卒業論文

# LLM による文生成と言語モデルによる文の流暢さを用いた大喜利生成

高橋巧実

2024 年 3 月 19 日

岐阜大学 工学部 電気電子・情報工学科 情報コース
鈴木研究室

本論文は岐阜大学工学部に
学士（工学）授与の要件として提出した卒業論文である.

高橋巧実

指導教員：
鈴木 優 准教授

# LLM による文生成と言語モデルによる文の流暢さを用いた大喜利生成*

高橋巧実

**内容梗概**

本研究では，LLM(Large Language Model; 大規模言語モデル) の出力結果を加工することにより，自動的に大喜利の回答を生成することを目的とする．「A みたいな B，どんなの？」という形式のお題に対する回答を生成する方法の提案を行う．本稿では，A を「属性語」，B を「基本語」と呼ぶ．教師あり学習を用いて回答の生成を行う場合，十分な回答データの収集が困難であるため面白い回答の生成が難しい．代表的な LLM の一つである ChatGPT で回答を生成したところ，面白いと感じる回答は少なかった．生成された回答は予想外かつ納得感のある回答でなかったためであると考えられる．そこで，ChatGPT で基本語の関連文を生成し，生成した文中の単語を属性語の関連語と入れ替えることで予想外な回答を生成できると仮定した．パープレキシティなどの言語モデルのスコアは，文の流暢さを表しているといえる．我々は言語モデルのスコアで回答の流暢さを計算することで，意外性を考慮できると仮定した．本研究の実験で，単語の入れ替えによって面白い回答が生成できるかどうか，言語モデルのスコアを利用して面白い回答を抽出できるかどうかを確認した．

**キーワード**

大喜利，言語モデル，生成モデル，LLM，単語入れ替え

*岐阜大学 工学部 電気電子・情報工学科 情報コース 卒業論文, 学籍番号: 1203033099, 2024 年 3 月 19 日.

# 目次

# 図目次

# 表目次

# 第 1 章　はじめに

本研究では，大喜利の回答の自動生成を目指す．大喜利とは，与えられたお題に対して面白い回答をすることによって笑いを生み出す演芸の一つである．我々は，大喜利の回答の自動生成システムが実現することで，人々が自分の興味があるお題の回答を見ることができるようになり，お笑いに触れる機会が増えると考える．

ChatGPT*が登場した 2022 年 11 月以降，LLM(Large Language Model; 大規模言語モデル) に関連する論文の投稿数が急激に増加している [1]．また，ChatGPT の他にも，Bard†などの，Web ページ上から LLM を利用できるサービスが登場している．LLM の長所の一つに，与えられたプロンプトに従って文章を自動生成できる点が挙げられる．我々は，複数のお題に対して ChatGPT を用いて大喜利の回答を生成した．生成された回答の中で，面白いと感じる回答は少なかった．LLM を訓練する際は主に二つの段階 1) 大規模コーパスの学習 2) 人間のフィードバックによる出力の調整 に分けて進める．1) では，文法や知識を獲得している．2) では，プロンプトの指示に従い，有害な表現や不正確な情報を含まない出力をするように調整される．1) で用いるコーパスは一般的な内容の文章で構成されている．2) では，意外性のある文章を出力するような調整は行われていない．以上から，我々は LLM では大喜利の面白い回答を生成することが困難であると考える．

我々は，意外性のある回答が面白い回答であると考えた．意外性をもつ文とは，その内容の予測はできないが納得はできる文である．面白い回答には聞き手の予想を裏切り，納得もさせるという絶妙なバランス感が必要である [2] といえるためである．

我々は，LLM で生成した文をベースに単語の入れ替えと言語モデルのスコアによって大喜利の回答の自動生成を試みる．本研究では「A みたいな B，どんなの？」というお題に対する回答の自動生成を目標とする．例えば，「ギャルみたいな裁判官，どんなの？」というお題である．本稿では，A を「属性語」，B を「基本語」と呼ぶ．

提案手法では最初に，属性語の関連語と，基本語の関連文に含まれる単語を入れ

*https://chat.openai.com
†https://bard.google.com

替えることで，回答を生成する．本稿では，単語を入れ替えを用いた回答の生成を，単語入れ替え法と呼ぶ．

次に，言語モデルで計算できるスコアをもとに面白い回答を抽出する．我々は，属性語の関連語と，基本語の関連文に含まれるを入れ替えることで，意外性をもち，お題に沿った回答を生成することができると考えた．また，言語モデルのスコアが回答の意外性を表していると考えた．

本研究では二つの実験を行う．一つ目の実験では，言語モデルのスコアによって面白い回答の抽出が可能かどうかを確認する．単語入れ替え法で生成した回答を言語モデルのスコアによってグループに分割し，各グループの回答の面白さに有意差があるかどうかを検定する．二つ目の実験では，提案手法が既存手法と比較し面白い回答を生成できるかどうかを確認する．提案手法と既存手法の回答を，パープレキシティで抽出したときの再現率適合率曲線を比較する．

一つ目の実験の結果，言語モデルのスコアによって，面白い回答を抽出できることがわかった．また，パープレキシティが小さい回答ほど，面白い傾向にあることがわかった．二つ目の実験の結果，提案手法で生成した回答は，既存手法で生成した回答よりも，面白いとはいえなかった．

本研究の貢献は以下のとおりである．

- 言語モデルのスコアによって，面白い回答が抽出できることを確認した．
- 言語モデルのスコアが比較的小さい回答が面白い傾向にあることを確認した．

本論文の構成は以下のとおりである．2 章では，本研究で用いた技術，手法について述べる．3 章では，関連研究について述べる．4 章では，大喜利の自動生成手法について述べる．5 章では，評価実験について述べる．6 章では，本研究のまとめ，今後の展望について述べる．

# 第 2 章　基本的事項

本稿で用いた技術，手法を本章で述べる．

## 2.1　大喜利

大喜利は，与えられたお題に対して面白い回答をする演芸の一つである．質問に回答する形式や，画像に対して回答する「写真で一言」形式，出題者とやりとりをして回答する形式などの様々なお題がある．本研究では，「A みたいな B，どんなの？」という質問に回答する形式のお題に対する回答の生成を目標とした．

## 2.2　Transformer

Transformer は，2017 年に Google が発表した，機械学習モデルである．Attention 機構で構成されている Encoder-Decoder モデルである．これまで主流であった RNN(Recurrent Neural Network; 再帰的ニューラルネットワーク) や CNN(convolutional Neural Network; 畳み込みニューラルネットワーク) と異なり，再帰構造や畳み込み処理を用いないことによって，処理の高速化，並列化，精度の向上が可能になった．

## 2.3　言語モデル

単語列の出現確率を確率分布によってモデル化したものである．現在では Transformer や RNN などを用いたニューラルネットワークで構成されるものが主流である．文章生成や感情分析などのタスクに利用することができる．

### 2.3.1　GPT

GPT(Generative Pretrained Transformer) は 2018 年に OpenAI が発表した，言語モデルである．GPT は主に Transformer の Decoder で構成されているニュー

ラルネットワークモデルである．2019 年に発表された後継の GPT-2 は，WebText で学習したモデルである．WebText は，ウェブ上の 40GB のテキストデータを収集し作成したコーパスである．GPT-2 は追加の学習なしでさまざまなタスクに対応できる Zero-shot が可能である．GPT-2 の Zero-shot は，七つの自然言語処理タスクで当時の SoTA(State of The Art) を達成した．

### 2.3.2 BERT

BERT(Bidirectional Encoder Representations from Transformers) は，2018 年に Google が発表した言語モデルである．BERT の隠れ層は Transformer の Encoder で構成されている．コーパスで事前学習を行ったモデルをファインチューニングすることによって，少量データと小さい計算費用で特定のタスクに特化したモデルを作成できるといわれている．BERT ではモデルが文脈理解できるように訓練するために，事前学習で MLM(Masked Language Model) と NSP(Next Sentence Prediction) の二つのタスクを行う．MLM は，入力文章の一部をランダムにマスクトークンに置換したマスク文章から，マスクトークンに置換されたトークンを予測するタスクである．MLM によって，単語に関連する文脈理解ができるようになるといわれている．NSP とは，二つの入力文が連続した文であるかどうかの二値分類を行うタスクである．NSP によって，文と文の間の文脈理解ができるようになるといわれている．

### 2.3.3 LLM

LLM(Large Language Model; 大規模言語モデル) は，言語モデルの一つであり，主に文章生成に利用されている．計算機の高性能化，Transformer による計算の並列化と高速化により，従来の言語モデルと比較しパラメータ数の多いモデルを大量のテキストデータで訓練することが可能になった．テキストデータでの文法や知識の学習に加え，人間の評価によるフィードバックを行い，モデルの調整をする．これによってモデルは，ユーザにとって有用で無害な文章を出力するようになる．近年では，OpenAI の ChatGPT，Google の Bard などの Web 上から LLM を利用

できるサービスがある．また，Meta の Llama のようにファイル形式で公開される学習済みのモデルも存在する．

## 2.4 パープレキシティ

パープレキシティ (*perplexity*) は，言語モデルを評価するときの指標の一つである．0 より大きい指標であり，0 に近いほど言語モデルの出現単語の予測性能が良いことを表している．ある文章 $W$ に対するパープレキシティは式 2.4.1 で計算する．

$$perplexity = \exp(-\frac{1}{M}\sum_{i=0}^{M}\log(p(w_i|W_{i-1}))) \tag{2.4.1}$$

$W$ は $M$ 個のトークンで構成されている．文章や単語をトークンという単位に分割することによって，言語モデルが入力を処理できるようにする．入力文章の $i$ 番目のトークンを $w_i$，入力文章の $i-1$ 番目までのトークン列を $W_{i-1}$ とする．$p(w_i|W_{i-1})$ は，言語モデルに $W_{i-1}$ を与えたときに，言語モデルが次に出現するトークンを $w_i$ であると予測する確率である．予測性能が高い言語モデルは，$W_{i-1}$ から $w_i$ を予測する能力が高いので，$p(w_i|W_{i-1})$ は大きくなり，パープレキシティは小さくなる．

## 2.5 形態素解析

形態素解析とは，文章を形態素に分割することである．形態素とは，意味合いを構成する最小の単位である．入力文を形態素に分けたのち，形態素に品詞付けをし，形態素の正規化を行う．本研究では，日本語に対応している形態素解析ツールの MeCab を使用した．他の日本語に対応している形態素解析ツールに Janome, Juman，Sudachi が挙げられる．

## 2.6 fastText

fastText は，2016 年に Facebook が発表した，単語分散表現の一つである．単語分散表現を用いることによって，コーパス内の単語を単語の種類数よりもはるかに小さい次元数のベクトルによって表現できる．単語ベクトルの類似度を用いた類似語の抽出や，単語ベクトルの演算を用いた king - man + woman = queen のような単語の意味合いを計算することができる．fastText では，skip-gram 法を用いて単語分散表現を獲得する．skip-gram 法とは，ニューラルネットワークを用いて，ある単語からその周辺にどのような単語が出現するかを学習することによって，単語分散表現を獲得する方法である．fastText では，単語の文字 n-gram ごとに分散表現を学習することによって，出現頻度が低い単語やコーパスに含まれない単語の分散表現を効率的に獲得することができる．ある単語 A を構成する n 個の文字を $a_0, a_1, \ldots, a_{n-1}$ とする．このとき，単語 A の文字 n-gram の集合 $n\text{-}gram(A)$ は $n\text{-}gram(A) = \{a_k a_{k+1} \ldots a_l | 0 \leq k \leq l < n\}$ である．例えば，「大喜利」の文字 n-gram 集合は{大，喜，利，大喜，喜利，大喜利}である．

## 2.7 評価指標

### 2.7.1 $t$ 検定

$t$ 検定とは，$t$ 分布を利用した検定である．本稿では，対応のない二群の平均の差に関する検定を行った．等分散性を仮定した, 母平均がそれぞれ $\mu_x$，$\mu_y$ である二つの母集団の，母平均の差を検定する．片側検定を行うため，帰無仮説 $H_0$:「$\mu_x - \mu_y \leq 0$」を設定した．また，有意水準 $\alpha = 0.05$ とした．二つの母集団から大きさがそれぞれ $m$,$n$ の標本を抽出する．それぞれの標本平均が $\bar{X}$,$\bar{Y}$ であり，それぞれの標本分散が ${S_x}^2$,${S_y}^2$ であるときの，検定統計量 $T$ は，式 2.7.1 で計算する．

$$T = \frac{\sqrt{m+n-2}\left(\bar{X} - \bar{Y}\right)}{\sqrt{\left(\frac{1}{m} + \frac{1}{n}\right)\left(m{S_x}^2 + n{S_y}^2\right)}} \tag{2.7.1}$$

このとき，$T$ は自由度 $m+n-2$ の t 分布に従う．$T$ が，有意水準 $\alpha$ としたときの棄却域内にあれば，帰無仮説 $H_0$ を棄却できる．

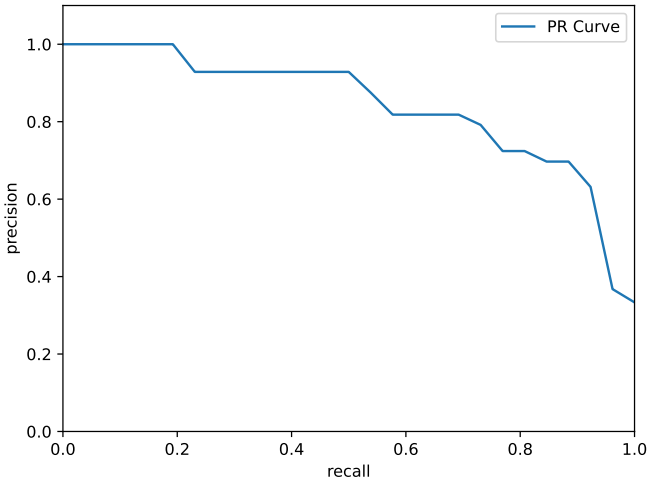


図 2.1　再現率適合率曲線

### 2.7.2　再現率適合率曲線

再現率適合率曲線 (PR Curve) とは，主に二値分類モデルの性能の指標の一つである．横軸が再現率 (recall)，縦軸が適合率 (precision) であり，再現率に対応する適合率を表す点を線で結んで描画する．再現率適合率曲線の例を図 2.1 に示す．モデルの予測ラベルと実際のラベルより，データは混同行列の TP,FN,FP,TN のいずれかに当てはまる．混同行列を表 2.1 に示す．再現率 $recall$ は式 2.7.2 で計算する．適合率 $precision$ は式 2.7.3 で計算する．

$$recall = \frac{TP}{TP + FN} \tag{2.7.2}$$

$$precision = \frac{TP}{TP + FP} \tag{2.7.3}$$

また，縦軸に補完適合率をとり，曲線が短調減少するものにして評価する場合もある．再現率が $r$ であるときの適合率が $P(r)$ とするとき，$r$ の補完適合率 $P_I(r)$

は式 2.7.4 で計算する．

$$P_I(r) = \max_{r' \geq r} P(r') \tag{2.7.4}$$

適合率と再現率はトレードオフの関係である．モデルを利用する場面によって，適合率と再現率のどちらを重視するかは異なる．

表 2.1　混同行列

| | | 実際のデータ | |
|---|---|---|---|
| | | Positive | Negative |
| モデルの予測 | Positive | TP | FP |
| | Negative | FN | TN |

# 第 3 章　関連研究

大喜利を含めた，お笑いやユーモアを持つ文章の自動生成を行う研究がいくつか存在する．大喜利については，Web ページ上のお題と回答のデータをもとに機械学習によって回答の生成を目標とする研究が複数ある．

内海ら [3] は，会話コーパスを学習した Seq2Seq モデルに，大喜利コーパスを転移学習させることによって大喜利の回答の自動生成を行なった．会話コーパスは Twitter の投稿とリプライから作成している．大喜利コーパスは大喜利のお題と回答を投稿できるサービスを提供している Web ページから取得したデータから作成している．結果，大喜利らしい面白い回答が生成されることがあった．生成した回答の面白さの定量的評価は行なっていない．

山富ら [4] は，お題となる画像と文章の回答のデータから，いわゆる「写真で一言」形式のお題に対する回答の生成を行なっている．学習に使用するデータは大喜利投稿サービスの Web ページから取得している．取得したデータを Pix2Seq を使用して学習している．また，画像の特徴量抽出に使用する事前学習済み画像認識モデルを大喜利のお題画像に対応させるために，AutoEncoder を用いている．結果，大喜利の回答らしい文章が生成できることが確認できている．生成した回答の面白さの定量的評価は行なっていない．

また，中村ら [2] は，どのような要素が大喜利の回答の面白さに寄与するかを調べている．Web ページから収集した大喜利の回答に面白さと六つの要素に対してクラウドソーシングを用いて評価付けを行った．六つの要素の評価の組み合わせを回帰モデルで面白さの評価を予測したところ，「お題との関係性」，「わかりやすさ」，「新しさ」の三つの要素が回答の面白さに寄与していることがわかった．また，三つの要素と関係がある機械的特徴量である「お題と回答の文コサイン類似度」，「平均単語難易度」，「お題と回答のトピックモデル類似度」を用いた回帰モデルで面白さの評価の予測をした．その結果，面白さに寄与するとわかった三つの要素より予測精度が悪化した．結果より，どの機械的特徴も面白さを表しているとはいえなかった．

大喜利以外のユーモアをもつ文章の生成を目標とする研究もある．呉ら [5][6] は，聞き間違えボケによるユーモア生成を行なっている．ユーザが入力した文の単語一

つを別の単語に置き換えることで，聞き間違えボケを生成している．置き換え元の単語は，入力文のトピックに最も意味が近い単語を，概念距離によって決定している．置き換え先の単語は，入力文や置き換え元の単語との概念距離，置き換え元の単語との後の近さ，コーパス内での出現頻度によって決定している．概念距離の計算には Word2Vec を用いている．生成したボケの面白さの定量的評価を行なったところ，概念距離を用いたトピックの考慮が面白さにある程度寄与することが示された．

# 第 4 章　提案手法

本研究では,「A みたいな B，どんなの？」という大喜利のお題に対する回答の生成を行う．我々は，単語入れ替え法と，言語モデルのスコアに着目した．提案手法では，以下の手順で回答を生成する．

Step 1 fastText を用いて属性語の関連語を抽出する．
Step 2 ChatGPT を用いて基本語の関連文を生成する．
Step 3 基本語の関連文に含まれる単語を属性語の関連語と入れ替えることによって，回答を生成する．
Step 4 言語モデルを用いて，生成した回答に対してスコアを計算する．
Step 5 計算したスコアをもとに面白い回答を抽出する．

提案手法の概要図を図 4.1 に示す．

我々は，大喜利の回答の生成にあたって，以下の二つの条件を満たしている文が回答として適していると考えた．

条件 1 面白みを感じる文である．
条件 2 文がお題に沿った内容である．

我々は，属性語の関連語を基本語の関連文に含まれる単語と入れ替えることによって，お題「A みたいな B, どんなの？」に沿った回答を生成できると考えた．理由は以下の二つである．一つ目の理由は，単語を入れ替えた箇所は前後の文脈から内容を予測することが難しくなり，意外性が生まれるためである．意外性を持つ文とはその内容について予測することはできないが，納得することはできる文である．我々は，意外性をもつ文に面白みを感じると考えた．面白い回答には聞き手を予想を裏切り，納得もさせるという絶妙なバランス感が必要である [2] といえるためである．以上より，我々は，属性語の関連語を基本語の関連文に含まれる単語と入れ替えることによって，文は条件 1 を満たすと考えた．

二つ目の理由は，単語入れ替え法によって文が属性語と基本語の両方の要素を持つことになり，文は条件 2 を満たすと考えたためである．よって,「A みたいな B, どんなの？」というお題に適した回答を生成できると考えた．

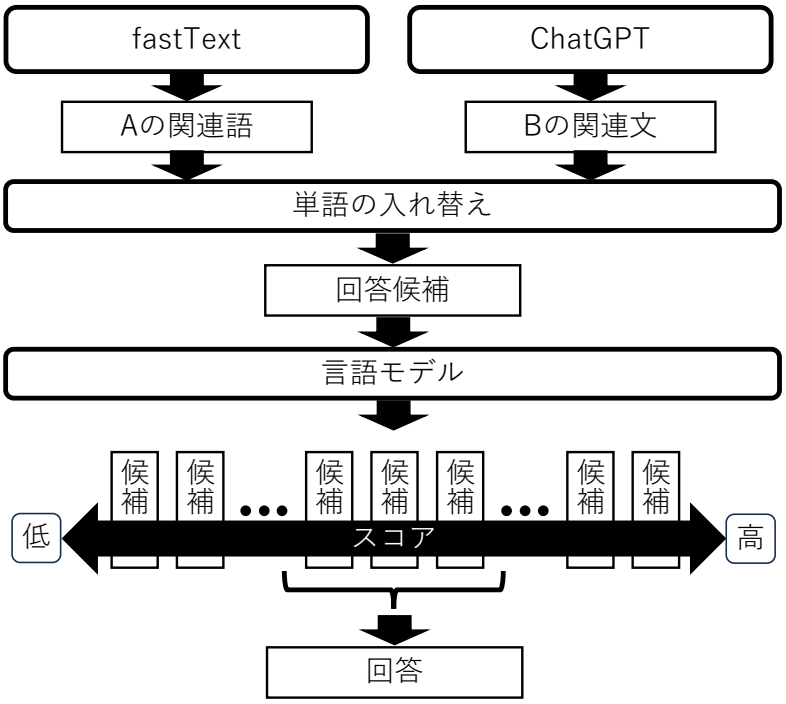


図 4.1　提案手法の概要図

単語入れ替え法で大喜利の回答を生成するとき，生成された回答の中には面白くない回答も含まれている．そのため，面白い回答のみを抽出する必要がある．我々は，言語モデルのスコアによって，面白い回答が抽出できると考えた．

我々は，意外性のある回答が面白い回答であると考えた．意外性をもつ文とは，その内容の予測はできないが納得はできる文である．面白い回答には聞き手の予想を裏切り，納得もさせるという絶妙なバランス感が必要である [2] といえるためである．

我々は，言語モデルのスコアが回答の意外性を表す指標になると考えた．言語モデルのスコアは，文の流暢さを表しているといえる．文が流暢であるほど，私たちが普段から使っている文に近いといえる．我々は，流暢である文章は内容の予測がしやすく，流暢ではない文章は内容に納得感が得られないと考えた．以上から，言語モデルのスコアが回答の意外性を表す指標になると考えたため，言語モデルのスコアをもとに面白い回答の抽出ができると考えた．

## 4.1　単語入れ替え法による回答の生成

単語入れ替え法によって大喜利の回答を生成する手順を説明する．最初に，属性語に関連する単語を 10 個抽出する．属性語の単語ベクトルとコサイン類似度が高い単語ベクトルを持つ単語上位 10 個を抽出する．類似度が高い単語ベクトルを持つ単語同士は，関連があると考えたためである．fastText[7] の学習によって獲得した単語ベクトルを，関連語の抽出に用いた．

抽出した単語を $w_{ai}(i = 1, \ldots, 10)$ とする．

次に，ChatGPT を用いて，基本語に関連する文を 10 個生成する．生成した基本語に関連する文を $s_j(j = 1, \ldots, 10)$ とする．ChatGPT に入力したプロンプトは以下のとおりである．

> 以下の条件を満たす文を 10 個生成してください。
> 1. 内容は「基本語」の具体的な行動である
> 2.「基本語」に関係し、「基本語」以外には関係しない単語を 3 個以上含む。また、「B」に関係する単語はどれであるかを必ず示してください
> 3. 文の長さは 7 単語以内にする
> 4. 文は「基本語」から始まる

以上のプロンプトを ChatGPT に入力して生成した $s_j$ に，形態素解析を行う．形態素解析器には MeCab[8] を用いた．$s_j$ に含まれる形態素の総数を $N_j$，形態素それぞれを $w_{bjk}(k = 1, \ldots, N_j)$ とする．

$w_{ai}$ と $w_{bjk}$ の組み合わせの総当たりを列挙する．文法的に不適な文が生成されるのを防ぐため，$w_{ai}$ と $w_{bjk}$ が共に名詞である場合のみに制限した．また，「する」とその活用形に接続している，サ変接続である $w_{bjk}$ とサ変接続ではない $w_{ai}$ の組み合わせは，文法的に不適であるため組み合わせから除外する．列挙した組み合わせの $s_j$ 中の $w_{bjk}$ と $w_{ai}$ を入れ替える．単語を入れ替えて得られた文を，お題「A みたいな B，どんなの？」に対する回答とする．

## 4.2 言語モデルのスコアに基づいた面白い回答の抽出

言語モデルのスコアに基づいた，面白い回答の抽出の手順を説明する．最初に，単語入れ替え法によって生成した回答に対するスコアを計算する．計算したスコアに基づいて，回答をソートする．面白い回答が分布すると考えられる範囲から，回答を抽出する．

本稿では，回答のスコアの計算に，パープレキシティと MLM の損失関数の二通りの方法を用いて，面白い回答の抽出を行う．我々は，パープレキシティと MLM の損失関数は，どちらも回答の意外性を表す指標と考えた．どちらのスコアも，文の流暢さを表すと考えたためである．また, パープレキシティの計算には GPT-2[9] の，MLM の損失関数の計算には BERT[10] の事前学習済みモデルを用いる．

### 4.2.1 パープレキシティ

GPT-2 を用いる場合，文に対するパープレキシティを計算し，回答のスコアとする．パープレキシティは 0 より大きい指標であり，0 に近いほど文は流暢であるといえる．入力文のトークン列 $W$ に対するパープレキシティ $perplexity$ の計算式を式 4.2.1 に示す．

$$perplexity = \exp\left(-\frac{1}{M}\sum_{i=0}^{M}\log\left(p_{GPT-2}\left(w_i|W_{i-1}\right)\right)\right) \tag{4.2.1}$$

$W$ にはトークンが $M$ 個含まれている．入力文の $i$ 番目のトークンを $w_i$，入力文の $i-1$ 番目までのトークン列を $W_{i-1}$ とする．$p_{GPT-2}(w_i|W_{i-1})$ は，モデルに $W_{i-1}$ を与えたときに，モデルが次に出現するトークンを $w_i$ であると予測する確率である．内容が予測しやすい文では全ての $i$ に対して $p_{GPT-2}(w_i|W_{i-1})$ が大きくなるため，$perplexity$ が小さくなる．内容に納得が得られない文は，$W_{i-1}$ から予測しづらい $w_i$ が存在するため，$p_{GPT-2}(w_i|W_{i-1})$ が小さくなる $i$ が存在する．そのため，内容に納得が得られない文は $perplexity$ が大きくなる．予測のしづらいトークンの有無で変動する $perplexity$ は文の流暢さを表しているといえる．そのため，我々はパープレキシティで回答の意外性を考慮できると考えた．

### 4.2.2 MLM の損失関数

BERT を用いる場合，置き換えた単語を [MASK] にしたときの MLM(Masked Language Model) の損失関数を計算し，回答のスコアとする．MLM の損失関数の値は 0 以上であり，0 に近いほど，文は流暢であるといえる．入力文のトークン列 $W$ に対する MLM の損失関数 $MLMloss$ の計算式を式 4.2.2 に示す．

$$MLMloss = -\frac{1}{N}\sum_{i=1}^{N}\log(p_{BERT}(mask_i|W)) \tag{4.2.2}$$

$W$ の中には [MASK] に置換したトークンが $N$ 個含まれている．$i$ 番目の [MASK] に置換されたトークンを $mask_i$ とする．$p_{BERT}(mask_i|W)$ は，モデルが $W$ をもとに入力文の $i$ 番目の [MASK] が $mask_i$ であると予測する確率である．内容が予測しやすい文では [MASK] されたトークンを前後の文脈から予測しやすいため，全ての $i$ に対して $p_{BERT}(mask_i|W)$ が大きくなる．そのため $MLMloss$ は小さくなる．内容に納得感が得えられない文では，前後の文脈から予測が困難なトークンが存在すると考えられる．そのため，予測が困難なトークンをマスクトークンに置き換えたとき，$p_{BERT}(mask_i|W)$ が小さくなり，$MLMloss$ は大きくなる．よって予測のしづらいトークンの有無で変動する $MLMloss$ は文の流暢さを表しているといえる．以上から，我々は MLM の損失関数で回答の意外性を考慮できると考えた．

# 第 5 章　評価実験

本稿では二つの実験を行う．一つ目の実験では，言語モデルによって面白い回答の抽出が可能かどうかを確認する．二つ目の実験では，単語入れ替え法で生成した回答と既存手法で生成した回答の面白さに差があるかどうかを確認する．本稿では，既存手法に LLM を使用する．LLM には ChatGPT(GPT-3.5) を選択した．

## 5.1　実験 1: 言語モデルのスコアによる比較

### 5.1.1　実験手順

実験 1 の手順は以下の通りである．

Step 1「A みたいな B，どんなの？」の形式のお題を用意する．

Step 2 Step1 で用意したお題の回答を 4.1 節で示した単語入れ替え法によって生成する．

Step 3 Step2 で生成した回答の面白さの評価をアンケートによって行う．

Step 4 Step2 で生成した回答のスコアを言語モデルで計算する．

Step 5 Step4 で計算したスコアをもとに回答を昇順でソートし．上位から 10% ごとのグループに分割する．分割したグループを $G_1, G_2, \ldots G_{10}$ とする．

Step 6 $G_1, G_2, \ldots G_{10}$ の回答がそのグループに含まれない回答と比較して有意に面白いかどうかを検定する．

お題は，YouTube 上の大喜利動画を投稿しているチャンネル*の動画で用いられているお題を参考に，我々が属性語と基本語をそれぞれ三つずつ決定し，その組み合わせの九つを用意した．実験に用いたお題を表 5.1 に示す．Step 2 で用いる fastText にはインターネット上で公開されているモデル†を使用した．Step 4 では二種類の方法を用いて回答ごとにスコアを計算する．用いた方法は，パープレキシ

*https://www.youtube.com/@oogiruhitotachi

†https://drive.google.com/file/d/0ByFQ96A4DgSPUm9wVWRLdm5qbmc/view?usp=sharing&resourcekey=0-of5Ks1fuoKNh1pEYE8uSFQ

ティと MLM の損失関数である．

パープレキシティの計算に用いる GPT-2 には，rinna 株式会社が公開している事前学習済みモデル[‡]を使用した．MLM の損失関数の計算に用いる BERT には，東北大学の乾・鈴木研究室が公開している事前学習済みモデル[§]を使用した．

協力者にアンケートを行い，回答の面白さの評価を行った．協力者に回答の面白さを，1. 全く面白くない，2. あまり面白くない，3. どちらともいえない，4. まあ面白い，5. とても面白いの 5 段階で評価してもらった．回答を 4，5 のいずれかと評価した人数を，回答の面白さとした．協力者は，岐阜大学の鈴木研究室の学生 13 名である．Step6 では，言語モデルのスコアによって面白い回答の抽出が可能であるかどうかを確認する．グループの回答のが有意に面白いかどうかを確認する．あるグループに含まれる回答の面白さの平均値が，そのグループに含まれない回答の面白さの平均値より有意に大きいか否かを対応のない 2 標本 $t$ 検定によって確認する．今回は比較する二群に等分散性を仮定した．有意水準を 0.05 に設定し，片側検定を行った．複数のお題で，回答が有意に面白いグループが存在する場合，言語モデルのスコアによって面白い回答の抽出をすることが可能であるといえる．

[‡]https://huggingface.co/rinna/japanese-gpt-1b
[§]https://github.com/cl-tohoku/bert-japanese

表 5.1　実験に用いたお題

| お題番号 | お題 |
|---|---|
| 1 | オタクみたいなラッパー、どんなの？ |
| 2 | オタクみたいな裁判官、どんなの？ |
| 3 | オタクみたいな詐欺師、どんなの？ |
| 4 | ギャルみたいなラッパー、どんなの？ |
| 5 | ギャルみたいな裁判官、どんなの？ |
| 6 | ギャルみたいな詐欺師、どんなの？ |
| 7 | 猫みたいなラッパー、どんなの？ |
| 8 | 猫みたいな裁判官、どんなの？ |
| 9 | 猫みたいな詐欺師、どんなの？ |

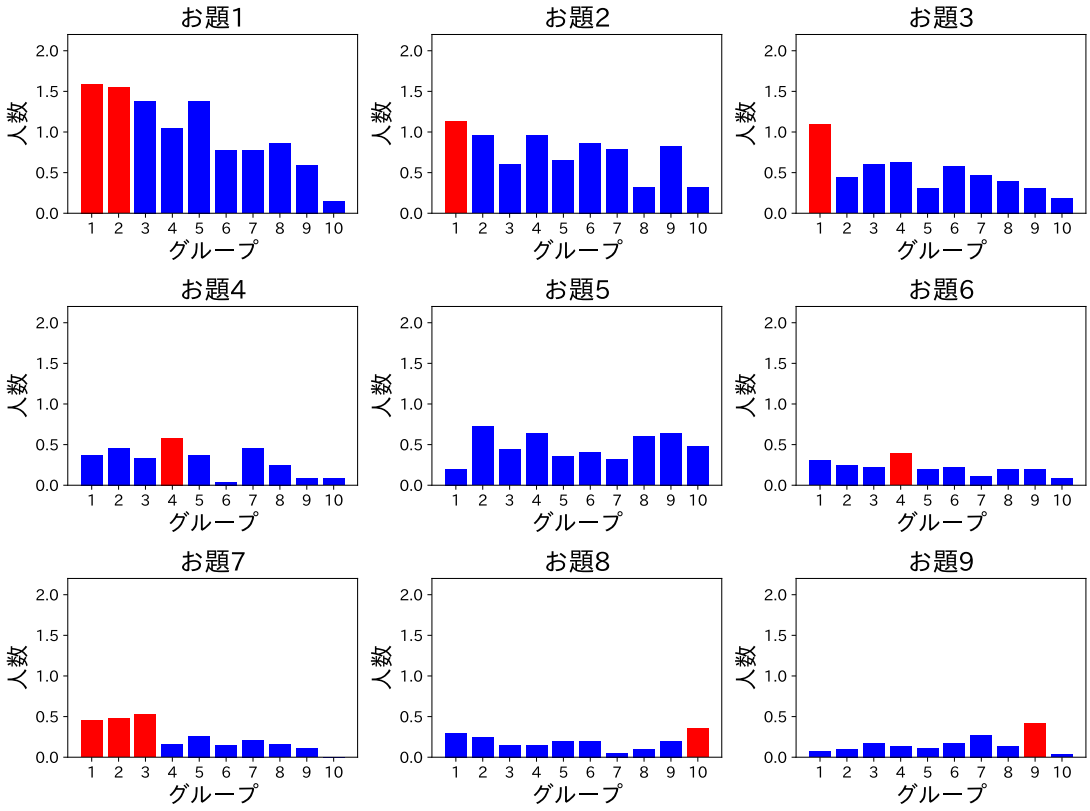


図 5.1　実験 1 の結果: パープレキシティでグループ化したときの面白さの平均

### 5.1.2　結果・考察

実験の結果を図 5.1 と図 5.2 に示す．図 5.1 で示す実験ではパープレキシティを，図 5.2 で示す実験では MLM の損失関数をもとに回答を 10 個のグループに分割した．横軸は各グループであり，スコアが昇順に左から並んでいる．縦軸は，各グループの回答の面白さの算術平均である．棒グラフでは，回答が有意に面白かったグループを赤色で表示している．

パープレキシティによるグループ化では 9 個中 8 個のお題で，面白さが他と比較し有意に大きいグループが確認できた．お題 1,2,3,7 では，パープレキシティが小さいグループが，お題 4,6 では，パープレキシティが中程度のグループが，お題 8,9 では，パープレキシティが大きいグループが有意に面白いことが確認できた．

MLM の損失関数によるグループ化では，9 個中 7 個のお題で，面白さが他と比

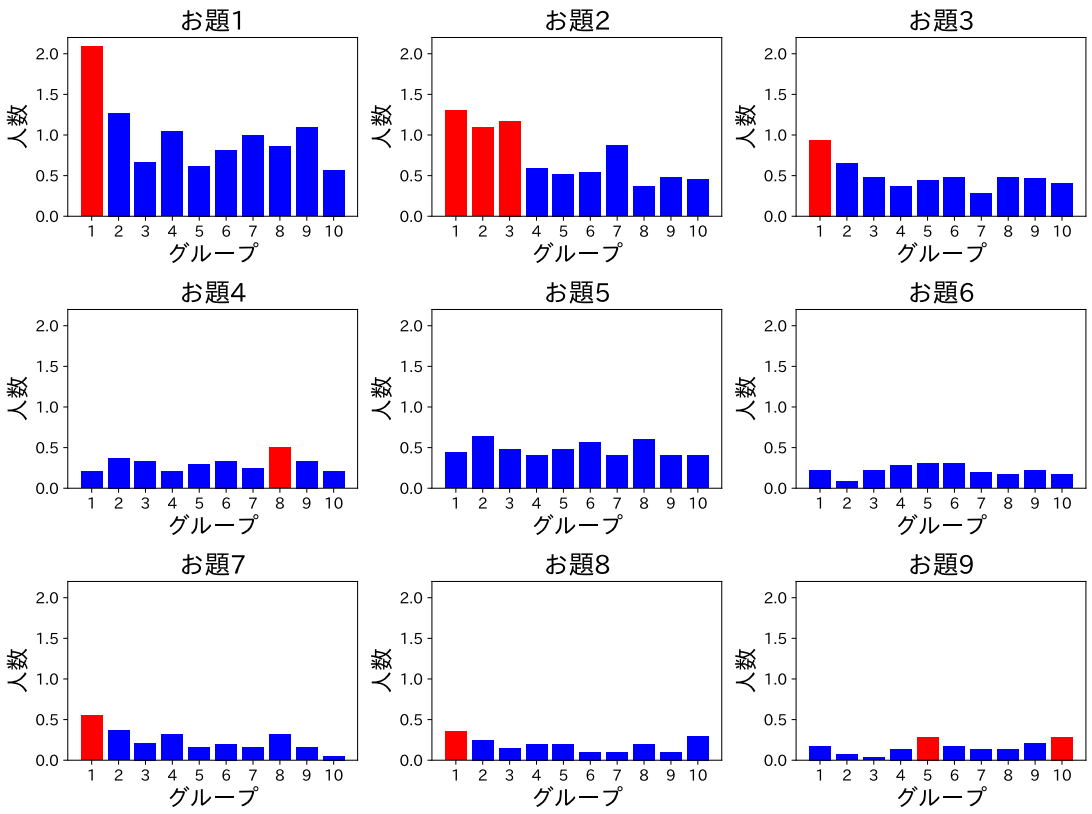


図 5.2　実験 1 の結果: MLM の損失関数でグループ化したときの面白さの平均

較し有意に大きいグループが確認できた．お題 1,2,3,7,8 では，MLM の損失関数の値が小さいグループが，お題 4 では，MLM の損失関数が比較的大きいグループが有意に面白いことが確認できた．お題 9 では，MLM の損失関数が上位 40% から 50% のグループと 90% から 100% のグループが有意に面白いことが確認できた．

お題 1,2,3,7 ではパープレキシティと MLM の損失関数で，回答が有意に面白いグループの傾向が同じであった．お題 5 では，パープレキシティと MLM の損失関数の両方で，回答が有意に面白いグループを確認できなかった．お題 4,6,9 では，回答が有意に面白いグループの傾向がパープレキシティと MLM の損失関数で異なっている．お題 4 では，パープレキシティで分割したときの回答が面白いグループは上位 40% から 50% であるが，MLM の損失関数で分割したときは 80% から 90% のグループとなっている．お題 6 では，パープレキシティで分割したときは

回答が有意に面白いグループが確認できたが，MLM の損失関数で分割したときでは確認できなくなっている．お題 9 では，MLM の損失関数で分割したとき，上位 40% から 50% のグループの回答が有意に面白いが，パープレキシティで分割したときは有意に面白くはなかった．

パープレキシティでは 9 個中 8 個のお題で，MLM の損失関数では 9 個中 7 個のお題で，回答が有意に面白いグループが存在した．よって，パープレキシティと MLM の損失関数をもとに，面白い回答の抽出が可能であるといえる．また，パープレキシティでは 9 個中 4 個のお題で，MLM の損失関数では 9 個中 5 個のお題で，言語モデルのスコアが比較的小さいグループの回答が有意に面白かった．そのため，パープレキシティと MLM の損失関数が小さい回答ほど，面白い傾向にあるといえる．

言語モデルのスコアが小さいほど，文章は流暢であるといえる．流暢さが高い文章は内容の予測がしやすく，流暢さが低い文章は納得感が得られない．そのため，言語モデルのスコアが小さい回答は，内容がわかりやすい回答であるといえる．結果より，言語モデルのスコアが小さい回答が面白いと評価されやすい傾向があることがわかった．よって，単語入れ替え法で生成した回答は，わかりやすい回答が面白いと評価されやすい傾向にあったといえる．

## 5.2 実験 2: 既存手法との比較

### 5.2.1 実験手順

実験 2 の手順は以下の通りである．

Step 1 5.1 節で生成したお題のうち，パープレキティが最も小さい回答 10 個を提案手法の回答とする．提案手法の回答のグループを $G_m$ とする．

Step 2 ChatGPT にプロンプトを入力し，お題に対する回答を 10 個生成する．ChatGPT によって生成した回答のグループを $G_g$ とする．

Step 3 $G_g$ の回答に対して，パープレキシティを計算する．

Step 4 $G_g$ の回答の面白さの評価をアンケートによって行う．

Step 5 $G_m$，$G_g$ それぞれを，パープレキシティの昇順でソートする．$G_m$，$G_g$ か

ら，パープレキシティの小さい順に回答を抽出し，それぞれを提案手法，既存手法の出力とする．抽出する回答の件数を 1 から 10 に変化させたときの，出力に含まれている面白い回答の件数をもとに，適合率と再現率を計算する．計算した適合率と再現率をもとに再現率適合率曲線をプロットする．

既存手法に，ChatGPT(GPT-3.5) を選択した．Step2 では，以下のプロンプトを使用して回答の生成を行なった．

> 以下の文章は大喜利のお題です．回答を 10 個生成してください．回答は一文にしてください．
> お題:「A みたいな B、どんなの？」

提案手法で生成する回答の長さは一文である．提案手法と回答の長さを合わせるために，プロンプトに「回答は一文にしてください.」と記した．Step5 における，回答が面白きかそうでないかは，アンケートで回答の面白さが 4 または 5 であるとした人が 1 人以上いるか否かで決定した．我々は，誰か一人でも面白いと感じたなら，それは面白い回答であると考えたため，このような基準にした．

この実験では，提案手法と既存手法の再現率適合率曲線を比較することで，それぞれの手法の面白さの差を確認する．再現率に対する適合率の減少量が小さい手法は，パープレキシティに基づいて回答を抽出した際に，面白い回答が占める割合が大きい可能性が高いといえるためである．

### 5.2.2 結果・考察

実験の結果を図 5.3 に示す．横軸は再現率，縦軸は適合率である．青色の曲線が提案手法の，赤色の曲線が既存手法の再現率適合率曲線である．

全てのお題にて，再現率に対する適合率の減少量は，提案手法が既存手法よりも大きかった．よって，パープレキシティをもとに回答の抽出を行う場合，抽出した回答の中で面白いものが占める割合について，提案手法が既存手法よりも低いといえる．そのため，提案手法で生成した回答は，既存手法で生成した回答よりも面白いとはいえないという結果となった．

我々は, fastText による単語ベクトルの類似度では, お題に十分に沿った単語を抽

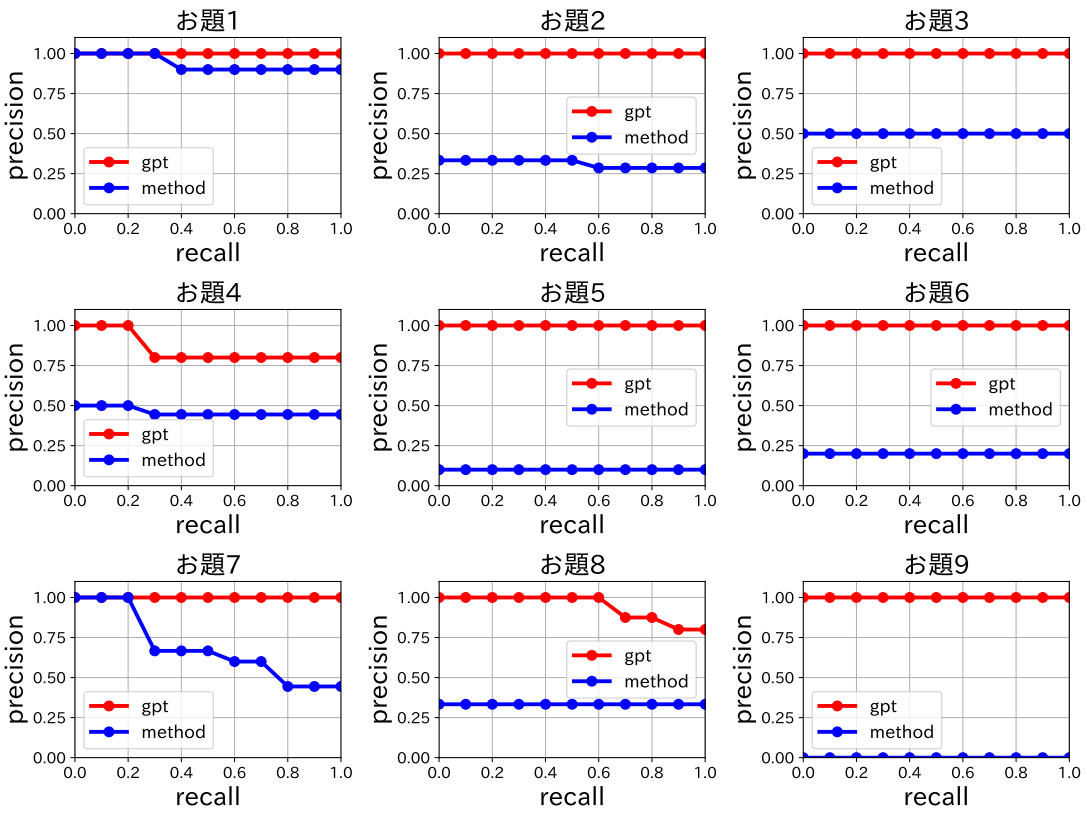


図 5.3　実験 2 の結果: 提案手法と既存手法の再現率適合率曲線

出できないことが，このような結果になった原因の一つであると考えた．fastText ではお題に十分に沿った単語を抽出できないと考えた理由は，二つある．一つ目の理由は，fastText では，複数の形態素からなるフレーズを抽出できないためであある．以下に示す例 1，例 2，例 3 は，お題 1「オタクみたいなラッパー、どんなの？」において，提案手法と既存手法で生成した回答の一部である．例 1 は，提案手法で生成した回答である．例 2 と例 3 は，既存手法で生成した回答である．

例 1 ステージでロリコンを炸裂させた。
例 2 ラップの合間に「これ、知ってる？」と何度もアニメの名言を挟むラッパー。
例 3 ブレイクダンスの最中に急にゲームの攻略法を語りだすラッパー。

回答に対する面白さの評価の平均値は，例 1 は 3.00 であり，例 2 は 3.54，例 3 は

3.23 である．提案手法で生成した回答よりも，既存手法で生成した回答が面白さの評価が高かった．例 1 は，提案手法で生成した回答である．「ラッパー」の関連文中の単語と「オタク」の関連語と入れ替えることで，お題に沿った回答を生成した．例 2 と例 3 は，いずれも「ラップ」「ブレイクダンス」という「ラッパー」要素を含む行動を表す文がベースとなっている．また，「アニメの名言」「ゲームの攻略法」という「オタク」要素を含むフレーズが回答に含まれている．よって，「ラッパー」の関連文と「オタク」の関連語を組み合わせることで，お題に沿った回答を生成したといえる．このことより，例 1，例 2，例 3 は類似した方法でお題に沿った回答を生成しているといえる．以上より，提案手法と既存手法で，類似した方法でお題に沿った回答を生成していることがわかる．ところが，提案手法と既存手法で，生成した回答の面白さに差がある．我々は，提案手法では面白さ寄与する単語を回答に入れ込むことが難しいため，提案手法と既存手法の面白さに差が生まれたと考えた．既存手法の回答に含まれるフレーズ「アニメの名言」「ゲームの攻略法」はいずれも二つ以上の形態素で構成されている．我々は，この二つのフレーズは「オタク」の要素を十分に含んだフレーズであると，広く理解されると考える．それと比較し，提案手法では一つの形態素でしか，「オタク」の要素を回答に入れ込むことができない．

fastText の学習に使用するデータは，形態素解析器で分かち書きをしたコーパスである．fastText で獲得した単語ベクトルの類似度を用いて抽出した単語は，一つの形態素からなるためである．提案手法では「オタク」の関連語として，「アニメ」や「ゲーム」のような単語が抽出可能である．しかし，「名言」や「攻略法」と合わせて抽出することはできない．提案手法では複数の形態素からなる有効なフレーズを入れ込んだ回答が生成できないため，既存手法より面白くなかったと考えた．

我々は，この問題の解決するために，言語モデルの使用した複数の形態素からなるフレーズが有効であると考えた．fastText で抽出した単語を，GPT をはじめとする文章生成が可能な言語モデルに入力する．入力した単語に続く文章が出力されるので，出力の一部を抽出することでお題の要素を含むフレーズが得られると考えられる．

我々が，fastText ではお題に十分に沿った単語が抽出できないと考えた二つ目の理由は，fastText では単語どうしの上位，下位関係を考慮できないためである．以

下に示す例 4 と例 5 は，お題 1「ギャルみたいな裁判官、どんなの？」において，提案手法が生成した，パープレキシティでソートしたときの上位 10% の回答の一部である．

例 4 ストリートファッションの解釈を明確に示す。
例 5 ファッションの解釈を明確に示す。

例 4 の回答は面白いと，例 4 の回答は面白くないと判断された．この二つはどちらも，「法律の解釈を明確に示す。」という共通の文から生成された回答である．もとにした文が同じであるため，「ストリートファッション」が「ファション」よりも面白さに寄与したといえる．「ストリートファッション」は「ファッション」の下位語であり，「ストリートファッション」は「ファッション」の中の特定の範囲を意味する単語である．「ファッション」から「ストリートファッション」に，意味する範囲を狭くすることで，より「ギャル」と関連する単語となると考えられる．

fastText で獲得した単語ベクトルは，単語間の上位，下位関係を考慮することができない．単語ベクトルの類似度で関連語を抽出したとき，「ギャル」と類似する「ファッション」と「ストリートファッション」が得られる．しかし，「ファッション」は「ストリートファッション」と比較して面白さに寄与しづらい単語であるため，「ファッション」を用いた面白くない回答が生成されてしまう．

我々は，この問題を解決するために，シソーラスが有効であると考えた．シソーラスでは「ファッション」という単語から，下位語である「ストリートファッション」が抽出できるといえる．fastText で我々は，抽出した関連語の中に上位，下位関係がある二単語が含まれる場合，上位の単語を用いて回答の生成をしないことで，面白い回答が全体を占める割合が大きくなると考えた．

# 第 6 章　おわりに

我々は本研究で，大喜利の回答の自動生成を試みた．本研究では，お題を「A みたいな B，どんなの？」という形式のお題に絞った．我々は，単語の入れ替えと言語モデルによる大喜利の回答の自動生成手法を提案した．提案手法は，大きく分けて二つの手順で回答の自動生成を行う．単語入れ替え法による大喜利の回答生成と，言語モデルのスコアに基づいた面白い回答の抽出である．提案手法の有効性を確認するために，二つの実験を行った．一つ目の実験では，言語モデルのスコアで面白い回答が抽出可能かどうかを確認した．単語入れ替え法によって生成した回答を 10 個のグループに分割し，有意に面白いグループが存在するかどうかを調べる．これを 9 個のお題それぞれに対して行った．実験の結果，ほとんどのお題で，有意に面白いグループが存在した．結果から，言語モデルのスコアによって，面白いお題の抽出ができるといえた．また，言語モデルのスコアが小さいグループが，回答が面白い傾向があることがわかった．二つ目の実験では，提案手法で生成した回答と既存手法で生成した回答の面白さの比較を行った，提案手法と既存手法の回答をパープレキシティをもとに抽出したときの，再現率適合率曲線を比較する．これを 9 個のお題それぞれに対して行った．実験の結果，全てのお題で，提案手法の再現率適合率曲線は，既存手法の再現率適合率曲線を下回った．よって，結果から提案手法で生成した回答の面白さは，既存手法で生成した回答より面白いとはいえなかった．提案手法と既存手法の回答を比較したところ，提案手法には二つの問題点が存在することがわかった．また，これらの問題点の解決策を考察した．

今後の展望を述べる．最初に提案手法に存在する問題点の解決を考えている．本稿の実験結果より考察した解決策の有効性を確認する．また，「A みたいな B，どんなの？」以外のお題に対する回答の生成の提案を考えている．

# 謝辞

本研究を進めるにあたって，指導教員である鈴木優准教授には大変お世話になりました．はじめての研究とはじめての論文執筆に慣れないことが多い中，たくさんのことをご教授いただきました．大喜利という少し変わったテーマについて研究することができたのは，指導教員が鈴木優准教授だったからこそだと思います．ありがとうございました．大学院での二年間もどうぞよろしくお願い申し上げます．

事務補佐員の井尾さんには，様々な事務作業や各種申請に関して，お世話になりました．何かと提出書類が多かった一年間を円滑な学生生活とすることができたのは井尾さんのおかげです．

鈴木研究室の先輩方には，研究のアドバイスや論文の添削をしていただきました．また，研究以外でも，ご飯や飲み，旅行や遊びに連れて行っていただいたり，駅まで送っていただいたり，一緒にゲームしたり，コーヒーおじさんに付き合っていただいたり，月刊の漫画雑誌を一緒に買ったり，冗談を言い合ったりしていただきました．全員素敵な人たちで，とても親切にしていただきました．鈴木研究室を第一志望にした大きな理由の一つは，先輩方が仲良しで，明るい人たちに感じたからです．皆さんと，1 年半を過ごすことができて本当によかったです．

同じ研究室の同期で友人でもある，川上くん，城所くんに感謝申し上げます．二人がいたから，私はメンタルがギリギリになりつつも，今日まで生きて来ることができました．また一緒にポテトを食べよう．

12 月に鈴木研究室に配属になった B3 の皆さん，これから長い間，何卒よろしくお願いします．

カウンセラーの先生には毎週のカウンセリングでお世話になりました．気分が落ち込んでどうしようもなかったとき，少し冷静になることができました．

大学の友人とは，一年生の頃から多くの時間を過ごしました．一緒に行った旅行は最高に楽しかったです．授業や研究の愚痴をはじめとするくだらない話や，ためになる話ができる友人は，私の大学生活には不可欠でした．

今までずっと支えてくれた家族に感謝します．おいしい食事と，帰る家があることはとても幸せなことだと思います．仲の良い両親と文武両道に長ける弟は，私の自慢です．また，祖母や叔母をはじめとする親戚の皆さんに感謝します．

一緒に散歩したり，コーヒーを飲んだり，お酒を飲みに行ったり，ラーメンを食べたり，風呂に行ったり，モノポリーをしたりしてくれた地元の友人に感謝申し上げます．

アルバイト先の某家電量販店と，そこで一緒に働いてくれている皆さんに感謝申し上げます．

日本学生支援機構，岐阜大学消費生活協同組合，カルディコーヒーファーム，名鉄，JR 東海，岐阜バスに感謝申し上げます．

聴くと前向きになれる曲を作ってくれた Perfume，あいみょん，スピッツ，Suchmos，Oasis，aiko，The Beatles，山下達郎，ano，相対性理論，大好きな漫画の作者である葦原大介先生，荒木飛呂彦先生，諫山創先生，藤本タツキ先生，芥見下々先生，ナガノ先生，心の支えになるラジオのパーソナリティであるマヂカルラブリー，真空ジェシカ，素敵な映画を世に送り出し続けているクリストファー・ノーラン監督，映画「ジョン・ウィック」で主人公を演じ，超クールなアクションと魅力的な演技を見せてくれたキアヌ・リーブス，私のモノマネのレパートリーである福山雅治，桑田佳祐，コロコロチキチキペッパーズのナダル，坂上忍，関暁夫，オードリー若林，仲間由紀恵に感謝申し上げます．

多くの方のお力添えにより，私は本研究を進めることができました．ChatGPT の提供元である OpenAI，使用した学習済み GPT-2 モデルの提供元である rinna 株式会社，学習済み BERT モデルをお借りしました東北大学の乾・鈴木研究室，fastText の単語分散表現を公開していただいた Hironsan さん，実験に用いるお題の参考にした「大喜る人たち」に感謝申し上げます．また，お忙しい中，大喜利の回答の評価をしていただきました鈴木研究室の皆さん，本当にありがとうございました．

多くの人たちのおかげで，本稿を書き上げることができました．お前たち，最高だぜぇ〜！！イエーイ！！

# 参考文献


[1] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. A survey of large language models, 2023.

[2] 中川裕貴, 村脇有吾, 河原大輔, 黒橋禎夫. クラウドソーシングによる大喜利の面白さの構成要素の分析. 言語処理学会第 25 回年次大会 (NLP2019), 2019. B3-2.

[3] 内海地竹, 孟春謝, 崇喜中嶋, 徹森. 転移学習を用いた大喜利回答システムの構築. 2022 年度 情報処理学会関西支部 支部大会 講演論文集, 第 2022 巻, sep 2022.

[4] 龍山富, マハブービシェヘラザード, 洋二宮. 大喜利生成 ai の実装に関する研究. 第 85 回全国大会講演論文集, Vol. 2023, No. 1, pp. 759–760, 02 2023.

[5] 呉健朗, 中原涼太, 長岡大二, 中辻真, 宮田章裕. ボケて返す対話型エージェント. 日本バーチャルリアリティ学会論文誌, Vol. 23, No. 4, pp. 231–238, 2018.

[6] 健朗呉, 大二長岡, 涼太中原, 章裕宮田. 文のトピックを考慮した単語置換によるユーモア発話を行う対話型エージェント. 情報処理学会論文誌, Vol. 61, No. 1, pp. 113–122, jan 2020.

[7] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information, 2017.

[8] Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. Applying conditional random fields to Japanese morphological analysis. In Dekang Lin and Dekai Wu, editors, *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pp. 230–237, Barcelona, Spain, July 2004. Association for Computational Linguistics.

[9] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova.

Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

# 発表リスト

[1] 高橋巧実，鈴木優『意外性のある関連語の組み合わせと言語モデルを用いた大喜利の回答生成手法の提案』，東海関西データベースワークショップ 2023，2023
[2] 高橋巧実，鈴木優『言語モデルによる文の流暢さを用いた大喜利生成』，第 16 回データ工学と情報マネジメントに関するフォーラム，2024