What's Real News Today? A Multimodal, Continual-Learning Approach for Detecting Fake News Over Time

Luca Maiano[1,2][0000−0001−7969−7821], Martina Evangelisti[1], Silvia Bianchini[1], and Aris Anagnostopoulos[1][0000−0001−9183−7911]

[1] Department of Computer, Control and Management Engineering, Sapienza University, 00185, Rome, Italy
{evangelisti.1796480,bianchini.1796898}@studenti.uniroma1.it
{maiano,aris}@diag.uniroma1.it
[2] Ubiquitous, 00185, Rome, Italy
http://ubiquitous.green

Abstract. Multimodal fake news detectors are typically trained to work on fixed distributions, making them hardly applicable to ever-changing events. Although it is possible to apply transfer learning to retrain a model on the most recent facts, it will tend to lose its ability to recognize old contents. We mitigate this problem by considering news as a stream of data that becomes available over time and by introducing a continual-learning solution that learns from new events as they become available. Our solution maintains good performance on previously known tasks without limiting the applicability of this solution to older news, leading to a substantial gain of +9.22% accuracy on average compared to transfer learning and a +3.65% increase in F1 score over the ideal scenario where you train the model on all data in one session. Besides this, we introduce the Tri-Encoder, a state-of-the-art multimodal model that allows the cross-attention mechanism between images and texts to be applied.

Keywords: Fake news · Continual learning · Multimodal learning.

1 Introduction

The massive adoption of social networks has made them a very effective tool for spreading false content. Fake news stories often spread faster and with a higher frequency than the real ones [1], but, more importantly, the more a user is exposed to the same content, the more she tends to perceive it as trustworthy [1]. This fact can have a more profound effect than one may expect. An example of this is the 2016 presidential election in the United States. Snopes[3] identified 529 social-media rumors about Donald Trump and Hillary Clinton that could have influenced the election outcome.

[3] https://www.snopes.com – Fact-checking website and reference source for urban legends, folklore, myths, rumors, and misinformation.

There are many challenges to face to counter this phenomenon. First, the most influential fake news contain both texts and images. For example, tweets with images obtain 18% more clicks, 89% more likes, and 150% more retweets than tweets with text-only content [30]. A similar trend takes place on Facebook, where the 87% of the posted photos have been liked, clicked, or shared [30]. Because of this fact, recent studies have analyzed the semantics of multimodal content to classify the news as real or fals with three main approaches: (1) *early-fusion* methods [31, 24] learn low-level features from different modalities that are immediately fused, and fed into a single prediction model, (2) *late-fusion* models [3] fuse unimodal decisions with some mechanisms such as averaging and voting, and (3) *hybrid-fusion* [8] combines the early fusion and late fusion. Using VisualBERT [16], MMBT, and ViLBERT, Dimitrov et al. [8] evaluated several fusion techniques (such as early-fusion, late-fusion and self-supervised models) for propaganda identification. According to their research, self-supervised joint learning models, and in particular VisualBERT, outperform other fusion techniques.

Although this type of approach seems to attain high levels of accuracy in most of the studies, its applicability in real scenarios is still somewhat limited [17, 12, 22]. Indeed, most state-of-the-art approaches apply these techniques in a static setting, where the training and test data belong to a *fixed distribution*, known at design time. This assumption, however, does not reflect the ever-changing nature of news [1] being spread online based on recent events. Some studies proposed to tackle this problem from a different perspective, by analyzing the propagation of news or the communities and the users' reactions to such content [19]. However, these interactions can sometimes be complex to capture because they require monitoring of the entire network, something that is not always feasible. Therefore, analyzing the content stream remains the most accessible way. Motivated by this discussion, in this work, we propose to model the latest news flow as an *incremental task*, where data arrive sequentially in batches, and each batch corresponds to some new events that we want to learn to classify. Our contributions can be summarized as follows. (1) We introduce a multimodal architecture (the *Tri-Encoder*) for fake-news classification based on the analysis of texts and images. (2) Next, we apply a *continual-learning strategy*, which allows to continually learn to classify new topics without losing the ability to classify previously known ones. The proposed solution allows not only to maintain good performance over time, but it even improves compared to the ideal case in which all the topics are immediately available in the first training session. (3) Finally, we perform various measurements and comparison of our approach with others.

The remainder of this paper is organized as follows. In Section 2, we introduce our proposed method. Section 3 present our experiments. We conclude in Section 4.

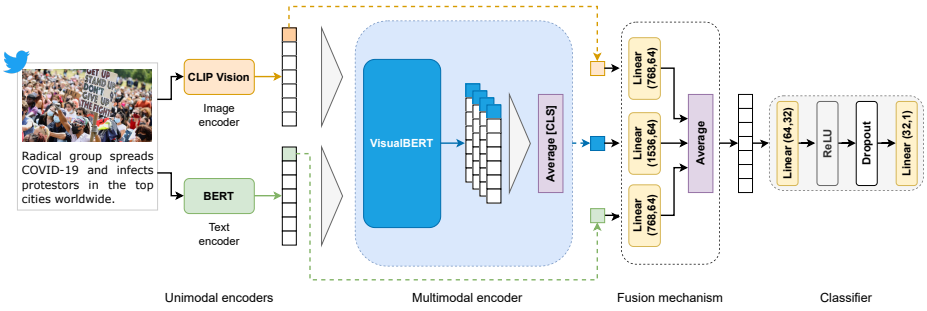


Fig. 1: Overview of the proposed *Tri-Encoder* multimodal architecture.

2 Methodology

Our model aims at learning the discriminable feature representations for fake-news detection in a way that can constantly adapt to the most recent events. We focus on the news spread on Twitter, but the same framework can be extended to other social networks. Formally, given a tweet $X = \{T, V\}$ comprising textual (T) and visual (V) information, our goal is to learn a target function $g(X, \theta) = Y$ that predicts whether the post is a fake ($Y = 0$) or true ($Y = 1$) content by examining the textual and visual information, as well as the semantic relationship between the two types of information. To this aim, we model news as a stream of unknown distributions $\mathcal{D} = \{\mathcal{D}^1, \ldots, \mathcal{D}^n\}$ over $X \times Y$, with X and Y input and output random variables, respectively. At time step i, the model learns a new function $f_i^{CL} = g(X, \theta^i)$ by updating its current parameters θ^{i-1} on a new fact $\mathcal{D}^i$ by training it on a training set $\mathcal{D}^i_{\text{train}}$ and testing it on a test set $\mathcal{D}^i_{\text{test}}$. The objective of the continual-learning algorithm is to minimize the loss $\mathcal{L}_D$ over the entire stream of data $\mathcal{D}$:

$$\mathcal{L}_D\left(f_n^{CL}, n\right) = \frac{1}{\sum_{i=1}^{n} |\mathcal{D}^i_{\text{test}}|} \sum_{i=1}^{n} \mathcal{L}_{\text{fact}}\left(f_n^{CL}, \mathcal{D}^i_{\text{test}}\right) \tag{1}$$

$$\mathcal{L}_{\text{fact}}\left(f_n^{CL}, \mathcal{D}^i_{\text{test}}\right) = \sum_{j=1}^{|\mathcal{D}^i_{\text{test}}|} \mathcal{L}_{\text{cls}}\left(f_n^{CL}(x_j^i), y_j^i\right) \tag{2}$$

where the loss $\mathcal{L}_{\text{cls}}\left(f_n^{CL}(x_j^i), y_j^i\right)$ represents the binary cross entropy loss.

2.1 The *Tri-Encoder* Model Architecture

The Tri-Encoder model architecture is shown in Figure 1. The model involves an *image encoder* and a *text encoder* to obtain unimodal image and text representations, and a *multimodal encoder* to fuse and align the image and text representations for multimodal reasoning.

Text encoder. Given the text of a tweet, we first tokenize and embed it in a list of word vectors using WordPiece [27] with a vocabulary of 30,000 tokens and append two special characters to the input: the class token `[CLS]`, which is appended in front of each input example, and the separator token `[SEP]`. Then, we apply a transformer model over the word vectors to encode them into a list of N_T hidden state vectors $h_T \in \mathbb{R}^H$, including $h_{\text{CLS},T}$ for the text classification token. In all our experiments, we use the bidirectional BERT-base [7] model with 12 layers and 12 attention heads, which produces 768-dimensional hidden vectors. In the training phase, all weights are frozen except for the last two layers.

Image encoder. For the image encoder, we use the pretrained CLIP's [20] visual feature extractor. Given an input image, we split it into 32×32 patches, which are then linearly embedded and fed into a ViT-B/32 [9] transformer model along with positional embeddings and an extra image classification token `[CLS]`. Similarly to the text encoder, the image-encoder output is a list of N_V image hidden state vectors $h_V \in \mathbb{R}^H$ ($H = 768$), each corresponding to an image patch, plus an additional $h_{\text{CLS},V}$ for the image classification token. Similarly to the text encoder, all weights are frozen except for the last two layers during training.

Multimodal encoder. We use an additional transformer model for learning a joint contextualized representation of the image and text hidden states. Specifically, we apply the VisualBERT [16] model that is pretrained on the Visual Commonsense Reasoning dataset [29]. The model consists of a stack of transformer layers that align the regions of the input image with the textual input through self-attention. Compared to a simple concatenation of the two unimodal embeddings, this configuration allows *cross-attention* between the projected unimodal image and text representations and fuses the two modalities. This encoder takes as input the visual (h_V) and textual (h_T) hidden representations extracted from the unimodal models and produces $N_T + N_V + 2$ multimodal hidden state vectors $h_M \in \mathbb{R}^H$ ($H = 768$), where N_T and N_V are the numbers of text tokens and image patches, respectively, and the two additional vectors are the special `[CLS]` and `[SEP]` tokens. The output of the last layer may not always be the best representation of the input when fine tuning for downstream tasks. Previous studies proved that for pretrained language models, the most transferable contextualized representations of input text tend to occur in the middle layers, whereas the top layers specialize in language modeling [5]. Therefore, inspired by the same considerations, we average the penultimate last three layers' output and concatenate the averaged hidden state vector with the `[CLS]` hidden state vector of the output layer, producing a 1536-dimensional output h_{MM}. We validate this choice in Section 3.1.

Fusion mechanism. In the fusion step, the visual, textual, and the multimodal `[CLS]` feature vectors $h_{T,\texttt{CLS}}$, $h_{V,\texttt{CLS}}$, and $h_{MM,\texttt{CLS}}$ are all projected onto a 64-dimensional subspace through a linear layer, producing the corresponding $h'_{T,\texttt{CLS}}$, $h'_{V,\texttt{CLS}}$, and $h'_{M,\texttt{CLS}}$ vectors. Finally, we calculate a weighted average of these vectors

$$h_{TVM} = \text{avg}(w_T h'_{T,\texttt{CLS}} + w_V h'_{V,\texttt{CLS}} + w_M h'_{M,\texttt{CLS}}) \tag{3}$$

where w_T and w_V are fixed to 0.25, and $w_M = 0.5$.

Classifier. The final step of the Tri-Encoder is the classification step. The classifier is composed of two linear layers generating a 32-dimensional and a 1-dimensional outputs, and are separated by the rectified linear unit (ReLU) and dropout operations. A sigmoid activation function follows the output of the last layer:

$$\sigma(x) = \frac{1}{1 + exp(-x)}$$

Values below a threshold τ are predicted *false*. Experimentally, we found $\tau = 0.46$ as the optimal value.

2.2 Continual-Learning Strategy

Now, we propose a continuous learning strategy that allows the Tri-Encoder to update its knowledge on the latest news as they become available. To this end, it is essential to avoid catastrophic forgetting [6], as we want the model to continue to classify previous content accurately. A naive idea might be to retrain the model on an ever-growing set of training data; however, such an approach can become prohibitively expensive as the volume of data grows over time. On the other hand, the model should not overfit to a new event because it would cause it to loose its previous skills.

We choose to adopt a *knowledge distillation* approach. We choose this simple regularization method since it allows maintaining the model size fixed as the data size increases and it does not require to store the previous data in *memory*. Distillation techniques were introduced by Hinton et al. [11] as a means to transfer knowledge from a neural network T (the *teacher*) to a neural network S (the *student*). The key idea behind knowledge distillation is that soft probabilities predicted by a network of trained "teachers" contain much more information about a data point than a simple class label. For example, if multiple classes are assigned high probabilities for an image, this could mean that the image must be close to a decision boundary between those classes. Forcing a student to mimic these probabilities should then cause the student network to absorb some of this knowledge that the teacher discovered, above and beyond the information in training labels alone. To implement this strategy, we modify the classification loss $\mathcal{L}_{\text{cls}}$ in Equation 2 by adding a regularization factor

$$\mathcal{L}'_{\text{cls}} = \alpha\mathcal{L}_{kd} + \beta\mathcal{L}_{\text{cls}}\big(f_n^{CL}(x_j^i), y_j^i\big), \tag{4}$$

where α and β are experimentally set to 0.5 and 0.6, respectively, and $\mathcal{L}_{kd}$ is the mean squared error (MSE) loss that measures the squared L2 norm between the teacher and the student outputs.

3 Experiments

In this section, we validate the model and the proposed continual-learning solution. All models were trained with the Adam optimizer, a learning rate fixed to

$3e^{-}5$ and a batch size of 32 samples. For the experiments reported in Sections 3.1, and 3.2, we train the models for 10 epochs, while for the other experiments we train for 5 epochs only. For evaluating our experiments, we chose three commonly used datasets: (1) *MediaEval Verifying Multimedia Use benchmark* [2], (2) *PolitiFact*, and (3) *GossipCop* [21].

3.1 Ablation Study

To evaluate the design choices of our model, we now analyze several possible multimodal variants. We consider three baseline models, which are combined with the following three feature extractors for the images: ResNet50 *(R)*, CLIP Vision *(C)*, and ViT *(V)*.

Model	All news		Fake News			Real News		
	Acc/F1	**F1-macro**	**Prec.**	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**
SE(R)	0.6347	0.6338	0.7180	0.5984	0.6528	0.5584	0.6837	0.6147
SE(C)	0.7586	0.7520	0.7820	0.8031	0.7924	0.7250	0.6987	0.7119
DE(R)	0.7094	0.6711	0.6844	**0.9158**	0.7834	**0.7921**	0.4316	0.5587
DE(C)	0.7058	0.7024	0.7623	0.7079	0.7341	0.6413	0.7029	0.6707
VB(R)-b	0.7613	0.7564	0.7948	0.7873	0.7910	0.7172	0.7264	0.7218
VB(R)-cat	0.7513	0.7460	0.7846	0.7809	0.7828	0.7070	0.7115	0.7092
VB(R)-avg	0.7367	0.7313	0.7728	0.7666	0.7697	0.6892	0.6965	0.6928
VB(C)-b	0.7522	0.7382	0.7479	0.8571	0.7988	0.7606	0.6111	0.6777
VB(C)-cat	0.7358	0.7337	0.8003	0.7190	0.7575	0.6672	0.7585	0.71
VB(C)-avg	**0.7978**	**0.7934**	**0.8248**	0.8222	**0.8235**	0.7617	0.7649	**0.7633**
VB(V)-b	0.6766	0.6766	0.7956	0.5873	0.6757	0.5892	**0.7970**	0.6775
VB(V)-cat	0.6493	0.6485	0.7351	0.6079	0.6655	0.5719	0.7051	0.6315
VB(V)-avg	0.7167	0.7115	0.7593	0.7412	0.7502	0.6625	0.6837	0.6729

Table 1: Multimodal methods performances on the MediaEval [2] dataset.

Simple-Encoder (SE). This model is based on the simple concatenation of the features extracted from images and texts. For text, we use BERT, taking the `[CLS]` representation for the last hidden state. The unimodal models are fed into a linear layer with an output size of 512 and concatenated, producing a 1024-dimensional vector that is passed through two linear layers of 1024×32 and 32×1 dimensions separated by a ReLU function, a dropout layer (set to 0.4), and a sigmoid activation function.

Dual-Encoder (DE). This architecture has been inspired by the double visual textual transformer model (DVTT) [18]. Each modality is conditioned by the other, enriching the text with visual information from the text encoder and vice-versa. The textual representation is taken from the last hidden state of the BERT model. Similarly, when CLIP Vision is employed as a visual feature extractor, the image representation comes from the last hidden state of the transformer encoder. In the case of ResNet50, feature maps are extracted from the second last layer, and a $(6, 6)$ pooling is applied. Finally, we concatenate the `[CLS]` tokens from both transformers, obtaining a 1024-dimensional embedding.

VisualBERT (VB). This model combines image regions and language with a transformer, allowing self-attention to discover implicit alignments between language and vision. It is pretrained on visual-reasoning tasks. We consider the following three variants with all the backbones:

- *base (b):* the representation of the `[CLS]` token representation from the last hidden state is fed into the classifier;
- *concatenation (cat):* the `[CLS]` token representations from the last four layers are concatenated before classification;
- *average (avg):* the `[CLS]` token representations from the penultimate three layers are averaged and concatenated with the last `[CLS]` token before classification.

Table 1 summarizes all the experiments. The results show a consistent advantage of the *VB(C)-avg* configuration over the others, which is the same configuration used for our Tri-Encoder. Besides that, we can observe that using CLIP Vision for the visual component achieves superior performance in all the configurations. As the original model is trained in a multimodal setting, we make the hypothesis that this is because of the fact that it manages to extract features more aligned with the textual component. Regarding the compared architectures, VisualBERT achieves, on average, superior performance compared to the Simple-Encoder and Dual-Encoder.

3.2 Fake-News Detection Performance

To validate the proposed Tri-Encoder, we propose a comparison with state-of-the-art methods. In particular, we validate the model's performance against (1) well-known unimodal deep-learning architectures and (2) multimodal solutions designed for fake-news detection. All the models have been trained on MediaEval for 10 epochs with a batch size of 32, a learning rate of 3e-05, and the Adam [14] optimizer.

Table 2 reports the results of this first experiment. We can notice that our Tri-Encoder architecture outperforms all the other methods, followed by CALM [28], which achieves comparable performance. In the next section we study the architectural choices that led to the proposed Tri-Encoder. We can generally observe that multimodal models perform better than unimodal ones, confirming the additive contribution of the images to an accurate classification. We can also notice that for the unimodal architectures, models that analyze images outperform BERT, a model based on text. A possible explanation for this could be that in the MediaEval dataset, many fake images have been manipulated in a way that makes the detection of such manipulation highly accurate by the image classifiers.

3.3 Robustness to Incremental Topics

Whereas the previous results demonstrate the effectiveness of the proposed method on a task, in this section, we evaluate the model's performance on new

Model	F1-micro	F1-macro
BERT [7]	0.6247	0.6238
ResNet50 [10]	0.7021	0.6962
VGG19 [23]	0.6275	0.6273
CLIP Vision [20]	0.7440	0.7353
VisualBERT [16]	0.7978	0.7934
MVAE [13]	0.745	0.744
EANN [25]	0.715	0.719
EANN- [25]	0.648	0.6385
SpotFake [24]	0.778	0.760
MFN [3]	0.808	0.785
MCAN [26]	0.809	0.808
CALM [28]	0.845	0.839
CAFE [4]	0.806	0.805
Simple-Encoder (ours)	0.7586	0.7520
Dual-Encoder (ours)	0.7058	0.7024
Tri-Encoder (ours)	**0.851**	**0.845**

Table 2: Fake news detection performance on the MediaEval [2] dataset.

tasks in a continual-learning scenario. Specifically, we evaluate the performance of knowledge distillation (KD) compared to two other strategies: the transfer learning (TL) and the elastic weight consolidation (EWC) [15], which can be seen as an improvement of the L2-regularization.

Static training sessions. Before illustrating the results on the continual-learning setting, let's evaluate the performance of the models when the Tri-Encoder is trained from scratch on *all datasets simultaneously*. This allows us to evaluate the performance of the continual learner in the ideal scenario where the data are immediately available in the first training session. In Table 4, we report the results in terms of F1 score when the training set is balanced or unbalanced between the three datasets. As expected, when the data are unbalanced, the F1 score on the MediaEval dataset is higher than the others, having three times the number of samples that the other slices have. By balancing the data, the overall accuracy doesn't change much, but the distribution between the different portions is more even. We can also notice that the model's performance on MediaEval drops slightly compared to the case in which the model is trained only on this dataset (see Table 2). This could be justified by the fact that when the model is trained on all datasets simultaneously, the broader distribution of facts present in all datasets leads the model to converge into a region where it minimizes errors on all topics, but which leads to a slight performance drop on the MediaEval topics.

Furthermore, we evaluate the model's performance in a scenario where we apply *transfer learning*. Starting from MediaEval as the first task ($T1$), in Table 3a we see the performance after applying transfer learning to $T2$ = PolitiFact

Task	MediaEval	PolitiFact	GossipCop
$T1$	0.8515	-	-
$T2$	0.7312	0.7932	-
$T3$	0.7218	0.5224	0.7117

a $T1$ = MediaEval, $T2$ = PolitiFact, $T3$ = GossipCop

Task	MediaEval	GossipCop	PolitiFact
$T1$	0.8515	-	-
$T2$	0.6860	0.7259	-
$T3$	0.6466	0.5123	0.7351

b $T1$ = MediaEval, $T2$ = GossipCop, $T3$ = PolitiFact

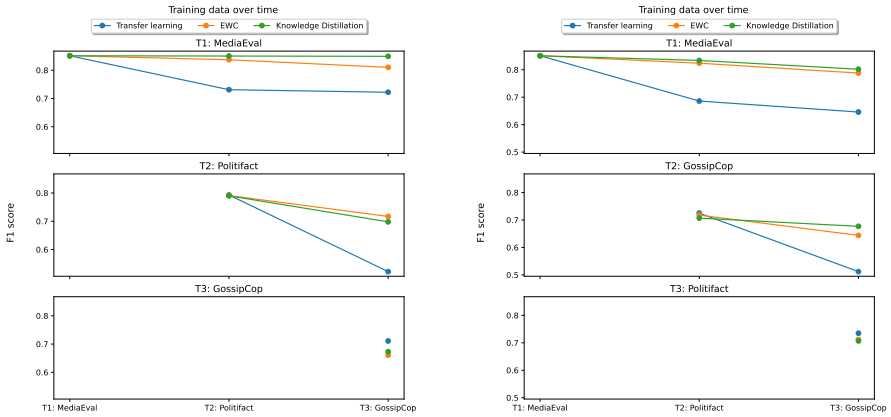
Table 3: Transfer learning performance in terms of F1 score of the model trained on $T1$, followed by TL to $T2$, followed by TL to $T3$. The rows indicate the dataset on which we perform the TL (or the initial training for row $T1$), and the columns indicate the dataset on which we perform evaluation.

Tested dataset	Not Balanced		Balanced	
	F1-micro	F1-macro	F1-micro	F1-macro
MediaEval	0.7810	0.7762	0.7105	0.6855
PolitiFact	0.6251	0.6232	0.6529	0.6519
GossipCop	0.6781	0.6776	0.6855	0.6848
All	0.6984	0.6978	0.6966	0.6895

Table 4: Results of the model trained from scratch on all three datasets available at once. *Balanced* indicates that in the training dataset, we balance a number of samples for each fact.

and to $T3$ = GossipCop, and in Table 3a we see the performance after applying transfer learning to $T2$ = GossipCop and to $T3$ = PolitiFact. We can see that in both cases, the model suffers from catastrophic forgetting. Indeed, as we train it on new datasets, it becomes less accurate on previously seen ones. In the following section we see how our incremental-learning strategy manages to reduce this problem.

Continual-training sessions. We now evaluate the robustness of the model through several continual-learning sessions. To do this, in the first training session $T1$, we train the model on MediaEval. We chose this dataset for the first session as it is the largest among those considered and it allows a first training phase of the Tri-Encoder without causing overfitting. In subsequent training sessions, we expose the model to the new facts in GossipCop and PolitiFact. To this end, we introduce two more training sessions, namely $T2$ and $T3$. The goal of the



(a) $T_1 = \text{MediaEval}, T_2 = \text{PolitiFact}, T_3 = \text{GossipCop}$

(b) $T_1 = \text{MediaEval}, T_2 = \text{GossipCop}, T_3 = \text{PolitiFact}$

Fig. 2: F1 score of the Tri-Encoder over all tasks during time. In each of figures (a) and (b), the first plot shows the F1 score evaluated on dataset $T1$ (MediaEval for both of them), the second one the F1 score for $T2$ (PolitiFact for (a), GossipCop for (b)), and the third one for $T3$ (GossipCop for (a), PolitiFact for (b)).

continual learner is to learn new tasks without encountering catastrophic forgetting of the previous ones. To validate this approach, we compare the performance of knowledge distillation (KD) with respect to EWC and transfer learning.

In Figure 2, we show the performance of all the strategies in terms of F1 score. In Figure 2(a) we train first on MediaEval ($T1$), and then on PolitiFact ($T2$) and GossipCop ($T3$). In Figure 2(b) we switch GossipCop ($T2$) and PolitiFact ($T3$). From both figures, we can see that the performance of the knowledge-distillation approach remains more or less constant on $T1$ during all the training sessions. For concreteness let us look at Figure 2(a). EWC's performance on this task is more or less comparable, although it suffers a more pronounced drop in F1 score when we switch from PolitiFact ($T2$) to GossipCop ($T3$). In the case of transfer learning, we notice a substantial drop in performance with the arrival of new tasks. This is absolutely justifiable because, in transfer learning, we do not impose to the model to perform the well also in the previous tasks. In the second plot, where we evaluate with respect to the dataset $T2 = \text{PolitiFact}$, we can observe a similar behavior. Knowledge distillation and EWC have more or less similar performance, with a small drop (about 10%) in F1 score from $T2$ to $T3$, and with a high drop in the case of transfer learning. Finally, in the last training session, transfer learning outperforms the other strategies, whereas knowledge distillation and EWC again have comparable performance. The results generally suggest greater robustness of continual learning methods compared to transfer learning. Although knowledge distillation and EWC obtain comparable perfor-

mance in all training sessions, the former is more robust on the oldest task ($T1$), guaranteeing superior stability on all learning sessions.

Table 5 shows the average accuracy and forgetting of all methods on the three tasks after the last training session. In all settings, transfer learning attains the worst results regarding average accuracy and forgetting. As for EWC and knowledge distillation, these achieve comparable accuracy values with a slight advantage of knowledge distillation. In terms of forgetting, however, knowledge distillation achieves the best performance, confirming the considerations made in the previous section. The only case that the continual-learning approaches give an inferior score compared to transfer learning is in the evaluation of the third training session ($T3$), but even there the difference is small (see the two bottom plots in Figure 2).

Method	$\boldsymbol{ACC}$	$\boldsymbol{BWT}$
Transfer learning	0.6520	0.2002
EWC	0.7294	0.0576
Knowledge distillation	**0.7401**	**0.0475**

a $T1$: MediaEval, $T2$: PolitiFact, $T3$: GossipCop.

Method	$\boldsymbol{ACC}$	$\boldsymbol{BWT}$
Transfer learning	0.6314	0.2092
EWC	0.7151	0.0681
Knowledge distillation	**0.7277**	**0.0399**

b $T1$: MediaEval, $T2$: GossipCop, $T3$: PolitiFact.

Table 5: Average accuracy (ACC) and forgetting (BWT) of the continual-leaning approaches on the three datasets. For ACC a higher value is better, for BWT a lower value is better.

For a more detailed report of the performance of the three strategies after the last training session, we report the results in terms of F1 score in Table 6. For each task, we report the best results in bold. We also mark with $*$ the strategy that achieves the best performance on a given task. As also mentioned in the discussion of Figure 2, transfer learning achieves the best performance only in the last task ($T3$). However, knowledge distillation is shown to be the most robust method on the first task ($T1$), followed by EWC. Compared to Table 4, we can see that although the performance degrades slightly on all tasks compared to standard training, continuous-learning strategies still achieve acceptable performance on all three tasks. Moreover, comparing the results with those of Table 4, we can even notice that with continual-learning strategies, we achieve a higher performance compared to training all three datasets in a single session.

It is interesting to note a detail that emerges both from the experiments presented in Table 5, as well as from those of Tables 6 and 3. We can observe a small difference in terms of performance in the order in which we train the model on the various tasks, which seems to suggest that it may have an effect at the model's capacity to generalize. Training on PolitiFact and then on GossipCop seems to improve performance in all experiments. This could be because the topics in GossipCop are very different from those in the other two datasets.

Training	Task	All news		Fake News			Real News		
		Acc/F1	F1-macro	Prec.	Rec.	F1	Prec.	Rec.	F1
TL	$T1$	0.722	0.721	0.839	0.656	0.736	0.621	**0.818**	0.706
	$T2$	0.522	0.521	0.522	0.478	0.499	0.522	0.566	0.543
	$T3^*$	**0.712**	**0.711**	**0.703**	**0.786**	**0.695**	**0.719**	**0.735**	**0.727**
EWC	$T1$	0.810	0.796	0.801	0.903	0.849	0.828	0.675	0.744
	$T2^*$	**0.717**	**0.717**	0.729	**0.687**	**0.707**	**0.706**	0.747	**0.744**
	$T3$	0.661	0.658	0.661	0.595	0.627	0.661	0.721	0.689
KD	$T1^*$	**0.849**	**0.841**	**0.845**	**0.913**	**0.878**	**0.857**	0.758	**0.804**
	$T2$	0.698	0.693	**0.758**	0.578	0.656	0.661	**0.817**	0.731
	$T3$	0.673	0.673	0.643	0.709	0.674	0.706	0.639	0.671

a $T1$: MediaEval, $T2$: PolitiFact, $T3$: GossipCop.

Training	Task	All news		Fake News			Real News		
		Acc/F1	F1-macro	Prec.	Rec.	F1	Prec.	Rec.	F1
TL	$T1$	0.647	0.584	0.650	0.873	0.745	0.633	0.318	0.423
	$T2$	0.512	0.428	0.495	**0.937**	0.647	0.683	0.123	0.208
	$T3^*$	**0.735**	**0.734**	0.702	**0.813**	**0.754**	**0.78**	0.658	0.714
EWC	$T1$	0.788	0.771	0.777	**0.902**	**0.835**	**0.814**	0.624	0.707
	$T2$	0.644	0.635	0.589	0.845	**0.694**	**0.765**	0.461	0.575
	$T3$	0.712	0.708	**0.773**	0.597	0.674	0.674	**0.826**	**0.742**
KD	$T1*$	**0.802**	**0.791**	**0.810**	0.868	0.838	0.787	**0.705**	**0.744**
	$T2*$	**0.677**	**0.677**	**0.636**	0.758	0.692	0.731	**0.604**	**0.661**
	$T3$	0.707	0.704	0.757	0.607	0.674	0.674	0.807	0.735

b $T1$: MediaEval, $T2$: GossipCop, $T3$: PolitiFact.

Table 6: F1 score of the Tri-Encoder over all tasks after the last training session. Bold values indicate the best performance on a task. Tasks marked with $*$ indicate the learning strategy that performed best on those specific tasks.

Consequently, introducing this dataset in the second training session could have a negative effect on the third session. We leave the exploration of this phenomenon as future work.

4 Conclusion

In this work, we have introduced a content-based multimodal continual-learning strategy, which allows to learn from event streams as they become available over time. This strategy offers the advantage of being able to model the problem of false-content detection in a more realistic way than what is traditionally done by the state-of-the-art approaches, in which one works on fixed distributions. In addition to being more realistic, this paradigm shift still allows for satisfactory performance and leads to a more than 9% improvement of the average accuracy. We have shown that not only does the performance of the continual learner remains high as new tasks arrive, but it can even improve compared to training on a dataset obtained by concatenating different datasets.

The finding in this work creates a lot of interesting future directions. First of all, regarding the number of topics, it is necessary to understand how much ev-

ery single topic can influence the performance of the learning strategy. Thus an extension of this study to other datasets and measurement of one topic at a time may provide useful insights. Furthermore, a higher number of training sessions may reveal more interesting patterns and may raise the question of whether there are some moments in time where complete retraining may be beneficial. Besides this, it would be interesting to test other incremental-learning techniques. Finally, it would be important to evaluate the possibility of integrating human feedback within the continuous-learning component, for example, by assessing the possibility of applying reinforcement learning.

Acknowledgements

Supported by the ERC Advanced Grant 788893 AMDROMA, the EC H2020RIA project "SoBigData++" (871042), the PNRR MUR project PE0000013-FAIR," the PNRR MUR project IR0000013-SoBigData.it, and the MUR PRIN project 2022EKNE5K "Learning in Markets and Society."

References

1. Alam, F., Cresci, S., Chakraborty, T., Silvestri, F., Dimitrov, D., Martino, G.D.S., Shaar, S., Firooz, H., Nakov, P.: A survey on multimodal disinformation detection. In: Proceedings of the 29th International Conference on Computational Linguistics. pp. 6625–6643. International Committee on Computational Linguistics, Gyeongju, Republic of Korea (Oct 2022), https://aclanthology.org/2022.coling-1.576
2. Boididou, C., Andreadou, K., Papadopoulos, S., Dang Nguyen, D.T., Boato, G., Riegler, M., Larson, M., Kompatsiaris, I.: Verifying multimedia use at mediaeval 2015 in mediaeval benchmarking initiative for multimedia evaluation (09 2015)
3. CHEN, J., WU, Z., YANG, Z., XIE, H., WANG, F., LIU, W.: Multimodal fusion network with latent topic memory for rumor detection. In: 2021 IEEE International Conference on Multimedia and Expo, ICME 2021. pp. 1–6. Proceedings - IEEE International Conference on Multimedia and Expo, IEEE Computer Society, United States (Jun 2021). https://doi.org/10.1109/ICME51207.2021.9428404
4. Chen, Y., Li, D., Zhang, P., Sui, J., Lv, Q., Tun, L., Shang, L.: Cross-modal ambiguity learning for multimodal fake news detection. In: Proceedings of the ACM web conference 2022. pp. 2897–2905 (2022)
5. Dai, Z., Lai, G., Yang, Y., Le, Q.: Funnel-transformer: Filtering out sequential redundancy for efficient language processing. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H. (eds.) Advances in Neural Information Processing Systems. NIPS'20, vol. 33, pp. 4271–4282. Curran Associates, Inc., Red Hook, NY, USA (2020), https://proceedings.neurips.cc/paper/2020/file/2cd2915e69546904e4e5d4a2ac9e1652-Paper.pdf
6. De Lange, M., Aljundi, R., Masana, M., Parisot, S., Jia, X., Leonardis, A., Slabaugh, G., Tuytelaars, T.: A continual learning survey: Defying forgetting in classification tasks. IEEE Transactions on Pattern Analysis and Machine Intelligence **44**(7), 3366–3385 (2022). https://doi.org/10.1109/TPAMI.2021.3057446

7. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding (2018). https://doi.org/10.48550/ARXIV.1810.04805, https://arxiv.org/abs/1810.04805
8. Dimitrov, D., Bin Ali, B., Shaar, S., Alam, F., Silvestri, F., Firooz, H., Nakov, P., Da San Martino, G.: SemEval-2021 task 6: Detection of persuasion techniques in texts and images. In: Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021). pp. 70–98. Association for Computational Linguistics, Online (Aug 2021). https://doi.org/10.18653/v1/2021.semeval-1.7, https://aclanthology.org/2021.semeval-1.7
9. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale (2020). https://doi.org/10.48550/ARXIV.2010.11929, https://arxiv.org/abs/2010.11929
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 770–778 (2016). https://doi.org/10.1109/CVPR.2016.90
11. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network (2015). https://doi.org/10.48550/ARXIV.1503.02531, https://arxiv.org/abs/1503.02531
12. Horne, B.D., Nørregaard, J., Adali, S.: Robust fake news detection over time and attack. ACM Trans. Intell. Syst. Technol. **11**(1) (dec 2019). https://doi.org/10.1145/3363818, https://doi.org/10.1145/3363818
13. Khattar, D., Goud, J.S., Gupta, M., Varma, V.: Mvae: Multimodal variational autoencoder for fake news detection. In: The World Wide Web Conference. p. 2915–2921. WWW '19, Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3308558.3313552, https://doi.org/10.1145/3308558.3313552
14. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization (2014). https://doi.org/10.48550/ARXIV.1412.6980, https://arxiv.org/abs/1412.6980
15. Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A.A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., Hassabis, D., Clopath, C., Kumaran, D., Hadsell, R.: Overcoming catastrophic forgetting in neural networks. Proceedings of the National Academy of Sciences **114**(13), 3521–3526 (03 2017). https://doi.org/10.1073/pnas.1611835114, https://doi.org/10.1073%2Fpnas.1611835114
16. Li, L.H., Yatskar, M., Yin, D., Hsieh, C.J., Chang, K.W.: Visualbert: A simple and performant baseline for vision and language (2019). https://doi.org/10.48550/ARXIV.1908.03557, https://arxiv.org/abs/1908.03557
17. M. Silva, R., R. Pires, P., Almeida, T.A.: Incremental learning for fake news detection. Journal of Information and Data Management **13**(6) (Jan 2023). https://doi.org/10.5753/jidm.2022.2542, https://sol.sbc.org.br/journals/index.php/jidm/article/view/2542
18. Messina, N., Falchi, F., Gennaro, C., Amato, G.: AIMH at SemEval-2021 task 6: Multimodal classification using an ensemble of transformer models. In: Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021). pp. 1020–1026. Association for Computational Linguistics, Online (Aug 2021). https://doi.org/10.18653/v1/2021.semeval-1.140, https://aclanthology.org/2021.semeval-1.140
19. Monti, F., Frasca, F., Eynard, D., Mannion, D., Bronstein, M.M.: Fake news detection on social media using geometric deep learning (2019). https://doi.org/10.48550/ARXIV.1902.06673, https://arxiv.org/abs/1902.06673

20. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision (2021). https://doi.org/10.48550/ARXIV.2103.00020, https://arxiv.org/abs/2103.00020
21. Shu, K., Mahudeswaran, D., Wang, S., Lee, D., Liu, H.: Fakenewsnet: A data repository with news content, social context and dynamic information for studying fake news on social media. arXiv preprint arXiv:1809.01286 (2018)
22. Siciliano, F., Maiano, L., Papa, L., Baccini, F., Amerini, I., Silvestri, F.: Adversarial data poisoning for fake news detection: How to make a model misclassify a target news without modifying it (2024)
23. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition (2014). https://doi.org/10.48550/ARXIV.1409.1556, https://arxiv.org/abs/1409.1556
24. Singhal, S., Shah, R.R., Chakraborty, T., Kumaraguru, P., Satoh, S.: Spotfake: A multi-modal framework for fake news detection. In: 2019 IEEE Fifth International Conference on Multimedia Big Data (BigMM). pp. 39–47 (2019). https://doi.org/10.1109/BigMM.2019.00-44
25. Wang, Y., Ma, F., Jin, Z., Yuan, Y., Xun, G., Jha, K., Su, L., Gao, J.: Eann: Event adversarial neural networks for multi-modal fake news detection. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. p. 849–857. KDD '18, Association for Computing Machinery, New York, NY, USA (2018). https://doi.org/10.1145/3219819.3219903, https://doi.org/10.1145/3219819.3219903
26. Wu, Y., Zhan, P., Zhang, Y., Wang, L., Xu, Z.: Multimodal fusion with co-attention networks for fake news detection. In: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021. pp. 2560–2569. Association for Computational Linguistics, Online (Aug 2021). https://doi.org/10.18653/v1/2021.findings-acl.226, https://aclanthology.org/2021.findings-acl.226
27. Wu, Y., Schuster, M., Chen, Z., Le, Q.V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., Kaiser, L., Gouws, S., Kato, Y., Kudo, T., Kazawa, H., Stevens, K., Kurian, G., Patil, N., Wang, W., Young, C., Smith, J., Riesa, J., Rudnick, A., Vinyals, O., Corrado, G., Hughes, M., Dean, J.: Google's neural machine translation system: Bridging the gap between human and machine translation (2016). https://doi.org/10.48550/ARXIV.1609.08144, https://arxiv.org/abs/1609.08144
28. Wu, Z., Chen, J., Yang, Z., Xie, H., Wang, F.L., Liu, W.: Cross-modal attention network with orthogonal latent memory for rumor detection. In: Zhang, W., Zou, L., Maamar, Z., Chen, L. (eds.) Web Information Systems Engineering – WISE 2021. pp. 527–541. Springer International Publishing, Cham (2021)
29. Zellers, R., Bisk, Y., Farhadi, A., Choi, Y.: From recognition to cognition: Visual commonsense reasoning. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)
30. Zhang, D., Shang, L., Geng, B., Lai, S., Li, K., Zhu, H., Amin, T., Wang, D.: Fauxbuster: A content-free fauxtography detector using social media comments. In: Proceedings of IEEE BigData 2018 (2018)
31. Zhou, X., Wu, J., Zafarani, R.: Safe: Similarity-aware multi-modal fake news detection (2020). https://doi.org/10.48550/ARXIV.2003.04981, https://arxiv.org/abs/2003.04981