Gated Recursive and Sequential Deep Hierarchical Encoding for Detecting Incongruent News Articles

Anonymous ACL submission

Abstract

With the increase in misinformation across digital platforms, incongruent news detection is becoming an important research problem. Earlier, researchers have exploited various feature engineering approaches and deep learning models with embedding to capture incongruity between news headlines and the body. Recent studies have also shown the advantages of capturing structural properties of the body using hierarchical encoding. Hierarchical encoding decomposes the body of a news article into smaller segments such as sentences or paragraphs. However, the existing hierarchical methods have not considered two important aspects; (i) deeper hierarchical level, and (ii) importance of different paragraphs in generating document encoding. Motivated by this, in this paper, we propose a *Gated Recursive And Sequential Deep Hierarchical Encoding (GRASHE)* method for detecting incongruent news articles by extends hierarchical encoding upto word level and incorporating incongruently weight of each paragraph. Experimental results show that the proposed models outperform the bag-of-word features, sequential and hierarchical encoding-based counterparts. We also perform various ablation analysis to support the proposed models.

1 Introduction

Detecting incongruity between news headline and its body has evolved as an important research problem in recent times to handle early detection of misinformation in electronic media (Ecker et al., 2014)(Chesney et al., 2017). A news article is considered to be incongruent if its headline does not represent its body due to fabricated, manipulated, false connection[1] or wrong context[2]. People mostly read news headlines only, instead of the full story (Gabielkov et al., 2016). The impressions created by news headline to readers are persistent and significantly contribute to becoming a news story viral in social media platform (Dos Rieis et al., 2015). As a result, detecting incongruent news headlines is becoming an important task to fight misinformation in electronic media.

Though initial study on incongruity detection in news article can be credited to `Fake News Challenge` *(FNC-1)* (Pomerleau and Rao, 2017), the importance of the problem can be traced back to the year 2007 (Andrew, 2007). In the recent times, researchers have exploited various methods for detecting incongruent news articles such as simple n-gram features-based models (Riedel et al., 2017) (Hanselowski et al., 2017), summarization-based models (Mishra et al., 2020) (Sepúlveda-Torres et al., 2021), and hierarchical encoding-based models (Yoon et al., 2021) (Yoon et al., 2019). While incongruent news articles with distinctive features between its headline and body are easy to identify, detecting a systematically created incongruent news article is a non-trivial task.

From the FNC-1 challenge (Pomerleau and Rao, 2017), it is observed that incorporating features extracted from diverged perspective helps in detecting incongruent news articles better. A classic example is *XGBoost*, the winner of the challenge, which considers various types of features such as n-grams, latent features, sentiment, etc., is still one of the top-performing systems even today. However, as observed in (Hanselowski et al., 2018), the models with bag-of-words features often fail to capture information like complex negations, deep semantic relationships, and propositional contents which are important for incongruity detection. To capture deeper contextual and sequential semantic relationship between texts, studies in (Hanselowski et al., 2017) (Conforti et al., 2018) (Borges et al., 2019) combine embeddings obtained from sequential

[1] When the caption of the image does not align with its image or headline does not support its content.

[2] Legitimate information is presented in the wrong context.

models (like LSTM) and the explicit features like n-grams. While the above studies define an article as a sequence of texts, studies in (Yoon et al., 2019) (Yoon et al., 2021) have defined an article as a hierarchical structure, i.e., *body as collection of paragraphs, and paragraph as collection of sentences*. Though the above hierarchical encoding methods provide promising results as compared to their sequential and bag-of-word counterparts (also observed in this paper), these methods have not considered two important aspects; (i) the hierarchy has considered only upto paragraph level, not till word level, and (ii) weights of different paragraphs in generating document encoding. As observed in (Li et al., 2015), extending the hierarchical structure till lower level helps in various text representation tasks such as semantic relatedness of sentence pairs, sentiment classification and natural language interface task etc. Therefore, extending hierarchical structure till the word level may also help in detecting incongruent news articles. Further, for an incongruent news article, the contributing texts may occupy a small part of the entire text. In such a scenario, the congruent part of the body will dominate the incongruent part. Therefore, it is also important to incorporate *the ability of a constituent paragraph* in representing the entire body while generating the encoding. Motivated by the above observations, this paper proposes a `Gated Recursive And Sequential Deep Hierarchical Encoding` *(GRASHE)* method for detecting incongruent news articles by extends hierarchical encoding upto word level and incorporating incongruently weight of each paragraph. From various experimental observations over three publicly available datasets, it is observed that the proposed method outperforms its bag-of-word, sequential, and hierarchical counterparts.

The key highlights of the contributions are summarised as follows:

1. Propose a Gated Recursive And Sequential Deep Hierarchical Encoding *(GRASHE)* model which captures hierarchical structure till word level, and also captures the incongruent weight of the paragraphs.

2. Perform ablation studies to understand the importance of considering deeper hierarchy and incongruently weight of the paragraphs.

3. Investigate the importance of incorporating explicit features in addition to the hierarchically embedded features in capturing incongruity.

The rest of paper is organised a follow. Section 2 briefly presents related studies. In section 3, we describe our proposed models. Sections 4 presents experimental setup, results, and analysis. The paper concludes in section 5.

2 Related Work

In literature, studies (Shu et al., 2017) (Kumar and Shah, 2018) (Zubiaga et al., 2018) (Sharma et al., 2019) (Zhou and Zafarani, 2020)(Parikh and Atrey, 2018)(D'Ulizia et al., 2021) have briefly reviewed and analysed work related to misinformation and disinformation detection. In this study, we have retrospect work related to incongruent news article detection only. As noted in the study (Chesney et al., 2017), incongruity detection is different from detecting other types of misinformation. Clickbait attempts to attract the attention of the reader, several stylistic and linguistic features are used, such as forward-referencing, mention of the attractive word, public figure, personality, and number. In contrast, though the headline is deceiving or the news article is incongruent, it does not use any stylistic and linguistic feature to attract readers attention. Interestingly, the incongruent headline of the news article does not force clicking some link and follow-up to find the conclusion.

In the literature, several models have been proposed for the detection of incongruent news article. The first fake news challenge (FNC-1) was organized by (Pomerleau and Rao, 2017) to detect the body's stance concerning headlines in news articles. The winner system *SLOAT* in the *SWEN* by talo intelligence of FNC-1 proposed an ensemble model which combines tree model and convolution neural network. Second winner system Team Athene (Hanselowski et al., 2017) trained multi-layer perceptions on Bag of word-based and domain-dependent features.(Riedel et al., 2017) forms concatenated feature vector by combining the term frequency–inverse document frequency (TF-IDF) vector of headline and body along with cosine similarity between TF-IDF vector of headline and body. Then these concatenated features are used to train multi-layer perceptron to classify the relationship between headline and body of news article. Realising the importance of contextual and sequential information, (Hanselowski et al., 2018) combines bag of words and topic modelling-based

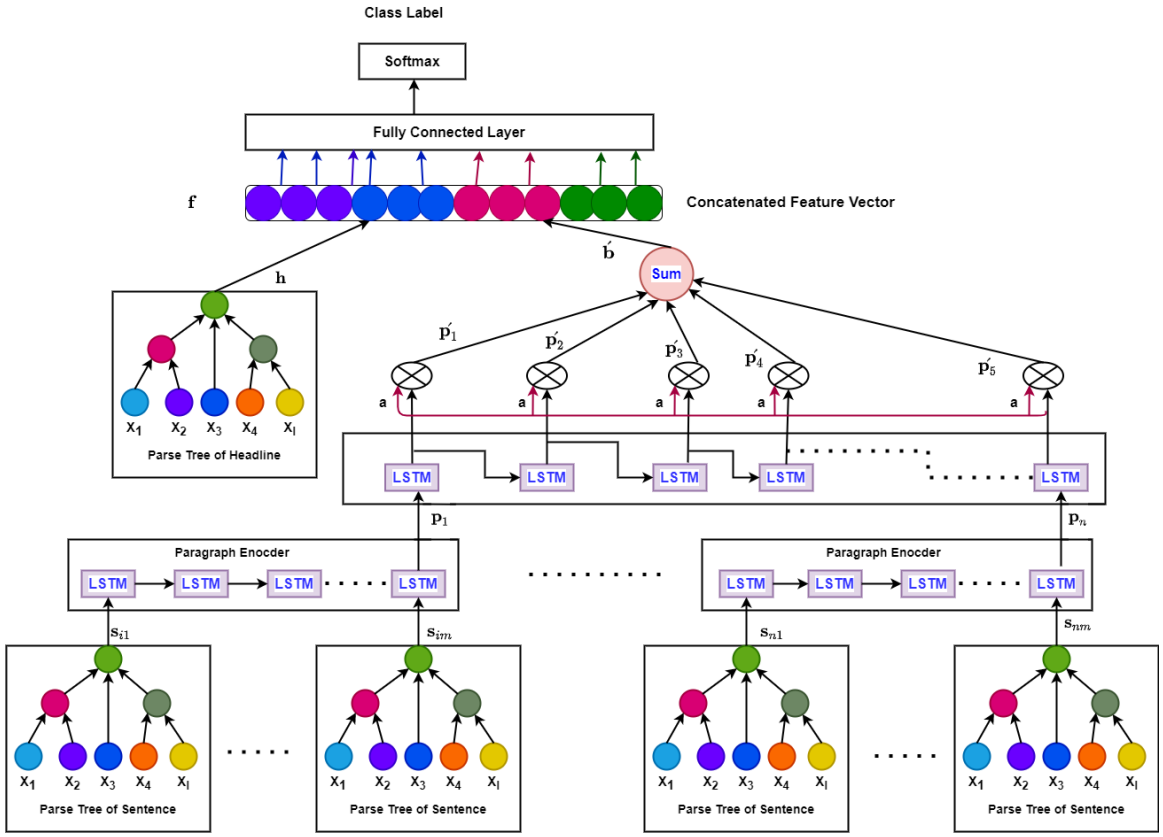

Figure 1: Schematic diagram of the proposed GRASHE model

features with two-layer stack LSTM for classification of the fake news article. (Conforti et al., 2018) adopted inverted pyramid writing style (Scanlan, 2000) of the news article and four-stage pipeline of rumor verification proposed by (Zubiaga et al., 2018). The primary motivation behind adopting an inverted pyramid writing style is to give more attention to the first few lines of news articles, while encoding to find semantic similarities between news headlines and the body. (Borges et al., 2019) encode headline, full-body and first two lines of the body and then apply semantic matching between them. Finally, the outcome of semantic matching between headline, full-body, first two body lines are combined with domain-dependent features used in the study [3] to train a fully connected neural network. Study (Saikh et al., 2020) use bidirectional GRU to encode words in the news article and then applies word-level attention to highlight the important word concerning the target. (Karimi and Tang, 2019) studied the importance of hierarchical structure for fake news classifications.

Recent study (Yoon et al., 2019) propose Attentive Hierarchical Dual Encoder *(AHDE)* and Attentive Hierarchical Dual Encoder Independent Paragraph *(AHDE-IP)* method, which utilises structural property present in the news article. The authors divide news articles into separate paragraphs and encode each separately using a recurrent neural network (RNN) followed by an attention mechanism to select the most relevant paragraphs concerning the headline. Studies (Mishra et al., 2020)(Sepúlveda-Torres et al., 2021) exploited summarization technique to match semantic similarity between news headline and body. (Yoon et al., 2021)proposed graph hierarchical dual encode model learns the similarity between headlines and every paragraph to detect incongruent paragraphs and news articles.

3 Proposed Model

As mentioned above, the objective of the paper is to study the effect of two important aspects of encoding news articles while detecting incongruity; (i) effect of deeper hierarchical encoding by extending the structure upto word level, and (ii) incorporating the weights of different paragraphs defining the ability to represent the encoding of the entire document. Figure 1 shows the schematic diagram of the proposed `Gated Recursive`

[3] TALOS Fake News Challenge

`And Sequential Deep Hierarchical Encoding` *(GRASHE)* model. It defines the hierarchical structure as follows. The body $\mathcal{B}$ of a news article is a sequence of n paragraphs, $\mathcal{B} = \{\mathcal{P}_1, \mathcal{P}_2, \cdots, \mathcal{P}_n\}$. A paragraph P_i is a sequence of m sentences, $\mathcal{P}_i = \{\mathcal{S}_1, \mathcal{S}_2, \cdots, \mathcal{S}_m\}$. A sentence $\mathcal{S}_j$ is defined recursively by a dependency parse tree $\mathcal{S}_j$ consisting of l words, $\mathcal{S}_j = \{w_1, w_2, \cdots, w_l\}$. The body is encoded using a gated sequential model over paragraphs. The paragraphs are encoded using sequential model over the sentences. The sentences are encoded using tree based encoding model over words. Similarly, the headline is also encoded using tree based model.

Given a sentence S, we apply child-sum Tree LSTM (proposed in (Tai et al., 2015)) over its dependency parse tree. Usage of tree encoding has been motivated from the earlier studies that the tree encodings such as *mTreeLSTM* (Tran and Cheng, 2018), *child-sum Tree LSTM* (Tai et al., 2015), *tree-transformer* (Wang et al., 2019) help in capturing long distance dependencies between words. Though ideally any state-of-the art tree encoding method may be used for encoding a sentence, this study has considered *child-sum Tree LSTM* (Tai et al., 2015). The details of the child-sum Tree implementation is explained in section A.1.

A sequence of sentences defines a paragraph. Once encoding of the sentences are obtained, the encoding of a paragraph can be estimated using a sequential model such as RNN, GRU, LSTM, BERT etc. We consider LSTM in this paper. If $\mathbf{s}_{ij}$ denotes the encoding of a sentence $\mathcal{S}_j$ in a paragraph $\mathcal{P}_i$, the encoding $\mathbf{p}_i$ of the paragraph $\mathcal{P}_i$ is obtained from the sequence of sentences encoding in the paragraph using a LSTM model (i.e., hidden state of the last input to LSTM), as follows.

$$\mathbf{p}_i = LSTM\Big(\mathbf{s}_{i1}, \mathbf{s}_{i2}, ..., \mathbf{s}_{im}\Big) \tag{1}$$

To capture the representation of the text from both the directions, the unidirectional LSTM in Equation 1 can also be replaced by bidirectional LSTM. However, we have employed unidirectional LSTM, due to considerable computational overhead of bidirectional LSTM.

Similarly, encoding of a body $\mathcal{B}$ of a news article can be learned from the sequence of underlying paragraph encoding as defined below.

$$\mathbf{a} = LSTM\Big(\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_n\Big) \tag{2}$$

where $\mathbf{a}$ represents the encoding of the body $\mathcal{B}$. As the encoding $\mathbf{a}$ is biased by last sentences in the paragraphs, we further combine all the intermediate encodings to reduce the bias. As the intermediate encoding of paragraphs capture local context, they need to be further regularized with a global context ($\mathbf{a}$ in our case) before combining them. Like in (Zhou et al., 2017), we employ a multi-layer perceptron based select gate as defined in equation 3 and 4.

$$\mathbf{c}_i = \sigma\Big(\mathbf{W}^p\mathbf{p}_i + \mathbf{U}^p\mathbf{a} + \mathbf{b}^p\Big) \tag{3}$$

$$\acute{\mathbf{p}}_i = \sigma\Big(\mathbf{p}_i \odot \mathbf{c}_i\Big) \tag{4}$$

where W^p, U^p and b^p are the select gate parameters and $\odot$ denotes element wise multiplication. The main motive behind using select gate is to capture important aspect of encoded representation $\mathbf{p}_i$ with respected to $\mathbf{a}$.

To obtain the overall representation of the body, we apply a weighted summation of the gated representation of the paragraphs as defined below.

$$\acute{\mathbf{b}} = \sum\nolimits_i \omega_i \acute{\mathbf{p}}_i \tag{5}$$

where ω_i is the weight of the paragraph $\mathcal{P}_i$ representing its importance. Let $\mathbf{M}$ be a matching matrix between the paragraphs i.e., $M_{ij} = \acute{\mathbf{p}}_i^T \cdot \acute{\mathbf{p}}_j$. The matching matrix is then converted to a probability distribution as defined in equation 6.

$$Pr(\mathcal{P}_i \rightarrow \mathcal{P}_j) = \frac{exp(M_{ij})}{\sum_{k \neq i} exp(M_{ik})}. \forall i \neq j \tag{6}$$

where $Pr(\mathcal{P}_i \rightarrow \mathcal{P}_j)$ denotes the probability of paragraph $\mathcal{P}_i$ representing paragraph $\mathcal{P}_j$. If the $Pr(\mathcal{P}_i \rightarrow \mathcal{P}_j)$ for all j, $j \neq i$ is uniformly distributed, it indicates that $\mathcal{P}_i$ represent all other paragraphs equally likely. It means $\mathcal{P}_i$ can represent the body. This can be quantified using entropy of the paragraph $\mathcal{P}_i$ as follows.

$$H(i) = -\sum_{j \neq i} Pr(\mathcal{P}_i \rightarrow \mathcal{P}_j) \log Pr(\mathcal{P}_i \rightarrow \mathcal{P}_j) \tag{7}$$

Thus, entropy of all the paragraphs are estimated. Then, the weight of the paragraph $\mathcal{P}_j$ is defined

Table 1: Characteristics of Experimental Datasets

	Dataset	True	Fake	Total	#Head	#Body	#Para	#Sen
ISOT	Train	17083	18232	35315	9.438	244.325	3.799	16.955
	Test	1726	1815	5313	9.377	236.379	3.729	16.606
	Dev	2607	2706	3541	9.388	241.136	3.733	16.607
FNC	Train	12057	32917	44974	8.478	217.216	11	19.465
	Test	7064	18349	25413	8.503	213.757	10.523	18.744
	Dev	1370	3628	4998	8.465	216.347	10.808	19.215
NELA-17	Train	35710	35710	71420	10.558	551.923	13.494	26.649
	Test	3151	3151	6302	10.529	566.921	13.851	27.526
	Dev	3151	3151	6302	10.547	541.188	13.49	26.256

using softmax over the entropy $H(i)$.

$$\omega_i = \frac{exp(H_i)}{\sum_{k \neq i} exp(H_k)} \tag{8}$$

If we want to assign equal weight to all the paragraphs, we set all the weights to $1/n$ i.e., $\omega_i = 1/n, \forall i = 1..n$. we denote this model as $GRASHE^{(=)}$.

Once we obtain the encoding of the body $\acute{\mathbf{b}}$ and headline $\mathbf{h}$ (obtained using child-sum Tree LSTM), we further estimate the following two vectors $\mathbf{sim}$ and $\mathbf{diff}$ capturing similarity and difference between the body and headline.

$$\mathbf{sim} = \acute{\mathbf{b}} \odot \mathbf{h} \tag{9}$$

$$\mathbf{diff} = \acute{\mathbf{b}} - \mathbf{h} \tag{10}$$

Now, we define the final feature for the classification as follow.

$$\mathbf{f} = \bar{\mathbf{b}} \oplus \mathbf{h} \oplus \mathbf{sim} \oplus \mathbf{diff} \tag{11}$$

where $\oplus$ denotes concatenation of vectors. The feature vector $\mathbf{f}$ is then passed through a dense layer with a *softmax* output layer. We apply cross entropy loss to learn the parameters.

4 Experimental setups and discussions

4.1 Dataset

This study uses three publicly available datasets namely ISOT fake news dataset (Ahmed et al., 2018) (Ahmed et al., 2017) , FNC dataset (Pomerleau and Rao, 2017), and NELA-17 dataset (Horne et al., 2018) (Yoon et al., 2019). Table 1 presents the characteristics these datasets. The FNC dataset has four classes namley agree, disagree, discuss, and unrelated. The samples in agreed, disagree, discuss classes are merged and named a *True* class, whereas the samples in unrelated class are considered *fake* class. For NELA dataset, we curate the samples following the procedure reported in (Yoon et al., 2019) over the news corpus provided at (Horne et al., 2018). The news articles published by authenticated sources are labelled as *true* class, and the fake samples are generated by randomly inserting paragraphs from another news article into true class news articles. The ISOT and NELA datasets are relatively balance, whereas the FNC dataset is highly imbalanced.

4.2 Experimental setups

To compare the performance of the proposed method with the other existing methods, we have considered the following baseline systems.

- `Fake News Challenge` *(FNC)* (Pomerleau and Rao, 2017): Three systems submitted to the challenge are considered namely the baseline provided by the organizer (FNC-1)[4], the winning system (XGBoost[5]), and *UCL Machine Reading (UCLMR)*[6]*(Riedel et al., 2017)*.

- `StackLSTM` (Hanselowski et al., 2018) [7]:It combines various topic modelling features (LSI-topic, NMF-topic, NMF-cos, LDA-cos) with embedding obtained with StackLSTM over the top two hundred words in the news article.

- `Attentive Hierarchical Dual Encoder` *(AHDE)* (Yoon et al., 2019)[8]: It is the first hierarchical model reported in literature for detecting incongruity.

- `Graph-Based Hierarchical Dual Encoder` *(GHDE)* (Yoon et al., 2021)[9]: It is the most recent study in incongruity detection. As it needs paragraph level annotations, it has been tested only with NELA dataset, where the inserted paragraphs are annotated as fake.

In addition to the above baseline methods reported in recent studies, we also build the following baselines locally.

- `RASHE`: This model is the $GRASHE$ without the gates and the weighting aggregation, i.e., the output of the top LSTM model is used as the encoding of the body.

[4]FNC-1 baseline by organizer
[5]FNC-1 Winner : XGBoost
[6]FNC Winner : UCLMR
[7]StackLSTM
[8]ADHE
[9]GHDE

Table 2: Comparison of the performances of different models over three benchmark datasets.

		Models	NELA-17 Acc	NELA-17 F	ISOT Acc	ISOT F	FNC Acc	FNC F
Baseline Systems	Explicit Features	FNC	0.586	0.586	0.844	0.844	0.586	0.496
		$XGBoost$	**0.699**	**0.699**	**0.989**	**0.989**	**0.977**	**0.971**
		$UCLMR$	0.589	0.588	0.997	0.997	0.964	0.955
		MLP	0.603	0.600	0.985	0.985	0.917	0.903
		$StackLSTM$	0.597	0.591	0.992	0.992	0.971	0.963
	Without Features	$LSTM$	0.555	0.55	0.990	0.990	0.616	0.504
		$BERT$	0.572	0.563	0.894	0.894	**0.722**	0.419
		$AHDE$	**0.606**	**0.606**	0.913	0.913	0.666	0.487
		$GHDE$	0.55	0.33	-	-	-	-
		$RASHE^{(LSTM)}$	0.603	0.603	**0.997**	**0.997**	0.689	**0.597**
Proposed Systems	Proposed	$GRASHE^{(=)}$	**0.664**	**0.663**	**0.999**	**0.999**	0.715	**0.629**
		$GRASHE$	0.63	0.63	0.998	0.998	**0.718**	0.505
		$RASHE$	0.652	0.652	**0.999**	**0.999**	0.712	0.624
	Extension with Features	$UCMLR^{(F)}$	0.589	0.588	0.997	0.997	0.964	0.955
		$StackLSTM^{(F)}$	0.597	0.591	0.992	0.992	**0.971**	**0.963**
		$RASHE^{(LSTM,F)}$	0.626	0.626	0.995	0.995	0.962	0.962
		$GRASHE^{(=,F)}$	**0.656**	**0.656**	**0.999**	**0.999**	0.963	**0.963**
		$RASHE^{(F)}$	0.601	0.599	0.995	0.995	0.963	**0.963**

- $RASHE^{(LSTM)}$: It is a $RASHE$ model with LSTM for encoding sentences instead of $child - sumTreeLSTM$.

- `BERT` (Devlin et al., 2018): Considering the encouraging observations of BERT (for various NLP tasks) in recent studies, we also build a classifier using BERT embedding generated over the text in headline and body.

- `LSTM`: Like BERT, we also build a classifier using LSTM embedding generated over the text in headline and body.

- `MLP`: Importance of explicit features have been evident in XGBoost. Considering count, TF-IDF similarity, SVD top-k vector and sentiment features used in XGBoost, a multilayer perceptron-based classifier is also built.

For all the experimentsGoogle's worde2vec (Mikolov et al., 2013) pre-trained embeddings are used, and F-measure (F), Accuracy (Acc) have been used as evaluation metrics.

4.3 Results and discussion

The performance of different systems are compared in Table 2. As shown in the table, the experiments are organized into four groups; (i) *Explicit Features*: FNC, XGBoost, UCLMR, MLP and StackLSTM (ii) *Without Features*: BERT, AHDE, GHDE and $RASHE^{(LSTM)}$ (iii) *Proposed*: $GRASHE^{(=)}$,$GRASHE$ and $RASHE$, and (iv) *Extension with Features*: $GRASHE^{(=,F)}$, $RASHE^{(LSTM,F)}$ and $RASHE^{(F)}$.

Table 3: Comparison performance of hierarchical structure-based model versus non-hierarchical sequential model.

Model	NELA-17 Acc	NELA-17 F	ISOT Acc	ISOT F	FNC Acc	FNC F
$LSTM$	0.555	0.55	0.991	0.99	0.616	0.504
$RASHE^{(LSTM)}$	0.603	0.603	**0.997**	**0.997**	**0.689**	**0.597**
$AHDE$	**0.606**	**0.606**	0.913	0.913	0.666	0.487

We first compare the methods in *baseline group*. As shown in the table, for NELA-17 and FNC datasets, XGBoost outperforms all other baseline models. Whereas, for ISOT dataset $RASHE^{(LSTM)}$ outperforms all other baseline models. Thereafter, we compare the proposed models with baseline models. Since the objective of the proposed method is to consider recursive encoding of sentences by exploiting the deep hierarchical structure of news article, the direct comparison is with AHDE method. The table 2 shows that our proposed methods ($GRASHE$ and $GRASHE^{(=)}$) outperform AHDE for all the three datasets. $GRASHE$ outperforms AHDE by 3.96%, 9.3% and 7.8% over NELA-17, ISOT and FNC datasets, respectively. Similarly, $GRASHE^{(=)}$ outperforms AHDE by 9.57%, 9.41% and 7.35% over NELA-17, ISOT and FNC datasets, respectively.

It clearly shows that recursive encoding of sentences helps in capturing better representation of the body, and hence provides better classification performance for detecting incongruent news articles. To validate these observations, we further compare the performance of the proposed models with $RASHE^{(LSTM)}$. It also shows that both $GRASHE^{(=)}$ and $GRASHE$ outperform the $RASHE^{(LSTM)}$.

From table 2, it is apparent that $GRASHE^{(=)}$ (which considers equal weight to all the paragraphs) outperforms $GRASHE$. It indicates that every paragraph in the body contribute in detecting incongruity. Lastly, we study the effect of incorporating explicit features with different systems. From table 2 it is evident that $GRASHE^{(=,F)}$ outperformed all other models in the group *extension with feature* over the NELA-17 and ISOT datasets. However, stackLSTM model outperform $GRASHE^{(=,F)}$ with small margin over FNC dataset.

Table 4: Performance of sequential encoding of sentence versus recursive encoding of sentence structure by exploiting hierarchical structure of news article.

Model	NELA-17		ISOT		FNC	
	Acc	F	Acc	F	Acc	F
$GHDE$	0.550	0.330	-	-	-	-
$AHDE$	**0.606**	**0.606**	0.913	0.913	0.666	0.487
$RASHE^{(LSTM)}$	0.603	0.603	**0.997**	**0.997**	**0.689**	**0.597**
$GRASHE^{(=)}$	**0.664**	**0.663**	**0.999**	**0.999**	0.715	**0.629**
$GRASHE$	0.630	0.630	0.998	0.998	**0.718**	0.505

Table 5: Comparison of Model performance over Different Number of Paragraphs.

Model	# of Paragraphs	NELA-17		ISOT		FNC	
		Acc	F	Acc	F	Acc	F
$RASHE$	*Full*	**0.652**	**0.652**	**0.999**	**0.999**	0.712	0.624
$RASHE$	*First* 2	0.550	0.547	0.988	0.988	**0.722**	**0.640**

4.3.1 Is hierarchical encoding important?

Considering the earlier studies on incongruity detection of news articles with and without hierarchical encoding, the following question is aroused. *Does hierarchical encoding of the body helps in detecting incongruent news articles?* To answer this question, we compare the performance of different hierarchical models and non-hierarchical models in table 3. This ablation study considers an LSTM model that encodes concatenated news headlines and bodies without exploiting any hierarchical structure of news articles. In contrast, $RASHE^{(LSTM)}$ and $AHDE$ models are hierarchical encoding-based models. Table 3 shows that both $RASHE^{(LSTM)}$ and $AHDE$ outperform LSTM model over NELA-17, FNC and ISOT datasets. However, the performance of LSTM model over ISOT dataset is very close to $RASHE^{(LSTM)}$ and $AHDE$ models. The size of the news articles in ISOT data are small (see table 1) as compared to FNC and NELA-17. From the above observations, we may claim that for the small articles, considering hierarchical structure may not be necessarily beneficial, as compared to that of larger articles.

4.3.2 Does recursive encoding of sentences help in improving body representation?

In our proposed model, we have used tree encoding rather than sequential encoding. *Do the tree-based encoding beneficial?* To understand this, Table 4 compares the hierarchical models with and without the tree-based encoding. The proposed $GRASHE^{(=)}$ and $GRASHE$ models are compared with $GHDE$, $AHDE$ and $RASHE^{(LSTM)}$ models without tree encoding. It can be observed from the table 4 that both the recursive encoding-based models (tree encoding) $GRASHE^{(=)}$ and $GRASHE$ outperform all sequential encoding-based methods $AHDE$, $GHDE$ and $SRASHE^{(LSTM)}$ models with significant margins for all three datasets. This is not surprising because recursive encoding of the sentence helps in capturing long-distance dependencies between words within sentences.

Considering recursive encoding of sentence structure, $GRASHE$ follows further deep hierarchical structure compared to other hierarchical structure models presented in table 4. GHDE, AHDE models hierarchical structure limits up to paragraph level only. Due to recursive encoding of sentences $GRASHE^{(=)}$ and $GRASHE$ models, the hierarchical structure goes further deeper upto word level. *From the above observations, it can be claimed that recursive encoding of sentence structure boosts the performance in incongruent news article detection.*

4.3.3 Do we need to consider full body?

Study (Scanlan, 2000) suggests that in a news article with a large number of paragraphs, every part may not be helpful in regards to incongruent news article detections. To investigate this, we further implement two experiments with $RASHE$ model as shown in Table 5; (i) considering only the first two paragraphs, and (ii) which encode full news articles. We observe from the table that the $RASHE$ model with reduced document provides comparable or better performance than $RASHE$ with full document except for the NELA-17 dataset. It shows that we will be able to reduce model computational time without compromising the classification performance with appropriate sentence selection approaches. It may even provide better performance after removing the noisy sentences. Document summarization to enhanced document encoding is left as a possible future work.

4.3.4 Importance of incorporating features of different aspects

Our empirical study suggested that similarity between headline and body based on TF-IDF, SVD top-k vectors and count, sentiment features play an essential role in incongruent news article detection.

Table 6: Comparison of performance of models over FNC dataset different topic distribution *(*FNC^0*)* versus FNC with similar topic distribution *(FNC^R)* in train and test.

	FNC		FNC^R	
Model	**Acc**	**F**	**Acc**	**F**
$AHDE$	0.666	0.487	0.661	0.472
$RASHE^{(LSTM)}$	0.689	0.597	0.809 ↑	0.764 ↑
$GRASHE^{(=)}$	0.712	0.624	**0.847** ↑	0.809 ↑
$RASHE$	0.715	0.629	0.842 ↑	0.805 ↑

This study identifies 151 features (TF-IDF and SVD similarity between headline and body, count and sentiment features) from the ten million-plus features used in Xgboost. We concatenate these 151 features with the concatenated feature vector obtained in equation 11 and pass to a fully connected layer. We do it with all the proposed model and its variations $RASHE^{(LSTM)}$, $GRASHE^{(=)}$ and $RASHE$, and named them $RASHE^{(LSTM,F)}$, $GRASHE^{(=,F)}$ and $RASHE^{(F)}$ respectively. From table 2, it is evident that by incorporating important features from a different domain, our proposed models provide comparable results over ISOT and FNC datasets. However, for the NELA-17 dataset, there is a reduction in performance after adding handcrafted features. It indicates that feature engineering needs an understanding of the underlying datasets. We further build another multi-layer perceptron-based classifier MLP over the 151 manual features to investigate the response of the features. From table 2, it is observed that MLP+Feature under-performs $RASHE^{(LSTM,F)}$ and $GRASHE^{(=,F)}$ and outperforms several other baselines. Dataset domain-specific feature engineering and adaptation of the proposed model is not included in this study but left as a future task.

4.3.5 Effect of Domain Dependency

The domain of the news article gives insight about the news article. So, it becomes crucial to study the impact of the domain on incongruent news article detection task. *Is incongruent news article detection is domain-independent or domain-dependent? What happens if models for incongruent news article detection model is trained over news article from domain of certain domain and test over news article from domain of different domain*. To answer such questions, we conduct an empirical ablation study over the FNC dataset. FNC dataset provided by FNC-1 contest has different topics distributions in training and test set. The training set has news articles from 200 domain, and the test has news articles from 100 domain, and there is no common domain in training and test. From table 2 it can be observed that the performance of deep learning-based models is inferior compared to feature-based models. The main reason behind the poor performance of deep learning-based model over FNC data sets is that the news articles from train and test belong to different domain. To confirm this observation, we created another dataset as follows: *we merged train and test set provided by the FNC organiser and randomly permute the sample in merged FNC data sets. Then created train and set with the same distribution as the original FNC dataset*. We called this newly created data set FNC topic overlap dataset FNC^R. From table 6 it can be observed that the performance of $RASHE^{(LSTM)}$, $GRASHE^{(=)}$ and $RASHE$ models significantly improved over the FNC^R data set compared to the performance of these models over the FNC dataset. Hence, we can conclude that incongruent news article detection is a domain dependent task. So training and test should have news articles from the same domain distribution.

5 Conclusions and Future works

This paper proposed Gated Recursive And Sequential Deep Hierarchical Encoder model, namely $GRASHE$, to detect incongruent news articles. The proposed models capture syntactic structures at the sentence level and sequential structures at the body and paragraph level. From various experiments over three datasets, it is observed that capturing structural properties at the sentence level improved the performance of incongruent news article detection tasks. From the above observations, we identify the following four potential future works; (i) Incorporating features of different nature with the hierarchical modelling, (ii) Identifying appropriate feature engineering for the datasets of different nature, and (iii) Devising appropriate document summarization to reduce document size for incongruent news article detection. (iv) Based on our observation from table 6 build a domain-independent deep learning model for incongruent news article detections.

References

Hadeer Ahmed, Issa Traore, and Sherif Saad. 2017. Detection of online fake news using n-gram analysis and machine learning techniques. In *International conference on intelligent, secure, and dependable systems in distributed and cloud environments*, pages 127–138.

Hadeer Ahmed, Issa Traore, and Sherif Saad. 2018. Detecting opinion spams and fake news using text classification. *Security and Privacy*, 1(1):e9.

Blake C Andrew. 2007. Media-generated shortcuts: Do newspaper headlines present another roadblock for low-information rationality? *Harvard International Journal of Press/Politics*, 12(2):24–43.

Luís Borges, Bruno Martins, and Pável Calado. 2019. Combining similarity features and deep representation learning for stance detection in the context of checking fake news. *Journal of Data and Information Quality (JDIQ)*, 11(3):1–26.

Sophie Chesney, Maria Liakata, Massimo Poesio, and Matthew Purver. 2017. Incongruent headlines: Yet another way to mislead your readers. In *Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism*, pages 56–61.

Costanza Conforti, Mohammad Taher Pilehvar, and Nigel Collier. 2018. Towards automatic fake news detection: cross-level stance detection in news articles. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 40–49.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Julio Cesar Soares Dos Rieis, Fabrício Benevenuto de Souza, Pedro Olmo S Vaz de Melo, Raquel Oliveira Prates, Haewoon Kwak, and Jisun An. 2015. Breaking the news: First impressions matter on online news. In *Ninth International AAAI conference on web and social media*, pages 357–367.

Arianna D'Ulizia, Maria Chiara Caschera, Fernando Ferri, and Patrizia Grifoni. 2021. Fake news detection: a survey of evaluation datasets. *PeerJ Computer Science*, 7:e518.

Ullrich KH Ecker, Stephan Lewandowsky, Ee Pin Chang, and Rekha Pillai. 2014. The effects of subtle misinformation in news headlines. *Journal of experimental psychology: applied*, 20(4):323.

Maksym Gabielkov, Arthi Ramachandran, Augustin Chaintreau, and Arnaud Legout. 2016. Social clicks: What and who gets read on twitter? In *Proceedings of the 2016 ACM SIGMETRICS international conference on measurement and modeling of computer science*, pages 179–192.

Andreas Hanselowski, PVS Avinesh, Benjamin Schiller, and Felix Caspelherr. 2017. Description of the system developed by team athene in the fnc-1. *Fake News Challenge*.

Andreas Hanselowski, Avinesh PVS, Benjamin Schiller, Felix Caspelherr, Debanjan Chaudhuri, Christian M Meyer, and Iryna Gurevych. 2018. A retrospective analysis of the fake news challenge stance detection task. *arXiv preprint arXiv:1806.05180*.

Benjamin Horne, Sara Khedr, and Sibel Adali. 2018. Sampling the news producers: A large news and feature data set for the study of the complex media landscape. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 12.

Hamid Karimi and Jiliang Tang. 2019. Learning hierarchical discourse-level structure for fake news detection. *arXiv preprint arXiv:1903.07389*.

Srijan Kumar and Neil Shah. 2018. False information on web and social media: A survey. *arXiv preprint arXiv:1804.08559*.

Jiwei Li, Minh-Thang Luong, Dan Jurafsky, and Eudard Hovy. 2015. When are tree structures necessary for deep learning of representations? *arXiv preprint arXiv:1503.00185*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Rahul Mishra, Piyush Yadav, Remi Calizzano, and Markus Leippold. 2020. Musem: Detecting incongruent news headlines using mutual attentive semantic matching. In *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 709–716.

Shivam B Parikh and Pradeep K Atrey. 2018. Media-rich fake news detection: A survey. In *2018 IEEE conference on multimedia information processing and retrieval (MIPR)*, pages 436–441.

Dean Pomerleau and Delip Rao. 2017. The fake news challenge: Exploring how artificial intelligence technologies could be leveraged to combat fake news. *Fake News Challenge*.

Benjamin Riedel, Isabelle Augenstein, Georgios P Spithourakis, and Sebastian Riedel. 2017. A simple but tough-to-beat baseline for the fake news challenge stance detection task. *arXiv preprint arXiv:1707.03264*.

Tanik Saikh, Arkadipta De, Asif Ekbal, and Pushpak Bhattacharyya. 2020. A deep learning approach for automatic detection of fake news. *arXiv preprint arXiv:2005.04938*.

Christopher Scanlan. 2000. *Reporting and writing: Basics for the 21st century*. Harcourt College Publishers.

Robiert Sepúlveda-Torres, Marta Vicente, Estela Saquete, Elena Lloret, and Manuel Palomar. 2021. Headlinestancechecker: Exploiting summarization to detect headline disinformation. *Journal of Web Semantics*, 71:100660.

Karishma Sharma, Feng Qian, He Jiang, Natali Ruchansky, Ming Zhang, and Yan Liu. 2019. Combating fake news: A survey on identification and mitigation techniques. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(3):1–42.

Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake news detection on social media: A data mining perspective. *ACM SIGKDD explorations newsletter*, 19(1):22–36.

Kai Sheng Tai, Richard Socher, and Christopher D Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. *arXiv preprint arXiv:1503.00075*.

Nam-Khanh Tran and Weiwei Cheng. 2018. Multiplicative tree-structured long short-term memory networks for semantic representations. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 276–286.

Yau-Shian Wang, Hung-Yi Lee, and Yun-Nung Chen. 2019. Tree transformer: Integrating tree structures into self-attention. *arXiv preprint arXiv:1909.06639*.

Seunghyun Yoon, Kunwoo Park, Minwoo Lee, Taegyun Kim, Meeyoung Cha, and Kyomin Jung. 2021. Learning to detect incongruence in news headline and body text via a graph neural network. *IEEE Access*, 9:36195–36206.

Seunghyun Yoon, Kunwoo Park, Joongbo Shin, Hongjun Lim, Seungpil Won, Meeyoung Cha, and Kyomin Jung. 2019. Detecting incongruity between news headline and body text via a deep hierarchical encoder. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 791–800.

Qingyu Zhou, Nan Yang, Furu Wei, and Ming Zhou. 2017. Selective encoding for abstractive sentence summarization. *arXiv preprint arXiv:1704.07073*.

Xinyi Zhou and Reza Zafarani. 2020. A survey of fake news: Fundamental theories, detection methods, and opportunities. *ACM Computing Surveys (CSUR)*, 53(5):1–40.

Arkaitz Zubiaga, Ahmet Aker, Kalina Bontcheva, Maria Liakata, and Rob Procter. 2018. Detection and resolution of rumours in social media: A survey. *ACM Computing Surveys (CSUR)*, 51(2):1–36.

A Appendix

A.1 TreeLSTM using Dependency Parse Tree

Given a sentence S and it's a dependency parse tree, let $ch(j)$ denotes the set of children nodes of node j. Like in LSTM, child-sum tree LSTM also has hidden state h and cell state c. The hidden state of the node j is defined by the sum of the initial hidden states of its children nodes as follows.

$$\acute{\mathbf{h}}_j = \sum_{k \in ch(j)} \mathbf{h}_k \tag{12}$$

Using the initial hidden state $\mathbf{h}_j$, the corresponding input, output and intermediate cell gates of node j are estimated as follows.

$$\mathbf{i}_j = \sigma\Big(\mathbf{W}^{(i)}\mathbf{x}_j + \mathbf{U}^{(i)}\acute{\mathbf{h}}_j + \mathbf{b}^{(i)}\Big) \tag{13}$$

$$\mathbf{o}_j = \sigma\Big(\mathbf{W}^{(o)}\mathbf{x}_j + \mathbf{U}^{(o)}\acute{\mathbf{h}}_j + \mathbf{b}^{(o)}\Big) \tag{14}$$

$$\mathbf{u}_j = tanh\Big(\mathbf{W}^{(u)}\mathbf{x}_j + \mathbf{U}^{(u)}\acute{\mathbf{h}}_j + b^{(u)}\Big) \tag{15}$$

where $\mathbf{x_j}$ denotes embedding of the word w_j, $\mathbf{b}^{(.)}$ denotes the bias, $\mathbf{W}^{(.)}$ and $\mathbf{U}^{(.)}$ denote the parameter matrices for respective gates. Unlike traditional LSTM, child-sum tree LSTM has multiple forget gates, one for each child node. It allows each child node to incorporate the information selectively. The forget gate for the k^{th} child of the node j is defined as follows.

$$\mathbf{f}_{jk} = \sigma\Big(W^{(f)}\mathbf{x}_j + \mathbf{U}^{(f)}\mathbf{h}_k + \mathbf{b}^{(f)}\Big) \tag{16}$$

The final cell state and hidden state of node j is defined as follows, respectively.

$$\mathbf{c}_j = \mathbf{i}_j \odot \mathbf{u}_j + \sum_{\mathbf{k} \in \mathbf{ch}(j)} \mathbf{f}_{jk} \odot \mathbf{c}_k \tag{17}$$

$$\mathbf{h}_j = \mathbf{o}_j \odot tanh(\mathbf{c}_j) \tag{18}$$

The hidden state of the root node defines the encoding of the sentence.