Low-Quality *DanMu* Detection via Eye-Tracking Patterns

Xiangyang Liu, Weidong He, Tong Xu(✉), and Enhong Chen

School of Data Science, University of Science and Technology of China, Hefei, China
{liuxiangyang,hwd}@mail.ustc.edu.cn,{tongxu,cheneh}@ustc.edu.cn

Abstract. With the development of online video platforms, a comment visualization system that inserts dynamic and contextualized comments on a video has become popular in Japan and China, known as *DanMu*, which provides a feeling of "virtual liveness". However, at the same time, it also brings some bad influences such as goal impediment and information overload, distraction problems, impolite and irrelevant comments. To solve this problem, there are several studies utilizing textual content for low-quality *DanMu* detection. However, they leave out the visual context and do not consider users' watching behavior. To this end, in this paper, we propose an end-to-end multimodal classification framework for low-quality *DanMu* detection. Specifically, we first design a lab-based user study to investigate users' watching patterns. Based on the discovered fixation patterns, we propose a new fusion method to fuse them with textual context. Moreover, visual content is also considered with a further fusion mechanism. Our model outperforms other baselines in almost all classification metrics in the real-world dataset.

Keywords: Datasets · Neural networks · Eye-Tracking pattern · Text tagging

1 Introduction

In the process of booming development of video market, one emerging type of user-generated comment named *DanMu* [10] has become more and more popular at many online video platforms, e.g., *niconico*[1] in Japan and *Bilibili*[2] in China. Unlike traditional online reviews displayed in a separate space outside the video, *DanMu* is overlaid directly on the top of videos by synchronizing the comment with specific playback time, somewhat similar in appearance to the film subtitles. Previous research has indicated that *DanMu* creates a feeling of "virtual liveness" [16] as well as an experience of co-viewing [17], which largely increases users' watching experience. However, *DanMu* technology also brings some bad influences, e.g., goal impediment and information overload [15], distraction problem [13], impolite and irrelevant comments (quarrels between fans, or spoilers)

[1] http://www.nicovideo.jp/.
[2] http://www.bilibili.com/.

G. Memmi et al. (Eds.): KSEM 2022, LNAI 13370, pp. 247–259, 2022.
https://doi.org/10.1007/978-3-031-10989-8_20

[4,22], etc. Some typical instances are illustrated in Fig. 1. The audience had a dispute over the food price mentioned in the video.

Fig. 1. A screenshot from a video which is introducing local food. *DanMu* in Dotted box are impolite and irrelevant comments that affect users' watching experience.

To alleviate this problem, some video platforms allowed users to filter *DanMu* utilizing pre-defined rules or regular expressions, which was not flexible. In academia, [23] proposed a Similarity-Base Network with Interactive Variance Attention to detect spoilers from *DanMu*. To better utilize the context information of *DanMu*, a graph convolutional encoder and a contextual encoder were used to capture the semantic feature of *DanMu* by [12]. At the application level, [18] designed and implemented a cloud-assisted *DanMu* filtering framework, including a CNN-based *DanMu* quality classifier that runs on the cloud server and a front-end Google Chrome browser extension. However, they usually only leveraged the textual context information, i.e., the surrounding *DanMu*, to judge whether a *DanMu* should be filtered. We argue that the visual context is necessary for low-quality *DanMu* detection, especially for irrelevant comments detection. Moreover, users usually exhibit specific patterns when watching videos with *DanMu*. For example, people tend to link textual descriptions to visual depictions in a simple manner and often become confused if the link is not clear [8]. However, how to utilize these patterns to improve performance is still largely underexplored.

Along this line, we collect a user-generated video dataset[3] and conduct a series of dedicated experiments to explore users' watching behavior with an eye-tracker. The eye-tracker could collect users' eye movements during the video watching process. Through analyzing users' eye movements, we found that users tend to pay attention to *DanMu* that have similar semantics when watching videos. In addition, we also discover several fixation patterns. We propose an

[3] We will publish the dataset after the acceptance of this paper.

end-to-end multimodal classification framework for low-quality *DanMu* detection based on these observations. To be specific, we first utilize a convolutional neural network (CNN) to encode visual context. Then, we leverage BERT [6] to obtain the embeddings of *DanMu*. To combine users' watching behavior, we design a pattern encoder to extract related features about defined eye-tracking patterns. Thereafter, we propose a new fusion method based on the bilinear model to fuse eye-tracking features and text representation. Finally, representations from different modalities are combined for *DanMu* classification.

In general, the contribution of this paper can be summarized as follows:

- We conduct a lab-based user study to collect various user behavior data when watching videos with *DanMu*, which will be released to the research community.
- We propose an end-to-end multimodal classification framework for low-quality *DanMu* detection with several discovered fixation patterns.
- We conduct extensive experiments to evaluate the proposed model, and the results show the effectiveness compared with several state-of-the-art baselines.

2 Eye-Tracking Pattern Mining

This section will introduce our dataset and make some preliminary analyses of eye-tracking data to explore the human eye-tracking patterns.

2.1 Data Preparation

We collect a user-generated videos dataset from *Bilibili*, one of the largest video-sharing platforms in China, which focuses on animation, movies, etc. To be specific, this dataset contains 62 videos, each lasting around 5 min long and containing about 1000 *DanMu*. All videos are divided into 12 groups. Each group includes 5 or 6 videos, and the total time is no more than 30 min.

We recruit 14 participants to take our tasks. There are eight males and six females with ages ranging from 21 to 24. All of them are undergraduate, or graduate students and their majors vary from natural science and engineering to humanities and sociology. All participants are familiar with *DanMu* and usually browse the online video platforms. In addition, we screen all applicants according to their visual acuity to ensure that the collected eye-tracking data are correct. Each group of videos is watched by 6 participants at least to eliminate random factors.

To obtain eye-tracking data, we use a *Tobii Pro*[4] eye tracker to record the eye-tracking of participants during watching videos whose deviation is within the character level. Before taking tasks, there is a calibration process for each participant to ensure that the data of eye movements can be recorded accurately. After that, they need to rate the correlation between the *DanMu* and the video scene where the *DanMu* appears, with 1 being related and −1 not. Considering

[4] www.tobiipro.com.

that we aim to filter out the worst *DanMu*, samples with labels mean larger than 0.8 are considered positive, and the remaining are negative samples. Finally, we get 11,000 positive samples and 27,039 negative samples to help us construct a binary classification problem.

Table 1. Toy example of eye-tracking data

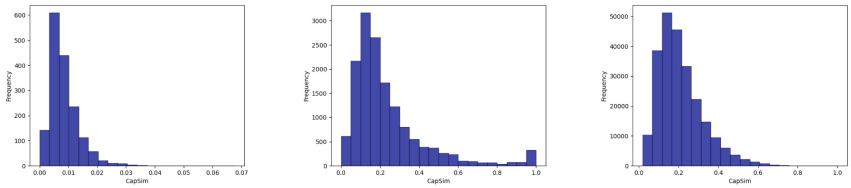
Time(s)	x	y	Type	Duration(ms)	Fixation_x	Fixation_y*	Focus_type	Focus_text**
19.642	1229	335	Fixation	183	1229	351	*DanMu*	Not really
19.658	1219	337	Fixation	183	1229	351	Image	
19.675	1212	338	Fixation	183	1229	351	Image	
19.692	1212	325	Fixation	183	1229	351	*DanMu*	Not really
19.708	1220	365	Fixation	183	1229	351	Image	
19.725	1235	369	Fixation	183	1229	351	*DanMu*	I have 6 years

*Fixation_x and Fixation_y stand for coordinates of gazed position.
**Tranlated from Chinese.

2.2 Eye-Tracking Pattern Generalisation

Some examples of eye-tracking data in our dataset are presented in Table 1. Before processing the collected eye-tracking data, we first need to understand how it reveals our watching behavior. From previous work, we can learn that the eye reads a frame of videos in discrete chunks by making a series of fixations and saccades. A fixation is a brief moment, around 250 ms, where the eye is paused on a *DanMu* or an area of this frame, and the brain processes the visual information. A saccade is a fast eye movement to take in the subsequent fixation. In other words, a fixation means the participant is concentrating on reading, and a saccade is a bond that connects two fixations.

Preliminary statistical results show that gazing at *DanMu* is a sparse behavior while watching a video, which naturally raises our question: does this sparse behavior include more information, or in other words, is the gazed *DanMu* significantly different from other *DanMu*? We attempt to analyze this issue from a semantic perspective. To be specific, we select a Chinese image caption dataset, calculate the BLEU score, and compare it with the *DanMu*' BLEU score, as shown in Fig. 2. We have found that the semantics of the overall *DanMu* is more diverse than that of the image caption dataset. But the gazed *DanMu* has a lower diversity, which means people tend to pay attention to *DanMu* with similar semantics when watching videos. However, a gazed *DanMu* by participants does not always mean that the *DanMu* is related to the video content. For example, [7] note that fixations focused on two points mean that participants got confused with elements on the screen. To distinguish between different types of fixations, inspired by [7] we have defined three fixation patterns as Table 2 shows. In practice, each *DanMu* corresponds to one or more patterns because different participants have differences in cognitive behavior.



(a) BLEU score of all *DanMu* (b) BLEU score of gazed *DanMu* (c) BLEU score of a Chinese image caption dataset

Fig. 2. Semantic analysis

Table 2. Definition to eye-tracking patterns (with ratio of occurrences of patterns)

Patterns id	Occurred ratio	Eye-tracking pattern generalisation
Pattern #1	36.45%	Many short fixations across several *DanMu*
Pattern #2	51.91%	Short fixations on specific *DanMu* followed by some regressions
Pattern #3	11.64%	Long fixations on specific *DanMu*

3 Technical Framework

Previous research [2,3,5] on multimodal tasks has provided a helpful paradigm that takes output vectors of independent network models of different modalities as input. Then the fusion module will combine the output vectors into a single vector as the multimodal joint representation.

We follow this paradigm to define the structure of the framework as having three components, a vision encoder, a joint *DanMu*-eyetracking encoder, and a classifier that predict the joint two prior components' embedding (Fig. 3). We opt for the "early fusion" scheme for joining predictions. The modular nature of this structure allows us to analyze the joint *DanMu*-eyetracking encoder quantitatively.

3.1 Vision Encoder

We utilize a standard CNN architecture ResNet50 [9] for the vision encoder. And we'll replace the last full connection layer of ResNet with a new full connection layer that maps pooling out to the d-dimension. Then we resize the input video frame to $480 * 270$ and rescale the pixel values to lie within the range $[-1, 1]$ and get the visual representation $\mathbf{v}_i \in \mathbb{R}^{batch \times d_t}$.

3.2 Joint *DanMu*-Eyetracking Encoder

As Fig. 3 shows, the joint *DanMu* eye-tracking encoder consists of two parts. One is used to represent the text with Bert [6] directly. Here we use Bert's pooled

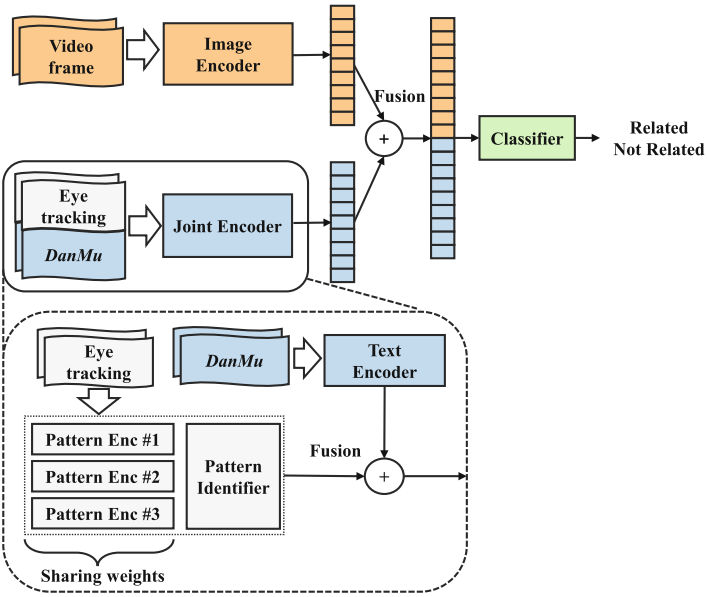


Fig. 3. End-to-end multimodal classification framework.

out vector as the direct text embedding. The other part is the eye-tracking pattern encoding module to extract eye-tracking pattern features to enhance text representation.

3.3 Eyetracking Pattern Encoder

As we discussed in Sect. 2, the eye-tracking data of a *DanMu* can be represented as

$$v_{raw} = [\chi(g), p_g, \bar{t}_{g/s}] \tag{1}$$

where $\chi(g)$ is a 0–1 variable used to indicate whether the *DanMu* is gazed at by more than one subject. p_g represents the probability of the subject gaze at the *DanMu*; $\bar{t}_{g/s}$ represents the average time for all participants to stare at this *DanMu*. We can calculate these features as follows:

$$p_g = \frac{N_s}{N}, \bar{t}_{g/s} = \frac{T_g}{N} \tag{2}$$

where N is the number of participants, N_s is the number of participants who gazed at a specific *DanMu*, T_g represents the total time for all participants to gaze at a specific *DanMu*.

After feature engineering, we get three synthetic features. Further encoding will be achieved through a linear affine transformation, where we use a diagonal matrix as the invertible matrix. This diagonal matrix can adjust the weight of each synthetic feature and provide interpretability for the eye-tracking pattern

encoder. As discussed above, each *DanMu* may correspond to multiple fixation patterns, so we use three eye-tracking pattern encoders with shared parameters to encode each pattern of a *DanMu* separately. After encoding, a pattern identifier module is used to sum the above vectors. To be specific, this part is calculated as follows:

$$\mathbf{v}_{et} = \sum_{p}^{\mathbb{P}} w_p(\mathbf{D} \cdot \mathbf{v}_{raw}) \in \mathbb{R}^{|\mathbf{v}_{raw}|} \tag{3}$$

where w_p is the weight of the pattern p given by pattern identifier, $\mathbf{D}$ is the parameter diagonal matrix.

3.4 Fusion Method

Most fusion methods can only be used for two input modalities. Some particular fusion method suitable for multimodalities fusion needs to convert each vector to a dockable vector. As the length of an eye-tracking vector is too short compared to the length of a vision or a text vector, forcibly mapping the eye-tracking vector into a much higher dimension space will bring unnecessary redundancy and make it hard to train. As we discussed in Sect. 2, eye-tracking data can be easily matched with *DanMu* text. Therefore, our fusion approach is divided into two steps. The first step is to integrate the eye-tracking vector with the text vector to obtain joint representation and then fuse it with the vision vector. To accomplish this fusion task, we apply convolution as the linearizing operation in the bilinear model [20] as our text-eye-tracking fusion method.

More specifically, we first extract text embedding $\mathbf{v}_t \in \mathbb{R}^{batch \times d_t}$ from Bert and eye-tracking pattern encoder $\mathbf{v}_{et} \in \mathbb{R}^{batch \times d_e}$ for tensor product operation: $\mathbf{v}_{et} \otimes \mathbf{v}_t \in \mathbb{R}^{batch \times d_e \times d_t}$. Before this step, to avoid possible null eye-tracking vectors that could adversely affect the final embedding, we replace the 0–1 variable $\chi(g)$ with its one-hot encoding in the eye-tracking representation. Then a convolution operation is applied to linearize the result of the tensor product. Defining $f_c(*)$ as a convolution operation and W as a linear transform, the joint *DanMu*-eyetracking representation can be calculated as follow:

$$v_{joint} = W \cdot f(\mathbf{v}_{et} \otimes \mathbf{v}_t) \in \mathbb{R}^{|\mathbf{v}_i|} \tag{4}$$

Finally, we obtain the multimodal representation:

$$v_{mm} = Concat(W \cdot f_c((\sum_{p}^{\mathbb{P}} w_p \mathbf{D}[\chi(g), p_g, \bar{t}_{g/s}]) \otimes \mathbf{v}_t), \mathbf{v}_i) \tag{5}$$

4 Experiments

This section will evaluate our method in a real-world dataset and compare it with the baselines we selected.

4.1 Experimental Setup

For a fair comparison with other methods, we use the same image encoder and text encoder as the backbone. In all experiments, we use an AdamW solver with a learning rate = 0.00002 and a schedule with a learning rate that decreases linearly from the initial learning rate set in the optimizer to 0.

As our dataset labels are imbalanced, we sample 1000 positive and negative samples as the validation and test sets, respectively. Accuracy and F1 score are our model criteria, while the loss function during the training stage is the cross-entropy loss function.

4.2 Comparison of Baseline Methods

To evaluate the performance of our proposed model, we compare it with the following methods as baselines:

- **fastText.** This is a lightweight library for efficient learning of word representations and sentence classification developed by [11]. FastText is often on par with deep learning classifiers in terms of accuracy and many orders of magnitude faster for training and evaluation.
- **Smartbullets.** TextCNN is introduced to tackle sentence-level classification tasks with convolutional neural networks. [18] construct a user-centered *DanMu* filter with this method named Smartbullets.
- **Multimodal classifier(Text+Image).** This is our model without eye-tracking data. We replace the joint encoder with a Bert text encoder.

We choose the model with the lowest validating loss during the training phase to test on the test set. The experimental results are shown in Table 3.

From the result of the experiments, we discover that both multimodal context and eye-tracking data can effectively improve model performance. Our model is better than the baselines we selected, of course.

Table 3. Overall performance

Methods	Accuracy	F1-score
fastText	0.6375	0.7180
Smartbullets	0.6540	0.7232
Bert classifier	0.6520	0.7240
MM classifer(Text+Image)	0.6975	0.7389
MM classifer(Text+Image+eye-tracking)	**0.7375**	**0.7552**

4.3 Fusion Method Experiment

To verify the effectiveness of the fusion module we have designed for this task, we have selected some typical fusion methods as the baselines to compare with our methods, which are as follows:

- **Concatenation.** This is the most basic multimodal fusion method with strong applicability and wide application. This method directly concatenates the two vectors together to get the target vector [19].
- **Co-attention.** This is a method proposed by [14] for image-text fusion. The specific fusion process is as follow:

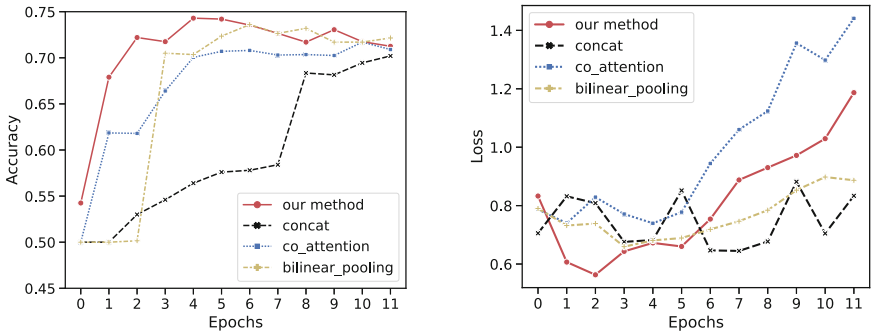
$$C^P = [H^q; H^p softmax(H^p(H^q)^T)] softmax(H^q(H^p)^T) \tag{6}$$

  Then the LSTM model is used to map C^P into a given output space.
- **Bilinear pooling with linearizing.** Bilinear pooling take the outer product of two vectors $p \in \mathbb{R}^{d_p}$ and $q \in \mathbb{R}^{d_q}$ and learn a linear transform W [21], the result vector z can be calculated as follow:

$$z = W \cdot Vec(p \otimes q) \tag{7}$$

  where $\otimes$ denotes the outer product and $Vec(*)$ denotes linearizing the outer product matrix in a vector.

We chose the model with the lowest validating loss during the training phase to test on the test set, and the experiment results are shown in Table 4. We also draw the loss curve and the accuracy curve of the validation set when different fusion methods are used during the training process, as shown in Fig. 4. The results prove that our proposed fusion method is superior to the given baseline in prediction accuracy and convergence performance.



(a) Accuracy curve of validation set during training phase

(b) Loss curve of validation set during training phase

Fig. 4. Performance of given fusion methods during training phase

Table 4. Fusion methods performance

Method	Accuracy	F1-score
Concatenation	0.5770	0.6887
Co-attention	0.7025	0.7403
Bilinear pooling with linearizing	0.7015	0.7326
Bilinear pooling with convolution (Ours)	**0.7375**	**0.7552**

5 Related Works

5.1 Low-Quality *DanMu* Detection

Recent studies of low-quality *DanMu* detection mainly focus on keyword matching and deep learning methods. Keyword matching methods are based on predefined keywords widely used in famous video platforms. However, the keyword matching methods require human-fixed input and have high recall and low precision performance since they treat many positive comments as low-quality *DanMu*. The other domain of research is machine learning methods. [23] propose a Similarity-Based Network with Interactive Variance Attention to detect spoilers from *DanMu*. They construct a word-level attentive encoder and a sentence-level interactive variance attention network to embedding *DanMu* text with their contextual information. To better utilize the context information of *DanMu*, a graph convolutional encoder and a contextual encoder are used to capture the semantic feature of *DanMu* by [12]. At the application level, [18] design and implement a cloud-assisted *DanMu* filtering framework, including a CNN-based *DanMu* quality classifier that runs on the cloud server and a front-end Google Chrome browser extension. However, they leave out the multimodal context as well as the human watching behavior.

5.2 Eye-Tracking

A review of the early literature provides a few examples of research regarding people's eye movements as they integrated both textual and visual elements in an information-seeking context [20]. Faraday and Sutcliffe [8] reveal that participants sought to link textual descriptions to visual depictions in a simple manner. If the link wasn't clear, participants often become confused about how the two channels could be synthesized into a coherent whole. [7] discover 23 human eye-tracking patterns of surfing the website and reveal the links between usability problems and eye-tracking patterns.

5.3 Fusion Methods

Fusion is a crucial research topic in multimodal studies, which integrates information extracted from different unimodal data sources into a single compact

multimodal representation. Three types of fusion methods are mainly used for the multimodal task, namely, simple operation-based, attention-based, and bilinear pooling-based methods.

A widely used operation-based fusion method is concatenation [24] which is not required the same number of elements arranged in an order. However, this simple fusion method is not ideal enough. To get a better fusion performance, some more complex but better methods are proposed. A typical method is attention fusion which utilizes attention mechanisms in modal fusion. Attention mechanisms often refer to the weighted sum of a set of vectors with scalar weights [1]. Among these attention fusion methods, co-attention is a representative method. In the original paper, the authors use co-attention to fuse the image modal and text modal in VQA. This method uses symmetric attention structures to generate attended not only image feature vectors but also language vectors[14]. Based on the bilinear model, [21] proposes a fusion method that facilitates multiplicative interactions between all elements in both input vectors via computing their outer product.

6 Conclusion

In this work, we tackled the problem of low-quality *DanMu* detection using the image, text contents, and eye-tracking data. Our main idea is human visual cognitive patterns imply the emotional tendency of the viewed object. To understand human cognitive processes, we collect an eye-tracking dataset to mine human cognitive patterns during watching behavior. Then a weight-shared pattern encoder is applied to adaptively represent different patterns. It is clear from the experiments that introducing human eye-tracking patterns and visual information can efficiently improve the quality and accuracy of predictions.

However, the eye-tracking patterns we defined do not fully utilize sequence features of the eye-tracking data. In further studies, we will try to extract features from eye-tracking sequences with a more efficient method.

Acknowledgements. This work was partially supported by the grants from the National Natural Science Foundation of China (No.62072423)

References

1. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473 (2014)
2. Chen, L., Li, Z., He, W., Cheng, G., Xu, T., Yuan, N.J., Chen, E.: Entity summarization via exploiting description complementarity and salience. IEEE Trans. Neural Netw. Learn. Syst. (2022)
3. Chen, L., Li, Z., Wang, Y., Xu, T., Wang, Z., Chen, E.: MMEA: entity alignment for multi-modal knowledge graph. In: Li, G., Shen, H.T., Yuan, Y., Wang, X., Liu, H., Zhao, X. (eds.) KSEM 2020. LNCS (LNAI), vol. 12274, pp. 134–147. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-55130-8_12

4. Chen, Y., Gao, Q., Rau, P.L.P.: Watching a movie alone yet together: understanding reasons for watching Danmaku videos. Int. J. Human-Comput. Interact. **33**(9), 731–743 (2017)
5. Choi, J.H., Lee, J.S.: EmbraceNet: a robust deep learning architecture for multimodal classification. Inf. Fusion **51**, 259–270 (2019)
6. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
7. Ehmke, C., Wilson, S.: Identifying web usability problems from eyetracking data (2007)
8. Faraday, P., Sutcliffe, A.: Making contact points between text and images. In: Proceedings of the Sixth ACM International Conference on Multimedia, pp. 29–37 (1998)
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
10. He, M., Ge, Y., Chen, E., Liu, Q., Wang, X.: Exploring the emerging type of comment for online videos: DanMU. ACM Trans. Web (TWEB) **12**(1), 1–33 (2017)
11. Joulin, A., Grave, E., Bojanowski, P., Mikolov, T.: Bag of tricks for efficient text classification. In: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, vol. 2, Short Papers, pp. 427–431. Association for Computational Linguistics, April 2017
12. Liao, Z., Xian, Y., Li, J., Zhang, C., Zhao, S.: Time-sync comments denoising via graph convolutional and contextual encoding. Pattern Recogn. Lett. **135**, 256–263 (2020)
13. Liu, L., Suh, A., Wagner, C.: Who is with you? Integrating a play experience into online video watching via Danmaku technology. In: Kurosu, M. (ed.) HCI 2017. LNCS, vol. 10272, pp. 63–73. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-58077-7_6
14. Lu, J., Yang, J., Batra, D., Parikh, D.: Hierarchical question-image co-attention for visual question answering. In: Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., Garnett, R. (eds.) Advances in Neural Information Processing Systems, vol. 29. Curran Associates, Inc. (2016)
15. Lv, G., Xu, T., Chen, E., Liu, Q., Zheng, Y.: Reading the videos: temporal labeling for crowdsourced time-sync videos based on semantic embedding. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 30 (2016)
16. Lv, G., et al.: Gossiping the videos: an embedding-based generative adversarial framework for time-sync comments generation. In: Yang, Q., Zhou, Z.-H., Gong, Z., Zhang, M.-L., Huang, S.-J. (eds.) PAKDD 2019. LNCS (LNAI), vol. 11441, pp. 412–424. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-16142-2_32
17. Lv, G., et al.: Understanding the users and videos by mining a novel DanMU dataset. IEEE Trans. Big Data. **8**, 535–551 (2019)
18. Niu, H., Li, J., Zhao, Y.: Smartbullets: a cloud-assisted bullet screen filter based on deep learning. In: 2020 29th International Conference on Computer Communications and Networks (ICCCN), pp. 1–2. IEEE (2020)
19. Nojavanasghari, B., Gopinath, D., Koushik, J., Baltrušaitis, T., Morency, L.P.: Deep multimodal fusion for persuasiveness prediction. In: Proceedings of the 18th ACM International Conference on Multimodal Interaction, pp. 284–288 (2016)
20. Rayner, K., Rotello, C.M., Stewart, A.J., Keir, J., Duffy, S.A.: Integrating text and pictorial information: eye movements when looking at print advertisements. J. Exp. Psychol. Appl. **7**(3), 219 (2001)

21. Tenenbaum, J.B., Freeman, W.T.: Separating style and content with bilinear models. Neural Comput. **12**(6), 1247–1283 (2000)
22. Wang, J.: How and why people are impolite in DanMU? Internet Pragmat. **4**, 295–322 (2021)
23. Yang, W., Jia, W., Gao, W., Zhou, X., Luo, Y.: Interactive variance attention based online spoiler detection for time-sync comments. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 1241–1250 (2019)
24. Zhou, B., Tian, Y., Sukhbaatar, S., Szlam, A., Fergus, R.: Simple baseline for visual question answering. arXiv preprint arXiv:1512.02167 (2015)