Product-harm crisis intelligent warning system design based on fine-grained sentiment analysis of automobile complaints

Haiju Hu^{1,2} · Yonghui Wei^{3,4} · Yu Zhou³

Received: 16 October 2020 / Accepted: 16 February 2021
© The Author(s) 2021

Abstract

Due to the huge losses caused by product-harm crises and subsequent recalls in the automobile industry, companies must urgently design a product-harm crisis warning system. However, the designs of existing warning systems use the recurrent neural network algorithm, which suffers from gradient disappearance and gradient explosion issues. To compensate for these defects, this study uses a long and short-term memory algorithm to achieve a final prediction accuracy of 90%. This study contributes to the research and design of automatic crisis warning systems by considering sentiment and improving the accuracy of automobile product-harm crisis prediction.

Keywords Product-harm crisis · Quality complaints · Fine-grained sentiment analysis · Deep learning

Introduction

With the increasing market volume of cars, the problems produced by car use are also increasing. Consumer complaints directly describe the performance of cars, which can hurt a car brand and even produce a product-harm crisis, which can lead to huge financial losses [1, 2]. Therefore, many scholars consider post hoc strategies to manage the crisis, such as Hu et al. [1], while others have focused on crisis prevention, such as Hu et al. [2] and Zhang et al. [3]. It is critical to monitor and detect defects by analysing online complaints and warn companies of potential product-harm crisis events.

With the continuing maturation and development of artificial intelligence algorithms, artificial intelligence technology based on machine learning and deep learning has continued to grow to process natural language.

Technologies, such as machine translation and dialogue systems, have made great progress, which justifies the superiority of the deep learning model in natural language processing. Deep learning has been used in human behaviour research (e.g., Qin et al. [4] and Li et al. [5]), fault detection (e.g., Ouadine et al. [6]), and prediction and classification (e.g., Nireesh et al. [7] and Voican [8]). In 2013, Habibi et al. proposed weighted calculations of the influence probability transition matrix transmitted by adjacent words using various graph centrality indicators, such as degree centrality and proximity centrality in a word graph, to improve keyword extraction [9]. Munoz-Guijosa proposed a neural network training strategy when simulating train drivers' vibration [10]. Siu mined topic information and keywords by training a hidden Markov model and achieved ideal results on a test corpus [11]. By constructing a semantic network graph and fusing the graph path conversion amount, word clustering coefficient and word window information, Li proposed a comprehensive calculation index and achieved good performance with a patent data set [12]. Zhang proposed integrating the relevant information of the edges and points in a word graph in the calculation of the transition probability of TextRank to improve the effect of keyword extraction, which also provides the possibility for the application of public opinion monitoring [13]. With the continuous development of Internet technology, particularly the development and application of 5G technology, both the speed

✉ Yonghui Wei
wyh4919@ysu.edu.cn

¹ School of Economics and Management, Yanshan University, Qinhuangdao, Hebei, China

² Regional Economics Development Research Center, Yanshan University, Qinhuangdao, Hebei, China

³ LiRen College, Yanshan University, Qinhuangdao, Hebei, China

⁴ Graduate School, Mongolian National University, Ulan Bator, Mongolia

and effectiveness of communication have been markedly enhanced, and the self-media era has arrived. The fast propagation speed and wide range of new media thus pose more severe challenges to public opinion monitoring and processing. However, the traditional means of public opinion supervision are unable to meet the demands of the new complex public opinion environment in terms of processing speed, handling mechanism, and coverage. The idea of historical public opinion and deep learning algorithms provides a new way to understand public opinion and is an important application of artificial intelligence technology, particularly the data mining method, in public opinion analysis. For example, Tse et al. developed a comprehensive data analysis framework alongside a text-mining framework to objectively analyse social media data during a horse meat scandal and helped companies handle the crisis [14]. Derakhshan and Beigy introduced a more advanced method (part-of-speech graphical model) to mine public opinion to predict stock prices [15]. Yang et al. even used the text mining method to build an early warning system for adverse drug reactions [16].

In the automobile industry, certain companies have described that traditional product-harm crisis warning systems cannot adapt sufficiently to the rapid development of new Internet-based media technologies. Conversely, other companies still use traditional methods to predict and manage public crises, which allows the crisis to continue before being managed appropriately. If a crisis is not handled effectively or in a timely manner, the crisis will continue out of control and damaging public opinion about the company, damaging the corporate brand and perhaps yielding large tangible and intangible losses to the company [1]. Therefore, it is important to study public opinions in automobile enterprises and develop an effective product-harm crisis warning system. This system will help predict and manage automobile product-harm crises in new media situations and maintain brand image. However, traditional statistical methods or keyword filtering and extraction methods cannot accurately define the emotional tendency and importance, or link historical complaints together. Existing warning systems use the RNN algorithm, which exhibits gradient disappearance and gradient explosion. To mitigate these issues and consider sentiment, this study uses the LSTM algorithm to improve prediction accuracy.

The structure of this study is as follows. In “[Related work](#)”, we review related studies, focusing on the methods that are used in this study. “[Fine-grained sentiment model design and training](#)” introduces the process of fine-grained sentiment analysis, including the web crawler design; data cleaning and annotation; a dictionary of participle and generative words; keyword extraction and vectorization; and model training. The analysis results are presented in

“[Intelligent product-harm crisis warning system design](#)”, and “[Conclusion](#)” provides the conclusions of the study.

Related work

LSTM, a type of RNN, is used in this study due to the following reasons: (1) rating each sequence based on its likelihood of appearing in the real world provides a measure of grammatical and semantic correctness, and language models often exist as part of sentiment analysis; and (2) a language model can be used to generate new text (e.g., language training models based on an author’s publications can generate a new text of the author’s style). This study also generates forecast data for decision-makers.

In a traditional neural network, input and output data are independent of each other, which is unfavourable when predicting public opinion. However, LSTM can predict the next word in a sequence, as well as a sentence describing complaints. Before prediction, a word that often appears should be detected. Even though the complaint is not written by the same person, the content of the complaint and the related components can be predicted. For example, complaints about airbags and brakes are more related to consumer safety threats. Thus, predicting topics about personal safety will be more accurate.

An RNN network has three layers. Unlike ordinary back-propagation (BP) neural networks, the input layer has two parts: the hidden layer vector (h_{t-1}) at the previous moment, and the current input word vector (x_t). The most famous word vector method is word2vec, which was proposed by Mikolov et al. [17] in 2013. word2vec uses shallow neural networks for model learning and maps words into the corresponding high-dimensional space, and the result obtained is the word vector. The hidden layer (h_t) will also have two directions: one is used as the current output at the output layer, and h' is used as the input of the next step. The formulae at time t are shown as follows [18]:

$$h_t = f(ux_t + wh_{t-1}), \quad (1)$$

$$a_t = g(vh_t), \quad (2)$$

where f is the hidden layer activation function, which is typically a tanh function; and g is the output layer activation function, which is typically a softmax function. Therefore, the formulae are as follows [18]:

$$h_t = \tanh(ux_t + wh_{t-1}), \quad (3)$$

$$a_t = \text{soft max}(vh_t). \quad (4)$$

For the loss function, we use cross entropy in Zheng [18]:

$$\text{Loss} = - \sum_{i=1}^n y_t(i) \ln a_t(i), \quad (5)$$

where $y_t(i)$ represents the true value; $a_t(i)$ represents the output value; and n is the length of the output vector.

The key contribution of this study is to establish model parameters and user preferences. Training the LSTM model is similar to training a traditional neural network [19]: both use the back-propagation through time (BPTT) algorithm. However, because LSTM shares the same parameters u , v , and w at all times [11], the back-propagation algorithm should also be changed to some extent. Model training aims to find appropriate values for u , v and w .

BPTT expands the chain connection of the RNN, where each recurrent unit corresponds to a "layer", and each layer is calculated according to the BP framework of the feed-forward neural network. Considering the parameter sharing nature of recurrent neural networks [11], the gradient of the weight is the sum of the gradients of all layers [18]:

$$L = \sum_{t=1}^{\tau} L^{(t)} = \sum_{t=1}^{\tau} \log p(y^t x^1, \dots, x^t), \left| \frac{\partial L}{\partial L^{(t)}} = 1, \quad (6)$$

where x^t is the loss function [12]. In this study, the calculation steps of BPTT are introduced by taking the multi-output network connected by the recurrent unit as an example. First, the update equations are as follows [18]:

$$h^{(t)} = f(ux^{(t)} + wh^{(t-1)} + b), \quad (7)$$

$$o^{(t)} = v h^{(t)} + c, \quad (8)$$

$$\hat{y}^{(t)} = g(o^{(t)})cc. \quad (9)$$

The weights to be solved in the formulae are u , v , w , b , and c . For the last time step τ , and the remaining time steps $t \in [1, \tau - 1]$, the partial derivative of the total loss

function with respect to the state of the recurrent unit is a set of recursive calculations [20]:

$$\frac{\partial^+ L}{\partial h^{\tau}} = v^T \left(\frac{\partial^+ L}{\partial o^{(\tau)}} \right), \quad (10)$$

$$\frac{\partial^+ L}{\partial h^{(t)}} = w^T \left(\frac{\partial^+ L}{\partial h^{(t+1)}} \right) f'(h^{(t+1)}) + v^T \left(\frac{\partial^+ L}{\partial o^{(t)}} \right). \quad (11)$$

Because the RNN parameters are shared, they must be fixed to the change in the loss function of other time steps when calculating the gradient of the current time step. In this study, the parameters u , v , and w will be set and adjusted based on historical data.

LSTM also has certain shortcomings. We thus increased the prediction accuracy by stacking hidden layers and using a large amount of historical data. Finally, LSTM can accurately classify customer complaints and provide risk warnings based on the data in each time period.

Fine-grained sentiment model design and training

Due to the long-term dependencies problem of RNN [21] when learning sequences, RNN will tend to exhibit gradient vanishing and gradient explosion; thus, RNNs are unable to grasp the non-linear relationship of a long time span [22]. We thus use LSTM to perform self-circulation calculations on three internal units: input gate, forget gate, and output gate. The overall process is shown in Fig. 1.

Web crawler design

To ensure the representativeness of the complaint data, this study crawled data from the website <http://www.12365auto.com>, which is one of the largest public platforms for automobile users in China. The users of this platform are active

Fig. 1 Flow chart

and can report any problem they encounter when using a product. Python is used to crawl the data, and we strictly followed the rules to guarantee both the stability of the website and the update frequency. Additionally, we used the random update strategy to effectively avoid the shielding of anti-robots.

We used the `Requests.Session()` functions to access the interface of the website for data retention. The code is as follows: `page=requests.Session().get(url)`. Then, we acquired the data of the current URL using the `Xpath()` function. The code is as follows: `result=tree.xpath('//table//text())`.

Next, we stored the obtained data and designed a loop function to move between web pages. The specific process is as follows: first, based on the URL, we identify the transfer rules between web pages; then, we replaced the specific web page transfer addresses with variables and then traversed the entire website iteratively. 20,000 pieces of data were obtained in total. The data set structure is shown in Table 1.

Data preparation and segmentation

We manually annotated the 20,000 pieces of raw crawled data and determined the risk level of each complaint. In

dictionary to manage the stopwords and special characters. Punctuation marks were included in the stopword dictionary. The detailed command we used is as follows:

```
from jieba import analyse as ana
ana.set_stop_words('stopwords_path(txt)')
ana.extract_tags(txt)
```

Because a sentence is composed of words, it must be partitioned into several words before being analysed. There are many partition algorithms that can be used with Chinese words, and the general logic of the Chinese word partition system is shown in Fig. 2.

The tokenizer “Jieba tokenizer” is used to partition the crawled data. Based on the trie tree structure, the Jieba tokenizer can achieve efficient word map scanning and generate a directed acyclic graph (DAG) that is composed of all possible terms that the characters in the sentence may represent. After each participle, we decomposed the complaint text into multiple words. The partition code is as follow:

```
train_x,test_x,train_y, test_y=model_selection.train_test_split(data1.Comment.values.astype('U'),
data1.score.values.astype('U'), test_size=0.1,shuffle=True)
print(train_x.shape, test_x.shape)
print(train_y.shape, test_y.shape)
#Train_x Training set data Test_x Test Set data train_y Training set label test_y test set label

(3139) (349) (3139) (349)

all_data=list(train_x)+list(test_x)
print(all_data)
```

this paper, we labelled each complaint with one of four risk levels, where level 4 represents the highest level of risk, and level 1 represents the lowest level of risk. Compared to existing research, we innovatively considered sentiment when annotating the data. There are two aspects in each complaint that must be recorded: the quality of the product, and the emotion of the complaint. In certain cases, these two aspects are independent because the emotion of the complaint depends on the quality of the product and the attributes of the consumer (e.g., gender, education, personality). When complaining about the same problem, different consumers will express different emotions. Therefore, when we annotated the data, we considered both factors. Certain criteria to determine the risk level of a complaint are shown in Table 2.

Next, we cleaned the data by deleting null strings and lines with missing values. Then, we loaded the stopword

Keyword extraction and vectorization

The Keras framework is used for keyword extraction and vectorization. Keras is an open source artificial neural network library written in Python that can be used as a high-level application program interface for TensorFlow, Microsoft-CNTK and Theano to design, debug, evaluate, apply and visualize deep learning models [23–25]. Keywords can describe the theme or primary content of the text. Keyword extraction sentences were determined with `tokenizer.texts_to_sequences()`.

In this study, we used an analysis based on attributes and performed two tasks: attribute identification (i.e., mining the attributes related to the car quality in each complaint), and sentiment identification of the attributes. Attribute identification includes attribute classification identification and attribute extraction identification. We predefined certain

Table 1 Data-set structure

No	Name of the field
1	Complaint number
2	Car brand complained
3	Car series complained
4	Complaints models
5	Question briefly
6	Typical problem
7	Complaint time
8	Status of complaint

attributes, such as “engine failure”, “very angry”, and “air-bags do not work”. A complaint can mention none or several attributes, such as “engine”, “airbag”, “bad”, “garbage”, etc. Attribute extraction refers to the extraction of words or phrases related to attributes directly from the original text. For example, “start-stop function” and “flutter out” can be considered to be words related to the engine.

Existing attribute identification methods include traditional machine learning methods and deep learning-based methods. In this paper, we use a deep learning-based method to automatically extract attributes and the probabilistic topic model proposed by Maas et al. to perform polarity judgment [26].

After extracting keywords from the complaints, we used keyword vectorization, which creates digital data that can be analysed.

Next, an index is added to the partitioned words to generate a dictionary. Thus, each word is labelled by a unique

index number in the data set, and each complaint can be represented by numbers.

To facilitate the subsequent matrix standardization process, we searched for the longest sentence, which was composed of the largest number of words, and found that the longest complaint sentence contained 13 words. Because sentence lengths vary, we standardized them to 13. If the number of words is below 13 words, missing words are indicated by 0.

Model training

After standardization, we obtained a standardized data matrix of $N \times 13$. The training set is $N \times 80\% \times 13$, and the test set is $N \times 20\% \times 13$. As mentioned before, we used the LSTM method to avoid gradient disappearance and gradient descent. The specific model parameters are set as follows: epochs = 10, batch_size = 128, and validation_split = 0.2. Additionally, when selecting the optimal number of stack layers, we found that the accuracy does not always increase as the number of layers increases. If the number of layers is greater than 7, the accuracy does not increase further. Therefore, we set the number of hidden layers to 7 and preserved the unique vector value of h^t [27, 28]. For other parameters, we set vocabulary = 10,000, embedding = 32, word_num = 13, state_dim = 20, and return_sequences = False.

Finally, we achieved a good result with accuracy near 90%, and these results are shown in Figs. 3 and 4. To justify the effectiveness of the proposed algorithm, we also

Table 2 Certain criteria to determine the risk level of a complaint

Risk level	Criteria
4	The complaint is about engine, airbag, brake or other life-threatening problem; or the consumer is extremely unsatisfied about the response
3	The complaint is about car lights, engine oil, clutch, gearbox and other problems that may lead to safety problems; or the consumer is unsatisfied about the response
2	The complaint is about a large scratch, paint, peculiar smell, etc.; or certain problems remain after a repair
1	The complaint is about minor problems, such as noise, a small scratch, paint, etc.; or the complainant did not express any negative emotion

Fig. 2 Basic logic of word segmentation

conducted the same analysis using the support vector machine (SVM) method, which achieved an accuracy of approximately 70%. Therefore, the accuracy and loss of the proposed method indicate a good fitness to the data.

Intelligent product-harm crisis warning system design

Thus, we have trained an early warning model for product-harm crises for automobile companies, and the accuracy of the model was near 90%, which was able to support the detection work well.

Based on the model's training results, we can build an intelligent product-harm crisis warning system. For example,

we can design a crawl program that will crawl complaint data from a public platform each morning. Then, all complaints will be processed by the LSTM model trained in this study to automatically determine the risk level of each complaint and count the number of complaints in each risk level. When the number of complaints of a given risk level is greater than a threshold (see Table 3), the system will automatically send a warning to the monitoring personnel.

An intelligent warning system can be designed based on managers' preferences, such as the time step, criteria, threshold, etc. However, the model and algorithm developed in this study provide the core mode of the system and help to improve managerial decision effectiveness and efficiency.

Fig. 3 Train and validation accuracy

Fig. 4 Train and validation loss

Table 3 Threshold of the product-harm crisis warning system

Warning level	Criterion of each warning level	Warning colours
W4	Number of Level 4 risks > 400	Red
W3	0 < number of Level 4 risks ≤ 400 and number of Level 3 risks > 200 or number of Level 2 risks > 500	Blue
W2	Outside the midst of W3 and W4, and the number of Level 2 risk is ≤ 500	Orange
W1	Not belong to W2, W3 and W4	Green

Fig. 5 Fine-grained affective analysis results **Fig. 6** Complaint content extraction analysis word cloud display

Fine-grained sentiment analysis results

By analysing each complaint, we determined the risk level of each message according to the model and classified statistics based on each complaint. We identified the problems with the most complaints and display the results as bar charts and word clouds in Figs. 5 and 6.

In the word cloud diagram (Fig. 6), the product features with more complaints are shown in larger fonts.

Conclusion

To help companies effectively prevent the occurrence of product-harm crises and avoid associated negative effects, this paper proposes a product-harm crisis early warning system based on the fine-grained sentiment analysis of car complaints. This system performs self-circulation

calculations on the model's three internal units (the input gate, forget gate and output gate) in the LSTM network. The model achieves an accuracy of 90%, which is markedly higher than the accuracy of existing early-warning systems.

This study provides valuable information for automobile companies to design an automatic product-harm crisis warning system and reduce the monitoring cost and associated tangible and intangible costs, including logistic costs, damage to the company's reputation, and reduced consumer loyalty, that can be caused by a product-harm crisis.

The proposed method (1) accurately locates and classifies many complaints; (2) reduces required manpower and can monitor the risk situation in real-time; and (3) provides automatic early warnings based on user-defined thresholds.

However, this study does have limitations. Due to the limitations of existing voice- and image-data processing technologies, this study did not analyse these two types

of data. In future research, we plan to add more types of data to the dataset, such as voice complaints and image complaints, and apply the model to the field of voice and image car product-harm crisis warnings.

Acknowledgements This work is supported by National Natural Science Foundation of China (no. 71704151), Qinhuangdao Science and Technology Research Development and Planning Project (no. 201805A012), and Research funding of Hebei Key Research Institute of Humanities and Social Sciences at Universities (no. J201809).

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Hu H, Djebarni R, Zhao X, Xiao L, Flynn B (2017) Effect of different food recall strategies on consumers' reaction to different recall norms: a comparative study. *Ind Manag Data Syst* 117(9):2045–2063
2. Hu H, Wu Q, Zhang Z, Han S (2019) Effect of the manufacturer quality inspection policy on the supply chain decision-making and profits. *Adv Prod Eng Manag* 14(4):472–482
3. Zhang M, Hu H, Zhao X (2020) Developing product recall capability through supply chain quality management. *Int J Prod Econ* 229:1–13
4. Qin L, Yu N, Zhao D (2018) Applying the convolutional neural network deep learning technology to behavioural recognition in intelligent video. *Tehnicky vjesnik Tech Gazette* 25(2):528–535
5. Li J, Pan S, Huang L, Zhu X (2019) A machine learning based method for customer behavior prediction. *Tehnicky vjesnik Tech Gazette* 26(6):1670–1676
6. Ouadine A, Mjahed M, Ayad H, Kari A (2020) UAV quadrotor fault detection and isolation using artificial neural network and Hammerstein-Wiener model. *Stud Inform Control* 29(3):317–328
7. Nireesh J, Archana N, Anand R (2019) Optimisation of linear passive suspension system using MOPSO and design of predictive tool with artificial neural network. *Stud Inform Control* 28(1):105–110
8. Voican O (2020) Using data mining methods to solve classification problems in financial-banking institutions. *Econ Comput Econ Cybern Stud Res* 54(1):159–176
9. Habibi M, Popescu-belis A (2013) Diverse keyword extraction from conversations. In: *Proceedings of the 51st annual meeting of the association for computational linguistics*, vol 2, pp 651–657
10. Munoz-Guijosa J, Riesco E, Olmedo M (2017) Neural network and training strategy design for train drivers' vibration dose simulation. *Int J Simul Model* 16(1):72–83
11. Kollias D, Tagaris A, Stafylopatis A et al (2018) Deep neural architectures for prediction in healthcare. *Complex Intell Syst* 4:119–131
12. Li J, Lv X, Zhou S (2015) Patent keyword indexing based on weighted complex graph model. *N Technol Lib Inf Serv* 3:26–32
13. Zhang Y, Chang Y, Liu X (2017) Keyphrase extraction by integrating multidimensional information. In: *Proceedings of the 2017 ACM on conference on information and knowledge management*. ACM Press, New York, pp 1349–1358
14. Tse Y, Hoh H, Ding J, Zhang M (2018) An investigation of social media data during a product recall scandal. *Enterpr Inf Syst* 12(6):733–751
15. Derakhshan A, Beigy H (2019) Sentiment analysis on stock social media for stock price movement prediction. *Eng Appl Artif Intell* 85:569–578
16. Jia Y, Chen X, Yu J et al (2020) Speaker recognition based on characteristic spectrograms and an improved self-organizing feature map neural network. *Complex Intell Syst*. <https://doi.org/10.1007/s40747-020-00172-1>
17. Wang Y, Rong W, Zhang J et al (2020) Multi-turn dialogue-oriented pretrained question generation model. *Complex Intell Syst* 6:493–505
18. Zheng J (2017) *NLP Chinese natural language processing principle and practice*. Publishing House of Electronics Industry, Beijing
19. Zhang Q, Lu J, Jin Y (2020) Artificial intelligence in recommender systems. *Complex Intell Syst*. <https://doi.org/10.1007/s40747-020-00212-w>
20. Li X, Lu M, Zhao X, Ji X (2020) *Deep learning in natural language processing*. Tsinghua University Press, Beijing
21. Corriveau G, Guilbault R, Tahan A et al (2016) Bayesian network as an adaptive parameter setting approach for genetic algorithms. *Complex Intell Syst* 2:1–22
22. Yi S, Liu X (2020) Machine learning based customer sentiment analysis for recommending shoppers, shops based on customers' review. *Complex Intell Syst* 6:621–634
23. Moon K, Kim H (2019) Performance of deep learning in prediction of stock market volatility. *Econ Comput Econ Cybern Stud Res* 53(2):77–92
24. Ma K, Jiang B (2020) Voice of urban park visitors: exploring destination attributes influencing behavioural intentions through online review mining. *Complex Intell Syst*. <https://doi.org/10.1007/s40747-020-00223-7>
25. Boodhun N, Jayabalan M (2018) Risk prediction in life insurance industry using supervised learning algorithms. *Complex Intell Syst* 4:145–154
26. Maas J (2011) Gradient flows of the entropy for finite Markov chains. *J Funct Anal* 261(8):2250–2292
27. Siddiqi S, Sharan A (2015) Keyword and keyphrase extraction techniques: a literature review. *Int J Comput Appl* 109(2):18–22
28. Shankar K, Perumal E (2020) A novel hand-crafted with deep learning features based fusion model for COVID-19 diagnosis and classification using chest X-ray images. *Complex Intell Syst*. <https://doi.org/10.1007/s40747-020-00216-6>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.