Towards Automatic Fake News Detection: Cross-Level Stance Detection in News Articles

Costanza Conforti cc918@cam.ac.uk
Mohammad Taher Pilehvar mp792@cam.ac.uk
Nigel Collier nhc30@cam.ac.uk
Language Technology Lab, University of Cambridge

Abstract

In this paper, we propose to adapt the four-staged pipeline proposed by Zubiaga et al. (2018) for the Rumor Verification task to the problem of Fake News Detection. We show that the recently released FNC-1 corpus covers two of its steps, namely the *Tracking* and the *Stance Detection* task. We identify asymmetry in length in the input to be a key characteristic of the latter step, when adapted to the framework of Fake News Detection, and propose to handle it as a specific type of *Cross-Level Stance Detection*. Inspired by theories from the field of Journalism Studies, we implement and test two architectures to successfully model the internal structure of an article and its interactions with a claim.

1 Introduction

The rise of social media platforms, which allow for real-time posting of news with very little (or none at all) editorial review at the source, is responsible for an unprecedented growth in the amount of the information available to the public. While this constitutes an invaluable source of free information, it also facilitates the spread of misinformation. In particular, the literature distinguishes between *rumors*, i.e., pieces of information which are unverified at the time of posting and therefore can turn out to be true or false, and *fake news* (or *hoaxes*), i.e., false stories which are instrumentally made up with the intent to mislead the readers and spread disinformation (Zubiaga et al., 2018).

Both Rumor Verification (RV) and Fake News Detection (FND) constitute very difficult tasks even for trained professionals. Therefore, approaching them in an end-to-end fashion has generally been avoided. Both tasks, however, can be easily split into a number of sub-steps. For instance, Zubiaga et al. (2018) proposed a model for RV which consists of four stages: a rumor *detection* stage, where potentially rumorous posts are identified, followed by a *tracking* stage, where posts concerning the identified rumor are collected; after determining the orientation expressed in each post with respect to the rumor (*stance detection*), the final truth value of the rumor is obtained by aggregating those single stance judgments (*veracity classification*). As shown in Figure 1, this pipeline can be naturally adapted to FND.

In recent years, several efforts have been made by the research community toward the automatization of some of these stages, in order to provide effective tools to enhance the performance of human journalists in rumor and fake news debunking (Thorne and Vlachos, 2018). Concerning FND, Pomerleau and Rao (2017) recently released a dataset for the Stance Detection step in the framework of the Fake News Challenge[1] (FNC-1). The core of the corpus is constituted by a collection of articles discussing 566 claims, 300 of which come from the EMERGENT dataset (Ferreira and Vlachos, 2016). Each article is summarized in a headline and labeled as *agreeing* (AGR), *disagreeing* (DSG) or *discussing* (DSC) the claim. Additionally, *unrelated* (UNR) samples were created by pairing headlines with random articles. The goal of the challenge was to classify the pairs constituted by a headline and an article as AGR, DSG, DSC or UNR.

Following the pipeline discussed above, it is clear that the FNC-1 actually covers two of the four steps, namely: (1) The *tracking* step, consisting in filtering out the irrelevant UNR samples; (2) The actual *stance detection* step, consisting in the classification of a related headline/article pair into AGR, DSC or DSC.

[1] http://www.fakenewschallenge.org/

Proceedings of the First Workshop on Fact Extraction and VERification (FEVER), pages 40–49
Brussels, Belgium, November 1, 2018. ©2018 Association for Computational Linguistics

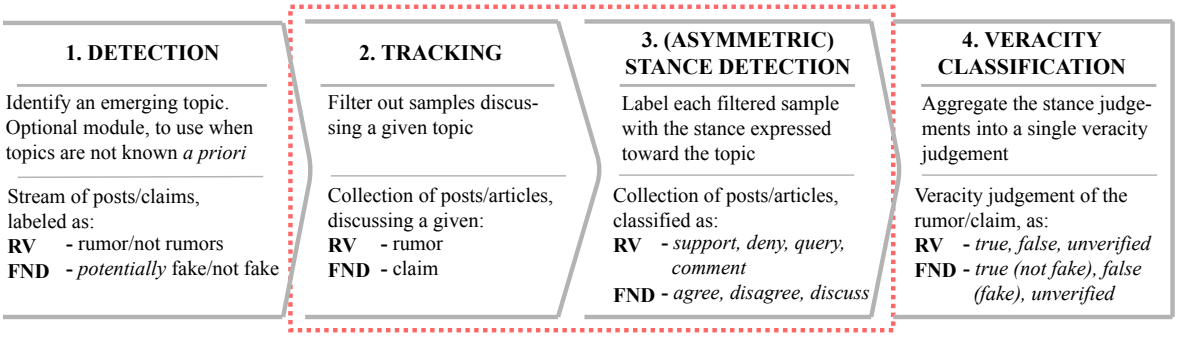

Figure 1: The rumor verification (RV) pipeline proposed by Zubiaga et al. (2018). The first row describes the corresponding step whereas the second row shows the outputs of each step for both the RV and the fake news detection (FND) tasks. The red rectangle indicates steps covered by the FNC-1 corpus. Figure adapted from Zubiaga et al. (2018).

Note that the amount of semantic understanding needed for the second task is much higher than for the first. In fact, even humans struggle in the related sample classification, as empirically demonstrated by Hanselowski et al. (2018): the inter-annotator agreement of five human judges drops from Fleiss' κ of .686 to .218, after filtering out the UNR samples. For this reason, we concentrate on the *stance detection* step, and we make the following contributions:

1. We identify asymmetry in length between headlines and articles as a key characteristic of the FNC-1 corpus: on average, an article contains more than 30 times the number of words contained in its associated headline. This is peculiar with respect to most of the commonly used datasets for stance detection (Mohammad et al., 2017) and require the development of architectures specifically tailored to this considerable asymmetry. Following on the terminology introduced by Jurgens et al. (2014) for Semantic Similarity, we propose to handle the problem as a *Cross-Level Stance Detection* task. To our knowledge, it is the first time that this task is investigated in isolation.

2. Inspired by theoretical principles in the field of Journalism Studies, we propose two simple neural architectures to model the argumentative structure of an article, and its complex interplay with a headline. We demonstrate that our systems can beat a strong feature-based baseline, based on one of the FNC-1 winning architectures, and that they can successfully model the internal structure of a news article and its relations with a claim, leveraging only word embeddings as input.

2 Related Work

2.1 Stance Detection

Stance Detection (SD) has been defined as the task of determining the attitude expressed in a *short piece of text* with respect to a target, usually expressed with one or few words (as *Feminism* or *Climate Change*, Mohammad et al. (2016)). In fact, most of the available corpora for SD consider very short samples, as Tweets. SD became very popular in recent years, resulting in a large number of publications (Mohammad et al., 2017).

To our knowledge, however, no one explicitly considered the problem of stance detection giving as input two items which are considerably asymmetric in length, that is, a long and structured document and a target expressed in the form of a complete sentence and not as a concept. For this reason, we propose to call the task introduced in the FNC-1 challenge *Cross-Level Stance Detection*. This is in line with the definition of Cross-Level Semantic Similarity, which measures the degree to which the meaning of a larger (in terms of length) linguistic item is captured by a smaller item (Jurgens et al., 2014).

After reporting on the systems participating to the FNC-1, which released the first SD dataset collecting long documents, we briefly mention some of the most relevant works on SD using Twitter data.

Fake News Challenge. With more than 50 participating groups, the FNC-1 drew high interest from both the research community and in-

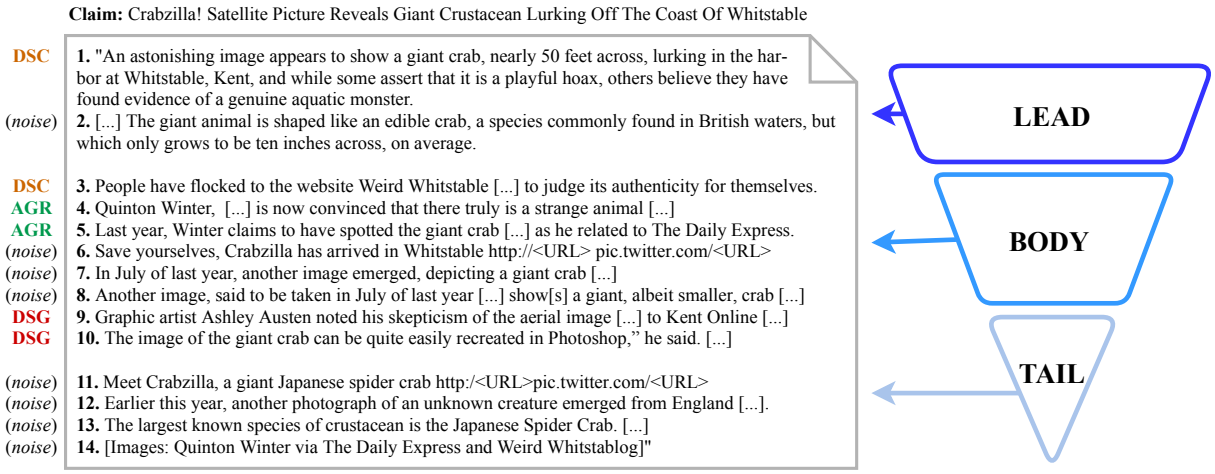

Figure 2: Article from the FNC-1 test set (sample no. 998), analyzed following the *inversed pyramid* principles (Scanlan, 1999). Notice that single sentences may express a different stance with respect to a claim, while others can be irrelevant, as shown in the leftmost column.

dustry. Due to the high number of UNR samples, which constituted almost three quarters of the training set, most of the groups proposed architectures which could perform well in this specific class - that is, in the *tracking* step of the FND pipeline. The second (Hanselowski et al., 2017) and third (Riedel et al., 2017) classified teams proposed multi-layer perceptrons (MLPs)-based systems. The best performing system (Baird et al., 2017) is an ensemble of a convolutional neural network (CNN) and a gradient-boosted decision tree. All models, with the exception of the CNN, take as input a number of hand-engineered features. Recently, Hanselowski et al. (2018) enriched the feature set used in Hanselowski et al. (2017) and added a stacked BiLSTM layer to their model, resulting in a modest gain in performance.

All models described above performed very well in the UNR classification (with F_1 usually above .98 for this class), achieving considerably worse results on the related samples (Hanselowski et al., 2018).

Rumor Stance Detection on Tweets. The most commonly used datasets for rumor stance detection, the RumorEval (Derczynski et al., 2017) and the PHEME (Zubiaga et al., 2016b) corpora, collect Tweets. State-of-the-art results on the PHEME corpus has been obtained by Aker et al. (2017), who used a very rich set of problem-specific features. Their model beat the previous state-of-the-art system by Zubiaga et al. (2016a), who modeled the tree-structured Twitter conversations using a LSTM, taking as input a conversation's branch at time.

2.2 Journalism Studies: News-writing Prose

Each genre develops its peculiar narrative forms, which allow for the most effective transmission of a message. In modern news-writing prose, especially in the Anglo-Saxon journalism, the *inverted pyramid* style is widely adopted (Scanlan, 1999).

Key element of this well standardized template consists in the fact that the most newsworthy facts (the so-called *5W*), are presented at the very beginning of the article - the *lead* - with the remaining information following, in order of importance, in the *body* of the article: in this section, we can find less essential element as quotes, interviews and background or explanatory information; any additional input, as related stories, images and credits, are put in the very last paragraphs, the *tail* (Scanlan, 1999). Usually, no more than one or two ideas are expressed in the same paragraph (Sun Technical Publications, 2003). Those characteristic elements are clearly visible in Figure 2. This style is particularly suited for rapidly evolving breaking news event, where a journalist can update an article by attaching a new paragraph with the last updates at the beginning of it. Moreover, putting most newsworthy facts at the beginning of an article allows the impatient readers to quickly decide on their level of interest in the report.

After manual analysis of excerpts of the FNC-1 corpus, we concluded that most articles were actu-

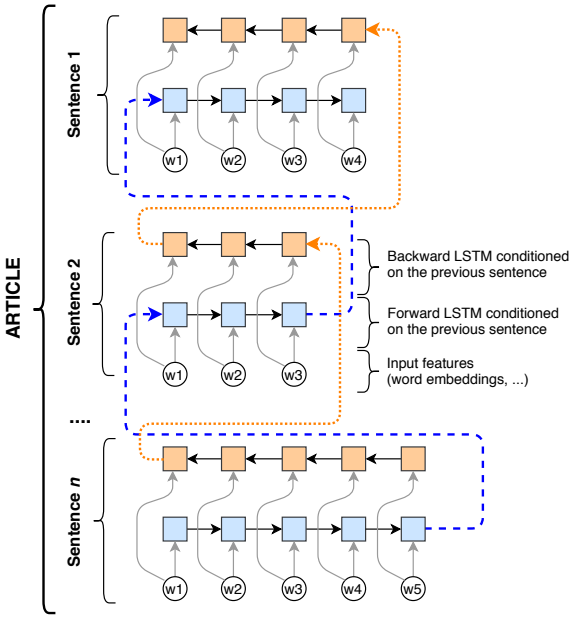

Figure 3: The architecture of our article encoder, which is based on that of Augenstein (2016). Dotted arrows represent conditional encoding and networks with the same color share the weights.

ally written following the *inverted pyramid* principles.

3 Modeling

3.1 Encoding the article

Based on the elements of Journalism Studies discussed above, we propose a simple architecture based on bidirectional conditional encoding (Augenstein et al., 2016) to encode an article split into n sentences.

Each sentence S_i is first converted into its embedding representation $E_{S_i} \in \mathbb{R}^{e \times s_i}$, where e is the embedding size and s_i is the length of the i_{th} tokenized sentence. Then, we encode the article using BiLSTM$_A$, a Bidirectional LSTM which reads the article sentence by sentence in backward order, initializing the first states of its forward and backward components with the last states it has produced after processing the previous sentence (Figure 3).

Notice that we process the article from the bottom to the top, as we assume the most salient information to be concentrate in the *lead*. By considering an article as an ensemble of sentences which are separately encoded conditioned to their preceding ones, we can model the relationship of each sentence with respect to the others and, at the same time, reduce the number of parameters.

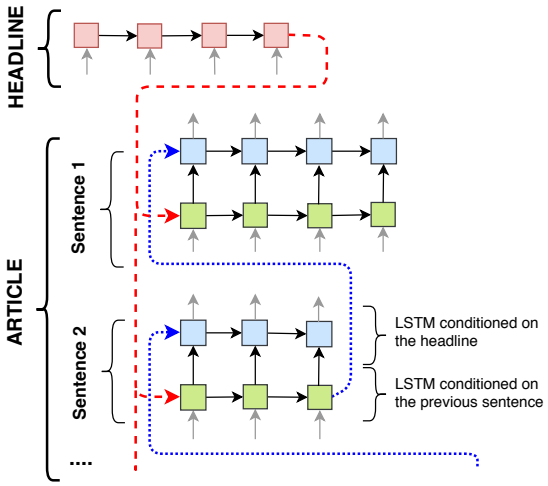

Figure 4: Detail of the forward component of the double-conditional encoding architecture (best seen in color). Dotted arrows represent conditional encoding and networks with the same color share the weights. The system reads an article from the last sentence to the first, processing each sentence twice: first conditioning on the headline, then conditioning on the previous sentence. Due to lack of space, only the first two sentences of the article are represented.

3.2 Encoding the relationship between the headline and the article

After having encoded the article, we model its relationship with the headline. As shown in Figure 2, we expect single sentences to express a potentially different stance with respect to the headline, while some sentences - especially in the *body* and the *tail* - can be irrelevant. For this reason, we separately evaluate the relationship of each sentence, conditioned on the previous sentence(s), with the headline. In this paper, we consider two approaches:

Double-conditional Encoding. As a first method, we modeled the relationship between the headline and the article using conditional encoding.

First, the headline is encoded using a bidirectional LSTM. Then, we separately process each sentence of the article with BiLSTM$_H$, a BiLSTM conditioned on the last states of the BiLSTM which processed the headline. We finally stack BiLSTM$_A$ on top of BiLSTM$_H$. In this way, we obtain a matrix $\overline{H}_{S_i} \in \mathbb{R}^{l \times s_i}$ for each sentence S_i.

Following Wang et al. (2018), we notate this as:

$$H_{S_i} = \text{Bi-LSTM}_H(E_{S_i}) \quad \forall i \in \{1, ..., n\} \quad (1)$$

$$\overline{H}_{S_i} = \text{Bi-LSTM}_A(H_{S_i}) \quad \forall i \in \{1, ..., n\} \quad (2)$$

This process is shown in Figure 4. In this way, we read each sentence S_i, which is encoded in a headline-specific manner, conditioning on the previous sentence(s). Clearly, it could have been possible to obtain a hidden representation for each sentence by first conditioning on the previous sentence(s), and then on the headline. Results of preliminary experiments, however, showed worse results for this variant, suggesting that having the conditioning on the previous sentence(s) nearer to the decoder is beneficial for the cross-level stance detection task.

Co-matching Attention. We also explored the use of attention in order to connect the headline $H_H \in \mathbb{R}^{l \times c}$, encoded using a BiLSTM layer, with the article's sentences $H_{S_1} ... H_{S_n}$, embedded as explained in Subsection 3.1. Inspired by the architecture proposed by Wang et al. (2018) for multi-choice reading comprehension, we obtain a matrix $\overline{H}_{S_i}$, attentively read with respect to the headline, for each sentence at position $i \in \{1, ..., n\}$ as follows: we first obtain an aggregated representation of the headline and the i_{th} sentence $\overline{H}_{S_i} \in \mathbb{R}^{l \times S_i}$ (Eq 4), obtained by dot product of H_H with the attention weights $A_i \in \mathbb{R}^{c \times s_i}$ (Eq 3); then, we obtain co-matching states of each sentence with $\overline{H}_{H_i}$ using Eq 5:

$$A_i = \text{softmax}((W_h H_H + b_h))^\top H_{S_i} \quad (3)$$

$$\overline{H}_{H_i} = H_H A_i \quad (4)$$

$$\overline{H}_{S_i} = \text{ReLU}(W_s \begin{bmatrix} \overline{H}_{Hi} \ominus H_{S_i} \\ \overline{H}_{Hi} \otimes H_{S_i} \end{bmatrix} + b_s) \quad (5)$$

where $W_h \in \mathbb{R}^{l \times l}$, $W_s \in \mathbb{R}^{l \times 2l}$, $b_h \in \mathbb{R}^l$ and $b_s \in \mathbb{R}^{2l}$ are the parameters to learn. As in Wang et al. (2018), we use the element-wise subtraction $\ominus$ and multiplication $\otimes$ to build matching representations of the headline.

Self-attention. After encoding of the relationship between the headline and the article, we employ a similar self-attention mechanism as in Yang et al. (2016) in order to soft-select the most relevant elements of the encoded sentence. Given the sequence of vectors $\{h_1, ..., h_S\}$ in $\overline{H}_{S_i}$, obtained with the double-conditional encoding or the co-matching attention approaches described above, the final vector representation of the i_{th} sentence S_i is obtained as follows:

$$u_{it} = \tanh(W_s h_{it} + b_s) \quad (6)$$

$$\alpha_{it} = \exp \frac{u_{it}^\top u_s}{\sum_t u_{it}^\top u_s} \quad (7)$$

$$s_i = \sum_t \alpha_t h_{it} \quad (8)$$

where the hidden representation of the word at position t, u_{it}, is obtained though a one-layer MLP (Eq 6). The normalized attention matrix α_t is then obtained though a softmax operation (Eq 7). Finally, s_i is computed by a weighted sum of all hidden states h_t with the weight matrix α_t (Eq 8).

3.3 Decoding

Following the *inverted pyramid* principles, according to which the most relevant information is concentrated at the beginning of the article, we aggregate the sentence vector representations $\{s_1, ..., s_n\}$ using a backward LSTM. The final prediction $\hat{y}$ is finally obtained with a softmax operation over the tagset.

4 Experimental Setup

4.1 Data and Preprocessing

We downloaded the FNC-1 corpus from the challenge website[2]. As we wanted to concentrate on the cross-level stance detection sub-task, we only considered *related* (AGR, DSC and DSC) samples, discarding the noisy UNR samples, which would constitute the output of the *tracking* step. The distribution of related samples is also very unbalanced, with the DSC class constituting more than a half of the subset and the DSG samples accounting for only 7.5% of the related samples, as shown in Table 1.

	samples	AGR	DSG	DSC	UNR
all	75,385	7.4%	2.0%	17.7%	72.8%
REL	20,491	27.2%	7.5%	65.2%	-

Table 1: Label distribution for the FNC-1 dataset, considering all classes, or only the related samples.

As discussed in the Introduction, the cross-level stance detection task is characterized by an asymmetry in length in the input: on average, headlines are 12.40 tokens long, while articles span from 4 up to 4788 tokens, with an average length

[2] https://github.com/FakeNewsChallenge/

	headline	entire article	sentence
avg no. tokens	12.40	417.69	30.88

Table 2: Asymmetry in length between headlines and articles in the FNC-1 corpus.

of 417.69 tokens. An article, however, presents a compositional internal structure, as it can be divided into smaller elements. We used the NLTK sentence tokenizer[3] to split articles into sentences, obtaining an average number of 11.97 sentences per article. On average, sentences are 30.88 tokens long, as reported in Table 2.

4.2 Baseline

As a baseline, we implemented the *Athena* model proposed by Hanselowski et al. (2017), which scored second in the FNC-1. We did not use the first-ranked system, as it is an ensemble model, nor the modification to *Athena* proposed in Hanselowski et al. (2018), as the new feature set and the BiLSTM layer did not significantly improve the performance of the original model. The model consists in a 7-layers deep MLP, with varying number of units, followed by a softmax. Input is presented as a large matrix of concatenated features, some of which separately encode the headline or the body:

- Presence of *refuting* and *polarity* words
- Tf-idf-weighted BoW unigram features, considering a vocabulary of 5000 entries.

while others jointly consider the headline/body:

- Word overlap between headline and article.
- Count of headline's token and ngrams which appear in the article.
- Cosine similarity of the embeddings of nouns and verbs of the headline and the article.

Moreover, they use topic models based on non-negative matrix factorization, latent Dirichlet allocation, and latent semantic indexing. This results in a final set of 10 features. In this way, the asymmetry in length between is solved by compressing both the headline and the article into two fixed-sized vectors of the same size.

The same hyperparameters as in Hanselowski et al. (2017) have been used for the implementation

Max headline length (in tokens)	15
Max sentence length (in tokens)	35
Max number of sentences for article	7
Word embedding size	300
BiLSTM cell size	2×128
Embedding dropout	0.1
BiLSTM dropout	0.3
Dense layer dropout	0.2
Epochs	70
Batch size	32
Optimizer	Adam
Learning rate	0.001
Experiments using additional input channels	
Max word length (in characters)	35
Character embedding size	30
Character BiLSTM cell size	2×64
Character BiLSTM dropout	0.2
NE embedding size	30

Table 3: Hyperparameters configuration

of the model. For training, we downloaded the feature matrices which had been used in *Athena* best submission [4], taking only the indices corresponding to the related samples.

4.3 (Hyper-)Parameters

The high-level structure of the models has been implemented with Keras, while single layers have been written in Tensorflow. (Hyper-)parameters used for training, useful for experiments replication, are reported in Table 3. Concerning vocabulary creation, we included only words occurring more than 7 times. The embedding matrix has been initialized with *word2vec* embeddings[5], which performed better than other set of pre-trained embeddings according to some preliminary experiments. This can be partially explained as *word2vec* embeddings are trained on part of the Google News corpus, thus on the same domain as the FNC-1 dataset. OOV words have been zero-initialized. In order to avoid overfitting, we did not update word vectors during training.

4.4 Evaluation Metrics

As we are not considering the UNR samples, the FNC-1 score would not constitute a good metric, as it distinguishes between related and unrelated samples for scoring[6]. Following Zubiaga et

[3] https://www.nltk.org/_modules/nltk/tokenize.html

[4] https://drive.google.com/open?id=0B0-muIdcdTp7UWVyU0duSDRUd3c

[5] https://code.google.com/archive/p/word2vec/

[6] https://github.com/FakeNewsChallenge/fnc-1-baseline/blob/master/utils/score.py

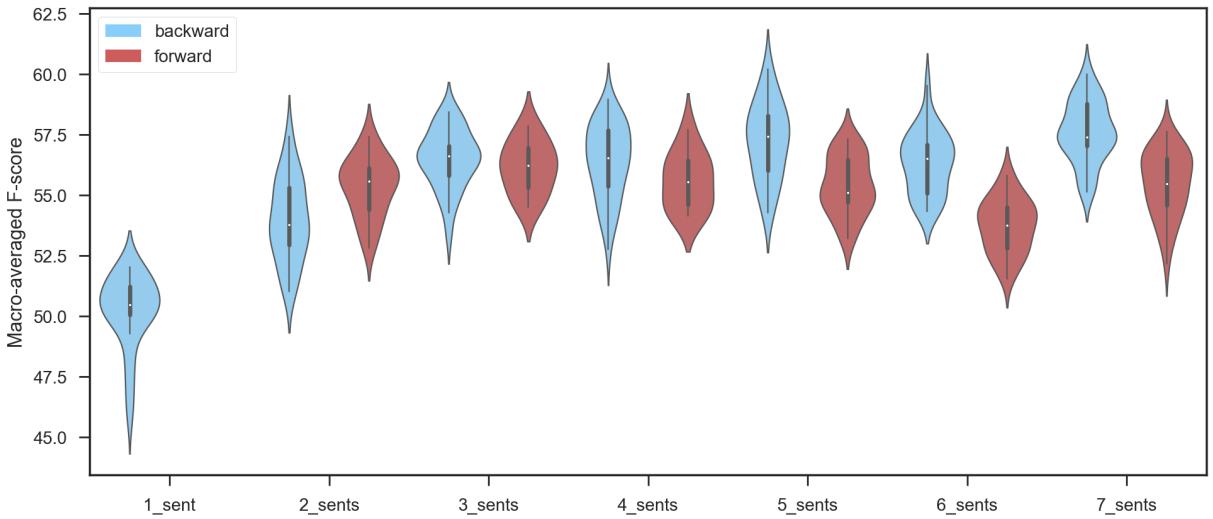

Figure 5: Performance of the co-matching encoder in term of macro-averaged F_1 score on the test set, considering the first n sentences of an article. Blue and red violins represent respectively backward and forward encoding of the considered sentences.

al. (2018) and Hanselowski et al. (2018), we use macro-average precision, recall and F_1 measure, which is less affected by the high class unbalance (Table 1). We also consider the accuracy with respect to the single AGR, DSG and DSC classes.

5 Results and Discussion

As shown in Table 4, both encoders described in Section 3 outperformed the baseline (line 1), despite having a considerably minor number of parameters. In particular, the feature-based model obtained a relatively good performance in classifying the very infrequent DSG labels, probably thanks to its large number of hand-engineered features. However, it shows some difficulties in discriminating between AGR and DSC samples. This is probably a consequence of the system flattening the entire article into a fixed-size vector: this inevitably causes the system to loose the subtle nuances in the argumentative structure of the news story, which allows for distinguishing between AGR and DSC samples, and to favor the most common DSC class. On the contrary, our architectures approach the asymmetry in length in the input by carefully encoding the articles as a hierarchical sequence of sentences, and by separately modeling their relative positions with respect to the headline. In this way, they are able to successfully discriminate between AGR and DSC samples.

In general, the encoder based on co-matching attention performed clearly better than the architecture based on double-conditional encoding (line 2 and 10), reaching a higher performance in all metrics but the classification of the DSC samples.

5.1 Modeling the *Inverted Pyramid*

In order to test our assumption that the great majority of the FNC-1 corpus were written following the *inverted pyramid* style, we took the co-matching attention model, which performed better than the double-conditionally encoded architecture, and progressively reducing the number of considered sentences. Moreover, we modified the article encoder (Subsection 3.1) in order to process the input sequence in forward and not in backward order. For each of these 13 settings[7], we run 10 simulations.

As the violin plots in Figure 5 show (blue violins), considering a reduced number sentences does not correlate with an overly big drop in performance, until a number of less than four sentences is taken. Below this threshold, the ability of the system to correctly classify the stance of the article is compromised. This can be explained with the inverted pyramid theory: until we consider a number of sentences sufficient in order to include the *lead* and part of the *body* of the article, the system can rely on a sufficient number of elements in order to discriminate its stance. On the contrary, if we consider only the very first sentences, the system can get confused, being exposed to only

[7] Specifically: 7 backward-encoded co-matching architectures (considering a number of sentences from 1 up to 7) and 6 forward-encoded co-matching architectures (considering a number of sentences from 2 up to 7).

	Model		anonymized input	accuracy AGR	accuracy DSG	accuracy DSC	macro-averaged P	macro-averaged R	macro-averaged F_1
1	Baseline		–	26.69	11.76	74.77	39.39	37.74	38.00
2	Double-conditional Encoding	–	no	68.84	9.61	**77.42**	52.50	51.25	49.81
3		–	yes	63.11	9.76	76.03	52.31	49.63	**51.77**
4		+ char	no	51.45	**23.96**	76.32	50.12	50.57	50.32
5		+ char	yes	59.64	16.93	77.64	53.27	**51.40**	51.38
6		+ ner	no	69.57	5.02	76.97	52.14	51.22	48.86
7		+ ner	yes	75.78	9.33	69.96	54.41	51.31	51.17
8		+ char + ner	no	62.11	13.20	77.11	52.83	50.80	50.42
9		+ char + ner	yes	**76.77***	12.34	67.47	**53.45**	49.85	50.56
10	Co-matching attention	–	no	69.57	**33.0***	74.91	**64.14***	**58.53***	**59.01***
11		–	yes	64.37	29.27	**78.94***	59.64	55.20	57.12
12		+ char	no	**74.46**	24.82	71.95	61.77	56.46	58.55
13		+ char	yes	70.52	15.06	76.39	63.88	55.23	56.01
14		+ ner	no	66.05	18.79	72.38	58.51	51.48	52.52
15		+ ner	yes	69.42	31.85	72.38	58.59	54.67	57.32
16		+ char + ner	no	63.95	20.95	77.76	57.32	53.10	53.90
17		+ char + ner	yes	67.26	11.05	76.99	57.19	51.84	54.85

Table 4: Results of experiments using double-conditional encoding or co-matching attention. Best results for each encoding type are shown in bold. Best results overall are indicated with an asterisk.

a portion of the (sometimes opposing) opinions expressed in the article. Interestingly, our system seems to be pretty robust to the noisy sentences which could be included when considering a higher number of sentences.

The assumption that most of the articles in the FNC-1 corpus are written following the *inverted pyramid* principles is further confirmed by the fact that, after the threshold of 4 considered sentences, simulations using forward encoding perform always consistently worse than using backwards encoding (the red violins in Figure 5). Reasonably, below this threshold, we do not observe a considerable difference in performance between backward and forward models.

5.2 Additional Experiments

5.2.1 Using additional Input Channels

To investigate the impact of features other than word embeddings, we consider two further input channels:

- **Named Entities (NE)** - NEs were obtained using the Stanford NE Recognizer (Finkel et al., 2005), resulting in a tagset of 13 labels.
- **Characters** - Each input word was split into characters. Only characters occurring more than 100 times in the training set were considered, obtaining a final vocabulary of 149 characters. As in Lample et al. (2016), in we concatenate the output of a BiLSTM run over the character sequence.

The output of each input channel is concatenated with the word embedding, and passed to the article encoder described in Section 3.1. Hyperparameters used for experiments are reported in Table 3.

5.2.2 Anonymizing the input

After manual analysis of the predictions, we suspected that some models could have spotted some correlations between certain Named Entities and a specific stance in the training set. Some of those correlations are well known and can be useful in veracity detection (Wang, 2017). In this paper, however, we wanted to train a model for stance detection only based on its language understanding, without counting on such possibly accidental correlations.

In order to avoid the systems to rely on chance correlations, which would not generalize on the test set, we modified the input sequences by substituting all input tokens labeled as `<PERSON>`, `<ORGANIZATION>` and `<LOCATION>` by the Stanford Named Entity Recognizer with the corresponding NE tags.

5.2.3 Results

Results of experiments concatenating the previously mentioned features to the word embedding input to both architectures are reported in Table 4 (even lines). In general, using NE embeddings alone with word embeddings was not beneficial for both models. Considering the architecture based on double-conditional encoding, using both

characters and NE features actually lead to (sometimes small) improvements in almost all considered evaluation metrics. Moving to the architecture using co-matching attention, adding characters or NE embeddings, even in combination, caused a considerable drop in all evaluation metrics, apart on some single label classification (as the AGR class).

As shown in Table 4 (odd lines), anonymizing the input was always useful for the architecture using double-conditional encoding, resulting in a consistently higher macro-averaged F_1 score. Considering the architecture based on co-matching attention, however, anonymizing the input was beneficial only for architectures leveraging NE tags (only with word embeddings, or in combination with character embeddings), which were also the ones showing the highest drop in performance with respect to the model using only word embeddings.

The best performance according to macro-averaged precision, recall and F_1 score is obtained using the co-matching attention model leveraging only word embeddings. The high performance of this model is mainly due to its ability to discriminate the very unfrequent DSG class.

6 Conclusions

We proposed two simple architectures for Cross-Level Stance Detection, which were carefully designed to model the internal structure of a news article and its relations with a claim. Results show that our "journalistically"-motivated approach can beat a strong feature-based baseline, without relying on any language-specific resources other than word embeddings. This indicates that an interdisciplinary dialogue between Natural Language Processing and Journalism Studies can be very fruitful for fighting Fake News.

In future work, we aim to put together the different stages of the FND pipeline. Following the work of Kochkina et al. (2018) for RV, it could be interesting to compare a sequential approach to separately solve each step of the pipeline in isolation, with a joint multi-task system. The generalizability of the models trained on the FND pipeline to other domains could be tested with the recently released ARC corpus (Hanselowski et al., 2018), which has similar statistical characteristics as the FNC-1 corpus.

Acknowledgments

Special thanks to Chiara Severgnini, journalist at *7* (the weekly magazine supplement of *Corriere della Sera*) for her comments on Section 2.2. The first author (CC) would like to thank the Siemens Machine Intelligence Group (CT RDA BAM MIC-DE, Munich) and the NERC DREAM CDT (grant no. 1945246) for partially funding this work. The third author (NC) is grateful for support from the UK EPSRC (grant no. EP/MOO5089/1). We thank the anonymous reviewers of this paper for their efforts and for the constructive comments and suggestions.

References

Ahmet Aker, Leon Derczynski, and Kalina Bontcheva. 2017. Simple open stance classification for rumour analysis. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017, Varna, Bulgaria, September 2 - 8, 2017*, pages 31–39.

Isabelle Augenstein, Tim Rocktäschel, Andreas Vlachos, and Kalina Bontcheva. 2016. Stance detection with bidirectional conditional encoding. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 876–885.

Sean Baird, Doug Sibley, and Yuxi Pan. 2017. Talos targets disinformation with fake news challenge victory. https://blog.talosintelligence.com/2017/06/.

Leon Derczynski, Kalina Bontcheva, Maria Liakata, Rob Procter, Geraldine Wong Sak Hoi, and Arkaitz Zubiaga. 2017. Semeval-2017 task 8: Rumoureval: Determining rumour veracity and support for rumours. In *Proceedings of the 11th International Workshop on Semantic Evaluation, SemEval@ACL 2017, Vancouver, Canada, August 3-4, 2017*, pages 69–76.

William Ferreira and Andreas Vlachos. 2016. Emergent: a novel data-set for stance classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pages 1163–1168.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 363–370. Association for Computational Linguistics.

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

Andreas Hanselowski, Avinesh P. V. S., Benjamin Schiller, Felix Caspelherr, Debanjan Chaudhuri, Christian M. Meyer, and Iryna Gurevych. 2018. A retrospective analysis of the fake news challenge stance-detection task. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 1859–1874.

David Jurgens, Mohammad Taher Pilehvar, and Roberto Navigli. 2014. Semeval-2014 task 3: Cross-level semantic similarity. In *Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014)*, pages 17–26.

Elena Kochkina, Maria Liakata, and Arkaitz Zubiaga. 2018. All-in-one: Multi-task learning for rumour verification. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 3402–3413.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 260–270.

Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. 2016. Semeval-2016 task 6: Detecting stance in tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 31–41.

Saif M Mohammad, Parinaz Sobhani, and Svetlana Kiritchenko. 2017. Stance and sentiment in tweets. *ACM Transactions on Internet Technology (TOIT)*, 17(3):26.

Dean Pomerleau and Delip Rao. 2017. Fake news challenge. http://www.fakenewschallenge.org/.

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

Christopher Scanlan. 1999. *Reporting and writing: Basics for the 21st century*. Oxford University Press.

Sun Technical Publications. 2003. *Read Me First!: A Style Guide for the Computer Industry*. Prentice Hall Professional.

James Thorne and Andreas Vlachos. 2018. Automated fact checking: Task formulations, methods and future directions. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 3346–3359.

Shuohang Wang, Mo Yu, Jing Jiang, and Shiyu Chang. 2018. A co-matching model for multi-choice reading comprehension. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 2: Short Papers*, pages 746–751.

William Yang Wang. 2017. "liar, liar pants on fire": A new benchmark dataset for fake news detection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 2: Short Papers*, pages 422–426.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489.

Arkaitz Zubiaga, Ahmet Aker, Kalina Bontcheva, Maria Liakata, and Rob Procter. 2018. Detection and resolution of rumours in social media: A survey. *ACM Computing Surveys (CSUR)*, 51(2):32.

Arkaitz Zubiaga, Elena Kochkina, Maria Liakata, Rob Procter, and Michal Lukasik. 2016a. 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 the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 2438–2448.

Arkaitz Zubiaga, Maria Liakata, Rob Procter, Geraldine Wong Sak Hoi, and Peter Tolmie. 2016b. Analysing how people orient to and spread rumours in social media by looking at conversational threads. *PloS one*, 11(3):e0150989.