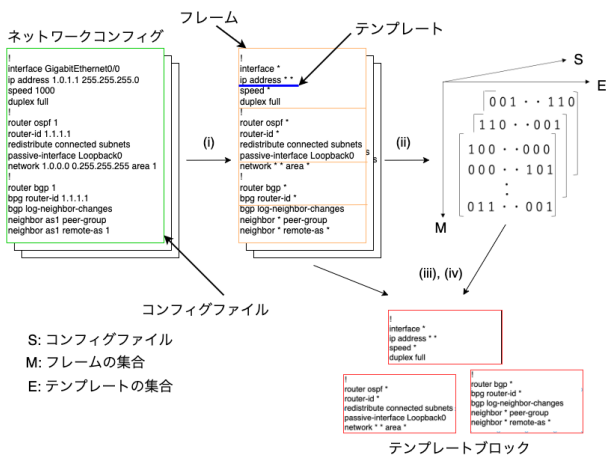


図 2: 提案手法の概要図

提案手法は図 2 のように、(i) コンフィグファイルに対する前処理、(ii) ネットワークコンフィグに対する 3次元のテンソルへの変換処理、(iii)NTD を利用した特徴行列の抽出、(iv) 特徴行列を基にテンプレートブロックを作成、以上の 4つの手順から構成される。はじめに、コンフィグファイルに対する前処理ではコメントアウトの削除、ネットワークコンフィグのコマンドのテンプレート化を行う。二つ目のネットワークコンフィグに対する 3次元のテンソルへの変換処理では、図 2 のように、各コンフィグファイルを複数のフレームに分割して、フレーム内にどのテンプレートが含まれているかを定量的に表現する。三つ目の NTD を利用した特徴行列の抽出処理では定量的に表現されたネットワークコンフィグを入力として非負値タッカー分解を用いることでテンプレートブロックの作成時に必要な特徴行列を抽出する。最後に特徴行列を基にテンプレートブロックを作成する処理では特徴行列を利用することで、ネットワークコンフィグのコマンドをグルーピングして関連性の高いネットワークコンフィグのコマンド同士が含まれるテンプレートブ

ロックを作成する。3.2 節から 3.5 節では、本節で示した手順を詳細に記述していく。

3.2 ネットワークコンフィグに対する前処理

テキスト化したネットワークコンフィグに対する前処理を行うことで、洗練されたテンプレートブロック抽出が可能となる。具体的な前処理として、ネットワークコンフィグのコメントアウトの削除やネットワークコンフィグのコマンドのテンプレート化などが挙げられる。ネットワークコンフィグのコマンドのテンプレート化とはネットワークコマンドコンフィグのコマンドのパラメータ部 (e.g., IP アドレス, ルータ ID) をアスタリスクで置き換える操作である。すなわち、ネットワークコンフィグのコマンドがパラメータ部のみ異なる場合、同一のコマンド (i.e., テンプレート) としてみなす。テンプレート化を行うことにより、ネットワークコンフィグの情報量を削減して、より良いネットワークコンフィグの特徴抽出が可能となる

3.3 ネットワークコンフィグに対する 3次元のテンソルへの変換処理

図 2 のように、前処理した各コンフィグファイルを任意の大きさのフレームに分割する。次に、各フレーム内にどのテンプレートが含まれているかをフラグを立てる (i.e., 0, 1 で表す) ことで各コンフィグファイルを行列として表現する。そうして定量的に表現された各コンフィグファイルを組み合わせることで、ネットワークコンフィグを 3次元のテンソルとして表現する。これにより、ネットワークコンフィグの形を崩すことなく、ネットワークコンフィグの特徴を定量化することができる。また、形式的に記述すると、以下、 S 個のコンフィグファイル C_1, C_2, \dots, C_S とテンプレート $e_i \in E (1 \leq i \leq |E| : |E|$ は C_1, C_2, \dots, C_S 内に含まれるテンプレートの全種類数) から、 $s (1 \leq s \leq S)$ 個目の L 行のテンプレートから構成されるコンフィグファイル C_s を $M (\geq 2)$ 個のフレームに分割して、 $C_{s,1}, C_{s,2}, \dots, C_{s,M}$ として、各フレーム $C_{s,m} (1 \leq m \leq M)$ を用いて、共起ベクトル $C_{s,m} = \{w_{e_1}, w_{e_2}, \dots, w_{e_{|E|}}\}$ を表現する。また、 $C_{s,m}$ の各要素 w_{e_i} に関して、テンプレート e_i が $C_{s,m}$ 内に含まれているならば $w_{e_i} = 1$, そうでなければ $w_{e_i} = 0$ と定義する。これらの共起ベクトルを組み合わせることで $M \times |E|$ の共起行列 $C_s = \{C_{s,1}, C_{s,2}, \dots, C_{s,M}\}$ を構築する。最後に、各コンフィグファイルを用いて表現した共起行列を組み合わせることで、コンフィグファイル、フレームの集合、テンプレートの集合方向に次元を持つ、 $S \times M \times |E|$ の 3次元のテンソル $X = \{C_1, C_2, \dots, C_S\}$ を構築できる。

3.4 NTD を利用した特徴行列の抽出

3.3 節で作成した 3 次元のテンソルに対して, NTD を用いて各次元方向に分解をすることによって各次元方向の特徴行列を獲得する. 本研究では, テンプレート方向の特徴行列を抽出することによって, 各テンプレートに対するグルーピングが可能となる. 初めに, NTD の一般的な定義について触れておく. NTD とは, 非負値のみを要素とする 3 次元のテンソル $\mathbf{X} \in \mathbb{R}^{A \times B \times C}$ が与えられた時, 以下のように潜在的特徴変数を用いて 3 次元のテンソルを分解する.

$$\mathbf{X} \approx \mathbf{Q} \times \mathbf{S} \times \mathbf{T} \times \mathbf{U} \quad (1)$$

$\mathbf{Q} \in \mathbb{R}^{L \times M \times N}$ ($L < A, M < B, N < C$) は潜在的特徴方向を次元に持つ 3 次元のコアテンソルを指す. また, 潜在的特徴行列である $\mathbf{S} \in \mathbb{R}^{A \times L}, \mathbf{T} \in \mathbb{R}^{B \times M}, \mathbf{U} \in \mathbb{R}^{C \times N}$ は各特徴方向に着目した行列である. 本研究では, 実際に潜在的特徴変数 F, G, J を用いて, $S \times M \times |E|$ の 3 次元のテンソル \mathbf{X} を各特徴方向の行列 ($S \times F$ の行列 \mathbf{Q} , $M \times G$ の行列 \mathbf{W} , $|E| \times J$ の行列 \mathbf{H}) に分解することで, 各特徴方向 (コンフィグファイル, フレーム, テンプレート) の潜在的特徴行列を抽出する. ここで, ネットワークコンフィグのテンプレートブロック抽出に潜在的特徴抽出手法の一つである NTD を適応した理由について言及していく. 主な理由は, NTD では潜在的な特徴パターンの抽出が可能のため, ネットワークコンフィグの高次なセマンティック抽出が実現できるためである. 他のアプローチとして, ネットワークコンフィグの連続的構造を比較することでパターンを抽出する手法も考えられる. しかしながら, この手法では表面的なパターンしか抽出することが出来ないため, Access Control List (ACL) などの高次なセマンティック抽出は難しい. 以上から, 本研究では NTD を用いたテンプレートブロック抽出を実施する.

3.5 特徴行列を基にテンプレートブロックを作成

3.4 節で求めた特徴行列 (i.e., テンプレート方向の特徴行列, $|E| \times J$ の行列 \mathbf{H}) を利用してテンプレートブロックを作成する. 具体的には, \mathbf{H} の行列値 $H_{i,j}$ ($1 \leq i \leq |E|, 1 \leq j \leq J$) を定義して以下の式を満たすテンプレート e_i はテンプレートブロック p の要素になる.

$$H_{i,p} \geq \delta \times \max_{j \leq J} H_{i,j} \quad (0 < \delta \leq 1, 1 \leq p \leq J) \quad (2)$$

4. 実験・評価

Cisco Packet Tracer と呼ばれるネットワークシミュレータにバンドルされているルータ 21 台のネットワークコンフィグをデータセットとして, NTD を利用したテンプレートブロックの抽出手法の性能を評価する [8]. また,

この手法に必要なパラメータ (i.e., $J = 4, M = 5, 10, \delta = 0.5$) は経験則に基づき設定した. さらに, NTD を利用した抽出手法の有用性を確認するため, 潜在的特徴抽出手法の一つである非負値因子行列分解 (Non-Negative Matrix Factorization, NMF) を利用したテンプレートブロック抽出手法の性能と比較する [9][10]. なお, NMF は NTD と異なり非負値のみを要素に持つ行列を潜在的特徴変数を基に分解するため, NTD で利用した非負値を要素のみを要素に持つ $S \times M \times |E|$ の 3 次元のテンソルを $(SM) \times |E|$ の行列に置き換える. 本実験における, 性能の評価基準は次のように定める. はじめに, 実験データを訓練用とテスト用に分割する. 次に訓練用のデータから NTD と NMF を利用したテンプレートブロック抽出手法を適用する. そして, 各手法によって抽出されたテンプレートブロックがテスト用のデータと多く一致している方がより優れた抽出手法とみなす.

表 1: 実験結果

フレーム (M)	NTD	NMF
5	17	12
10	18	10

表 1 では, フレームの数を変化させ (i.e., 図 2 で示すような 1 つのフレーム内に含まれるテンプレートの行数を変化させる), NTD もしくは NMF を利用したテンプレートブロック抽出を行い, 何個のテンプレートブロックがテストデータに含まれるテンプレート群と一致しているかを示している. 表 1 から, NTD の方が NMF よりもネットワークコンフィグのテンプレートブロックの抽出性能が良いことが分かる.

5. 結論

本論文では NTD を利用したネットワークコンフィグのテンプレートブロック抽出手法を提案した. また, 同じ特徴抽出手法である NMF とのテンプレートブロックの抽出性能を比較して, NTD を利用した抽出性能の方が優れていることが実験・評価から確認された. 今後の課題として, さらに大規模なネットワークコンフィグに本論文で提案した手法を適応することや, より精度の高いテンプレートブロック抽出などが挙げられる.

謝辞

本研究は JSPS 科研費 20H04185 の助成を受けたものです.

参考文献

- [1] 総務省 情報通信白書 2020 <https://www.soumu.go.jp/johotsusintokei/whitepaper/ja/r02/>

pdf/index.html

- [2] 総務省 平成 29 年 8 月に発生した大規模なインターネット接続障害に関する検証報告 https://www.soumu.go.jp/main_content/000523153.pdf
- [3] Tamara G Kolda et al, "Tensor Decomposition and Applications". *SIAM Review*, vol. 51, pp. 455-500, 2009. DOI:10.1137/07070111X
- [4] Y. Kim et al, "Nonnegative Tucker Decomposition," in Proc. 2007 IEEE Conference on Computer Vision and Pattern Recognition, 2007, pp. 1-8, DOI:10.1109/CVPR.2007.383405
- [5] Non-Negative Tucker Decomposition. http://tensorly.org/stable/auto_examples/decomposition/plot_nn_tucker.html
- [6] Tatsuaki KIMURA et al, "Mining Network Logs for Diagnosing Large-scale Networks Problems". The Institute of Electronics, Information and Communication Engineers Technical Reports, vol. 111, No. 468, pp. 261-264, 2012.
- [7] Daisuke HANAMITSU et al, "A Scheme of Extracting Network Configuration Template Blocks Based on Tucker Decomposition" in Proc. International Conference on Emerging Technologies for Communications, 2021. DOI:10.34385/proc.68.P1-14
- [8] Cisco Packet Tracer <https://www.netacad.com/ja/courses/packet-tracer>
- [9] Daniel D. Lee et al, "Learning the parts of objects by non-negative matrix factorization". *Nature*, Vol. 401, No. 6755, pp. 788-91, 1999. DOI: 10.1038/44565
- [10] Non-Negative Matrix Factorization. <https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.NMF.html>