

## メタファー写像と後編集を利用する物語文章生成フレームワーク

## A Framework of Generating a Narrative Using Metaphor Mapping and Post-editing

松吉 俊\*<sup>1</sup>      内海 彰\*<sup>1</sup>  
Suguru Matsuyoshi      Akira Utsumi

\*<sup>1</sup>電気通信大学大学院 情報理工学研究所

Graduate School of Informatics and Engineering, The University of Electro-Communications

We propose a framework of generating a narrative. Using simulators and metaphor mapping knowledge, our system generates various narratives which can be revised better by post-editing.

## 1. はじめに

人工知能や認知科学の分野において、物語文章自動生成システムの研究は古くから取り組まれてきた [秋元 13]。物語生成の主な流れは、次のとおりである。

1. 物語内容 (story) の決定: 何を語るか
2. 物語言説 (discourse) の決定: どのような構成・技法を用いるか
3. 文字列化: 内部表現をどのように表層文字列に変換するか

本研究では、この流れを利用しない新しいパラダイムを提案する。具体的には、人間と同等以上の推論・探索能力が期待できる問題やゲームにおいて、システムにその世界をシミュレート [松尾 16] させ、その履歴である状態遷移列をメタファー写像に基づいて解釈させるという方法を提案する。

## 2. 関連研究

物語文章自動生成の関連研究について、主に利用されるフレームワークという観点からまとめる。

**内部構造** 物語文法などにに基づき、独自の記法で記述されたスクリプト (事象集合のグラフ) を操作する手法 [秋元 13, Akimoto 17, Montfort 13, 佐藤 16]

**テンプレート** テンプレート列を利用して文章を生成する手法 [長尾 96]。テンプレートには、スロットと呼ばれる穴があり、事前に定めた選択肢集合の中から 1 つを選択して穴を埋めることにより、テキストを生成する

**連接尤度** これまで生成した語句の列を入力として受け取り、次の語句を生成する手法 [浅川 15]。文の境界を越えつつ、この過程を繰り返すことによりテキストを生成する

**単純シミュレート** ゲームやシミュレーターを実行して得られた出力や実行ログの列を文字列に翻訳する手法 [福田 15, 鳥海 16, 松山 17]

「内部構造」手法は、前章で述べた 3 つのステップを実現するものであり、理論的で汎用的な手法である。必要に応じて、「テンプレート」手法が併用されることもある。研究者独自の内部表現が多く、複雑な体系をしていることが多い。

「テンプレート」手法は実装が容易であるため、駅における列車の行き先案内など、一般の言語生成タスクにおいても古くから利用されている。スロットの位置や選択肢集合をどのように定めるのが適切かを問題ごとに決定する必要があり、分野専門性が求められる。

「連接尤度」手法は、N-gram モデルやリカレントニューラルネットワークモデルを利用するものである。現在の技術では、意味の通る文章を生成できるのはたかだか数文程度である。

「単純シミュレート」手法は、これまで人力で行われていたテレビゲームの小説化作業を、計算機に実行させるものである。目的の物語を生成するためには、そのシミュレーターを用意しなければならないという短所がある。

これら上記の手法は、他の自然言語処理タスクと比べると、システムの評価 (人手評価、自動評価ともに) が難しいものとなっている。一般に、あるシステムが複数の構成要素 (モジュール) からなる場合、全体の評価とモジュールごとの評価が可能である。コストや方法論などの面においてシステム全体の評価が難しい場合でも、モジュールごとの自動評価が相対的に容易であれば、各モジュールの性能を向上させることにより、ひいては、システム全体の性能を向上させることができる。しかしながら、先行研究においては、モジュールの入出力に、複雑な独自内部表現や数値行列が採用されていることが多いため、その出力を人間に評価させることも、コミュニティ共有の評価データを用意することも難しい。

その他の比較については、論文 [松吉 18] を参照してほしい。

## 3. 提案手法

提案手法の流れを図 1 に示す。

### 3.1 概要

本手法では、以下の手順により、自然言語で書かれた物語文章を自動生成する。

#### 1. メタファー写像データベースの構築

テキスト集合からメタファー写像の集合を自動抽出し、表 1 に示される形式でデータベースに格納する。この表の最左列には、ゲーム名やシミュレート世界が示されており、この世界を構成する事象と、物語を構成する事象と

連絡先: 松吉 俊、電気通信大学大学院 情報理工学研究所、東京都調布市調布ヶ丘 1-5-1 電気通信大学 西 5 号館 505 号室、042-443-5292、matuyosi@uec.ac.jp

表 1: メタファー写像の例 (左辺はシミュレーター内の行動や状態。右辺は物語内の表現)

	物語	メタファー写像	
チェス	部屋の片付け	ボーンを取る ⇒ 本を整理する	ボーンが取られる ⇒ 本を読みふける
	悩み	ボーンを取る ⇒ ほんの少し前向きになる	ナイトを取る ⇒ 誰かに相談してみようと思う
	多対1の恋愛競争	チェックをかける ⇒ 対応人物が告白する	駒が取られる ⇒ 対応人物が誘惑され離脱
将棋	探し物	王手が防がれる ⇒ 見つけたと勘違い	駒が取られる ⇒ 探す気力が減る
迷路探索	プロジェクト進行	ゴールに近づく ⇒ 順調に進む	袋小路に向かう ⇒ 泥沼に陥る
果物育成	発掘調査	水を撒く ⇒ 地道に掘る	実が小さい ⇒ 成果が少ない
野球の打席	会話の駆け引き	直球を投げる ⇒ ストレートに伝える	ボール球に手を出す ⇒ うっかり話す

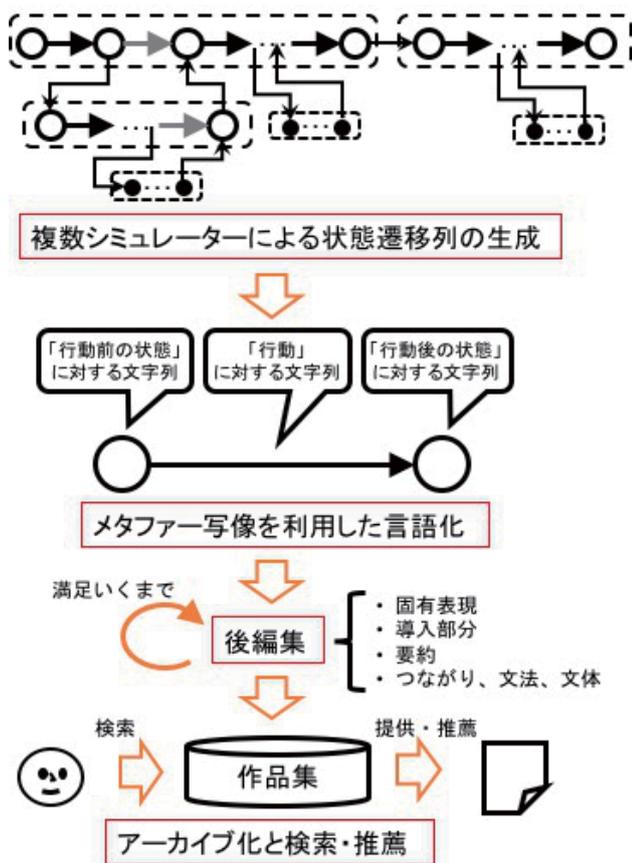


図 1: 提案手法の流れ

の間の関係を写像として蓄積する。メタファー写像に関する理論的説明は、3.2 節で述べる。

## 2. 複数シミュレーターによる状態遷移列の生成

ゲーム AI やシミュレーターを実行し、実行に付随する「状態」や「行動」の列 (状態遷移列) を取得する。スペースの都合により、図 1 では、この段階ですでに世界同士が繋がるように描いたが、実際に繋がるのは、言語化の後である。

## 3. メタファー写像を利用した言語化

表 1 のようなメタファー写像データベースを参照し、状態遷移列の各要素を文字列に置き換える。対応する写像が存在しない場合、置き換えは実施せず、その要素を削除する。

## 4. 後編集

人間が草稿を書いた後に実施するのと同様に、文章の質を高めるためにテキストを校正する。具体的なサブモジュールについては、3.3 節で説明する。

## 5. アーカイブ化

自動生成された多くの物語文章を作品集としてまとめる。ユーザーは、検索により、望む作品を見つける。

割り当てられた文字列が高い類似度を持つ「状態」を媒介として、複数の世界を繋げる。繋ぎ方には次の 2 とおりがある。

- 複数シミュレーターを並列的に走らせてから、のちに繋げる。
- 1つのシミュレーターを走らせ、状態遷移列を生成する。新しい「状態」が生成されれば、すぐにそれを言語化し、その「状態」を始点とできるシミュレーターの有無を調べる。存在すれば、そのシミュレーターを走らせるとともに、世界を繋げる。

## 3.2 メタファー写像に関する不変性原理

認知言語学概念メタファー理論において、メタファーは、2つの意味領域間の連結であると捉えられる [鍋島 11, アリス 10]。表 1 はメタファーの例を表すものであるが、例えば表の 6 行目において、「プロジェクト進行」という意味領域 (根源領域) が、「迷路探索」という意味領域 (目標領域) に結び付けられている。この連結において、例えば、根源領域内の「順調に進む」という表現は目標領域内の「ゴールに近づく」という表現に写像可能であるとみなされる。

メタファー写像に関して、Lakoff [Lakoff 93] は、「写像において、根源領域の表現間の (位相) 構造は目標領域内でも保たれる \*1。」という不変性原理を提唱した。この原理に則ると、任意の 2 つの意味領域間において数個のメタファー写像の存在が確認されれば、根源領域内の他の表現も位相構造を保ったまま目標領域内に対応づけることができる見込みが非常に高い。言い換えると、次の 2 つのことが成り立つ見込みが非常に高い。

- テキストのジャンルを問わず、数個のメタファー写像の存在が確認されれば、表 1 のようなデータベースが高い品質で構築できる
- ゲームやシミュレーター内に実装されている種々の制約や論理が、物語文章生成においても制約・論理として働くので、物語の展開において、「狭い意味でのフレーム問題」 [人工知能学会 05] を回避できる

我々は、これらの成立の検証も兼ねて本研究を進めている。

\*1 正確には、「目標領域が課す制約が許す範囲において」が付く。

### 3.3 モジュール

この節では、提案手法に関連するモジュールについて考察する。

多くのモジュールがすでに自然言語処理において研究が進められているタスクであり、その部分のみの評価は比較的容易であると思われる。また、後編集のサブモジュールは、すべて、文章を入力として別の文章を出力するものである。文字列から文字列への系列変換問題とみなすことができるため、必要に応じて訓練データを用意すれば、既存の多くの手法が適用可能である。

#### 3.3.1 メタファー写像データベースの構築

##### メタファー認識

テキストからメタファーを自動認識する技術が必要である。「のように」や「みたいな」のような手がかり語が利用できる直喩のみでなく、隠喩も認識することが望まれる。

##### 知識獲得

数個のメタファー写像の存在が確認された場合、明示的な写像がテキスト中になかった事象も、物語の構成要素として獲得する。従来の知識獲得では、時間的に近い事象のみを収集することが望ましいと思われるが、時間的に離れた事象を収集することも大事であると思われる。例えば、ツイート間の間隔が数日ある場合、物語の階層構造の章レベルもしくは段落レベルの知識として収集することが考えられる。

高い精度での知識獲得が実現できれば、人手を全く介さずに、新語を含むテキストからデータベースを更新し、新語を含む物語を生成できるようになるとと思われる。

##### フィルタリング

出力される物語に不適切な表現が含まれないように、あらかじめデータベースからそのような表現を除去することが可能である。除去した表現を別のデータベースに蓄積しておくことにより、文字列完全一致でなくとも、別の不適切表現を既存の自動分類技術により自動認識することができるとと思われる。

##### クラウドソーシング

データベースの構築にクラウドソーシングも有効である。

##### データベースの共有

メタファーは文化に依存する。少資源言語におけるデータベースを構築する際に、文化の共通性という観点に基づいて取捨選択すれば、リソースが豊富な他言語のデータを一部共有可能であると思われる。

#### 3.3.2 複数シミュレーターによる状態遷移列の生成

状態遷移列の生成自体は、言語非依存である。それゆえ、このモジュールは日本語以外にも直接応用可能である。

##### ゲームのバリエーション

システムの設定を変更することにより、物語としての一貫性を保ったまま、容易に出力にバリエーションを持たせることができる。例えば、チェスの場合、駒の数や配置を変えて物語を生成することができる。迷路探索の場合、複雑な迷路やほぼ一本道の迷路などを選択できる。

##### 機械学習

失敗もストーリーである。それゆえ、通常のタスクにおいては失敗であるとみなされる学習モデルであったとしても、魅せる物語を生成するために利用できる可能性が高い。単に強いだけの AI はおそらく望ましくなく、接待将棋 AI や指導碁 AI を開発するような特別な方法論が必要となるかもしれない。

また、物語の構成単位(部、章、節、段落)のどこを担当するかによって、良い学習モデルが変わる可能性がある。

### マルチエージェント

登場人物が複数である場合、それぞれ別のシミュレーターを走らせることができる。

##### 伏線

伏兵が絡むようなシミュレーターの場合、伏線を持った物語を自動生成できる可能性がある。

##### (登場人物が実行する) 予測

1つのゲームに複数の AI が関与する場合、「今後こうなると思う」と登場人物に予測を語らせ、予測と実際の結果の差分が楽しめるような物語が生成できる可能性がある。

#### 3.3.3 メタファー写像を利用した言語化

##### 比喩表現

提案手法と「単純シミュレート」手法を併用することにより、生成される物語に比喩表現を混ぜることができる。例えば、将棋利用時に「穴熊です」のような隠喩表現を生成したり、果物育成シミュレーター利用時に「植物に水を撒くように地道に掘る」のような直喩表現を生成したりすることができる。

##### レトリック表現

メタファー写像を用いることにより、擬人法、誇張法、抑言法、婉曲法(忌語を避ける)の表現が自然に生成される可能性がある。

#### 3.3.4 後編集

後編集モジュールは、システムが生成したテキストのみならず、人間が書いたテキストにも適用可能である。それゆえ、これらのモジュールは、子供や日本語学習者を支援するツールに直接応用できる。

##### 固有表現認識

本提案手法の枠組みでは、メタファー写像データベースにおいて固有表現は制御されていない。それゆえ、物語全体として一貫性を保つためには、後編集において、適切な固有表現を選択する必要がある。後編集すべきテキストから固有表現を認識する技術が必要である。

##### 固有表現の置き換え

固有表現の制御は大きく2種類ある。1つは、絶対的な置き換えであり、例えば、主要登場人物が「太郎」である場合、テキスト中の主要人物の名前をすべて「太郎」に置換する。もう1つは、相対的な置き換えである。例えば、イギリスの物語を生成している最中に、「近くのパリに出かける」のような表現があった場合、<イギリス → フランス>の関係から類推して<ロンドン ← パリ>の関係を得て、「パリ」を「ロンドン」に置換する。このような類推ができる機構が必要である。

相対的な置き換えにおいて、現在の知識では置き換え候補が存在しないことが認識できれば、新たに造語を用意する必要があることをシステムに認識させることができる。状況に合ったもっともらしい造語を生成できる機構が必要である。

##### 導入部分

物語の導入部分には、登場人物の紹介があると良い。主要な物語文章を生成し終えた後で、その人物の行動を抽出し、それらから帰納的に性格や容姿を推論させる方法が有効であると思われる。

##### 要約・ハイライト抽出

本フレームワークでは、世界をシミュレートしているので、冗長で退屈な部分が多い。この問題を解消するために、要約技術やハイライト抽出の技術が必要である。

##### つながり訂正、文法訂正、文体訂正

言語学や日本語教育学における知見に基づいた訂正機構が必要である。特に、1文内の訂正のみに限らず、文章構成論に基づく校正[石黒 04]の研究が必要である。

### 3.3.5 アーカイブ化

#### 検索、推薦

おそらく既存の枠組みで十分であると思われる。

### 3.3.6 全体のフレームワーク

#### 質問応答

表1のようにチェスで「多対1の恋愛競争」を描写する場合、生成時に盤面情報を保有しているため、全登場人物の現状について質問応答（「ジョンは奥手で全く動いていない。」など）が可能である。このとき、要望があれば、特定の邪魔な駒を排除するなど、物語の現状を途中で変更し、変更後の世界で物語を引き続き生成し続けることも可能であると思われる。

#### 文章の新しい分類軸

これまで、文章は、ジャンルや難易度で分類されることが主であった。本フレームワークを使用することにより、計算論的に次のような分類軸を考慮することができるかもしれない。

- 生成の計算量 (時間計算量と領域計算量)
- 階層構造の複雑さ

## 4. 提案手法の試作

提案手法に関して、「複数シミュレーターによる状態遷移列の生成」と「メタファー写像を利用した言語化」を実装した。プログラミング言語は、Python 3を用いた。

### 4.1 クラス定義

実装したシステムは、物語生成のために、次のようなクラスを利用する。

**main クラス** 全体の制御

**Instrument クラス** 楽器に見立てたシミュレーター

**Score クラス** パート譜に見立てたメタファー写像群

**Performer クラス** 楽器とパート譜をもつ演奏者

**Orchestra クラス** すべての楽器と演奏者からなる

**Story クラス** 物語を制御

**Part クラス** 演奏者を雇い、文字列を生成

### 4.2 現状

現在の実装においては、シミュレーターとしてチェスのみが利用可能である。メタファー写像を記述することにより、物語が生成可能である。

## 5. おわりに

本論文では、メタファー写像に基づく変換と階層化可能な状態遷移列を組み合わせることで、多様な物語文章を自動生成することができるフレームワークを提案した。今後は、知識獲得に関するモジュールの実装と後編集に関する学習コーパスの構築を実施する予定である。

本フレームワークで生成しうるものは、人間が創作しうる物語集合のうち、ほんの一部である。本フレームワークでは生成できない差分を見ることにより、人間の言語能力に感嘆すると共に、フレームワークのさらなる改良を行いたい。

## 参考文献

- [秋元 13] 秋元 泰介, 小方 孝: 物語生成システムにおける物語言説機構に向けて, 認知科学, Vol. 20, No. 4, pp. 396-420 (2013)
- [Akimoto 17] Akimoto, T.: Computational Modeling of Narrative Structure: A Hierarchical Graph Model for Multidimensional Narrative Structure, *International Journal of Computational Linguistics Research*, Vol. 8, No. 3, pp. 92-108 (2017)
- [アリス 10] アリス ダイグナン著, 渡辺 秀樹, 大森 文子, 加野 まきみ, 小塚 良孝訳: コーパスを活用した認知言語学, 大修館書店 (2010)
- [浅川 15] 浅川 伸一: ニューラルネットワーク物語生成モデル, 人工知能学会第2種研究会ことば工学研究会資料, Vol. 49, pp. 15-21 (2015)
- [福田 15] 福田 清人, 森 直樹, 松本 啓之亮: 既存小説に依存しない小説の自動生成に関する一考察, 言語処理学会第21回年次大会発表論文集, pp. 992-995 (2015)
- [石黒 04] 石黒 圭: よくわかる文章表現の技術 II 文章構成編, 明治書院 (2004)
- [Lakoff 93] Lakoff, G.: The Contemporary Theory of Metaphor, in Ortony, A. ed., *Metaphor and Thought (2nd edition)*, Cambridge University Press (1993)
- [松尾 16] 松尾 豊: Deep Learning から身体性、シンボルグラウンディングへ, 人工知能学会全国大会発表論文集 1A5-OS-27c-2 (2016)
- [松山 17] 松山 諒平, 佐藤 理史, 松崎 拓也: 人狼ログからの小説の自動生成, 言語処理学会第23回年次大会発表論文集, pp. 32-35 (2017)
- [松吉 18] 松吉 俊, 内海 彰: メタファー写像に基づく物語文章の自動生成, 言語処理学会第24回年次大会発表論文集, pp. 1288-1291 (2018)
- [Montfort 13] Montfort, N., Pérez y Pérez, R., Harrell, D., and Campana, A.: Slant: A Blackboard System to Generate Plot, Figuration, and Narrative Discourse Aspects of Stories, in *Proceedings of the International Conference on Computational Creativity (ICCC) 2013* (2013)
- [鍋島 11] 鍋島 弘治朗: 日本語のメタファー, くろしお出版 (2011)
- [長尾 96] 長尾 真 (編): 岩波講座ソフトウェア科学 15 自然言語処理, 岩波書店 (1996)
- [佐藤 16] 佐藤 理史: コンピュータが小説を書く日, 日本経済新聞出版社 (2016)
- [鳥海 16] 鳥海 不二夫, 大澤 博隆: AI 達は物語を生み出すか, 人工知能学会全国大会発表論文集 2F4-5 (2016)
- [人工知能学会 05] 人工知能学会 (編): 人工知能学事典, 共立出版 (2005)