

文間及びカテゴリ間の関係性を捉えたレーティング予測

外山洋太 三輪誠 佐々木裕

豊田工業大学 知能数理研究室

{sd12056, makoto-miwa, yutaka.sasaki}@toyota-ti.ac.jp

1 序論

企業において商品の評判分析のためのレビューの評判分類は重要な問題である。何万件という大量のレビューデータを人手で処理することは難しく、計算機による自動化が望まれる。その中でレビューを複数のカテゴリにおいて分類する問題がある。カテゴリとは、宿泊施設のレビューを例にすると、サービス、立地、食事等のレーティングが付けられる各項目のことである。この問題に関する従来手法 [1] は、文間の関係性を考慮しておらず、カテゴリ間については考慮しているものの深い関係性を捉えることができていない。

近年、その評判分類において、ニューラルネットワークを用いた手法が提案されており、従来の手法を上回る正答率を達成している。ニューラルネットワークを分類問題に用いることの利点の一つは層の数を増やすことによって入力となる特徴量の深い意味を捉えられることである。多カテゴリの分類問題に適用すれば、カテゴリ間の関係性を捉えた分類が実現できる。さらに、文毎の特徴量を入力とすれば文間の関係性も捉えることができる。

文や文章の特徴量としては、パラグラフベクトル [2] が分類問題に対して優れていることが示されている。

以上より、本研究では、複数カテゴリにおける評判分類について、パラグラフベクトル [2] とニューラルネットワークを用いて文間及びカテゴリ間の関係性を捉えた分類を実現し、従来手法 [1] から正答率を向上させることを目的とする。

提案手法では、パラグラフベクトル [2] によって生成された各レビューの特徴量をニューラルネットワークの分類器において分類しレーティング予測を行う。特徴量としては、各レビューの文書ベクトルに加え、各レビュー内の文ベクトルの重み付け平均を用いた。分類器は全結合ニューラルネットワークによって構成されており、文間及びカテゴリ間の関係性を捉えた分類を行う。

実験において、提案手法は従来手法 [1] に対して 2pp

以上上回る正答率を示した。正解レーティングと予測レーティングの RMSE を元にした評価基準では、従来手法 [1] において弱点となっていたカテゴリについてそれを上回る結果を示した。ただし、いくつかのカテゴリでは従来手法 [1] と変わらないか、それを下回る結果が示された。また、パラグラフベクトルが文書と文について適用された場合、文書ベクトルと文ベクトルはいくらか異なる特徴を捉えていることが示された。

2 関連研究

2.1 隠れ状態を用いたホテルレビューのレーティング予測

藤谷ら [1] は複数のカテゴリにおける評判分類問題に対して、レビュー内の各文毎に予測した隠れレーティングからレビュー全体のレーティングを予測する手法を提案している。文毎のレーティングからレビュー全体のレーティングを予測する際のカテゴリ間の繋がりを手動で変化させカテゴリ間の関係性を考慮している。各文の組成は BOW または n-gram であり、単語や n-gram の順序は無視されている。

実験により、各文毎に隠れレーティングを予測することによって正答率が向上すること、また、カテゴリ間の繋がりによって正答率が変化することが示されている。

2.2 パラグラフベクトル

パラグラフベクトルは、文や文章といった大きな単位の言語表現の意味表現を学習する手法である。これは、Continuous Bag Of Words (CBOW) または Skip-gram という単語の意味表現の学習手法を応用した手法である。ここでは CBOW を応用した PV-DM について説明する。



図 1: 文書及び文ベクトルの学習の概略

以下に具体的なアルゴリズムを示す。ここでは文章の意味表現を学習する場合について考える。学習の概略を図 1 に示す。

まず、意味表現を学習する対象となる文章に含まれる単語を初めから一つずつ読んでいく。その際、現在の単語及びその周辺の単語、現在の文章について考える。以下の式 1 に示す目的関数を最大化するように各パラメータの学習を行う。

$$L = \frac{1}{T} \sum_{t=k}^{T-k} \log p(w_t | w_{t-k}, \dots, w_{t+k}) \quad (1)$$

$$p(w_t | w_{t-k}, \dots, w_{t+k}) = \frac{e^{y_{w_t}}}{\sum_i e^{y_i}} \quad (2)$$

$$y = b + Uh(w_{t-k}, \dots, w_{t+k}, d; W, D) \quad (3)$$

ここで、 d は文章、 w_i は単語、 W は全ての単語の分散表現を表す行列、 D は全ての文章の分散表現を表す行列である。 k は片側のウィンドウサイズ、 T は現在の文章に含まれる単語数である。ある単語の周辺を表す区間をウィンドウという。 y は現在の単語とウィンドウ内の単語及び現在の文章から導出される正規化されていない対数確率である。 p は softmax 関数により正規化された、文脈から現在の単語が導かれることの尤度である。 $h(w_{t-k}, \dots, w_{t+k}, d; W, D)$ は引数となるベクトルを平均したベクトルまたは結合したベクトルを返す関数である。

これにより、Bag Of Words (BOW) と異なり、単語の並び順を考慮した文や文章の分散表現を生成することができる。

2.3 ニューラルネットワークを用いた評判分析

ニューラルネットワークを用いた評判分析の手法が、Nal ら [3]、Rie ら [4]、Duyu ら [5] 等によって提案されている。

これらの方法に共通するのは、単語の意味表現から畳み込みニューラルネットワークと全結合ニューラル

ネットワークを用いて分類を行うことである。まず、単語の意味表現から畳み込みニューラルネットワークを用いて単語同士の関係を捉えた特徴量を抽出する。その後、そこから得られた文書あるいは文全体の特徴量を全結合ニューラルネットワークの入力とし多値または二値分類を行う。また、Duyu ら [5] と Nal ら [3] の手法はニューラルネットワークのモデルの中にパラメータとして単語の意味表現を取り込んでいる。これにより、特定の分類問題に対してそれを直に微調整することが可能となる。

3 提案手法

提案手法では、PV-DM によってレビュー内の各文及び文章の分散表現を生成し、それらをニューラルネットワークの入力として分類を行う。以下に具体的なアルゴリズムを示す。

3.1 アルゴリズム

アルゴリズムの入力はレビュー等の文書の集合、出力は各文書について予測されたカテゴリ毎のクラスである。

初めに、PV-DM を用いて、各レビューの文書ベクトルとそれに含まれる各文のベクトルを生成する。文書ベクトルと文ベクトルについては別々に学習し生成する。以下の目的関数を最大化するように学習を行う。

$$L_d = \sum_{t=1}^T \{ \log \sigma(s(w_t, w_{t-n}, \dots, w_{t-1}, d)) + \sum_{w'_t \sim P_n} \log(1 - \sigma(s(w'_t, w_{t-n}, \dots, w_{t-1}, d))) \} \quad (4)$$

$$s(w_t, w_0, \dots, w_n, d) = W_{map}(w_t) \cdot \begin{bmatrix} W(w_0) \\ \vdots \\ W(w_n) \\ D(d) \end{bmatrix} \quad (5)$$

ここで、 T は現在の文書内の単語数、 t は現在の単語位置、 d は現在の文書、 w_i は位置 i にある単語である。 $W(w_i)$ は単語 w_i に相当する単語ベクトルを単語行列 W から抜き出す関数を表す。 $D(d)$ は文書 d に相

当する文書ベクトルを文書行列 D から抜き出す関数を表す。関数 $s(w_t, w_0, \dots, w_n, d)$ はある単語とそれが出現する文脈との類似度を計算する。行列 W_{map} は。内積によって文脈と単語との類似度を計算するための単語毎のベクトルを保持する。文書行列内の各文書ベクトルはレビュー全体を表す文書ベクトル、または、各レビュー内の文ベクトルを表す。 σ はシグモイド関数である。また、式 4 の中括弧内の右項はネガティブサンプリングを表す。ネガティブサンプリングとは、文脈外の単語をデータセットにおける出現確率でサンプリングし、それらと文脈の意味が遠ざかるように学習する手法である。 $w'_t \sim P_n$ は文脈外の単語 w'_t を単語の出現頻度 P_n によってサンプリングすることを示す。ただし、出現頻度は極端に頻出する単語の影響を抑えるため各単語に対して $3/4$ 乗している。現在の単語と同じ単語や同一回の学習で一度サンプリングされた単語はサンプリングされない。

次に、各レビュー内の全ての文ベクトルに対して重み付け平均を行い、圧縮された文ベクトルを生成する。この過程により、各レビューで疎らだった文の数に統一される。以下に重み付け平均を行う関数を示す。

$$t_{i_{sect}} = \sum_{i_{sent}} \frac{\mathbf{w}(x_{i_{sect}}(i_{sent}))}{|\mathbf{w}(x_{i_{sect}}(i_{sent}))|} s_{i_{sent}} \quad (6)$$

$$x_{i_{sect}}(i_{sent}) = \frac{i_{sent} - i_{sect}}{\#sections} \quad (7)$$

$$\mathbf{w}(x) = (w_1(x), w_2(x), \dots, w_{i_{sect}}(x), \dots, w_{\#sentences}(x)) \quad (8)$$

$$w(x) = \frac{1}{2}(\cos(\pi|x|) + 1) \quad \text{if } |x| \leq 1 \quad (9)$$

ここで、 i_{sent} はレビュー内の文のインデックス、 $\#sections$ は重み付け平均された後の文ベクトルの数、 i_{sect} は重み付け平均された後の文ベクトルのインデックスである。 $s_{i_{sent}}$ は文ベクトル、 $t_{i_{sect}}$ は圧縮された後の文ベクトルである。

次に、レビュー毎の特徴量を用いて分類器によってレーティング予測を行う。各レビューの特徴量については後述する。分類器は全結合ニューラルネットワークによって構成される。図 2 に各層の結合の様子を示す。

ニューラルネットワークの活性化関数には、シグモイド関数を用いた。また、出力層はカテゴリの数とラベルの数の積だけのユニットを持ち、各ユニットの出力はそれが含まれるカテゴリ内でそれが示すクラスが選ばれることの正規化されていない対数確率を表す。

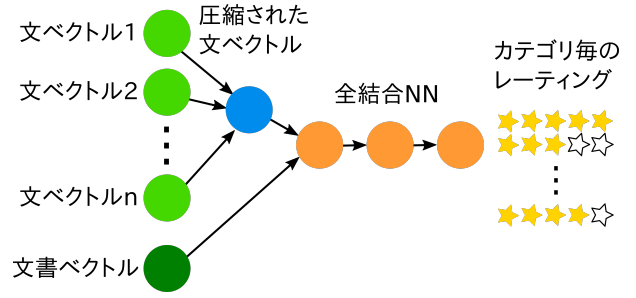


図 2: 全結合ニューラルネットワークによる分類器

表 1: 各基準手法に用いられる特徴量

手法	特徴量
基準手法	レビュー全体の文書ベクトル
提案手法 1	レビュー内で平均した文ベクトル
提案手法 2	レビュー内で重み付け平均によって圧縮された文ベクトル
提案手法 3	レビュー全体の文書ベクトル、 レビュー内で重み付け平均によって圧縮された文ベクトル

ニューラルネットワークの最小化すべき目的関数を式 10 に示す。

$$E = - \sum_{n=1}^N \sum_{c=1}^C \sum_{k=1}^K d_{nck} \log y_{ck}(x_n; w) \quad (10)$$

$$y_{ck}(x_n; w) = \frac{e^{u_k}}{\sum_j^K e^{u_j}} \quad (11)$$

各ユニットの出力はカテゴリ毎にソフトマックスクロスエントロピー関数によって損失に変換される。ここで、 $y_{ck}(x_n; w)$ はカテゴリ c においてクラス k が選ばれる対数確率である。 N はミニバッチサイズ、 C はカテゴリの総数、 K はクラスの総数である。

提案手法としては、Quoc ら [2] によって提案されている基準手法を拡張し、NN への入力となる特徴量が異なる 3 つの手法を提案する。基準手法に加え各提案手法に用いられる特徴量を表 1 に列挙する。

4 実験

提案手法と従来手法 [1] 及び基準手法を比較するため 2 つの評価基準を用いて実験を行った。

表 2: 各手法のパラメータ設定

手法	学習回数	中間層でのユニット数
基準手法	20	512
提案手法 1	55	256
提案手法 2	24	256
提案手法 3	30	512

表 3: 各手法における正答率

手法	正答率
従来手法 [1]	0.4832
基準手法	0.4969
提案手法 1	0.4848
提案手法 2	0.4866
提案手法 3	0.5038

4.1 実験設定

実験は、各手法の正答率を測定するものと、その実験で最も正答率の高かった提案手法における予測レーティングと回答レーティングの RMSE を測定するものの 2 つを行った。

正答率を測定する実験は基準手法及び 3 つの各提案手法について行った。データセットとしては、先行研究 [1] と同様に、ホテル予約サイト楽天トラベルにおけるレビュー 337,266 件からレビューの番号順に訓練データ 300,000 件、開発データ 10,000 件、評価データ 10,000 件を用いた。

レーティングの RMSE を測定する実験は、上記の正答率を測定する実験で得られた予測レーティングを用いて行った。

表??に各手法におけるニューラルネットワークのパラメータ設定を示す。全ての手法において、中間層の数は 1、入力層及び中間層におけるドロップアウト率はそれぞれ 0.2 と 0.5 で共通である。提案手法 2 と提案手法 3 において圧縮された文ベクトルの数はそれぞれ 3 つと 2 つとした。全ての実験において文書及び文ベクトルについては、学習回数は 1024 回、学習する単語の範囲は前 3 単語、単語の最少出現回数は 5 回、ベクトルの次元数は 600 次元に設定し学習したものをを用いた。

4.2 結果

表 3 に正答率を測定する実験の実験結果及び従来手法 [1] による結果を示す。表 4 に RMSE を測定する実験の実験結果を示す。

表 4: 提案手法 3 と従来手法 [1] におけるレーティングの RMSE

手法	提案手法 3	従来手法 [1]
立地	0.87	0.82
部屋	0.88	0.92
食事	2.54	3.23
風呂	1.00	1.45
サービス	0.85	0.85
設備	0.90	0.97
総合	0.73	0.67

4.3 考察

表 3 より、提案手法 3 が従来手法 [1] の正答率を 0.0206 上回っていることから、提案手法 3 が従来手法 [1] より正答率において優れていることが示された。また、提案手法 2 の正答率が提案手法 1 の正答率を 0.018 上回っていることから、文の位置関係の考慮がレーティング予測に有効であることが示された。さらに、提案手法 3 が基準手法や提案手法 2 に比べ高い正答率を示していることから、文書ベクトルと文ベクトルを同時に特徴量として用いることがレーティング予測に有効であることが示された。これは文書ベクトルと文ベクトルがいくらか異なる特徴を学習していることを示す。

表 4 より、提案手法 3 は従来手法 [1] が苦手としていた食事と風呂のカテゴリにおいてそれぞれ 0.69 及び 0.45 だけ低い誤差を示した。藤谷ら [1] より、本実験で用いたデータセットには、食事のカテゴリにおいて 0 点が付与されたレビューが 108,079 件存在する。一般に、0 点はユーザが何らかの理由でレーティング不可能と判断したことを示す。よって、提案手法 3 は従来手法 [1] よりレビュー中の上記のような意味をよく捉えていると考えられる。しかし、その他のカテゴリにおいては提案手法 3 と従来手法との間に有意な差は見られなかった。

5 結論

本研究では、レビュー全体の文書の分散表現に加え、レビュー内の各文ベクトルに対する分散表現の重み付き平均を用いた評判分類の手法を提案した。

実験により、従来手法 [1] 及びレビュー全体の文書ベクトルのみを用いた手法に比べ、提案手法が高い正答率を示すことが分かった。同時に、これはレビュー内の文の並びが評判分類に重要であることを示す。

5.1 今後の課題

今後の課題は、2 つに分かれているモデルの統合である。

提案手法は、分類すべき文書とそれが含む文の分散表現を生成する段階、及び、それらを用いて分類を行う段階の2つの段階に分かれている。このことは、問題を2つに分けることで個々の問題を単純にしているが、同時に文書の分類を一つずつ行うことを難しくしている。また、提案手法において、文書や文の分散表現を事前に生成するための PV-DM の手法におけるパラメータは、実際には最大の正答率を達成するため分類器のパラメータとして最適化されることが望ましい。

今後は、これらの問題を解決するために、文書や文の分散表現を生成する過程をニューラルネットワークによる分類器に統合する。これによって、学習方法の柔軟性を高めると共にさらなる正答率の向上を目指す。

参考文献

- [1] 藤谷宣典ら, 隠れ状態を用いたホテルレビューのレーティング予測. 言語処理学会第 21 回年次大会, 2015.
- [2] Quoc Le, and Tomas Mikolov, Distributed Representations of Sentences and Documents. ICML 2014, 2014.
- [3] Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom, A Convolutional Neural Network for Modelling Sentences. ACL 2014, 2014.
- [4] Rie Johnson, and Tong Zhang, Effective Use of Word Order for Text Categorization with Convolutional Neural Networks. NAACL 2015, 2015.
- [5] Duyu Tang, Bing Qin, and Ting Liu, Learning Semantic Representation of Users and Products for Document Level Sentiment Classification. ACL 2015, 2015.