

2023 年度
卒業論文

法政大学における
キャッチコピー自動生成の試み

指導教員
五島洋行 教授

法政大学
理工学部経営システム工学科

20X4116 中山悠次
20X4140 安田陸

学科名	経営システム工	学籍番号	20X4116 20X4140
申請者氏名		中山悠次 安田陸	
指導者教員		五島洋行	

論文要旨

論文題名	法政大学における キャッチコピー自動生成の試み
------	----------------------------

目次

1	序論	3
1.1	研究の背景	3
1.2	本研究の目的	4
1.3	論文構成	4
2	関連知識	6
2.1	自然言語処理	6
2.2	形態素解析	7
2.3	形態素解析ライブラリ	10
2.4	韻	11
2.5	キャッチコピー	12
3	先行・類似研究	14
3.1	自然言語処理によるキャッチコピー自動生成の研究	14
3.2	広告業界によるクリエイティブ自動生成 AI ソリューションの研究	14
4	前提条件	16
4.1	母音辞書	16
4.2	拗音・促音・撥音・長音	17
4.3	参照辞書	18
5	実験概要	19
5.1	フローチャート	19
5.2	プログラム概要	20
6	結果	24
6.1	「法政」の実装結果	24
6.2	「木曜」の実装結果	25
7	考察	27
8	おわりに	28

1 序論

本章の 1.1 では、本研究の背景について説明する。また、1.2 では目的について、1.3 では論文の構成について説明する。

1.1 研究の背景

現代のマーケティングと広告業界では、キャッチコピーの重要性が年々高まっている。世界の広告市場をみると、2022 年度のインターネット広告が 3 兆 912 億円、テレビメディア、新聞、雑誌、ラジオを含むマスコミ 4 媒体広告が 2 兆 3,985 億円となっている。またデジタル広告は前年度に比べて 13.7%増加しており、今後も大幅に成長すると見込まれている。[1]

消費者の関心を引き、製品やサービスの特長を効果的に伝えるキャッチコピーは、ブランドイメージを形成し、市場での競争力を高めるための不可欠な要素である。これらのコピーは、広告の成否に直結し、消費者に長く印象に残るブランドメッセージを構築する役割を担っている。

韻を踏む言葉の使用は、キャッチコピーの魅力高め、メッセージの記憶に残りやすくする効果的な手法として知られている。韻を踏む言葉は、リズムカルで耳に心地よい響きを持ち、言葉が持つ意味を超える印象を消費者に与える。音楽的な要素としての韻は、言葉の力を強化し、より深い感情的な共鳴を引き出す。また近年では、日本で HIPHOP が流行し韻の存在を認知する人々が増加した。このことから、他ジャンルのアーティストも歌詞の中に韻を取り入れた曲が数多く生み出されている。

しかし、効果的な韻を踏むキャッチコピーを創り出すことは、創造性と技術的な知識を要求され、容易な作業ではない。特に多言語環境では、異なる文化や言語における響きやニュアンスを正確に理解し、それを活用する必要がある。この複雑さは、キャッチコピー制作に多大な時間と労力を要する要因となっている。

さらに、デジタルメディアの普及により、広告の手法は急激に進化し、消費者の期待も変化している。オンラインプラットフォーム上での広告は競争が激しく、短時間で消費者の注意を引き、強い印象を残す必要がある。このような状況では、迅速かつ効率的に魅力的なキャッチコピーを生成する能力がますます重要になっている。

このため、キャッチコピーの自動生成技術への関心が高まっている。自然言語処理 (NLP) や機械学習といった先進技術を活用することにより、創造的かつ効果的なキャッチコピーを迅速に生成することが可能になると考えられる。これらの技術は、大量のデータを分析し、多様なスタイルや文化的背景に適応したキャッチコピーを生成することができる。

一方で、このような自動生成システムの開発は、言語の理解と生成に関する多くの課題を

含んでいる。言語の微妙なニュアンスや文化的な要素を機械に理解させることは、現在の技術においても容易ではない。これらの課題を克服することは、自然言語処理技術の進展に大きく貢献し、より人間らしい感情を持った自然な言語生成を可能にする。

また、自動生成されたキャッチコピーは、従来の手法では考えられなかった創造的なアイデアや表現を提供する可能性がある。これにより、広告業界に新たな風を吹き込み、消費者に新鮮な体験を提供することが期待される。

総じて、韻を踏むキャッチコピーの自動生成は広告業界の効率化だけでなく、言語技術の進展にも寄与する重要な研究テーマである。この研究は、言語の魅力と可能性を最大限に活用し新しいコミュニケーションの形を開発することを目指している。

1.2 本研究の目的

本研究の主要な目的は、自然言語処理（NLP）と機械学習技術を活用し、韻を踏むキャッチコピーを自動生成するシステムを開発することである。このシステムは、多言語に対応し文化的背景を考慮した効果的なキャッチコピーを迅速に生成する能力を持つことを目指す。この技術の進展により言語の響きとリズムを生かした新しい形のコミュニケーション手法が提供される。

この研究はマーケティングと広告業界においてキャッチコピー制作の時間とコストを削減し、広告の効果を最大化することを目標とする。自動化されたシステムにより広告制作プロセスが効率化され、より迅速かつ柔軟な広告展開が可能となる。これにより市場の変化に素早く対応し、ターゲットオーディエンスに適切なメッセージを届けることが可能になる。

さらに言語学や自然言語処理の分野においても創造的なテキスト生成の新たな可能性を探ることができる。この研究によって開発される技術は、言語の深い理解と創造性を融合させ自然言語生成の新たな領域を切り開くことが期待される。これは言語技術の進歩だけでなく、クリエイティブなコミュニケーションの手法に新たな展望を提供する。

総じて本研究は広告業界のみならず、言語技術の発展にも寄与することを目指している。広告の効果を高めると同時に言語の新たな可能性を探求し、多様な文化的背景を持つオーディエンスに対してより深い理解と共感を促すことを目標としている。

1.3 論文構成

本論文は、全7章で構成している。

第2章では、本研究の関連知識について述べる。

第3章では、本研究の先行研究について述べる。

第4章では、本研究の前提条件について述べる。

第5章では、本研究の研究内容・プログラムの概要について述べる。

第6章では、第5章の実験から得られた結果を述べる。

第7章では、第6章の結果を踏まえ、得られた考察について述べる。

第8章では、本研究を通じた結論を述べる。

2 関連知識

本章の 2.1 では自然言語処理とそれを応用した AI を説明する。2.2 では形態素解析を中心に各処理や専門用語を、2.3 では形態素解析ライブラリを論ずる。また 2.4 では韻について、2.5 ではキャッチコピーについて解説する。

2.1 自然言語処理

2.1.1 自然言語処理とは

自然言語処理 (Natural Language Processing, NLP) は、人間の言語をコンピュータが理解し、処理するための技術分野である。この分野は、コンピュータ科学、人工知能、言語学が交差する領域に位置し、テキストや音声データの解析、理解、生成を行うための方法論を提供する。

自然言語処理の主な目的は、人間の言葉をコンピュータが解釈し、それに基づいて有用なタスクを実行できるようにすることである。この技術は、テキストの意味解析、言語の翻訳、音声認識、感情分析など、多岐にわたるアプリケーションに応用されている。例えば、チャットボット、翻訳ソフトウェア、音声アシスタントなどがこれに該当する。

NLP のアプローチには、統計ベースの手法とルールベースの手法が存在する。統計ベースの手法では、大量のテキストデータからパターンを学習し、言語モデルを構築する。一方、ルールベースの手法は、言語学的な規則に基づいて言語の構造を分析する。近年では、ディープラーニング技術を用いた手法が主流となり、より複雑で精密な言語理解が可能になっている。

最大の課題は、言語の多様性と複雑性に対応することである。異なる文化や言語に固有のニュアンス、イディオム、比喩的表現を理解することは困難であるが、機械学習技術の進歩と大量のデータの利用により、この分野は急速に進化している。

類似問題には、機械翻訳、質問応答システム、テキスト要約などがある。機械翻訳では、異なる言語間での文の意味を正確に伝えることが求められる。質問応答システムは、自然言語の質問に対して適切な回答を生成する能力が必要である。テキスト要約では、大量のテキストから主要な情報を抽出し、簡潔にまとめる技術が重要となる。これらの問題は、言語の深い理解と文脈の把握を必要とし、NLP の研究において重要な役割を果たしている。

さらに、感情分析や意見マイニングのような応用分野では、テキスト内の主観的な情報や感情を検出し、分析する技術が重要である。これらの技術は、消費者の意見や感情を理解し、製品やサービスの改善に役立てることができる。

自然言語処理は、人間とコンピュータ間のコミュニケーションを改善し、多くの産業分野に革新をもたらす可能性を秘めた重要な技術である。その進歩は、より自然で効率的な人

間と機械の相互作用を可能にし、情報アクセス、処理、分析の新たな領域を開くことが期待される。

2.1.2 言語理解 AI

自然言語処理を用いた言語理解 AI サービスの発展は様々な分野で行われている。一番有名なサービスが、OpenAI 社が開発した ChatGPT である。人間のような文章をリアルタイムに生成できる最先端の自然言語処理モデルであり、様々なプロンプトに対して守備一貫した応答を行うことが可能なサービスだ。チャットボットとしての側面はもちろんのこと、コンテンツ生成や言語翻訳などの様々なタスクに対応した有益なツールとして 2023 年頃に一躍有名となった。ChatGPT は、OpenAI によって開発された自然言語処理モデルの一つであり、GPT (Generative Pre-trained Transformer) シリーズの進化の一環である。自然言語処理は、コンピュータによる人間の言語の理解と生成を目的とする技術分野であり、その発展は目覚ましいものがある。ChatGPT は、特にトランスフォーマーと呼ばれるモデルに基づいており、このモデルは自己注意機構を用いることで文の文脈を深く理解する能力を有している。この技術により、GPT シリーズは大規模なテキストデータから学習し、様々な言語タスクに適応する能力を身につけた。特に GPT-3 や GPT-4 では、その巨大なパラメータ数と広範な学習データにより、高度な言語理解と生成能力を示している。これにより、ChatGPT は質問応答、対話生成、文章要約、言語翻訳など、多岐にわたる応用が可能となった。将来的には、これらのモデルがさらに進化することで、人間の言語により近い理解と生成が可能となり、人間とコンピュータとのコミュニケーションがより自然かつ効果的になることが期待される。[2]

また、Google 社が開発した Google Bard も言語理解 AI として注目を集めている。2018 年に発表された Google Bard は Google の大規模言語モデル PaLM2 を活用しており、対話アプリケーションとして LamaDa を搭載した実験的な会話型 AI サービスである。自然言語処理の進歩を表したサービスであると共に、Google 社が発表したことから ChatGPT と並ぶ注目度を得ている。Google が持つ検索データベースとの照合やマーケティング技術からも、ビジネスなど幅広いシーンで応用が期待されている。

このように自然言語処理の進歩は様々な産業や学術分野に革新をもたらす可能性を秘めていると考えられており、今後も様々な言語理解 AI がリリースされると考えられる。

2.2 形態素解析

2.2.1 形態素解析とは

形態素解析は自然言語処理の一環であり、テキストを意味を持つ最小単位である形態素に分解する作業である。この分析は文章の意味や構造を理解する上で基礎的な役割を果たす。

例えば、「彼女は図書館に行った」という文を形態素解析すると「彼女」「は」「図書館」「に」「行っ」「た」という形態素に分割される。ここで、「彼女」は名詞、「は」は助詞、「図書館」は名詞、「に」は助詞、「行っ」は動詞の連用形、「た」は助動詞としての役割を持つ。このように文を形態素に分割することで文の意味構造が明確になる。

形態素解析ではまずテキストを単語に分割し、それぞれの単語に対して品詞のタグ付けを行う。この過程では辞書データや文法規則が用いられる。しかし、この過程は言語によって複雑さが異なり特に日本語のように単語間に明確な区切りがない言語では困難を伴う。

形態素解析の応用例としては、機械翻訳、音声認識、情報検索などがある。例えば機械翻訳では正確な形態素解析が翻訳品質を大きく左右する。また、情報検索では検索キーワードを形態素に分解し、関連する文書を効率的に見つけるために用いられる。

形態素解析の課題には曖昧性の解消や新しい単語への対応がある。言語は常に進化し新しい単語や表現が生まれるため、形態素解析システムはこれらの変化に柔軟に対応する必要がある。また、文脈によって意味が変わる単語の解析は特に難しい。

将来的にはより高度なアルゴリズムと大規模なデータセットを用いて、これらの課題を克服しより正確で効率的な形態素解析を実現することが期待されている。

2.2.2 トークナイザ

形態素解析を行う為にはテキストデータを処理しやすい形式に変換するために必要なトークナイザという作業が必要である。テキストを小さな単位（＝トークン）に分割する作業を指し、形態素解析を行う為には欠かせない作業である。

トークンの単位は多様であり、単語単位・文字単位・部分文字単位と大別される。単語単位の分割とは、文字通り文に登場する1単語を1トークンとする分割の行い方である。

「日本人の日本語」と入力された場合は「日本」「人」「の」「日本」「語」と分割される。他の単位に比べて簡単に実装できるというメリットがある反面、複合的な単語を1トークンと計算した場合に正しく形態素解析が出来ない場合も存在する。

その問題を解決するための案として文字単位での分割方法が存在する。1文字を1トークンとして計算する分割方法で、「日本人の日本語」という文字が入力されると「日」「本」「人」「の」「日」「本」「語」と1文字ずつを切り取って分割を行う。総文字数が少ない英語圏で扱われることが多い方法であり、日本語に対して使われることはあまり多くないのが現状である。

部分文字列単位の分割方法では、単語単位で分割しても文字単位で分割しても上手くいかない問題点を解決することが出来る。これは単語を部分文字列に分割し、学習対象の文章に多く現れるものをトークンとする手法である。「日本人の日本語」という文字の入力に対し、「日本」のみを分割する。この手法は単語を意味のあるトークン単位に分割できるだけでなく、トークナイザが持つ語彙の数を小さくできるという利点も存在する。日本語における長文に対して有効的な分割方法である。

単語と単語の区切りに空白が用いられる英語と違い、日本語では単語と単語の区切りに空白が存在しない。形態素解析という行為を通すことによって初めて日本語の単語を区切る事が可能となるのである。トークナイザはその形態素解析において重要なパラメータであり、特に韻を踏むという目的に対しては重要な意味合いを持つ。日本語の単語をしっかりと認識できるよう、本研究では単語単位での分割を行なった。

2.2.3 構文解析

形態素解析から発展した他の解析処理も記載する。自然言語処理は形態素解析・構文解析・意味解析・文脈解析の4つに大別される。自然言語は人工言語と異なって解釈の余地が大きい為、解析が進むにつれてコンピュータの意味理解の精度は落ちる。だが、解析処理を多く行うことによって私たちが普段目にしているテキストや文章を細かに応用・処理することが可能となる。下記より解析のフェーズ毎に分けて、解説を行う。

構文解析とは、単語同士の関係を推測する処理である。文の内容を解析し、その構文要素がどのような関係にあるかを明らかにすることによってコンピュータの処理が容易に行えるのだ。「眩しい光に感動を覚えた」という文が与えられた時、形態素解析を行うことで「眩しい」「光」「に」「感動」「を」「覚えた」と分割が行える。対して構文解析では、「眩しい：形容詞」「光：名詞」「に：助詞」「感動：名詞」「を：助詞」「覚え：動詞」「た：助動詞」と形態素に対して品詞付けを行う。あくまで一例ではある為、複数の解釈の余地がある場合があることから品詞を絞ることが出来ない場合も存在する。修飾関係や目的関係を細かにすることで、各形態素の品詞と役割を明らかにする作業となる。このように文の中からいくつかの構文を推測するという作業が構文解析である。

2.2.4 意味解析

意味解析とは、主に統計的な方法を用いて単語同士の結びつきを見出す作業を指す。「眩しい光に感動を覚えた」という文が与えられた時、「光」が「眩しい」ものであり「感動」は「覚える」ものと理解する事が可能となる。入力・学習させる情報量が多ければ多いほど意味解析の精度は上がり、単語同士の関連性と相関関係が結びついていく。検索エンジンの候補予測などはこの解析が用いられており、検索される言葉が増えるたびに意味解析のデータベースは蓄積されていく。

2.2.5 文脈解析

文脈解析とは、文と文の繋がりを考える工程である。複数の文に対して文同士のつながりを解析し、文章の背景を含む多くの情報量を必要とする。解析における難易度は他の3つと比べてとても高く実用分野への応用は未だ遠いと考えられていたが、前述したOpenAI社のChatGPTやGoogle社のGoogle Bardの登場により実用化の可能性が一気に高まった。文章中に登場する指示語の内容理解や登場人物の心情を解析するなどといった

人間に近いコミュニケーションをコンピュータが理解する作業である為、今後の AI 市場において最重要とも言える解析手段である。

2.3 形態素解析ライブラリ

形態素解析を行う上で必要となるツールを形態素解析ライブラリと呼称する。いくつかの種類が存在し、それぞれのライブラリにおける独自の特徴と利点を説明する。

2.3.1 MeCab

MeCab は、京都大学情報学研究科-日本電信電話株式会社コミュニケーション科学基礎研究所 共同研究ユニットプロジェクトを通じて開発されたオープンソースである。言語・辞書に依存しない汎用的な設計を基本方針としており、精度が高いことや Python・R などのプログラミング言語で利用しやすいことからエンジニアから研究者まで幅広い利用者の間で活用されている。MeCab が行う解析の元となる日本語データベース (=辞書) は 2 種類存在し、最新のワードなどを含む IPA 辞書と古文まで含んだ日本語辞書の Unidic を参照している。解析対象の言葉が辞書に未登録の場合に品詞を推定する未知単語推定という機能や、文章の単語感に空白文字を挿入する処理を行う分かち書きという機能など、他のエンジンに比べて多彩な機能の多さが強みと言える。[3]

2.3.2 Janome

日本人のソフトウェアエンジニアが開発した形態素解析ライブラリ Janome は、特に日本語の形態素解析に強く多くの日本特有の語彙や文法に対応している。名前の由来が「蛇の目」となっており、Python (蛇) で作られた形態素解析ツールである。その為 MeCab のデフォルトとして内蔵されている辞書を流用しており、解析結果の精度としては MeCab と同等のものとなる。日本語に特化した形態素解析ライブラリとなっており、本研究では MeCab と Janome を利用してプログラミングの記載を行った。

2.3.3 Sudachi

ワークス徳島人工知能 NLP 研究所により提供されているソフトウェアの 1 つで、日本語に特化した形態素解析ライブラリである。Sudachi に内蔵されている辞書は Sudachi 辞書と呼称され、MeCab 内蔵辞書の 1 つである UniDic をベースとした見出しや品詞などの形態素解析の基礎となる情報を整えた最先端の辞書となっている。既存の辞書の長所を取り込んだ上で形態素解析精度を上げている為、近年利用される事が増加したライブラリである。

2.3.4 GiNZA

リクルートの AI 研究機関である MegagonLabs と国立国語研究所との共同研究成果の学習

モデルを用いた自然言語処理ライブラリである。高速・高精度な解析処理を謳っており、Python 環境で利用する事が可能となっている。多言語対応している自然言語処理ライブラリ spaCy と前述した Sudachi の技術が採用されており、簡単に日本語の自然言語処理に取り掛かることが出来る。形態素解析だけでなく、構文解析や意味解析などにも対応している点が最大の強みである。

2.4 韻

2.4.1 韻とは

韻は、言葉のリズムや響きを生み出す要素であり、詩や歌詞、キャッチコピーなどの創造的なテキストで特に重要な役割を果たす。韻は、言葉の美しさや音楽的な要素を強調し、聞き手の感情に訴えかけ、記憶に残りやすくする。

韻の種類は多岐にわたり、その主なものには完全韻、部分韻、内韻、脚韻などがある。完全韻は、単語の最後の母音とそれに続く音が完全に一致する場合を指し、特に詩や歌のリフレインでよく使われる。部分韻は、音が完全には一致しないが似ている場合で、より自然で流れるような韻を作り出す。内韻は、一つの詩行内で複数の単語が韻を踏むことを指し、テキストにリズム感をもたらす。脚韻は、詩の末尾や近い行で韻を踏むことを意味し、詩や歌の結びつきを強化する。

韻の使用は、リズム感や調和を生み出し、言語表現に深みを与える。特にキャッチコピーのような短いテキストでは、韻を用いることでメッセージの印象を強化し、記憶に残りやすくなる。詩や歌詞においては、韻の使用が読み手や聞き手の感情に訴えかける力を増す。

自然言語処理の分野では、韻のパターンを識別し、生成するアルゴリズムが開発されている。これにより、創造的なテキストの生成や歌詞の自動作成、キャッチコピーの生成が可能になる。音韻論や詩学の理論を取り入れた分析は、テキスト生成における韻の自然さや美しさを高める。

韻の分析における課題には、言語の多様性や音の複雑性への対応が含まれる。異なる言語には異なる韻の形式が存在し、それらを正確に理解し再現することは容易ではない。また、言語によっては韻を形成する音の組み合わせが限られており、創造性を発揮するのが難しい場合もある。これらの課題を克服するためには、より高度な音韻分析技術や多言語対応のアルゴリズムの開発が必要である。

将来的には、これらの進歩により、多様な言語と文化に対応した韻の生成と分析が可能になり、言語の美しさや表現の幅をさらに広げることが期待されている。韻は、言葉のリズムや響きを通じて、メッセージの伝達や感情表現に深い影響を与える要素であり、その研究と応用は言語学と自然言語処理の分野で重要な位置を占めている。

2.4.2 頭韻

押韻方法は頭韻・脚韻・全踏の3種類に大別される。

頭韻とは、単語の頭で韻を踏む事を指す。詩歌などにおける単語・句の始まりに同じ韻を合わせる事で、綺麗な聴き心地を演出する。

2.4.3 脚韻

脚韻とは、単語の終わりで韻を踏む事を指す。詩歌などにおける単語・句の終わりに同じ韻を合わせる事で、綺麗な聴き心地を演出する。頭韻に比べて韻を踏むという行為が分かりやすく伝わる為、歌詞やキャッチコピーなどで多用される傾向にある。韻を踏んだ言葉の多くは脚韻である事が多い。

2.4.4 全踏

全踏とは、全く同じ母音・文字数で韻を踏む事を指す。HIPHOP ジャンルの音楽で多用される事が多く、頭韻や脚韻に比べて聴き心地が格段に良い。クリエイティビティの高いキャッチコピーや何かと繰り返してしまうコピーライティングなどにも応用される。「あいうえお」の母音が設定されている日本語だからこそ完成される韻の美学における最高傑作である。

2.5 キャッチコピー

キャッチコピーは、マーケティングや広告において使用される短いフレーズであり、製品やサービスの特徴を効果的に伝え、消費者の注意を引きつける役割を果たす。キャッチコピーは、ブランドのイメージを形成し、メッセージを強調するために重要な手段である。

キャッチコピーの重要性は、限られた言葉の中で製品やサービスの魅力を伝える必要がある。創造性と簡潔さが求められることにある。効果的なキャッチコピーは、視覚的な広告やメディアキャンペーンにおいて消費者の記憶に残り、ブランド認知度の向上に寄与する。

キャッチコピーを作成する際には、ターゲットオーディエンスの興味やニーズを考慮し、製品の独自性や価値を明確にする必要がある。また、感情的な反応を引き出すために、説得力のある言葉を選ぶことが重要である。効果的なキャッチコピーは、短くて印象的であり、繰り返し使われることで消費者の心に定着する。

キャッチコピーの作成における課題は、オリジナリティと親しみやすさのバランスを見つけることである。また、文化的背景や現在のトレンドを理解し、それに応じてメッセージを調整する必要がある。自然言語処理技術を利用することで、キャッチコピーの創造過程を支援し、より効果的なフレーズの提案が可能になる。

将来的には、自然言語処理と人工知能の進歩により、より洗練されたキャッチコピーの自動生成が期待される。これにより、広告業界におけるクリエイティビティの向上と効率化が実現される可能性がある。キャッチコピーは、言葉の力を通じて製品やサービスの価値を伝える重要なツールであり、その研究と開発はマーケティングと広告の分野で引き続き重視されている。

3 先行・類似研究

本章の 3.1 では他大学で行われている類似研究について説明する。また、3.2 では広告業界である電通でリリースされているクリエイティブ自動生成 AI ソリューションについて説明する。

3.1 自然言語処理によるキャッチコピー自動生成の研究

我々の研究テーマである自然言語処理によるキャッチコピー自動生成の研究は実際にも様々な方法で行われている。それらの中で2つの研究に注目する。

1 つ目は松平・萩原の研究[4]であり、電子化辞書と遺伝的プログラミングを用いたキャッチコピー作成支援システムである。知識ベースとしては膨大な知識を蓄積した EDR 電子辞書を活用しており、遺伝子的プログラミングではユーザーが選択したいくつかの単語を使用してキャッチコピーを作成するように作成されている。また、質より量、批判厳禁を旨とした発散的発想支援に基づいた手法を提案している。

2 つ目は山根・萩原の研究[5]であり、既存のキャッチコピーコーパスや Google 日本語 N-gram コーパス、そして Web 上のユーザーの嗜好を反映したキャッチコピーシステムである。提案システムは 24,472 個のキャッチコピーを収録したコーパスから選択したモデルキャッチコピーを用いて、SNS 投稿から得られる知識を用いてキャッチコピー候補を生成する。候補は、大規模均衡コーパスから得られる自然度・広告キャッチコピーを用いた意味関係スコア・SNS サイトから取得した嗜好度の 3 つが指標になって選定される。

3.2 広告業界によるクリエイティブ自動生成 AI ソリューションの研究

広告業界ではキャッチコピー自動生成の研究のみにとどまらず、クリエイティブ自動生成 AI ソリューションが研究・開発されている。2017 年に静岡大学情報学部行動情報学科の狩野研究室と広告代理店の電通が共同でキャッチコピー生成システム「AICO」を開発し、その後電通グループ 4 社共同でクリエイティブ自動生成 AI ソリューション「CXAI」が開発された。[6][7]

「CXAI」とは、バナー自動生成&効果予測ツール「ADVANCED CREATIVE MAKER」や独自開発チャットボットツール「Kiku-Hana」、流行予測ツール「TREND SENSOR」などの AI モジュールが 15 種類以上利用することが出来る。それらの中でバナー自動生成&効果予測ツールである「ADVANCED CREATIVE MAKER」に注目する。

「ADVANCED CREATIVE MAKER」とは、過去に配信されたインターネット広告バナーの表現とクリック率実績をディープラーニングで分析し、パフォーマンスの高いバナーを約 5 秒間に 1 枚生成するシステムであり、2 つの予測エンジンが搭載されている。同シス

テムでは、入力されたキーワード・訴求軸といったオリエンテーションに対して 1 つ目の予測エンジンが、パフォーマンスの高いと考えられるクリエイティブ要素を割り出す。次に「AICO」を用いて自動生成したキャッチコピーと出来上がったクリエイティブを、2 つ目の予測エンジンがより精緻にクリック率をシミュレーションし、効果が高いと予測されるものを残す。[8]

4 前提条件

本章の 4.1 では韻を踏む上で必要となる母音辞書を論ずる。4.2 では通常の平仮名ではない特殊な文字をパターン毎に解説し、4.3では入力した単語と同じ母音を参照する辞書についての設定を説明する。

4.1 母音辞書

日本語において韻を踏むという事は、「あいうえお」に大別される母音が同じ言葉を繰り返すという意味合いである。意味や内容は違っていても、母音に直した時に「あいうえお」が一致していれば韻を踏む作業は遂行されているのである。

本研究においては入力された文字から母音を取得し、同じ母音の単語を出力するという流れに沿う。その為、母音辞書と呼ぶ母音を参照する辞書を作成した。母音を「ア」「イ」「ウ」「エ」「オ」の5種類に分類し、日本語の文字をそれぞれの分類に区別。

```
# 母音辞書
rhyme_map = {
  'ア': 'アカガサザタダナハバパマヤラワウ',
  'イ': 'イキギンジチヂニヒビピミリ',
  'ウ': 'ウクグスズツツヌフブムユルン',
  'エ': 'エケゲセゼテデネヘベペメレ',
  'オ': 'オコゴソゾトドノホボポモヨロヲ',
}

# 拗音辞書
youon_map = {
  'ャ': 'ア',
  'ュ': 'ウ',
  'ョ': 'オ',
  'ア': 'ア',
  'イ': 'イ',
  'ウ': 'ウ',
  'エ': 'エ',
  'オ': 'オ'
}
```

図1：母音・拗音辞書

濁音・鼻濁音・小文字などもそれぞれの発音から5種類に当てはまるよう選択した。

4.2 拗音・促音・撥音・長音

日本語の平仮名は 3.1 で記した母音辞書だけでは収まらない。小文字とも呼ばれる「ゃゅょ」などの拗音という音や、「っ」で表される一拍分の長さを持つ長音などが存在する。いくつかの要点に分けてそれらを記載し、私たちがどのように母音辞書に分類したかを説明する。

4.2.1 拗音

「邪魔（じゃま）」「修理（しゅうり）」「病気（びょうき）」などに含まれる「ゃ」「ゅ」「ょ」が拗音に当たる。小さな文字の「ゃ」「ゅ」「ょ」は、いの段（き・し・ち・に・ひ・み・り・ぎ・じ・び・び）の文字の後ろにつく事が決まりとなっている。その為、音節としては前の文字と合わせて一拍分（一文字分）の長さとなる。「邪魔」という単語が入力された場合は、「ああ」という母音検索となり「修理」と入力された場合は「ううい」という母音検索となるのだ。よって前の文字と合わせて一文字分であり、母音は拗音の文字に依存するというプログラミングを拗音辞書として新たに追加した。

4.2.2 促音

「尻尾（しっぽ）」などに含まれる「っ」が促音に当たる。促音自体がはっきりと発音される事はないものの、小さく詰まったような音は確かに存在する。また、母音はないものの一拍分（一文字分）の長さを持つことが特徴的な音となる。「尻尾」という単語が入力された場合は「いうお」という母音検索となるよう、促音「っ」は「ウ」の母音判定となるように母音辞書を設定した。

4.2.3 撥音

「団子（だんご）」などに含まれる「ん」が撥音に当たる。鼻にかかるような音で、速音と同じように母音自体はないものの一拍分の長さを持っているのが特徴的である。撥音の発音に関しては様々な記述を調べたが、一番近い発音としては「ウ」の母音設定に近いことが確認できた。本来であれば「ん」自体は「ア」「イ」「ウ」「エ」「オ」のどの母音にも該当しない文字であるが、「ン」として新たな母音設定をしてしまうと「ん」の入力単語には「ん」でしか返せなくなってしまう、韻の出力幅が大幅に減ってしまう。本研究では韻を利用したキャッチコピーが題材である為、「ん」自体の発音を「ウ」に大別する母音辞書を設定した。

4.2.4 長音

「ロープ」などに含まれる「ー」が長音に当たる。伸ばし棒とも言われる単語で、母音を通常の倍伸ばす音のことを指す。音節として「ー」の文字のみで一音節の長さを持ち、母音はパターンによって分類されるように設定した。前の文字母音が「オ」の時には「ー」

の母音は「ウ」と設定され、前の文字母音が「ア」「イ」「ウ」「エ」の時には「ー」の母音は前の文字母音と同じ韻に設定するように設定。「オール」と入力した際には「おうる」という母音検索となり、「ゲーム」と入力した際には「ええう」という母音検索となる。他の音と比べて特殊ではあるものの、幅広い単語に対応できるよう画策した。

4.3 参照辞書

入力した文字と同じ韻を検索する上で必須となる母音辞書を設定した。国語辞典など日本語の単語が網羅されている辞書を参照にした場合、出力される韻が一辺倒となってしまう。対極に、参照する辞書に登録されている単語が少なければ出力される韻も少なくなってしまう。双方を考慮した結果、キャッチコピーを生成する上で必要となる単語を自身で設定し参照辞書を作成した。設定した母音辞書の内容を下記に記す。

- ・ AppleMusic - 2023 年トップソング 50
- ・ 日本の大学 48 校のキャッチコピー
- ・ 新聞広告賞 2019-2023 年 入賞作品
- ・ フジサンケイグループ 2018-2022 年 入賞作品
- ・ 読売広告大賞 2018-2022 年度 受賞作品

5 実験概要

本章の5.1ではフローチャートを使い、本研究の流れを説明する。また、5.2では自作関数の重要な部分を説明する。

5.1 フローチャート

本研究では、以下の流れで実験結果を出力する。青色の部分は韻出力、赤色の部分はキャッチコピー出力のフローチャートである。

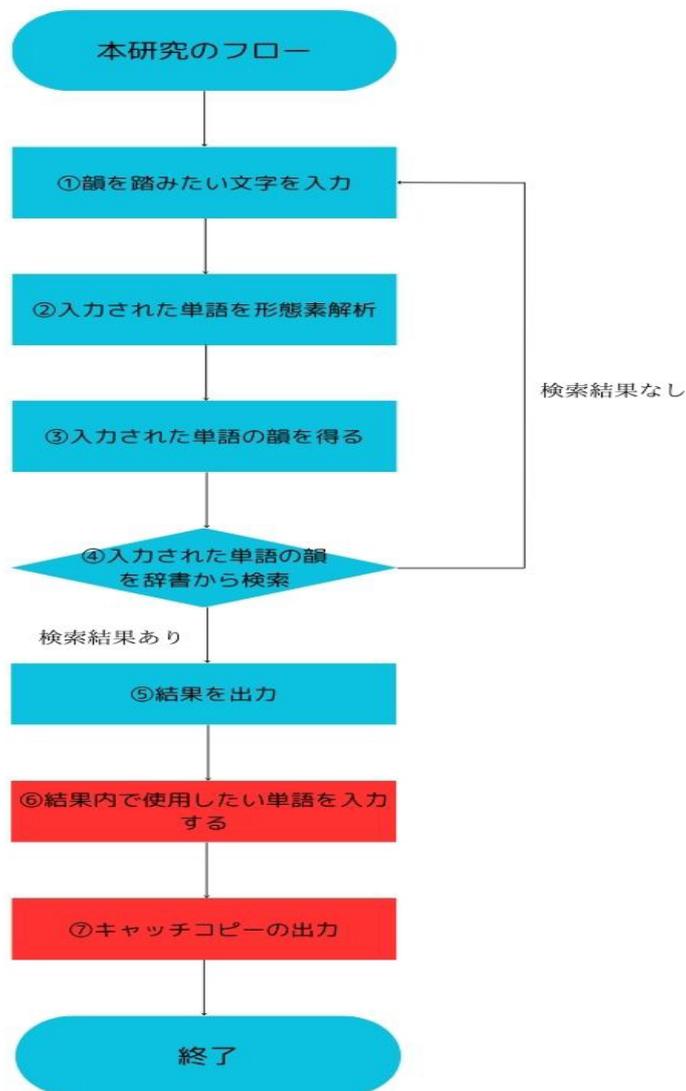


図2：本研究のフローチャート

辞書や韻へのアプローチは前章で説明した前提条件のとおりである。

また、main 関数は以下のようになる。

```
def main():  
  
    with open('jisyo.txt', 'r', encoding='utf-8') as f:  
        content = f.read()  
        rhyme_map = list(analyze(content))  
  
    while True:  
        try:  
            s = input('読みたい文字は?> ')  
        except KeyboardInterrupt:  
            break  
  
        if not s:  
            continue  
  
        input_words = list(Analyzer().analyze(s))  
  
        for word in input_words:  
            rhyme = parse_reading(word.reading)  
  
            toks = find_toks(rhyme_map, rhyme)  
  
            show_toks(toks)  
  
            word2 = input('以上から使用する単語を入力してください: ')  
  
            catchphrases = generate_catchphrase(s, word2)  
  
            print(' [ キャッチコピー結果一覧 ] ')  
            for phrase in catchphrases:  
                print(phrase)  
  
main()
```

図 3 : main 関数

5.2 プログラム概要

図 2 に記載されているフローチャートで使われている数字でそれぞれの自作関数を説明していく。

②入力された単語を形態素解析

図 4 のプログラムで入力された単語を解析し、トークン列を生成する。また図 5, 6 のプログラムで重複しているトークンを除外し、カタカナに変換する。

```

def analyze(text):
    """
    テキストを解析してトークン列を生成する
    """
    a = Analyzer()
    toks = a.analyze(text)
    toks = unique(toks)

    for tok in toks:
        tok.rhyme = parse_reading(tok.reading)

    yield tok

```

図 4 : analyze 関数

```

def unique(toks):
    dst = []

    for tok in toks:
        if not is_in(dst, tok):
            dst.append(tok)

    return dst

```

図 5 : unique 関数

```

def is_in(toks, tok):
    """
    toksにtokが入っていたらTrue, でなければFalseを返す
    """
    for t in toks:
        if t.surface == tok.surface:
            return True

    return False

```

図 6 : is_in 関数

③入力された単語の韻を得る

前提条件で記載した長音・拗音・促音の韻を出力するために if・else 文を使用する。また母音辞書は図 1 を使用している。

```

def parse_reading(reading):
    global rhyme_map, youon_map
    rhyme = ''

    for i, c in enumerate(reading):
        if c == ' ':
            if rhyme[-1] == 'オ':
                rhyme += 'ウ'
            else:
                rhyme += rhyme[-1]
        elif c in youon_map:
            rhyme = rhyme[:-1] + youon_map[c]
        else:
            for k, v in rhyme_map.items():
                if c in v:
                    rhyme += k
                    break

    return rhyme

```

図7：parse_reading 関数

④入力された単語の韻を辞書から検索

③で得た単語の韻と一致する辞書内のトークン列を探す。また、文字数も一致する単語を出力させるために len 関数を使用している。

```

def find_toks(map_toks, rhyme):

    for tok in map_toks:
        if rhyme in tok.rhyme and len(rhyme) == len(tok.rhyme):
            yield tok

```

図8：find_toks 関数

⑤結果を出力

④で一一致したトークン列を表層形、読み方、母音の順で出力させる。

```

def show_toks(toks):
    toksを表示する

    for tok in toks:
        print(f'{tok.surface}|({tok.reading}, {tok.rhyme})')

```

図9：show_toks 関数

⑦キャッチコピーの出力

word1 は入力された単語であり、word2 は⑥で選択した単語である。また、図10の phrases は他大学のキャッチコピーを参考にして作成したものである。

そのため、入力した単語と韻出力された中から選択した単語から作成されていないので、文脈として正しくないキャッチコピーが生成されることもある。

```

def generate_catchphrase(word1, word2):
    """
    |
    |
    # フレーズ
    phrases = [
        f"{word1}と{word2}、新しい未来への一歩。",
        f"発見と感動の{word1}、{word2}の世界。",
        f"{word1}の革新、{word2}の輝き。",
        f"変わる世界、変わらない{word1}と{word2}。",
        f"無限の可能性を秘めた{word1}、{word2}の力。"
    ]

    # 生成されたキャッチコピーを返す
    return phrases

```

図 1 0 : generate_catchphrase 関数

6. 結果

本章の 5.1 では、本研究のメインテーマである法政大学のキャッチコピー実装結果について説明する。また 5.2 では、ゼミの曜日である木曜日のキャッチコピー実装結果について説明する。

6.1 「法政」の実装結果

入力する文字を「法政」とすると、図 1 1 のような韻出力結果になる。また、phrases は図 1 0 を使用している。

```
-----
〈読みたい文字は?〉: 法政
共生(キョウセイ, オウエイ)
創成(ソウセイ, オウエイ)
早慶(ソウケイ, オウエイ)
透明(トウメイ, オウエイ)
探的(ヒョウテキ, オウエイ)
証明(ショウメイ, オウエイ)
特定(トクテイ, オウエイ)
```

〈以上から使用する単語を入力してください〉:

図 1 1 : 「法政」での韻出力結果

出力された単語の中から「創成」を選択すると以下のような出力結果になった。

法政と創成、新しい未来への一歩。

発見と感動の法政、創成の世界。

法政の革新、創成の輝き。

変わる世界、変わらない法政と創成。

無限の可能性を秘めた法政、創成の力。

```
〈以上から使用する単語を入力してください〉: 創成
【キャッチコピー結果一覧】
法政と創成、新しい未来への一歩。
発見と感動の法政、創成の世界。
法政の革新、創成の輝き。
変わる世界、変わらない法政と創成。
無限の可能性を秘めた法政、創成の力。
```

図 1 2 : 法政のキャッチコピーの実装結果①

また、選択する単語を「共生」にすると、図 1 3 のような実装結果が出力される。

<読みたい文字は?>: 法政
 共生(キョウセイ, オウエイ)
 創成(ソウセイ, オウエイ)
 早慶(ソウケイ, オウエイ)
 透明(トウメイ, オウエイ)
 標的(ヒョウテキ, オウエイ)
 証明(ショウメイ, オウエイ)
 特定(トクテイ, オウエイ)
 <以上から使用する単語を入力してください>: 共生
 【キャッチコピー結果一覧】
 法政と共生、新しい未来への一歩。
 発見と感動の法政、共生の世界。
 法政の革新、共生の輝き。
 変わる世界、変わらない法政と共生。
 無限の可能性を秘めた法政、共生の力。

図 1 3 : 法政のキャッチコピーの実装結果②

6.2 「木曜」の実装結果

5.1 で扱っていた phrases は法政大学のキャッチコピーに合う文章で作成していたため、今回は図 1 4 を使用していく。また、phrases はキャッチコピーサイトを参考にして作成したものである。

```

# フレーズ
phrases = [
  f"2024年{word2} 忘れられない{word1}。",
  f"隠れた{word1} 今すぐ{word2}",
  f"毎週通った{word1} 気づけば三月{word2}。",
  f"ホッとすると{word1}と{word2}。",
  f"無限の可能性を秘めた{word2} {word1}の力。"
]
  
```

図 1 4 : 木曜のキャッチフレーズ

入力する文字を「木曜」とすると、図 1 5 のような韻出力結果になる。

<読みたい文字は?>: 木曜
 総合(ソウゴウ, オウオウ)
 創造(ソウゾウ, オウオウ)
 東京(トウキョウ, オウオウ)
 協力(キョウリョク, オウオウ)
 行動(コウドウ, オウオウ)
 校長(コウチャウ, オウオウ)
 卒業(ソツギョウ, オウオウ)
 彫刻(チョウコク, オウオウ)
 賞状(ショウジョウ, オウオウ)
 本当(ホンドウ, オウオウ)
 道標(ドウヒョウ, オウオウ)
 妄想(モウソウ, オウオウ)
 本能(ホンノウ, オウオウ)
 能力(ノウリョク, オウオウ)
 臆病(オクビョウ, オウオウ)
 包装(ホウソウ, オウオウ)
 宝物(ホウモツ, オウオウ)
 問答(モンドウ, オウオウ)

図 1 5 : 「木曜」での韻出力結果

出力された単語の中から「卒業」を選択すると以下のような出力結果になった。

2024 年卒業 忘れられない木曜。

隠れた木曜 今すぐ卒業
毎週通った木曜 気づけば三月卒業。
ホッとする木曜と卒業。
無限の可能性を秘めた卒業 木曜の力。

〈以上から使用する単語を入力してください〉：卒業
【キャッチコピー結果一覧】
2024年卒業 忘れられない木曜。
隠れた木曜 今すぐ卒業
毎週通った木曜 気づけば三月卒業。
ホッとする木曜と卒業。
無限の可能性を秘めた卒業 木曜の力。

図16：法政大学のキャッチコピーの実装結果

7 考察

本研究の目的は、自然言語処理（NLP）と機械学習技術を活用し、韻を踏むキャッチコピーを自動生成するシステムを開発することであった。結果として韻を踏むことが出来たが、単語の意味からキャッチコピーを自動生成するところまでは完成することは叶わなかった。原因は様々あると考えられるが、1番の原因はデータベースの知識不足だと考える。3.2 で記載したキャッチコピー生成システム「AICO」で使用されているデータベースは今までの電通の知識から作られたものであり、3.1 で説明した2つの研究も莫大なキャッチコピーをデータベースに入力していた。このことから、データベースの知識不足からキャッチコピーの自動生成が困難だったと考えられる。その為、本研究では韻を踏む単語以外のキャッチコピーをプログラム内の phrases に入力することによって、韻を踏んでいるキャッチコピーを出力させることにした。

また、4.2 の②で記載した「入力された単語を解析し、カタカナに変換する」能力の低さが目立った。例えば、入力する単語をゼミ教授である「五島先生」と入力すると結果として「ゴシマセンセイ」と変換されてしまう。以上の問題点は、本研究で用いた Janome で変換する応用揚力がないことだと考える。しかしながら、我々が使用できる形態素解析エンジンの中で改善することが出来るエンジンはなかった。

しかしながら韻を踏んだ言葉を利用したキャッチコピー生成としては完成に近づく事が出来た為、人間が考える時間の短縮や創造性の拡張という本来の目的を果たすこともできたと言える。歌詞や句だけでなく、広告業界にも応用できるプログラミング・実施結果である。

8 おわりに

本論文の目的は、自然言語処理（NLP）と機械学習技術を活用し韻を踏むキャッチコピーを自動生成するシステムを開発することであった。研究方法としては主に Python を用いて研究を行い、その結果以下の 2 点が明らかになった。

第一に、韻を自動生成するプログラミングの構築を行うことができた。第二に、キャッチコピーのテンプレートを利用した状態で同一の韻を利用したキャッチコピーの生成を行うことができた。以上の調査結果から、言語の響きとリズムを生かした新しい形のコミュニケーション手法が提案できたと考える。

本研究の社会的意義はキャッチコピー制作の時間とコストを削減し、広告の効果を最大化することである。自動化されたプログラムにより広告制作プロセスが効率化され、より迅速かつ柔軟な広告展開が可能となる。これにより市場の変化に素早く対応し、ターゲットオーディエンスに適切なメッセージを届けることが可能になる。

最後に、本研究の残された課題と今後の発展について主に 3 点記載する。

1 つ目の課題は形態素解析ライブラリ Janome における機能性の弱さである。現代の流行語や個人名などといった内包辞書に参照できない単語入力に対して、正確な韻が出力されないケースが多々見受けられた。形態素解析における単語分割が上手くできていないという問題点が顕著に現れた結果だと考えられる。

2 つ目の課題は韻を踏むという判断基準を音節ではなく文字基準にした点である。本研究では出来るだけ発音した時に韻を踏メルよう画策してプログラミング記載を行った。しかしながら、文字基準で踏めている単語同士でも発音をすることで少し韻が踏めていないように感じる単語も出力された。こればかりは感覚的なものである為数値化したデータで記載することは難しいが、音節や発音に関してもう少し知識をつけた上でプログラミングを行えば更に発展性があると考えられる。

3 つ目の課題はキャッチコピーの本質的な自動生成が行えなかったことである。本研究のプログラミングでは、キャッチコピーを生み出すテンプレートをいくつかプログラミング内に記載することで「自動生成」を謳っていた。本来であれば意味解析などを利用した機械学習を用いた上で「自動生成」を名打つべきだったと反省している。

今後の展望としては、上記の問題点 3 つを解決したプログラミングを実行し、2 つとして被ることのないキャッチコピーの自動生成を行えるように設置したい。また、出力する韻の幅やニュアンスの可変性などもパラメータで設定できる精度まで引き上げたいと考える。

参考文献

- [1] 総務省, 第2部 情報通信分野の現状と課題,
<https://www.soumu.go.jp/johotsusintokei/whitepaper/ja/r05/html/nd243220.html#:~:text=%EF%BC%882%EF%BC%89%20%E5%BA%83%E5%91%8A,%E3%81%AB%E6%88%90%E9%95%B7%E3%81%97%E3%81%A6%E3%81%84%E3%82%8B%E3%80%82>
(参照 2024-01-02)
- [2] ITmedia, AI プロ集団から見た「ChatGPT の歴史」 たった5年で何が起こったのか,
<https://www.itmedia.co.jp/news/articles/2303/17/news200.html>
(参照 2024-01-06)
- [3] 工藤 拓, 形態素解析の理論と実装 (実践・自然言語処理シリーズ), 言語処理学会, 東京 (2018)
(参照 2024-01-07)
- [4] 松平智史, 萩原将文, 電子化辞書と遺伝的プログラミングを用いたキャッチコピー作成支援システム,
https://www.jstage.jst.go.jp/article/ieejeiss/124/1/124_1_164/_article/-char/ja/
(参照 2024-01-04)
- [5] 山根宏彰, 萩原将文, ウェブにおけるユーザの嗜好を繁栄させたキャッチコピー自動生成システム,
https://www.jstage.jst.go.jp/article/jjske/13/4/13_493/_article/-char/ja
(参照 2024-01-04)
- [6] 狩野芳伸, 自然言語処理によるキャッチコピー自動生成の評価に関する研究,
https://www.yhmf.jp/.assets/52_05.pdf
(参照 2024-01-05)
- [7] 電通報, 電通がいよいよ「クリエイティブ AI」を解き放つ! 自動生成 AI ソリューション「CXAI」,
<https://dentsu-ho.com/articles/7769>
(参照 2024-01-05)

[8] 電通報, バナーを自動生成する AI ツール「ADVANCE CREATIVE MAKER」(β版)を開発,

<https://dentsu-ho.com/articles/6021>

(参照 2024-01-05)