Applied Intelligence (2023) 53:10789–10799
https://doi.org/10.1007/s10489-022-04063-5

Shooting review spam with a weakly supervised approach and a sentiment-distribution-oriented method

Jiandun Li[1] · **Liu Yang**[1] · **Pengpeng Zhang**[1]

Accepted: 2 August 2022 / Published online: 25 August 2022

Abstract

Untruthful opinions, ballot-stuffing or bad-mouthing online commodities, are challenging to identify because of two prominent obstacles, i.e., lack of ground-truth annotations and cracking deceptive sentiment along review contexts. To rise to these challenges, inspired by a recent algorithm called Learning with Label Noise, we first recruit volunteers to write annotated reviews and then label more unannotated public reviews with a neighborhood graph. Furthermore, based on statistical analysis, we introduce a Sentiment-Distribution-Oriented Clustering (SDOC) method to ferret review spam out, in which product usage aspects and their sentiment polarities are highlighted. Evaluations and comparisons with several state-of-the-art approaches indicate that SDOC is effective and outperforms them with statistical significance. We have also arrived at an interesting conclusion, i.e., genuine reviewers' feelings tend to fluctuate across different product aspects, whereas spammers always have uniform sentiments along aspects.

Keywords Review spam · Untruthful opinion · Sentiment distribution · Weakly supervised learning · Learning with label noise

1 Introduction

Driven by profit, deceptive opinions, also known as review spams, abound in almost every e-business platform and opinion sharing community. Technically, they can be shilling attacks towards reputations of their own merchandises as well as items out of their allies, or bad-mouthing manipulations targeting at fames of their opponents' commodities [1, 2]. Because of their destruction, product reputations are untrustworthy, causing both bad decisions for consumers and financial loss for producers. Ever since the pioneer work contributed by Ott and Liu [3], the field of review spam detection has been suffering from two key challenges, i.e., weak supervision and camouflage expose.The weak supervision problem talks about review sampling, and it is threefold, including incompleteness, inexactness and inaccuracy (See Fig. 1). Incompleteness means samplings are difficult to cover most spaces in review distribution, especially positive (i.e., spam) samplings. On one hand, writing styles greatly differ among people and semantic comprehension is still challenging; on the other hand, review spammers always adopt various camouflages for review manipulation. Inexactness denotes the recent case that user's sentiments across multiple aspects or modalities are challenging to comprehend, and probability-based models always arrive at biased judgements. Inaccuracy means that samplings inevitably have mislabeled examples. The other challenge, camouflage expose highlights the difficulty in cracking spammer's elaborately fabricated deceptive experiences. In fact, it is challenging even for domain experts. Take the following two aspect-oriented reviews for example, given the prior knowledge that only one is genuine. It turns out that people can hardly identify the spam out of them$*$. One can refer to the footnote for the answer.

> $*$ Review A: *The appearance is very good. I like the color, too. The photo effect is OK. The running speed was OK. There is still some noise in the voice, but it's already better than others. It's worth buying for students.*

✉ Jiandun Li
lijd@sdju.edu.cn

[1] School of Electronic and Information Engineering, Shanghai Dianji University, 201306 Shanghai, China

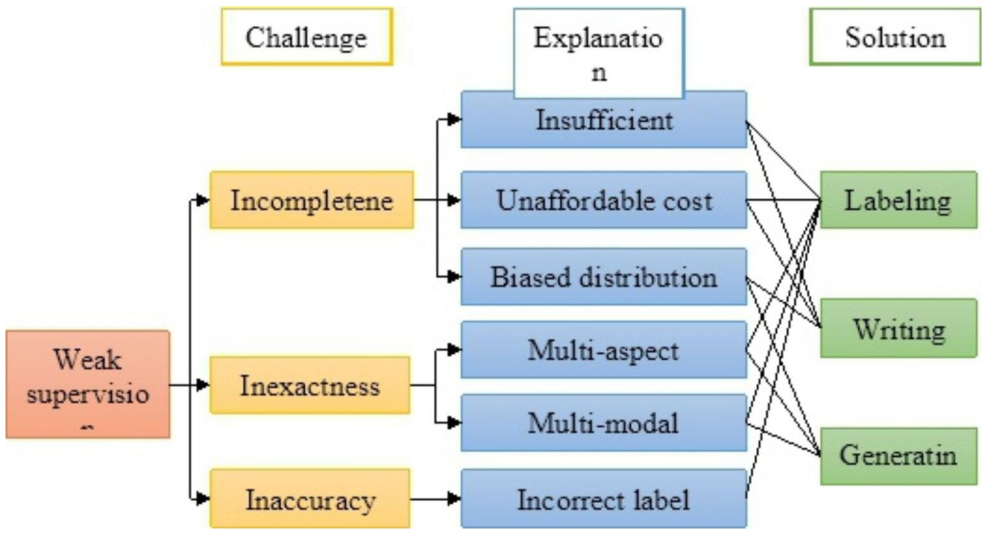


Fig. 1 The weak supervision problem of review spam detection

[1]Review B: *The appearance is so amazing. The photo effect is better than Apple's, and the price is very good. It's fast, it doesn't get stuck. The standby is superior to similar products. The magic laughter of the stereo video is very real.*

For the former challenge, state-of-the-art studies have applied crowdsourcing workers, human judges, commercial websites and Generative Adversarial Networks (GAN) to write, label, filter and generate annotated reviews; for the latter, linguistic attributes, behavior features and metadata have been introduced by previous researchers. However, these two challenges are still far from being conquered. Their shortcomings are depicted in Fig. 1. Inspired by the algorithm of Learning with Label Noise, in this paper, we propose a hybrid semi-supervised technique to expand the annotated review set and introduce a sentiment-distribution-oriented clustering (SDOC) method to identify review spam. The contribution of this paper is threefold: (1) we introduce a semi-supervised method to tackle the weak supervision problem based on Learning with Label Noise; (2) a dataset of annotated reviews is collected and released; and (3) we also propose an unsupervised approach SDOC to expose deceptive reviews with considerable performance.

The rest of the paper is organized as follows. We explore and discuss recent related work in Section 2, handle the weak supervision problem in Section 3 and propose the unsupervised approach SDOC to uncover review spam in Section 4. In Section 5, we validate our solution and compare it with several state-of-the-art solutions, and we conclude this paper in Section 6.

2 Related work

In this section, we survey some recent studies related the above two prominent challenges, i.e., the weak supervision problem and the identification of disguised sentiment.

2.1 To manage the weak supervision problem

For decades, targeting at the weak supervision problem, researchers have come up with multiple solutions including sampling from crowdsourcing workers, generating from software and applying filtering results of commercial e-business websites.

Sampling from crowdsourcing workers. Through recruiting Amazon Mechanical Turk, Ott et al. [4] collected and published a dataset of 800 reviews as respect to 20 hotels in Chicago, which is widely acknowledged as the gold standard dataset for this field. Furthermore, Li et al. [5] enriched this dataset including real-world genuine reviews and deceptive reviews written by filed experts, especially reviews on restaurants and hospitals. Oh et al. [6] requested 887 people to write deceptive and truthful opinions on randomly picked social issues. To make them sound authentic, short texts, i.e., less than 150 Korean characters, are excluded. There are also commercial datasets owned by Expedia, Hotels.com, Orbitz or Priceline. However, subsequent researches showed that crowdsourcing workers fail to represent online spammers, and some classifiers trained by these standard reviews could only achieve the accuracy of 52 ~ 54%. The reason behind this deficiency is still the incompleteness problem.

Filtering results of commercial websites. Business-to-customer websites or opinion sharing communities often have their own anti-spam models, like Amazon, Yelp,

[1] * Review B is spam.

Dianping, Meituan, Ele [7]. These commercial systems achieve high performance because of their affordable access to user's complete registered profiles. However, not only untruthful opinions, but also reviews with shallow experiences are evicted; therefore, these models cannot be applied directly to review spam detection.

Labeling reviews by human experts. Confronting the absence of ground-truth annotations, multiple researches have trained and validated their models using human judges, filed experts or just their working colleagues [8, 9]. Nevertheless, considering the camouflage, most of them are biased and unreliable; according to the Truth Bias in psychology and our previous study [1], people tend to overtrust anything.

Generating from software. Recently, GAN (Generative Adversarial Networks) has been introduced to this filed [10, 11]. Generally, with an annotated review input and random noise, researchers utilized the generators to automatically compose a spam, and then applied the discriminators to evaluate its quality. This attempt alleviates the incompleteness problem; however, since its unsupervised nature, the noise is difficult to customize to cover the specialty of review distribution.

Based on the analysis above, we can conclude that existing solutions have provided several solutions to this field, however, the weak supervision problem of annotated datasets still hindering us from understanding manipulation patterns and uncovering review shills. Inspired by recent progress in weakly supervised learning, we endeavor to collect some first-hand reviews from volunteers, and further label more public unannotated reviews through comparison.

2.2 To expose review spam

2.2.1 Reviews as free texts

Traditionally, a post of review texts is deemed as a sentiment atom towards the product. Because spammers have less product contacts, they always struggle to mimic actual experiences. To identify these deceptive sentiments, a lot of features have been introduced by existing studies to track them; wherein, duplication is widely adopted, which can obtain a great proportion of crowdsourcing or grouped spammers [3, 7]. Besides, researchers have introduced plenty of lexical features, e.g., the ratios of superlatives, exclamations, capitalizations, nouns, adjectives, prepositions, determiners and coordinated conjunctions, as indicators of the lack of using experience. Moreover, other researchers argued that the average number of syllables per word could also be utilized as an attribute [12]. Aggressively, Mukherjee [7] applied review length to evaluate a review's truthfulness. Also, Li et al. [1] stated that longer texts with less diversity or complexity might be spam attacks. However, these studies are biased and controversial; based on them, we cannot discriminate spam from the majority of innocent reviews, which are posted to make platform credits with naïve words and trivial phrases. Besides, an existing study found that family members or relatives are often talked about by spammers to as a disguise to conceal their shortage of experience [13]; however, this argument is also biased, because innocent reviewers might actually have bought the product for their relatives or friends. Wang et al. [14] argued that a review addressing multiple uncorrelated aspects is suspicious; however, this hypothesis is incapable of covering another spamming case where the review is independent of any domains. Recently, Asghar et al. [15] highlighted a bunch of text features, such as positive words, negative words, numeric, capital words, brand names, review length, and trained corresponding weights for these features to calculate the spamicity score. To sum up, except for duplication, these lexical facts are less dependable and cannot generate to recent scenarios, where user's feelings fluctuate along different aspects of the product. Based on these attributes, many Machine Learning models has been adopted, including Support Vector Machine, Naïve Bayes, Random Forest, etc. [16].

Recent years, several deep neural network models have also been introduced to this field [17, 18]. Through Amazon's reviews, Hajek et al. [19] completed the feature engineering through n-gram, word embedding and 30 sentiment attributes, and the models were deep feedforward network and Convolutional Neural Network (CNN). A similar model proposed by Ren et al. [20] trained a Gated Recurrent Neural Network (GRU), where Continuous Bag-of-Words (CBOW) was highlighted for vectorizing. Concentrating on Chinese reviews, Zhang et al. [21] portrayed a character from three sides, i.e., single character, pinyin, and pinyin vector, and made good use of CNN for classification. Recently, by embedding high-dimensional unigram-based word vectors into self-organizing graphs, Neisari et al. [17] transformed the review spam recognition problem into an image classification problem. Besides these deep models, attention techniques have also been adopted [22–24]. Remarkablly, Aghakhani et al. [25] introduced a revised GAN with duplex discriminators to ease the mod collapse problem in learning diverse distributions of truthful and spammed reviews. Together with Reinforcement Learning and Monte Carlo algorithm, the proposed semi-supervised model achieved great performance using TripAdvisor's review data. Instead of applying GAN as a review generator, Venkateswarlu et al. [26] also adopted it as a spam discriminator aided by other algorithms. In one word, these deep models contributed brilliant ideas to this field; nevertheless, compared to shallow models, deep neural networks heavily rely on much more annotated reviews, which cannot be met because of the weak supervision problem.

2.2.2 Attributes for reviews as aspect-oriented texts

Nowadays, supported by platform's nominated review topics, such as appearance, camera, battery, speed to a cell phone, consumer's reviews always have multiple aspects, explicitly or implicitly. For implicit ones, at least two steps are necessary to comprehend its sentiment, i.e., aspect discovery and sentiment measurement [27].

For aspect nominating, unsupervised approaches, e.g., LDA and its variants, seem more reliable and efficient than word frequency or grammar-based solutions. Li et al. [28] first introduced Latent Dirichlet Allocation (LDA) algorithm to this field. They argued that it is effective in telling the composing difference between truthful and untruthful reviews. Also using LDA, Lee et al. [29] exploited five topics based on part of speech to label shams. Overall, for latent topic-oriented long texts, these techniques are effective in labeling suspect reviews fabricated by professional spammers; yet, for recently nominated topics, they often give redundant results with time-consuming effort.

To measure the sentiment following an aspect, word-based methods use synonyms, antonyms or WordNet to evaluate each word's polarity and then apply an accumulative technique to output the final tag [30]. Its drawback is obvious where it cannot effectively manage the polysemy problem and contradict feelings. Recently, pretrained models have been borrowed to this field, such as Word2Vec, Bidirectional Encoder Representation from Transformers (BERT), Long Short-Term Memory network (LSTM), Recurrent Neural Network (RNN), Memory Network [31]. Nevertheless, since these models highly rely on data quantity, they achieved limited performance because of the weak supervision problem.

Recently, Xue et al. [32] abstracted four topics and then utilized Word2Vec to quantifying the difference between topics. They adopted sentiment's deviation over the average as the punishment and further broadcasted it into a "reviewer-review-sentiment" network to find more spammers. You et al. [30]applied lexical analysis to locate "aspect-sentiment" pairs and labeled untruthful reviews by distribution density. Our previous work [33] used dictionaries to recognize any latent aspects and their corresponding attitudes, and then took a clustering method together with the top two metrics, duplication and burstiness, to unveil spammer groups. Inspired by these studies, this paper first expands the review set, and then highlights the sentiment distribution as the key attribute to identify review spams.

3 Expanding the review dataset

To overcome the weak supervision problem, we propose a novel method by adapting existing algorithms (check Fig. 2 for the procedure). As respect to two cell phones, i.e., Huawei P30 and iPhone 11, we recruit volunteers to compose aspect-oriented reviews, including both authentic and deceptive ones via Wenjuan.com. Sentiments across six topics are collected ($p = 6$), e.g., appearance, camera, speed, battery, sound and others. Because of volunteer's devotion, we have collected a Chinese review set for each phone which have 2000 texts uniformly distributed across two dimensions, i.e., authentic/deceptive, positive/negative. One can access and make good use of this dataset via Github (https://github.com/smellydog521/chinese-review). Inspired by the Learning with Label Noise algorithm [34], we build a neighborhood graph to alleviate the weak supervision problem, which is comprised of four steps: selecting features, training the graph, identifying suspects and reversing them. After training, we input untagged texts and arrive at an expanded dataset of annotated reviews.

3.1 Building a supervised neighborhood graph

3.1.1 Selecting features

In order to vectorize an aspect-oriented review, we highlight the distribution of sentiment variations along six aspects, from which we can abstract the problem space as $R^p = R^6$. To quantify sentiment variations, we track the fluctuation of user's attitude within every aspect and across different aspects (see Algorithm I for detail).

Algorithm I: feature selecting

```
input: aspect_oriented_texts, adversative_conjunctions,
stopwords, positive_words, negative_words
foraspect, text in aspect_oriented_texts:
    Cut text into tokens
    iftokens intersect with adversative_conjunctions:
        variation [aspect] = 1
    end if
    Remove any tokens intersecting with stopwords and
    update tokens
    Set positive_count = number of tokens that intersecting with positive_words
    Set negative_count = number of tokens that intersecting with negative_words
    ifnegative_count ≥ positive_count:
        polarity [aspect] = 0
    else:
        polarity [aspect] = 1
    end if
    ifvariation [aspect] = 0 and negative_count × positive_count != 0:
        variation [aspect] = 1
    end if
end for
returnpolarity, variation
```

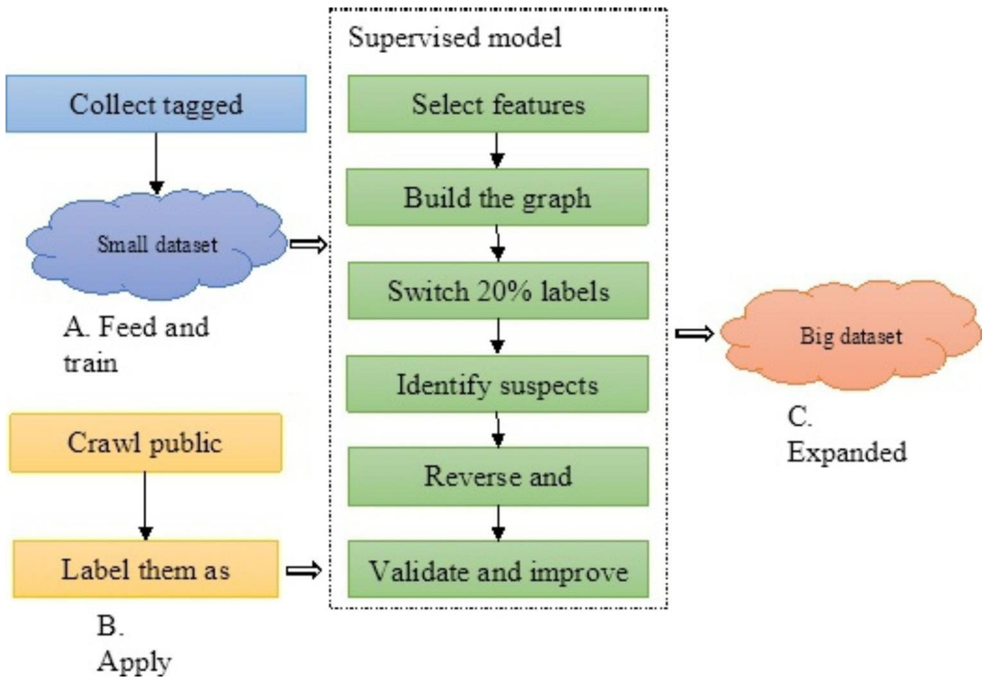


Fig. 2 Expanding the dataset

In Algorithm I, with the aspect-oriented texts, an adversative conjunction list, a stop word list, a positive word list and a negative word list as the input, reviews as multiple pairs of aspects and sentiments are parsed over a loop (line 1–17). For a review text, first we cut it into tokens (line 2). If any adversative words (e.g., "*but"*, "*however*" in Chinese) were found, the inner variation of the current aspect is deemed as positive (line 3–5). Then, we remove trivial words by referring to the list of stop words (line 6). Furthermore, we evaluate the sentiment polarity through two intersection operations (line 7–8), and the greater polarity is taken as this consumer's overall feeling (line 9–13) considering that people's average sentiment is always positive. Wherein two libraries of words together with their polarities, i.e., *positive_words* and *negative_words* are constructed by adapting existing studies [35]. Moreover, the variation for the current aspect is updated in line 14–16. It is inspired by the intuition that if an aspect-oriented review has both positive and negative words, its sentiment variation can't be overlooked. Finally, the vector with variation and polarity indices is returned from this algorithm (line 18). In this way, each aspect-oriented review is embedded into a 12-dimension vector with six polarities and six variations. Putting the six polarities together, we obtain the inter variation of this review.

3.1.2 Training a neighborhood graph

In this section, aided by annotated reviews, we construct a geometrical neighborhood graph to label more unannotated reviews inspired by Learning with Label Noise [34]. First, let's give the formal description of this graph.

Definition 1 (joint neighborhood): Assume u and v are two reviews in the real space R^{12}. Their joint neighborhood is defined as two hyper spheres' intersection, which take their Euclidian distance as the radius, u and v as centers respectively.

Definition 2 (neighborhood graph): Assume V is the review set in a real space R^{12} and the edge set E is composed of multiple review pairs (u, v). Reviews u and v are correlated if there is no review in their joint neighborhood, where their Euclidian distance serves as the weight. With V and E, we obtain a neighborhood graph G (V, E).

For a given review, it can be labeled as genuine or spam; therefore, edges in E can be categorized into two clusters, i.e., edges with uniformed or distinct annotations.

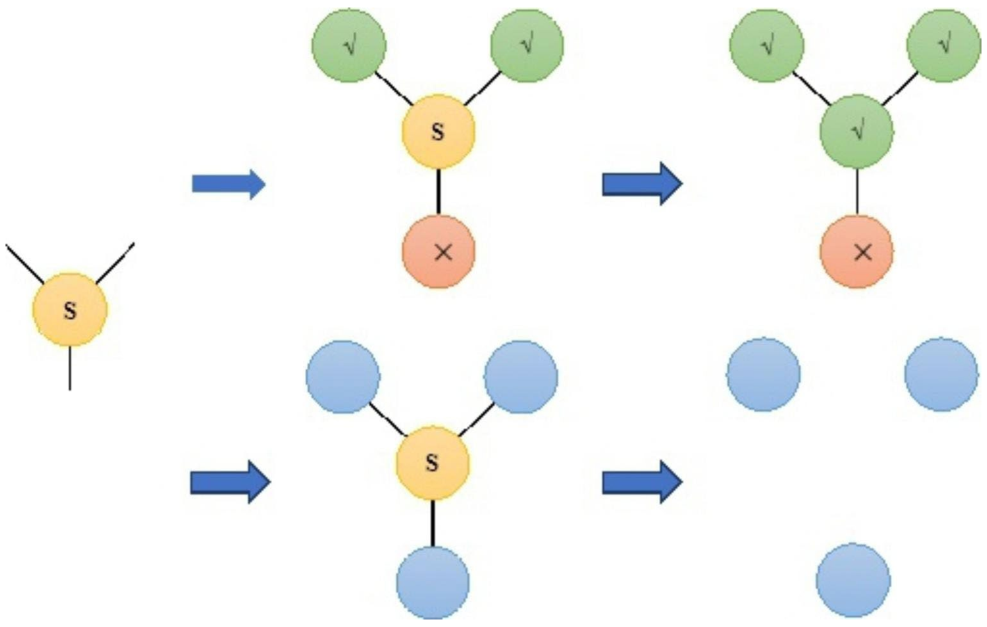


Fig. 3 Suspect instances. (S: suspect; √: truthful; ×: untruthful; no sign: any review)

Definition 3 (cut edge): Given an edge in E, we call it a cut edge if and only if labels out of its two nodes differ.

By breaking cut edges, we obtain subgraphs with all reviews labeled with the same annotation. Note that, there can be two or more subgraphs.

3.1.3 Identifying suspect reviews using cut edge weight

Cut edges bridge genuine reviews and spams together. From another viewpoint, we can label, validate a node's annotation or identify suspects via cut edges. Specifically, in order to identify suspect reviews, we choose to compute its cut edge weight and compare it to a threshold. Given a review node u and its annotation $y(u)$; it is considered as legitimate when the proportion of cut edges is significantly less than 1-$p(y(u))$, where $p(y(u))$ is the proportion of $y(u)$ in the dataset.

According to the null hypothesis that every review is independently written, the accumulative cut edge weight of review u can be evaluated by $W = \sum_{v=1}^{n_u} w_{uv} I_{uv}$, where n_u represents the total number of u's neighbors, v denotes one of its neighbors, w_{uv} stands for the distance, I_{uv} is the probability that u and v have distinct annotations with the expectation as 1-$p(y(u))$. Therefore, the expectation of W is $(1 - p(y\,(u)\,))\sum_{v=1}^{n_u} w_{uv}$.

In this paper, we randomly pick 20% reviews (according to Muhlenbach [34], referred as review set F) and switch their labels as F'. Based on the null hypothesis, we conduct a unilateral z-test and arrive at several p-values. Then we rank all reviews in the dataset according to their p-values and classify them into three categories by two thresholds θ_1 and θ_2, *secure*, *suspect* and *insecure*. To traversal the entire graph, we apply the breadth-first-search algorithm to inspect the cut edge weight one by one.

3.1.4 Reversing and removing

For *insecure* instances, we reverse their annotations. For a *suspect* instance (See Fig. 3), we reverse its label according to the flag of its most *secure* neighbors based on the majority vote decision; if there are no *secure* neighbors, we remove it from the graph.

3.1.5 Validation and improvement

We use F to validate this neighborhood graph. Five-fold cross validation is adopted along with many metrics including the confusion matrix, F1 and AUC (Area Under Curve). From these performances, we improve our settings on θ_1 = 0.1 and θ_2 = 0.9.

3.1.6 Time complexity

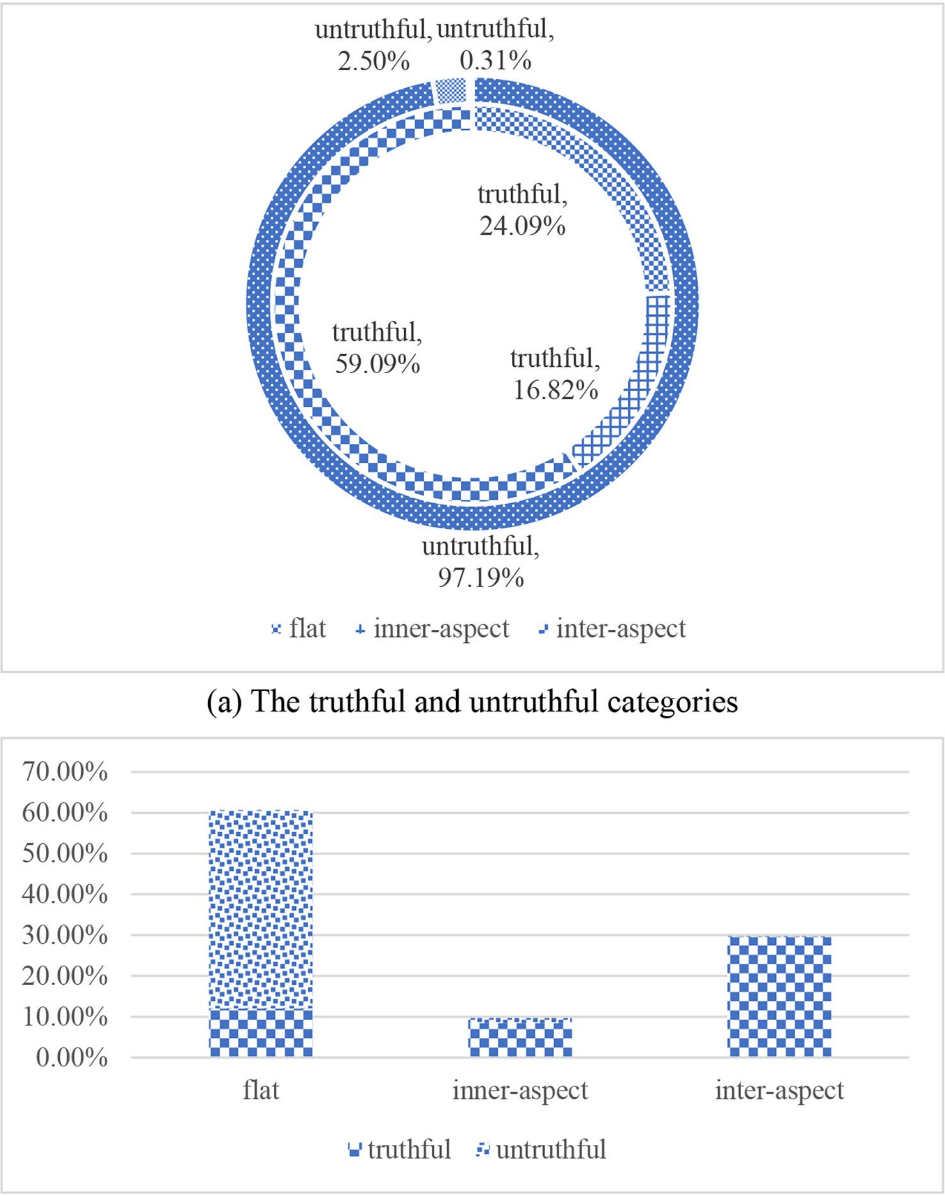
The most time-consuming phase of the instance expanding model is suspect recognition, which traversals across the entire graph and analyze every node's correlating edges. In another word, each edge is considered twice, so the time complexity is $O(\|V\|\cdot\|E\|)$.

3.2 Tagging more unannotated reviews

Facing the weak supervision problem of annotated reviews, in this section, we first crawl 5 million public reviews (unannotated) from the top two most influential e-business websites in China, i.e., JD.com and TMALL.com, with two products (Huawei P30 and iPhone 11) centered. According to existing statistics, at least 2/3 of these public reviews can be trusted [1]; therefore, we tag them all with trustful opinions initially. Then, these reviews are mapped into the real space R^{12} and further merged into the previously constructed neighborhood graph. After fitting, we have removed *suspect* instances, save *secure* ones and tag *insecure* ones as spam.

Table 1 Model settings

Model	Settings
NB	probability density: Gaussian
RF	criterion: Gini impurity
SVM	kernel: RBF; gamma: scale
DBA	similarity: Cosine distance
Bi-LSTM	layers: 2; hidden layer size: 128; dropout: 0.5; activation: Log Softmax



(a) The truthful and untruthful categories

(b) Integrated share

Fig. 4 Sentiment distribution

Table 2 Performances and comparisons

	accuracy	precision	recall	F1 score	Brier score	AUC in 95% CI
NB	0.68	0.69	0.65	0.68	0.32	0.68 ± 0.04
RF	0.81	0.82	0.78	0.81	0.19	0.81 ± 0.03
SVM	0.82	0.78	0.88	0.82	0.18	0.82 ± 0.03
DBA	0.84	0.81	0.88	0.83	0.16	0.84 ± 0.03
Bi-LSTM	0.85	0.86	0.84	0.85	0.15	0.85 ± 0.03
SDOC	0.87	0.80	0.97	0.88	0.13	0.87 ± 0.03

To further improve the performance of spam recognition, we balance truthful reviews and untruthful reviews as 100, 000 instances.

4 The SDOC approach

4.1 Exploratory data analysis

As respect to the expanded annotated dataset, we perform some statistical analyses (see Fig. 4 where "flat" represents unified sentiments without fluctuation). The result shows that innocent and spammed reviews differ a lot on sentiment distributions, no matter within commodity aspects (inner-aspect) or along them (inter-aspect). We take a hypothesis testing to verify this statistic. Since the review dimension of either subcategory is greater than 30, we assume it follows a Gaussian distribution and take the z-test. It turns out that the *p*-value is less than 0.05, so we argue that sentiment fluctuation can be adopted as an attribute in telling review spam.

4.2 Identifying review spam using inner- and inter-variation

Based on the statistical result, we propose a Sentiment-Distribution-Oriented Clustering (SDOC) approach to expose spammed reviews, in which only inner- and inter-variation attributes are adopted. The detailed pseudocode to discriminate whether a given review is spam is described in Algorithm II. First, we input the 12-dimentional vector of the review. If any sentiment fluctuation is recognized across aspect-oriented texts, the entire review is deemed as spam (line 1–3). Besides, this review cannot be accepted as innocent if there is deviation across sentiment polarities (line 4–6). If the review passes both examinations, we label it as an honest review (line 7). Because this algorithm only scans the review once, the time complexity is linear.

Table 3 The confusion matrix of SDOC

	Predicted-positive	Predicted-negative
Actual-positive	9725	275
Actual-negative	2414	7586

Algorithm II: identifying spam

```
input: polarity, variation
01 if any element of variation equals 1:
02     return "spam"
03 end if
04 if $\prod_{i=0}^{5} polarity\left[i\right] = 0$ and $\sum_{i=0}^{5} polarity\left[i\right] > 0$:
05     return "spam"
06 end if
07 return "truthful"
```

5 Evaluation and discussion

We conduct some experiments to validate our approach using Python v3.6.4 along with several packages such as *os*, *re*, *numpy*, *pandas*, *matlibplot* and *sklearn*. We apply *jieba* to cut Chinese texts into tokens. The list of stop words is adapted from previous studies, such as Harbin Institute of Technology, Sichuan University and Baidu.com. Sentiment polarity is evaluated using three dictionaries supported by HowNet and Tsinghua University.

Considering the state-of-the-art of review spam recognition, we enumerate five recent used models for performance comparison, including four supervised models, i.e., Naïve Bayes (NB), Random Forest (RF), Support Vector Machine (SVM) and Bidirectional Long Short-Term Memory network (Bi-LSTM), and an unsupervised model Duplication-Based Algorithm (DBA). For shallow models, we use TF-IDF to vectorize reviews. As respect to Bi-LSTM, a pre-trained Word2Vec model is adopted for embedding. Detailed settings and configurations for these models can be found in Table 1. After 10-fold cross-validation, their performances are depicted and compared in Tables 2 and 3; Figs. 5 and 6.

From these tables and figures, we can observe the performance rank sequence as NB < RF/SVM < DBA/Bi-LSTM < SDOC, where SVM and RF, as well as DBA and Bi-LSTM are neck to neck across multiple metrics, e.g., accuracy, precision, recall, F1 score, Brier score and AUC in 95% Confident Interval (CI). NB is inefficient mostly due to its conditional independence assumption that high dimensional token-oriented features are independent; yet, these tokens of Chinese words or phrases are correlated, not to mention aspect-sentiment pairs. Considering the considerable performance in labeling email spam, it has been

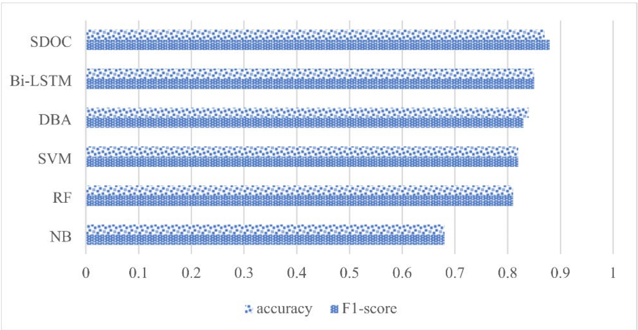


Fig. 5 Comparisons on accuracy and F1

verified again that review spam cannot be simply managed by existing solutions in other spam recognition fields, such as email spam or web spam [1]. Upon its widely adoption in text mining tasks, here we can see that NB fails to expose spammers' deceptive feelings. The decision tree model has the same problem, which affects the performance of RF's independent classifiers, although RF can further improve their efficiencies by ensemble. SVM is less efficient because selecting an appropriate kernel function is always a challenging job. Using the *Sklearn* module, we have selected different kernels, such as linear kernel, polynomial kernel, Radial Basis Function (RBF) and sigmoid kernel, and it manifests that all their performances are limited. These results above also verify the fact that generic shallow models aided by the bag-of-word vectorizing technique are less effective to explore the difference between innocent reviews and spammed reviews beneath Chinese languages.

By examining text duplication between posts, DBA have found most review spams. It echoes our previous studies on plagiarism-dominated manipulations [33, 36]. However, this approach falls short to unveil more spams because aspect-related traits are omitted. Bi-LSTM is pervasively applied in multiple text comprehension tasks. In this paper, for its capability in handling longer correlations between tokens, it achieves a better performance. However, since only token-level features are utilized, LSTM is also misled by the camouflages elaborately fabricated by professional spammers which are challenging to pick out from truthful experiences even for human judges.

By making good use of aspect-oriented reviews, especially sentiment fluctuations along aspects, the unsupervised method SDOC surpasses the other four models with significant margins. Instead of rushing into the fog of sincere sentiments accompanied by fake recommendations, it comprehends sentiment polarities and their distributions within/along aspects. In another word, SDOC avoids the challenge of truthfulness evaluation; meanwhile, it embeds token-wise higher dimensions to a lower dimension space, so the computing efficiency is greatly improved. As respect to the mislabeled examples tagged by SDOC (See Table 3), the number of false positive (2414) is much bigger than the number of false negative (275). This observation can also be found in Fig. 6, where SDOC's true positive rate

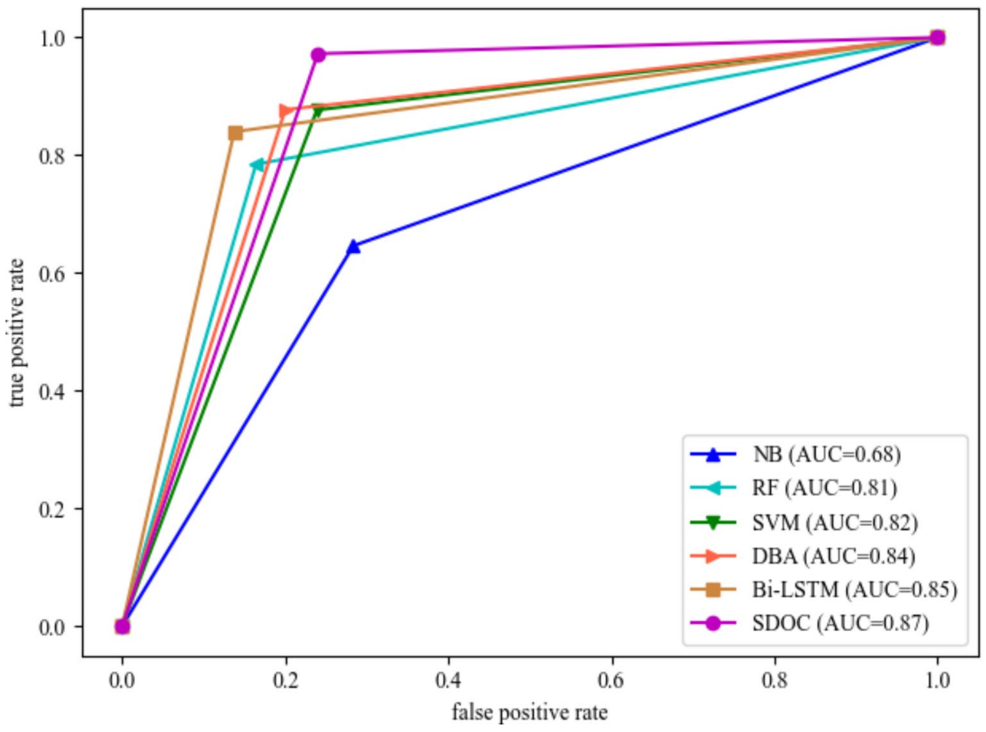


Fig. 6 Comparisons on ROC/AUC

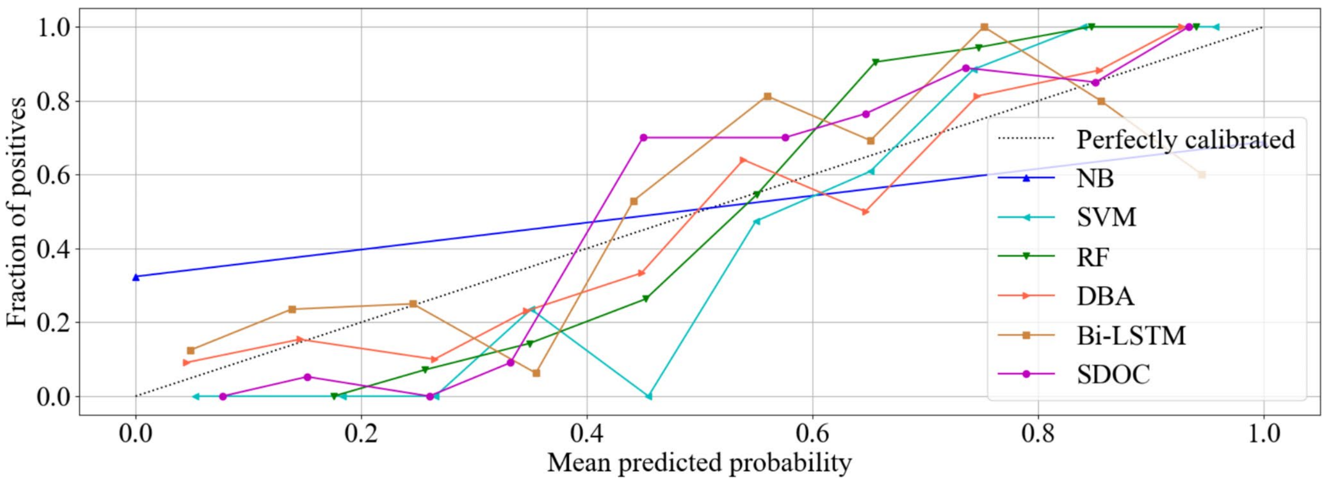


Fig. 7 Calibration curves

moves faster than the false positive rate. The success on true positive rate verifies that spammed reviews hardly fluctuate along review aspects; whereas the deficit on false positive rate is primarily because the fact that some genuine reviewers also have a unique sentiment within/across aspects, which is quite similar to spammers' behavior. A key difference between them is that truthful reviewers are chasing platform dollars, but spammers are after manipulation rewards. This phenomenon echoes the fact that frauds often attach great importance to manipulation efficiency [1].

After fitting, we take *Sklearn.CalibrationDisplay* to visualize calibrated probalities of these trained models. As illustrated in Fig. 7, we observe that most models ouput fine calibrated predictions with limited biases except NB. NB fails to be well calibrated and accumulates probabilites around 0 and 1, which is due to its aggressive assumption on attribute independence. This figure validates that we can trust the predicted results of SVM, RF, DBA, Bi-LSTM and SDOC and apply these models in real review spam dection scenarios with confidence.

6 Conclusion

The weak supervision problem has held back the field of review spam recognition for years. In this paper, by adapting the Learning with Label Noise algorithm, we expand the supervised dataset. Through this dataset, we propose a Sentiment-Distribution-Oriented Clustering (SDOC) solution to expose review spam. Performance comparisons with state-of-the-art models manifest that this method could avoid being fooled by most expert spammers and achieves a better performance. Based on experiment results, we argue that sentiment variance is effective and robust in recognizing review spam. Our conclusion is that innocent consumers often have fluctuating feelings about commodities; whereas spammers' sentiments are always biased and unified along different aspects.

A limitation of SDOC is that most of the time it could only works on review sets of long texts (with explicit/implicit review aspects), so short (trivial) spams are out of reach. For them, extra attributes other than texts, such as user's footprints, posting burstiness, the item's reputation curve, would help. For the future, we plan to follow the latest trend on product reviews and extend SDOC to a multi-modal scenario, including texts, photos and clips.

Acknowledgements This work was supported in part by the National Nature Science Foundation of China, under grant 61801285 and 61802247. We thank Kun Huang, Jingwen Lin, Yingsheng Wang and other anonymous volunteers for writing product reviews. We also appreciate anonymous reviewers for their constructive comments.

References

1. Li J, Wang X, Yang L, Zhang P, Yang D (2020) Identifying ground truth in opinion spam: an empirical survey based on review psychology. Appl Intell 50(11):3554–3569
2. Vidanagama DU, Silva TP, Karunananda AS (2019) Deceptive consumer review detection: a survey.Artificial Intelligence Review, :1–30
3. Jindal N, Liu BO, Spam (2008) and Analysis. Proceedings of the *International Conference on Web Search and Web Data Mining (WSDM 2008)*, of Conference, ACM Press, 219–230
4. Ott M, Choi Y, Cardie C, Hancock JT (2011) Finding Deceptive Opinion Spam by Any Stretch of the Imagination. Proceedings of the *the 49th annual meeting of the association for computational linguistics: Human language technologies (volume 1)*, Association for Computational Linguistics, 309–319
5. Liu Y, Pang B (2018) A unified framework for detecting author spamicity by modeling review deviation. Expert Syst Appl 112:148–155

6. Oh YW, Park CH (2021) Machine Cleaning of Online Opinion Spam: Developing a Machine-Learning Algorithm for Detecting Deceptive Comments. Am Behav Sci 65(2):389–403
7. Mukherjee A, Venkataraman V, Liu B, Glance NW (2013) Yelp Fake Review Filter Might Be Doing. Proceedings of the *the Seventh International AAAI Conference on Weblogs and Social Media* of Conference, AAAI, 409–418
8. Mukherjee A, Liu B, Wang J, Glance N, Jindal N, Detecting Group Review Spam. Proceedings of the Proceedings of the 20th international conference companion on World Wide Web (WWW 2011) (2011) of Conference, 93–94
9. Vrij A (2008) Detecting Lies and Deceit: Pitfalls and Opportunities. John Wiley & Sons
10. Tang X, Qian T, You Z (2020) Generating behavior features for cold-start spam review detection with adversarial learning. Inf Sci 526:274–288
11. Kabir HD, Khosravi A, Nahavandi S, Kavousi-Fard (2019) A.J.I.T.o.E.T.i.C.I. Partial adversarial training for neural network-based uncertainty quantification. 5:595–6064
12. Somayeh S et al (2013) Detecting Deceptive Reviews Using Lexical and Syntactic Features. Proceedings of the *13th International Conference on Intellient Systems Design and Applications*, of Conference, IEEE, 53–58
13. Anderson E, Simester D (2013) Deceptive reviews: the influential tail. Proceedings of the *Working paper, Sloan School of Management.*, of Conference, Northwestern University, 1–40
14. Wang Qianqian LB, Wenchang S, Zhaohui L, Wei S (2010) Detecting Spam Comments with Malicious Users' Behavioral Characteristics. Proceedings of the *ICITIS2010: 2010 IEEE International Conference on Information Theory and Information Security*, of Conference, IEEE, 563–567
15. Asghar MZ, Ullah A, Ahmad S, Khan A (2020) Opinion spam detection framework using hybrid classification scheme. Soft Comput 24(5):3475–3498
16. Tian Y, Mirzabagheri M, Tirandazi P, Bamakan SMH (2020) A non-convex semi-supervised approach to opinion spam detection by ramp-one class SVM. Inf Process Manag 57(6):102381
17. Neisari A, Rueda L, Saad S (2021) Spam review detection using self-organizing maps and convolutional neural networks. Computers & Security 106:102274
18. Fahfouh A, Riffi J, Adnane Mahraz M, Yahyaouy A, Tairi H (2020) PV-DAE: A hybrid model for deceptive opinion spam based on neural network architectures. Expert Syst Appl 157:113517
19. Hajek P, Barushka A, Munk M (2020) Fake consumer review detection using deep neural networks integrating word embeddings and emotion mining. Neural Comput Appl 32(23):17259–17274
20. Ren Y, Ji D (2017) Neural networks for deceptive opinion spam detection: An empirical study. Inf Sci 385:213–224
21. Zhang F, Qiu L, Qi P, Luo HA (2020) Novel Text Features Jointing Model for Review Spam Filtering of Chinese. Proceedings of the *International Wireless Communications and Mobile Computing (IWCMC)*, 2020, IEEE, 2051–2056
22. Li L, Qin B, Ren W, Liu T (2017) Document representation and feature combination for deceptive spam review detection. Neurocomputing 254:33–41
23. Bhuvaneshwari P, Rao AN, Robinson YH (2021) Spam review detection using self attention based CNN and bi-directional LSTM. Multimedia Tools and Applications 80(12):18107–18124
24. Gao Y, Gong M, Xie Y, Qin AK (2021) An Attention-Based Unsupervised Adversarial Model for Movie Review Spam Detection. IEEE Trans Multimedia 23:784–796
25. Aghakhani H, Machiry A, Nilizadeh S, Kruegel C, Vigna G (2018) Detecting Deceptive Reviews Using Generative Adversarial Networks. Proceedings of the *IEEE Security and Privacy Workshops (SPW)*, 2018, IEEE Computer Society, 89–95
26. Venkateswarlu B, Shenoi V (2021) Optimized generative adversarial network with fractional calculus based feature fusion using Twitter stream for spam detection.
27. Schouten K, Frasincar F (2016) Survey on Aspect-Level Sentiment Analysis. Knowl Data Eng IEEE Trans on 28(3):813–830
28. Jiwei Li CC (2013) Sujian Li. Topicspam: a topic-model based approach for spam detection. Proceedings of the *the 51st Annual Meeting of the Association for Computational Linguistics (ACL 2013)*, Sofia, Bulgaria, 217–221
29. Lee KD, Han K, Myaeng S-H (2016) 2016 of Conference, Nimes, France, 1–7
30. You L, Peng Q, Xiong Z, He D, Qiu M, Zhang X (2020) Integrating aspect analysis and local outlier factor for intelligent review spam detection. Future Generation Computer Systems 102:163–172
31. Zhou J, Huang JX, Chen Q, Hu QV, Wang T, He L (2019) Deep Learning for Aspect-Level Sentiment Classification: Survey, Vision, and Challenges. IEEE Access 7:78454–78483
32. Xue H, Wang Q, Luo B, Seo H, Li F (2019) Content-aware trust propagation toward online review spam detection. J Data Inform Qual (JDIQ) 11(3):1–31
33. Li J, Lv P, Xiao W, Yang L, Zhang P (2021) Exploring groups of opinion spam using sentiment analysis guided by nominated topics.Expert Systems with Applications,171
34. Muhlenbach F, Lallich S, Zighed DA (2004) Identifying and Handling Mislabelled Instances. J Intell Inform Syst 22(1):89–109
35. Chen C-C, Huang H-H, Chen H-H (2018) NTUSD-Fin: a market sentiment dictionary for financial social media data applications. Proceedings of the *Proceedings of the 1st Financial Narrative Processing Workshop (FNP* 2018
36. Li J, Zhang P, Yang L (2021) An unsupervised approach to detect review spam using duplicates of images, videos and Chinese texts. Comput Speech Lang 68:101186

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