Multi-Task Learning Framework for Stance Detection and Veracity Prediction

Fatima Alkhawaldeh [a, b, *], Tommy Yuan [a, c] and Dimitar Kazakov [a, d]
[a] *Department of Computer Science, University of York, YO10 5GH, York, UK*
[b] E-mail: *ftma500@york.ac.uk*
[c] *E-mail: tommy.yuan@york.ac.uk*
[d] *E-mail: dimitar.kazakov@york.ac.uk*

Abstract: Detecting unverified information automatically has become an essential topic of research as many people get their news through online sources. Stance detection and rumour verification tasks have gained increasing interest in recent research. Most of the existing models train both tasks separately or consider the sources contributes equally. This paper proposes a multi-task learning framework for jointly predicting evidence source stance and veracity claim on news, to enhance the performance of veracity prediction. One of the main goals of this model is to generate the best conclusion from the available evidence in case of a lengthy article. Another goal is to detect each evidence stance toward a particular claim then qualify its confidence among conflict facts using a unified model that could be adapted to the news domain. This work implements an argumentation-based truth discovery approach to reason about contradiction beliefs, given by various sources with different reliability levels. Experiments on Emergent and SemEval 2019 Task 7 datasets show that this method outperforms previous methods on both stance classification and veracity prediction.

Keywords**:** Stance Detection, Rumour Veracity, Truth Discovery, Fake News

1. Introduction

The emergence of social media provides a cheaper and more rapid way to publish information compared to traditional media such as newspapers and television; this, however, contributes to more false and fake news that could mislead people. The speed of spreading online information increases the amount of unverified or inaccurate information, particularly in the news domain. Checking the truthfulness of information is an essential task in the field of journalism. Computational fact checking tries to discover unverified information from published posts by checking them against reliable sources using qualified journalists and other experts. The misleading and manipulated information could be in different forms, such as texts, images, and videos. In this work, the focus only on the text form.

In the news domain, rumour or fake news are news articles published by a news source that is intentionally misleading the reader [1]–[3]. Harsin [4] defines rumour as a claim whose truthfulness is doubtful where the intention is ambiguous, which comes from unknown origin like ideological or partisan sources. Several efforts have been devoted to automatically checking the veracity of information such as transforming Wikipedia into a network of knowledge by Ciampaglia et al. [5] where the right information is connected in a graph. Other systems consider the language style of the article as style-based fake news detection, e.g. [6]. Fact-checking websites (e.g., Emergent, PolitiFact, Snopes) verify the claims sent by users or found on the website itself by human professionals. The term ‘rumour’ refers to unverified information that can be believed later to be true or false [7]. Due to the contradiction of relevant sources towards the published posts, the problem of rumour detection is complicated. To address this issue, several truth discoveries

approaches have been proposed by different works to estimate the reliability of attacking/supporting sources to decide the more trustworthy ones.

Truth discovery algorithms aim to solve conflict information that comes from multiple sources focusing on the same matter [8]. Due to variant qualities and reliabilities of multi-sourced information that attack/support a claim, which should be considered to check the veracity of information, the analysis and assessment of the information credibility based on single-source fails where it could be biased and is less likely to be trustworthy compared with multiple supporting sources. The challenge of scarcity of labelled data for source reliability leads to the dependency on source reliability estimation without any supervision [9]. Truth discovery algorithms from data mining and crowdsourcing perspectives have been expanded, where other perspectives such as argumentation may play a key role as discussed in [10], [11]. They claimed that in both truth discovery and argumentation issues, there are contradictions between the information of objects: conflicting 'facts' for truth discovery and attacking arguments for argumentation. They further argue that truth discovery can be mapped to a particular case of argumentation, e.g. bipolar argumentation, where arguments may support as well as attack each other, and the aim is to find 'acceptable' arguments from a collection of conflicting arguments.

Mostly, the state-of-the-art methods for fake news detection are proposed for either stance detection or veracity checking separately in spite of these tasks are closely related and the stances from different people can be utilised to predict the final veracity of the claim; in other words, stance aggregation features are essential for effective veracity prediction. This limitation restricts the ability to generalise models. To tackle this problem, this work proposes to integrate the two tasks and learns them jointly to facilitate veracity prediction based on stance detection. Moreover, to distinguish fake from real information it is important to check their source reliability who initially spread information about the claim so that this work combine truth discovery to estimate the claim credibility and user reliability simultaneously.

In summary, the limitations of the state-of-the-art methods are as follows:

- The limited applicability to handle fake news of emerging topics and inability to generalise models.
- The scarcity of large enough dataset with reliably labelled fake news for training useful models.
- Depending highly on feature engineering which needs intense manual effort.
- Treating stance detection, rumour veracity and truth discovery separately.
- Considering all sources contributing equally.

The proposed framework alleviates these limitations and consider claims that are either supported by the supporting sources or refuted by the attacking sources. For each claim paired with relevant evidence sources, there is a stance such as agree, disagree, discuss, or unrelated. The public stance from conflicting information describes the factuality of the claim. This paper focuses on two tasks. The first task is stance detection, aiming to determine the stance of each claim, which belongs to supporting, attacking, discussing, or not related. The second task is veracity prediction to predict the veracity of the claim belonging to true, false or unverified. Predicting the veracity of the identified news articles is highly related to the problem of rumours veracity classification. When a rumour's veracity value is *untrue*, some researches call them *fake news* [7], [12], [13].

This paper focuses on the two tasks in SemEval-2019 Task 7 (Rumour Eval 2019), as one task in this work, with datasets from Twitter and Reddit [14] and Emergent dataset [15]. Task A refers to stance classification where a claim-post is labelled with one of four options: Support, Deny, Query and Comment. Task B concerns the veracity prediction released where the rumour is classified as True, False or Unverified. Current studies treat stance detection and veracity prediction as separate tasks while they are highly correlated so they should be treated as a joint. Since analysing various stances on the concerned information is meaningful and valuable for claim veracity prediction, this work proposes a multi-task learning scheme to combine both tasks: stance detection and veracity checking. Moreover, it considers truth discovery to integrate the source credibility information, i.e. not consider equal contribution for diverse sources. Paying more attention to the source reliability as the estimated credibility for various sources is distinct, and it has a strongly connected task with veracity prediction tasks and incorporates the source reliability information into the veracity prediction detection.

The **contributions** of this work are:

- proposing a new framework to tackle fake news stance classification and veracity checking jointly. To the best of authors' knowledge, this work is the first to employ argumentation-based truth discovery.

- Proposing a novel model for the best evidence conclusion generator within the framework.
- Experimental results on Emergent and SemEval 2019 datasets demonstrate that this framework performs better than existing methods in both stance classification and veracity checking.
- This framework can handle the emerging rumours (unseen data) spreading very fast in social media and minimise the harmful effect of it, as well as the seen data.

The remainder of this paper is organised as follows. Section 2 briefly reviews the related work from the literature. Section 3 presents the proposed Argumentation-based Truth Discovery Model and shows how truth discovery may be formulated in terms of bipolar argumentation. Experiments and results analysis are discussed in section 4, and section 5 concludes this paper.

2. Related Work

Fake news articles can be considered deception texts [16]. Three types of fake news are described in [17]: Serious Fabrications, Hoaxes and Satire. Linguistic information is considered in [18] as a critical factor for false information detection. For fake news detection, different methods are used. Classic machine learning algorithms use a set of pre-defined linguistic features and a large amount of labelled data. Others find a good influence by implementing modern neural network models relying on pre-trained word vectors and embedded representations [19], [20]. Four components for a rumour classification system are described in [21]: rumour detection, rumour tracking, rumour stance classification, and rumour veracity classification. A cluster ranking algorithm is applied in [22] where their likelihood of being rumours ranks the clusters of tweets. The unsupervised manner is also implemented in [23] based on a sequential classifier where the classifier learns features of rumours such as lexical and temporal. For stance detection, some works employ NLP component, e.g. Saikh et al. [24] integrated textual entailment with stance classification using statistical machine learning and deep learning approaches. This work briefly discusses some models close to the primary goal of this work. It starts with stance classification, then veracity checking, next joint stance and veracity prediction, after that truth discovery and finally the related datasets.

2.1. Stance Classification

In [15], Ferreira and Vlachos proposed a logistic regression model using the lexical and semantic features of news headlines as evidence to predict whether it is for, against or observing a claim. Table 1 shows some state-of-the-art that uses different models for the stance detection task are used as baselines in [25], to compare their model against them. This models' results compared with the model in [25] on Emergent data since their work outperforms all of them, as shown in table 1. They achieve the best stance detection performance for the relative score.

Table 1: A summary of stance detection related work

The model	The implementation details	Relative score
LSTM (BiLSTM)	Stance Detection with Bidirectional Conditional Encoding. The encoded claim is used as initial states to encode the evidence [26] after the 100-d GloVe word embedding is applied [27]	78.70
Attentive CNN (AtCNN)	For both claim and evidence feature representations, the convolutional neural network is used and attention mechanism to extract the most relevant features [28]	75.77
Memory Network (MN)	Combination of convolutional and recurrent neural networks by an end-to-end memory network is implemented [29]	79.92
Ranking Model (RM)	Ranking model to maximise the difference between the four stances representation agree, disagree, discuss, unrelated [30]	86.66
Official Baseline (OB)	gradient boosting decision trees model for stances [31]	75.20
Logistic Regression (LR)	After checking whether the source is related or not by n-gram matching and rule-based methods. The stances: agree, disagree and discussed are decided by Logistic Regression [32]	80.63
Gradient Boosted Decision Trees (GBDT)	Apply Gradient Boosted Decision Trees to detect related stance and apply another Gradient Boosted Decision Trees to detect the remaining three stances [33]	86.72
Multi-Layer Perception (MLP).	Cosine similarity between claims and evidence, and Multi-Layer Perception for the four stances [34]	81.72
Hierarchical representation of a neural network	Hierarchical representation of these classes, which combines agree, disagree, and discuss classes under a new related class where the hierarchical architecture alleviates the class imbalance problem. One neural network layer for related stance detection and the second layer is for the three stances detection [25]	**88.15**

Stance detection is proved to be the essential task for rumour verification in different studies [22], [35]–[42] and some studies proposed multi-task learning framework for jointly predicting rumour stance and veracity, e.g. [43]. Different models were submitted to *Rumour Eval 2019 competition* [14]. In this competition, for the *stance detection task*, the best performing system for Twitter and Reddit datasets is reported in [44] which uses the inference chain of conversation from source post to replies and depends on features such as the number of question words, presence of BiLSTM and Transformer rumour words, false synonym and false antonym. The second rank is [45], which uses an ensemble of BERT, and the third-best system [46] uses pre-trained representation with OpenAI GPT. The Pre-training representation models [47] and ELMO [48] have shown promising results where the representation for each word is based on the entire context in which it is used.

2.2. Veracity Checking

Most veracity checking systems have been developed over FEVER [49]. FEVER is a large-scale dataset for fact extraction and verification that consists of 185,445 claims and their related evidence. The best performance on the first FEVER shared task recently are the Bi-Directional Attention Flow (BiDAF) network [50], Neural Semantic Matching Networks (NSMNs) [51] and the contextualised representations of a pre-trained BERT [52] as in [53]. In BiDAF [50], two vector sequences are produced from the embedding layer for both claim and evidence, and the attention scores are computed by the attention layer which sends them to the output layer where the semantic similarity between the original sequences and the new vectors is computed. Finally, the label is yielded by the output layer. In NSMNs [51], the alignment layer is applied for the encoded claim and evidence sequences then semantic matching is performed by an LSTM matching layer, where the output is sent as input to the output layer to produce a label . In BERT, 12 encoder layers with self-attention with a classification layer are applied to get a highly embedded representation of the claim and the evidence; this representation is received by the classification layer to output labels.

In [43], the post representation is obtained with pre-trained Longformer using sliding window-based self-attention [54]. In SemEval 2017 and SemEval 2019 rumour detection tasks, the models that are reported in [55] and [56] achieve competitive results. Multi-task approach for joint prediction of rumour stance and veracity using deep learning models such as BiLSTM applied in [43], outperforms earlier methods on both rumour stance classification and veracity prediction in SemEval 2019 Task 7 dataset. As a result, this work depends on the work in [43] to evaluate this model on SemEval 2019 Task 7 dataset.

Some studies apply stance detection and use the labels extracted from them as the input feature of the veracity prediction models to enhance the performance, which is shown important indicators to predict the veracity of rumours [35], [22], [55], [57]–[59], [36]. They combine the stance detection task with the rumour veracity classification task by using the idea of multi-task learning adopted in different ways such as feature learning in a parallel manner [57]–[59], [13] and hierarchically structural design [36]. In [58], Ma et al. use GRU layer for individual tasks, and the tasks also share a GRU layer to gain patterns common to both tasks. Similar to [58], joint learning is applied in [36] depending on a shared layer and task-specific layers, both models do not incorporate user information while Li et al.[13] utilise user credibility information in addition to the attention mechanism. Table 2 shows the performance comparison of different methods for rumour stance classification (single task) and veracity (multi-task) [36]. The macro-averaged F1 of Hierarchical graph convolutional network GCN-RNN [36], and Hierarchical- predicting rumour Stance and Veracity PSV [36] for stance and rumour detection respectively, are better than the baselines models they consider for evaluation [57], [60].

Table 2: Results of veracity prediction [35].

Setting	Method	SemEval dataset	
		Macro-*F*1	Accuracy
Single-task	top-down tree structure using a recursive neural network TD-RvNN [60]	0.509	0.536
	Hierarchical graph convolutional network GCN-RNN [36]	**0.540**	**0.536**
Multi-task	BranchLSTM+NileTMRG [57]	0.539	0.570
	MTL2 (Veracity+Stance) [57]	0.558	0.571
	Hierarchical- predicting rumour Stance and Veracity PSV [36]	**0.588**	**0.643**

The single-task setting means that stance labels cannot be used to train the models [36].

For the veracity prediction task in Rumour Eval 2019 competition [9], the best performing model is presented in [56], where different classifiers (Support Vector Model, Random Forest, Logistic Regression) with features obtained from LSTM attention network are used. For both tasks, the systems [14] with the best performances that shared in RumorEval 2019 competition are presented in Table 3 and 4.

Table 3: Test results for Task A Stance Detection

Rank	System	MacroF
	Khandelwal's [43] Method – Top N_s using (D + E + F)	0.6720
1	BLCU NLP	**0.6187**
2	BUT-FTT	0.6167
	Hierarchical graph convolutional network GCN-RNN [36]	0.540
3	EventAI	0.5776
4	UPV-28-UNITO	0.4895
5	HLT(HTTSZ)	0.4792

Table 4: Test results for Task B veracity prediction

Rank	System	MacroF
1	Li et al.'s model [13]	0.606
2	Khandelwal's [43] Method – Top N_s using (D + E + F)	0.5868
3	Hierarchical- predicting rumour Stance and Veracity PSV [36]	0.588
4	EventAI	0.5765
5	WeST (CLEARumor)	0.2856
6	GWU NLP LAB	0.2620
7	BLCU NLP	0.2525
8	Shaheyu	0.2284

2.3. Truth Discovery

Generally speaking, there are four categories of methods that have been applied in the previous research for truth discovery:

i. Iterative methods where the trustworthiness of sources and the confidence of claims from each other are computed iteratively and until convergence [61],
ii. Optimisation that measures the difference between the information provided by sources and the truth-based methods [62]
iii. Probabilistic graphical model-based methods where expectation maximisation is commonly used to infer the latent variables (parameters of truth and source reliability) [63]
iv. Neural network [64]

Researchers have built several methods to find true information from multiple sources of conflicting data started by TruthFinder [65] and Voting [66] that iteratively update-source reliability and true facts. Other works use other factors for truth discovery [67]–[71], e.g. information extraction such as entity profiling in [69] and knowledge graph in [71]. Recently truth discovery is formulated as an optimisation framework as in [72]–[74] where truths and source reliability are updated iteratively.

Other works depend on probabilistic approaches where source reliability is incorporated as a random variable into the probabilistic models and maximise likelihood or posterior distributions of multi-source data as the authors in [75] developed a maximum likelihood estimator for source reliability. In [76] the Probabilistic Soft Logic (PSL) framework is used to estimate source reliability and claim correctness while in [77], language objectivity analysis in addition to Subject-Predicate-Object (SPO) triplets are used to detect the veracity of value. Other people use other approaches for truth discovery; the authors in [78] split the sources and values to groups according to the user then analyse the credibility of the information. To infer the truth value, probabilistic graphical models can be used; probabilistic graphical models with three measures: silent, false spoken and true spoken rates are used in [79] and generative process for modelling used in [80]. Bayesian analysis can be used to decide dependence between sources [66]. Bayesian probabilistic modelling on the dependencies among source quality, truth, and claimed values [81] estimates the source reliability by considering the confidence interval of the estimation [72].

In the truth discovery task, neural network models show competitive accuracies [82]–[84][64]. Despite showing significant performance improvement in using stance information in their rumour detection model, they depend on hand-crafted user features like the number of followers, the number of posts to reflect user credibility, which is separated from stance labels for predicting the veracity of rumour [35][55][13]. The model in [55] performs well in stance classification system in RumourEval 2019. It is proved that lots of rumours come from either fake news websites or Hyperpartisan websites [85]. Liu and Wu [86] constructed user representations using network embedding and have shown that user

credibility information is particularly valuable for the veracity of rumour.

Unlike these models, the method (presented in section 3) learns representative features of stance detection using a different model architecture which learns to generate a conclusion of an article with respect to a particular target. Furthermore, this work's method jointly predicts stance and veracity and links bipolar argumentation with truth discovery methods. The vision is that a joint inference can improve the performance for these tasks: stance detection and veracity checking.

2.4. Datasets

For veracity checking, where evidence comes from trusted information sources, different datasets were developed to train the systems. Vlachos and Riedel [87] built a dataset with 221 statements and hyperlinks to the evidential source. Other datasets focus more on their information as features, such as metadata on the speaker, in LIAR dataset [88] with about 12k labelled claims. Recently, the most used dataset is FEVER [49], a large-scale dataset for fact extraction and verification with about 185k labelled claims. For contradictory claims, SemEval-2019 Task 7 dataset was developed by Gorrell, et al. [14] and Emergent dataset was developed by Ferreira and Vlachos [15] with 300 claims and 2,595 associated news articles.

This paper uses the data released at Rumour Eval-2019 for both stance detection and veracity prediction besides, and follows the evaluation metric as in [72]. It uses macro-averaged F1 to evaluate the performance on both tasks because it solves the imbalanced data problem. The statistical information for Rumour Eval-2019 datasets is described in table 5 and 6. It also evaluates the performance of the proposed framework using an additional dataset, Emergent corpus, since headline annotations draw attention to the article where Emergent is a dataset of rumours (claims) coupled with news headlines and their stances. Since this model focuses on generating conclusions for news articles, and summarising a long article into a conclusion, it uses extra information, like the headline in Emergent data that represents the news store, to increase the accuracy. The statistical information for the Emergent datasets is illustrated in table 7 and 8.

Table 5: Rumour Eval-2019 Task A corpus [14]

	Support	**Deny**	**Query**	**Comment**	**Total**
Twitter Train Reddit Train	1004 23	415 45	464 51	3685 1015	5568 1134
Total Train	1027	460	515	4700	6702
Twitter Test Reddit Test	141 16	92 54	62 31	771 705	1066 806
Total Test	157	146	93	1476	1872
Total Task A	1184	606	608	6176	8574

Table 6: Rumour Eval-2019 Task B corpus [14]

	True	**False**	**Unverified**	**Total**
Twitter Train Reddit Train	145 9	74 24	106 7	325 40
Total Train	154	98	113	365
Twitter Test Reddit Test	22 9	30 10	4 6	56 25
Total Test	31	40	10	81
Total Task B	185	138	123	446

Table 7: Emergent dataset [15]

Claims	300
Headlines	2,595
Minimum number of articles per claim	1
Maximum number of articles per claim	50
Training instances	2,071
Test instances	524

Table 8: Statistics of the Emergent dataset

Subject	**Stance**	**Emergent Number**	**Percentage**
Training	agree	992	24.37
	disagree	303	7.44
	discuss	776	19.06
	unrelated	2,000	49.13
		4,071	
Testing	Agree	246	24.02
	disagree	91	8.89
	discuss	776	19.06
	unrelated	500	48.83
		1,024	

3. The Proposed Argumentation-based Truth Discovery Model

This section proposes an Argumentation-based Truth Discovery Model, abbreviated as ATD. Since the length of an article source is longer than the user claim, and this is particularly the case for the Emergent dataset, the article may cover various aspects, but the user only concerns a specific aspect. The proposed model attempts to extract the right aspect of the document with respect to the user. To achieve this, this paper proposes a claim target-aware conclusion generation, which uses the main target of the user to help the generator to produce a better conclusion. After the main target is extracted, the most relevant clauses are derived from the evidence source. The selected clauses are sent to the sequence-to-sequence generator, which considers the target to guide the generation process, focusing more on the target. Several administrators are used to guiding the training of this model in an adversarial manner with training signals to optimise the model parameters, i.e. to find the difference between generated conclusion and ground truth conclusion. Finally, the generated conclusions are used to check the stance of the original evidence source toward this claim before veracity prediction in the end.

The closest work to this model's conclusion generator is abstractive text summarisation; most of them generate summaries by the decoder based on the encoded information from the encoder; some of them apply a copy mechanism to solve the out-of-vocabulary problem [89], [90]. Different earlier approaches are proposed in [91]–[94] to capture the main aspect and summarise based on the main aspect. Aspect aware summarisation by rewriting the most silent sentences is proposed in [94], which achieves the best performance on CNN/Daily Mail benchmark Dataset [95].

The architecture of ATD is shown in Figure 1, with the main components as follows:

- Claim Target Extraction component captures the claim focused aspect.
- Clause Selection component detects the most relevant clauses.
- Conclusion Generator component uses a seq2seq based architecture with attention and copy mechanisms.
- Stance Detection component to detect the position of the article toward a claim.
- Veracity Prediction component adopts Argumentation-Based Truth Discovery to decide the truth of the claim.

The design for each of the components is discussed in turn below. The example case in table 9 will be used to demonstrate the proposed ATD in action.

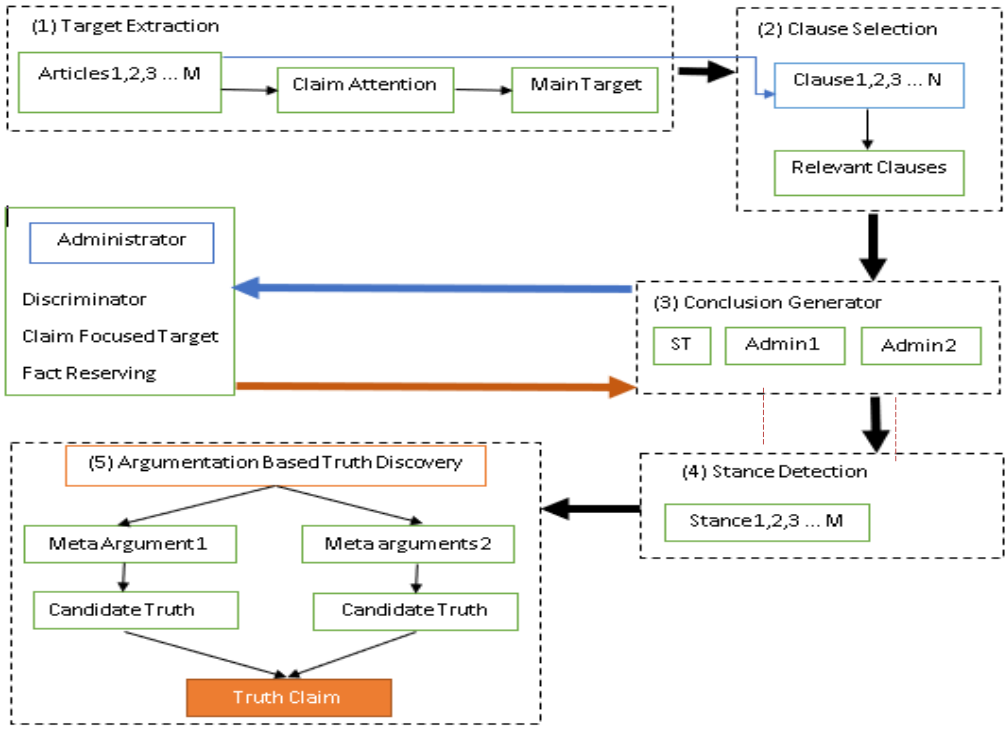

Fig. 1. The architecture of Argumentation-Based Truth Discovery Model

Table 9: An example of ATD from emergent data [15]

Source: law360.com
Claim: Two Australian men kept a McDonald's Quarter Pounder with Cheese for 20 years
Targets: Australia, Food, Hamburger, McDonald's, Quarter + Pounder
Claim Target Extraction: McDonald's
Clause Selection component detects the most relevant clauses: e.g. from article 4 as quoted and highlighted in bold below "Wonder **how long a Quarter Pounder with cheese can last**? **Two Australians say they bought a few McDonald's burgers for friends back in 1995** when they were teens, and one of the friends never showed up. So, the kid's burger went uneaten and stayed that way, Australia's News Network reports. "We're pretty sure it is the oldest burger in the world," says one of the men, Casey Dean. Holding onto the burger for their friend "started as a joke," he adds, but "the months became years and **now, 20 years later, it looks the same as it did the day we bought it**, perfectly preserved in its original wrapping." Dean and his burger-buying mate, Eduard Nitz, even took the burger on the Australian TV show The Project last night and "showed off the mould-free specimen" News 9 reports. The pair offered to take a bite of it for charity but were dissuaded by the show's hosts. They have also started a Facebook page for the burger called "**Can This 20-Year-Old Burger Get More Likes Than Kanye West**?" that has more than 4,044 likes as of this writing. Furthermore, they are selling an iTunes song, "Free the Burger," for $1.69, and giving proceeds to the charity Beyond Blue, which helps Australians battle anxiety and depression. (A few years ago, a **man sold a 20-year-old bottle of McDonald's McJordan sauce for $10,000**. Here's why Mickey D's food seemingly, never decays.)."
Conclusion Generator: For 20 years, two Australian men held a McDonald's Quarter Pounder with Cheese
Stance Detection
Source: 9news.com.au
Headline: Two blokes dared to eat a 20-year-old burger for charity
Stance: for
Source: mirror.co.uk
Headline: Is this the world's oldest burger? Man claims to have kept McDonald's Quarter Pounder for 20 YEARS
Stance: for
Source: examiner.com
Headline: 20-year-old burger: McDonald's Quarter Pounder looks nearly new after 2 decades
Stance: observing
Source: techinsider.net
Headline: 20-Year-Old Quarter Pounder Looks About the Same
Stance: observing
Veracity Prediction component adopts Argumentation-Based Truth Discovery: | **Veracity**: true

3.1. Claim Target Extraction

The purpose of this component is to extract the primary target that is common between a claim and its relevant evidence among candidates' targets, as shown in figure 2. First, extracting the nouns from the claim and the article. For each noun, they need to be represented as a vector with a probability distribution. Then applying a Jensen-Shannon Divergence and the ones with a distance score greater than threshold ϵ are labelled as candidate aspects. A good conclusion should have the main target of the claim. Jensen-Shannon Divergence (JSP) is a way of measuring the matching between two distributions and indicates the distances between two probability distributions, e.g. p, q vectors as in Equation 1. The distance score is used as a relevance score. Jensen-Shannon Distance is the symmetric version of the Kullback-Leibler Divergence (DKL), using difference measure for probability distributions [96]:

$$1/2(\mathrm{D}(\mathrm{p}\|\mathrm{m}) + \mathrm{D}(\mathrm{q}\|\mathrm{m})) \quad (1)$$

Where m = 1/2 (p + q)
An example of two distributions:
p = as array ([0.10, 0.40, 0.50])
q = as array([0.80, 0.15, 0.05])
Jensen-Shannon divergence (P || Q): 0.42
Jensen-Shannon distance (P || Q): 0.648, distance is sqrt of divergence.

To find the more significant distance between the pair of nouns, each claim extracted from a noun paired with a noun extracted from an evidence article (main target). After the top five candidate aspects are selected using the Jensen-Shannon Divergence, e.g. {Australia, Food, Hamburger, McDonald's, Quarter and Pounder}, this model re-rank them based on the maximum alignment score of two noun embeddings of claim- article. A max operation over the alignment is used to select the highly focused noun in the article by the claim as in equation 2, 3, respectively, close to the work in [97]. This work computes the semantic word alignment of a claim using its embeddings towards the article to model the claim concentrated aspect. So that the alignment score gives us an indication of the attention of a word in a claim toward an article where $e(A_i^s)^T$ is word embedding in the article, and $e\left(A_{j,n}^c\right)$ is word embedding in the claim, $ASPECT_{i,j,n}$ is the attention wei for the i-th claim word with the j-th article word, s is article, c is claim, n is article number, i is index word of article, and j is the index word in the claim.

$$\mathrm{ASPECT}_{\mathrm{i,j,n}} = \mathrm{e}(\mathrm{A}_{\mathrm{i}}^{\mathrm{c}})^{\mathrm{T}}\, \mathrm{e}\left(\mathrm{A}_{\mathrm{j,n}}^{\mathrm{s}}\right), \quad (2)$$

$$maximum_{i,j} = max\left(\left\{ASPECT_{i,j,1}, \dots, ASPECT_{i,j,T_j^c}\right\}\right), \quad (3)$$

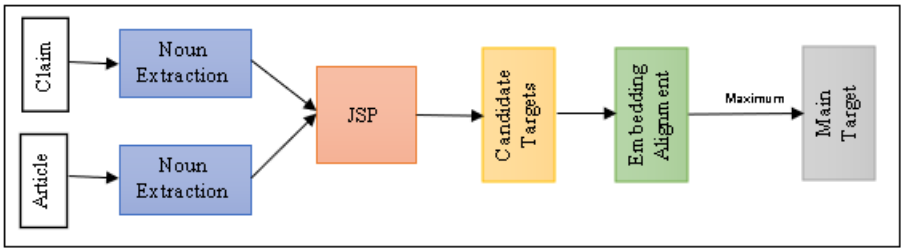

Fig. 2. The General Architecture of Claim Target Extraction

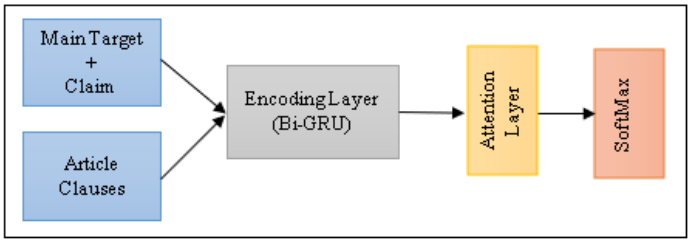

Fig. 3. The Architecture of Clause Selection Model

3.2. Clause Selection Model

The purpose of this component is to retrieve the most target- relevant clauses and ignore the irrelevant clauses. This component can be considered as two layers Encoding Layer and *Attention layer*, as shown in figure 3.

3.2.1. Encoding Layer

For each evidence article relevant to the claim, this work utilises clause encoding layer that applies Bi-directional GRU to capture the context clause representations. The contextual information of claim c concatenated with target and each clause cl in the article is obtained by BI-GRU. This model uses GRU as it is more efficient than LSTM in training to learn the hidden semantics of words. It applies two GRU neural networks: a forward GRU and a backward-GRU, which processes the sentence from left to right and the reverse order respectively by handling the word vectors in order. Finally, the forward-GRU and backward-GRU units are concatenated to learn the bidirectional semantics of claim and each clause in the article to emphasise the importance of claim, and then utilise attention mechanism to capture the valuable information in article clauses. he states from the forward-GRU, and backward-GRU are denoted by $\vec{h}_i$ and $h^{\leftarrow}{}_i$, for claim c and clause cl respectively and the final hidden state $h_i$is the concatenation of the states. This is done by the encoding layer for both clauses and claim, i.e equations 4-6 for a claim and 7-9 for clauses:

$$\vec{h}_i = \overrightarrow{GRU}_{(c_i)}; \quad i \in [1, N] \tag{4}$$

$$h^{\leftarrow}{}_i = GRU^{\leftarrow}{}_{(c_i)}; \quad i \in [N, 1] \tag{5}$$

$$h_i = \vec{h}_i \oplus h^{\leftarrow}{}_i \tag{6}$$

$$\vec{h}_j = \overrightarrow{GRU}_{(cl_j)}; \quad j \in [1, Cl] \tag{7}$$

$$h^{\leftarrow}{}_j = GRU^{\leftarrow}{}_{(cl_j)}; \quad j \in [Cl, 1] \tag{8}$$

$$h_j = \vec{h}_j \oplus h^{\leftarrow}{}_j \tag{9}$$

3.2.2. Attention Layer

Attention layer focuses on the terms that are important to the meaning of the clause concerning the claim, producing clause vectors. Attention mechanisms will be implemented to concentrate on those words in the evidence clause with respect to a specific claim c concatenate with the target and combine the representation of all of them to form a clause vector of evidence. Cl is the clause representation with respect to the claim and target. Attention mechanism computes the attention weight between each claim-clause representation to produce contextual information conditioned on the claim representation. The attention weight between each clause and the representation of a specific claim will be computed as follows:

3.2.3. Attention equations

$$a_{vi} = avg\left(\left\{h^c_{i,1}, h^c_{i,2}, \ldots, h^c_{i,T^c_i}\right\}\right), \tag{10}$$

$$m_i = tanh\left(W_{cl} \cdot \left[a_{vi}; h^{cl}_{j,T^{cl}_j}\right] + b_{cl}\right) \tag{11}$$

$$a_i = softmax(m_i) = \frac{exp(m_i)}{\sum_{t=1}^{Cl} exp(m_t)} \quad (12)$$

$$clr = \sum_{i=1}^{|cl|} a_i \,.\, h_j^{cl} \cdot \quad (13)$$

$h_{j,T_j^c}^{cl} l$ is the final state of the clause, cl is clause and c is a claim, a_{vi} is the average of hidden states for claim and target, a_i is attention weights and the clause representation clr is calculated based on the attention vectors a_i. In this model, to select relevant clauses for claim with the target, conditional probability using SoftMax Layer is used to perform target clause relevant classification. Then, feeding the clause representation clr to a SoftMax classifier. This model trained by cross-entropy, W and b are the parameters for the model. W is weight matrix, and b is the bias.

$$o = W^* clr + b \quad (14)$$

3.3. *Conclusion Generator*

This component is for conclusion generation that conveys a particular stance towards some target as key to understanding an argument from its premises. The general overview of this component is shown in figure 4. This work uses pointer generator architecture with attention and copy mechanisms for claim-target-Biased generator, i.e. Pointer generator as decoder considering the claim vector concatenated with the target. The input for the model is the most relevant clauses selected, and a claim with its main target passed to the article encoder and a claim encoder with its target respectively. The representation outputs of each encoder are passed to the generator (decoder). Both encoders, as well as the decoder, use Recurrent Neural network, i.e. Bi-GRU for encoders and GRU for the decoder.

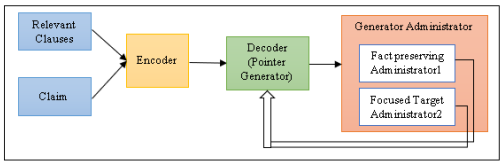

Fig. 4. Overview of Conclusion generator

3.3.1. *Article (Relevant Clauses) and Claim Encoder*

A bidirectional GRU is used to get both the context before and after the word as follows:

$$\vec{h}_i = GRU_{\overrightarrow{evi}}\left(\vec{h}_{i-1}, E_{(w_i)}\right) \quad (15)$$

$$h^{\leftarrow}{}_i = GRU_{evi^{\leftarrow}}\left(h^{\leftarrow}{}_{i-1}, E_{(w^{\leftarrow}{}_i)}\right) \quad (16)$$

$$h_i = \left[\vec{h}_i, h^{\leftarrow}{}_i\right] \quad (17)$$

$$\vec{h}_j = GRU_{\vec{c}}\left(\vec{h}_{j-1}, E_{(w_j)}\right) \quad (18)$$

$$h^{\leftarrow}{}_j = GRU_{c^{\leftarrow}}\left(h^{\leftarrow}{}_{j-1}, E_{(w^{\leftarrow}{}_j)}\right) \quad (19)$$

$$h_j = \left[\vec{h}_j, h^{\leftarrow}{}_j\right] \quad (20)$$

Where E(wi) is the word embedding of the word in both directions; Word Embeddings method implemented in this work is GoVe [27]. The encoder generates a state for each input word by BiGRU for claim and evidence to obtain the context representation around a word. The evidence includes all relative clauses retrieved by clause selection model. $GRU_{\overrightarrow{evi}}$ *and* $GRU_{evi^{\leftarrow}}$ are the forward and backwards representation, h_i denotes hidden state. $\left[\vec{h}_i, h^{\leftarrow}{}_i\right]$ is the merge of the forward and backward hidden state, *evi* is the evidence and *c* is the claim. h_i and h_j are the annotations for evidence and claim. Equations from 15 to 17 compute a hidden representation for evidence from both directions and the same for equations from 18 to 20 for claim encoding.

3.3.2. *Decoder*

The decoder uses unidirectional GRU, where claim concatenated with its target representation is used as input at each decoder time step. Depending on the final state of input encoder, and target representation, the decoder starts the decoding process to generate the conclusion from the evidence. The target embedding is fed as input at each decoder time step to provide the decoder with the ability to alter the output sequence to generate a statement about the main target finally.

The distribution of words is computed, and a SoftMax function picks the word with the highest probability on the output of the decoder state and context vector. A sigmoid activation function is used at each decode time step to decide from two options: copy from the original input or generate from the vocabulary, so is the final article encoder state, Ct is the context vector at time step t from the attention mechanism, and Yt-1 is the predicted output word at time step -1. The attention mechanism is used to identify the relevant parts of the input by learning the decoder to focus on different portions of the claim and target at different time steps [98]. This could be done by applying the following equations, inspired by [99], with modification to suit the proposed model. Attention mechanism for the evidence is applied to help the decoder to output focused claim tokens at each time step using equations 21,22 whereα_{ti} represents weights to each in the claim at each decoder timestep, St is the current state of the decoder at time step t. The final claim

representation at time step t is computed in equation 23:

Attention mechanism for the claim which assigns weights to each word in the claim at each decoder timestep, h_j^c is claim word hidden states, using the following equations

$$a_{t,j}^c = v_{cl} \cdot tanh\left(W_{cl} s_t + U_{cl} h_j^c\right) \quad (21)$$

$$\alpha_{tj}^c = \frac{exp\left(a_{t,j}^c\right)}{\sum_{j=1}^{|cl|} exp\left(a_{t,j}^c\right)} \quad (22)$$

The final claim representation at time step *t*, which is computed as

$$cc_t = \sum_{i=1}^{|c|} \alpha_{t,i}^c \, h_j^c \cdot \quad (23)$$

The final evidence representation at time step *t*, the context vector for the decoder c_t, is computed using the following equations

$$c_t = \sum_i \alpha_{ti} \, h_i \quad (24)$$

the evidence attention model which assigns weights to each word in the evidence article considering the final claim representation cct as follows:

$$\alpha_{ti} = \frac{exp(e_{ti})}{\sum_k exp(e_{tk})} \quad (25)$$

$$e_{ti} = tanh\left(h_i, s_{t-1}, E_{(Y_{t-1})}, cct\right) \quad (26)$$

$h_i$hidden representation for each time-step for evidence word I, $E_{(Y_{t-1})}$ is the previous word embeddings, st is the current state of the decoder at time step t, and cct is final claim representation at a time step. The hidden state of the decoder st at each time t is computed as follow, considering the previous state s_{t-1}, the embedding distribution of the claimed target $target$, the previous evidence context vector c_{t-1} and the previous word embeddings:

$$s_t = GRU_{dec}\left(s_{t-1}, \left[c_{t-1}, target, E_{(Y_{t-1})}\right]\right) \quad (27)$$

The probability distribution over the output vocabulary o_t to decide the word which has the highest probability is computed from the context vector ct, the decoder state st as follow:

$$o_t = W_g^{(2)}\left(W_g^{(1)}[s_t, c_t] + b_g^{(1)}\right) + b_g^{(2)}, \quad (28)$$

Inspired by Hasselqvist et al. [100], the pointer mechanism is applied in the decoder of this work to take a decision to either copy from the original document or generate from the vocabulary based on the pointer output; the next word can then be selected. $p_t^{pointer}$ is used as a switch to select between (a) copying words from the source text via pointing (copying a word from the input sequence by selection according to the attention distribution) or (b) generating a word from the vocabulary by selecting based on Pv in Equ. 31.

$$p_t^{pointer} = sigmoid\left(v_{ptr}^T\left[s_t, E_{(Y_{t-1})}, c_t\right] + b_{pointer}\right) \quad (29)$$

the generation probability$p_{tj}^{gen} \in$ [0,1] for timestep t is computed as equation (30). If p_{tj}^{gen} > 0.5, word is copied from the input that is determined by the attention distribution where the attention is the highest, else the generator output is used.

$$p_{tj}^{gen} = \frac{exp(o_{tj})}{\sum_k exp(o_{tk})} \quad (30)$$

The model then generates distribution Pv over vocabulary. It concatenates on administrator 1, and 2, (details below) and the output of the decoder to guide the decoder. P_v *is* probability distribution over all words in the vocabulary and gives us the final distribution from which to expect words. It concatenates administrators $adm1_t; adm2_{t;}$ and the output of decoder st as the input of the output projection layer. The goal $adm1_t$ *is* to monitor the gap between the generated conclusion and the focused claim, and it will demonstrate the details of these variables in equation 31 in the next subsection.

$$P_v = softmax\left(W_v\left[s_t; E_{(Y_{t-1})}, c_t; adm1_t; adm2_{t;}\right] + b_v\right) \quad (31)$$

For the evaluation metrics, this model uses the following ROUGE scores to evaluate the quality of the generated conclusion [101]. ROUGE stands for Recall-Oriented Understudy for Gisting Evaluation used to determine the quality of a summary including ROUGE-1, ROUGE-2 and ROUGE-L, which demonstrate the effectiveness of discriminator.

- ROUGE-1: the unigram overlaps describes the overlap of each word between the candidate and reference summaries.
- ROUGE-2: bigram-overlap between the reference summary and the summary to be assessed.
- ROUGE-L: the longest common subsequence between the reference summary and the summary to be assessed.

3.3.3. Generators' Administrators

In order to reduce the noisy and irrelevant generated words by the decoder, in addition, to guide the generator to focus on the information related to the target claim and preserve the fact information, this work employs two administrators allowing the discriminator to provide additional information, denoted as

administrator 1 and 2. The discriminator is a binary classifier using a convolutional feature extractor and a sigmoid classification layer as a signal to the generator. The details of the two administrators are explained in turn below.

Administrator 1 to check decoder focused target

Administrator 1 measures the semantic divergence between decoder focused target and claim target. There is also a discriminator which uses a convolutional neural network to extract features and then distinguishes how similar is decoder focused aspect to user-focused aspect. For example, the claimed target is" McDonald's Quarter", but the generated conclusion may focus on the "charity" aspect. This model checks if the generated argument focuses on claim attention in the past decoding steps.

In order to model the difference between the target attention and conclusion attention and then identify the unfocused target by the decoder, this model applies element-wise difference. Then, in order to help the decoder to generate a conclusion focusing more on the target, it uses the attention distribution difference to the weighted sum of the document states as the context vector.

For encouraging the generator to focus on the claimed target, this model employs a CNN based discriminator to signify the difference between the generator focused target, and the claim focused target. Sentence encoding module word-level BiGRU is applied to produce the sentence vector after concatenating the aspect. To further study the interactions and information exchanges between sentences, the model establishes a Bi-directional GRU (Bi-GRU) network taking the sentence representation as input. This architecture allows information to flow back and forth to generate new sentence representation. Attention-based CNN model for this administrator will be used. The final target representation of the conclusion is fed into an output layer to predict the probability distribution on the target is defined as Adm1 via equations 32-36. It is trained via cross-entropy minimisation for training aspect-based conclusion generation.

$$target = \frac{1}{m}\sum_{i=1}^{m} e_{x_i} \quad (32)$$

$$score_i = tanh(h_i^T W_1 target) \quad (33)$$

$$attention_i = \frac{exp(score_i)}{\sum_{j=1}^{n} exp(score_j)} \quad (34)$$

$$S = \sum_{i=1}^{n} attention_i \, h_i \quad (35)$$

$$\text{Adm1} = \text{SoftMax}\,(W.\,S + b) \quad (36)$$

Where the target is the target representation computed as the averaged word embedding of the target as in equation 32, the score is a content-based function that captures the semantic association between the target of decoder output and the target of claim in equation 33. Equation 34 assigns weights to each word in the conclusion. S is used to encode the relevant information n from the conclusion as well as the claim in equation 35. Adm1 maximises the probability of generating a conclusion toward a target. W is the weight matrix, and b is the bias.

Administrator 2 to check the fact preserving

This model applies a denoising autoencoder to evaluate the fact preserving related to the target of the conclusion with respect to the evidence source, i.e. integrate knowledge from the source article; this is represented as a factual score. At each decoding time step t, GRU reads the previous output y_{t-1} and context vector c_{t-1} as inputs to compute the new hidden state s_t. After extracting the fact related to the target in the source article, it applies BiGRU to extract hidden state for both facts in the article and the generated fact. Then the context vectors are computed. For the fact vector equations from 37 to 39 are applied and from 40 to 42 for a generated fact. Besides the current state of the decoder, a combination of both context vectors is used to guide the decoder to generate more factual words. Adm2 represents the gap content between the factual and non-factual generated conclusion. The gap guides the generator to preserve the fact. The probability of generating the next word is based on the SoftMax layer result.

Attention mechanism for the facts in the article

$$e_{t,i}^{fact} = \text{MLP}\left(s_t, h_i^{fact}\right) \quad (37)$$

$$a_{t,i}^{fact} = \frac{\exp\left(e_{t,i}^{fact}\right)}{\sum_j \exp\left(e_{t,j}^{fact}\right)} \quad (38)$$

$$c_t^{fact} = \sum_i a_{t,i}^{fact} h_i^{fact} \quad (39)$$

Attention mechanism for the generated conclusion

$$e_{t,i}^{gen} = \text{MLP}\left(s_t, h_i^{gen}\right) \quad (40)$$

$$a_{t,i}^{gen} = \frac{\exp\left(e_{t,i}^{gen}\right)}{\sum_j \exp\left(e_{t,j}^{gen}\right)} \quad (41)$$

$$c_t^{gen} = \sum_i a_{t,i}^{gen} h_i^{gen} \quad (42)$$

The context vectors are merged:

$$c_t = \left[c_t^{fact}; c_t^{gen}\right] \quad (43)$$

$$s_t = \text{GRU}(Y_{t-1}, c_t, s_{t-1}) \quad (44)$$

$$m_i = \tanh\left(W_1 \cdot \left[[s_t, c_t]\right] + b_1\right) \quad (45)$$

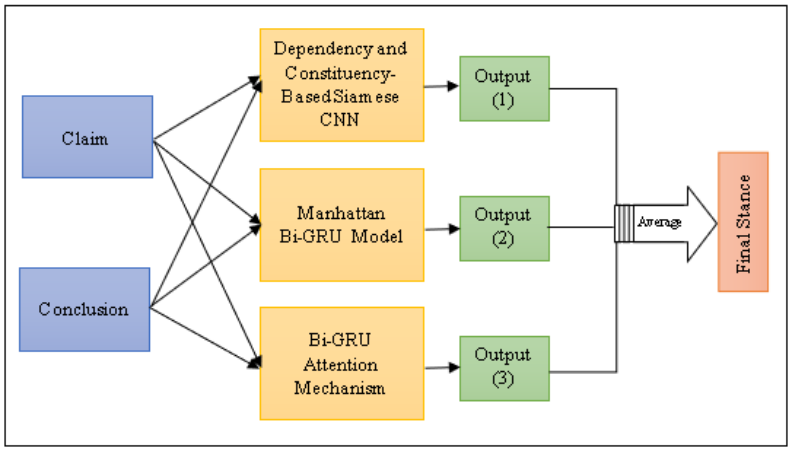

Fig. 5. The proposed Stance Detection Model

$$\text{output}_i = \text{softmax}(m_i) = \frac{\exp(m_i)}{\sum_{t=1}^{C} \exp(m_t)} \quad (46)$$

$$f_t = c_t^{fact} - c_t^{gen},\ \text{adm}_2 = \sum_{i=1}^{T^d} f_t, h_i^d \quad (47)$$

3.4. Stance Detection

The stance of the generated conclusion argument toward a claim is detected in current model by applying three methods, as in figure 5, method 1 for stance detection using dependency and constituency-based Siamese CNN, Method 2 for stance detection using Manhattan distance and Method 3 for stance detection using an attention mechanism.

3.4.1. Method 1 for stance detection: Dependency and constituency based Siamese CNN

This work extracts sentence-level features *c* from the input claim and conclusion. It proposes performing constituency and dependency parse on the inputs using the Stanford Parser, to obtain the most important sentence information such as the main linguistic structure which provide information; like (the subject, predicate, and object of a sentence which are the most important roles in a sentence). It is necessary to learn better sentence representations to observe the structure of sentences and the relation among the words for each of them.

For both, claim and conclusion, word sequence based on constituency words of their text are concatenated with their word sequence based on dependency parser as an input of this work's CNN-based model, where each of them is fed to convolution operations separately. For words representing constituency or dependency that occur in the sentence, the convolution operation applies a filter *w* and bias as in equation 48 with sigmoid function, all of them are concatenated to represent the entire feature map for all the words in the sentence. To exploit more powerful information and capture different features, this model combines information for both: constituency and dependency-based CNN models by concatenating these representations, then generate the statement (claim or conclusion) representation which is the concatenation of the all vectors. The generated conclusion representation is transformed from word embedding vectors to the semantic sentence hidden states using a convolutional neural network. Then pooling operation is used to reduce the spatial size of the representation and retain important features. Both aspect and conclusion representation feature vectors are connected into one vector, then the attention vector is generated. Matching distance is used to get the degrees of the similarity between a claim and a conclusion [102].

Constituency based CNN for claim X_c

For a given claim, the convolution operation applies a filter W for each constituency based concatenated word sequence X_c (or words with is constituent structures words from the constituency sub-tree) from the claim where b_c is the bias as follow:

$$Y_{ci} = sigmoid(W_c X_c + b_c) \quad (48)$$

All words constituency information is concatenated to generate the feature map const:

$$const = Y_{c1}, Y_{c2}, Y_{c3}\ Y_{cl} \quad (49)$$

Dependency-based CNN for Claim X_{dep}

For a given claim, the convolution operation applies a filter W for each dependency-based concatenated word sequence X_{dep} (or words with is dependent structures words from the dependency sub-tree) from the claim where b_c is the bias as follow:

$$Y_{depi} = \text{sigmoid}(W_c X_{dep} + b_c) \quad (50)$$

All words dependency information is concatenated to generate the feature map dep:

$$\text{dep} = \text{Y}_{\text{dep1}}, \text{Y}_{\text{dep2}}\text{Y}_{\text{dep3}}\ \ \text{Y}_{\text{depl}} \quad (51)$$

Dependency-based CNN and constituency-based CNN concatenation for claim

the max pooling operation is applied for both feature maps, const. and dep., to extract the most significant features from each of them, then they are concatenated as follow:

$$max1 = max\ (dep) + max(const) \quad (52)$$

The same equations that are used for claim feature representation, 48 to 52 will be used for the conclusion, equations from 53 to 57

Constituency-based CNN for the conclusion X_e

$$\text{Y}_{\text{ei}} = \text{sigmoid}(\text{W}_\text{e}\text{X}_\text{e} + \text{b}_\text{e)} \quad (53)$$

$$\text{e= Y}_{\text{e1}}, \text{Y}_{\text{e2}}\text{Y}_{\text{e3}}\ \ \text{Y}_{\text{en}} \quad (54)$$

Dependency-based CNN for the conclusion X_{dep2}

$$\text{Y}_{\text{depi}} = \text{sigmoid}(\text{W}_\text{e}\text{X}_{\text{depi}} + \text{b}_{\text{ce}}) \quad (55)$$

$$\text{dep2} = \text{Y}_{\text{dep1}}, \text{Y}_{\text{dep2}}\text{Y}_{\text{dep3}}\ \ \text{Y}_{\text{depl}} \quad (56)$$

Dependency-based CNN and constituency-based CNN concatenation for the conclusion

$$max2 = max\ (dep2) + max(const) \quad (57)$$

After generating the vector representation of sentences, 3 matching methods are applied to extract relations between ($max1; max2$)

1. Concatenation of individual representation ($max1; max2$) to produce r1
2. Element-wise product ($max1 * max2$) to produce r2
3. Absolute element-wise difference ($max1 - max2$) to produce r3

All the resulting vectors r1,r2, and r3 are concatenated and fed to a SoftMax classifier to predicts the stance label between Claim and conclusion as follow:

$$f = r1 \oplus r2 \oplus r3 \quad (58)$$

f is a fully connected neural network

$$output1 = softmax(W_1 f + b_1) \quad (59)$$

3.4.2. Method 2 for stance detection: Manhattan- Bi-GRU Model

Bi-GRU is applied to extract the representation of the final hidden state, which as a vector representation for each claim and conclusion and then use them to compute the semantic similarity between them. The semantic similarity is computed by Manhattan Bi-GRU Model [102] where the distance is transformed into a similarity score to measure the strength of the conclusion toward the claim. $h^{(c)}$ and $h^{(e)}$ are the last hidden representations for the claim and conclusion respectively.

$$output\ 2 = exp\left(-\|h^{(c)} - h^{(e)}\|_1\right) \quad (60)$$

3.4.3. Method 3 for stance detection: Bi-GRU Attention mechanism

To detect the stance of the conclusion toward the main target, attention mechanisms are used to capture the most relevant features. After extracting the hidden states for both target and conclusion by Bi-GRU, this model merges them as one vector. The word attention weights are computed using equations 62-67 where $h^i_{conc}\ and\ h^i_{claim}$ are the average of hidden states from Bi-GRU for the conclusion and claim respectively, $\alpha_i\ and\ \beta_i$ are attention vectors for both claim and conclusion that used to compute word attention weights. Then the text representation considers the common features between them as in equation 68:

$$att(h^i_{conc}) = tanh(h^i_{conc} \cdot W_1 + b_1) \quad (62)$$

$$\alpha_i = \frac{exp\left(att(h^i_{conc})\right)}{\sum_{j=1}^{n+1} exp\left(att(h^i_{conc})\right)} \quad (63)$$

To generate the final representation for the conclusion representation with the main target, the following equation is applied.

$$conclusion_r = \sum_{i=1}^{n+1} \alpha_i\, h^i_{conc}. \quad (64)$$

For the claim concatenated with the target, the attention vector is calculated by the following equations:

$$att\left(h^i_{claim}, h^p_{conc}\right) = tanh\left(h^i_{claim} \cdot h^p_{conc} \cdot W_2 + b_2\right) \quad (65)$$

$$\beta_i = \frac{exp\left(att\left(h^i_{claim}, h^p_{conc}\right)\right)}{\sum_{j=1}^{m} exp\left(att\left(h^i_{claim}, h^p_{conc}\right)\right)} \quad (66)$$

$$claim_r = \sum_{i=1}^{m} \beta_i h^i_{claim}. \quad (67)$$

$$output3 = softmax([claim_r \oplus conclusion_r]) \quad (68)$$

The final stance of each article is based on the average of output1, output2 and outpu3.

3.5. Argumentation-based Truth Discovery

Preliminary steps towards truth discovery methods based on bipolar argumentation are

presented in [22], where a truth discovery network which assigns each source a trust score and each fact a belief score is mapped to a bipolar argumentation framework. Cayrol & Lagasquie-Schiex [103] also suggest linking Truth Discovery with Bipolar Abstract Argumentation. They consider a truth discovery network as disjoint sets S, O and F, which represent sources, objects and facts respectively. They suggest assigning each source a trust score and each fact a belief score to represent the trust and belief rankings in sources and facts, respectively. For argumentation frameworks, they consider that arguments interact through attacks and support relations. They provide an example as in figure 6 to illustrate graph representation of a truth discovery network where sources are s, t, u, v, objects are o, p and facts are f, g, h, i. For the facts related to objects o and p, the sources s and t have contradiction views toward while the sources u and v agree source s particularly t on object p. They propose a truth discovery operator that assigns each source a trust score and each fact a belief score. Argumentation-based Truth Discovery is inspired by abstract argumentation by identifying such arguments with the sources and facts. They propose to encode source trustworthiness by introducing an argument, e.g. "s is a trustworthy source", and introduce an argument "f is a believable fact" for identifying fact believability. According to the example, B(N) yields two meta-arguments where each argument attacks the other: X1,X2, where X1 = {s,f,h} , X2={ t,u,v,g,i}

This work applies an argumentation-based truth discovery, where different arguments support contrary conclusions for certain information from multiple sources with different degrees of trustworthiness.

For two meta arguments X1, X2, where each argument attacks the other, including the sources with their supported claims, estimation source reliability weight is applied to compute the strength of supporting compared to the strength of attacking. First, each meta-argument, including sources with its supported claim, is expressed, e.g.:

X1{S1, S4, S6, S7, c1, c2, c3, c4, c5}
X2{S2, S3, S5, S8, c6, c7, c8, c9}

1. Siamese adaptation of the Long Short-Term Memory (LSTM): Manhattan LSTM Model [102] is applied for each pair of the claim and its supporting source word embeddings in the same meta-argument. This model uses an LSTM where the final hidden state is a vector representation for each claim and source. Then, the semantic similarity between them is computed. Subsequently, all outputs of this model are averaged for all claim-source pairs.
2. The dependency between the data sources: the highest correlated source with other sources is computed in the same way as the computation for the claim-source pair. The difference between all sources and its supported sources vector with all other vectors from other sources, e.g. the Manhattan distance, then: average, compare, then rank them (reliable to unreliable).
3. For each claim of each meta-argument, this model computes the probability of supporting the claim by its sources:

$$p(s_i|c_i, u_i) = \frac{exp(c_i, u_i)}{\sum_i exp(c_i, u_i)} \quad (69)$$

$p(s_i)$ is the probability of supporting a claim c based a source u, i.e. to which extent the claim is supported by it associated sources, u_i is the source vector representation and c_i is the claim vector representation. If the probability is >= 0.5, the claim is selected as a candidate truth.

4. For each source in each meta-argument, this model computes the probability of correlating with other sources, u and v are vectors of different sources.

$$p(s_i|v_i, u_i) = \frac{exp(v_i, u_i)}{\sum_i exp(v_i, u_i)} \quad (70)$$

- If the probability is >=0.5, then the source is selected as a candidate trustworthy source.
- The sources with more correlation and dependency with other sources are picked as a trustworthy source.
- For each target, the candidate's truth claims, and reliable sources are put in a set for both attacking arguments e.g.

X1{S1, S6, S7, c1, c2}
X2{S2, S8, c6, c7, c8, c9}

- Truth claim of a certain target: for each meta-argument, claims embeddings are averaged as a ground truth claim, the claim with the highest similarity to the average is considered as truth claim from this argument with its supporting sources.
- For each meta-argument, source embeddings are averaged as ground truth source, the source with the highest similarity to the average is considered as truth source from this argument with its supporting claims
- Finally, vocabulary richness features and readability features are applied, as shown in [104] to decide the most trustworthy source and truth claim from both meta-arguments. All features results are weighted to decide the final veracity label.

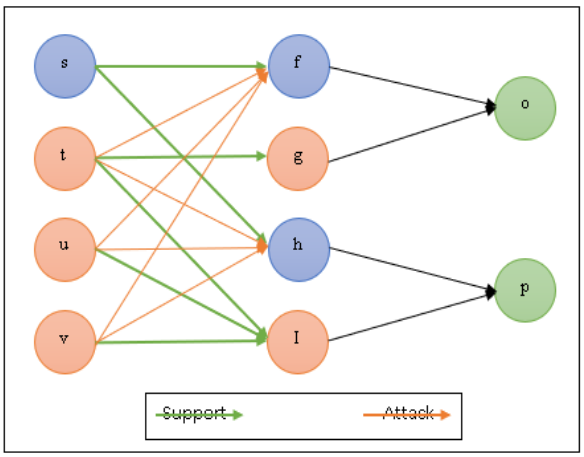

Fig. 6. Graph representation of a truth discovery network [71]**, s and t disagree on the true fact for objects o and p. Sources u and v do not comment on object o but agree with t on object p.**

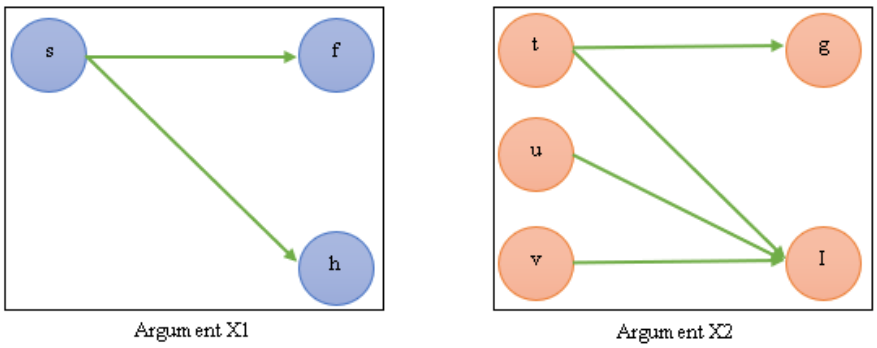

Fig. 7. Argumentation-Based truth Discovery

4. Experiments and Results

4.1. Datasets

These models are evaluated using macro-averaged F1 as the evaluation metric for the two tasks in RumorEval 2019 [14] as discussed in section 2.5 above. It includes 325 rumorous conversation threads, and the RumorEval dataset was provided with a training/development/testing split. Additional Experiments are conducted on Emergent publicly available datasets [15] as discussed in section 2.5 above, since news article headline, and evidence of news article content are essential information for the this model training. Since the Emergent dataset is the largest, collected from a fact-checking website, more balanced and annotated to help the model to train, therefore, this data will be used in the experiment. The claims are paired with news headlines and their stances and the public veracity of the claim.

4.2. Experimental Setting

In experiments, the models in this paper are implemented using Keras. All word vectors are initialised by word2vec [105]. The word embedding dimension is 300, the size of units in GRU is 100. The batch size is set 32 and the hyperparameters are chosen, which achieves the largest value on the validation set then used to train the model on the entire dataset. The learning rate is set at 0.001. The rule activation function used in the hidden layers is set to evaluate the performance on both Task A and B. The used evaluation metric: “macro-average F1 score” since the class labels are imbalanced.

4.3. Baselines: Compared Methods

In this section, the performance of stance detection and rumour verification, for the proposed model against the state-of-the-art models, has been evaluated and discussed in section 2. These models are described in detail as follows:

Zhang et al.'s Hierarchical Representation and Detection model [25] on Emergent dataset [15]:

This model is compared against the state-of-the-art model reported in [25] on the augmented Emergent dataset. Experiments are performed on the publicly available Emergent dataset [15] which consists of news article headlines, evidence of news article content and stances. Stances can be classified as a support, oppose, and discuss, which can be used to infer claim veracity.

This model compares their performance with four-way classification baselines (OB, MLP, BiLSTM, AtCNN, MN and RM) and demonstrates state-of-the-art accuracy performance. They develop a two-layer neural network that learns from this hierarchical representation to alleviate the class imbalance problem; the first layer for relatedness stance detection and the second layer is for the agree, disagree and discuss stances. They study the various levels of dependence assumptions between the two layers: controls the error propagation between the two layers using the Maximum Mean Discrepancy MMD regularizer. They demonstrate that their work outperforms the state-of-the-art accuracy for the stance detection task, (i.e. the mentioned stance detection models in the related work section.

When Experiments are conducted on the Emergent dataset, the learned model [25] with MMD regularisation accuracy results for agree, disagree and discuss stances are: 82.52, 69.05, 84.30, respectively. Performance and analysis of their model with different feature sets show that for relatedness class, Cosine Similarity feature leads to higher accuracy, WordLap feature contributes to relatedness and agree class. For both agree and disagree, classes, Reference Word and Polarity features have an increase of the accuracy while removing NGram features to enhance the detection for discussing class. This works' stance results for an emergent dataset for agree, disagree and discuss stances are: 83.12, 73.89, 89.13, outperform the best performing model. A performance comparison can be seen in table 10.

The results of the proposed model and the state-of-the-art stance detection model are reported in [25], in table 10. Some observations are made: (1) Dependency-based CNN, and constituency-based CNN improves the overall performance by detecting the complex syntactic structures of the claim and the conclusion. It can capture long-distance syntactic dependency. (2) using the Manhattan distance to infer the claim and the conclusion underlying semantic similarity based on the vector representation (final hidden states), help to capture the semantically equivalent of claim and conclusion. (3) the effectiveness of attention mechanism via emphasizing the words important to the semantics of the claim and conclusion by automatically search for the most relevant parts of an input sequence and assigns weights to those parts. (4) Current model's stance classification achieves better than the state-of-the-art [25], the agree stances with no big improvement, about 0.6 % while disagree and discuss stances are with significant improvement more than 4 % for both.

Table 10: Performance comparison of the model against the State-of-the-Art model [25] **for stance detection task on Emergent dataset.**

Model	Accuracy (%)		
	agree	disagree	discuss
Zhang et 's model [25]	82.52	69.05	84.30
This model	83.12	73.89	89.13

For veracity detection, this model obtains an accuracy of 69.8% Since most previous models on the Emergent dataset focus only on the stance detection task and leave the veracity task; no comparison with the baseline is made here.

Li et al.'s [13] Multi-task Network on RumorEval 2019 dataset. The model in [13] and [43] show better performance compared with the top-5 systems in RumourEval 2019 [45]. The model in [13], achieves the best performance for veracity detection, but they did not present results for stance detection as a single task. Regarding stance detection results, the best results are shown in [43], so that to evaluate this work's stance detection, a comparison is made with [43] model and for veracity checking, the comparison with [13]. The authors in [13] propose a multi-task learning approach for rumour detection and stance classification tasks using two task-specific layers and a middle layer as a shared layer between these two tasks. They incorporate user-level information as a supplementary indicator of credibility where the goal is to resolve the veracity of a rumour and apply attention-based LSTM network in the rumour detection process, utilising the hidden states of the stance detection layer in its attention step. They compare their model against the state-of-the-art models showing that the results outperform the state-of-the-art rumour detection approaches, as illustrated in table 11. They claim that the three characterises they flow contribute to enhance the performance results:

- Merging user credibility information in the rumour verification process
- Utilising attention mechanism to pay more attention to the important tweets
- Combining the stance information into the attention computation

Table 11: Rumour verification result on RumorEval [13] **and performance comparison of proposed model against them**

Method	Accuracy	Macro F1
Majority (False)	0.438	0.304
NileTMRG	0.57	0.539
BranchLSTM	0.5	0.491
MTL2	0.571	0.558
Li et al.'s model[13]	0.638	0.606
The proposed model	0.668	0.647

Khandelwal's Model [43]

They proposed the multi-task learning framework for jointly predicting rumour stance and veracity and trained different models by varying the type of sentence encode like Identity Encoder Transformer, Bi-LSTM and Longformer; different NLP features are analysed such as Structural, Content, Conversational, Affective, Emotion, LIWC and Speech-Act features. Models are named as Longformer + Identity Encoder, Longformer + Transformer and Longformer + BiLSTM as shown in table 12 and 13. Khandelwal [43] studied the positive effect of using the sentence encoder with or without NLP Features using the model named as Longformer + Identity Encoder and Longformer + Identity Encoder + NLP Features. In both cases, the performance results show that the sentence encoders help to learn the stance evolution to determine the proper category of veracity and neighbouring posts in the conversation thread helps in determining the stance category by extracting the features from each post in the conversation thread. For task A, stance detection, the work in [43] achieves the best Macro-f of 0.6720, while for task B, veracity checking, the work in [13] achieves the best Macro-f of 0.606

Table 12: Test results for Task B

Rank	The model	Macro-F
A	Longformer + Identity Encoder	0.3795
B	Longformer +Transformer	0.3363
C	Longformer +BiLSTM	0.4004
D	Longformer +Identity Encoder + NLP Features	0.4962
E	Longformer +Transformer + NLP Features	0.5327
F	Longformer +BiLSTM + NLP Features	0.5275
	Khandelwal's [43] Method – Top N_s using (D + E + F)	0.5868
	The proposed model	0.647

Table 13: Test results for Task A

Rank	The model	Macro-F
A	Longformer + Identity Encoder	0.5782
B	Longformer +Transformer	0.5807
C	Longformer +BiLSTM	0.5886
D	Longformer +Identity Encoder + NLP Features	0.6371
E	Longformer +Transformer + NLP Features	0.6389
F	Longformer +BiLSTM + NLP Features	0.6487
	Khandelwal's[43] Method – Top N_s using (D + E + F)	0.6720
	The proposed model	0.695

4.4. Discussions

From the results given above, it is obvious that the proposed method shows the best performance among these models. The proposed model outperforms in both tasks, achieving Macro F1 0.695 for Task A and 0.647 for task B. The remainder of the subsection provides an analysis and evaluation of the proposed model and the results

First, training this model with and without applying the first component: the main claim target extraction. The performance results revealed this guide the model to focus more on the main target of claim and contribute positively to performance for stance classification. Target-clause retrieving help to ignore noise information as the noise may give wrong indications to deceive the model. This model is trained to classify stances without considering the target information and a decreased accuracy is obtained. To show that the stance and rumour detection here benefits from target aware conclusion, experiments are conducted to detect the stance of evidence against claims without doing a conclusion based on the target. The change of macro-F1 scores on the two datasets shows the improvements by capturing certain words related to the target and eliminating the irrelevant. It outperforms previous best baseline methods by 3.8% for rumour data and 2.4 % for the Emergent data. The reason for this could be that the model detects the stance of the article against a claim by paying more attention to the claimed target, while the original article may have various aspects to talk about. It is observed that word alignment can capture the target information for better performance of stance detection as target-specific attention gives more concise information, discarding another target the claim does not concern with.

The results emerging from these experiments confirmed the effectiveness of generating conclusion conditioned on the target representation that is finally presented to the target claim, and showed that it could be useful by extracting salient information from a long article without including less salient information., A significant improvement on the general results of this model, on both tasks a and b is achieved. By comparing with baselines for the stance detection, the advantage of aspect aware conclusion is demonstrated. A significant improvement on an emergent dataset from 82.52 %, 69.05 %, 84.30 % to 83.12%, 73.89%, 89.13% for the three stance labels respectively. For rumour data, the macro- F is increased from 0.627 to 0.695.

Other observations have been made:

- For the Emergent data, the veracity detection accuracy decreases by 1.4% when the headline is not considered. This is particularly the case for a long article since the headline captures the main information in making first impressions to readers.
- When the generated conclusion does not cover the target of the claim or the extracted target is not true, the model fails to predict
- To investigate the applicability of the proposed model on new unseen data, where there is no knowledge related to this event, truth discovery is very beneficial to generalise in veracity prediction since it depends on estimation without supervision. In spite of unobserved samples, they may have semantic and syntactic features to that unseen news.
- Obviously, models augmented with truth discovery perform better than those without, i.e. assigning more scores to the claims inferred by more trusty sources.
- A significant improvement in integrating both tasks stance detection with humour prediction.
- Since the available information of trustworthiness of sources are not available and no prior information, the method in this work can greatly enhance reliability source inference by estimating the trust based on Argumentation-based Truth Discovery.
- The model fails to predict some stance labels correctly, may be due to the lack of current information, and other external evidence, e.g. warrant is needed so merging them may make additional enhancements, especially in the case.
- Utilising the claimed target helps the generator to produce a concise conclusion, and the administrator can narrow the cosine distance.
- For size limitation, deep learning models need high-volume of data for training, it requires larger datasets than what is currently available, so this model expected to perform better if more samples are obtained.

1. Conclusion

A hierarchical multi-task learning framework for jointly predicting rumour stance and veracity is proposed, where the source reliability is considered. A new deep learning model with a novel architecture is designed and studied the problem of discovering multiple truths from conflicting sources by connecting truth discovery methods with bipolar argumentation. The experiments on two datasets show that the proposed model outperforms the state-of-the-art models for the tasks of stance detection and rumour verification. Argumentation-based Truth Discovery provides an effective way towards veracity detection by discovering the acceptable arguments through reframing truth discovery in terms of argumentation; this implies describing the arguments and the attack and support relations. There are several ways to move the current work forward. The immediate work here involves source-claim and source-source relationships and mainly consider information richness to score the confidence of information. For future work, there is a plan to modify this model by considering other argumentation components as warrant and backing in Toulmin model and take into account other factors like the source reputations and user profile.

References

[1] S. Vosoughi, D. Roy, and S. Aral, “The Spread of True and False News Online,” *Science (80-.).*, vol. 1151, no. March, pp. 1146–1151, 2018, doi: 10.1126/science.aap9559.

[2] K. Shu, H. R. Bernard, and H. Liu, “Studying Fake News via Network Analysis: Detection and Mitigation,” in *Nitin Agarwal, Nima Dokoohaki, Serpil Tokdemir: Emerging Research Challenges and Opportunities in Computational Social Network Analysis and Mining*, Springer, Cham, 2019, pp. 43–65.

[3] Z. Jin, J. Cao, Y. Zhang, and J. Luo, “News Verification by Exploiting Conflicting Social Viewpoints in Microblogs,” in *30th AAAI Conference on Artificial Intelligence, AAAI*

2016, 2016, pp. 2972–2978.

[4] J. Harsin, "[Proto-Post-truth] The Rumour Bomb: Theorizing the Convergence of New and Old Trends in Mediated US Politics," *South. Rev. Commun. Polit. Cult.*, vol. 39, no. 1, 2006, pp. 84–110, 2018, [Online]. Available: https://search.informit.com.au/documentSummary;dn=264848460677220;res=IELAPA.

[5] G. L. Ciampaglia, P. Shiralkar, L. M. Rocha, J. Bollen, F. Menczer, and A. Flammini, "Computational fact checking from knowledge networks," *PLoS One*, vol. 10, no. 6, pp. 1–13, 2015, doi: 10.1371/journal.pone.0128193.

[6] M. Potthast, J. Kiesel, K. Reinartz, J. Bevendorff, and B. Stein, "A Stylometric Inquiry into Hyperpartisan and Fake News," *ACL 2018 - 56th Annu. Meet. Assoc. Comput. Linguist. Proc. Conf. (Long Pap.*, vol. 1, pp. 231–240, 2018, doi: 10.18653/v1/p18-1022.

[7] S. A. Alkhodair, S. H. H. Ding, B. C. M. Fung, and J. Liu, "Detecting Breaking News Rumors of EmergingTopics in Social Media," *Inf. Process. Manag.*, vol. 57, no. 2, p. 102018, 2020, doi: 10.1016/j.ipm.2019.02.016.

[8] Y. Li *et al.*, "A Survey on Truth Discovery," *ACM SIGKDD Explor. Newsl.*, vol. 17, no. 2, pp. 1–16, 2016, doi: 10.1145/2897350.2897352.

[9] B. Yorgancı, "Multi-sourced Information Trustworthiness Analysis: Applications and Theory," State University of New York at Buffalo, 2018.

[10] J. Singleton, "Truth Discovery : Who to Trust and What to Believe," in *Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020)*, 2020, no. May 9-13, pp. 2211–2213.

[11] J. Singleton, "On the Link Between Truth Discovery and Bipolar Abstract Argumentation," *Online Handb. Argumentation AI*, vol. 1, no. June, pp. 43–47, 2020.

[12] Q. Li, Q. Zhang, L. Si, and Y. Liu, "Rumor detection on social media: Datasets, methods and opportunities," in *Proceedings of the Second Workshop on Natural Language Processing for Internet Freedom: Censorship, Disinformation, and Propaganda*, 2019, pp. 66–75, doi: 10.18653/v1/d19-5008.

[13] Q. Li, Q. Zhang, and L. Si, "Rumor Detection By Exploiting User Credibility Information, Attention and Multi-task Learning," in *Proceedings ofthe 57th Annual Meeting ofthe Association for Computational Linguistics*, 2020, pp. 1173–1179, doi: 10.18653/v1/p19-1113.

[14] G. Gorrell, K. Bontcheva, L. Derczynski, E. Kochkina, M. Liakata, and A. Zubiaga, "RumourEval 2019: Determining Rumour Veracity and Support for Rumours," in *Proceedings ofthe 13th International Workshop on Semantic Evaluation (SemEval-2019)*, 2019, pp. 845–854.

[15] W. Ferreira and A. Vlachos, "Emergent: A Novel Data-set for Stance Classification," in *2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT 2016 - Proceedings of the Conference*, 2016, vol. June 12-17, no. 1, pp. 1163–1168, doi: 10.18653/v1/n16-1138.

[16] I. E. [Ed] Chiluwa and S. A. [Ed] Samoilenko, *Handbook of Research on Deception, Fake News, and Misinformation Online*. 2019.

[17] V. L. Rubin, Y. Chen, and N. J. Conroy, "Deception Detection for News: Three Types of Fakes," in *Proceedings of the Association for Information Science and Technology*, 2015, vol. 52, no. 1, pp. 1–4, doi: 10.1002/pra2.2015.145052010083.

[18] N. J. Conroy, V. L. Rubin, and Y. Chen, "Automatic deception detection: Methods for finding fake news," in *Proceedings of the Association for Information Science and Technology*, 2015, vol. 52, no. 1, pp. 1–4, doi: 10.1002/pra2.2015.145052010082.

[19] B. D. Horne and S. Adali, "This Just In-Fake News Packs a Lot in Title, Uses Simpler, Repetitive Content in Text Body, More Similar to Satire than Real News," in *The 2nd International Workshop on News and Public Opinion at ICWSM*, 2017, pp. 40–49, doi: 10.18653/v1/w18-5507.

[20] F. Yang, A. Mukherjee, and E. Gragut, "Satirical News Detection and Analysis using Attention Mechanism and Linguistic Features," in *EMNLP 2017 - Conference on Empirical Methods in Natural Language Processing, Proceedings*, 2017, pp. 1979–1989, doi: 10.18653/v1/d17-1211.

[21] A. Zubiaga, A. Aker, K. Bontcheva, M. Liakata, and R. Procter, "Detection and Resolution of Rumours in Social Media: A Survey," *ACM Comput. Surv.*, vol. 51, no. 2, p. Article 32, 2018, doi: 10.1145/3161603.

[22] Z. Zhao, P. Resnick, and Q. Mei, "Enquiring Minds: Early Detection of Rumors in Social Media from Enquiry Posts," in *Proceedings of the 24th i International World Wide Web Conference*, 2015, pp. 1395–1405, doi: 10.1145/2736277.2741637.

[23] A. Zubiaga, G. Wong Sak Hoi, M. Liakata, and R. Procter, "PHEME Dataset of Rumours and Non-rumours," *figshare. Dataset*, 2016. https://figshare.com/articles/dataset/PHEME_dataset_of_rumours_and_non-rumours/4010619/1 (accessed Nov. 20, 2020).

[24] T. Saikh, A. Anand, A. Ekbal, and P. Bhattacharyya, "A Novel Approach Towards Fake News Detection: Deep Learning Augmented with Textual Entailment Features," in *International Conference on Applications of Natural Language to Information Systems*, 2019, vol. 11608 LNCS, pp. 345–358, doi: 10.1007/978-3-030-23281-8_30.

[25] Q. Zhang, S. Liang, A. Lipani, Z. Ren, and E. Yilmaz, "From Stances' Imbalance to Their Hierarchical Representation and Detection," in *The Web Conference 2019 - Proceedings of the World Wide Web Conference, WWW 2019*, 2019, vol. May 13-17, no. 1, pp. 2323–2332, doi: 10.1145/3308558.3313724.

[26] I. Augenstein, T. Rocktäschel, A. Vlachos, and K. Bontcheva, "Stance Detection with Bidirectional Conditional Encoding," in *EMNLP 2016 - Conference on Empirical Methods in Natural Language Processing, Proceedings*, 2016, pp. 876–885, doi: 10.18653/v1/d16-1084.

[27] Jeffrey Pennington, R. Socher, and C. D. Manning, "GloVe: Global Vectors for Word Representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, no. October, pp. 1532–1543, [Online]. Available: https://www.aclweb.org/anthology/D14-1162.pdf.

[28] S. Bajaj, "' The Pope Has a New Baby !' Fake News Detection Using Deep Learning," *CS 224N*, pp. 1–8, 2017, [Online]. Available: https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1174/reports/2710385.pdf.

[29] M. Mohtarami, R. Baly, J. Glass, P. Nakov, L. Màrquez, and A. Moschitti, "Automatic stance

detection using end-To-end memory networks," *NAACL HLT 2018 - 2018 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. - Proc. Conf.*, vol. 1, pp. 767–776, 2018, doi: 10.18653/v1/N18-1070.

[30] Q. Zhang, E. Yilmaz, and S. Liang, "Ranking-based Method for News Stance Detection," in *The Web Conference 2018 - Companion of the World Wide Web Conference, WWW 2018*, 2018, pp. 41–42, doi: 10.1145/3184558.3186919.

[31] A. Hanselowski, A. Pvs, B. Schiller, and F. Caspelherr, "Description of the System Developed by Team Athene in the FNC-1," *Technical report*, 2017. https://github.com/hanselowski/athene_system/blob/master/system_description_athene.pdf (accessed Nov. 24, 2020).

[32] P. Bourgonje, J. Moreno Schneider, and G. Rehm, "From Clickbait to Fake News Detection: An Approach based on Detecting the Stance of Headlines to Articles," in *Proceedings ofthe 2017 EMNLP Workshop on Natural Language Processing meets Journalism*, 2017, pp. 84–89, doi: 10.18653/v1/w17-4215.

[33] B. Riedel, I. Augenstein, G. P. Spithourakis, and S. Riedel, "A simple but tough-to-beat baseline for the Fake News Challenge stance detection task," *arXiv:1707.03264*, pp. 1–6, 2018.

[34] X. Wang, C. Yu, S. Baumgartner, and F. Korn, "Relevant Document Discovery for Fact-Checking Articles," in *Companion Proceedings of the The Web Conference 2018, International World Wide Web Conferences Steering Committee*, 2018, pp. 525–533, doi: 10.1145/3184558.3188723.

[35] J. Li, X. Hu, J. Tang, and H. Liu, "Unsupervised Streaming Feature Selection in Social Media," in *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, 2015, no. October 13, pp. 1041–1050, doi: http://dx.doi.org/10.1145/2806416.2806501.

[36] P. Wei, N. Xu, and W. Mao, "Modeling Conversation Structure and Temporal Dynamics for Jointly Predicting Rumor Stance and Veracity," in *EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference*, 2019, pp. 4787–4798, doi: 10.18653/v1/d19-1485.

[37] M. Glenski, T. Weninger, and S. Volkova, "Identifying and Understanding User Reactions to Deceptive and Trusted Social News Sources," *ACL 2018 - 56th Annu. Meet. Assoc. Comput. Linguist. Proc. Conf. (Long Pap.*, vol. 2, no. 1, pp. 176–181, 2018, doi: 10.18653/v1/p18-2029.

[38] M. Mendoza, B. Poblete, and C. Castillo, "Twitter Under Crisis: Can we trust what we RT?," in *SOMA 2010 - Proceedings of the 1st Workshop on Social Media Analytics*, 2010, pp. 71–79, doi: 10.1145/1964858.1964869.

[39] R. Procter, A. Voss, and F. Vis, "Reading the Riots on Twitter: Methodological Innovation for the Analysis of Big Data," *Int. J. Soc. Res. Methodol. Comput. Soc. Sci. Res. Strateg. Des. Methods*, vol. 16, no. 3, pp. 197–214, 2013, doi: 10.1080/10439463.2013.780223.

[40] V. Qazvinian, E. Rosengren, D. R. Radev, and Q. Mei, "Rumor has it Identifying Misinformation in Microblogs," in *Proceedings ofthe 2011 Conference on Empirical Methods in Natural Language Processing*, 2011, no. July 27-31, pp. 1589–1599, [Online]. Available: https://www.aclweb.org/anthology/D11-1147.pdf.

[41] Q. Zhang, S. Zhang, J. Dong, J. Xiong, and X. Cheng, "Automatic Detection of Rumor on Social Network," *NLPCC2015, Nat. Lang. Process. Chinese Comput. Springer, Cham*, vol. LNAI 9362, pp. 113–122, 2015, doi: 10.1007/978-3-319-25207-0_10.

[42] A. Zubiaga, E. Kochkina, M. Liakata, R. Procter, and M. Lukasik, "Stance Classification in Rumours as a Sequential Task Exploiting the Tree Structure of Social Media Conversations," in *COLING 2016 - 26th International Conference on Computational Linguistics, Proceedings of COLING 2016: Technical Papers*, 2016, pp. 2438–2448.

[43] A. Khandelwal, "Fine-Tune Longformer for Jointly Predicting Rumor Stance and Veracity," *arXiv Prepr.*, 2020, doi: 10.1145/3430984.3431007.

[44] R. Yang, W. Xie, C. Liu, and D. Yu, "BLCU_NLP at SemEval-2019 Task 7: An Inference Chain-based GPT Model for Rumour Evaluation," in *Proceedings ofthe 13th International Workshop on Semantic Evaluation (SemEval-2019)*, 2019, no. June 6-7, pp. 1090–1096, doi: 10.18653/v1/s19-2191.

[45] M. Fajcik, L. Burget, and P. Smrz, "BUT-FIT at SemEval-2019 Task 7: Determining the Rumour Stance with Pre-Trained Deep Bidirectional Transformers," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, 2019, pp. 1097–1104, doi: 10.18653/v1/s19-2192.

[46] I. Baris, L. Schmelzeisen, and S. Staab, "CLEARumor at SemEval-2019 Task 7: ConvoLving ELMo against rumors," in *3th International Workshop on Semantic Evaluation*, 2019, no. June 06-07, doi: 10.18653/v1/s19-2193.

[47] A. Radford, K. Narasimhan, A. Tim Salimans, and I. Sutskever, "Improving Language Understanding with Unsupervised Learning," *Technical report, Open AI*, 2018. https://openai.com/blog/language-unsupervised/ (accessed Nov. 20, 2020).

[48] M. Peters *et al.*, "Deep Contextualized Word Representations," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, (Long Papers)*, 2018, vol. 1, no. June 1-6, pp. 2227–2237, doi: 10.18653/v1/N18-1202.

[49] J. Thorne, A. Vlachos, C. Christodoulopoulos, and A. Mittal, "FEVER: a Large-scale Dataset for Fact Extraction and VERification," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, (Long Papers)*, 2018, vol. 1, no. June, pp. 809–819, doi: 10.17863/CAM.40620.

[50] M. Seo, A. Kembhavi, A. Farhadi, and H. Hajishirzi, "Bidirectional Attention Flow for Machine Comprehension," in *ICLR 2017 conference submission*, 2016, pp. 1–13, [Online]. Available: http://arxiv.org/abs/1611.01603.

[51] Y. Nie, H. Chen, and M. Bansal, "Combining Fact Extraction and Verification with Neural Semantic Matching Networks," in *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*, 2019, vol. 33, no. 01, pp. 6859–6866, doi: 10.1609/aaai.v33i01.33016859.

[52] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *NAACL -HLT 2019 - 2019*

Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2019, vol. 1, no. Mlm, pp. 4171–4186, doi: 10.18653/v1/N19-1423.

[53] A. Soleimani, C. Monz, and M. Worring, "BERT for Evidence Retrieval and Claim Verification," *J. M. Jose al. ECIR 2020, LNCS 12036*, no. 3, pp. 359–366, 2020, doi: 10.1007/978-3-030-45442-5.

[54] M. E. Peters and A. Cohan, "Longformer: The Long-Document Transformer," *arXiv Prepr. arXiv2004.05150v1*, 2020.

[55] O. Enayet and S. R. El-Beltagy, "NileTMRG at SemEval-2017 Task 8: Determining Rumour and Veracity Support for Rumours on Twitter.," in *Proceedings ofthe 11th International Workshop on Semantic Evaluations (SemEval-2017)*, 2017, pp. 470–474, doi: 10.18653/v1/s17-2082.

[56] Q. Li, Q. Zhang, and L. Si, "eventAI at SemEval-2019 Task 7: Rumor Detection on Social Media by Exploiting Content, User Credibility and Propagation Information," in *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*, 2019, pp. 855–859, doi: 10.18653/v1/s19-2148.

[57] E. Kochkina, M. Liakata, and A. Zubiaga, "All-in-one: Multi-task Learning for Rumour Verification," in *Proceedings of the 27th International Conference on Computational Linguistics*, 2018, pp. 3402–3413, [Online]. Available: https://www.aclweb.org/anthology/C18-1288.

[58] J. Ma, W. Gao, and K. Wong, "Detect Rumor and Stance Jointly by Neural Multi-task Learning," in *Proceedings of the Web Conference (WWW 2018 Companion)*, 2018, pp. 585–593, doi: 10.1145/3184558.3188729.

[59] L. Poddar, W. Hsu, M. L. Lee, and S. Subramaniyam, "Predicting Stances in Twitter Conversations for Detecting Veracity of Rumors: a Neural Approach," in *Proceedings of the 30th International Conference on Tools with Artificial Intelligence, ICTAI*, 2018, vol. 2018-Novem, pp. 65–72, doi: 10.1109/ICTAI.2018.00021.

[60] J. Ma, W. Gao, and K. Wong, "Rumor Detection on Twitter with Tree-structured Recursive Neural Networks," in *Proceedings ofthe 56th Annual Meeting ofthe Association for Computational Linguistics (Long Papers)*, 2018, pp. 1980–1989, doi: 10.18653/v1/P18-1184.

[61] J. Pasternack and D. Roth, "Knowing What to Believe (when you already know something)," in *Proceedings ofthe 23rd International Conference on Computational Linguistics (Coling 2010)*, 2010, vol. 2, no. August, pp. 877–885.

[62] Y. Li *et al.*, "On the Discovery of Evolving Truth," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015, vol. 2015-Augus, pp. 675–684, doi: 10.1145/2783258.2783277.

[63] F. Ma *et al.*, "FaitCrowd: Fine Grained Truth Discovery for Crowdsourced Data Aggregation," in *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015, vol. 2015-Augus, pp. 745–754, doi: 10.1145/2783258.2783314.

[64] J. Marshall, A. Argueta, and D. Wang, "A Neural Network Approach for Truth Discovery in Social Sensing," in *Proceedings - 14th IEEE International Conference on Mobile Ad Hoc and Sensor Systems, MASS 2017*, 2017, pp. 343–347, doi: 10.1109/MASS.2017.26.

[65] X. Yin, J. Han, and P. S. Yu, "Truth Discovery with Multiple Conflicting Information Providers on the Web," *IEEE Trans. Knowl. Data Eng.*, vol. 20, no. 6, pp. 796–808, 2008, doi: 10.1109/TKDE.2007.190745.

[66] X. L. Dong, L. Berti-Equille, and D. Srivastava, "Integrating Conflicting Data: The Role of Source Dependence," in *Proceedings of the VLDB Endowment*, 2009, vol. 2, no. 1, pp. 550–561, doi: 10.14778/1687627.1687690.

[67] J. Pasternack and D. Roth, "Making Better Informed Trust Decisions with Generalized Fact-Finding," in *Proceedings of the 22nd International Joint Conference on Artificial Intelligence Making*, 2011, pp. 2324–2329, doi: 10.5591/978-1-57735-516-8/IJCAI11-387.

[68] A. Galland, S. Abiteboul, A. Marian, and P. Senellart, "Corroborating Information from Disagreeing Views," in *Proceedings of the third ACM International Conference on Web Search and Data Mining (WSDM)*, 2010, pp. 131–140, doi: 10.1145/1718487.1718504.

[69] F. Li, M. L. Lee, and W. Hsu, "Entity Profiling with Varying Source Reliabilities," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014, no. August, pp. 1146–1155, doi: 10.1145/2623330.2623685.

[70] C. Dai, D. Lin, E. Bertino, and M. Kantarcioglu, "An Approach to Evaluate Data Trustworthiness Based on Data Provenance," in *Proceedings of the 5th VLDB Workshop on Secure Data Management*, 2008, pp. 82–98, doi: 10.1007/978-3-540-85259-9_6.

[71] X. Dong *et al.*, "Knowledge Vault: A Web-scale Approach to Probabilistic Knowledge Fusion," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014, pp. 601–610, doi: 10.1145/2623330.2623623.

[72] Q. Li *et al.*, "A Confidence-Aware Approach for Truth Discovery on Long-Tail Data," *Proc. VLDB Endow.*, vol. 8, no. 4, pp. 425–436, 2014, doi: 10.14778/2735496.2735505.

[73] Q. Li, Y. Li, J. Gao, B. Zhao, W. Fan, and J. Han, "Resolving Conflicts in Heterogeneous Data by Truth Discovery and Source Reliability Estimation," in *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data (SIGMOD '14)*, 2014, pp. 1187–1198, doi: 10.1145/2588555.2610509.

[74] D. Zhou, J. C. Platt, S. Basu, and Y. Mao, "Learning from the Wisdom of Crowds by Minimax Entropy," in *Proceedings of the 25th International Conference on Neural Information Processing Systems NIPS*, 2012, vol. 2, no. December, pp. 2195–2203, [Online]. Available: http://dblp.uni-trier.de/db/conf/nips/nips2012.html#ZhouPBM12.

[75] D. Wang, L. Kaplan, H. Le, and T. Abdelzaher, "On Truth Discovery in Social Sensing: A Maximum Likelihood Estimation Approach," in *IPSN'12 - Proceedings of the 11th International Conference on Information Processing in Sensor Networks*, 2012, no. April, pp. 233–244, doi: 10.1145/2185677.2185737.

[76] M. Samadi, P. Talukdar, M. Veloso, and M. Blum, "ClaimEval: Integrated and Flexible Framework for Claim Evaluation Using Credibility of Sources," in *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, 2016, no. February, pp. 222–228, [Online]. Available: http://dblp.uni-trier.de/db/conf/aaai/aaai2016.html#SamadiTVB

16.
[77] N. Nakashole and T. M. Mitchell, “Language-Aware Truth Assessment of Fact Candidates,” in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014*, 2014, vol. 1, pp. 1009–1019, doi: 10.3115/v1/p14-1095.
[78] X. Wang, Q. Z. Sheng, X. S. Fang, X. Li, X. Xu, and L. Yao, “Approximate Truth Discovery via Problem Scale Reduction,” in *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, CIKM ’15*, 2015, vol. 19-23-Oct-, pp. 503–512, doi: 10.1145/2806416.2806444.
[79] S. Zhi *et al.*, “Modeling Truth Existence in Truth Discovery,” in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015, vol. 2015-Augus, pp. 1543–1552, doi: 10.1145/2783258.2783339.
[80] B. Zhao, B. I. P. Rubinstein, J. Gemmell, and J. Han, “A Bayesian Approach to Discovering Truth from Conflicting Sources for Data Integration,” in *Proceedings of the VLDB Endowment (PVLDB)*, 2012, vol. 5, no. 6, pp. 550–561, doi: 10.14778/2168651.2168656.
[81] B. Zhao and J. Han, “A Probabilistic Model for Estimating Real-valued Truth from Conflicting Sources,” 2012.
[82] L. Li, B. Qin, W. Ren, and T. Liu, “Truth Discovery with Memory Network,” *Tsinghua Sci. Technol.*, vol. 22, no. 6, pp. 609–618, 2017, doi: 10.23919/TST.2017.8195344.
[83] K. Broelemann, T. Gottron, and G. Kasneci, “Restricted Boltzmann Machines for Robust and Fast Latent Truth Discovery,” *CoRR*, vol. abs/1801.0, 2018, [Online]. Available: http://dblp.uni-trier.de/db/journals/corr/corr1801.html#abs-1801-00283.
[84] N. Choudhary, R. Singh, I. Bindlish, and M. Shrivastava, “Neural Network Architecture for Credibility,” in *In the proceedings of CICLING 2018*, 2018, pp. 1–13, [Online]. Available: http://dblp.uni-trier.de/db/journals/corr/corr1803.html#abs-1803-10547.
[85] Q. Li, X. Liu, R. Fang, A. Nourbakhsh, and S. Shah, *User Behaviors in Newsworthy Rumors : A Case Study of Twitter*, no. ICWSM. Association for the Advancement of Artificial Intelligence (www.aaai.org), 2016, pp. 627–630.
[86] Y. Liu and Y. F. B. Wu, “Early Detection of Fake News on Social Media Through Propagation Path Classification with Recurrent and Convolutional Networks,” in *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, 2018, pp. 354–361, [Online]. Available: http://dblp.uni-trier.de/db/conf/aaai/aaai2018.html#LiuW18.
[87] A. Vlachos and S. Riedel, “Fact Checking: Task Definition and Dataset Construction,” in *Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science*, 2015, no. January, pp. 18–22, doi: 10.3115/v1/w14-2508.
[88] W. Y. Wang, “‘Liar, Liar Pants on Fire’: A New Benchmark Dataset for Fake News Detection,” in *ACL 2017 - 55th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, 2017, vol. 2, pp. 422–426, doi: 10.18653/v1/P17-2067.
[89] B. Keller, A. Labrique, K. M. Jain, A. Pekosz, and O. Levine, “Incorporating Copying Mechanism in Sequence-to-Sequence Learning,” *J. Med. Internet Res.*, vol. 16, no. 1, p. e8, 2014, [Online]. Available: https://www.jmir.org/2014/1/e8/.
[90] A. See, P. J. Liu, and C. D. Manning, “Get To The Point: Summarization with Pointer-Generator Networks,” in *ACL 2017 - 55th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, 2017, vol. 1, pp. 1073–1083, doi: 10.18653/v1/P17-1099.
[91] W. T. Hsu, C. K. Lin, M. Y. Lee, K. Min, J. Tang, and M. Sun, “A Unified Model for Extractive and Abstractive Summarization using Inconsistency Loss,” in *ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, 2018, vol. 1, pp. 132–141, doi: 10.18653/v1/p18-1013.
[92] Q. Zhou, N. Yang, F. Wei, and M. Zhou, “Selective Encoding for Abstractive Sentence Summarization,” in *ACL 2017 - 55th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, 2017, vol. 1, pp. 1095–1104, doi: 10.18653/v1/P17-1101.
[93] Y. C. Chen and M. Bansal, “Fast Abstractive Summarization with Reinforce-Selected Sentence Rewriting,” in *ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, 2018, vol. 1, no. 2017, pp. 675–686, doi: 10.18653/v1/p18-1063.
[94] X. Chen, S. Gao, C. Tao, Y. Song, D. Zhao, and R. Yan, “Iterative Document Representation Learning Towards Summarization with Polishing,” in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018*, 2020, pp. 4088–4097, doi: 10.18653/v1/d18-1442.
[95] K. M. Hermann *et al.*, “Teaching Machines to Read and Comprehend,” in *Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS’15)*, 2015, pp. 1693–1701, [Online]. Available: http://dblp.uni-trier.de/db/conf/nips/nips2015.html#HermannKGEKSB15.
[96] G. Heinrich, “Parameter Estimation for Text Analysis,” 2009. [Online]. Available: http://www.arbylon.net/publications/text-est.pdf.
[97] S. Gao *et al.*, “Abstractive Text Summarization by Incorporating Reader Comments,” *33rd AAAI Conf. Artif. Intell.*, vol. 33, no. 01, pp. 6399–6406, 2019, doi: 10.1609/aaai.v33i01.33016399.
[98] D. Bahdanau, K. H. Cho, and Y. Bengio, “Neural Machine Translation by Jointly Learning to Align and Translate,” in *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2015, pp. 1–15.
[99] P. Nema, M. M. Khapra, A. Laha, and B. Ravindran, “Diversity driven Attention Model for Query-based Abstractive Summarization,” in *ACL 2017 - 55th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, 2017, vol. 1, pp. 1063–1072, doi: 10.18653/v1/P17-1098.
[100] J. Hasselqvist, N. Helmertz, and M. Kågebäck, “Query-Based Abstractive Summarization Using Neural Networks,” *CoRR, abs/1712.06100*, 2017, [Online]. Available: http://dblp.uni-trier.de/db/journals/corr/corr1712.html#abs-1712-06100.

[101] C.-Y. Lin, “ROUGE: A Package for Automatic Evaluation of Summaries,” in *Proceedings of the ACL-04 Workshop on Text Summarization Branches Out (WAS 2004)*, 2004, no. July, pp. 74–81, [Online]. Available: https://www.aclweb.org/anthology/W04-1013.

[102] J. Mueller and A. Thyagarajan, “Siamese Recurrent Architectures for Learning Sentence Similarity,” in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)*, 2016, no. November, pp. 2786–2792, [Online]. Available: http://dblp.uni-trier.de/db/conf/aaai/aaai2016.html#MuellerT16.

[103] C. Cayrol and M. C. Lagasquie-Schiex, *Gradual Valuation for Bipolar Argumentation Frameworks*, vol. 3571 LNAI, no. June. Berlin, Heidelberg: Springer, Berlin, Heidelberg, 2005.

[104] A. Barrón-Cedeño, I. Jaradat, G. Da San Martino, and P. Nakov, “Proppy: Organizing the News Based on Their Propagandistic Content,” *Inf. Process. Manag.*, vol. 56, no. 5, pp. 1849–1864, 2019, doi: 10.1016/j.ipm.2019.03.005.

[105] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed Representations of Words and Phrases and their Compositionality,” in *Advances in Neural Information Processing Systems 26*, 2013, pp. 3111–3119.