Data Science and Engineering (2020) 5:180–192
https://doi.org/10.1007/s41019-020-00122-4

Contextual Sentiment Neural Network for Document Sentiment Analysis

Tomoki Ito[1] · Kota Tsubouchi[2] · Hiroki Sakaji[1] · Tatsuo Yamashita[2] · Kiyoshi Izumi[1]

Received: 12 March 2020 / Revised: 28 April 2020 / Accepted: 30 April 2020 / Published online: 20 May 2020

Abstract
Although deep neural networks are excellent for text sentiment analysis, their applications in real-world practice are occasionally limited owing to their black-box property. In this study, we propose a novel neural network model called contextual sentiment neural network (CSNN) model that can explain the process of its sentiment analysis prediction in a way that humans find natural and agreeable and can catch up the summary of the contents. The CSNN has the following interpretable layers: the word-level original sentiment layer, word-level sentiment shift layer, word-level global importance layer, word-level contextual sentiment layer, and concept-level contextual sentiment layer. Because of these layers, this network can explain the process of its document-level sentiment analysis results in a human-like way using these layers. Realizing the interpretability of each layer in the CSNN is a crucial problem in the development of this CSNN because the general back-propagation method cannot realize such interpretability. To realize this interpretability, we propose a novel learning strategy called initialization propagation (IP) learning. Using real textual datasets, we experimentally demonstrate that the proposed IP learning is effective for improving the interpretability of each layer in CSNN. We then experimentally demonstrate that the CSNN has both the high predictability and high explanation ability.

Keywords Interpretable neural networks · Text mining · Support system

1 Introduction

1.1 Motivation and Purpose

Massive web documents such as micro-blogs and customer reviews are useful for public opinion sensing and trend analysis. The sentiment analysis approach (i.e., to automatically predict whether a review is overall positive or negative) has been commonly used in this area. Deep neural networks (DNNs) are some of the best-performing machine learning methods [1]. However, DNNs are often avoided in cases where explanations are required because these networks are generally considered as black boxes. Thus, developing a high predictable neural network (NN) model that can explain the process of its prediction process in a human-like way is a critical problem. In the development of such NN model, we should consider how humans usually judge the positive or negative polarity of each review. As described in some previous linguistic researches [2–4], it is well known that humans judge the positive or negative document-level polarity of each review with extracting four types of word-level scores in the following order.

1. Word-level original sentiment score: this score means the sentiment that each word in a review originally has (e.g., scores in a word sentiment dictionary [5]).
2. Word-level sentiment shift score: this score means the sentiment of each term in a review is shifted or not (e.g., "good" in "not good" and "goodness" in "decrease the goodness.")
3. Word-level global important point score: This score means the important part of the entire review.
4. Word-level contextual sentiment score: this score means the positive or negative sentiment score of each term

Electronic supplementary material The online version of this article (https://doi.org/10.1007/s41019-020-00122-4) contains supplementary material, which is available to authorized users.

✉ Tomoki Ito
m2015titoh@socsim.org

[1] Graduate School of Engineering, The University of Tokyo, 7-3-1, Hongo, Bunkyo-ku, Tokyo, Japan

[2] Yahoo Japan Corporation, Tokyo, Japan

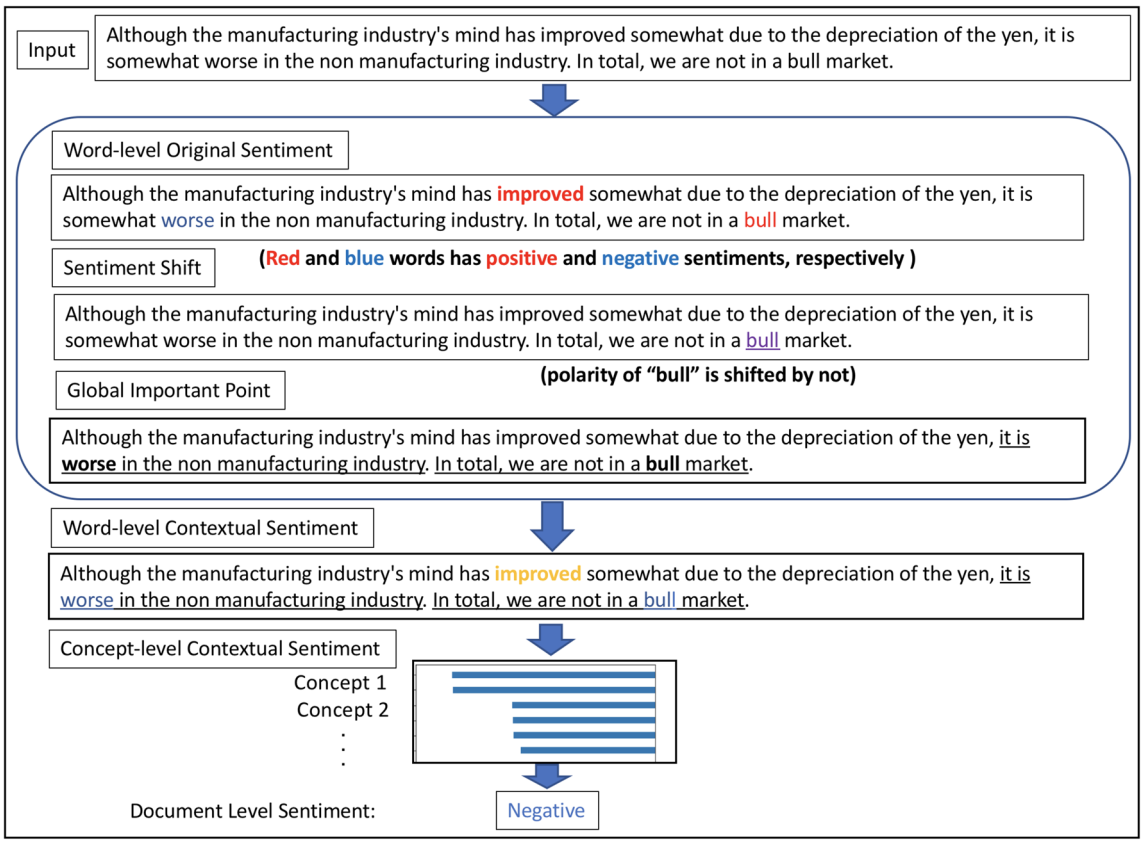

Fig. 1 Goal: development of neural network (NN) that can explain its prediction results using four types of sentiments

after considering the sentiment shift and global important point.

In addition, as described in previous text visualization research [4], the following concept-level contextual sentiment score is important for readers to catch up the summary of the review content.

5. Concept-level contextual sentiment score: this score means the concept-level positive or negative sentiment of each review where a concept means a set of similar terms.

Therefore, neural network models that can (1) analyze document-level sentiment with high predictability and (2) explain the prediction results using the above five types of sentiments as shown in Fig. 1 should have a great demand in the industry:

However, a method for developing such NNs is yet to be established. Many studies have been done to address the black-box property of the NNs [4, 6–14]; however, it is hard to say that these previous works can realize the interpretability in the form that humans can find natural and agreeable because these previous studies alone cannot describe the above five types of scores. For example, interpretable NNs with attention mechanism [6, 7] can describe the global important point of each term in a review; however, they cannot describe the other three types of word-level sentiment scores. Interpretable NNs that include word-level original sentiment scores (i.e., original sentiment interpretable NN) [4, 8, 9] can describe the word-level original sentiment scores; however, they cannot describe the word-level global and local contextual sentiment scores. As for other approaches, methods for interpreting NNs can describe the word-level global sentiment scores [10–14]; however, they cannot describe the other scores.

1.2 Approach

To solve this problem, we propose a novel NN model called contextual sentiment neural network (CSNN) and a novel learning strategy called initialization and propagation (IP) learning.

1.2.1 CSNN

CSNN has the following four interpretable layers: word-level original sentiment layer (WOSL), sentiment shift

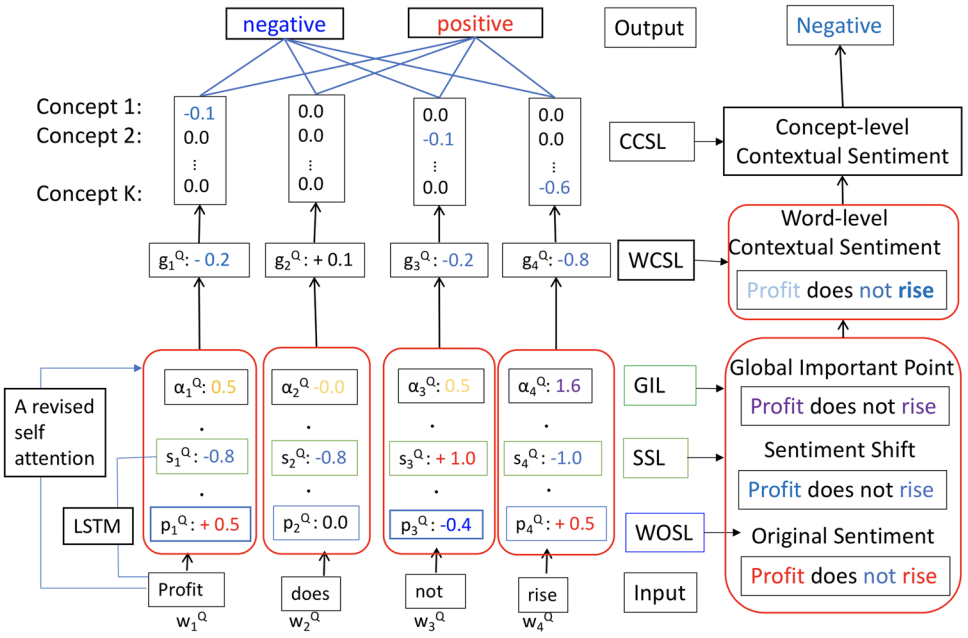

Fig. 2 Structure of CSNN

layer (SSL), global important point layer (GIL), and word-level contextual sentiment layer (WCSL), and concept-level contextual sentiment layer (CCSL) as shown in Fig. 2. The WOSL and WCSL represent the word-level original and contextual sentiment of each term in a review, respectively. The SSL indicates whether a sentiment of each term in a review is shifted or not, and GIL indicates the global important points in a review. The WOSL is represented in a word sentiment dictionary manner. The SSL and GIL are represented using long short-term memories (LSTM) cells [15] and attention mechanism [16, 17], respectively. The values of WCSL are represented by multiplying the values of WOSL, SSL, and GIL. The values of CCSL are represented by the WCSL and the K-means clustering results with the word embeddings following the strategy in [4].

Therefore, using the WOSL, SSL, GIL, and WCSL, the CSNN can explain the process of the sentiment analysis prediction in a form that humans find natural.

1.2.2 IP Learning

In developing this CSNN, realizing the interpretability for WOSL, SSL, GIL, and WCSL is a crucial problem. Generally, sentiment analysis models are developed using the back-propagation method with the gradient values for the loss value between the predicted document-level sentiment and the positive or negative tag of each review; however, when such general back-propagation method is used, each layer does not represent the corresponding sentiment. Thus, to realize the interpretability of layers in CSNN, we propose a novel learning strategy called the initialization and propagation (IP) learning.

IP learning includes two specific strategies called Init and Update. Update is a strategy of regularization for the final weight matrix, which is expected to improve the interpretability in WCSL. Init is a strategy for initialization of the WOSL using a small word sentiment dictionary that is composed of a few hundreds of word-level original sentiment scores, which is expected to improving the interpretability in WOSL and GIL. Using both the Update and Init, the interpretability in SSL is also expected to be improved. IP learning requires only reviews, their sentiment tags, and a small word sentiment dictionary. It does not require any sentiment shift information or syntactic text analysis. This is a valuable point in our approach because we can develop CSNN even for minor language or non-grammatical documents.

We experimentally evaluated the performance of the proposed approach using real textual datasets. We first demonstrated that IP learning is useful for realizing the interpretability of each layer in the CSNN. We then demonstrated that the CSNN developed with IP learning has both the high predictability and high explanation ability.

1.3 Contribution

The contributions of this paper are as follows:

- We proposed a novel NN architecture called CSNN that can explain its sentiment analysis process in a form that humans find natural and agreeable.
- To realize the interpretability of CSNN, we proposed a novel learning strategy called IP learning.
- We experimentally demonstrated the high interpretability and high predictability of the proposed CSNN.

The remainder of this paper is structured as follows. In Sect. 2, the CSNN architecture and IP learning are explained in detail. Section 3 pre-experimentally evaluates the effect of the proposed IP learning. Section 4 presents the experiments and results. Section 5 presents the related works. In Sect. 6, the conclusion and directions for future work are discussed.

2 CSNN

This section introduce the proposed CSNN. A CSNN as described in Sect. 2.1 can be developed through IP learning (Sect. 2.1) using a training dataset $\{(\mathbf{Q}_i, d^{\mathbf{Q}_i})\}_{i=1}^{N}$, and a small word sentiment dictionary. Note that N is the training data size, $\mathbf{Q}_i$ is a comment, and $d^{\mathbf{Q}_i}$ is its sentiment tag (1 is positive and 0 is negative).

2.1 Structure of CSNN

This section introduces the CSNN structure. The CSNN includes the following layers: WOSL, SSL, GIL, WCSL, CCSL, and outputs the document-level sentiment.

Notation. Before explaining the construction of the CSNN model, we define several symbols. Let $\{w_i\}_{i=1}^{v}$ represent the terms that appear in a text corpus of a dataset, and v be the vocabulary size. We define the vocabulary index of word w_i as $I(w_i)$. Therefore, $I(w_i) = i$. Let $\boldsymbol{w}_i^{em} \in \mathbb{R}^e$ be an embedding representation of word w_i, and the embedding matrix $\boldsymbol{W}^{em} \in \mathbb{R}^{v \times e}$ be $[\boldsymbol{w}_1^{emT}, \ldots, \boldsymbol{w}_v^{emT}]^T$. Here, e is the dimension size of word-level embedding. Then, for each i, $\|\boldsymbol{w}_i^{em}\|_2 = 1$ is satisfied. $\boldsymbol{W}^{em}$ is the constant value obtained using the skip-gram method [18] and the text corpus in a training dataset.

2.1.1 WOSL

Given a comment $\mathbf{Q} = \{w_t^{\mathbf{Q}}\}_{t=1}^{n}$, this layer converts the words $\{w_t^{\mathbf{Q}}\}_{t=1}^{n}$ to original word-level sentiment representations $\{p_t^{\mathbf{Q}}\}_{t=1}^{n}$:

$$p_t^{\mathbf{Q}} = w^p_{I(w_t^{\mathbf{Q}})} \tag{1}$$

where $\boldsymbol{W}^p \in \mathbb{R}^v$ represents the original sentiment scores of words, and w_i^p is the i−th element of $\boldsymbol{W}^p$. The w_i^p value corresponds to the original sentiment score of the word w_i.

2.1.2 SSL

First, this layer converts terms $\{w_t^{\mathbf{Q}}\}_{t=1}^{n}$ in comment $\mathbf{Q}$ into their word-level embeddings $\{\boldsymbol{e}_t^{\mathbf{Q}}\}_{t=1}^{n}$ using $\boldsymbol{W}^{em}$, and converts them to context representations $\{\overrightarrow{\boldsymbol{h}}_t^{\mathbf{Q}}\}_{t=1}^{n}$ and $\{\overleftarrow{\boldsymbol{h}}_t^{\mathbf{Q}}\}_{t=1}^{n}$ using forward and backward long short-term memories, $\overrightarrow{\text{LSTM}}$ and $\overleftarrow{\text{LSTM}}$ [15]:

$$\overrightarrow{\boldsymbol{h}}_t^{\mathbf{Q}} = \overrightarrow{\text{LSTM}}(\boldsymbol{e}_t^{\mathbf{Q}}), \overleftarrow{\boldsymbol{h}}_t^{\mathbf{Q}} = \overleftarrow{\text{LSTM}}(\boldsymbol{e}_t^{\mathbf{Q}}). \tag{2}$$

Second, it converts $\{\overrightarrow{\boldsymbol{h}}_t^{\mathbf{Q}}\}_{t=1}^{n}$ and $\{\overleftarrow{\boldsymbol{h}}_t^{\mathbf{Q}}\}_{t=1}^{n}$ to right- and left-oriented sentiment shift representations, $\overrightarrow{s}_t$ and $\overleftarrow{s}_t^{\mathbf{Q}}$:

$$\overleftarrow{s}_t^{\mathbf{Q}} = \tanh(\boldsymbol{v}^{left} \cdot \overleftarrow{\boldsymbol{h}}_t^{\mathbf{Q}}), \overrightarrow{s}_t^{\mathbf{Q}} = \tanh(\boldsymbol{v}^{right} \cdot \overrightarrow{\boldsymbol{h}}_t^{\mathbf{Q}}). \tag{3}$$

Here, $\boldsymbol{v}^{right}, \boldsymbol{v}^{left} \in \mathbb{R}^e$ are parameter values. $\overrightarrow{s}_t^{\mathbf{Q}}$ and $\overleftarrow{s}_t^{\mathbf{Q}}$ denote whether or not the sentiment of $w_t^{\mathbf{Q}}$ is shifted by the left-side and right-side terms of $w_t^{\mathbf{Q}}$: $\{w_{t'}^{\mathbf{Q}}\}_{t'=1}^{t-1}$ and $\{w_{t'}^{\mathbf{Q}}\}_{t'=t+1}^{n}$, respectively.

Finally, this layer converts $\{\overrightarrow{s}_t^{\mathbf{Q}}\}_{t=1}^{n}$ and $\{\overleftarrow{s}_t^{\mathbf{Q}}\}_{t=1}^{n}$ into word-level sentiment shift scores $\{s_t^{\mathbf{Q}}\}_{t=1}^{n}$:

$$s_t^{\mathbf{Q}} := \overrightarrow{s}_t^{\mathbf{Q}} \cdot \overleftarrow{s}_t^{\mathbf{Q}}. \tag{4}$$

$s_t^{\mathbf{Q}}$ denotes whether the sentiment of $w_t^{\mathbf{Q}}$ is shifted ($s_t^{\mathbf{Q}} < 0$) or not ($s_t^{\mathbf{Q}} \geq 0$).

The overall structure of this SSL is shown in Fig. 3.

2.1.3 GIL

This layer represents the word-level global important point representations $\{\alpha_t^{\mathbf{Q}}\}_{t=1}^{n}$ using a revised self-attention mechanism [16, 17] as

$$\alpha_t^{\mathbf{Q}} := \sum_{t'=1}^{T} \frac{e^{\tanh(\overrightarrow{\boldsymbol{h}}_t^{\mathbf{Q}T} \overrightarrow{\boldsymbol{h}}_{t'}^{\mathbf{Q}} + \overleftarrow{\boldsymbol{h}}_t^{\mathbf{Q}T} \overleftarrow{\boldsymbol{h}}_{t'}^{\mathbf{Q}})}}{\sum_{t=1}^{T} e^{\tanh(\overrightarrow{\boldsymbol{h}}_t^{\mathbf{Q}T} \overrightarrow{\boldsymbol{h}}_{t'}^{\mathbf{Q}} + \overleftarrow{\boldsymbol{h}}_t^{\mathbf{Q}T} \overleftarrow{\boldsymbol{h}}_{t'}^{\mathbf{Q}})}}. \tag{5}$$

2.1.4 WCSL

Using the WOSL, SSL, and GIL, this layer represents word-level contextual sentiment representations $\{g_t^{\mathbf{Q}}\}_{t=1}^{n}$:

$$g_t^{\mathbf{Q}} := p_t^{\mathbf{Q}} \cdot s_t^{\mathbf{Q}} \cdot \alpha_t^{\mathbf{Q}} \tag{6}$$

2.1.5 CCSL

This layer converts $\{g_t^{\mathbf{Q}}\}_{t=1}^{n}$ into the concept-level contextual sentiment representations $\{\boldsymbol{v}_t^{\mathbf{Q}}\}_{t=1}^{n}$:

$$\boldsymbol{v}_t^{\mathbf{Q}} = g_t^{\mathbf{Q}} \boldsymbol{b}_t^{\mathbf{Q}} \tag{7}$$

where $\boldsymbol{b}_t^{\mathbf{Q}} := \max(\text{Softmax}(\boldsymbol{W}_c \boldsymbol{e}_t^{\mathbf{Q}} - t_c), 0)$, $\boldsymbol{v}_t^{\mathbf{Q}} \in \mathbb{R}^K$, $\boldsymbol{b}_t^{\mathbf{Q}} \in \mathbb{R}^K$, $t_c > 0$ is a hyper-parameter value, $\boldsymbol{W}_c \in \mathbb{R}^{K \times e}$ is centroid vectors of $\{\boldsymbol{w}_i^{em}\}_{i=1}^{v}$ calculated using a spherical k-means method [19] where the cluster number is K. Here, the (i, k) element of $\boldsymbol{b}_t^{\mathbf{Q}}$ represents the cluster weight of word $w_t^{\mathbf{Q}}$ to cluster k. Therefore, from the values in the CCSL, we can catch up the concept-level contextual sentiment scores.

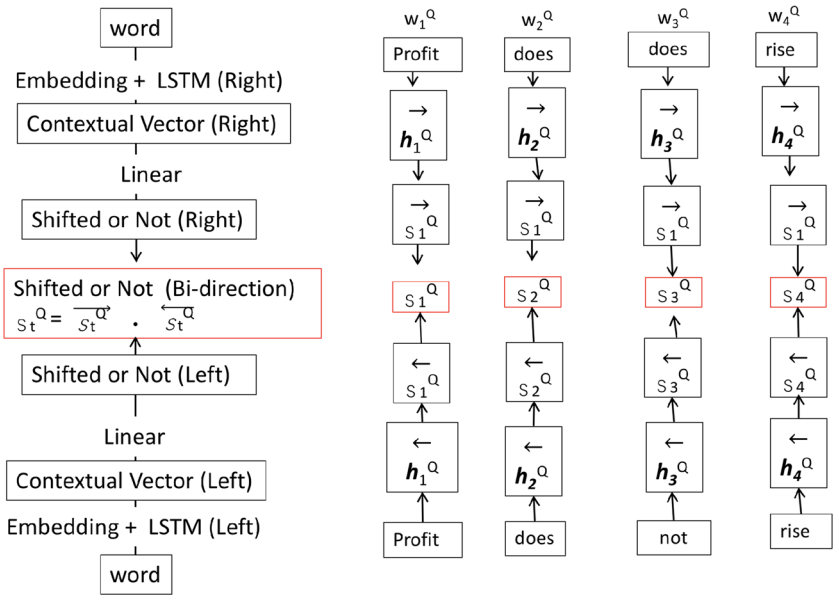

Fig. 3 SSL architecture

2.1.6 Output

Finally, this NN converts $\{\boldsymbol{v}_t^{\mathbf{Q}}\}_{t=1}^n$ into a predicted sentiment tag $y^{\mathbf{Q}} \in \{0(\text{negative}), 1(\text{positive})\}$:

$$\boldsymbol{a}^{\mathbf{Q}} = \text{Softmax}\left(\boldsymbol{W}^O \tanh\left(\sum_{t=1}^{n} \boldsymbol{v}_t^{\mathbf{Q}}\right)\right),$$

$$y^{\mathbf{Q}} = \text{argmax}\boldsymbol{a}^{\mathbf{Q}}$$

where $\boldsymbol{W}^O \in \mathbb{R}^{2\times K}$ is the parameter value.

2.2 Key Idea in IP learning

In developing CSNN, the realization of the interpretability in WOSL and SSL is especially difficult. Through the learning with $L^{\mathbf{Q}}$ and Update (will be defined later), WCSL learns to represent corresponding sentiments. However, this learning strategy alone cannot realize the interpretability in WOSL and SSL because in the case where the polarity of $c_t^{\mathbf{Q}}$ is accurately negative, the following two cases are possible: (1) $p_t^{\mathbf{Q}} > 0$ and $s_t^{\mathbf{Q}} < 0$, or (2) $p_t^{\mathbf{Q}} < 0$ and $s_t^{\mathbf{Q}} > 0$, and the accurate case cannot be chosen automatically in general learning. We assume that this problem can be solved by initially limiting the polarity of $p_t^{\mathbf{Q}}$ to the accurate case for a few words because this limitation leads to the accurate choice from the above two cases. Therefore, this limitation can lead to the learning of $s_t^{\mathbf{Q}}$ within the appropriate case. The effect of this limitation works for only the limited words, first; however, this effect is assumed to be propagated to the other non-limited terms whose meanings are similar to any of the limited words thorough learning, afterward. To realize this idea, we utilize the Init (will be defined later) in IP learning.

2.3 Initialization and Propagation (IP) Learning

This section describes the learning strategy of the CSNN. Overall process is described in Algorithm 1 where $w_{i,j}^o$ is the (i, j) element of $\boldsymbol{W}^O$, and $L^{\mathbf{Q}}$ is the cross entropy between $\boldsymbol{a}^{\mathbf{Q}}$ and $d^{\mathbf{Q}}$. IP learning utilizes the two specific techniques called *Update* and *Init*. *Update* is a strategy for improving the interpretability in *WCSL*. *Init* is a strategy for improving the interpretability in *WOSL* and *GIL*. Using both the *Update* and *Init*, the interpretability in *SSL* is also expected to be improved (as theoretically analyzed in Appendix A in the supplementary material).

2.3.1 Update

First, $\boldsymbol{W}^O$ is updated according to processes 6–7 in Algorithm 1. This makes *WCSL* to represent the corresponding sentiment scores (Proposition A.3 in Appendix) without violating the learning process after sufficient iterations (Proposition A.7 in Appendix A).

2.3.2 Init

Then, $\boldsymbol{W}^p$ is initialized as process 2 in Algorithm 1, where $PS(w_i)$ is the sentiment score for word w_i given by the word sentiment dictionary, and S^d is a set of words from the dictionary. Init makes *WOSL* and *SSL* represent the corresponding scores in the condition that Update is utilized.

Through this IP learning, for every word sufficiently similar to any of the words in S^d, the *WOSL*, *SSL*, *GIL*, and *WCSL* learn to represent the corresponding scores, as theoretically analyzed in Appendix A. After the learning, the CSNN can explain its prediction result using these layers.

Algorithm 1 Initialization and Propagation (IP) Learning

1: **for** $i \leftarrow 1$ to v **do**
2: $w_i^p \leftarrow \begin{cases} PS(w_i) & (w_i \in S^d) \\ 0 & (\text{otherwise}) \end{cases}$;
3: **while** learning has not been finished **do**
4: Update $\boldsymbol{W}^p$, $\boldsymbol{v}^{right}$, $\boldsymbol{v}^{left}$, $\boldsymbol{W}^O$ and the LSTM cells in CSNN using the gradient values by $L^{\mathbf{Q}}$. ;
5: **for** $k \leftarrow 1$ to K **do**
6: **if** $w^o_{1,k} > 0$ **then** $w^o_{1,k} \leftarrow 0$;
7: **if** $w^o_{2,k} < 0$ **then** $w^o_{2,k} \leftarrow 0$;

3 Pre-experimental Evaluation for IP Learning

This section experimentally tests the explanation ability and predictability of the CSNN and investigate the effect of IP learning for the interpretability of the layers in the CSNN.

3.1 Dataset

3.1.1 Text Corpus

We used the following four textual corpora, including reviews and their sentiment tags, for this evaluation. They were used for developing CSNN.

(a) *EcoRevs I and II.* These datasets are composed of comments on current (I) and future (II) economic trends and their positive or negative sentiment tags[1]
(b) *Yahoo review.* This dataset is composed of comments on stocks and their long (positive) or short (negative) attitude tags, extracted from financial micro-blogs.[2]
(c) *Sentiment 140.* This dataset contains tweets and their positive or negative sentiment tags.[3]

EcoReviews and Yahoo review were Japanese datasets, and Sentiment 140 was an English dataset. We used them to verify whether the CSNN can be used irrespective of the language or domain. We divided each dataset into the training, validation, and test datasets, as presented in Table 1.

[1] https://www5.cao.go.jp/keizai3/watcher-e/index-e.html.
[2] http://textream.yahoo.co.jp.
[3] https://www.kaggle.com/kazanova/sentiment140.

Table 1 Dataset organization for text corpus

	EcoRev I	EcoRev II	Yahoo	Sentiment 140
Training				
Positive reviews	20,000	35,000	30,612	650,000
Negative reviews	20,000	35,000	9388	650,000
Validation				
Positive reviews	2000	2000	3387	50,000
Negative reviews	2000	2000	1613	50,000
Test				
Positive reviews	4000	4000	7538	100,000
Negative reviews	4000	4000	2462	100,000
Vocabulary size v	8071	11,130	33,080	71,316

3.1.2 Annotated Dataset

For this evaluation, we prepared the Economy, Yahoo, and message annotated datasets. The Economy annotated dataset has 2200 reviews (1100 positive and 1100 negative) in the test dataset of EcoReviews I. The Yahoo annotated dataset has 1520 reviews (760 positive and 760 negative) in the test dataset of Yahoo reviews. The message annotated dataset has 10258 reviews obtained from the test datasets in SemEval tasks [20, 21]. In these datasets, part of the terms in reviews had word-level contextual sentiment tags and word-level sentiment shift tags.

Word-level contextual sentiment tags indicate whether the word-level contextual sentiments of terms are positive or negative as shown in the following examples.

(1) In total, we are in a *bull*$^+$ market.
(2) This room is not *clean*$^-$.
(3) Products in this shop are too *expensive*$^-$.

Word-level sentiment shift tags indicate whether the sentiments of terms were shifted (1: shifted tags) or not (0: non-shifted tags) as shown in the following examples.

(1) In total, we are in a *bull*$^{(0)}$ market.
(2) This room is not *clean*$^{(1)}$.
(3) Products in this shop are too *expensive*$^{(1)}$.

Moreover, in the message annotated dataset, part of phrases in reviews have positive or negative tags for contextual sentiments (phrase-level sentiment tags) as the following examples.

(1) In total, we are in a {*bull market*}$^+$.
(2) This room is {*not clean*}$^-$.
(3) Products in this shop are {*too expensive*}$^-$.

Table 2 Dataset details for text corpus and annotated data

(i) Word polarity list					
	EcoRev I	EcoRev II	Yahoo	Sentiment 140	
Positive	348	337	422	1843	
Negative	391	387	372	947	
(ii) Sentiment shift tags					
	EcoRev I	EcoRev II	Yahoo	Sentiment 140	
Shifted tags	872	859	378	429	
Non-shifted tags	3762	3740	2391	4504	
(iii) Word-level global important point tags					
	EcoRev I	EcoRev II	Yahoo	Sentiment 140	
Important tags (1)	6632	6631	1526	-	
Unimportant tags (0)	62,652	62,652	48,890	-	
(iv) word-level and phrase-level contextual polarity tags					
	EcoRev I	EcoRev II	Yahoo	Sentiment 140	
Level	Word	Word	Word	Word	Phrase
Shifted negative	776	756	227	169	–
Non-shifted negative	1491	1483	1187	1294	–
Shifted positive	96	96	151	260	–
Non-shifted positive	2271	2179	1204	3210	–
Negative (total)	2267	2239	1414	1463	3634
Positive (total)	2367	2275	1355	3470	5907

In addition, a gold global important point (0: not important or 1: important) is assigned to each term of the reviews included in the Economy and Yahoo annotated datasets. This gold global important point indicates that each term in a review is important (1) or not (0) for deciding the overall positive or negative polarity of the review as the following examples.

(1) $We^{(0)}$ $are^{(0)}$ $in^{(0)}$ $a^{(0)}$ $bull^{(1)}$ $market^{(1)}$.
(2) $This^{(0)}$ $room^{(0)}$ $is^{(0)}$ $not^{(1)}$ $1clean^{(1)}$.

These tags were used in evaluating the explanation ability of the CSNN. We used the Economy, Yahoo, and message annotated datasets when developing CSNNs with the EcoReviews, Yahoo reviews, and Sentiment 140, respectively. We only employed tags of terms that were not used in Init and appeared in the training dataset, and only used tags of the phrases that include at least one term involved in the training dataset. Table 2 summarizes the numbers of tags used. See the supplementary material for details.

3.2 CSNN Development Setting

We developed the CSNN using each training and validation datasets in the following settings.

Setting in Init. *Init* used a part of a Japanese financial word sentiment dictionary (JFWS dict) developed by six financial professionals and the Vader word sentiment dictionary (Vader dict) [5]. These dictionaries contain words and their sentiment scores. After we excluded the words with zero sentiment scores and those with absolute sentiment scores of less than 1.0 from JFWS dict and the Vader dict, respectively, we extracted most frequent 200 words in each training dataset from these dictionaries and used their sentiment scores in *Init*. To analyze the results in the cases where *Init* used fewer words, we evaluated the results with CSNNs developed with only 50 or 100, or 200 words: CSNN (50), CSNN (100) and CSNN (200).

Other settings. We calculated the word embedding matrix $\boldsymbol{W}^{em}$ by the skip-gram method (window size = 5) [18] based on each textual dataset. We set the dimensions of the hidden and embedding vectors to 200, epoch to 50 with early stopping, K to [100, 500, 1000], t_c to $1/K$, and mini-batch size to 64. We used stratified sampling [22] to analyze imbalanced data, and the Adam optimizer [23], and the dropout [24] method (rate = 0.5) for the BiRNNs and CSNNs. We calculated $\boldsymbol{W}^{em}$ using the skip-gram method (window size = 5) with each text corpus. We determined the hyper-parameters using the validation data. We used the mean score of the five trials for the evaluations in this paper.

3.3 Evaluation Metrics in Explanation ability

Evaluation Metric. We evaluated the explanation ability of the CSNN based on the validity in WOSL, SSL, GIL, and WCSL in the following way.

3.3.1 Validity of WOSL

We evaluated the validity of WOSL based on how accurately the polarities of word w_i and w_i^p agree using the economic, Yahoo, and LEX word polarity list[4]). These lists include words and their positive or negative polarities. The economic and Yahoo word-polarity lists include Japanese economic terms, and LEX word-polarity list includes English terms. If we used the EcoReview I or II, Yahoo reviews, and Sentiment 140 in training, we utilized the economic, Yahoo, and LEX word polarity lists, respectively. Moreover, we used only those terms that appeared in the training dataset but were not used in *Init*. Table 1 summarizes the number of words used in evaluating the CSNN developed with each dataset.

3.3.2 Validity of SSL

Using the sentiment shift tags in the annotated datasets, we evaluated the validity of the SSL based on whether the sentiment shift tags of $w_t^{\mathbf{Q}}$ and the polarity of $s_t^{\mathbf{Q}} > 0$ (shifted: $w_i^p < 0$ and non-shifted: $w_i^p > 0$) is accurately agreed well.

3.3.3 Validity of GIL

Using the gold word-level global important points in the annotated datasets, we evaluated the validity of the GIL based on whether the values of GIL $\{\alpha_t^{\mathbf{Q}}\}_{t=1}^n$ and gold word-level global important points were correlated. We used the Pearson correlation coefficient for this evaluation.

3.3.4 Validity of WCSL

Using the word-level or phrase-level contextual sentiment tags in the annotated datasets, we evaluated the validity of the WCSL with regard to whether the values of WCSL in CSNN could accurately assign the word or phrase-level contextual sentiments, that is, whether $g_t^{\mathbf{Q}}$ was accurately positive (negative) when the contextual word-level sentiment of $w_t^{\mathbf{Q}}$ was positive (negative) or whether the polarity of the summed scores for terms involved in each phrase accurately presented its sentiment. We used the macro average score between the macro F_1 score for shifted terms and that for non-shifted terms for the evaluation basis. We used this score to test whether each method could accurately correspond to both shifted and non-shifted terms.

In the above, the values for the WOSL, SSL and WCSL are evaluated using the F1 Score because the of range of values for the WOSL and WCSL is $[-\infty, \infty]$ and the range of values for the SSL is $(-1, 1)$. In contrast, the range of values for the GIL is $[0, \infty]$. Thus, we evaluated the validity of GIL by the Pearson Correlation.

Baselines. To evaluate the effect of IP learning, we compared the results of the CSNNs developed with IP learning and those of the following baseline models, namely, $CSNN^{Base}$, $CSNN^{NoInit}$, and $CSNN^{NoUp}$. The structures of these baseline models are the same as the structure of CSNN; however, they are different in the following points:

- $CSNN^{Base}$ is developed using the general backpropagation and without Update or Init strategy.
- $CSNN^{Random}$ is developed with only Update strategy.
- $CSNN^{NoUp}$ is developed with only Init strategy.

Comparison Method. To evaluate the explanation ability of CSNN, we compared the evaluation result of CSNN with other comparative methods in each layer validity.

(1) WOSL: This evaluation compared the CSNN with the other word-level original sentiment assignment methods, namely, PMI [25], logistic fixed weight model (LFW) [8], sentiment-oriented NN (SONN) [9], and gradient interpretable neural network (GINN) [4].
(2) SSL: This evaluation compared the CSNN with the baseline and NegRNN methods. In the baseline, we predicted $w_t^{\mathbf{Q}}$ as "shifted" if the sentiment of $d^{\mathbf{Q}}$ predicted by the RNN and sentiment tag of $w_t^{\mathbf{Q}}$ assigned by the PMI were different and as "not shifted" in other cases. In NegRNN, we used the RNN that predicts polarity shifts [26] developed with the polarity shifting training data created by the weighed frequency odds method [27].
(3) GIL: This evaluation compared the CSNN with the other word-level important point assignment methods using the RNNs using attention mechanism: word attention network (ATT) [28], hierarchical attention network (HN-ATT) [28], sentiment and negation neural network (SNNN) [29], and lexicon-based supervised attention (LBSA) [6]. SNNN and LBSA are set up in a form that the attention weights of terms with the strong word-level original sentiment are strengthened. We used the attention score of each model as the score.
(4) WCSL: This evaluation compared the CSNN with the other word-level sentiment assignment methods: PMI, LFW, SONN, GINN, Grad + a bidirectional LSTM model (RNN) [12], LRP + RNN [30], and IntGrad + RNN [11].

[4] http://quanteda.io/reference/data_dictionary_LSD2015.html.

Table 3 Evaluation for explanation ability in WOSL (Macro F_1 score)

	EcoRev I	EcoRev II	Yahoo	Sentiment 140
PMI	0.734	0.745	0.793	0.733
LFW	0.715	0.740	0.766	0.725
SONN	0.702	0.724	0.725	0.705
GINN	0.723	0.755	0.754	0.735
$CSNN^{Base}$	0.417	0.381	0.499	0.373
$CSNN^{NoUp}$	**0.832**	**0.846**	**0.798**	**0.754**
$CSNN^{Rand}$	0.452	0.543	0.460	0.430
CSNN (200)	0.837	**0.865**	**0.825**	**0.742**
CSNN (100)	0.838	0.851	0.817	**0.744**
CSNN (50)	**0.843**	**0.865**	0.805	**0.743**

Best scores are in bold

4 Experimental Evaluation for CSNN

4.1 Evaluation Metrics in Predictability

Evaluation Metric. We evaluates the predictability of the CSNN based on whether it can predict the sentiment tags of reviews in each test dataset. **Comparison Method.** We compared the CSNN and the following methods: logistic regression (LR), LFW [8], SONN [9], GINN [4], a bi-LSTM based RNN (RNN), convolutional NN (CNN)[1], ATT[28], HN-ATT [28], SNNN [29], LBSA [6]. We used the macro F_1 score as the evaluation basis.

Among the above methods, LR is a linear representation model. LFW, SONN, and GINN are original sentiment interpretable NNs. ATT, HN-ATT, SNNN, and LBSA are NNs with attention mechanism, and especially, SNNN and LBSA are set up in a form that the attention weights of terms with the strong word-level original sentiment are strengthened.

4.2 Result

4.2.1 Explanation ability and Predictability

Tables 3, 4, 5 and 6 summarize the results for explanation ability, indicating that the proposed CSNN outperformed the other methods in most cases. Table 7 summarizes the results, indicating that HN-ATT had greater predictability than the proposed CSNNs in most cases; however, CSNN (200) had greater predictability than LR and some deep NNs such as CNN and SNNN, and had predictability equivalent to that of ATT or LBSA. These results demonstrate that the proposed CSNN has both the high explanation ability and high predictability.

Table 4 Evaluation for explanation ability in SSL (Macro F_1 score)

	EcoRev I	EcoRev II	Yahoo	Sentiment 140
Baseline	0.660	0.712	0.579	0.560
NegRNN	0.536	0.626	0.564	0.558
$CSNN^{Base}$	0.661	0.311	0.244	0.314
$CSNN^{NoUp}$	0.374	0.246	0.360	0.417
$CSNN^{Rand}$	0.263	0.531	0.315	0.293
CSNN (200)	0.777	0.804	0.691	0.743
CSNN (100)	0.780	0.816	0.681	0.751
CSNN (50)	0.784	0.809	0.675	0.762

Table 5 Evaluation for explanation ability in GIL (Pearson correlation)

	EcoRev I	EcoRev II	Yahoo	Sentiment 140
ATT	− 0.015	− 0.081	0.062	–
HN-ATT	0.108	0.188	0.262	–
SNNN	0.281	0.456	0.192	–
LBSA	0.333	0.344	**0.405**	–
$CSNN^{Base}$	0.014	0.170	0.171	–
$CSNN^{NoUp}$	**0.607**	**0.590**	**0.329**	–
$CSNN^{Rand}$	0.207	0.224	0.164	–
CSNN (200)	0.595	0.580	0.325	–
CSNN (100)	0.584	0.567	0.308	
CSNN (50)	0.585	0.562	0.321	–

Best scores are in bold

Table 6 Evaluation for explanation ability in WCSL (Macro F_1 score)

	EcoRev I	EcoRev II	Yahoo	Sentiment 140	
Level	Word	Word	Word	Word	Phrase
PMI	0.578	0.548	0.575	0.631	0.822
Grad + RNN	0.578	.621	.601	0.681	0.743
IntGrad + RNN	0.607	0.621	0.625	0.679	0.796
LRP + RNN	0.597	0.518	0.579	0.638	0.808
LFW	0.549	0.545	0.578	0.587	0.749
SONN	0.555	0.542	0.566	0.600	0.787
GINN	0.569	0.555	0.577	0.623	0.831
$CSNN^{Base}$	0.355	0.521	0.490	0.575	0.595
$CSNN^{NoUp}$	0.416	0.316	0.526	0.509	0.512
$CSNN^{Rand}$	0.606	0.621	0.516	0.794	0.748
CSNN (200)	0.676	0.711	0.669	**0.788**	0.858
CSNN (100)	0.679	**0.723**	**0.675**	0.784	**0.862**
CSNN (50)	**0.692**	0.719	0.670	**0.788**	0.857

Best scores are in bold

Table 7 F_1 score results for the predictability evaluation

	EcoRev I	EcoRev II	Yahoo	Sentiment 140
LR	0.878	0.879	0.741	0.785
LFW	0.876	0.840	0.751	0.745
SONN	0.863	0.876	0.717	0.776
GINN	0.860	0.859	0.740	0.782
CNN	0.894	0.911	0.757	0.820
RNN	0.922	0.932	0.749	**0.837**
ATT	0.924	0.937	0.750	0.835
HN-ATT	**0.927**	**0.940**	0.750	**0.837**
SNNN	0.918	0.928	0.752	0.827
LBSA	0.922	**0.941**	0.762	0.832
CSNN (200)	0.921	0.938	**0.768**	0.833
CSNN (100)	0.914	0.937	**0.762**	0.835
CSNN (50)	0.916	0.939	**0.765**	0.833

Best scores are in bold

4.2.2 Effect of IP Learning

The results of CSNNs, $CSNN^{Base}$, $CSNN^{NoUp}$, and $CSNN^{Rand}$ for explainability demonstrate the effect of IP learning as follows. The $CSNN^{Rand}$ outperformed the $CSNN^{Base}$ in WCSL, indicating that Update promoted the validity in WCSL; whereas, the $CSNN^{NoUp}$ outperformed the $CSNN^{Base}$ in WOSL and GIL, indicating that Init promoted the validity in WOSL and GIL. Consequently, the validity in all the five layers were improved by using both Update and Init, and the CSNNs outperformed the $CSNN^{Base}$ in all the cases. This is the expected result as described in Sect. C (and Appendix A in the supplementary).

4.3 Discussion

We then discuss the performance of the CSNN in detail.

4.3.1 Predictability

The reason behind the good performance of HNATT in the predictability evaluation may lie in whether the sentence-level importance is considered or not. The HNATT considers the sentence-level importance, whereas the CSNN does not consider it. Therefore, it is possible that the performance for the CSNN can become better by adding the sentence-level importance attention mechanism to the CSNN. Additionally, it should be noted that the performance for the CSNN was better than the others in Yahoo dataset. It is possible that this is because sentiment shift representations in Yahoo dataset are more general and complex than those in EcoReviews. The CSNN directly strengthens the word-level sentiment score and its sentiment shift. Thus, the CSNN can address the sentiment shift representations in Yahoo dataset.

4.3.2 Effect of IP Learning

It should be noted that the interpretability for the CSNN has succeeded even when we used only fifty terms for the Init and there has been significant difference for the setting of Init. These results indicate that the number of the required minimum words for the learning was less than fifty and our algorithm was sufficiently practical.

4.3.3 Sentiment Shift Detection Performance in Yahoo Dataset

Sentiment shift representations in Yahoo dataset are more general and complex than those in EcoReviews. We consider that this is the reason for the better performance of the CSNN. The CSNN directly strengthen the word-level sentiment score and its sentiment shift. Thus, the CSNN can address the sentiment shift representations in Yahoo dataset.

4.4 Text-Visualization Example

This section introduces some examples of text-visualization produced by the CSNN. Figures 4 and 5 show the text-visualization examples for visualizing a review in Yahoo review and a review in the Sentiment 140 using the CSNN. Users can explain the CSNN's prediction process based on this type of text-visualizations.

In addition, based on the values of the right- and left-oriented sentiment shift representations, we can interpret the sentiment shift processes in the CSNN. Figure 5 shows examples. Based on Fig. 5, we can interpret that "uru (bearish)" is shifted by its right-side terms, and term "aoru (manipulate)" caused a sentiment shift because in the right-oriented sentiment shift representations, the terms to the left side of "aoru (manipulate)" become blue. In the same manner, we can interpret that "great" is shifted by "not" (right-oriented shift layer) in Fig. 4.

5 Related Work

There are many studies for addressing the black-box property of the deep NNs. As a useful technique for explaining the prediction results of NNs, we can present methods for interpreting prediction models [10–13, 31, 32]. These methods calculated the gradient score of each input feature in the prediction and visualized an important feature in their predictions. The LRP method is one of the state-of-the-art methods. Interpretable NNs [4, 6–9, 28, 29] are also useful

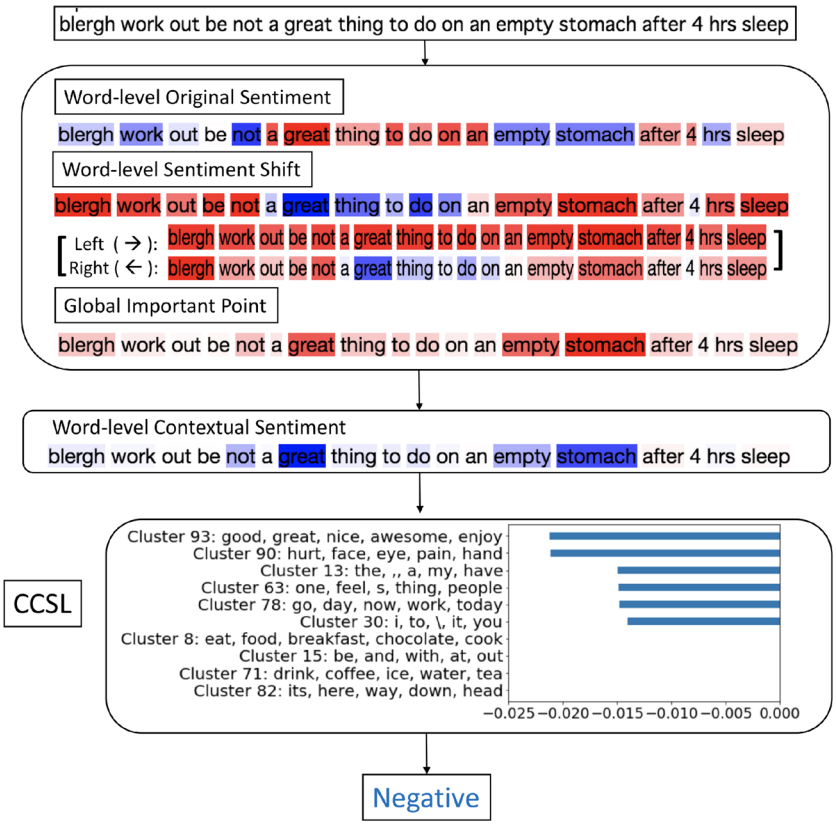

Fig. 4 Text-visualization Example for an English review in Sentiment 140. The color and depth of terms mean polarity (red: > 0 and blue: < 0) and scale of word-level sentiments in each layer

in these aspects. In this context, several methods developed a neural network including the layer that represents word-level original score [4, 8, 9]. Other methods developed a neural network including the layer that represents word-level global context using the attention mechanism [6, 7, 28, 29]. However, these previous methods do not satisfy our purpose because they alone cannot represent all the five types of scores, namely, word-level original sentiment score, word-level sentiment shift score, word-level global important point score, word-level contextual sentiment score, and concept-level contextual sentiment score in the explanation. In contrast, the proposed CSNN can explain the prediction results using the above five types of scores.

Many existing studies explored sentiment shift detection [2, 3, 26, 33, 34]. However, because most of these methods require specific knowledge of sentiment shifts, we cannot always use them in the real world. Unlike these methods, the CSNN can detect sentiment shifts without any specific knowledge on sentiment shifts. Although a method for detecting sentiment shifts without specific knowledge was developed in a previous study [27], the CSNN was better than this method in detecting sentiment shifts. Other studies dealt with assigning original sentiment scores to words using the sentiment tags of documents [8, 9, 25, 35]. The proposed CSNN outperformed them.

6 Conclusion

A novel NN architecture called CSNN that can explain its prediction process is proposed. To realize the explainability of CSNN, we proposed a novel learning strategy called IP learning. We experimentally demonstrated the effectiveness of IP learning for improving the explainability of CSNN. Using real textual datasets, we then experimentally demonstrated that the CSNN had higher predictability compared to that of some DNNs and that the explanation provided by the CSNN was sufficiently valid. In the future, we will apply this CSNN to documents pertaining to other domains or languages. Dataset, code, and the supplementary material are available[5].

[5] Available at bit.ly/CSNN20190606.

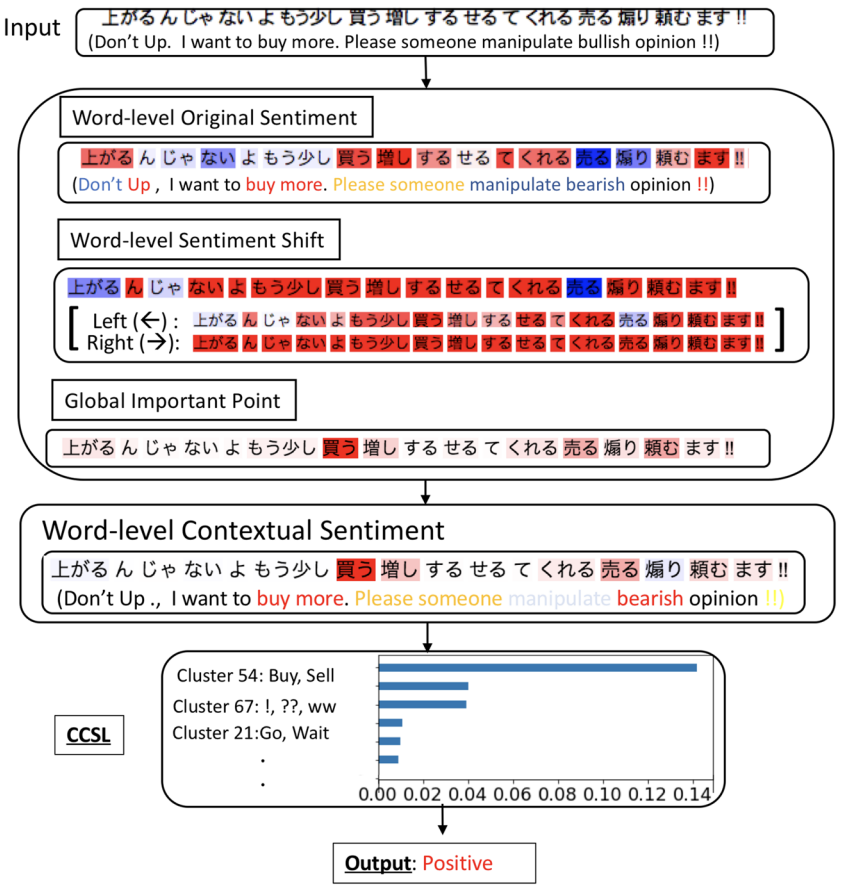

Fig. 5 Text-visualization Example for an Japanese review in Yahoo Review. The color and depth of terms mean polarity (red: > 0 and blue: < 0) and scale of word-level sentiments in each layer

Funding Funding was provided by Japan Society for the Promotion of Science (Grant No. JP17J04768).

References

1. Kim Y (2014) Convolutional neural networks for sentence classification. In: EMNLP 2014
2. Li S, Wang Z, Lee SYM, Huang C-R (2013) Sentiment classification with polarity shifting detection. IALP 2013:129–132
3. Schulder M, Wiegand M, Ruppenhofer J, Roth B (2017) Towards bootstrapping a polarity shifter lexicon using linguistic features. IJCNLP 2017:624–633
4. Ito, T, Sakaji, H, Tsubouchi, K, Izumi, K, Yamashita, T (2018) Text-visualizing neural network model: understanding online financial textual data. In: PAKDD 2018
5. Hutto C, Gilbert E (2014) Vader: a parsimonious rule-based model for sentiment analysis of social media text. In :ICWSM-14
6. Zou QZXHY, Gui T (2018) A lexicon-based supervised attention model for neural sentiment analysis. In: COLING 2018
7. Quanshi Z, Wu YN, Zhu SC (2018) Interpretable convolutional neural networks. In: CVPR 2018
8. Vo DT, Zhang Y (2016) Don't count, predict! an automatic approach to learning sentiment lexicons for short text. ACL 2016:219–224
9. Li Q (2017) Learning stock market sentiment lexicon and sentiment-oriented word vector from stocktwits. CoNLL 2017:301–310
10. Bach S, Binder A, Montavon G, Klauschen F, Muller KR, Samek W (2017) On pixel-wise explanations for nonlinear classifier decisions by layer-wise relevance propagation. PLoS ONE 10(7):1–46
11. Sundararajan M, Taly A, Yan Q (2017) Axiomatic attribution for deep networks. In: ICML
12. Karen S, Andrea V, Zisserman A (2013) Deep inside convolutional networks: visualising image classification models and saliency maps. arXiv:1312.6034
13. Hechtlinger Y (2016) Interpretation of prediction models using the input gradient. In: arXiv:1611.07634
14. Springenberg, JT, Dosovitskiy A, Brox T, Riedmiller MA (2015) Striving for simplicity: the all convolutional net. In: ICLR workshop

15. Schuster M, Paliwal K (1997) Bidirectional recurrent neural networks. IEEE Trans Signal Process 45(11):2673–2681
16. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Lu, Polosukhin I (2016) Attention is all you need. In: NIPS 2017
17. Wang W, Yang N, Wei F, Chang B, Zhou M (2017) Gated self-matching networks for reading comprehension and question answering. In: ACL 2017
18. Mikolov T, Sutskever I, Chen K, Corrado G, Dean J (2013) Distributed representations of words and phrases and their compositionality. In: NIPS 2013
19. Hornik MKK, Feinerer I, Buchta C (2012) Spherical k-means clustering. J Stat Softw 50(10):1–22
20. Nakov P, Rosenthal S, Kozareva, Stoyanov V, Ritter A, Wilson T (2013) Semeval-2013 task 2: sentiment analysis in twitter. In: SemEval 2013
21. Rosenthal S, Nakov P, Ritter A, Stoyanov V (2014) Semeval-2014 task 9: sentiment analysis in twitter. In: SemEval 2014
22. Zhao P, Zhang T (2014) Accelerating minibatch stochastic gradient descent using stratified sampling.arXiv:1405.3080v1
23. Kingma JLBDP (2014) Adam: a method for stochastic optimization. arXiv:1412.6980
24. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R (2014) Dropout: a simple way to prevent neural networks from overfitting. JMLR 15(1):1929–1958
25. Mohammad S, Kiritchenko S, Zhu XD (2013) NRC-Canada: building the state-of-the-art in sentiment analysis of tweets. In: SemEval-2013
26. Fancellu F, Lopez A, Webber B (2016) Neural networks for negation scope detection. In: ACL 2016
27. Li S, Yat S, Lee M, Chen Y, Huang CR, Wang G (2010) Sentiment classification and polarity shifting. COLING 2010:635–643
28. Yang Z, Yang D, Dyer C, He X, Smola A, Hovy E (2016) Hierarchical attention networks for document classification. In: NAACL 2016
29. Hu Q, Zhou J, Chen Q, He L (2018) SNNN: promoting word sentiment and negation in neural sentiment classification. In: AAAI 2018
30. Arras L, Montavon G, Muller KR, Samek W (2017) Explaining recurrent neural network predictions in sentiment analysis. In: EMNLP workshop
31. Ribeiro MT, Singh S, Guestrin C (2016) why should i trust you? Explaining the predictions of any classifier. In: KDD 2016
32. Shrikumar A, Greenside P, Kundaje A (2017) Learning important features through propagating activation differences. In: ICML
33. Wilson T, Wiebe J, Hoffman P (2005) Recognizing contextual polarity in phrase level sentiment analysis. EMNLP 2005:347–354
34. Kiritchenko S, Mohammad SM (2016) The effect of negators, modals, and degree adverbs on sentiment composition. NAACL-HLT 2016:43–52
35. Labille K, Alfarhood S, Gauch S (2016) Estimating sentiment via probability and information theory. KDIR 2016:121–129