Spectrum of Engineering Sciences
ISSN (e) 3007-3138 (p) 3007-312X Volume 3, Issue 11, 2025

SENTIMENT ANALYSIS IN SOCIAL MEDIA USING MULTIMODAL DATA FUSION TECHNIQUES

Muhammad Zarar, [2]Talha Nadeem, [3]Huzaifah Bin Khawar, [4] Hood Khizer

[1] AI Researcher, Research and Development Department, Trouve Labs, AHOY DMCC and The Superior University Lahore, Pakistan muhammadzarar.ieee@gmail.com

[2] Systems Architect AI/ML, Research and Development Department, Trouve Labs, AHOY DMCC and The Superior University Lahore, Pakistan talhanadeem.ieee@gmail.com

[3] AI Researcher, Research and Development Department, Trouve Labs, AHOY DMCC and The Superior University Lahore, Pakistan huzaifahkhawar.ieee@gmail.com

[4] CEO Trouve Labs, CTO AHOY DMCC, Research and Development Department, Trouve Labs, AHOY DMCC, Dubai hoodkhizer@gmail.com

DOI:

Keywords
Multimodal sentiment analysis, feature fusion, deep learning, attention mechanism.

Article History
Received: 10 Nov, 2025

Accepted: 18 Nov, 2025

Published: 30 Nov, 2025

Copyright @Author
Corresponding Author: * Talha Nadeem*

Abstract

In recent years, with the popularity of social media, users are increasingly keen to express their feelings and opinions in the form of pictures and text, which makes multimodal data with text and pictures the content type with the most growth. Most of the information posted by users on social media has obvious sentimental aspects, and multimodal sentiment analysis has become an important research field. Previous studies on multimodal sentiment analysis have primarily focused on extracting text and image features separately and then combining them for sentiment classification. These studies often ignore the interaction between text and images. Therefore, this paper proposes a new multimodal sentiment analysis model. The model first eliminates noise interference in textual data and extracts more important image features. Then, in the feature-fusion part based on the attention mechanism, the text and images learn the internal features from each other through symmetry. Then the fusion features are applied to sentiment classification tasks. The experimental results on two common multimodal sentiment datasets demonstrate the effectiveness of the proposed model.

1. INTRODUCTION

People are now more likely to share their thoughts on different social media platforms as the use of social media continues to grow. Daily hundreds of millions of data records are being generated of which many are text and images, and there is huge multimodal data. The sentiments found in this data offer a wealth of sentiment information and can be analyzed to gauge people's latest emotions and ideas of different events.

A multitude of sectors such as box office forecasting, stock market predictions, political elections, etc. can be benefited by the application of Multimodal sentiment analysis. In the graphic data being exchanged in Social Media, text and image both contain emotional data on their own, but complementary. Multimodal data offers more complete and detailed relief of the true sentiment of users as compared to single modal data. But there are some problems while performing multimodal Sentiment analysis. There are several types of modal data where the sentimental information is different, which also requires a suitable representation of the sentimental features to analyze.

Not all parts of a picture are relevant to the expression of sentiment and not all parts of a text are relevant to the sentiment. Hence, during the process of feature extraction the salient sentiment zones must be highlighted in images, while reducing the noise in textual data. Further, various types of modal data represent underlying features under different dimensions and attributes. Most of the research in sentiment analysis of single mode texts has heavily depended on statistical techniques and they are based on the quality of features extracted. For example, in the field of word sense disambiguation, some researchers have used envelopment analysis as a WSD method that is based on the interpretation of word meanings in their context, in order to determine the voters' intention when voting for political candidates (Rodger et al. [3]). Likewise, sentiment analysis of unimodal images relies on the ability for manually developed lexicon based extraction rules; this could create duplicate sentiments.

As technology licks' machine learning and deep learning methods have evolved, numerous new approaches for sentiment analysis of multimodal data have arisen, producing satisfactory results. Multimodal sentiment analysis is fast emerging as a vital research field, with many studies carried out to try to better understand the nuances involved. One of the major problems in multimodal sentiment analysis is the utilization of the multimodal internal and interactive information in an effective way. The data types gathered locally in each mode are considered internal information, and social actors make up the interactions and connections between the different modalities, are called the interaction information.

Multimodal tasks differ from single-modal tasks because of the question of how to use the information in the interaction across modalities. This makes it necessary to discover approaches that are efficient at fusing all the information from each individual mode in an effective multimodal affective analysis. These similarities in structure across a number of modes are recognized to identify those that belong to the same class and those that do not.

The use of multimodal versus single-modal tasks lies in determining how the information in the modalities might be utilized. Therefore, effective MMAA needs to find effective ways to integrate the information given by each of the modes.

Given the similarities of the structures of different modal data, researchers can frame significant information that traditional data does not usually contain, which can maximize the fusion of modal information and inter-modal interaction.

However, the majority of the previous studies have been inexpressive in the case of utilizing or employing internal text information and the interaction information of modalities – text-image. For multimodal sentiment analysis, some researchers have suggested by using deep semantic networks, long short term memory (LSTM) networks with attention mechanism are used to extract keywords. This method takes the one-way relationship between image and text, but ignores the two-to-two relationship between text and image [8]. In other research the co-memory networks were employed to capture the interaction information between the text content and the image content, then the combination of these feats was used for sentiment classification. However, coarse-grained attention mechanisms used in these approaches also can nuke some aspects and may cause information loss issues [9].

In order to tackle these issues, this paper presents a multimodal sentiment classification system based on a fine-grained attention mechanism. First, a denoising autoencoder is used to reduce noise in textual data, learning features so that these features are more representative of the original text. Secondly, a better variational autoencoder is used along with the attention mechanism to obtain features from the image. For this, a fusion model incorporating attention mechanism is proposed for learning an interactive modal fusion representation vector that weaves in the salient features of textual and image-based information, both together.

This paper makes the following contributions:

The utilization of a denoising autoencoder along with an improved variational autoencoder combined with an attention mechanism to extract more precise text and image features.

Adopting an attention mechanism type multimodal cross-feature fusion model to effectively extract different modal features in the process of fusion, thereby obtaining more accurate sentiment classification information.

The proposed approach was tested data-set-wise on two different standard Twitter datasets. Comparative tests were carried out, which showed the feasibility and efficiency of the proposed model.

The rest of the article follows the structure of recent efforts in multimodal sentiment, followed by the description of the proposed classification model, followed by the analysis of experimental results with the two Twitter datasets, and finally a summary of results and thoughts on future areas of research.

. RELATED WORK

Most of the traditional sentiment analysis is related to textual data. But the internet has accelerated progress in social media platforms to allow users to use many mediums such as text, pictures, audio and video to transmit opinions and views. This change has increased the interest in Multimodal sentiment analysis with a focus on the incorporation of textual and image information.

Most of the previous works focused on the analysis of sentiment classification based on traditional feature based classification techniques. A machine learning approach has been adopted, for instance, by Borth et al., who set about extracting 1200 adjective-noun pairs and paired them with image features from the original image set to derive sentiments scores at can be

considered as mid-level features. Likewise, Cao et al., extracted the text features by referring to a sentiment dictionary and developed a visual sentiment ontology by extracting adjective-noun pairs from images, and further integrated them with text features to classify the sentiment of images. Poria et al. performed the research by using text, voice and video featured text with a support vector machine (SVM) classification algorithm. Wang et al. designed a single cross-media bag of words model to extract features from images and text, and used logistic regression, SVM and Naive Bayes classifiers for comparison. Baecchi et al. proposed a semi-supervised sentiment classification model by extending the Continuous Bag of Words (CBOW); features extracted from images were based on using a denoising autoencoder (DA, denoising autoencoder); and they performed the classification of the fused textual and image features in both unsupervised and semi-supervised learning. The following approaches are outlined in [11-15].

Recently, the development of machine learning and deep neural networks has resulted in notable developments in multimodal sentiment analysis based on deep learning methods. For example, You et al. present a cross-modal consistent regression (CCR) method which focuses on using similar sentiment patterns between image, text and target,

Modern machine learning and deep neural networks have facilitated huge strides in multimodal sentiment analysis with the use of deep learning. You et al., for example, introduced a 'cross-modal consistent regression' (CCR) approach which stresses the importance of common consistent sentimental disposition between image and text. This method brought consistency constraint at the fusion stage to create a feature vector with a more coherent piece that could be better used for sentiment classification. Another work by You et al. researched a TreeLSTM model using a tree model and visual attention mechanism which detected the polarity of sentiment direction in text and images based on intra-dependencies, with good results. The cross modal regression is reported in [16] and the TreeLSTM method is reported in [17].

Chen et al. used the attention mechanism combined with SVM, CNN to build separate sentiment classification model for text and image and studied the effects of various fusion methods on the classification results. Yu et al. used CNN to capture word-level, phrase-level, and sentence-level text features to feed into the final sentence level for sentiment classification as part of the feature layer text features. Cai et al. utilized two separate CNNs to learn text and image features, and then fed this to a third CNN to learn the relationship between the two. References to CNN based methods are found in [18-20].

In a more advanced method, Xu et al. introduced a deep semantic network of multimodal sentiment analysis, wherean LSTM network along with attention mechanism is used to gain the keywords from those image features. The authors also presented a co-memory network in order to learn interaction information of the content in text information and image information, and used the obtained fusion features to classify the sentiment of text information and image information. Details of the two models for interactions presented in [8,9].

Interestingly, video content can also be regarded as multimodal data, as Poria and his colleagues developed a multi-kernel learning (MKL) framework for video sentiment classification,

which combined video, audio and textual data. Zadeh et al. proposed tensor fusion algorithms for the construction of joint feature, modeling to single-mode, dual-mode and triple-mode features; and learning interaction information across modalities from video information. Fusion schemes are reported in [21, 22] that are related to the vision-based fusion.These vision based fusion methods are reported in [21, 22].

3. PROPOSED MODEL

The CFF-ATT framework takes a text plus an image that are both from social media as a single pair of inputs [28]. The information contained in each modality is captured in text denoising and attention enhancement of image encoding. Then two cross feature-fusion directions are learned: text only input, image context; and image only input, text context. Two joints representations are joined together for 3-class sentiment classification. The overall structure is as depicted in Figure 1.

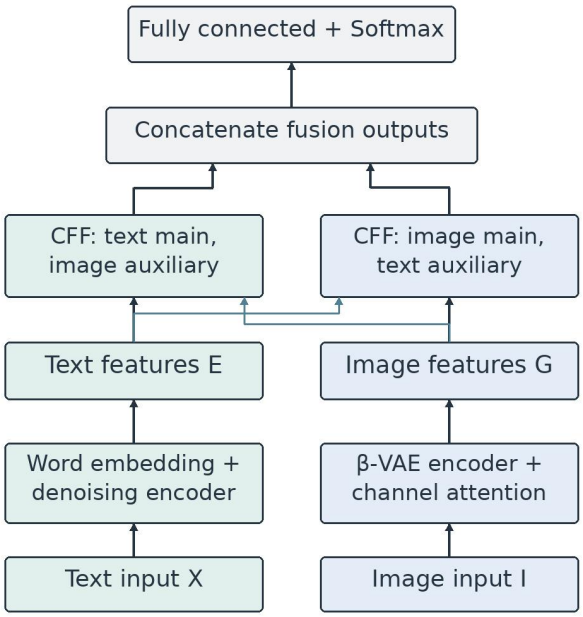


Figure 1. CFF-ATT architecture with text denoising, attention-enhanced image encoding, and bidirectional feature fusion

3.1 Mathematical Notation and Problem Formulation

Table 1 defines the notation used consistently in the mathematical description and algorithms. The generic symbols E and G identify main and auxiliary feature matrices, not separate modalities permanently. Their roles are exchanged in the reverse fusion direction. The VAE regularization coefficient β and channel-attention residual scale ρ are distinct parameters.

Table 1. Mathematical notation and its relevance to the CFF-ATT methodology

Symbol	Definition	Methodological relevance
X, x_t, T	Text, token at position t, and token count.	Define the textual member of each text-image pair.
I, I-hat	Input image and its VAE reconstruction.	Specify the visual input and reconstruction target.
Y, d_w	Word-embedding matrix and embedding width.	Y has shape d_w by T before denoising.
x, x-tilde, x-hat, d_x	Normalized DAE vector, corrupted vector, reconstruction, and input width.	Separate clean targets from masked inputs and decoded outputs.
h_x, f, g	DAE hidden code, encoder, and decoder.	Extract denoised features and reconstruct the clean input.
W, b_h, b_x, theta	DAE weight, hidden/output biases, and parameter set.	Use a tied decoder weight W transpose.
m, p_keep	Binary corruption mask and keep probability.	Random masking trains the DAE to recover missing entries.
sigma, tanh, odot, :	Sigmoid, hyperbolic tangent, elementwise product, and concatenation.	Define nonlinear transforms and feature-combination operations.

Symbol	Definition	Methodological relevance
BCE, L_DAE, D_DAE	Binary cross-entropy reconstruction error, DAE objective, and DAE training vectors.	Measure denoising error for inputs scaled to the unit interval.
phi, psi, q_phi, p_psi	VAE encoder/decoder parameters and conditional distributions.	Map images to a latent distribution and reconstruct images.
z, d, epsilon, I_d, p_0	Latent vector/width, Gaussian noise, identity matrix, and prior distribution.	Specify sampling and the standard Gaussian prior.
mu_phi, sigma_phi	Learned latent mean and standard deviation.	Parameterize the diagonal Gaussian posterior.
beta, D_KL, L_beta-VAE	KL weight, Kullback-Leibler divergence, and minimized image objective.	Balance reconstruction against latent regularization.
gamma, b, h, w, c, a	Encoder feature map, batch size, spatial dimensions, channels, and a = hw.	Reshape each image feature map to a by c for channel attention.
Q, K, V, omega_Q, omega_K	Query, key, value matrices and learned convolution parameters.	Compute cross-channel affinities with matrices of shape a by c.
S, alpha, O_j	Channel scores, normalized channel weights, and output channel j.	S and alpha are c by c; each output channel mixes source channels.
rho	Learnable channel-attention residual scale, initialized to zero.	Gradually introduces the attention correction without changing beta.
E, G, n, l	Main/auxiliary matrices and their respective position counts.	Represent the direction-specific inputs to cross-feature fusion.
d_e, d_g, d_v	Main width, auxiliary width, and shared projection width.	Use compatible projections; the reported shared width is 100.
P_E, P_G, b_E, b_G	Shared-space projection matrices and bias vectors.	Map both modality representations to the same feature width.
E_emb, G_emb	Projected main and auxiliary features.	Their inner products define fine-grained position affinities.
M, A	Unnormalized affinity matrix and row-normalized fusion weights.	Both have shape n by l; scores and probabilities are not conflated.
J	Auxiliary context matrix with shape d_g by n.	One weighted auxiliary vector is assigned to each main position.
P_U, b_U, U	Fusion projection, bias, and output matrix of shape d_e by n.	Combine each main vector with its attention-derived context.
U_t, U_v, F, vec	Text-main/image-main outputs, concatenated classifier input, and vectorization.	Preserve both fusion directions in the final feature representation.
W_o, b_o, p, u, u-hat	Classifier weight/bias, class probabilities, true label, and prediction.	Apply Softmax and select positive, negative, or neutral sentiment.
Theta, L_cls, B	Trainable classifier-stage parameters, supervised loss, and minibatch size.	Fit sentiment predictions using labeled training examples.
C, N, n_k, y_rk, p_rk	Class count, test count, class support, one-hot label, and probability.	C = 3; distinguish training probabilities from evaluation counts.
TP_k, FP_k, FN_k	One-versus-rest true positives, false positives, and false negatives for class k.	Define class-wise precision, recall, and F1 in a three-class task.

The equations below give formal tensor shapes. Attention and fusion are evaluated independently for each example in a minibatch.

For a token sequence X with T positions, word embedding gives a matrix Y whose columns are token vectors:

$$X = (x_1, \ldots, x_T), \quad Y = [y_1, \ldots, y_T] \in \mathbb{R}^{d_w \times T} \tag{1}$$

The paired input is mapped to class probabilities p over C = 3 sentiment classes. The prediction chooses the most probable class:

$$\hat{u} = \arg\max_{k \in \{1,2,3\}} p_k$$
$$\sum_{k=1}^{3} p_k = 1 \tag{2}$$

3.2 Text Feature Extraction Module

The role of any text-feature extraction layer is to assign a low dimensional vector for each word (called word embedding). The volume of textual data in social media contains a considerable amount of uncertainty that can impact the correctness of the features extracted. Featuring in text data was achieved with a denoising autoencoder (DAE) used to extract text features to remove noise interference and get stronger features [23]. The denoising autoencoder erases the original input matrix with a certain probability distribution (usually using the binomial distribution), that is, each value is randomly set to zero. This way, some features of certain data are missing. If the data does not include either feature(s) that are lost or feature(s) added by mistake, then the original data are considered to be "impure". This type of data "noise" is comparable to adding noise. In order to eliminate the interference of noise, the encoder will first run a training set, and eventually put into effect the purpose of the reconstruction of original data, called denoising. Specifically, to get the matrix, the vectorized representation of the matrix was destroyed. The damaged matrix was fed into the encoder to get the encoding and then a reconstruction matrix was received from the reconstruction of the decoder. The difference between the reconstructed and the original matrix was taken to get the reconstructed error. Then the parameter values of the encoder and decoder were adjusted so as to minimize the reconstruction error and final encoding was obtained. Taking the encoding features obtained from the upper layer as the input of the lower layer, the coding of the lower layer was obtained using the same method. Lastly the above was single looped until a desired number of layers was encoded. In brief, high-dimensional text data were reduced in dimensionality by creating a deep network from the bottom up until a low dimensional feature code layer was obtained, and finally extracting the low dimensional feature code that most represented the text from the layer. The structure of the denoising autoencoder is shown in Figure 2.

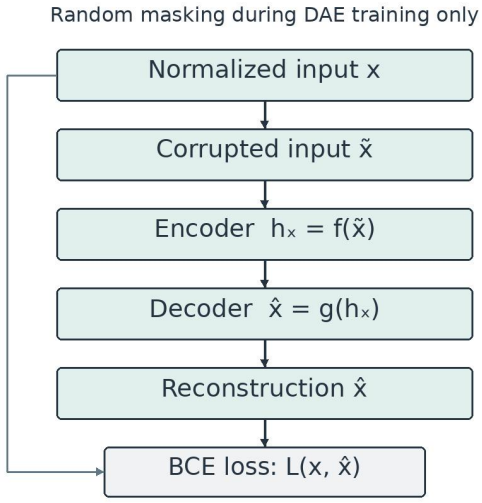


Figure 2. Denoising autoencoder with masked input, tied-weight reconstruction, and clean-input reconstruction loss

Let X be one vector that is normalized given to a DAE layer, containing d X entries. Each entry in a Bernoulli mask is set independently with a given probability. It is the vector that is not corrupted that is reconstructed again.

$$\tilde{x} = m \odot x, \quad m_i \sim \text{Bernoulli}\left(p_{\text{keep}}\right) \quad x \in [0,1]^{d_x} \tag{3}$$

The encoder and decoder use sigmoid activations and tied weights. The hidden code and reconstruction are distinct from the token embedding matrix Y:

$$h_x = f(\tilde{x}) = \sigma(W\tilde{x} + b_h) \tag{4}$$

$$\hat{x} = g(h_x) = \sigma(W^{\mathrm{T}} h_x + b_x) \tag{5}$$

$$\sigma(s) = \frac{1}{1 + \exp(-s)} \tag{6}$$

For each normalized vector, binary cross-entropy computes the average of the reconstructions over the entries in the input. This average is the reduction convention that is adopted in the mathematical description:

$$\mathrm{BCE}(x,\hat{x}) = -\frac{1}{d_x}\sum_{i=1}^{d_x} x_i \ln\hat{x}_i - \frac{1}{d_x}\sum_{i=1}^{d_x}(1-x_i)\ln(1-\hat{x}_i) \quad (7)$$

The DAE objective is defined as the sum of reconstruction errors over all the training vectors and averaging over all the random corruption masks:

$$L_{\mathrm{DAE}}(\theta) = \sum_{x\in D_{\mathrm{DAE}}} \mathbb{E}_m \left[\mathrm{BCE}\left(x, g\left(f(\tilde{x})\right)\right)\right] \quad (8)$$

With "layer-wise training", this denoising is performed layer-by-layer with each learned hidden code used as the input of the next denoising layer (i.e., the next encoder layer). The learned encoder gets the uncorrupted (clean) preprocessed input at prediction time. Its features are the textual representation provided to cross-feature fusion.

3.3 Image Feature Extraction Module

We used an improved attention-based variational auto-encoder (VAE-ATT) to extract image features. We will want to learn from the data, and do it automatically, from the raw data to the representation of the data. The deep neural network VAE comprises an encoder and decoder. As displayed in Figure 3 the essence of VAE is to extract the hidden features of data and build a model from hidden features to the generated targets. The input data is used to train the encoder into a posterior distribution on latent variables that is not explicitly visible, and latent variables are then sampled. The decoder designates the hidden features to reconstruct the probability distribution of the output, the best approximation to the original one.

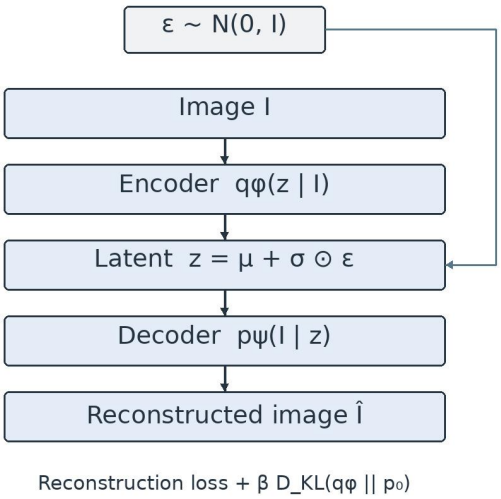


Figure 3. Variational autoencoder with a Gaussian posterior, reparameterized latent sample, and reconstructed image

The VAE encoder has parameters ϕ, and the decoder has parameters ψ. For an image I, the diagonal Gaussian posterior and unit Gaussian prior are

$$q_\phi(z|I) = \mathcal{N}\left(\mu_\phi(I), \mathrm{diag}\left(\sigma_\phi^2(I)\right)\right) \quad p_0(z) = \mathcal{N}(0, I_d) \quad (9)$$

A latent sample uses the reparameterization below, allowing reconstruction gradients to reach the encoder distribution parameters:

$$z = \mu_\phi(I) + \sigma_\phi(I) \odot \epsilon \quad \epsilon \sim \mathcal{N}(0, I_d) \quad (10)$$

The evidence lower bound (ELBO) is maximized in the standard VAE formulation. It is not a positive reconstruction error to be minimized:

$$\mathcal{L}_{\mathrm{ELBO}} = \mathbb{E}_{q_\phi}\left[\log p_\psi(I|z)\right] - D_{\mathrm{KL}}\left(q_\phi(z|I)\|p_0(z)\right) \quad (11)$$

The stated beta-VAE [24] changes the KL coefficient. Equivalently, training minimizes the negative weighted bound:

$$L_{\beta-\mathrm{VAE}} = -\mathbb{E}_{q_\phi}\left[\log p_\psi(I|z)\right] + \beta D_{\mathrm{KL}}\left(q_\phi(z|I)\|p_0(z)\right) \quad (12)$$

The first term is the negative log-likelihood reconstruction cost term – a Bernoulli likelihood reconstructs binary cross-entropy at the right scale on image pixels. The KL term drives the posterior to align with the prior which is Gaussian. This trade-off is regulated by the coefficient β which

can be adjusted, but is not reported in the set-up. The encoder, along with the high level feature maps are fed into the subsequent channel-attention module [25].

3.3.1 Channel Attention

T The image encoder returns a minibatch shape by h by w by c. feature tensor γ. The spatial positions for one image are flattened, such that a = hw. With dimensionally-compatible matrices, the channel attention is presented in Figure 4. Query and key features are obtained from learned 1-by-1 convolutions and features of the original feature map are reshaped to form value features.

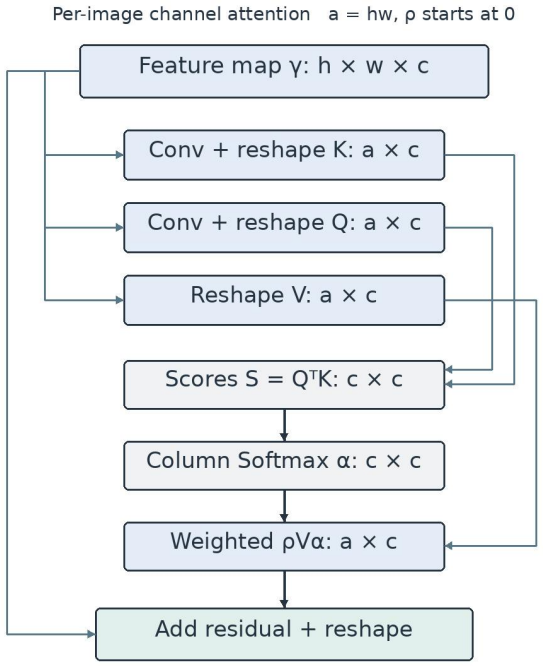


Figure 4. Dimensionally consistent channel-attention module with c-by-c weights and a learnable residual scale

$$Q = reshape(Conv_{1\times1}(\gamma; \omega_Q)) \quad (13)$$
$$Q \in R^{(a\times c)}$$

$$K = \text{reshape}\left(\text{Conv}_{1\times1}(\gamma; \omega_K)\right) \in \mathbb{R}^{a\times c} \quad (14)$$

$$V = \text{reshape}(\gamma) \in \mathbb{R}^{a\times c} \quad (15)$$

The channel scores are made up of Q transpose times K, not of Q times K transpose, because channels correspond to columns of Q and K, and Q times K transpose would yield an a-by-a score matrix:

$$S = Q^{\mathsf{T}}K \in \mathbb{R}^{c\times c}$$
$$S_{ij} = Q_{:,i}^{\mathsf{T}}K_{:,j} \quad (16)$$

For every destination channel j, it normalizes over source channels i. The sum of all the resulting weights is 1 for each column:

$$\alpha_{ij} = \frac{\exp\left(S_{ij}\right)}{\sum_{k=1}^{c}\exp\left(S_{kj}\right)}$$
$$\sum_{i=1}^{c}\alpha_{ij} = 1 \quad (17)$$

At each output channel, the values from the source channels are added to the value of the output channel;

$$O_j = \gamma_j + \rho\sum_{i=1}^{c}\alpha_{ij}V_{:,i} \quad (18)$$
$$\rho(0) = 0$$

The attention correction is then reshaped back to the dimensions of the feature-map. Again the process initializes the parameter σ, namely the learned scale ρ, to zero and keeps the rest of the parameters, namely the independent VAE regularizer β, unchanged. The image-side input into the fusion module are the attention enhanced image features.

3.4 Cross Feature Fusion Module

Assess and modify the problem of cross feature with 3.4 Cross Feature Fusion Module.

Cross-feature fusion asks the one modality to be the principle input E and the other as the secondary input G. Text features in E and image features in G (direction of text main). The roles are reversed in the other direction. Let E be a shape de and G be the shape dg by l. The stated width dv shows for the shared space projections: = 100:

$$E_i^{\text{emb}} = \tanh\left(P_E E_i + b_E\right) \quad (19)$$

$$G_j^{\text{emb}} = \tanh\left(P_G G_j + b_G\right) \quad (20)$$

The bias vectors have width dv, and the projection shapes are dv by de, and dv by dg, with PG being the projection shape and PE the un-projection shape. Each fine-grained affinity is defined by the

main position (i) and auxiliary position (j). Different position indexes are needed: T G j

$$M_{ij} = (E_i^{\text{emb}})^{\top} G_j^{\text{emb}} \qquad M \in \mathbb{R}^{n \times l} \tag{21}$$

The scores are stored in the affinity matrix M while the normalized probabilities are stored in the matrix A. Row-wise Softmax ranks auxiliary positions by importance for each main position:

$$A_{ij} = \frac{\exp(M_{ij})}{\sum_{k=1}^{l} \exp(M_{ik})} \qquad \sum_{j=1}^{l} A_{ij} = 1 \tag{22}$$

The attention-based auxiliary context is

$$J = GA^{\top} \in \mathbb{R}^{d_g \times n} \tag{23}$$

The main vector and its auxiliary context are concatenated and projected to produce one fused vector for each main position:

$$U_i = \tanh(P_U[E_i; J_i] + b_U) \in \mathbb{R}^{d_e} \tag{24}$$

The projection PU has shape de by (de + dg), and bU has width de. Figure 5 shows the data flow for one fusion direction. The same formulation is evaluated with the roles reversed to obtain U t and U v.

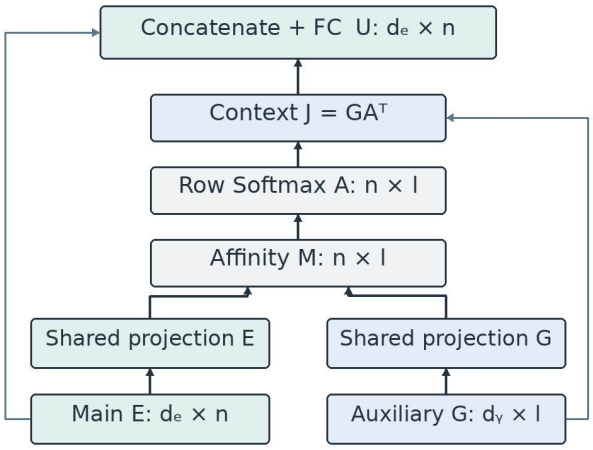


Figure 5. One cross-feature-fusion direction with shared-space projections, row-normalized attention, and auxiliary context

3.5 Output Layer and Supervised Objective

The classifier uses both text-main and image-main fusion outputs. For fixed-size representations, vectorization provides an explicit notation for the concatenation described in the output layer:

$$F = [\text{vec}(U_t); \text{vec}(U_v)] \tag{25}$$

A fully connected classifier produces probabilities over positive, negative, and neutral sentiment:

$$p = \text{Softmax}(W_o F + b_o) \tag{26}$$

The corresponding supervised cross-entropy objective on a labeled minibatch is

$$L_{\text{cls}}(\theta) = -\frac{1}{B} \sum_{r=1}^{B} \sum_{k=1}^{C} y_{r,k} \log p_{r,k} \qquad C = 3 \tag{27}$$

The probabilities are the outputs of the classifier and the labels are one-hot class indicators. Any text padding, masking, and the minimum size of the readout should be the same during training as during inference time. The DAE, beta-VAE and the classifier objectives represent distinct stages of training and it is never assumed that any combinations of these objectives are not supported.

3.6 Training and Prediction Algorithms

This prediction path, containing both the two directions of fusion, is described by algorithm 1. The feature pretraining and supervised fitting is explained with algorithm 2. During prediction, no DAE data will be corrupted or drop out; reported reconstruction and classification stages are not affected.

Algorithm 1. CFF-ATT forward prediction for one text-image pair

Input: Paired text X and image I; learned encoder, attention, fusion, and classifier parameters; documented preprocessing and fixed-size readout.

Output: Three-class probabilities p and predicted sentiment u-hat.

1 Set all modules to evaluation mode and disable training-only dropout.

2 Apply the same text and image preprocessing used for training.

3 Embed the text and pass clean inputs through the learned denoising encoder.

4 Extract image feature maps gamma using the pretrained image encoder.

5 Compute Q, K, and V using Eqs. (13)-(15).

6 Form channel scores and column-normalized alpha using Eqs. (16)-(17).

7 Apply the residual attention correction from Eq. (18); prepare image features.

8 For each direction in {text-main, image-main} do.

9 Assign E and G to the main and auxiliary modality features.

10 Project both inputs and compute affinities M using Eqs. (19)-(21).

11 Compute row-normalized A and context J using Eqs. (22)-(23).

12 Construct fused vectors U using Eq. (24); retain the directional output.

13 End for.

14 Form F from U_t and U_v; compute p using Eqs. (25)-(26).

15 Select u-hat using Eq. (2); return p and u-hat.

Algorithm 2. Feature pretraining and supervised sentiment fitting

Input: Filtered labeled pairs; documented pretraining collection; 8:1:1 split; DAE configuration; beta-VAE configuration; encoder-update policy; stopping criteria and classifier optimizer.

Output: Selected sentiment-model parameters and held-out evaluation predictions.

1 Create disjoint training, validation, and test partitions in the stated ratio.

2 Fit learned text preprocessing on training data; document pretraining sources.

3 For each DAE layer do.

4 Corrupt clean training vectors using the configured Bernoulli masks.

5 Minimize L_DAE from Eq. (8), then retain the learned encoder.

6 Use its hidden codes as training input to the next configured layer.

7 End for.

8 Initialize image-encoder and decoder parameters; set the configured beta.

9 Configure Adam at learning rate 0.001 and batch size 128 for image pretraining.

10 Train the beta-VAE on the specified pretraining collection using Eq. (12).

11 Initialize attention, fusion, and classifier parameters; set rho to zero.

12 Apply the documented freeze/fine-tune policy to the pretrained encoders.

13 While the configured supervised stopping criterion is not met do.

14 For each labeled training minibatch do.

15 Run the CFF-ATT forward path with training-mode dropout enabled.

16 Compute L_cls from Eq. (27).

17 Clear gradients, backpropagate, and update the configured trainable parameters.

18 End for.

19 Evaluate validation predictions with artificial corruption and dropout disabled.

20 Apply early stopping and retain the selected validation checkpoint.

21 End while.

22 Evaluate the selected checkpoint on the held-out test set with Algorithm 1.

23 Return the selected parameters and test predictions.

EXPERIMENTS AND RESULTS

This section presents the experimental results for our model on two different datasets of twitter and comparative experiments with other baseline methods. Finally, we make a visual verification of the effect of the attention mechanism on image feature extraction.

4.1 Datasets and Experimental Setup

Two commonly used multimodal sentiment datasets, MVSA-Single and MVSA-Multiple [26] were used for this experiment. The data set of MVSA-Single consists of 5129 pairs of text images from twitter. The texts and pictures were each annotated by an annotator to one of three sentiments (positive, negative and neutral). The MVSA-Multiple dataset includes 19,600 text-image combination from Twitter. The two databases differ in that each instance of a text-image pair was annotated with one of the three sentiments by three different annotators. The actions taken by each annotator did not have an impact on the others' interpretation of the text or on the image.

When at least two of the three annotators agree on a text or image label, it is retained in MVSA-Multiple. For both sets, if any two numbers are labeled as '+' and '-', they are taken off. If there is a neutral label with a positive or negative label, it's the non-neutral sentiment that takes the neutral label; if there is a neutral label with a neutral label, it's the neutral sentiment that takes the neutral. Following this filtering, the aforementioned has 4,511 MVSA-Single pairs and 17,024 MVSA-Multiple pairs.

The retained pairs are randomly divided into training, validation, and testing partitions in the stated 8:1:1 ratio, corresponding to 80%, 10%,

and 10%. Table 2 summarizes the dataset sizes; exact integer partition counts and the random seed are not supplied.

Table 2. Dataset sizes and the stated partition ratio after sentiment-label filtering

Dataset	Original pairs	Retained pairs	Excluded pairs	Train / validation / test
MVSA-Single	5,129	4,511	618	80% / 10% / 10%
MVSA-Multiple	19,600	17,024	2,576	80% / 10% / 10%

Excluded counts are the number of pairs that have been removed. The percentages in the partition are not actual, independent-counted digits but rather should be interpreted in terms of the ratio.

4.2 Experimental Parameter Settings

The resolution of images is lowered to 224x224 pixels. There are 4 CNN layers: one that creates 2048 feature maps in the image encoder, and a decoder with deconvolutions. The parameters of the image-pretraining procedure and the fusion procedure are summarized in Table 3. To achieve a proper reconstruction objective and to minimize overfitting, a set of dropout and early stopping is introduced, and the KL objective is given in Section 3.3.

Table 3. Reported model settings and implementation details requiring confirmation

Setting	Value or status	Applicable component
Image size	224 x 224 pixels	Image preprocessing
Encoder feature maps	2,048 channels; 10 x 10 spatial positions	Stated CNN image output
Optimizer / learning rate	Adam / 0.001	Beta-VAE pretraining
Batch size	128	Reported beta-VAE stage
Shared-space width	100	Cross-feature projection
Dropout probability	0.3	Feature-fusion module
Attention residual scale	Zero initialization; learned thereafter	Channel-attention correction
Regularization and readout	Beta value, latent width, and readout not specified	Confirm against implementation
Other training settings	DAE settings, classifier optimizer, epochs, and stopping patience not specified	Confirm against implementation

The learning rate and batch size during the image-pretraining stage are not agnostic to all of the training stages. The reported setup does not specify the particular pre-trained dataset nor the pretraining/freeze/finetune methodology.

4.3 Baseline Methods

We evaluated our model against the baseline models listed below with twoMVSA datasets.

There comes a time when it is necessary to isolate the characteristic elements of the image, i.e. the descriptive words used in the text and calculate the sentiment score on the part of text.So there's some point when it matters to separate out the features from the image, the words that describe what is in there, and to calculate the sentiment score from the part of text. [11]

CBOW + DA + LR learned word representations and a denoising auto encoder to embed internal features in unsupervised and semi-supervised manner of the text and image and then linked them to classify the sentiment. [15]

CNN-Multi used two independent CNNs to extract text and image features, and input these features into another CNN for sentimental classification. [20]

DNN-LR trained a neural network for both text and image retrieval, and used this learned neural network model to (re)extract the text and image features and passed them to and from a logistic regression model for sentiment classification. [19]

MultiSentiNet attempted to extract deep semantic features from the image such as the visual features,

object and scene related features, and suggested the use of visual feature-guided attention LSTM model to absorb the deep semantic features extracted from the image that are useful to sentiment analysis. [8]

By exploring the win-win relationship in image and text, CoMN proposed a stacked co-memory network to represent the interaction of the visual and text information iteratively, in order to perform sentimental analysis. [9]

4.4 Evaluation Metrics

Positive, negative and neutral sentiment are judged by accuracy and F1 measure. Accuracy: Percentage of test pairs correctly classified. Given the following equations, let N be the number of test set entries, C = 3 and let the indicator be 1 if the predicted and real label is the same.

$$\text{Accuracy} = \frac{1}{N}\sum_{r=1}^{N} \mathbf{1}\left[\hat{u}_r = u_r\right] \tag{28}$$

This task is one-versus-rest class-wise, giving three classes which leads to precision and recall definition one-versus-rest class-wise. A true positive is correctly predicted by the model; A false positive is an example from a different class as predicted by the model; A false negative is an example from class k that is predicted as a class other than k.

$$\text{Precision}_k = \frac{TP_k}{TP_k + FP_k} \tag{29}$$

$$\text{Recall}_k = \frac{TP_k}{TP_k + FN_k} \tag{30}$$

$$F1_k = \frac{2\text{Precision}_k\text{Recall}_k}{\text{Precision}_k + \text{Recall}_k} \tag{31}$$

A three-class F1 can be aggregated in different ways. Macro F1 weights classes equally, whereas support-weighted F1 uses their test counts:

$$F1_{\text{macro}} = \frac{1}{C}\sum_{k=1}^{C} F\,1_k \tag{32}$$

$$F1_{\text{weighted}} = \sum_{k=1}^{C} \frac{n_k}{N} F1_k \tag{33}$$

It is not possible to determine their averaging convention from the F1 scores which have been reported. They should not be specifically interpreted as the macro or weighted F1 without class level predictions, supports or evaluation records. If the ratio is undefined for a class, then the zero division convention should be explicitly stated.

4.5 Comparative Results

The MVSA scores, as reported, are presented in Table 4. The baseline results reported here are from published papers and have not been rerun here. CoMN(6) is the memory configuration referenced with six memory hops. When statistics are given numerically, the value has not been rounded to the significant figures.

Table 4. Reported accuracy and F1 for baseline methods and CFF-ATT on two MVSA datasets

Method	Single ACC up	Single F1 up	Multiple ACC up	Multiple F1 up
SentiBank + SentiStrength	0.5205	0.5008	0.6562	0.5536
CBOW + DA + LR	0.6386	0.6352	0.6422	0.6373
CNN-Multi	0.6120	0.5837	0.6630	0.6419
DNN-LR	0.6142	0.6103	0.6786	0.6633
MultiSentiNet	0.6984	0.6963	0.6886	0.6811
CoMN(6)	0.7051	0.7001	0.6892	0.6883
CFF-ATT (proposed)	**0.7144**	**0.7106**	**0.6962**	**0.6935**

ACC – accuracy (the higher the better for each column). Note that the scores are in proportion, not percentages. Bold identifies CFF-ATT. The averaging convention for the reported F1 is unspecified.

CFF-ATT has the highest reported accuracy and F1 among the methods listed in Table 4. On

MVSA-Single, its accuracy is 0.7144 and F1 is 0.7106, compared with CoMN(6) at 0.7051 and 0.7001. The absolute gains are 0.0093 in accuracy and 0.0105 in F1, equivalent to 0.93 and 1.05 percentage points on their percentage scales. On MVSA-Multiple, its accuracy is 0.6962 and F1 is 0.6935, compared with 0.6892 and 0.6883 for CoMN(6); the gains are 0.0070 and 0.0052, or 0.70 and 0.52 percentage points.

SentiBank + SentiStrength has the lowest reported F1 on both datasets and the lowest accuracy on MVSA- Single, but it is not lowest in MVSA-Multiple accuracy: CBOW + DA + LR scores 0.6422, below 0.6562 for SentiBank + SentiStrength. The overall ranking must therefore be read metric by metric. CNN-Multi and DNN-LR have lower scores than CBOW + DA + LR on MVSA-Single and higher scores on MVSA-Multiple. That pattern alone does not isolate a causal effect of training-set size, because the datasets and published evaluation protocols may differ.

In MultiSentiNet, the semantic information carried through by the visual features was taken into account and the importance of the image information on the text was emphasized while processing text and images. This way, it captured the sound of the music.That is why it has received good experimental results.

Furthermore, the mutual influence and interactive promotion of text information and image information were thoughts by CoMN during the experiment and CoMN's experimental results was better than MultiSentiNet (only considering the influence of the image information on the text information). The single attention vector of CoMN used, however, introduces a risk of losses in information because it had to be involved in several kinds of information.

CFF-ATT is used to correct the corruption of text and assess the relevance of text-image features prior to performing reciprocal fine-grained fusion. The highest reported benchmark scores agree with good interactions in the modalities. But just without ablations of components and repeated-run estimates, the table does not quantify the net effect of each component or establish a statistical significance.

4.6 Attention Visualization

In the attention comparison, the selected features in the test image are mapped to 2D using t-SNE [27] to compare with the 2D representation of attention. In panel (a) of Figure 6, features are displayed prior to the attention module and in panel (b), features are displayed after attention. A positive sample is shown as a class1 circle; a negative sample is indicated in class2 with a cross; a negative/neutral sample is indicated in class3 with a triangle. In the after-attention panel, it seems that the class groupings are more apparent.

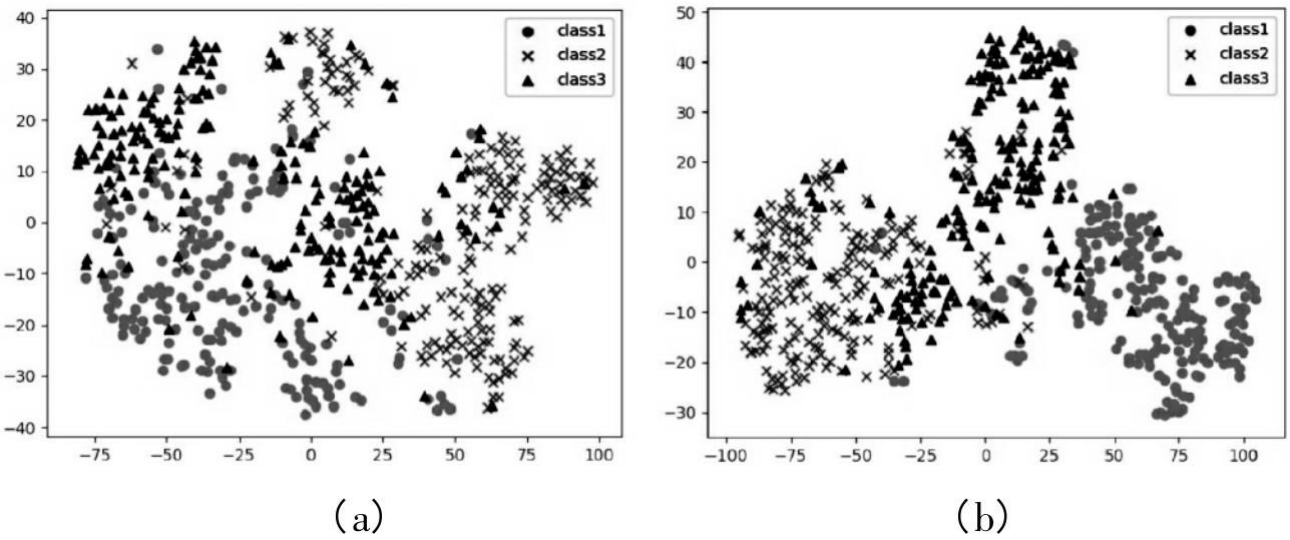


Figure 6. Original t-SNE projection of selected MVSA-Single image features: (a) before attention and (b) after attention

This visualization is qualitative is not exact. No hyperparameters, in-class or between-class dispersion statistics and seed for sampling are provided. The image is not a result of a distance standard deviation reduction or a statistical significance test, therefore. The quantitative assessment is the classification that the results of the quantitative assessment will give.

4.7 Benchmark Visualizations

Figure 7 shows the accuracy-F1 pairs from Table 4 one point for each method, for the negative class and the positive class respectively. The same four benchmark columns as an annotated heatmap are displayed in figure 9. The colors are independent min-max normalized per column, and are not necessarily a newly measured attention weight or a composite sentiment score of the like.

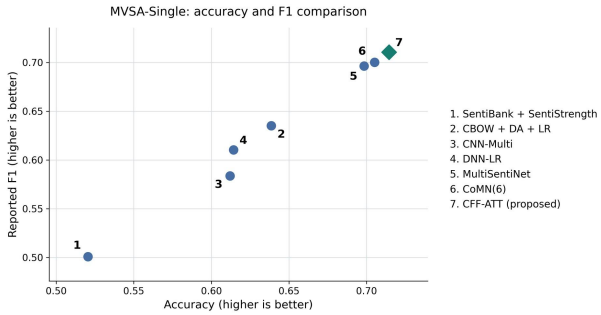


Figure 7. Accuracy versus reported F1 on MVSA-Single; each numbered point represents one method in Table 4

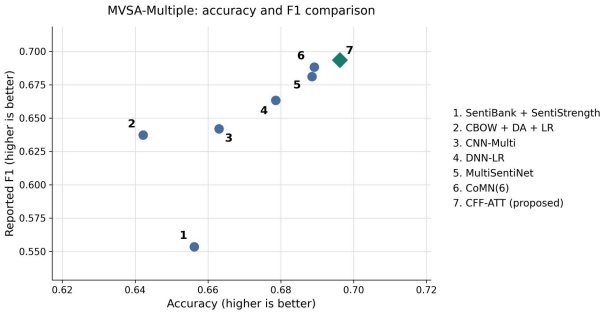


Figure 8. Accuracy versus reported F1 on MVSA-Multiple; each numbered point represents one method in Table 4

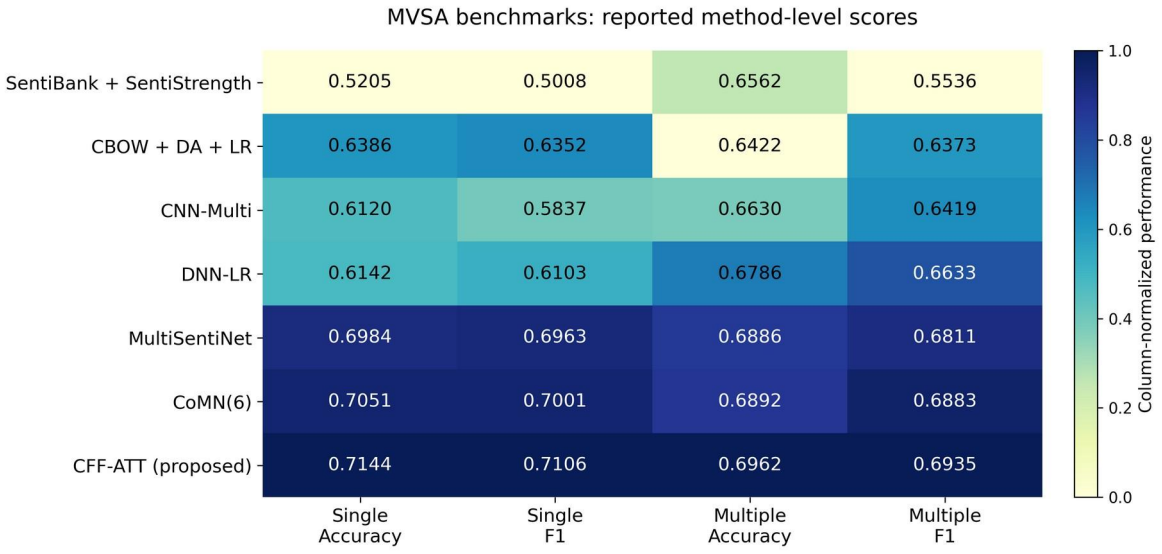


Figure 9. Benchmark heatmap with original numeric annotations and independently normalized columns

These charts are representations of the published method-level aggregates that remain in the manuscript. It does not imply any prediction, correlation, confidence intervals, nor infers anything to do with additional experiments of any individual test.

5. CONCLUSIONS AND FUTURE WORK

Multimodal sentiment analysis of social media is a challenging task. This paper presents a multimodal model of the sentiment analysis based on the attention mechanism. This approach is useful to successfully remove noise from social media's textual information and derive its text properties. The more important images to classify sentiment from the image are extracted, along with the attention mechanism. For feature fusion, attention mechanism is inserted once again, so that features are fused in different ways; learns the interdependence between text and images. The model can successfully extract modal information and modal interaction information to gain the information that is sentimental for multiple modal data, effectively determine the sentential polarity degree of users' tweets, express users' actual emotion better, and better understand the attitude and opinion of the users towards the events that they interested in on social media. We validate our proposed model through experiments on two open datasets: the success and superiority of the model are highlighted.

As a next step we intend to enhance the existing model and process and investigate additional modalities such as audio, video, and others. We will continue to possible more efficient feature extraction and feature fusion techniques for giving more efficient information for sentiment analysis.

Author Contributions: Conceptualization, K.Z.; methodology, K.Z.; software, J.L.; validation, W.L.; formal analysis, Y.G. and K.Z.; investigation, J.L. and W.L.; resources, Y.G. and J.Z.; data curation, J.Z.; writing-original draft preparation, K.Z.; writing-review and editing, Y.G.; visualization, J.L. and W.L.; supervision, J.Z.; project administration, Y.G. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Key R&D Program of China (No. 2019YFB1404700).

Acknowledgments: The authors are very thankful to the editor and referees for their valuable comments and suggestions for improving the paper.

Conflicts of Interest: The authors declare no conflict of interest.

REFERENCES

1. Liu, B. Sentiment analysis and opinion mining. Synth. Lect. Hum. Lang. Technol. 2012, 5, 1-167.
2. Wilson, T.; Wiebe, J.; Hoffmann, P. Recognizing contextual polarity in phrase-level sentiment analysis. In Proceedings of the Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, Vancouver, BC, Canada, 6-8 October 2005; pp. 347-354.
3. Rodger, J.; Murrar, A.; Chaudhary, P.; Foley, B.; Balmakhtar, M.; Piper, J. Assessing American Presidential Candidates Using Principles of Ontological Engineering, Word Sense Disambiguation, and Data Envelope Analysis. Management 2020, 20, 22.
4. Fan, F.; Feng, Y.; Zhao, D. Multi-grained attention network for aspect-level sentiment classification. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October-4 November 2018; pp. 3433-3442.
5. Li, Z.; Wei, Y.; Zhang, Y.; Yang, Q. Hierarchical attention transfer network for cross-domain sentiment classification. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, Hilton, New Orleans Riverside, New Orleans, LA, USA, 2-7 February 2018.
6. Woo, S.; Park, J.; Lee, J.Y.; So Kweon, I. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8-14 September 2018; pp. 3-19.
. Li, X.; Xie, H.; Chen, L.; Wang, J.; Deng, X. News impact on stock price return via sentiment analysis. Knowl.- Based Syst. 2014, 69, 14-23.
. Xu, N.; Mao, W. Multisentinet: A deep semantic network for multimodal sentiment analysis. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, Singapore, 6-10 November 2017; pp. 2399-2402.
. Xu, N.; Mao, W.; Chen, G. A co-memory network for multimodal sentiment analysis. In Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, Ann Arbor, MI, USA, 8-12 July 2018; pp. 929-932.
0. Liu, B.; Zhang, L. A survey of opinion mining and sentiment analysis. In Mining Text Data; Springer: Boston, MA, USA, 2012; pp. 415-463.
1. Borth, D.; Ji, R.; Chen, T.; Breuel, T.; Chang, S.F. Large-scale visual sentiment ontology and detectors using adjective noun pairs. In Proceedings of the 21st ACM International Conference on Multimedia, Barcelona, Spain, 21-25 October 2013; pp. 223-232.
2. Cao, D.; Ji, R.; Lin, D.; Li, S. A cross-media public sentiment analysis system for microblog. Multimed. Syst. 2016, 22, 479-486.
3. Poria, S.; Cambria, E.; Hussain, A.; Huang, G.B. Towards an intelligent framework for multimodal affective data analysis. Neural Netw. 2015, 63, 104-116.
4. Wang, M.; Cao, D.; Li, L.; Li, S.; Ji, R. Microblog sentiment analysis based on cross-media bag-of-words model. In Proceedings of the International Conference on Internet Multimedia Computing and Service, Xiamen, China, 10-12 July 2014; pp. 76-80.

15. Baecchi, C.; Uricchio, T.; Bertini, M.; Del Bimbo, A. A multimodal feature learning approach for sentiment analysis of social network multimedia. Multimed. Tools Appl. 2016, 75, 2507-2525.
16. You, Q.; Luo, J.; Jin, H.; Yang, J. Cross-modality consistent regression for joint visual-textual sentiment analysis of social multimedia. In Proceedings of the Ninth ACM International Conference on Web Search and Data Mining, San Francisco, CA, USA, 22-25 February 2016; pp. 13-22.
17. You, Q.; Cao, L.; Jin, H.; Luo, J. Robust visual-textual sentiment analysis: When attention meets tree-structured recursive neural networks. In Proceedings of the 24th ACM International Conference on Multimedia, Amsterdam, The Netherlands, 15-19 October 2016; pp. 1008-1017.
18. Chen, Y.; Zhang, Z. Research on text sentiment analysis based on CNNs and SVM. In Proceedings of the 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA), Wuhan, China, 31 May-2 June 2018; pp. 2731-2734.
19. Yu, Y.; Lin, H.; Meng, J.; Zhao, Z. Visual and textual sentiment analysis of a microblog using deep convolutional neural networks. Algorithms 2016, 9, 41.
20. Cai, G.; Xia, B. Convolutional neural networks for multimedia sentiment analysis. In Natural Language Processing and Chinese Computing; Springer: New York, NY, USA, 2015; pp. 159-167.
21. Poria, S.; Peng, H.; Hussain, A.; Howard, N.; Cambria, E. Ensemble application of convolutional neural networks and multiple kernel learning for multimodal sentiment analysis. Neurocomputing 2017, 261, 217- 230.
2. Zadeh, A.; Chen, M.; Poria, S.; Cambria, E.; Morency, L.P. Tensor fusion network for multimodal sentiment analysis. arXiv 2017.
3. Vincent, P.; Larochelle, H.; Bengio, Y.; Manzagol, P.A. Extracting and composing robust features with denoising autoencoders. In Proceedings of the 25th International Conference on Machine Learning, Helsinki, Finland, 5-9 July 2008; pp. 1096-1103.
4. Higgins, I.; Matthey, L.; Pal, A.; Burgess, C.; Glorot, X.; Botvinick, M.; Mohamed, S.; Lerchner, A. Beta-Vae: Learning Basic Visual Concepts with a Constrained Variational Framework. 2016.
5. Bengio, Y.; Courville, A.; Vincent, P. Representation learning: A review and new perspectives. IEEE Trans. Pattern Anal. Mach. Intell. 2013, 35, 1798-1828.
6. Niu, T.; Zhu, S.; Pang, L.; El Saddik, A. Sentiment analysis on multi-view social data. In Proceedings of the International Conference on Multimedia Modeling, Miami, FL, USA, 4-6 January 2016; pp. 15-27.
7. Maaten, L.v.d.; Hinton, G. Visualizing data using t-SNE. J. Mach. Learn. Res. 2008, 9, 2579-2605.
8. K. Zhang, Y. Geng, J. Zhao, J. Liu, and W. Li, "Sentiment Analysis of Social Media via Multimodal Feature Fusion," Symmetry, vol. 12, no. 12, article 2010, 2020. doi: 10.3390/sym12122010.