Journal of Ambient Intelligence and Humanized Computing
https://doi.org/10.1007/s12652-022-04447-y

ORIGINAL RESEARCH

Transformer-based models for multimodal irony detection

David Tomás[1] · Reynier Ortega-Bueno[2] · Guobiao Zhang[3] · Paolo Rosso[2] · Rossano Schifanella[4]

Received: 3 March 2022 / Accepted: 4 October 2022

Abstract

Irony is nowadays a pervasive phenomenon in social networks. The multimodal functionalities of these platforms (i.e., the possibility to attach audio, video, and images to textual information) are increasingly leading their users to employ combinations of information in different formats to express their ironic thoughts. The present work focuses on the study of irony detection in social media posts involving image and text. To this end, a transformer architecture for the fusion of textual and image information is proposed. The model leverages disentangled text attention with visual transformers, improving F1-score up to 9% over previous existing works in the field and current state-of-the-art visio-linguistic transformers. The proposed architecture was evaluated in three different multimodal datasets gathered from Twitter and Tumblr. The results revealed that, in many situations, the text-only version of the architecture was able to capture the ironic nature of the message without using visual information. This phenomenon was further analysed, leading to the identification of linguistic patterns that could provide the context necessary for irony detection without the need for additional visual information.

Keywords Irony detection · Transformer · Multimodality · Image text fusion

Both first (David Tomás) and second (Reynier Ortega-Bueno) authors are contributed equally to this manuscript.

✉ David Tomás
dtomas@dlsi.ua.es

Reynier Ortega-Bueno
rortega@prhlt.upv.es

Guobiao Zhang
zgb0537@whu.edu.cn

Paolo Rosso
prosso@dsic.upv.es

Rossano Schifanella
rossano.schifanella@unito.it

[1] Department of Software and Computing Systems, University of Alicante, Alicante, Spain

[2] PRHLT Research Center, Universitat Politècnica de València, Valencia, Spain

[3] School of Information Management, Wuhan University, Wuhan, China

[4] Applied Research on Computational Complex Systems, University of Turin, Turin, Italy

1 Introduction

Irony implies the use of words that mean the opposite of what is really intended.[1] It is a type of expression where the surface sentiment differs from the implied sentiment. Irony is a pervasive phenomena in social media and resolving this challenge is a necessary task for a wide range of applications such as sentiment analysis.

Most approaches to automatic irony detection have tried to address this task as a text classification problem. These approaches rely only on textual information, but in many circumstances the ironic intention also depends on contextual clues, such as images, audio or video. Although there is a large body of literature in the field of irony detection on text, the number of studies in multimodal irony detection is still quite limited.

This work addresses the problem of multimodal irony detection in social networks, considering both textual information and the images associated to those texts. To this end, a deep learning architecture based on textual and visual transformers is proposed. These models have demonstrated to achieve state-of-the-art results in many natural language processing (NLP) tasks.

[1] Irony is used in this paper as an umbrella term for related phenomena such as sarcasm.

Published online: 19 October 2022

The proposed model has been tested with three different datasets from the literature, outperforming previous state-of-the-art systems in this field as well as current visio-linguistic transformer models, which have demonstrated to achieve remarkable results in other multimodal tasks such as fake news (Alam et al. 2021) and hate speech detection (Kiela et al. 2021).

The main contributions of this paper are as follows:

- textual and visual transformers are fused in a deep neural network architecture[2] for the first time in the task of multimodal irony detection;
- the architecture proposed significantly outperforms previous approaches on the three datasets evaluated;
- the proposed system is compared with a baseline that relies on non-contextual word embeddings and Convolutional Neural Networks (CNN) for visual features, with the aim of assessing the performance of transformer models with respect to these traditional approaches;
- the experimental results also reveal that text-only models achieve high performance in this task disregarding visual features. An in depth analysis shows the existence of textual patterns that provide clues of the ironic nature of the posts without requiring visual context.

The rest of the paper is structured as follows: Sect. 2 describes related work in the field of multimodal irony detection. Section 2.2 shows the main features of transformer models; Sect. 3 shows the architecture of the system proposed; Sect. 4 reports the evaluation and discusses the main outcomes of the experiments; finally, conclusions and future work are shown in Sect. 5.

2 Related work

This section summarises the related work in the field of multimodal irony detection and provides an overview of textual, visual, and multimodal transformer models.

2.1 Multimodal irony detection

Although there is a large body of papers in the field of irony detection (Joshi et al. 2017), it is only recently that multimodal settings in this area have been addressed. This section reviews the work done in this scenario, focusing on fusions of text and image used to tackle this challenge.

The work by Schifanella et al. (2016) was the first attempt to create a corpus of text and images for multimodal irony detection. They developed their own corpus on Twitter, Instagram, and Tumblr, running a crowdsourcing task to quantify the extent to which images were perceived as necessary by human annotators to understand the sarcastic nature of the posts. They proposed two different computational frameworks integrating textual and visual modalities. The first one exploited visual semantics trained on an external dataset, concatenating this semantic features with textual features. The second method adapted a visual neural network initialised with parameters trained on ImageNet to multimodal ironic posts.

Cai et al. (2019) presented a corpus of Twitter posts and images for multimodal irony detection. This corpus is freely available to the research community and has been used as a reference in other works, including the present paper (see Sect. 4). The authors proposed a multimodal hierarchical fusion model using three modalities: text features, image features, and image attributes. The model first extracted image features and attributes, and then leveraged these attributes and a bidirectional Long Short-Term Memory (LSTM) network to extract text features. Features of the three modalities were then reconstructed and fused into one feature vector for prediction.

Pan et al. (2020) used the aforementioned Twitter corpus, proposing a contextual language model that concentrated on both intra- and inter-modality incongruity for multimodal irony detection. They designed inter-modality attention to capture incongruity. This attention mechanism was applied to identify the contradiction within text, which was used for the final prediction.

Wang et al. (2020) also used the corpus developed by Cai et al. (2019). They proposed an image-text model for irony detection using the pretrained Bidirectional Encoder Representations from Transformers (BERT) and ResNet models. The vector spaces of BERT and ResNet were connected using the multi-head attention architecture of BERT. The last stage of the model included a 2D-intra-attention layer to extract relationships between words and images.

Finally, another work that leverages this corpus is Xu et al. (2020). They proposed a method for modelling cross-modality contrast in the associated context. Their architecture is composed of two networks: one represents the commonality and discrepancy between image and text, whereas the other models the semantic association in cross-modality context.

It can be observed that previous approaches in the area had focused on developing architectures that mix representation learning by CNN-based architectures with textual information. Although these systems have tried to capture complex cross-modality contrasts through attention mechanisms, none of them has proposed a full transformer-based architecture. As far as the authors know, the system proposed

[2] The source code of the two models proposed is available at https://github.com/reynierortegabueno86/MMID/.

here is the first approach to multimodal irony detection that fully relies on transformed-based fusion.

2.2 Transformer models

Transformers are neural network architectures based on attention mechanisms, dispensing with the use of convolutional and recurrent networks. The original architecture consisted of a multi-head self-attention mechanism combined with an encoder-decoder structure (Vaswani et al. 2017). These models have achieved the state-of-the-art in many natural language processing and computer vision tasks.

In the NLP field, the arrival of BERT supposed a breakthrough in language models (Devlin et al. 2019). This architecture presents a bidirectional encoder that learns information from both left and right side of a word's context during the training phase. The model allows transfer learning in NLP tasks, i.e., the BERT model originally trained on a dataset (the *pre-trained model*) can be used to perform similar tasks on another dataset (the *fine-tuned* model).

DeBERTa (Decoding enhanced BERT with disentangled Attention) (He et al. 2020) is another recent language model, which extends BERT with two novel modifications. Firstly, instead of using the self-attention mechanism, the model proposes a disentangled attention. In DeBERTa every token in the input is represented as two independent vectors that encode its word embedding and position. On this paired representation, disentangled matrices are used to learn the attention weights among tokens. Secondly, an Enhanced Mask Decoder (EMD) is applied to predict the masked tokens during the pre-training phase. Whereas BERT relies on relative positions, the EMD allows DeBERTa to obtain more accurate predictions, as the syntactic roles of the words also depend heavily on their absolute positions in the sentence.

In the same vein, Nguyen et al. (2020) introduced BERTweet, which was the first public large-scale pre-trained language model for English posts obtained from Twitter. This model, having the same architecture as BERT, was trained using the RoBERTa (Liu et al. 2019) pre-training procedure. Experimental results carried out on several NLP tasks in Twitter showed that BERTweet outperformed strong state-of-the-art competitors such as RoBERTa and XLM-R (Conneau et al. 2020). In the sarcasm domain, this model achieved remarkable results on *Task 3: Irony detection in English Tweets* proposed in the context of SemEval 2018 (Van Hee et al. 2018).

In the realm of computer vision, ViT (Vision Transformer) (Dosovitskiy et al. 2021) presented the first attempt to leverage transformer models to solve the problem of image classification. ViT relied on the original transformer model on a sequence of image patches. Instead of a $1D$ sequence of word embeddings, $2D$ image patches are flattened in a vector form and fed to the transformer as a sequence.

Recently, different transformer models have emerged that combine textual and visual features. One of them is VisualBERT (Li et al. 2019), which consists of a stack of transformer layers that implicitly align elements of an input text and regions in an associated input image with self-attention. The experiments showed that this model could ground elements of language to image regions without any explicit supervision.

Along the same lines, LXMERT (Tan and Bansal 2019) is a large-scale transformer model that consists of three encoders: an object relationship encoder, a language encoder, and a cross-modality encoder. The model was pre-trained with large amounts of image and sentence pairs in different tasks: masked language modeling, masked object prediction, cross-modality matching, and image question answering. These tasks helped in learning both intra-modality and cross-modality relationships. After fine-tuning, the model achieved state-of-the-art results on two visual question answering datasets.

3 Architecture of the system

This section introduces the Dual Transformer for Multimodal Irony Detection (DT4MID) architecture proposed in this work. The system relies on a fusion of pretrained unimodal transformer models. An extension of this model (called EDT4MID) is also described, which addresses the problem of multiple textual inputs. Such is the case of textual content extracted from images (e.g. by means of OCR technologies) or tags from specific social media (such as Tumblr) that complement the textual messages posted by users.

The architecture of DT4MID and EDT4MID is depicted in Figs. 1 and 2, respectively. It is worth noting that the same unimodal transformer is used for encoding multiple textual inputs. The main reason to share parameters is to decrease the number of weight updates during back-propagation, consequently reducing memory size and training time.

3.1 Transformer-based representation

Due to the success of transformer models to fit and generalise well on subjective and complex tasks, the architecture proposed here relies on unimodal textual and visual transformers to learn modality-dependent representations.

This proposal uses a BERT-based model to encode textual content. Specifically, three different BERT-like models, already described in Sect. 2.2, were evaluated. The first one is the original BERT model pretrained for English. The

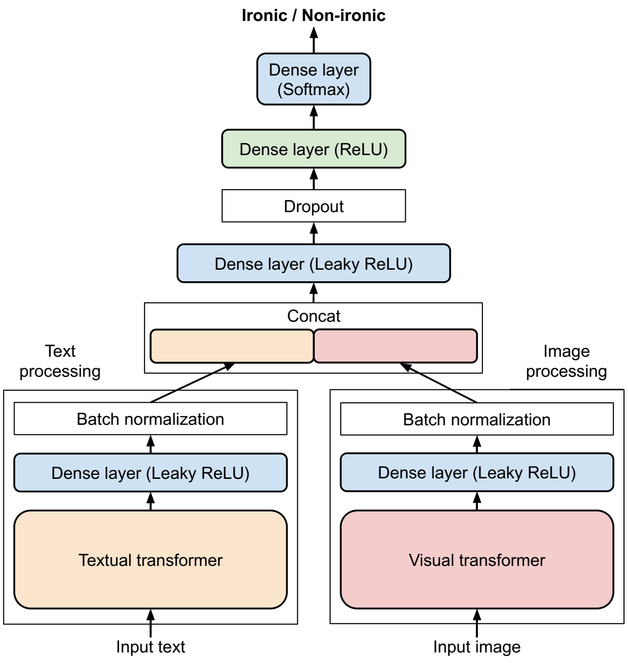


Fig. 1 Dual Transformer for Multimodal Irony Detection (DT4MID) architecture. The model receives one textual and one visual input

second model is DeBERTa, that has proven to be very effective capturing complex semantic and syntactic relationships among words from the input texts. Unlike previous BERT models, DeBERTa disentangles the attention mechanism, using different matrices for computing attention weights associated to the token embeddings and their positional embeddings. This provides the model the ability to infer global dependency relations that could be ignored in BERT, capturing syntactic patterns that have demonstrated to play an important role in irony detection (Cignarella et al. 2020a, b). The third model studied is BERTweet. The reason for this choice is that this model was trained and tuned on a very large corpora of *tweets* written in English. This quality makes it specially suitable on NLP tasks where texts are short and informal.

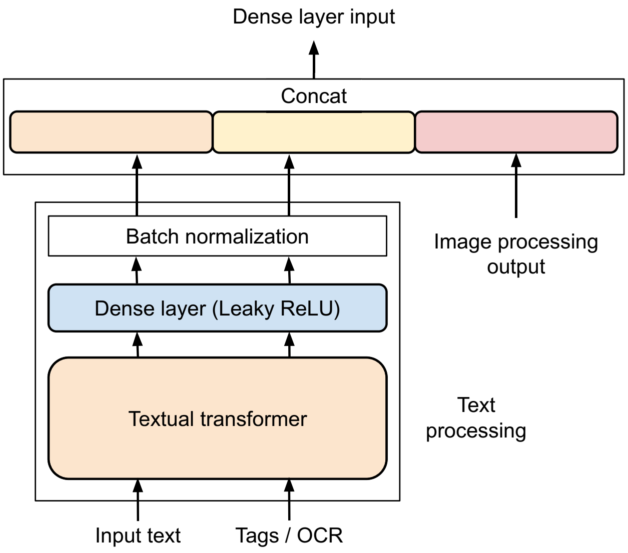


Fig. 2 Extended Dual Transformer for Multimodal Irony Detection (EDT4MID) architecture. Only the changes with respect to DT4MID are shown in the diagram. The model receives the visual input and two different textual inputs in parallel that are processed by the textual encoder sharing the same parameters

Regarding visual content, the proposed architecture uses ViT, since this model has achieved strong performance in several tasks such as object recognition, image segmentation, image classification, and scene understanding (Dosovitskiy et al. 2021). ViT has demonstrated to outperform state-of-the-art CNN architectures in severe occlusions for foreground objects, non-salient background regions, and random patch locations. Moreover, the model shows robustness against other image processing problems such as spatial patch-level permutations, adversarial perturbations, and common natural corruptions (e.g., noise, blur, contrast, and pixelation artefacts) (Naseer et al. 2021).

3.2 Early fusion

The architecture proposed follows an early fusion approach (Gadzicki et al. 2020) to combine all the modalities in DT4MID and EDT4MID. The main idea behind this strategy is to separately learn different feature spaces from the training data, which can thus capture different characteristics that can be beneficial to combine the abstract representations learned by the textual and visual transformers. The goal is to jointly use these representations to retain discriminant information while reducing redundant data.

Let's define CLS_1, CLS_2 and CLS_3 as the two transformer-based textual representations[3] and the transformed-based visual representation, respectively. Each representation is pass through a dense layer to reduce and unify its dimensionality as expressed in the following equation:

$$reduce(CLS_i) = LeakyReLU(W^i CLS_i + b^i), \qquad (1)$$

where W^i and b^i $i = \overline{1, \dots, 3}$ are parameters learned during the training process, and *LeakyReLU* is the Leaky Rectified Linear Unit activation function (Maas et al. 2013). After these representations are reduced ($reduce_1$, $reduce_2$, and $reduce_3$), batch normalisation (Ioffe and Szegedy 2015) is applied before combining them within an aggregation layer as follows:

[3] In the case of DT4MID, there is only one textual input and consequently one transformer.

$$H_0 = aggregate(reduce_1, reduce_2, reduce_3) \tag{2}$$

In the next step, the mixed representation denoted as H_0 is passed through a dense layer with *LeakyReLU* activation to fuse all the deep representations into a new feature space:

$$H_1 = LeakyReLU(W^0 H_0 + b^0) \tag{3}$$

The output of this layer (H_1) is a multimodal encoding of the inputs, which is then passed through a dropout layer (H_2) (Srivastava et al. 2014) to prevent model overfitting:

$$H_2 = \text{ Dropout } (H_1, \delta) \tag{4}$$

The result of the dropout layer is passed through a dense layer with ReLU activation (Agarap 2018):

$$H_3 = max(0, W^2 H_2 + b_2) \tag{5}$$

Finally, the output of this layer (H_3) is provided as an input to a dense layer with two output neurons with the *softmax* function (Goodfellow et al. 2016, Ch. 6, p. 180) to obtain the labels (*ironic* or *non-ironic*):

$$O = softmax(W^3 H_3 + b_3) \tag{6}$$

The EDT4MID architecture can be trained in an end-to-end way using the back-propagation method. To this end, categorical cross-entropy is used as the loss function:

$$\begin{aligned}\mathfrak{L}(\theta) &= \mathbb{E}_{\mathfrak{D}}[\mathfrak{L}(f(x,\theta),y)] \\ &= -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{\|\mathbb{G}\|} y_{ij} * \log(f(x_i,\theta)_j)w_j,\end{aligned} \tag{7}$$

where $\mathfrak{D}$ is the dataset, $\mathfrak{L}$ is the loss function, f is our model parameterised by θ, and $\mathbb{G} = \{1, 0\}$ is the set of labels in the binary classification task.

4 Evaluation

This section describes the evaluation carried out. First, the corpora used are described, including two different multimodal datasets from the literature obtained from Twitter and Tumblr social networks. Next, the baselines employed in the experimentation are depicted. Finally, results and discussion are shown at the end of the section.

4.1 Corpora

The first part of the evaluation focuses on the corpus developed by Cai et al. (2019), which was already mentioned in Sect. 2. The authors collected English tweets containing a picture and special hashtags (e.g. `#sarcasm`) as ironic examples, and without such hashtags as non-sarcastic representatives. The dataset consists of 19,816 tweets for training (8642 positive and 11,174 negative), 2410 for validation (959 positive and 1451 negative) and 2409 for testing (959 positive and 1450 negative), all of them including their textual content and the associated image. This dataset is called *Twitter corpus* in the evaluation section.

The second dataset was described in Schifanella et al. (2016). This corpus comprises two parts. The first one is called the *silver dataset*. In this corpus, data (text and images) were collected from three major social platforms: Instagram, Tumblr, and Twitter. The only corpus to which access was available for reproducing the experiments in this paper was Tumblr. This microblogging platform allows users to post different types of content, including *text* or *photo*. Regardless of the post type, images (one or more) can be added to *text* posts, and captions can be included in *photo* posts. Users can add hashtags to the content by writing them in a separate field.

To collect ironic (positive) examples, they followed a hashtag-based approach by retrieving posts that include the tag `sarcasm` or `sarcastic`. To clean up the data and build the final dataset, different common filters were applied, such as discarding posts without associated images, posts that contained mentions or external links, posts where *sarcasm* or *sarcastic* was a regular word (not a hashtag), and posts whose text contained less than four regular words. To build the final dataset, 10,000 sarcastic posts were randomly sampled. Another 10,000 negative examples were collected by randomly sampling posts that do not contain *sarcasm* or *sarcastic* in either the text or the tag set. In the experiments below, this dataset is called the *Tumblr silver corpus*.

The second part of the dataset in Schifanella et al. (2016), called *gold dataset*, is a curated corpus whose ironic nature was agreed on by a group of human annotators that considered the textual and visual components as required to decode the sarcastic tone. A crowdsourcing platform was used to perform the annotation task on 1000 positive samples (5 annotators for each post). The users were asked to annotate the posts as `text only` (the text was enough to identify the nature of the post as ironic) and `text + image` when both modalities were required. From the 1000 posts, 319 were labelled as non-ironic, 236 were considered a `text only` ironic, and 445 were `text + image` ironic. Only the dataset comprising `text + image` posts were used to further evaluate the system.

Depending on the agreement achieved by the annotators on whether both text and image were required, three subsets were created: *D-50* (50% agreement, 445 samples), *D-80* (80% agreement, 197 samples) and *D-100* (100% agreement, 141 samples). In Sect. 4.4, only the *D-100* subset (full agreement) is used. This dataset is referred to as *Tumblr gold corpus* in the experiments. It is important to highlight that we used the same corpus proposed in Cai et al. (2019) and the corpus introduced in Schifanella et al. (2016), to make a fair comparison with the previous works in state of the art.

4.2 Baselines

The proposed DT4MID and EDT4MID models are based on the transformer architecture. In order to compare their performance with other neural network models, the proposal described in Giachanou et al. (2020) has been adopted as a baseline. It was originally intended for multimodal fake news detection and has been adapted in the present work for irony detection. This model is based on a neural network architecture that combines textual and visual features. The textual information is encoded using static word embeddings (Word2vec) (Mikolov et al. 2013). Additionally, the textual feature vector also includes the sentiment expressed in the post, extracted using the Valence Aware Dictionary for sEntiment Reasoning (VADER) (Hutto and Gilbert 2014).

The visual information is extracted from the image of the posts and includes image tags and Local Binary Patterns (LBP) (Ojala et al. 2002), a texture operator that has proven to be very effective in many visual tasks, such as face recognition. Image tags are extracted using different pretrained CNN-based models, such as VGG19 and ResNet. The top ten image tags for every pretrained model are encoded using Word2vec. The system calculates the similarity between text and image by computing the cosine between the word embeddings of the text and the word embeddings of the image tags extracted from visual features. The final input vector is the result of concatenating all the features mentioned in the previous paragraphs. A fully connected neural network is then feed with this information to produce the output of the system.

This approach is referred to as *Baseline* in the experiments below and have been tested on the three datasets mentioned in the previous section. In the case of *Twitter corpus*, the results of previous works (Cai et al. 2019; Xu et al. 2020; Wang et al. 2020; Pan et al. 2020) have been also included as a reference in the evaluation. For the *Tumblr silver corpus* and *Tubmlr gold corpus*, the results of their authors (Schifanella et al. 2016) are also provided. All these approaches were described in Sect. 2.

Additionally, two visio-linguistic transformers described in the related work have been also tested and compared in the experiments carried out: VisualBERT and LXMERT. The first one is representative of the single-stream models, where images and text are jointly processed by a single encoder. The second one belongs to the dual-stream category, where inputs are encoded separately before being jointly modelled. The two models were fine-tuned using the same datasets as the DT4MID and EDT4MID architectures. The number of epochs was set to 30 and the minibatch size to 8. Both RMSprop and Adam optimisers were evaluated. Only the best configuration results are shown in the evaluation section.

4.3 Training procedure

The proposed architecture was implemented using PyTorch[4] and the pre-trained models in the Huggingface[5] library. The models used and compared for textual encoding were *bert-uncased*, *deberta-base* and *bertweet-base*. The visual encoding was carried out using *vit-base*.

The models were fine-tuned following an end-to-end approach, where all weights were learned together. Specifically, the strategy adopted was that proposed in ULMFiT (Howard and Ruder 2018) for tuning pre-trained models in a gradual unfreezing-discriminative fashion. This strategy has outperformed the standard schema for fine-tuning transformer models on different NLP tasks. Taking that into account, different learning rates were assigned for each layer in the model, increasing the rate as the neural network gets deeper, i.e $\alpha_i = \alpha_0 + \lambda_i$ and $\lambda_i = (\delta_{lr})^{layers}$, where α_i is the learning rate of the i^{th} layer, λ_i is the increasing of the learning rate to compute α_i from α_0 and δ_{lr} is maximum increase permitted. Thus, the shallowest layers that receive the input message have the lowest increase, whereas the layers at the top of the model receive a greater learning rate. This dynamic learning rate keeps most information from the shallow layers and biases the deeper ones to learn about the target tasks. The learning rate updating procedure is summarised in Algorithm 1.

Algorithm 1 Learning rate schedule

```
    Input: $lr$ (Learning rate)
    Input: $mlr$ (Minimum learning rate)
Require: $lr > 0 \wedge mlr > 0$
Ensure: $lr = mlr$
  $i \Leftarrow 12$
  $\alpha_0 \Leftarrow mlr$
  $\delta_{lr} \Leftarrow abs(lr - mlr)$
  while $i \geq 1$ do
      $\lambda_i \Leftarrow (\delta_{lr})^i$
      $\alpha_i \Leftarrow \alpha_0 + \lambda_i$
      $i \Leftarrow i - 1$
  end while
```

[4] https://pytorch.org/.

[5] https://huggingface.co/.

Table 1 Experiments on the *Twitter corpus*

Approach	F1-score	Accuracy
Baseline	0.7800	0.7870
Cai et al. (2019)	0.8018	0.8344
Xu et al. (2020)	0.8060	0.8402
Wang et al. (2020)	0.8605	0.8851
Pan et al. (2020)	0.8618	0.8875
VisualBERT	0.8513	0.8581
LXMERT	0.8164	0.8248
DT4MID		
BERT (only text)	0.8309	0.8360
DeBERTa (only text)	0.9372	0.9394
BERTweet (only text)	0.8941	0.8983
ViT (only image)	0.7050	0.7148
BERT + ViT	0.8516	0.8555
DeBERTa + ViT	**0.9377**	**0.9398**
BERTweet + ViT	0.8956	0.8995
EDT4MID		
BERT + ViT + OCR	0.8445	0.8506
DeBERTa + ViT + OCR	0.9368	0.9390
BERTweet + ViT + OCR	0.8974	0.9012

Best results for each measure are boldfaced

Table 2 Experiments on the *Tumblr silver corpus*

Approach	F1-score	Accuracy
Baseline	0.7650	0.7670
Schifanella et al. (2016)	–	0.810
VisualBERT	0.7874	0.7957
LXMERT	0.7624	0.7728
DT4MID		
BERT (only text)	0.7714	0.7771
DeBERTa (only text)	0.7972	0.8077
BERTweet (only text)	0.8181	0.8252
ViT (only image)	0.7200	0.7241
BERT + ViT	0.7884	0.8014
DeBERTa + ViT	0.8297	0.8365
BERTweet + ViT	0.8105	0.8260
EDT4MID		
BERT + ViT + tags	0.8421	0.8483
DeBERTa + ViT + tags	**0.8485**	**0.8563**
BERTweet + ViT + tags	0.7677	0.7765

Best results for each measure are boldfaced

Regarding the number of hidden neurons, in the reduction layer it was set to 64, in the first dense layer after the aggregation layer to 64, in the second to last layer to 32 neurons, and the last layer contained only one neuron. The *dropout* was set to 0.30. The hyperparameters of *batch size* and *maximum sequence length* were constrained by the hardware resources available. Based on that, the sequence length was set to 100 and mini-batches to 10. In an intent to prevent overfitting in the training step, no fixed number of *epochs* was defined. Instead, the process relied on the *early stopping* criteria, setting the value of *patience* to 15 and the maximum number of epochs to 50. *RMSprop* and *Adam* optimisation rules were both evaluated during the training phase.

4.4 Results

The evaluation was made using two standard metrics: accuracy and F1-score. One of the reasons for this decision was to be able to compare this work with previous approaches on the same benchmarks. Table 1 shows in the upper part the results obtained by the baseline described in Sect. 4.2, the previous work in the field, and the visio-linguistic transformers (VisualBERT and LXMERT). The different variants of the proposed architectures (DT4MID and EDT4MID) are shown in the lower part of the table. BERT (only text), DeBERTa (only text) and BERTweet (only text) refer to the DT4MID architecture including only the textual transformer. ViT (only image) refers to the DT4MID architecture using only the visual transformer, whereas BERT + ViT, DeBERTa + ViT and BERTweet + ViT cover the full DT4MID architecture combining textual and visual transformers. Finally, BERT + ViT + OCR, DeBERTa + ViT + OCR and BERTweet + ViT + OCR refer to the extended architecture EDT4MID with multiple textual inputs, including in this case an OCR analysis to extract embedded text from images. This configuration follows the idea proposed by Pan et al. (2020), where the system obtained a boost in performance by including OCR features in their model (their system improved from 0.8292 F1-score to the final 0.8618 reported in Table 1). The general purpose Tesseract OCR engine[6] was used to analyse the 19,816 images in the dataset. Tesseract was able to extract text from 56.35% of the images.

The baseline proposed obtained results close to the original paper (0.787 vs 0.8018 F1-score). The text-only version DT4MID using DeBERTa and BERTweet performed extremely well (0.9372 and 0.8941 F1-score respectively), surpassing all the previous multimodal approaches on this dataset. DeBERTa improved almost 9% the performance of the best previous system (F1-score = 0.8618). VisualBERT (using RMSprop optimiser) was the best visio-transformer model, outperforming the previous state-of-the-art approach in terms of precision but not recall, which led to a lower F1-score (0.8513).

The image-only transformer version using ViT achieved F1-score = 0.7050, significantly lower than the text-only versions of the model. When both modalities were combined, the text-only version of BERT significantly improved

[6] https://tesseract-ocr.github.io/.

Table 3 Experiments on the *Tumblr gold corpus*

Approach	F1-score	Accuracy
Baseline	0.9000	0.8200
Schifanella et al. (2016)	–	0.897
VisualBERT	**0.9754**	**0.9521**
LXMERT	0.9382	0.8836
DT4MID		
BERT (only text)	0.9343	0.8767
DeBERTa (only text)	0.9682	0.9384
BERTweet (only text)	0.9420	0.8904
ViT (only image)	0.8594	0.7534
BERT + ViT	**0.9754**	**0.9521**
DeBERTa + ViT	0.9304	0.8699
BERTweet + ViT	**0.9754**	**0.9521**
EDT4MID		
BERT + ViT + tags	**0.9754**	**0.9521**
DeBERTa + ViT + tags	0.9645	0.9315
BERTweet + ViT + tags	0.9496	0.9041

Best results for each measure are boldfaced

its performance (2.5% higher), but DeBERTa and BERTweet marginally improved it. In the case of the EDT4MID architecture including OCR features, the results did not significantly improve the performance of the corresponding *text + image* models, slightly worsening them in the case of BERT and DeBERTa. It is worth noting that Tesseract did not extract text in almost half of the cases, and that this text is not always a reliable representation of what can be found in the image.

The second corpus evaluated is the *Tumblr silver corpus*. Table 2 shows the results obtained in the experiments carried out, which are the same conducted in the previous corpus. The only difference is the comparison with related work, which in this case is limited to Schifanella et al. (2016). Since the authors reported two approaches, only the best results are included here, corresponding to the information fusion with SVM approach.

Taking into account that there was no improvement using OCR features in the previous experiments, this information was discarded in this corpus. Instead, a new feature was added that corresponds to the list of tags that Tumblr users added to their posts. These tags were treated separately from text, as an additional textual input in the EDT4MID architecture.

The results show that the baseline performed worse (accuracy = 0.7670) than the original system provided in Schifanella et al. (2016) (accuracy = 0.810).[7] The best visio-linguistic model was again VisualBERT (using Adam optimiser), although in this case the results did not improve the previous work (accuracy = 0.7957). Regarding the text-only version of the architecture, the best performing model was BERTweet (accuracy = 0.8252), followed by DeBERTa (accuracy = 0.8077). BERTweet improved the original system and DeBERTa achieved almost its performance. Again, using only the visual transformer performed comparatively lower than the text-only version (accuracy = 0.7241).

Table 4 Training time (hours and minutes) of DT4MID and EDT4MID models on the three datasets studied: *Twitter* (TW), *Tumblr silver* (TS) and *Tumblr gold* (TG)

Approach	TW	TS	TG
DT4MID			
BERT (only text)	00:29	00:11	00:27
DeBERTa (only text)	00:36	00:13	00:31
BERTweet (only text)	00:37	00:09	00:24
ViT (only image)	05:29	01:00	01:25
BERT + ViT	07:56	01:33	02:37
DeBERTa + ViT	05:06	01:33	02:15
BERTweet + ViT	08:48	01:33	01:59
EDT4MID			
BERT + ViT + OCR/tags	06:14	02:17	02:21
DeBERTa + ViT + OCR/tags	05:29	01:43	02:24
BERTweet + ViT + tags	04:43	01:24	02:34

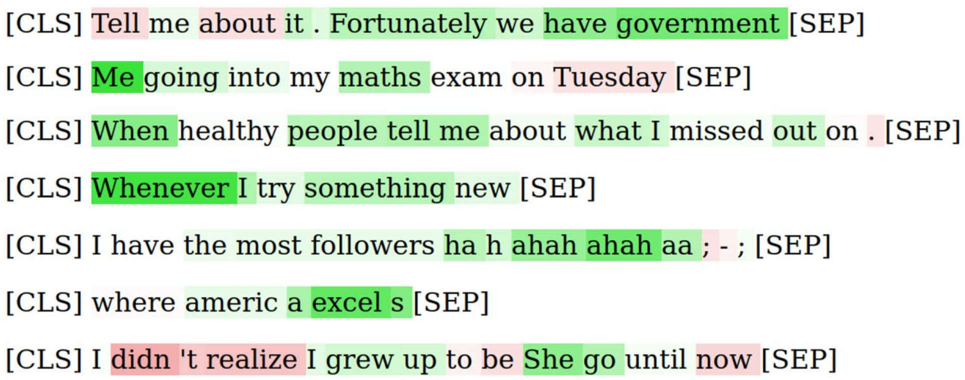
When both textual and visual features were considered the system significantly improved its performance in the case of DeBERTa (accuracy = 0.8365) and BERT (accuracy = 0.8014), but not for BERTweet. The best results were obtained when tags were included, improving 2.4% the accuracy of DeBERTa + ViT and 5.9% in the case of BERT + ViT. Globally, DeBERTa + ViT + tags obtained the best results, improving almost 6% the performance provided by Schifanella et al. (2016).

The last corpus used in the experiments is the *Tumblr gold corpus*. Unlike the previous corpora, this is a curated dataset that can provide more reliable insights on the role of text and image in multimodal irony detection. Table 3 shows the results obtained using the same architectures and variants used on the *Tumblr silver corpus*.

In this experiment, the models were trained on the whole *Tumblr silver corpus* and tested on the *Tumblr gold corpus* (*D-100* subset with unanimous inter-annotator agreement). This corpus contains only positive samples (i.e., ironic tweets), thus the performance measures provided in Table 3 only consider this class.

In these experiments, the text-only version of DeBERTa already provided a significant improvement in performance with respect to the best model presented by Schifanella et al. (2016) (again, the authors only showed accuracy values). DeBERTa improved the original result (accuracy = 0.897)

[7] Accuracy was the only measure provided in their work.



[CLS] Tell me about it . Fortunately we have government [SEP]

[CLS] Me going into my maths exam on Tuesday [SEP]

[CLS] When healthy people tell me about what I missed out on . [SEP]

[CLS] Whenever I try something new [SEP]

[CLS] I have the most followers ha h ahah ahah aa ; - ; [SEP]

[CLS] where americ a excel s [SEP]

[CLS] I didn 't realize I grew up to be She go until now [SEP]

Fig. 3 Contribution of each token to the model's output

by almost 5% (accuracy = 0.9384). The best proposed architecture in this subset was BERT + ViT together with BERTweet + ViT, which achieved the same performance. As in the previous experiments, BERT significantly benefited from the addition of visual features, improving almost 9% the text-only version. The same results were obtained by VisualBERT (using Adam optimiser). Adding images to text improved the performance in the case of BERT and BERTweet, but not for DeBERTa, which reduced its performance. The image-only version of the architecture performed significantly lower than the text-only transformers (accuracy = 0.7534). Finally, adding tags did not enhance the results of BERT + ViT and BERTweet + ViT, but improved DeBERTa + ViT by 7%.

Regarding processing time, experiments involving DT4MID and EDT4MID models were carried out on a computer with a CPU Intel(R) Core(TM) i7-7800X 3.50GHz, 126GB RAM and GPU NVIDIA GeForce RTX 2080 8GB. Training time for each model in the three datasets studied is shown in Table 4. LXMERT and VisualBERT required a more powerful computer for training, with GPU NVIDIA Titan RTX 24GB, whereas the baseline was trained on CPU. For this reason, the training time of these three models is not included in the aforementioned table, since using a different hardware configuration prevents a fair comparison between them. As can be observed, training time increases from text to image models and from unimodal to multimodal models.

4.5 Discussion

The most remarkable result in the experiments carried out on the *Twitter corpus* is the high performance obtained by the text-only versions of the transformers analysed. In the case of DeBERTA, the DT4MDI architecture improved almost 9% the previous state-of-the-art in this task. In this model, adding the information from the visual transformer did not significantly improve the performance of the system. These results could reflect that part of the multimodal posts in this corpus do not necessarily require from an image to determine the ironic nature of the message, and considering only the text is effective in some situations (recall that this dataset was automatically retrieved from Twitter and lacks of human supervision).

To further investigate this point, a random sample of 100 posts were selected from the *Twitter corpus* and manually analysed by three annotators. The results revealed that, on average, almost 40% of the posts were considered as ironic by looking only at text. The reasons for that vary. In some cases, there were hashtags such as *#lolsarcasm* or *#funnymemes* that provide a clear hint of the nature of the post. It was also interesting the presence of specific topics that can reveal the ironic intention, such as combining government affairs with positive vocabulary (e.g. "Tell me about it. Fortunately we have #government" and "Oh, this is a totally fair move that really shows that the government is doing it right!"). Finally, some posts that could be strictly considered as non-ironic in the surface, reveal their ironic intention to regular users of social networks. For instance, "My life in a picture" or "Me reflecting on the weekend" are usually followed by an image that shows the implicit ironic intention of the user. Thus, textual transformers could be learning this type of linguistic patterns in text, making unnecessary the presence of an image to correctly classify them as ironic, and therefore justifying the high accuracy of the text-only models and the lack of improvement when image context is added.

A subsequent manual analysis of the *Tumblr gold corpus* confirmed the presence of these patterns in the corpora. In addition to those, it was also common the presence of posts with "When" / "Whenever" at the beginning of the sentence. E.g. "When healthy people tell me about what I missed out on." and "Whenever I try something new". The use of very positive claims was another common pattern in ironic posts. For instance: "Tumblr mobile works perfectly" and "Where America excels".

A set of these examples were further analysed using the Captum[8] library for model interpretability. Figure 3 shows

[8] https://captum.ai/.

the contribution of each token to irony identification using the text-only version of DeBERTa. Green colour indicates positive contribution of tokens towards the predicted class, whereas red shows tokens contributing negatively. The intensity of the colour signifies the magnitude of the contribution. All the examples shown, except the last one, were correctly classified as ironic.

These samples exemplify all the patterns mentioned before. It is clear the contribution of "Fortunately ... government", "Me going", "When ... people", "Whenever", "hahahahahahah" and "excel" to the class prediction. In the last example, wrongly classified by the model as non-ironic, no clear linguistic pattern can be identified.

In the *Tumblr silver corpus* the combination of textual and visual transformers outperformed the text-only version of the architecture. DeBERTa + ViT + tags obtained the best results, improving more than 6% the F1-score of DeBERTa. Similar results were obtained for BERT, where BERT + ViT + tags improved 9% the F1-score of the text-only BERT. In the case of BERTweet, the performance of the model did not significantly improved by adding the visual transformer to the text-only version. In general, the text-only models performed significantly lower than in the previous experiment. For instance, DeBERTa obtained F1-score = 0.7972, almost 15% drop in performance.

This fact reflects that this corpus is more challenging for the models than the previous one. One reason for this variation is the different nature of the posts in Twitter and Tumblr. In the latter, posts tend to be longer and more elaborated than in Twitter, which can make it more challenging to identify the ironic nature of the text. In the *Twitter corpus*, the average length of the posts is 89 characters, whereas in the *Tumblr silver corpus* this value rises up to 273.

The best text-only results in the *Tumblr gold corpus* were obtained again by DeBERTa (F1-score = 0.9682). Taking into account the high performance of the textual transformer, it was expected that the addition of a visual features in the ensemble did not improve the final results. In fact, not only did ViT not improve the results, it actually significantly reduced the performance. Unlike DeBERTa, the combination of BERT and BERTweet with ViT significantly improved the text-only version to achieve the best performance in this corpus. In general, performance was higher than in both previous datasets. The curated nature of this corpus reveals the shortcomings of the automatic data collection procedures for irony detection, which lead to lower performance in machine learning approaches.

The relative performance of the models is stable across the three datasets, with the two proposed architectures systematically improving the results of the baselines. One of the main conclusions of this set of experiments is that the text-only version of the models, properly fine-tuned, can achieve high performance in multimodal irony detection. This reflects that in many occasions the text contains linguistic patterns that give clear clues of the ironic nature of the post without requiring additional visual context, as shown in the examples provided above.

Regarding the comparison between textual transformers and previous static word embeddings, transformers clearly outperformed the Word2vec model used in the baseline architecture in all the experiments. In the case of the text-only version of DeBERTa, the F1-score improved over 20% for the *Twitter corpus*, over 4% in the *Tumblr silver corpus* and almost 8% for the *Tumblr gold corpus*. Textual transformers also surpassed the proposal by Cai et al. (2019) using GloVe and the approach by Xu et al. (2020) that relied on a BiLSTM network to represent textual sequences.

Finally, in terms of the performance of current CNN-based networks compared to visual transformers, the experiments in Table 1 revealed that both the architecture proposed (using ViT) and the visio-linguistic transformers (VisualBERT and LXMERT) outperformed the baseline using ResNet. The proposed system using ViT also outperformed previous works in the field shown in this table, which also relied on ResNet to analyse the visual content.

5 Conclusions and future work

This paper presented a deep learning architecture that combines textual and visual transformers for the task of irony detection. Unlike previous approaches, the architecture proposed fully rely on transformers for both textual and visual inputs, performing an early fusion approach to integrate both types of media.

The architecture was evaluated in three different corpora and compared with different baselines, including current visio-linguistic transformer models. As far as the authors know, this is the first attempt to apply and evaluate these models in the task of multimodal irony detection.

The evaluation results revealed that the architecture proposed significantly outperformed (up to 9%) current state-of-the-art systems and all the baselines proposed in this task. The most remarkable finding was the high level of performance of the text-only models, being DeBERTa the most salient of them. These results lead to the conclusion that, in some situations, the image provided as context for the multimodal post is not necessary to catch the ironic nature of the message, even in situations when a group of humans could unanimously determine that it is required to understand its sarcastic meaning. Transformer models were able to identify linguistic patterns in text that were leveraged to identify the irony without further context.

In the case of the *Twitter corpus* and the *Tumblr gold corpus*, the performance of the text-only transformers was so high that the addition of visual information did not improve the results. In the case of *Tumblr silver corpus*, the task resulted more challenging and adding the visual transformer significantly improved the results.

The present study has shown that multimodal irony is a subtle phenomenon, where textual input provides more information about their ironic nature than expected. As a future work, it is mandatory in the first place to carry out a deeper quantitative and qualitative study of the nature of multimodal irony in order to precisely identify the type of posts that really benefit from adding contextual information in the form of images. To this end, a study comparing unimodal (text-only) and multimodal messages could reveal the real differences existing between these two modalities.

Funding Information Open Access funding provided thanks to the CRUE-CSIC agreement with Springer Nature. This work was partially supported by the Spanish Ministry of Science and Innovation and Fondo Europeo de Desarrollo Regional (FEDER) in the framework of project "Technological Resources for Intelligent VIral AnaLysis through NLP (TRIVIAL)" (PID2021-122263OB-C22).

Declarations

Conflict of interest The authors have no competing interests to declare that are relevant to the content of this article.

References

Agarap AF (2018) Deep learning using rectified linear units (ReLU). arXiv:1803.08375

Alam F, Cresci S, Chakraborty T, et al (2021) A survey on multimodal disinformation detection. arXiv:2103.12541

Cai Y, Cai H, Wan X (2019) Multi-modal sarcasm detection in Twitter with hierarchical fusion model. In: Proceedings of the 57th annual meeting of the ACL. Association for Computational Linguistics, pp 2506–2515. https://doi.org/10.18653/v1/P19-1239

Cignarella AT, Basile V, Sanguinetti M, et al (2020a) Multilingual irony detection with dependency syntax and neural models. In: Proceedings of the 28th international conference on computational linguistics. International Committee on Computational Linguistics, Barcelona, Spain (Online), pp 1346–1358. https://doi.org/10.18653/v1/2020.coling-main.116

Cignarella AT, Sanguinetti M, Bosco C, et al (2020b) Marking irony activators in a Universal Dependencies treebank: the case of an Italian Twitter corpus. In: Proceedings of the 12th language resources and evaluation conference. European Language Resources Association, Marseille, France, pp 5098–5105. https://aclanthology.org/2020.lrec-1.627

Conneau A, Khandelwal K, Goyal N, et al (2020) Unsupervised cross-lingual representation learning at scale. In: Proceedings of the 58th annual meeting of the association for computational linguistics. Association for Computational Linguistics, Online, pp 8440–8451. https://doi.org/10.18653/v1/2020.acl-main.747

Devlin J, Chang MW, Lee K, et al (2019) Bert: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 conference of the north American chapter of the association for computational linguistics. Association for Computational Linguistics, pp 4171–4186. https://doi.org/10.18653/v1/N19-1423

Dosovitskiy A, Beyer L, Kolesnikov A, et al (2021) An image is worth 16x16 words: transformers for image recognition at scale. In: International conference on learning representations, pp 1–21. https://openreview.net/forum?id=YicbFdNTTy

Gadzicki K, Khamsehashari R, Zetzsche C (2020) Early vs late fusion in multimodal convolutional neural networks. In: 2020 IEEE 23rd international conference on information fusion (FUSION), pp 1–6. https://doi.org/10.23919/FUSION45008.2020.9190246

Giachanou A, Zhang G, Rosso P (2020) Multimodal fake news detection with textual, visual and semantic information. Text, speech, and dialogue. Springer, Cham, pp 30–38

Goodfellow I, Bengio Y, Courville A (2016) Deep learning. MIT press, Oxford

He P, Liu X, Gao J, et al (2020) Deberta: decoding-enhanced BERT with disentangled attention. arXiv:2006.03654

Howard J, Ruder S (2018) Universal language model fine-tuning for text classification. In: Proceedings of the 56th annual meeting of the ACL. Association for Computational Linguistics, pp 328–339. https://doi.org/10.18653/v1/P18-1031

Hutto C, Gilbert E (2014) Vader: a parsimonious rule-based model for sentiment analysis of social media text. Proc Int AAAI Conf Web Soc Media 8(1):216–225

Ioffe S, Szegedy C (2015) Batch normalization: accelerating deep network training by reducing internal covariate shift. In: International conference on machine learning, PMLR, pp 448–456

Joshi A, Bhattacharyya P, Carman MJ (2017) Automatic sarcasm detection: a survey. ACM Comput Surv 50(5):1–22. https://doi.org/10.1145/3124420

Kiela D, Firooz H, Mohan A, et al (2021) The hateful memes challenge: competition report. In: Escalante HJ, Hofmann K (eds) Proceedings of the NeurIPS 2020 competition and demonstration track, proceedings of machine learning research, vol 133. PMLR, pp 344–360

Li LH, Yatskar M, Yin D, et al (2019) Visualbert: a simple and performant baseline for vision and language. arXiv:1908.03557

Liu Y, Ott M, Goyal N, et al (2019) Roberta: a robustly optimized bert pretraining approach, pp 1–13. arXiv preprint arXiv:1907.11692

Maas AL, Hannun AY, Ng AY (2013) Rectifier nonlinearities improve neural network acoustic models. In: Proceedings of the ICML workshop on deep learning for audio, speech and language processing, Atlanta, Georgia, USA, pp 1–6

Mikolov T, Sutskever I, Chen K, et al (2013) Distributed representations of words and phrases and their compositionality. In: Proceedings of the 26th international conference on neural information processing systems, vol 2. Curran Associates Inc., NIPS'13, pp 3111–3119

Naseer M, Ranasinghe K, Khan S, et al (2021) Intriguing properties of vision transformers. arXiv:2105.10497

Nguyen DQ, Vu T, Tuan Nguyen A (2020) BERTweet: a pre-trained language model for English tweets. In: Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations. Association for Computational

Linguistics, Online, pp 9–14. https://doi.org/10.18653/v1/2020.emnlp-demos.2, https://aclanthology.org/2020.emnlp-demos.2

Ojala T, Pietikainen M, Maenpaa T (2002) Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans Pattern Anal Mach Intell 24(7):971–987. https://doi.org/10.1109/TPAMI.2002.1017623

Pan H, Lin Z, Fu P, et al (2020) Modeling intra and inter-modality incongruity for multi-modal sarcasm detection. In: Findings of the association for computational linguistics: EMNLP 2020. Association for Computational Linguistics, pp 1383–1392. https://doi.org/10.18653/v1/2020.findings-emnlp.124

Schifanella R, de Juan P, Tetreault J, et al (2016) Detecting sarcasm in multimodal social platforms. In: Proceedings of the 24th ACM international conference on multimedia. Association for Computing Machinery, New York, NY, USA, MM '16, pp 1136–1145. https://doi.org/10.1145/2964284.2964321

Srivastava N, Hinton G, Krizhevsky A et al (2014) Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res 15(1):1929–1958

Tan H, Bansal M (2019) LXMERT: learning cross-modality encoder representations from transformers. In: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP). Association for Computational Linguistics, pp 5100–5111. https://doi.org/10.18653/v1/D19-1514

Van Hee C, Lefever E, Hoste V (2018) SemEval-2018 task 3: irony detection in English tweets. In: Proceedings of The 12th international workshop on semantic evaluation. Association for Computational Linguistics, New Orleans, Louisiana, pp 39–50. https://doi.org/10.18653/v1/S18-1005, https://aclanthology.org/S18-1005

Vaswani A, Shazeer N, Parmar N, et al (2017) Attention is all you need. In: Advances in neural information processing systems, vol 30. Curran Associates, Inc., pp 5998–6008

Wang X, Sun X, Yang T, et al (2020) Building a bridge: a method for image-text sarcasm detection without pretraining on image-text data. In: Proceedings of the first international workshop on natural language processing beyond text. Association for Computational Linguistics, pp 19–29. https://doi.org/10.18653/v1/2020.nlpbt-1.3

Xu N, Zeng Z, Mao W (2020) Reasoning with multimodal sarcastic tweets via modeling cross-modality contrast and semantic association. In: Proceedings of the 58th annual meeting of the association for computational linguistics. Association for Computational Linguistics, pp 3777–3786. https://doi.org/10.18653/v1/2020.acl-main.349