優先度学習による推薦文からの見出し抽出

竹野　峻輔，氏原　淳志，岩永　二郎

Retty 株式会社は実名ユーザの投稿によるレストラン情報を，WEB サイトとアプリにて発信するグルメサービス Retty を運営している．サービス内に掲載されるレストランの一部には見出しが付与されており，簡潔で魅力的な見出しを作成することは，ユーザに対して価値のある情報を届けることであり，レストランの詳細ページに誘導し，有益なレストラン探しの体験を提供する点で重要な課題である．本稿では，Retty に投稿された大量の推薦文データと人手によって付けられた見出しをデータセットとして利用し，優先度学習の手法を用いる．優先度学習を用いることで比較的小さなデータセットからでも大量の訓練事例データを生成することができ，未知語処理やデータ拡張の手法を利用することで過学習を避けた汎化性能が高いモデルを構築することができる．

キーワード：自然言語処理，優先度学習，機械学習

1. はじめに

近年，インターネット上でレストラン情報を掲載するグルメサービスが広く普及している．Retty 株式会社は実名ユーザの投稿によるレストラン情報を，WEB サイトとアプリにて発信するグルメサービス Retty の運営を行っている．ユーザがお薦めのレストランに関する推薦文・画像・お薦め度を含む「投稿」を行うことで，Retty は CGM (Consumer Generated Media) サービスとして成立する．2017 年には約 80 万のレストラン，および数百万の投稿によって構成される大規模 CGM サービスとなっている．

Retty に掲載されるレストランの一部には見出しが付与されており，簡潔で魅力的な見出しを作成することは，ユーザに対して価値のある情報を届けることであり，レストランの詳細ページに誘導し，有益なレストラン探しの体験を提供する点で重要な課題である．

ここで見出しとは，レストランの宣伝を目的として，価値と魅力を簡潔に伝える文を指す．たとえば次のような文がレストランの見出しである．

1.「天然素材にこだわった，優しさと滋味に溢れたラーメンが食せる店」
2.「五反田で半世紀以上続く老舗洋食レストラン」
3.「ムーディーな雰囲気に美味しい鶏料理，大人の隠れ家的なお店」

たけの　しゅんすけ，うじはら　あつし，いわなが　じろう
Retty 株式会社
〒 108–0073 東京都港区三田 1–4–1
住友不動産麻布十番ビル 3F
shunsuke.takeno@retty.me
ujihara@retty.me
iwanaga@retty.me

見出しの特徴として，レストランの特徴や魅力を表す単語から構成されることに加え，動詞などを省略した非文が許されることが挙げられる．また，一般には作成する見出しに文字数制限が設けられていることも多い．

次に推薦文から見出しを抽出するタスクについて説明する．たとえば次の推薦文が与えられているとする．

「麻布十番のムーディーでしっぽりとした隠れ家．…
そのまた奥に，広い個室から 2 人席の個室まであり，
秘密基地のようなお店．…」

このとき，何かしらの基準に従ってレストランの特徴を簡潔に表す一文「麻布十番のムーディーでしっぽりとした隠れ家」を抽出するタスクが見出し抽出である．

見出し抽出タスクは，サービスに掲載されている約 80 万のレストランを対象としており，全レストランに人手で見出しを作成することは現実的に困難である．そのため見出しの作成を自動化することで，多くのレストランに自動で見出しを付与し，定期的に更新できる仕組みを構築できるようになる．

本稿では，上記の課題に対して，Retty に投稿された大量の推薦文データと人手によって付けられた見出しを学習データとして利用し，優先度学習を用いた方法でレストランの見出しを抽出する方法を紹介する．優先度学習を利用することで，大量のデータセットを作成することができ，見出しのよさを定義することも可能となる．また，未知語処理やデータ拡張の手法を利用することで過学習を避けた汎化性能が高いモデルを構築することができる．

モデル構築には，約 24 万店舗のレストランに対し人手によって作成された見出し文と，各々のレストランに投稿された延べ 324 万の投稿を利用し，学習を行った．

数値実験では，擬似的に作成したデータセットに対して見出しらしい文を 2 値分類モデルにより学習し，適切な見出し文を 99.3%の精度で判別することができた．

2. 関連研究

見出しの作成に向けた研究では，これまでにさまざまな手法が提案されている．これらの研究はテンプレート手法による見出しの作成，生成ベース手法による見出しの作成，抽出ベース手法による見出しの作成の 3 種類に大別することができる．

2.1 テンプレート手法による見出しの作成

テンプレートによる見出し生成は，事前にスロットをもつ見出しパタンを数多く用意しておく．見出しを作成したい文書が与えられたときに，適切なパタンを選択し，文書中からスロットを埋める要素を抽出してパタンにはめ込むことで見出しを作成する．Alfonseca et al. [1] は大規模なニュース記事集合から見出しの原型となるパタンをあらかじめ抽出しておき，新しい文書が与えられた際にはキーワード抽出を行い，パタンのスロットを埋めることで見出しを作成している．

テンプレート型の手法では，パタンを利用しているため文法を制御しやすいメリットがある一方で，事前にパタンを大量に用意しなければならないこと，パタンのスロットに適切な語を埋めるための情報抽出を高精度で行う必要があることなどのデメリットがある．

2.2 生成ベース手法による見出しの作成

深層学習の台頭により自然性の高い文を生成できるようになったことから，近年では生成手法を利用した研究が盛んに行われている [2, 3]．人手で作文した質の高い大規模なデータセットが用意されている場合において，文法的に正しく自然性の高い文を生成することができる．

一方でこれらの手法においては出力の制御に問題が発生する．深層学習を利用した手法の場合は，非線形変換が多層に重なることによってモデルの内部状態の解釈が難しく，線形モデルと比べ出力に対する解釈が難しくなるデメリットが存在する．また生成手法を利用した場合，見出しに対する事実性を保証することが難しく，ときに入力に含まれていない表現を利用した見出しが作成される可能性がある．

2.3 抽出ベース手法による見出しの作成

見出しをつけたい文書から文や単語列の抽出，削除などの加工操作を加えることによって見出しの作成を行う．Filippova [4] は入力の単語列から有向グラフを作成し，単語による文章の網羅率を上げるような経路探索の手法を利用して見出しの作成を行った．Jing [5] や Toutanova et al. [6] らは構文解析結果に対し，規則に基づく単語の枝刈りを行うことによる見出しの作成手法の提案を行っている．Clarke and Lapata [7] は n-gram の統計情報に基づき最適化問題として見出しの作成を行っている．Morita et al. [8] は述語項構造解析により，劣モジュラ最大化による部分木の抽出問題として最適化問題に帰着させている．

抽出型のアプローチをとるメリットとしては，作成された見出しが少なくとも推薦文に含まれるため事実性の担保ができることがある．また生成ベースの手法と違い，規則に基づくフィルタリング処理を入れやすく，制約を加えやすい点も挙げられる．

一方で，単語列の抽出や単語の削除などの処理を加えることで非文も生成されるため，出力の自然性を保つことが難しい．

3. 優先度学習

優先度学習 (Preference Learning) とは，訓練事例に基づき事例間の選好性を学習する機械学習手法の一つである [9, 10]．

本節では，優先度学習の手法の一つであるペアワイズ手法について導入する．ある事例対 (S_a, S_b) に対し事例間の順序関係 $\geq$ が存在するとき，優先度学習では次の関数の獲得を目的とする．

$$f(S_a, S_b) = \begin{cases} 1 \ (S_a \geq S_b) \\ 0 \ (\text{otherwise}) \end{cases} \tag{1}$$

任意の識別関数 F と，事例の特徴量のベクトルを出力する関数 ϕ を用いて上式を次のように表す．

$$f(S_a, S_b) = F(\phi(S_a) - \phi(S_b))$$

関数 F には任意の識別モデルを導入することができ，たとえばサポートベクターマシン (SVM) やロジスティック回帰 (LR) といった識別モデルが利用できる．

優先度学習は事例間の順序のみに注目したモデルを構築できる．本手法は見出し作成のように絶対的な評価指標を定義することが難しい場合において有効な手段である．本稿で取り扱う見出し作成のような問題においては，見出しに対して絶対的な数値による評価を下すことは人手でも難しいが，二つの見出しに対しよい見出し文を選ぶことは容易であり，相対評価の問題とすることで，比較的学習が容易な問題設定とすることができる．

3.1 オンライン学習によるモデルの更新

上述したように優先度学習では任意の識別関数を利用することができる．優先度学習では，順序付けられたデータから任意の二つのデータを選ぶことで一つの訓練事例を作り出すことができるため，大量の訓練事例の作成ができる．一方で大規模な機械学習は困難になり，バッチ処理的な手法を用いるには取り扱える事例数に限界がある．そこでオンライン処理による学習方法を導入する．ここでバッチ処理とはデータをすべて読み込んでから学習する方法であり，オンライン処理とはデータを少しずつ読み込んでモデル更新を繰り返す学習方法である．本稿では，オンライン学習の中でデータを小規模に分割して，モデル更新を繰り返すミニバッチ学習を用いる．

4. 優先度学習を利用した見出しらしさの学習

優先度学習を用いた見出しらしさを学習する方法を説明する．

4.1 見出しらしさの優先度学習

大量の推薦文と人手で付けた見出しのデータセットに対して，見出しらしさの順序関係に関する優先度学習を行う．すなわち，人手で付けた見出しは，推薦文に含まれる任意の文よりも「見出しらしい」という順序を優先度学習で学習する．あるレストランに対する見出し S_h と，そのレストランに対する推薦文に含まれる文 S_r が与えられたとき，正例 $f(S_h, S_r) = 1$ と負例 $f(S_r, S_h) = 0$ となる訓練事例を定める．これにより見出しが付与されているレストランに対して，推薦文数を上限とした訓練事例の機械的な作成が可能となる．

4.2 素性の設計

自然言語処理分野ではモデルの特徴量を素性（そせい），そして素性の抽出を行う関数を素性関数，素性関数により抽出されたベクトルを素性ベクトルと呼ぶ．

素性ベクトルの各成分は，特定の条件を入力文 S が満たす場合に 1，満たさない場合に 0 をとる．また頻度のように整数で表される素性の場合には，素性ベクトルの対応する成分は整数値をとる．

単語列 $x_1, \ldots, x_N$ からなる入力文 S に対し，本稿では次の素性を利用する．

1. 単語 unigram (x_n) [2 値]
2. 単語 bigram $(x_n \cdot x_{n-1})$ [2 値]
3. 単語 trigram $(x_n \cdot x_{n-1} \cdot x_{n-2})$ [2 値]
4. 入力文 S に含まれる括弧は対となっているか否か [2 値]（「」，『』，【】，()，“”，[] のそれぞれに対して個別に適用）
5. 文中に含まれる単語数 N [整数値]

単語に関する unigram，bigram，trigram 素性は n-gram 素性と呼ばれる自然言語処理において基本的な特徴量であり，特定の系列が S 中に含まれる場合には 1，含まれない場合には 0 をとる関数である．すなわち単語 unigram の場合はある単語が S 中に含まれるか否かを表し，単語 trigram は特定の順序の 3 単語の連なりが，文 S 中に含まれるか否かを表す値となる．括弧に関する素性は文中の長期の依存関係を表す素性の一つであり，部分文字列を抽出した際の括弧の不一致を防ぐ目的をもつ．単語 trigram では，これらの構造的な特徴量を表すことが不可能なため，素性として組み込んでいる．

優先度学習における上記の素性の解釈を述べる．単語 unigram は見出しにおける単語の重要度に関する特徴量であり，単語 bigram，単語 trigram は次節の未知語処理と合わせることで見出しのパタンに関する特徴量となる．

4.3 未知語処理

自然言語処理の問題においては，評価データセット中には出現するが訓練データセット中に出現しないような単語を一般に未知語 (Unknown Words) と呼ぶ．訓練データセットだけでは見出し語に成り得る単語をすべて網羅することは現実的に困難であるため，未知語処理を加えることで効果的な学習ができる．

訓練データセットで特定の単語を未知語として処理することで，学習するモデルが汎化される．訓練データセットで頻度の低い単語を未知語を表す形式的な単語 (i.e., unk) に置き換える．たとえば，「麻布十番のムーディーでしっぽりとした隠れ家」に対して「麻布十番」「ムーディー」「しっぽり」を未知語として処理すると「unk の unk で unk とした隠れ家」という文となる．このように未知語処理を行うことで事例の抽象化を行い，見出しのパタンを学習することができる．とりわけ固有名詞などは低頻度であることが多く，未知語として処理することでモデルの過学習を防ぐことができる．後述の数値実験では単語の出現頻度として 10 を閾値として未知語処理を行った．

4.4 データ拡張

画像処理の分野では，画像に対して拡大縮小処理や回転処理を加えたデータの拡張操作 (Data Augmentation) を行うことで過学習を回避する手法が一般的に用いられる．

本稿では，次の手順に従って順序関係を与えること

表 1　データセットの統計情報

		訓練	開発	評価
見出し文	文数	239,873	6,312	6,313
	平均単語数	14.3	13.6	13.6
	平均文字数	29.4	28.1	28.2
推薦文	文書数	3,238,221	80,955	80,956
	平均文数	4.5	4.6	4.5
	平均単語数	15.3	14.9	14.8
	平均文字数	133.5	136.0	135.6
事例数	対	14,571,995	372,393	364,302

でデータ拡張を行った．訓練事例に含まれる見出し S_a と推薦文に含まれる文 S_r をランダムに一つずつ選ぶ．S_r に含まれる連続した単語部分列をランダムに選び $\acute{S_r}$ とし，$(S_a, \acute{S_r})$ を新たに訓練事例として加える．この操作を全体の訓練事例数が一定量になるまで繰り返すことで，データ拡張を行った．

4.5　推薦文からの見出し抽出

訓練事例より獲得したモデルを利用し，実際に推薦文から見出しを抽出する方法について説明を行う．抽出方法は推薦文に対して部分文字列をすべて列挙し，one-versus-one による見出し候補の順序付けを行う．One-versus-one による順序付けは，すべての見出し候補の文の組み合わせについて予測を行い，すべての中で最も見出しらしいと判別された回数が多い文を見出しとして採用する．

5.　実験

見出しの抽出に関する数値実験結果について説明する．

5.1　実験設定

実験に利用するデータセット及び利用するツールを説明する．単語分割器は MeCab[1] を利用し，形態素解析辞書は mecab-unidic[2] を利用する．

人手によって作成された見出し 252,498 文と，各々のレストランに紐づけられた文分割済みの 3,400,132 文書（約 1,530 万文）に対し，レストランごとの見出しと文書中の推薦文を対として事例を作成する．

このうち 95%を訓練データセット（14,571,995 対），2.5%を開発データセット(372,393対)，2.5%(364,302対) を評価データセットとして実験に用いる．これらデータセットの詳細な統計情報を表 1 に示す．

5.2　自動評価による精度の比較

データセットの前処理として，文頭記号 <s> と文末記号 </s> を文の前後に加える．これにより文頭や文末に関する素性を明示的に取り扱う．未知語処理として見出し全体で出現頻度 10 未満の単語をすべて未知語として取り扱う．

学習に利用するモデルには，オンライン学習可能な識別モデルであるパーセプトロン (Perceptron)，ロジスティック回帰モデル (LR)，サポートベクターマシン (SVM) の三つの手法を利用する．

各モデルのハイパパラメータ選択は開発データセットを利用し，対象のハイパパラメータ以外を固定して探索を行い，最もよいものを実際の学習に用いた．サポートベクターマシンおよびロジスティック回帰モデルについては，L1 正則化または L2 正則化において正則化の強さを $\{1, 0.1, 0.01\}$ より選択した，3 手法共通のハイパパラメータである学習率は $\{0.1, 0.05, 0.005\}$ より選択した．

評価方法は，2 値ラベルにおける精度 (accuracy) を利用する．評価データセットは 2 種類用意する．一つは，見出しと文分割された推薦文一文を対とした事例に対する評価である．もう一つは前節で説明したデータ拡張を評価データセットに適用したものである．評価セットに対してもデータ拡張を行うことで，部分文字列から見出しを抽出する実際の状況設定と近い評価が行える．

結果を表 2 に示す．訓練事例に対してデータ拡張を適用し学習を行った結果を表 3 に示す．

本稿の実験設定において，一様に無作為の予測をした場合の期待正解率（チャンスレート）は 50%であるのに対して，実際に得られた精度は比較的高い．本実験では擬似的にデータセットを作成したため，容易に判別が可能な事例が評価データに多数含まれており，このことにより高精度に学習ができたと考えられる．

表 2　手法による優先度学習の精度の比較

	開発	評価	評価（データ拡張）
Perceptron	99.3%	99.2%	98.2%
LR	99.3%	99.1%	98.6%
SVM	99.5%	99.3%	98.7%

表 3　訓練データに対してデータ拡張を行った際の精度の比較

	開発	評価	評価（データ拡張）
Perceptron	99.4%	99.3%	98.7%
LR	99.3%	99.2%	99.3%
SVM	99.7%	99.4%	99.1%

[1] http://taku910.github.io/mecab/
[2] https://ja.osdn.net/projects/unidic/

表 4　優先度学習により抽出された見出しの例

成功例 1	⟨s⟩ 東京 で 一番 美味しい もつ 鍋 屋 さん 。 もつ鍋 ダイニング の お 店 ⟨/s⟩
成功例 2	⟨s⟩ 日曜日 でも ランチ が ある コスパ が 良い 焼肉 屋 さん ⟨/s⟩
誤り例 1（非文）	⟨s⟩ の 木 々 の 緑 が 気持 ち の 良い カフェ ⟨/s⟩
誤り例 2（主観の入り込み）	⟨s⟩ 立ち 食い そば の お 店 で は 好き な お 店 ⟨/s⟩
誤り例 3（不必要な情報の入り込み）	⟨s⟩ 実家 近く の うどん 屋 ⟨/s⟩

もっとも精度がよくなった場合は SVM を採用した場合であり，評価データにおいて 99.3%，データ拡張した評価において 98.7%の精度を得ることができた．

表 3 と表 2 を比較すると，訓練データに対するデータ拡張を行ったことで，同じくデータ拡張を行った評価セットにおいて効果的に学習が行えていることがわかる．データ拡張時において最も精度が高くなった LR においてはデータ拡張前が 98.6%であるのに対して，データ拡張を行うことで精度は 99.3%まで改善している．これよりデータ拡張がモデルの汎化性能の向上に貢献していることがわかる．

5.3　部分文字列の抽出により作成された見出しの定性評価

5.2 節において，擬似データセットによる評価を行った．本節では，見出しのないレストランに対し実際に見出し抽出を行い，得られた出力に対して定性的な評価を行う．

見出しが存在していない店舗に対して，投稿の推薦文より 4.5 節に記載した方法により見出しの抽出を行う．投稿が 5 投稿以上存在する店舗のうち 100 店舗を対象に，文字数が 15 文字以上 35 文字未満であるもののみを候補文として選択し，one-versus-one による予測により最もよい候補文を選ぶ．

上記の見出し抽出の処理を行い，筆者による見出しの適切さの判定を行った．この結果から成功例 2 例および誤り例 3 例を抽出したものを表 4 に示す．

十分な投稿数が用意されることで，表 4 の成功例のような，適切な見出しを抽出できることがわかった．

表 4 に示した事例を基に手法の誤り分析を行う．誤り例 1 では，文字数制限の制約を満たすために，文頭に不要な“の”が付与された文が抽出されている．誤り例 2 は，推薦文投稿者の主観が見出しに表れているため見出しとして不適切である．誤り例 3 は，推薦文投稿者の“実家”は読者にとって情報量がないため見出しとして不適切である．

誤り例 1 に関する問題は，候補選択時に制限を満たさないものを一括で除去してしまうことによる問題であるため，文字数の制限をソフトな制約に変えることで修正可能と考えられる．たとえば，見出しの出力が決定された後に，字数制限なしで 4.5 節の操作を再度行うことが考えられる．誤り例 2, 3 は語彙選択の問題であり，規則による単語のフィルタリングが対策として考えられる．

6.　結論

本稿では，Retty に投稿された大量の推薦文データと人手によって付けられた見出しをデータセットとして優先度学習によって見出しらしさを学習し，推薦文データから見出しの抽出をした．

抽出ベースのアプローチは，作成された見出しが少なくとも推薦文に含まれるため，事実として正しいことが期待される．また，生成ベースの手法と違い，規則に基づくフィルタリングができるため制約を加えやすい．

優先度学習を用いることで，比較的小さなデータセットからでも大量の訓練事例データを生成することができる．また，未知語処理やデータ拡張の手法を利用することで，非文も見出しの候補として採用でき，かつ過学習を避けた汎化性能が高いモデルを構築することができた．

本稿では，店舗に関する情報を一切利用せずに学習を行ったが，今後の展望として，対象店舗の情報の属性に応じた見出し作成が考えられる．

参考文献

[1] E. Alfonseca, D. Pighin and G. Garrido, “HEADY: News headline abstraction through event pattern clustering,” In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pp. 1243–1253, 2013.

[2] A. M. Rush, S. Chopra and J. Weston, “A neural attention model for sentence summarization,” In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 379–389, 2015.

[3] A. M. Rush, S. Chopra and J. Weston, “Abstractive sentence summarization with attentive recurrent neural networks,” In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 93–98, 2016.

[4] K. Filippova, “Multi-sentence compression: Finding shortest paths in word graphs,” In *Proceedings of the 23rd International Conference on Computational Linguistics*, pp. 322–330, 2010.
[5] H. Jing, “Sentence reduction for automatic text summarization,” In *Proceedings of the Sixth Conference on Applied Natural Language Processing*, pp. 310–315, 2000.
[6] K. Toutanova, C. Brockett, M. Gamon, J. Jagarlamudi, H. Suzuki and L. Vanderwende, “The pythy summarization system: Microsoft research at duc 2007,” In *Proceedings of Seventh Document Understanding Conference*, 2007.
[7] J. Clarke and M. Lapata, “Constraint-based sentence compression an integer programming approach,” In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pp. 144–151, 2006.
[8] H. Morita, R. Sasano, H. Takamura and M Okumura, “Subtree extractive summarization via submodular maximization,” In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pp. 1023–1032, 2013.
[9] R. Herbrich, T. Graepel, P. Bollmann-Sdorra and K. Obermayer, “Learning preference relations for information retrieval,” In *Proceedings of ICML-98 Workshop: Text Categorization and Machine Learning*, pp. 80–84, 1998.
[10] T. Joachims, “Optimizing search engines using clickthrough data,” In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 133–142, 2002.

