Decision Support Systems 134 (2020) 113302

Contents lists available at ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss

Transparency and accountability in AI decision support: Explaining and visualizing convolutional neural networks for text information

Buomsoo Kim[a], Jinsoo Park[b], Jihae Suh[c,*]

[a] Eller College of Management, The University of Arizona, Tucson, AZ 85721, United States of America
[b] Business School, Seoul National University, Seoul 08826, Republic of Korea
[c] AI Institute, Seoul National University, Seoul 08826, Republic of Korea

ARTICLE INFO

Keywords:
Convolutional neural network
Machine learning interpretability
Class activation mapping
Explainable artificial intelligence

ABSTRACT

Proliferating applications of deep learning, along with the prevalence of large-scale text datasets, have revolutionized the natural language processing (NLP) field, thereby driving the recent explosive growth. Nevertheless, it is argued that state-of-the-art studies focus excessively on producing quantitative performances superior to existing models, by playing "the Kaggle game." Hence, the field requires more effort in solving new problems and proposing novel approaches and architectures. We claim that one of the promising and constructive efforts would be to design transparent and accountable artificial intelligence (AI) systems for text analytics. By doing so, we can enhance the applicability and problem-solving capacity of the system for real-world decision support. It is widely accepted that deep learning models demonstrate remarkable performances compared to existing algorithms. However, they are often criticized for being less interpretable, i.e., the "black box." In such cases, users tend to hesitate to utilize them for decision-making, especially in crucial tasks. Such complexity obstructs transparency and accountability of the overall system, potentially debilitating the deployment of decision support systems powered by AI. Furthermore, recent regulations are emphasizing fairness and transparency in algorithms to a greater extent, turning explanations more compulsory than voluntary. Thus, to enhance the transparency and accountability of the decision support system and preserve the capacity to model complex text data at the same time, we propose the *Explaining and Visualizing Convolutional neural networks for Text information* (EVCT) framework. By adopting and ameliorating cutting-edge methods in NLP and image processing, the EVCT framework provides a human-interpretable solution to the problem of text classification while minimizing information loss. Experimental results with large-scale, real-world datasets show that EVCT performs comparably to benchmark models, including widely used deep learning models. In addition, we provide instances of human-interpretable and relevant visualized explanations obtained from applying EVCT to the dataset and possible applications for real-world decision support.

1. Introduction

With the prevalence of big data and developments in related technologies, there has been considerable interest in computational linguistics and text analytics [1,2]. One of the critical driving forces that brought about such a trend is proliferating applications of deep learning in natural language processing (NLP). Reports indicate that more than seven out of ten contemporary work in the most prestigious NLP conferences propose or utilize deep learning models [3]. Furthermore, novel deep learning models are demonstrating breakthrough performances in various NLP tasks, e.g., machine translation, video captioning, and speech recognition [4]. Nevertheless, it has also been argued that state-of-the-art studies at the intersection of deep learning and NLP are more concerned to demonstrate superior quantitative performances over existing models, i.e., playing "the Kaggle game." Hence, it is imperative for the long-term growth of the field and progress if more effort is invested into new problems, approaches, and architectures [5].

We claim that one of the venues for a significant contribution to the field would be to design transparent and accountable artificial intelligence (AI) systems for text analytics. By doing so, we can enhance the applicability and problem-solving capacity of the system for real-world decision support. It is evident that cutting-edge deep learning algorithms, such as Convolutional Neural Network (CNN) or Recursive Neural Network (RNN), demonstrate outstanding performances compared to conventional machine learning (ML) algorithms, sometimes

* Corresponding author.
E-mail addresses: buomsookim@email.arizona.edu (B. Kim), jinsoo@snu.ac.kr (J. Park), jihaesuh77@snu.ac.kr (J. Suh).

https://doi.org/10.1016/j.dss.2020.113302
Received 6 July 2019; Received in revised form 10 April 2020; Accepted 10 April 2020
Available online 13 April 2020

super-human [6]. However, they are often criticized for being excessively complex and large, and thereby less transparent and interpretable – i.e., the "black box" algorithms [7,8]. In such cases, users tend not to trust the system for decision support [8,9]. Moreover, recent regulations are emphasizing fairness and transparency in algorithms, requiring explanations [10]. Finally, inherent biases and quality issues in big data [11–14] are intensifying the need to enhance the transparency and interpretability of underlying prediction models.

In this study, we propose a novel framework, *Explaining and Visualizing Convolutional neural networks for Text information* (EVCT), to effectively model non-linear text data and explain the results at the same time. Our framework efficiently reduces the dimensionality of the input space while considering the semantic structure of the sentence by obtaining low-rank subspaces [15]. By doing so, we can minimize the loss of information while enhancing the computational efficiency of the deep learning model and human-interpretability of results by adding sparsity constraints [16,17]. Also, we adopt a state-of-the-art method to explain CNN models for image data, i.e., the Class Activation Mapping (CAM) technique [18], to localize semantically salient tokens in the sentence and explain the prediction process. We demonstrate the applicability of our proposed framework with large-scale text datasets for sentiment prediction, automated essay scoring, and review mining. Our proposed model demonstrates a comparable performance to existing benchmark algorithms, including RNN, CNN, and contemporary sentence embedding [19]. Moreover, we demonstrate how we can enhance the transparency and accountability of the model by providing more human-interpretable and relevant explanations.

The rest of the paper is organized as follows. In Section 2, we describe the background of the study by reviewing relevant literature in (1) text analytics and deep learning and (2) explainable AI and decision-making. Section 3 presents the proposed method of this paper, i.e., the EVCT framework. Section 4 shows the analysis results using real-world text datasets. In addition, we show how the EVCT framework can be used to effectively interpret the prediction results and explanations that can be utilized for real-world decision support. Finally, the conclusion of this study, along with potential limitations and future work are discussed in Section 5.

2. Research background

2.1. Text analytics and deep learning

With the exponential growth in the amount of unstructured data and developments in information technologies to store and process such data, there has been a huge surge of interest in text analytics. According to the International Data Corporation, the size of the global data sphere is projected to grow astronomically, estimated to be 175 Zettabytes by 2025 compared to 33 Zettabytes in 2018. Moreover, the vast majority of the data generated are in unstructured formats, such as text, image, and video [20]. Fueled by the coincidental advances in deep learning during the last decade including CNN and RNN, deep neural networks have been increasingly adopted for computational linguistics [4]. Deep learning guru Geoffrey Hinton mentioned, "the most exciting areas over the next five years will be understanding text and videos" in 2014 [5]. Reflecting the trend, the proportion of deep learning papers in major NLP conferences has increased dramatically between 2012 and 2017, reaching around 70% [3]. That is, seven out of ten cutting-edge studies presented in the most prestigious NLP conferences propose or utilize deep learning models. However, at the same time, it has been claimed that recent studies inordinately focus on outperforming benchmark performances by playing "the Kaggle game." It would be desirable to invest more effort into problems, approaches, and architectures [5]. Therefore, we aim attention at proposing a novel method to outperform the state-of-the-art and enhance the applicability and problem-solving capacity of the method for real-world decision support. Accordingly, in this section, we review relevant deep learning methods to effectively model high-dimensional textual data used for real-world decision-making.

One of the distinctive characteristics of text data is extremely high-dimensional feature spaces. With the prevalence of extensive datasets, such tendency is more pronounced, wherein many new corpora have tens of thousands, sometimes millions, of unique tokens. To overcome the curse of dimensionality problem from large-scale data, embedding methods were proposed to express the distributed representation of words in a lower dimension. Word2vec [21] is the most established and widely used method for obtaining low-space word representations. Utilizing the Word2vec architecture, one can represent each token as an n-dimensional vector. Usually, n is set much smaller than the number of unique tokens in the corpus, e.g., 300. Inspired by the success of Word2vec, other approaches to represent text on a sentence, paragraph, or document level with a slightly tweaked neural network architecture were proposed. Nevertheless, they have not been as successful as Word2vec. An alternative approach to obtain efficient representation on a higher level is to utilize post-processing techniques for dimensionality reduction. There are largely two methods in reducing the dimensionality of the embedding space – row-wise reduction [15,19,22] and column-wise reduction [23,24] in the projection matrix. Column-wise reduction techniques further reduce the dimensionality of the embedding space (n) by eliminating the common mean vector [23] or by obtaining principal components [24]. Row-wise reduction techniques attempt to reduce the number of features by obtaining low-rank subspaces or fixed-dimensional representation, i.e., sentence embedding, for multiple words. It has been demonstrated that most information in a sentence can be captured by a low-dimensional subspace spanned by top-n principal components. On an average, top-3, top-4, and top-5 components capture 70%, 80%, and 90% of information, respectively [15]. By concatenating top-k components, a fixed-dimensional feature vector for a document can be obtained. It has been demonstrated that a simple averaging of word vectors in a sentence can outperform sophisticated models in NLP tasks [22]. Furthermore, Arora et al. [19] proposed a simple weighting scheme of word vectors in a sentence, followed by subtracting the first singular vector, i.e., the "common component." The weight of a word w is set to $\frac{a}{a + p(w)}$, where a is a hyperparameter and $p(w)$ is the estimated frequency of the word in the corpus. This simple smooth inverse frequency (SIF) method outperforms baselines by approximately 10–30%.

Though it has been rarely discussed in-depth, post-processing methods for word embedding can be highly relevant to decision support utilizing text data with large volumes. In general, large-scale text datasets have a substantial amount of redundancies and duplicated information. These, in turn, lead to computational inefficiency and opaqueness in the deployment of the results. Thus, we argue that by truncating mostly irrelevant information, the efficiency and effectiveness of analytics procedures can be significantly enhanced. However, state-of-the-art methods to reduce the number of features in the embedding space utilize principal component analysis (PCA) [25], which does not ensure human-interpretability. Within our EVCT framework, we propose a novel method to enhance interpretability and transparency by adding sparsity constraints and utilizing surrogate functions.

Another vital research stream in applications of deep learning models in NLP is concerned with exploiting the flexibility of deep neural networks in modeling non-linear unstructured data. Text data significantly differs from other types of data in that the information is context-dependent. There are two architectures to deal with such sequential characteristics of text information, i.e., RNN and CNN. RNN and its variants have a distinctive architecture from feedforward networks to preserve the memory of previous inputs, i.e., hidden states. RNN-based neural network structure revolutionized many crucial NLP tasks such as machine translation [26]. However, RNNs are not without shortcomings. The augmented model structure and additional parameters to keep and update hidden states lead to increased computation

costs. Furthermore, the training process of RNN is computationally expensive and difficult due to exploding and vanishing gradients - gradients of errors tend to be near zero (vanishing), or too large (exploding) [27].

An alternative approach to model high-dimensional, context-dependent text data is to adopt the CNN architecture, a widely accepted method in computer vision. CNN is a highly effective neural network architecture to detect and classify objects in an image, mimicking the mammalian visual system [28]. However, since text data have a dissimilar structure to image and video data, conventional CNN needs to be modified to model text data. Adapting CNN for text analysis has been popularized by recently proposed approaches [29] that are "radically different" from previous ones that exploit the flexibility of modern CNN architecture. Collobert et al. [30] set the foundation of CNN for sentence modeling that can be used for various NLP tasks such as part-of-speech tagging, semantic role labeling, and named entity recognition. Kim [29] proposed a slight variant to classify sentences, demonstrating impressive performances despite a simple structure with few parameters to train. Given a text input, a $p \times n$ matrix representation can be obtained with word embedding methods such as Word2vec. First, a maximal number of words to be included (p) has to be arbitrarily set; sentences having more than p words have to be trimmed and less than p words padded with zero entries. Then, in contrast to CNN for image classification, one-dimensional convolution operation and max-over-time pooling are performed. The maximum element of the vector is passed onto the next layer, which is connected to a fully connected layer. In this case, the final output is the calculated probability of each class, and the sentence is classified as one of the classes with the highest predicted probability.

We extend and improve the generic CNN architecture for classifying sentences [29,30] in the EVCT framework for a few reasons. CNN is a widely accepted method in both image processing and text analytics, showing a high-performance record in practice. It shows state-of-the-art performances in several NLP tasks such as sentiment analysis and speech recognition [29]. Besides, it has a flexible structure to model various types of unstructured data and allow efficient hardware implementation, adapted by major tech companies and many start-ups [4]. Finally, there have been many attempts to explain and visualize the inner workings of CNNs, which remained opaque for a long time. Many of them, e.g., [18,31–33], are highly successful in providing robust and reliable explanations of classification results, increasingly used for interpretable text mining [34]. Such explanations and visualizations can enhance scientific understanding, safety, ethics [7]. Accordingly, we review previous work in explainable AI in the following section.

2.2. Explainable AI and decision making

Although we covered recent developments mostly in text analytics so far, state-of-the-art deep neural networks show overwhelming performances in various tasks that have been regarded as insurmountable for the machine. Machines demonstrating super-human performance became no more of startling news – cases of world-class humans defeated by AI are more and more frequently reported. Sedol Lee, the 18-time world champion in Go, was defeated by DeepMind's AlphaGo in 2016 [35]. Two years later, AlphaZero, an AI software powered by a general reinforcement learning, defeated world champion programs in Go, chess, and shogi (Japanese chess) [6].

Deep learning is not, however, without limitations. Major criticism toward state-of-the-art complex deep learning models stems from the inherent opaqueness that hinders trust in the system. Nevertheless, trust is a critical issue in deploying big data-driven solutions for decision making in the wild [14]. Users are not willing to accept the solution if they do not have trust in the system. Therefore, making the model interpretable by providing appropriate explanations is a fundamental step in building trust and, ultimately, rendering the system more user-friendly and applicable [8,9]. Furthermore, recent legislative and regulatory trends are increasingly demanding fairness and transparency in algorithms, forcing explanations to be more mandatory than preferable. For instance, the General Data Protection Regulation, enacted in 2016 and started taking effect in 2018, entitles humans the right "to obtain an explanation of the decision reached after such assessment and to challenge the decision" [10]. Another fundamental reason for advocating transparency in algorithms in the context of real-world decision-making arises from inherent biases in data and data quality issues. Despite its vast potential to create enormous business value in diverse sectors, big datasets are often incomplete, incorrect, or outdated [11], and issues related to measurement and dependencies among data are becoming more substantial [36]. Poor data quality not only hinders good research in academia [12] but also has a considerable impact on organizations' decision-making quality in the real world [13,14]. One of the widely known cases of detecting a data quality issue by improving the transparency of the overall system utilized the Layerwise Relevance Propagation method that provides pixel-wise explanations for image classifiers [37]. The AI system designed to identify horses in images seemed to be highly successful in the task at a first glance. However, an ensuing attempt to explain and interpret the classifier revealed that its decision was primarily based on pixels at the bottom left corner of the horse images that contained a copyright tag [38]. In other words, the classifier was overfitted to implicit biases in data such that it won't be generalized to new image data without the copyright tags. Without the attempt to reverse-engineer the system by making it transparent, the users would likely have been complacent with the model's performance on hold-out data, and a significant pitfall might have gone unrecognized.

Consequently, explainable AI and ML interpretability are gathering a substantial level of interest among many practitioners and researchers. Explainable AI aims to build AI systems that can be understood, appropriately trusted, and effectively managed by humans [39]. One of the most critical reasons for advocating for explainable AI in real-life decision-making scenarios is that it can enhance the transparency and accountability of the decision support system. Fig. 1 is the explanation framework adapted from [39]. In general, a well-designed AI system can provide a reliable recommendation, decision, or action assumed by the target task, functioning as a decision support system. Nonetheless, if the system is tremendously large and too complex to be understood by humans, e.g., in the case of modern deep neural networks for large-scale text data, the lack of transparency and accountability in the decision-making process may arise. In such cases, explanations are fundamental in addressing the gap, thereby enhancing the interpretability of the overall system. In turn, more transparent and accountable prediction models can improve other crucial desiderata of decision support systems, including trust, safety, ethics, and fairness [7,8]. Several studies reported improved decision support with explainable systems, such as those predicting the remaining useful life of machinery [40], flight trajectory prediction and safety assessment [41], and risk assessment in cardiovascular diseases [42].

From the data-science lifecycle perspective, interpretable ML has three major desiderata in light of real-world decision-making – predictive accuracy, descriptive accuracy, and relevancy [43]. Each is related to different sources of error in different stages of the data science lifecycle. Predictive accuracy pertains to minimizing the error arising in the model stage in which prediction models are designed and constructed. In the supervised learning scenario, test accuracy on a hold-out set is a typical example of predictive accuracy. Descriptive accuracy, which is relevant to the post hoc analysis stage, refers to the interpretation capacity of the method to recognize the relationship previously learned by the ML model. In many cases, practitioners face the dilemma of maximizing either predictive accuracy or descriptive accuracy for the sake of one another. For instance, primitive ML models such as logistic regression and decision trees demonstrate high descriptive accuracy since they have a simple structure that allows for straightforward interpretations. However, they tend to show lower

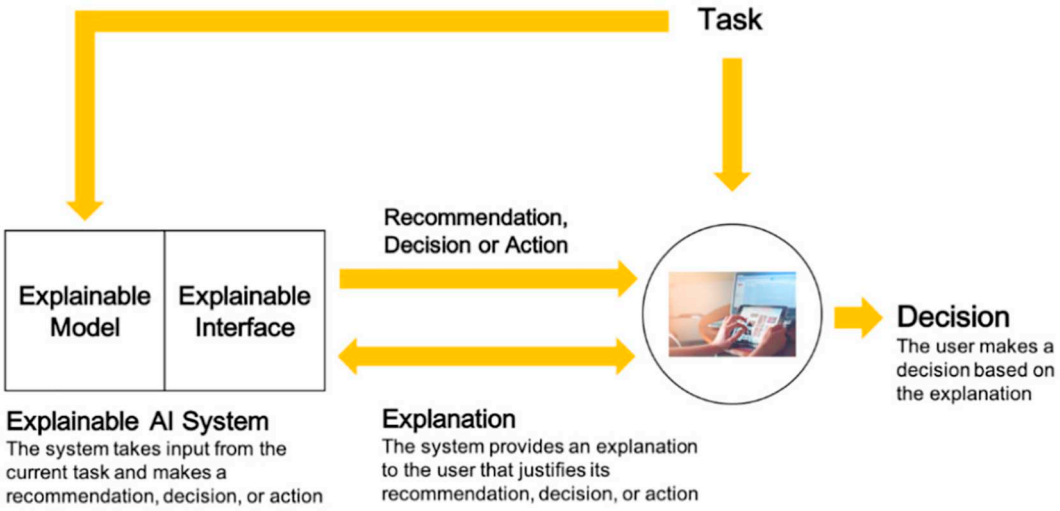

Fig. 1. Explainable AI Framework (adapted from [39]).

predictive accuracy compared to sophisticated models such as deep neural networks, especially from high-dimensional and complex data. Finally, relevancy is attained when an interpretation can provide meaningful insights into the problem of interest. Relevancy is critical in the deployment stage, considering various factors such as the balance between predictive and descriptive accuracy, end-user experience, and fairness and accountability of the model.

A constant effort to build transparent and explainable complex deep learning models is starting to show significant progress in the field. According to Doshi-Velez and Kim [7], there are more than 20,000 publications related to ML interpretability between 2012 and 2017. Given the vast amount of previous work, we restrict our attention to visualizing and understanding CNN in our study. As mentioned, modern CNN architecture shows significantly higher performance than conventional ML algorithms and are increasingly applied to solve real-world problems [28]. Hence, many researchers developed interested in explaining the algorithm. Since CNN is a suitable method for image recognition, attempts to visualize the network with tools such as saliency maps and pre-images emerged right from the outset [31,32]. The primary role of convolution operations in CNN is to extract features from grid-like data. By observing at which part of the instance CNN is focusing on, humans can understand the inner workings of the model on an abstract level. Based on the intuition, CNN is increasingly adopted for interpretable text mining, e.g., as described by Arras et al. [34]. Nevertheless, text data possess distinctive characteristics from image data, as mentioned in the previous section. Although CNN models for text and image present similarities, assumptions and intuitions differ to a certain extent [44]. As they share some common characteristics, understanding the similarities will enable us to benefit recent advances in understanding CNN. Meanwhile, recognizing the differences will let us deepen the knowledge and create improvements that are text data-specific.

We argue that among diverse methods, weakly-supervised object localization [45] methods can be effectively adopted for visualizing CNN for text classification. It is a promising method to obtain a saliency map to describe features that the network is focused on. By localizing a salient object that is correlated with a predicted label, one can understand where CNN is focusing on to make a prediction. In the text classification context, tokens can be regarded as objects and significant tokens, e.g., sentiment words or proper nouns, probably possess a strong correlation with the final output. Then, by localizing tokens that the model is focusing on while classifying an instance, we can have a grasp on how the model predicted that instance. This is similar to a human process of understanding language and making judgments. If we see a restaurant review saying "*the food is bland and service is terrible*," it would be natural to localize salient tokens that convey sentiment or appraisal, e.g., "*bland*" or "*terrible*".

CAM [18,33] is an established method in visualizing with object localization that utilizes the global average pooling (GAP) technique to detect features in the input space and weigh them internally. Each convolution filter generates a distinct feature map, or activation map, for the input image $f_u(x,y)$, which represents the activation of unit u at a spatial location (x,y). GAP generates the spatial average of each feature map, which can be considered as a scalar summary or representation of the feature map $f_u(x,y)$:

$$F^u = \sum_{x,y} f_u(x,y) \tag{1}$$

Then, the flattened vector obtained by GAP is taken into a consequent fully connected layer:

$$S_c = \sum_u w_u^c F^u \tag{2}$$

where $w_u{}^c$ is the weight of the fully connected layer, connecting to the final weight corresponding to class c. Finally, each S_c is normalized with a Soft-max function to calculate the probability for each class [18]:

$$o_c = \frac{exp(S_c)}{\sum_i exp(S_i)} \tag{3}$$

Here, the critical element of CAM is $w_u{}^c$, whose elements imply the importance of each feature map, i.e., $f_u(x,y)$. Thus, by combining each $w_u{}^c$ and $f_u(x,y)$ for all u while fixing c, we can obtain a class activation map, which is essentially an annotated feature map. Further, with a high localization ability and ease of interpretation, generalizations of CAM, e.g., grad-CAM [33], were proposed to enhance the applicability of CAM to diverse deep learning models. However, we identified potential problems of blindly applying CAM to text data without fully considering the unique characteristics of text data and the target task. Moreover, we recognized that recent work in the explainable AI and ML interpretability primarily focused on maximizing descriptive accuracy while retaining the level of predictive accuracy. Nevertheless, we argue that maximizing relevancy is also indispensable for model deployment and decision support. If the interpretation is not sufficiently relevant for real-world problem solving, users are unlikely to accept it despite high predictive/descriptive accuracy. We elaborate on this point in the method section.

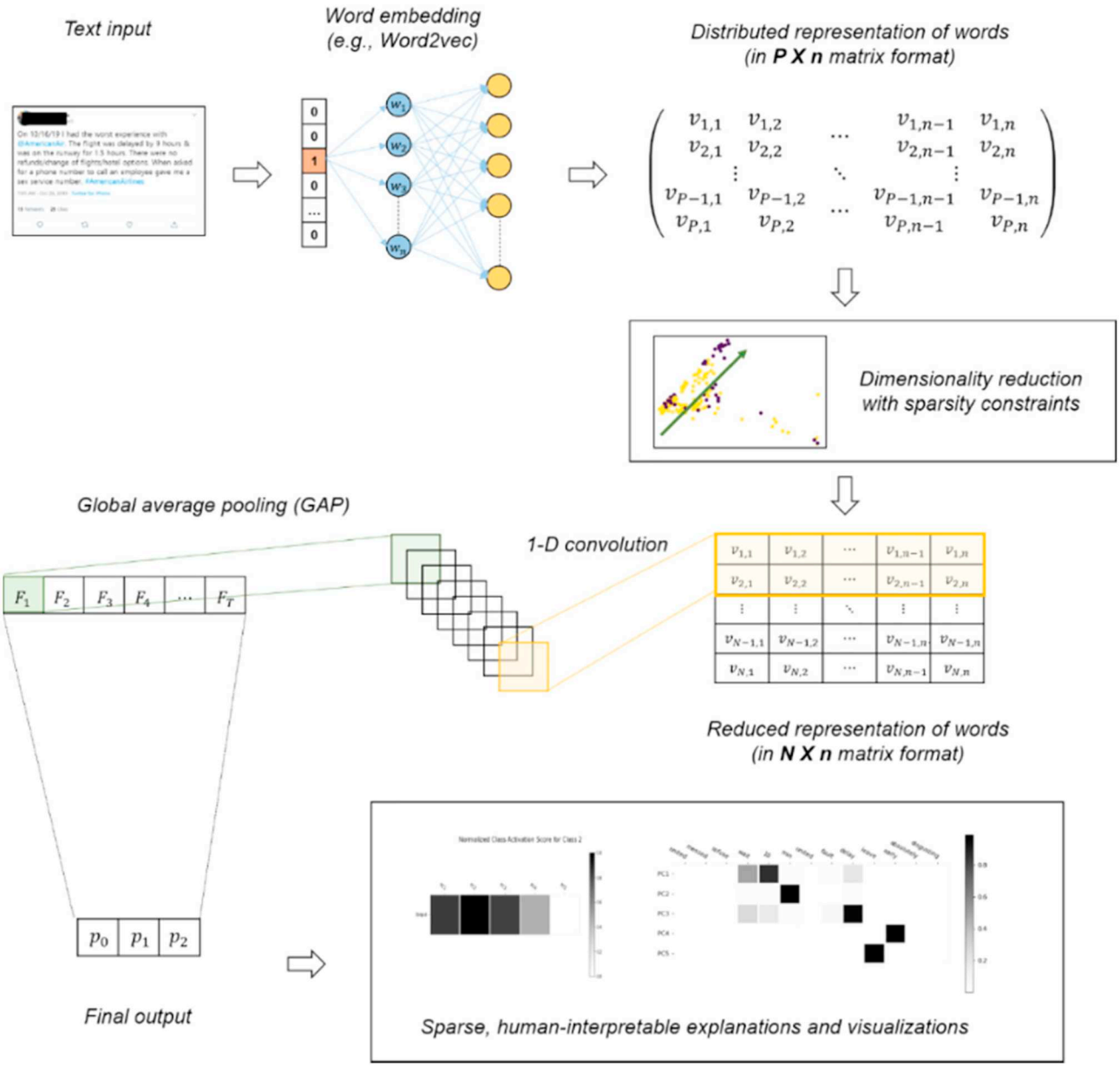

Fig. 2. Bird's-eye View of the EVCT Framework.

3. Method

In this section, we describe the proposed EVCT framework (Fig. 2). The framework encompasses state-of-the-art methods in both image processing [18,33] and NLP [15,21,29] while improving human-interpretability by imposing sparsity constraints. EVCT differs from existing methods since it (1) considers inherent characteristics of natural language, e.g., sequential information and low-rank subspaces, (2) provides human-interpretable explanations of the results by sparsity constraints, and (3) is generalizable to other decision support problems with text information such as sentiment prediction and document summarization.

First, we adapt and revamp the CAM method [18] to provide visualized explanations for deep CNNs for text classification. As mentioned, CAM is an effective method to localize objects in grid-format data. Similarly, we can identify essential elements of the sentences, e.g., words or phrases, by localizing a part of a sentence that is strongly correlated with the final output. To achieve such an objective, we construct a CNN model similar to the one proposed by Kim [29] that utilizes a one-dimensional convolution operation instead of a two-dimensional convolution operation for image data [28]. However, we replace max-over-time pooling with GAP to obtain a scalar summary of the feature map (F^u). Then, the concatenated F^u's are taken into a fully connected network to compute final outputs, e.g., probabilities for each class.

Another aspect of text data that should be adequately considered is a different format compared to that of image data in general. Data instances in most image datasets have a square or rectangular shape of fixed width, length, and depth [4]. For instance, all image instances in the CIFAR-10 dataset have the shape of 32 (width) - 32 (height) - 3 (depth). Moreover, it is relatively easy to reshape an image data instance without the loss of information. However, most instances in text data, e.g., sentences, paragraphs, or documents, differ drastically from each other in terms of token lengths. As inputs to CNNs should have equal lengths, a common approach to deal with varying instance lengths is to set the hyperparameter p to designate the maximum length of instances. Doing so will trim the instance if the length of the instance is larger than p and pad the instance with zero if the length is smaller

Table 1
Loading results with PCA and SPCA [16].

	PCA			SPCA		
Variables	PC1	PC2	PC3	PC1	PC2	PC3
V1	−0.404	0.218	−0.207	−0.477	N/A	N/A
V2	−0.406	0.186	−0.235	−0.476	N/A	N/A
V3	−0.124	0.541	0.141	N/A	0.785	N/A
V4	−0.173	0.456	0.352	N/A	0.620	N/A
V5	−0.057	−0.170	0.481	0.177	N/A	0.640
V6	−0.284	−0.041	0.475	N/A	N/A	0.589
V7	−0.400	−0.190	0.253	−0.250	N/A	0.492
V8	−0.294	−0.189	−0.243	−0.344	−0.021	N/A
V9	−0.357	0.017	−0.208	−0.416	N/A	N/A
V11	−0.379	−0.248	−0.119	−0.400	N/A	N/A
V12	0.011	0.205	−0.070	N/A	N/A	N/A
V13	0.115	0.343	0.092	N/A	0.013	N/A
V14	0.113	0.309	−0.326	N/A	N/A	−0.015
Variance	32.4	18.3	14.4	28.0	14.0	13.3

Note: 'N/A' indicates zero loadings

than p. After processing, one obtains a p by n matrix, where n corresponds to the dimensionality of representational space obtained by token embedding methods such as Word2vec [29].

For demonstration, assume that we have two tweets, **T1:** *"Just bought new iPhone!"* and **T2:** *"I definitely love the design of new iPhone 11, but do not want to spend a fortune on a device"* while setting $p = 5$. **T1** (4) is shorter than p, and **T2** (20) is considerably longer – two instances as inputs without processing would not work for CNN. Hence, a common approach entails padding the last three rows with zero for the input matrix corresponding to **T1** and trimming **T2** to *"I definitely love the design"* to fit into p by n matrix. A potential problem is that a significant loss of information may arise while processing questions much longer than p. In the example above, the information in the latter part of **T2**, i.e., "*but do not want to spend a fortune on a device",* is lost as a result of trimming. However, there might reside critical information in this part of the question in classifying this tweet. It is not difficult to recognize that the central message of this tweet resides in the latter part of the sentence, after "*but.*" Thus, when classifying this sentence as having positive or negative sentiment, the model (and possibly most humans) will judge the sentence as positive if only the first five tokens are provided. Nonetheless, the tweet conveys more of a negative message (iPhone is too expensive) than positive (the design is attractive).

We adapt the algorithm for low-rank sentence representation, i.e., row-wise word embedding reduction [15], to reduce the dimensionality of the sentence while minimizing the information loss. The algorithm represents a sentence as low-rank subspaces obtained from applying a dimensionality reduction technique, i.e., PCA, to vector representations of words in the sentence. After running PCA, vectors representing top-k components can be concatenated to represent a sentence. By doing so, we can set the dimensionality of features representing each sentence and minimize the loss of information from reducing it. However, though PCA is a transparent model per se, it does not guarantee human-interpretability. Even though principal components are linear combinations of input features, such linear mapping patterns can be less interpretable than expected. Especially, if we set the number of tokens (p) and/or the number of components (k) high, it would be more challenging to perceive which token is correlated with which component. If interpretations are not human-interpretable, they are less relevant in the data science lifecycle and unable to support real-world decision-making. A similar claim was made by Lage et al. [46]. A decision tree model with 5000 nodes is not human-interpretable while one with five nodes can be. That is, it is not the choice of the algorithm itself, but the complexity in the representation and the perception of the user that matters more in light of human-interpretability.

Algorithm 1. Procedure for obtaining sparse, low-rank sentence representation.

Input: a sentence s, word embeddings $v(\cdot)$ with dimensionality n, and Sparse Principal Component Analysis (SPCA) rank N.

Compute the first N principal components of samples $v(w')$, $w' \in s$,
$u_1, u_2, \ldots, u_N \leftarrow SPCA(v(w'), w' \in s)$,
$V \leftarrow (u_1, u_2, \ldots, u_N)$

Output: N by n matrix representing the sentence s

To improve the representation and maximize human-interpretability, we propose adding a sparsity constraint on the component loadings by replacing PCA with sparse principal component analysis (SPCA). SPCA imposes regularization penalty on the components to provide sparse loadings, i.e., derives a small set of components that are most efficient in reconstructing the input. Among many formulations of SPCA, we adopt the algorithm presented by Mairal et al. [17], which is a widely accepted algorithm for sparse coding. Given $N \times K$ data matrix $\boldsymbol{X}$, where N is the number of instances and K is the number of features, we attempt to find $\boldsymbol{U}$ and $\boldsymbol{V}$ such that.

$$\underset{U,V}{minimize} \frac{1}{2} \| X - UV \|_2^2 + \lambda \| V \|_1 \tag{4}$$

while satisfying $\| U_t \|_2 = 1$ for all $0 \le t < n_{components}$. Here, $\| V \|_1$ is the L1-norm of the matrix $\boldsymbol{V}$ and λ is the regularization parameter controlling the level of regularization for $\boldsymbol{V}$. As a result, the variance of SPCA is smaller than that of PCA in general, thereby enhancing the human-interpretability of findings [16]. Table 1 is the demonstration of loading results with PCA and SPCA [16]. It can be easily observed that the loading result with SPCA is more understandable and intuitive than with PCA. Algorithm 1 describes our improved procedure for obtaining sparse, low-rank sentence representations. Coefficients of such sparse combinations for loadings can be easily obtained by learning linear regression as a surrogate function, with input as word vectors and output as sparse components.

4. Analysis

4.1. Experimental settings

We evaluate and validate our framework with three large-scale public text datasets – airline Twitter sentiment, automated essay scoring, and LibraryThing reviews datasets. Twitter sentiment dataset is provided by Figure Eight (formerly Dolores Lab, Crowdflower), a human-in-the-loop ML and AI company. They provide crowdsourced datasets for various tasks, including image classification, linguistic relationships, drug relations, etc. In the airline sentiment dataset, 14,605 tweets posted during February 2015 regarding six US airline companies are scraped and manually annotated as positive, negative, or neutral. Among 14,605 tweets, there are 9169, 3082, 2354 tweets that are tagged as negative, neutral, and positive, respectively. After pre-processing to remove stopwords and training a Word2vec model [21], we randomly split the data into training and test datasets in a 7:3 ratio. Then, we train the proposed model in the EVCT framework and benchmark models for multi-class prediction – i.e., predicting each instance as neutral, positive, or negative. The benchmark models include (1) RNN with Gated Recurrent Unit cells [26], (2) CNN without post-processing [29], (3) conventional ML algorithms – logistic regression (LR), support vector machine (SVM), random forest (RF), and decision tree (DT) – with post-processed features by SIF, (4) LR, SVM, RF, DT models with post-processed features by PCA, (5) CNN with post-processed features by PCA, and (6) LR, SVM, RF, DT with post-processed features by SPCA.

The automated essay scoring dataset provided by the Hewlett Foundation comprises 12,978 essays written by students in grade levels between seven and ten. There are six types of essay sets in the training data, and each essay is hand-graded by two or three independent raters. As a reliable automated essay scoring system can provide an affordable

and scalable solution from a pedagogical and economic perspective, there have been many studies to predict ratings with the dataset, e.g., [47]. Since essays in different sets have diversified structures and various possible ranges for ratings and there are sufficient numbers of words in each essay, we only included 1800 essays in the second set. The maximum value of scores is 6.0 and the minimum is 1.0; the scores are normalized to fit into the range [0, 5]. Similar to the airline dataset, the essay scoring dataset is split into training and test datasets in a 7:3 ratio after training a Word2vec model, i.e., 1260 training data instances and 540 test instances. Then, the same benchmark models except Linear Regression (LinR) instead of LR, along with the proposed model, are trained and evaluated. LibraryThing.com is a website for cataloging information for a large number of books. It stores diverse metadata for books and user-generated reviews and rating information. Accordingly, the LibraryThing reviews dataset comprises user comments and ratings on books [48]. Though the dataset also includes social network information among users for the social recommendation, we focus on mining user comments to predict the ratings on each item in this study. Among 979,053 user-item interaction records each comprising the rating of the user on the item and review comment, we randomly sampled 50,000 instances for efficient training and evaluation. Then, the data are processed and analyzed similarly to the automated essay scoring dataset except that ratings lie in the range of [0, 5] though not normalized.

All in all, there are different types of data with short documents, long documents, and mixed-length documents in varying domains such as social media, automated scoring, and product cataloging in our pool of train/test datasets. Tweets are generally short, limited to less than 280 characters by Twitter. Conversely, essays written by students are generally longer than tweets with an average length of about 350 words. Reviews vary significantly in terms of lengths, the shortest one having just one word and the longest review with 3497 words among sampled instances. Therefore, we can more rigorously evaluate the proposed framework with various testbeds having different data sizes, tasks, and domains. In the following section, we illustrate the evaluation results.

4.2. Evaluation

This section evaluates the EVCT framework in terms of text classification performance and compares it with benchmark methods delineated in the previous section. We fixed the hyperparameters as summarized in Table 2. The sizes of word embedding and prediction window were set comparatively small since the documents and size of unique tokens are mid-sized, and we chose the skip-gram model architecture as it shows better performance in semantic tasks in general [21]. Finally, the number of principal components (k) is set to five. According to Mu et al., [15] approximately 90% of energy in a sentence can be captured by rank-5 subspaces on average.

We first compare the classification performances of different models in the sentiment prediction task with airline tweets. In multi-class classification, i.e., classifying each tweet as negative/positive/neutral – we report the average accuracy score and the balanced accuracy score to consider class imbalance. There are more negative tweets (9169) than neutral (3082) or positive (2354) ones in the dataset. The balanced average score is the average of recall on each class, which is claimed as superior generalizability than the average accuracy and cross-validation results [49]. Table 3 is a summary of the classification results. It appears that the proposed method in this paper, combining post-processing methods for word embeddings and one-dimensional convolution operation, bears superior results in both classification accuracy and balanced accuracy scores to benchmarks. Also, perhaps surprisingly, adding sparsity constraints to the post-processing method shows a superior result than the existing method. We conjecture that this is due to further suppressing the curse of dimensionality problem by reducing the number of features.

Table 3
Summary of multiclass prediction results.

Post-processing method	Prediction model	Accuracy	Balanced accuracy
None	RNN	0.6142	0.486
None	CNN	0.6909	0.591
SIF	LR	0.7445	0.6244
	SVM	0.7534	0.6271
	RF	0.7402	0.6048
	DT	0.6411	0.5553
PCA	LR	0.6597	0.4485
	SVM	0.72	0.5563
	RF	0.6643	0.5042
	DT	0.5716	0.4857
	CNN	0.7527	0.6365
SPCA	LR	0.665	0.48
	SVM	0.7092	0.5396
	RF	0.6677	0.5079
	DT	0.5734	0.4738
	CNN	**0.768**	**0.6479**

Note: Accuracy is rounded to the fourth digit after the decimal point.

Then, we compare the prediction performance of the automated essay scoring task. Since the output variable, i.e., the essay scores, is continuous, we compare the performance in terms of metrics for regression tasks – i.e., mean absolute error, median absolute error, and mean squared error. Table 4 is a summary of automated essay scoring prediction results. It seems that SIF is a superior choice than PCA and SPCA in this case, resulting in slightly better results. One plausible explanation for the better performance of SIF is a lower level of noise and variance in the documents. Compared to tweets and online reviews, essays are highly structured with refined language. In other words, documents are relatively similar to each other and there is not much inessential information. Therefore, the simple weight-and-average scheme is more effective than obtaining principal components and dropping a vast majority of information in a document.

Finally, we report the rating prediction results (Table 5). It appears to be a contrasting trend to the essay scoring results, and PCA-based methods perform superior to other methods. Comparing with the trend in Table 4, this implies a subtle yet essential insight into the nature of text data. We suspect that one of the probable reasons for such a trend is a significant variance in the length of documents – the longest review comprises 3497 words whereas the shortest one is only one word in the training data. In general, longer texts that are posted anonymously on the Internet have more superfluous and repetitious tokens compared to shorter ones. In such cases, removing unnecessary information by dimensionality reduction is prone to result in more efficient learning of indispensable patterns by the supervised model.

To summarize, combining a post-processing method for word

Table 2
Hyperparameter settings for evaluation.

Hyperparameter	Setting	Description
Size of word embedding	30	Dimensionality of the projected vector space
Word2vec architecture	Skip-gram	Choice of neural network architecture
Size of the prediction window	5	Ten words (five preceding and five succeeding ones) are predicted with the Skip-gram model
Number of principal components	5	Sentences are reduced to five principal components

Table 4
Summary of automated essay scoring prediction results.

Post-processing Method	Prediction model	Mean absolute error	Median absolute error	Mean squared error
None	RNN	0.6756	0.6549	0.6785
None	CNN	2.43	6.52	2
SIF	LinR	0.5182	0.4444	0.4403
	SVM	**0.4985**	**0.3989**	0.4207
	RF	0.5152	0.4202	0.45
	DT	0.5944	0.787	1
PCA	LinR	0.6366	0.6499	0.5183
	SVM	0.6639	0.6163	0.5612
	RF	0.5127	0.4172	**0.3999**
	DT	0.624	0.824	1
	CNN	0.5916	0.5411	0.553
SPCA	LinR	0.6515	0.6759	0.5236
	SVM	0.6639	0.6163	0.5612
	RF	0.5453	0.4765	0.5
	DT	0.6814	0.9444	1
	CNN	0.5206	0.4445	0.4339

Note: Errors are rounded to the fourth digit after the decimal point.

Table 5
Summary of rating prediction results.

Post-processing Method	Prediction model	Mean absolute error	Median absolute error	Mean squared error
None	RNN	0.7624	0.9898	0.8997
None	CNN	0.7624	0.9903	0.9
SIF	LinR	0.7271	0.8537	0.6649
	SVM	0.6952	0.8143	0.5993
	RF	0.6975	0.8052	0.605
	DT	0.9395	1.65	1
PCA	LinR	0.672	0.8225	0.5519
	SVM	**0.654**	0.8097	0.5308
	RF	0.6373	**0.7888**	0.49
	DT	0.8381	1.4851	1
	CNN	0.6723	0.8542	0.5819
SPCA	LinR	0.6824	0.8454	0.5716
	SVM	0.6536	0.8372	**0.5199**
	RF	0.6879	0.911	0.5499
	DT	0.8761	1.592	1
	CNN	**0.654**	0.854	0.526

Note: Errors are rounded to the fourth digit after the decimal point.

embedding and CNN deep learning architecture allows for comparable, if not superior, performances to existing approaches in sentence classification and embedding, e.g., [19,26,29]. We discovered when row-rank approximation can be more effective compared to average sentence embedding, i.e., modeling documents with widely differing lengths, and vice versa. Also, we observed that adding sparsity constraints to the post-processing method to obtain principal components of the projected representation space can further reduce the level of test error in some cases by suppressing the curse of dimensionality. In the next section, we demonstrate that adding sparsity constraints can enhance the human-interpretability of text information and the overall relevancy of the system as well.

4.3. Explanations and visualizations

Explanations and visualizations of the classification results with EVCT can be carried out hierarchically. First, class activation maps of k principal components ($k = 5$ in our case) can be obtained using the linear regression algorithm as a surrogate function for interpretability. Then, the word vectors comprising each component are back-tracked with the embedding matrix. In a sense, there is a hierarchy of explanations – identifying principal components for keyword extraction of sentences and visualizing activation maps to interpret the learning process of the CNN model.

Fig. 3 is an example of a visualized explanation provided by EVCT on a component level, with one of the sentences in the training dataset. The sentence is the 4001st instance in the airline dataset, i.e., tweet #4001, with original content "*@united it's messed up to refuse to wait 10 min after united is at fault for delay but to leave early is absolutely disgusting.*" The sentence is identified as negative (class label 2) with the probability of 0.989 and the true label is negative. That is, the model is correctly classifying the tweet as conveying negative sentiment. A saliency map for principal components of the sentence is provided: the darker the block, the more salient the role of the component in classification. The first three components, i.e., PC1, PC2, and PC3, are crucial in classifying the sentence as negative.

By observing visualizations such as illustrated in Fig. 3, one cannot know which individual words contributed to classification. Only the effect of principal components can be estimated. Hence, another saliency map can be created, which maps each token to different principal components (Fig. 4). This is obtained by applying the absolute value of the coefficients obtained by the linear regression function. Now, one could see which word has a significant influence on which principal

Fig. 3. Component-level visualization and explanation for tweet #4001.

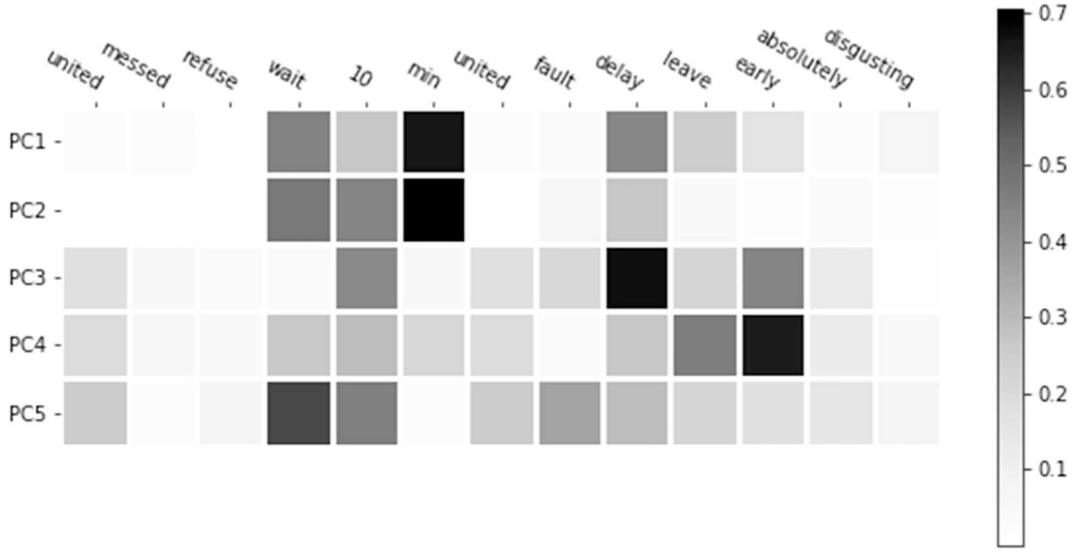

Fig. 4. Token-level Visualizations for Components of tweet #4001 obtained by PCA.

component. For instance, the token "min" is relevant in components one and two and "*delay*" in component three. Nevertheless, it is not easy to interpret the matching patterns between the components and tokens since the weights are widely distributed with a large variance. That is, the interpreter tends to be distracted by excessive information in the visualization that surpasses the information processing capability. Such information overload possibly leads to poor reasoning and ineffective decision-making.

Therefore, we propose to replace the visualization for component-token mapping with one obtained by SPCA. Fig. 5 is the token-level visualization of tweet #4001 obtained by training linear regression surrogate function with SPCA loadings. It is easier to interpret the results since the weights are less distributed. Tokens "*10*", "min", and "*delay*" are highly correlated with PC1, PC2, and PC3, respectively. According to the component-level explanations, the three principal components (PC1, PC2, and PC3) are crucial in determining the sentence as negative. Thus, we could explain that with a common-sensical explanation, the user is unhappy with a *10-minute delay* that occurred while using the United Airlines flight service. Such arguments meet the explanation evaluation criteria, which are coherence, simplicity, and generalizability [50]. In short, sparsity constraints can improve the quality of explanations and the ensuing decision-making process.

5. Conclusion

In this study, we proposed a novel framework for text classification while providing human-interpretable explanations and visualizations of results to enhance the transparency and accountability of the decision-making process. Our proposed EVCT framework integrates established methods from NLP [15,21,29] and computer vision [18,28] while improving the predictive performance and relevancy of results. We tested the applicability of EVCT using three large-scale public text datasets having distinct characteristics. Evaluation results reveal that the EVCT framework can effectively minimize the loss of information while enhancing the explanations for the prediction performed by the algorithm.

The present study has a few limitations. It can be computationally expensive to complete the three-step process of learning word embeddings, finding the sparse principal components, and training the supervised prediction model. With recent developments in large-scale computation such as Platform as a Service (PaaS) and cloud computing, we were able to carry out the analysis with a single commodity

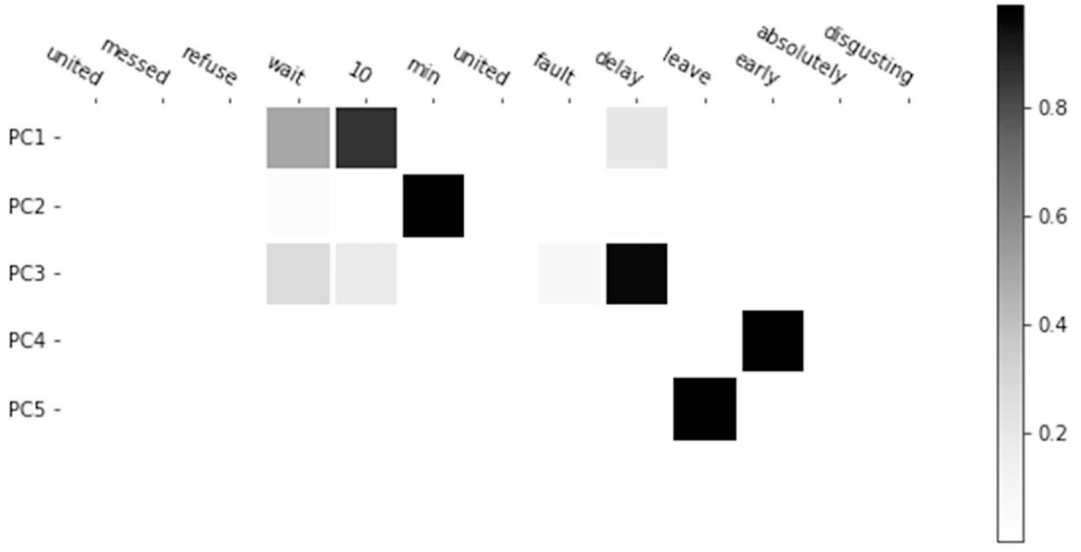

Fig. 5. Token-level Visualizations for Components of tweet #4001 obtained by SPCA.

machine. However, computational costs can increase exponentially for other text datasets with a much larger size. Especially in the context of real-world decision support, optimizing the overall process to minimize the burden of computation and obtain actionable results on time gets even more compelling. Therefore, future work can improve the training process of each step to minimize the burden of computation even in the presence of very large data. For instance, more recent word embedding methods than Word2vec with improved convergence time can be utilized and the optimization process of SPCA can be revamped for better computation and representation. Also, transfer learning, a widely accepted method for effective and efficient training of large-scale prediction models [51], can be utilized with massive external text data. It has been claimed that combining deep learning and transfer learning can bear promising results in decision support [52]. Another potential limitation of this study is that we were only able to provide intuitive evaluations of the visualized explanations and the following interpretations. Although objectively assessing the engendered explanations is a vital task, there is a lack of widely accepted standard methods for evaluating explanations in the field [53]. Some studies have relied on simulated experiments on human participants, e.g., [9,54], while others proposed quantitative metrics to evaluate synthetic explanations e.g., [53,55]. Nonetheless, we agree with Doshi-Velez and Kim [7] in that there is no one-size-fits-all solution to evaluating explanations. For instance, when evaluating explanation mechanisms in terms of sparsity, a model sparse in features can differ considerably from one sparse in prototypes. Also, the notion of human-interpretability can vary across end-users, depending on their ultimate purpose, data literacy, domain knowledge, etc. Therefore, we again emphasize maximizing the relevancy of the overall data-science lifecycle perspective [43]. Explanations should be evaluated in an application-driven manner while mindfully considering the ultimate purpose of the data-science project and end-users' expectations. Since our deep learning models for text analysis were intended for non-expert users in a low-stake situation, we focused on evaluating the explanations with generic, common-sense evaluative criteria for the explanations provided by Thagard [50]. Future work can extend the existing work, including our study exploring how the nature of a problem and possible applications can change appropriate evaluation methods. On top of widely used quantitative and experimental evaluation, rigorously assessing the feasibility and value of text explanations by rendering them as pragmatic solutions [56] would be an indispensable future direction for both practice and research.

The critical contribution of the paper in light of real-world decision support is two-fold: (1) improving the computational efficiency and representation capacity by effectively compressing the dimensionality of the feature space with minimal information loss and (2) enhancing human interpretability of post hoc explanations by sparsity constraints and surrogate functions. Therefore, our EVCT framework attempts to advance both the explainable model and interface in the explainable AI framework (Fig. 1). We argue that both are highly pivotal directions to improve the quality of decision-making in practice from a broader data-science lifecycle perspective [43]. Finally, the EVCT framework can be generalized to other text analytics problems beyond those covered in this study, e.g., fake news detection and question & answering. Future work can investigate the applicability of the EVCT framework on solving various other problems, at the same time improving and sophisticating the model architecture.

CRediT authorship contribution statement

Buomsoo Kim:Conceptualization, Methodology, Software, Writing - original draft, Software, Data curation, Visualization.**Jinsoo Park**:Supervision, Conceptualization, Validation.**Jihae Suh**:Supervision, Validation, Writing - review & editing, Investigation.

Acknowledgments

This study was supported by the Institute of Management Research at Seoul National University.

References

[1] A. Gandomi, M. Haider, Beyond the hype: big data concepts, methods, and analytics, Int. J. Inf. Manag. 35 (2015) 137–144.
[2] H. Chen, R.H.L. Chiang, V.C. Storey, Business intelligence and analytics: from big data to big impact, MIS Q. 36 (2012) 1165–1188.
[3] T. Young, D. Hazarika, S. Poria, E. Cambria, Recent trends in deep learning based natural language processing, IEEE Comput. Intell. Mag. 13 (2018) 55–75.
[4] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (2015) 436–444.
[5] C.D. Manning, Computational linguistics and deep learning, Computational Linguistics 41 (2015) 701–707.
[6] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, T. Lillicrap, K. Simonyan, D. Hassabis, A general reinforcement learning algorithm that masters chess, shogi, and go through self-play, Science 362 (2018) 1140–1144.
[7] F. Doshi-Velez, B. Kim, Considerations for Evaluation and Generalization in Interpretable Machine Learning, Explainable and Interpretable Models in Computer Vision and Machine Learning, Springer, 2018.
[8] Z.C. Lipton, The mythos of model interpretability, Proceedings of the 2016 ICML Workshop on Human Interpretability in Machine Learning (WHI 2016), New York, NY, 2016.
[9] M.T. Ribeiro, S. Singh, C. Guestrin, "Why Should I Trust You?": Explaining the Predictions of Any Classifier, Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, ACM, San Francisco, California, USA, 2016.
[10] European Parliament and Council of the European Union, Regulation (EU) 2016/679 of the European Parliament and of the Council, (2016).
[11] Executive Office of the President, Big Data: A Report on Algorithmic Systems, Opportunity, and Civil Rights, (2016).
[12] J.R. Marsden, D.E. Pingry, Numerical data quality in IS research and the implications for replication, Decis. Support. Syst. 115 (2018) A1–A7.
[13] M. Ghasemaghaei, G. Calic, Can big data improve firm decision quality? The role of data quality and data diagnosticity, Decis. Support. Syst. 120 (2019) 38–49.
[14] B. Baesens, R. Bapna, J.R. Marsden, J. Vanthienen, J.L. Zhao, Transformational issues of big data and analytics in networked business, MIS Q. 40 (2016) 807–818.
[15] J. Mu, S. Bhat, P. Viswanath, Representing sentences as low-rank subspaces, Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Vancouver, BC, Canada, 2017, pp. 629–634.
[16] H. Zou, L. Xue, A selective overview of sparse principal component analysis, Proc. IEEE 106 (2018) 1311–1320.
[17] J. Mairal, F. Bach, J. Ponce, G. Sapiro, Online dictionary learning for sparse coding, Proceedings of the 26th Annual International Conference on Machine Learning - ICML '09, ACM Press, Montreal, Quebec, Canada, 2009, pp. 1–8.
[18] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, A. Torralba, Learning deep features for discriminative localization, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Las Vegas, NV, USA, 2016, pp. 2921–2929.
[19] S. Arora, Y. Liang, T. Ma, A simple but tough-to-beat baseline for sentence embeddings, Proceedings of the 5th International Conference on Learning Representations, Toulon, France, 2017.
[20] D. Reinsel, J. Gantz, J. Rydning, The Digitization of the World from Edge to Core, IDC White Paper, US44413318 (2018), p. 28.
[21] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, Adv. Neural Inf. Proces. Syst. 26 (2013) 3111–3119.
[22] J. Wieting, M. Bansal, K. Gimpel, K. Livescu, Towards universal paraphrastic sentence embeddings, Proceedings of the 4th International Conference on Learning Representations, San Juan, Puerto Rico, 2016.
[23] J. Mu, S. Bhat, P. Viswanath, All-but-the-top: simple and effective postprocessing for word representations, 6th International Conference on Learning Representations, Vancouver, BC, Canada, 2018.
[24] V. Raunak, Simple and effective dimensionality reduction for word Embeddings, Advances in Neural Information Processing Systems 31, Long Beach, CA, US, 2017.
[25] S. Wold, K. Esbensen, P. Geladi, Principal component analysis, Chemom. Intell. Lab. Syst. 2 (1987) 37–52.
[26] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using RNN encoder-decoder for statistical machine translation, Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 2014.
[27] R. Pascanu, T. Mikolov, Y. Bengio, On the difficulty of training recurrent neural networks, Proceedings of the 31st International Conference on Machine Learning, Atlanta, GA, US, 2013.
[28] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, Adv. Neural Inf. Proces. Syst. 25 (2012) 1097–1105.
[29] Y. Kim, Convolutional neural networks for sentence classification, Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 2014, pp. 1746–1751.
[30] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, P. Kuksa, Natural language processing (almost) from scratch, J. Mach. Learn. Res. 12 (2011) 2493–2537.

[31] K. Simonyan, A. Vedaldi, A. Zisserman, Deep inside convolutional networks: Visualising image classification models and saliency maps, 2nd International Conference on Learning Representations Workshop Track Proceedings, Banff, AB, Canada, 2014.
[32] M.D. Zeiler, R. Fergus, Visualizing and understanding convolutional networks, Proceedings of the 13th European Conference on Computer Vision (ECCV), Zurich, Swiss, 2014, pp. 818–833.
[33] R.R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-CAM: Visual explanations from deep networks via gradient-based localization, 2017 IEEE International Conference on Computer Vision (ICCV), IEEE, Venice, 2017, pp. 618–626.
[34] L. Arras, F. Horn, G. Montavon, K.-R. Müller, W. Samek, "What is relevant in a text document?": an interpretable machine learning approach, PLoS One 12 (2017) e0181142.
[35] D. Silver, A. Huang, C.J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, D. Hassabis, Mastering the game of go with deep neural networks and tree search, Nature 529 (2016) 484–489.
[36] D. Boyd, K. Crawford, Critical questions for big data: provocations for a cultural, technological, and scholarly phenomenon, Inf. Commun. Soc. 15 (2012) 662–679.
[37] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, W. Samek, On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation, PLoS One 10 (2015).
[38] I. Sample, Computer Says No: Why Making AIs Fair, Accountable and Transparent Is Crucial, The Guardian, 2017.
[39] D. Gunning, Explainable Artificial Intelligence (XAI), (2017).
[40] M. Kraus, S. Feuerriegel, Forecasting remaining useful life: interpretable deep learning approach via variational Bayesian inferences, Decis. Support. Syst. 125 (2019) 113100.
[41] X. Zhang, S. Mahadevan, Bayesian neural networks for flight trajectory prediction and safety assessment, Decis. Support. Syst. 131 (2020) 113246.
[42] W.-Y. Hsu, A decision-making mechanism for assessing risk factor significance in cardiovascular diseases, Decis. Support. Syst. 115 (2018) 64–77.
[43] W.J. Murdoch, C. Singh, K. Kumbier, R. Abbasi-Asl, B. Yu, Interpretable machine learning: definitions, methods, and applications, Proc. Natl. Acad. Sci. 116 (2019) 22071–22080.
[44] A. Jacovi, O.S. Shalom, Y. Goldberg, Understanding convolutional neural networks for text classification, Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, Brussels, Belgium, 2018.
[45] M. Oquab, L. Bottou, I. Laptev, J. Sivic, Is object localization for free? - weakly-supervised learning with convolutional neural networks, 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Boston, MA, USA, 2015, pp. 685–694.
[46] I. Lage, A. Ross, S.J. Gershman, B. Kim, F. Doshi-Velez, Human-in-the-loop interpretability prior, Advances in Neural Information Processing Systems 31, Montreal, Canada, 2018, pp. 10180–10189.
[47] K. Taghipour, H.T. Ng, A neural approach to automated essay scoring, Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP), Austin, Texas, 2016, pp. 1882–1891.
[48] T. Zhao, J. McAuley, I. King, Improving latent factor models via personalized feature projection for one class recommendation, Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, Melbourne, Australia, 2015, pp. 821–830.
[49] K.H. Brodersen, C.S. Ong, K.E. Stephan, J.M. Buhmann, The balanced accuracy and its posterior distribution, Proceedings of the 20th International Conference on Pattern Recognition, IEEE, Istanbul, Turkey, 2010, pp. 3121–3124.
[50] P. Thagard, Explanatory coherence, Behav. Brain Sci. 12 (1989) 435.
[51] S.J. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 22 (2010) 1345–1359.
[52] M. Kraus, S. Feuerriegel, Decision support from financial disclosures with deep neural networks and transfer learning, Decis. Support. Syst. 104 (2017) 38–48.
[53] D. Nguyen, Comparing automatic and human evaluation of local explanations for text classification, Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2018, pp. 1069–1078 New Orleans, LA, US.
[54] I. Lage, E. Chen, J. He, M. Narayanan, B. Kim, S. Gershman, F. Doshi-Velez, An evaluation of the human-interpretability of explanation, Conference on Neural Information Processing Systems (NeurIPS) Workshop on Correcting and Critiquing Trends in Machine Learning, Montreal, Canada, 2018.
[55] S. Hooker, D. Erhan, P.-J. Kindermans, B. Kim, Evaluating Feature Importance Estimates, Proceedings of the 2018 ICML Workshop on Human Interpretability in Machine Learning (WHI 2018), 2018.
[56] J.F. Nunamaker, R.O. Briggs, D.C. Derrick, G. Schwabe, The last research mile: achieving both rigor and relevance in information systems research, J. Manag. Inf. Syst. 32 (2015) 10–47.

Buomsoo Kim is a Ph.D. student in Management Information Systems at Eller College of Management, University of Arizona. He received bachelor's degree and a master's degree in Business Administration from Seoul National University. His research interests include data mining, business intelligence, social networks, and database systems.

Jinsoo Park is Professor of MIS in the Business School at Seoul National University. He was formerly on the faculties of the Department of Information and Decision Sciences in the Carlson School of Management at the University of Minnesota and the Department of Management Information Systems in the College of Business Administration at Korea University. He received a PhD degree in Management Information Systems from the University of Arizona in 1999. His research interests are in the areas of ontology, semantic interoperability and metadata management in interorganizational information systems, schema matching, and data modeling. His research has been published in *MIS Quarterly*, *IEEE Transactions on Knowledge and Data Engineering* (TKDE), *IEEE Computer*, *ACM Transactions on Information Systems* (TOIS), *Data & Knowledge Engineering*, *Journal of Database Management*, and several other journals and conference proceedings.

Jihae Suh is the corresponding author of this paper and received a PhD. degree in Management Information Systems from the Seoul National University in 2017. She received her master degree from Carnegie Mellon University and bachelor's degree from Kyung Hee University. Currently, she is a research professor at Seoul National University AI Institute. Her research interests are in the areas of data mining, database system and social networks. Her research has been published in *Electronic Commerce Research and Applications*, *Data Base for Advances in Information Systems*, *Journal of Database Management*, and several other journals and conference proceedings.