HeadlineStanceChecker: Exploiting summarization to detect headline disinformation
RobierSepúlveda-Torres *, Marta Vicente, Estela Saquete, Elena Lloret, Manuel Palomar
Department of Software and Computing Systems, University of Alicante, Apdo. de Correos 99 E-03080, Alicante, Spain

A B S T R A C T
The headline of a news article is designed to succinctly summarize its content, providing the reader with a clear understanding of the news item. Unfortunately, in the post-truth era, headlines are more focused on attracting the reader’s attention for ideological or commercial reasons, thus leading to mis- or disinformation through false or distorted headlines. One way of combating this, although a challenging task, is by determining the relation between the headline and the body text to establish the stance. Hence, to contribute to the detection of mis- and disinformation, this paper proposes an approach (HeadlineStanceChecker) that determines the stance of a headline with respect to the body text to which it is associated. The novelty rests on the use of a two-stage classification architecture that uses summarization techniques to shape the input for both classifiers instead of directly passing the full news body text, thereby reducing the amount of information to be processed while keeping important information. Specifically, summarization is done through Positional Language Models leveraging on semantic resources to identify salient information in the body text that is then compared to its corresponding headline. The results obtained show that our approach achieves 94.31% accuracy for the overall classification and the best FNC-1 relative score compared with the state of the art. It is especially remarkable that the system, which uses only the relevant information provided by the automatic summaries instead of the whole text, is able to classify the different stance categories with very competitive results, especially in the discuss stance between the headline and the news body text. It can be concluded that using automatic extractive summaries as input of our approach together with the two-stage architecture is an appropriate solution to the problem.

© 2021 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

1. Introduction
Nowadays, disinformation and misinformation are two major problems that are increasing at great velocity [1] in pace with the exponential growth of information on the web and the need for robust verification methods. If handling this information overload is an arduous and complex task for both humans and machines, verifying its veracity has become a daunting yet unavoidable challenge. Both terms, misinformation and disinformation, allude to the inaccuracy and lack of veracity of certain information; however, while in the first case the delusion can be caused unintentionally, the latter actually seeks to deceive or misdirect deliberately [2]. In either case, they represent a type of phenomenon that, in the domain of digital news, can easily result in a massive confusion about the real facts, spreading on a viral scale. This is actually what the New York Times meant when they referred to a “Fake news” piece as a “made up story with the intention to deceive, often with monetary gain as a motive” [3].

The ideological and economic interests that potentially gain from this “information disorder” are the drivers of fake news. These interests aim to manipulate social opinion and reinforce preconceived opinions, thereby making people focus on thinking or acting in a specific way by, most of the time, appealing to their emotions rather than presenting the facts. This trend has even prompted the advent and consolidation of a new term, “post-truth”, which, according to the Cambridge Dictionary, refers to “a situation in which people are more likely to accept an argument based on their emotions and beliefs, rather than one based on facts”. For instance, this disturbing phenomenon played an important role in President Trump’s election campaign 2016 [4] and the Brexit referendum 2016 [5]. In the same way, business and commercial interests fabricate fake news to generate income through clickbait and misleading information. For instance, the National Report website, Disinformedia [6] or Victory

https://doi.org/10.1016/j.websem.2021.100660
1570-8268/© 2021 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).
In this context, headlines can be classified into two classes [14]: through erroneous/false facts or headline/body dissonance [13].

Following this approach, great attention and effort has been focused on the analysis and study of one of the most essential elements of a news item, its headline; in some cases focusing on the relationship between the body of the article and the headline, and in others considering the constitution of the headline itself. Headlines are fundamental parts of news stories, they summarize the article so that the reader clearly understands the content of the news story [9]. Nevertheless, the headline acts also as the prelude to the complete news story, and it should be written as an invitation for the reader to discover the full piece. A headline is therefore expected to be as effective as possible, without losing accuracy or becoming misleading [10].

In the scenario we have outlined, where the information stream is permanently growing and filtering content can be overwhelming, the role of headlines is crucial. On the one hand, an appropriate headline can help us to identify the content of most interest to us, but on the other hand, and due to this data deluge, it can be tempting to read only the headlines and share the news feed without having read the entire story. Consequently, stories can often go viral because of an attractive headline despite the lack of true information in the body text. This phenomena manipulates public opinion and affects the credibility of social media [11,12]. In particular, the research conducted in [11] found that 59% of the URLs mentioned on Twitter were not clicked at all. This suggests that people are more willing to share an article than access and read it, so they directly read and share the headline (and link), without making the effort to go deeper and check it. Considering this, the headline of a news article should faithfully summarize the body text, without including deception or misinformation, in order to maintain accuracy and veracity of the entire article.

Unfortunately, in practice, headlines in digital media tend to be more focused on attracting the reader’s attention (with little regard for accuracy) thus leading to mis- or disinformation through erroneous/false facts or headline/body dissonance [13]. In this context, headlines can be classified into two classes [14]:

- **Clickbait headlines**: Clickbait refers to content whose main purpose is to attract attention and encourage visitors to click on a link to a particular web page with the purpose of monetizing the “views” through advertising revenue (the more clicks, the more money earned). This type of headline is often ambiguous and exhibits a particular writing style to directly exploit human curiosity, for instance by using exclamatory or interrogative headlines that urge audiences to click on the link to discover the missing information [14]. Typically, clickbait headlines are spread on social media in the form of short teaser messages that may read like the following cited examples:
  - “Man tries to hug a wild lion, You won’t believe what happens next!”^2
  - “The first lady of swearing! How a ten-year-old Michelle Obama lost out on a ‘best camper’ award because she wouldn’t stop cursing”^3

Existent methods for automatically detecting clickbait headlines usually treat the task as a classification problem (clickbait/non-clickbait), and exclusively focus on the headline (its writing style or structure) rather than considering the content of the news itself [13,15].

- **Misleading headlines**: Headlines thus classified significantly misrepresent the findings reported in the news article [16], by exaggerating or distorting the facts described in the news article. The reader can only discover the inconsistencies after reading the news body text [14]. Although in the literature these headlines are sometimes referred to as incongruent headlines, in this work we will refer to them as misleading headlines since the term represents a more comprehensive concept.

Some important nuances that are part of the news body text are missing in the headline, causing the reader to come to the wrong conclusion. In contrast to clickbait headlines, the language used does not necessarily incite the reader to click on it, but it is designed to trigger emotion or excitement [16]. Examples of misleading headlines are shown below (also reported in [13,16], respectively):

- “Ebola in the air? A nightmare that could happen”^4
- “Air pollution now leading cause of lung cancer”^5

In order to automatically detect misleading headlines, the news body text must be analyzed to extract the evidence from which the headline has been derived, thereby detecting the headline/body text discrepancy in the absence of such evidence. The task of identifying the relation between a headline and the news article it refers to has been addressed in recent research (see Section 2) as a stance detection problem. This type of approach involves estimating the relative perspective, namely the stance, of one piece of text, such as a claim or a news article, towards another, for example, a topic, a statement or a headline [17].

In the context of headline/body text dissonance, the main objective of this research is to propose an approach that relies on semantics and deep learning techniques to automatically determine the stance of the headline with respect to its body text. By this means, the problem of misleading headlines can be addressed. The approach is hereafter referred to as HeadlineStanceChecker. Given a news headline and its corresponding body text, our proposal assigns the headline one of these four classes (unrelated, agree, disagree or discuss), indicating the headline stance, and validating and checking whether the headline is faithfully reflecting the information provided in the news article.

The most interesting aspect of solving misleading headline detection as a stance detection task is that it is not only focused on determining whether or not a headline is consistent with its body text, but it is also a fine-grained classification that determines the type of dissonance involved.

We explore the treatment of this task as a two-stage neural classification problem in which only the essential information of the news item is processed, rather than the whole news item. We therefore use the summaries because besides containing the key information of the news story, we hypothesize that the abridged version will not only increase task efficiency but also that of the neural models. Neural models can have a negative impact on efficiency when processing long texts, so previous studies either used the first sentence of the text [18] or a specific fragment [19] to combat this problem. Therefore, the use of text summarization,
which, to the best of our knowledge, has not been previously exploited for stance detection, could be beneficial in this context.

To summarize, the main novelties of HeadlineStanceChecker are twofold:

- the adoption of a divide-and-conquer strategy by proposing a two-stage neural classifier for performing the headline stance task; and
- the use of summarization techniques based on Positional Language Models (PLM). These models leverage semantic knowledge to detect the evidences and essential information within the news article so as to generate automatic summaries that will be used as substitutes of the full body text for the whole classification process. We expect this approach to be more efficient in dealing with the headline stance classification task.

The paper is structured as follows: Section 2 presents the related work regarding misleading headlines, as well as a brief review of the state of the art in text summarization; Section 3 presents our proposed architecture for HeadlineStanceChecker (explaining each of the stages in detail); Section 4 describes the experiments carried out and the evaluation environment; Section 5 reports and discusses the results of the proposed approach (comparing them to other competitive systems); and, Section 6 presents conclusions and outlines the main direction for future work.

2. Related work

HeadlineStanceChecker has been conceived as an automatic method to classify a news story in terms of the relation between its body text and its headline. The main motivation of developing such approach is to provide a tool that helps both professionals and readers to identify misleading or fraudulent media and information, thus preventing harmful consequences.

Fake news research has opened up an immense field of work that encompasses multiple areas and approaches. Both linguistic and non-linguistic aspects are being studied, so that elements as diverse as image verification, analysis of reputation and authorship, or the network dissemination patterns of misleading stories fall within its field of interest. For brevity, we focused on the research directly related to our proposal, but comprehensive studies can be found in [8,20,21].

Therefore, in this section, first we present an overview of recent work done in Stance Detection and, next, an in-depth review of the existing detection strategies for misleading headlines is conducted. Finally, given that one of the novelties of the paper is using summarization techniques leveraging essential information to characterize headlines, a brief review is presented of the state of the art in text summarization.

- Stance Detection Overview

From an overall perspective, stance detection can be defined as the task of identifying the perspective of an author or text against a given target in the form of one topic, claim, headline or even a personality [22,23]. Hence, there exists a tuple of elements (the text on the one side, the target on the other side) and a classification process shaped to determine how the former stands towards the second: does the text support the topic? does it disagrees with the claim? The names of the classes (e.g. support, against, for or neutral) depend on the precise problem. The task, which concerns a diverse range of domains, is studied in such varied areas as political debates [24,25], student essays [26], online forum debates [27] or even internal company discussions [28,29].

A great deal of work in opinion mining has been devoted to detect the stance of tweets or other types of short texts as rumors [30] or microblogging statements. Examples of targets posed in the available datasets could be “Hillary Clinton” for personality, “Atheism” as a particular topic or the claim “E-cigarettes are safer than normal cigarettes”. Shared tasks offering such datasets and fostering the research on the matter have arisen in different languages. SemEval-2016 posed the sub-task for detecting stance in tweets [31], providing around 5 thousand tweets in English covering five commonly known topics. The task has inspired numerous approaches that develop either traditional proposals (e.g. K nearest neighbor [32], Support Vector Machines [33] or latent features provided by methods such as Latent Dirichlet Allocation [34]); or those inspired by neural network frameworks, by using, for example, bidirectional conditional encoding [35], bidirectional Long-Short-Term Memory neural networks [36] or Attention based Convolutional Neural Networks [37]. Besides, there are available public datasets that support the development of new interesting work, such as the Multi Perspective Consumer Health Qc dataset [38] dedicated to detecting the stance of sentences collected from quality articles towards five different claims (e.g., “Sun exposure causes skin cancer”). In [23], an in-depth study on different approaches to the two tasks mentioned above can be found. Regarding languages other than English, the necessity for well-annotated data led to the proliferation of both annotation efforts and shared tasks aimed to advance research, such as StanceCat, presented at IberEval 2017 as a stance detection task for tweets in Spanish and Catalan [39], a proposal and a dataset of short messages in Russian internet forums [40], or even projects combining a larger number of languages (French, Italian, Spanish, English) [41,42].

In contrast to such approaches, research on stance detection based on longer documents, as in the current scenario, faces different challenges. Dealing with discourse, as a coherent and cohesive set of sentences, adds a certain complexity not present when processing shorter utterances. Within the discourse, an argument may develop in such a way that some sentences may show support for the claim, while others may seem to deny it, and only by considering the document as a whole can the stance be effectively identified. It is in this context that HeadlineStanceChecker has been developed, and next, we introduce the related work concerning the specific task.

- Misleading headlines

The task of detecting misleading headlines for the present research involves classifying the stance of the article body with respect to the claim made in the headline into one of the following four classes: (a) agrees (agreement between body text and headline); (b) disagrees (disagreement between body text and headline); (c) discusses (same topic discussed in body text and headline, but no position taken); and, (d) unrelated (different topic discussed in body text and headline).

This task (headline stance detection) quickly emerged in the context of fake news analysis, triggered by a demand for new technologies to prevent and combat the phenomenon, together with an increase in the availability of annotated corpora [8]. In this context, research challenges and competitions were proposed. The most recent and important ones are next reviewed in detail.

The Fake News Challenge[^6] (FNC-1) [43] was created using Emergent dataset [17] as a starting point (this dataset has been extracted from the Emergent Project [44], a rumor

debunking project). FNC-1 aims to compile a gold standard to explore Artificial Intelligence technologies, especially ML and Natural Language Processing (NLP), applied to detection of fake news. To carry out this macro-challenge, the organizers decided to start with stance detection. In this case, the FNC-1 dataset was released, with around 75,000 instances that were classified as follows: agree, disagree, discuss and unrelated.

For example, given the headline “Robert Plant Ripped up $800M Led Zeppelin Reunion Contract”, the following fragments1 would illustrate the different classes mentioned, according to the gold-standard annotations in the FNC-1 dataset:

- **Agrees**: The body text agrees with the headline. Example evidence: “[...] Led Zeppelin’s Robert Plant turned down 500 MILLION pounds to reform supergroup”.
- **Disagrees**: The body text disagrees with the headline. Example evidence: “[...] No, Robert Plant did not rip up a $800 million deal to get Led Zeppelin back together”.
- **Discusses**: The body text discusses the same topic as the headline, but does not take a position. Example evidence: “[...] Robert Plant reportedly tore up an $800 million Led Zeppelin reunion deal”.
- **Unrelated**: The body text is not related with the headline. Example evidence: “[...] Richard Branson’s Virgin Galactic is set to launch SpaceShipTwo today”.

The FNC-1 competition received a total of 200 submissions achieving relative scores7 of around 82% in the best ranked submissions. The organization proposed a simple baseline using hand-coded features and a gradient boosting classifier, available at Github.9 The three best systems in this competition were Talos [45], Athene system [46] and UCLMR [47] in this order. Talos [45] applied a one-dimensional convolutional neural networks (CNN) on the headline and body text, represented at the word level using Google News pretrained vectors. The output of this CNN is then sent to a multi-layer perceptron (MLP) with 4-class output: agree, disagree, discuss, and unrelated, and trained end-to-end. Using this combination CNN-MLP, the system outperformed all the submissions and achieved the first position in the FNC-1 challenge.

Recently, other works used the FNC-1 for their experiments and the performance obtained in the competition improved. For instance, [48] addressed the problem proposing a hierarchical representation of the classes, which combines agree, disagree and discuss in a new related class. A two-layer neural network is learning from this hierarchical representation of classes and a weighted accuracy of 88.15% is obtained with their proposal. Furthermore, [49] constructed a stance detection model by performing transfer learning on a RoBERTa deep bidirectional transformer language model by taking advantage of bidirectional cross-attention between claim-article pairs via pair encoding with self-attention. They reported a weighted accuracy of 90.01%.

Outside the FNC-1 Challenge and dataset, there is other research that also addresses the stance detection tasks, determining the relation of a news headline with its body text. Some authors extracted key quotes [50] or claims [51] to facilitate the detection. There is also work related to argument mining analysis, in which the headline represents an argument that is not supported by claims in the text. Moreover, in addition to using argument mining for solving stance detection, this problem could benefit from other tasks which detect semantic relations within the text, such as contradiction [52], contrast [53] and entailment [54].

**Text Summarization**

Previous research in Text Summarization has been shown to have a positive impact on society since the use of summaries has been beneficial in different areas, such as education (where summaries are used to support reading comprehension tasks [55–58]) business, by producing, for instance, an automatic summary of event logs to help analysts [59], or health, regardless of whether the summaries were created manually [60,61], or automatically [62]. This is partly due to the capability of summarization methods to identify the most relevant information of a document, and condense it into a new text, thereby helping to reduce time and resources when it comes to manage large amounts of data. These methods have proven to be effective when integrated as an intermediate component of more complex systems. The journalism field, and specifically the news domain, has been one of the most representative areas in which summarization has traditionally focused from the outset, partly thanks to the development of appropriate corpora (e.g. DUC, Gigaword, CNN/DailyMail) [63], and the wide range of techniques and approaches to help digest this type of information [64–67].

Besides the various summarization types that have been developed for this domain (single-document, multi-document, extractive, abstractive, generic, topic-oriented, etc.), there is a significant amount of research on the task of headline generation using summarization techniques [68–70], and more recently using Deep Learning [71–73]. However, none of them have exploited either the headline or the summarization techniques as an intermediate stage to further extract the semantic relationship between the headline and the news body text, and detect possible incongruities to fight against the fake news problem.

Although summarization has been used for fake news detection [74,75], as well as in the context of online discussions and social media to detect whether the author of a comment is in favor of or against a given target (e.g. entity or topic) [76,77], to the best of our knowledge, summarization has not been directly applied to the stance detection problem of misleading headlines, as proposed in this study.

Summarization was mentioned as a potential effective methodology for dealing with the problem of incongruent headlines in [78], but from a different perspective, which involved using headline generation to create a new headline that could be then compared to the existing headline by measuring the distance between them. More recently, an updated comprehensive survey concerning the stance detection task [79] shows that there is a lack of research where summarization is applied to this task, although a new type of summarization, called stance summarization, is outlined. However, stance summarization involves the generation of a new type of summary which includes a stance, but it is not comparable to the approach presented in this paper as the summaries are not incorporated into the stance detection process.

In another survey, conducted by [80], the authors compile the available information regarding existing research addressing this problem, and only the work of [81] summarized the news body into a single sentence to be compared to the given claim and determine its overall veracity, an approach which aligns

---

7 Examples extracted from Fake News Challenge website fakenewschallenge.org.
8 Measure score used in the Fake News Challenge competition.
with that suggested in [78] as aforementioned. By contrast, our research goes beyond summarizing the whole document into just one sentence, and provides a summary that could be acted as a substitute of the whole body text.

The HeadlineStanceChecker proposal is based on the fact that semantic information and discourse structure are captured through PLMs which, in turn, are exploited as a summarization technique. PLMs allow key spots and relevant information to be located in the news body text, and they are then used to create a summary of the news. By this means, the news article is reduced to its essential information, which is then compared to its headline. Our proposed model to detect misleading headlines, by relying on their stance towards the article’s content, directly uses this summary of the news instead of the whole news body text, enabling a more accurate comparison to its headline.

3. HeadlineStanceChecker architecture

The HeadlineStanceChecker approach involves two stages (see Fig. 1), thus addressing the task as a two-level classification problem. The first level corresponds to a Relatedness Stage, while the second corresponds to a Stance Stage. An additional novelty is the use of summaries generated in the first stage for the whole process instead of the full body text (i.e., the Relatedness Stage).

In this manner, given the inputs, namely the candidate headline and the news article body text, a summary of the news body will be created in the Relatedness Stage to later determine the headline’s stance regarding the news article as either related or unrelated. Afterwards, in the Stance Stage, the examples classified as related in the previous stage, are further classified into three possible values: agree, disagree, or discuss.

A more detailed description of both stages and the different modules involved in performing the stance classification is provided here-under.

3.1. Relatedness stage

The Relatedness Stage will determine whether the headline is classified as related or unrelated with respect to the body text of the news article. The inputs of this stage are both the text body and the headline, resulting in a binary classification. The outputs of this stage are:

- The headlines classified as related or unrelated.
- The summary of the news content, obtained in a relevant information detection module.

To produce the above outputs, three modules are proposed: (i) relevant information detection; (ii) relatedness feature extraction; and, (iii) relatedness classification.

3.1.1. Relevant information detection module

The relevant information detection module aims to create a summary revealing the important information of the input news article in relation to its headline.

The task of summarization has generally been carried out from a statistical perspective that only considers the elements of the text with no regard to their structure (or in those cases where the structure is taken into account, it is already known beforehand, such as in the case of scientific articles). Conversely, PLMs represent a type of statistical language model that allows information to be considered by taking into account both the relevant elements of the text and also their location in the document. They define a dynamic method for detecting key aspects of the text independently of the domain and textual genre to which it is related. Besides, PLMs have proved valuable in other areas such as information retrieval [82] and language generation [83].

From the semantic perspective of the text, its essence can be more effectively captured and synthesized by considering the document not as a mere sequence of sentences, but as a coherent and cohesive source of meaning, traversed by semantically related entities and actions. Considering this, we chose PLMs as the cornerstone of our module given that they can be configured to include the identification of named entities within the story, together with the representation of the words as synsets (sets of synonyms accounted under an identifier), allowing a further abstractive step on the basis of Wordnet [84], a hierarchical database of semantic relations. Consequently, PLMs help to incorporate both the semantics derived from the relevant lexical units together with the meaning derived from the text as coherent discourse. Previous studies demonstrated that PLMs were suitable for summarization tasks [85] and, moreover, a preliminary research was conducted analyzing and comparing different summarization methods for the stance detection task (including extractive, abstractive and hybrid ones) also showing that PLM-based summarization yielded the most stable results [86].
PLM essentials. Fundamentally, the PLMs state that for every position \( i \) within a document \( D \) it is possible to calculate a score for each element \( w \) that belongs to the document’s vocabulary. The decision as to the kind of elements that compose the vocabulary is made when designing the module. The calculated score displays the relevance of each element \( w \) in every precise position \( i \), based on its distance to other occurrences of the same element throughout the document. The score is higher when the neighbor element is closer within a scope to compute the value that goes beyond the sentence limits, taking into account the whole document. In order to express the distance to the occurrence of the entity within a document, a propagation function \( f(i,j) \) is applied.

Eq. (1) defines how the score for word \( w \) in position \( i \) is computed:

\[
P[w, i] = \frac{\sum_{j=1}^{\|D\|} c(w, j) \times f(i, j)}{\sum_{w' \in V} \sum_{j=1}^{\|D\|} c(w', j) \times f(i, j)}
\]

where \( \|D\| \) refers to the length of the document, \( V \) is the vocabulary, \( c(w, j) \) indicates the presence of element \( w \) in the position \( j \), and \( f(i,j) \) is the propagation function that rates the distance between \( i \) and \( j \). In this case, and taking as bases previous work on the matter [83], a Gaussian kernel is adopted as the propagation function.

Fig. 2 illustrates the idea behind the PLM reasoning.

PLM for summarization. The manner in which PLMs are employed to perform the summarization task comprises three stages. First, we need to conduct the definition of the vocabulary as a parameter for the PLM module. In our current configuration, the vocabulary is composed of the synsets corresponding to nouns, verbs and adjectives, together with the named entities that appear along the text. In order to get this semantic information, we use Freeling [87], an open source tool that allows linguistic analysis with different levels of granularity.

From this stage, a representation of the text that involves both the vocabulary and the positions of its elements is obtained.

Second, we create a seed, i.e., a set of words that can be significant for the text and will help the system to discard irrelevant parts of the discourse. The given headline is taken as seed in our configuration. It needs to be analyzed with the same tools as the source text (Freeling). As a result, a second vocabulary is then built from it.

Finally, the processing of the PLM against the seed allows us to compute scores for the text elements that are now conditioned by the information in the headline. At this stage, we have already calculated a collection of values associated with every relevant element using the PLMs for each position of the text. The aggregation of the different values related to each of those positions results in a vector with same length of the document, the Score Counter (SC), so that the score held in the index \( i \) will express the value for the position \( i \) in the text. Those positions in the text that show local maximums in the SC are retrieved as the most relevant points of the document. The sentences to which these positions belong are then selected as candidates for the summary. Since a value has been calculated for each position in the sentence, we can obtain a score for the sentence itself:

\[
S_{\text{score}} = \sum_{i \in S} SC[i]
\]

with \( S \) representing the sentence, and \( i \) indicating the positions within the document for that sentence.

These values also allow us to select from the candidates the sentences that will constitute the news extractive summary. Moreover, the computed values are necessary to define a new feature, named \textit{PLM Salience Score}, which will be used for the relatedness classifier in the next step. Its value derives from the aggregation of each score \( S_i \) associated with each sentence \( t \) included in the summary, following Eq. (3). Let \( S^* \) represent the set of the sentences belonging to the summary, the PLM Salience Score for a document \( D \) would be calculated as:

\[
\text{PLM Salience Score}_D = \sum_{i \in S^*} S_i
\]

3.1.2. Relatedness feature extraction

Besides the relevant information (i.e., the summary) and the PLM Salience Score obtained in the previous module, two similarity features are used as input to the relatedness classifier applied next. To obtain the features, the headline and the generated summary are used. They are described next:

- **Cosine similarity**: The cosine similarity between headline and summary of body text is computed. This feature is used to measure how similar the headline and summary are, irrespective of their size. Mathematically, it measures the cosine of the angle between two vectors projected in a multi-dimensional space [88]. The cosine similarity is advantageous because even if the two similar documents are far apart by Euclidean distance (due to the size of the document), chances are that they may still be oriented closer together. The smaller the angle, the higher the cosine similarity [89]. Although this metric is relatively basic [90], it usually brings significant improvements to retrieval models [88]. The cosine similarity measure between two vectors \( X \) and \( Y \) is obtained following Eq. (4) [91]:

\[
\text{Cosine similarity}(|X, Y|) = \frac{x \cdot y}{\|x\| \|y\|}
\]

where \( x \cdot y = \sum_{i=1}^{n} x_i y_i \), and \( \|x\| = \sqrt{x \cdot x} \).

For the calculation of cosine similarity, the text pairs are converted into Term Frequency–Inverse Document Frequency (TF–IDF) vectors, using the tools provided by scikit-learn [92].

- **Overlap coefficient**: This feature is defined as the intersection between two sets \( A \) and \( B \). In the current scenario, these sets contain the ngrams belonging either to the headline or the summary [93]. The overlap coefficient is given by Eq. (5) [94]:

\[
\text{Overlap coefficient} (A, B) = \frac{A \cap B}{\min(|A|, |B|)}
\]

If set \( A \) is a subset of \( B \) or the converse, then the overlap coefficient is equal to 1 else overlap coefficient should be between 0 to 1 [95].
3.1.3. Relatedness classification

The relatedness classification module exploits the PLM Salience Score, the relatedness features previously processed, as well as the summary to finally classify the headlines as related or unrelated. The proposed architecture is flexible to choose any model that allows classifiers to be improved.

In this case, the design of the relatedness classification module is based on fine-tuning the RoBERTa (Robustly optimized BERT approach) pre-trained model [96], applying a classifier to its output afterwards.

First, the headline and the summary are concatenated and processed with the RoBERTa model. The resulting vector is consecutively multiplied by the three features (PLM Salience Score, Cosine similarity, Overlap coefficient) to finally carry out the classification using a Softmax activation function in the output layer.

Specifically, we have chosen RoBERTa Large model (24 layer and 1024 hidden units) since it achieves state-of-the-art results in General Language Understanding Evaluation (GLUE) [97]. Reading Comprehension Dataset From Examinations (RACE) [98] and Stanford Question Answering Dataset (SQuAD) benchmark. Similar to [49,96,99], in this work we fine-tune RoBERTa to efficiently address a task that involves comparing sentences. RoBERTa optimizes Bidirectional Encoder Representations from Transformers (BERT) [100] by adding several modifications but without altering the original architecture, an approach that improves the results with respect to BERT in the main NLP tasks [96]. Some of those modifications involve: eliminating the prediction of the next sentence; performing the training on a greater volume of data; enlarging the batch size; and, lengthening the input sequence.

To create the classifier, the Simple Transformers library [10] was used, which creates a wrapper around HuggingFace’s Transformers library for using Transformer models [101]. Simple Transformers is an NLP library that allows the modification of hyperparameters so as to train, evaluate, and make predictions using the best state-of-the-art models.

In our model, the hyperparameter values are: maximum sequence length of 512; batch size of 4; training rate of 1e-5; and, training performed for 3 epochs. These values were established after successive evaluations, following previous experiments on this model [49,96,99].

3.2. Stance Stage

Given the related headlines obtained through the first stage on the proposed architecture, the main goal of this stage is to determine their type considering the remaining stances: agree, disagree or discuss. Therefore, the claim made in the headline can be finally classified into one of three classes left.

The inputs of this stage are:

- The headlines classified as related.
- The summary of the news content.

The output of this stage then is the final classification of the related headlines, where each of them is assigned one of the following possible stance values: agree, disagree or discuss. These classified headlines together with the unrelated headlines determined before, will comprise the final output for the whole HeadlineStanceChecker approach.

### Table 1

<table>
<thead>
<tr>
<th>Description FNC-1 dataset considering number of documents.</th>
<th>Documents</th>
<th>Headlines</th>
<th>Instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train set</td>
<td>1,683</td>
<td>1,683</td>
<td>49,972</td>
</tr>
<tr>
<td>Test set</td>
<td>904</td>
<td>904</td>
<td>25,413</td>
</tr>
<tr>
<td>Complete dataset</td>
<td>2,587</td>
<td>2,587</td>
<td>75,385</td>
</tr>
</tbody>
</table>

3.2.1. Stance classification

As in the Relatedness Stage (Section 3.1.2), the extractive summary generated in Section 3.1.1 is also used here.

Similar to the Relatedness classification module, this stage has been built using RoBERTa as the selected model capable of improving the classification. In this case, no additional features are considered, only two dense layers are included to reduce dimensions and, finally, the Softmax classification layer. The hyperparameters of the model used in this classifier are the same as those of the Relatedness classification, except for the classification output which in this case is of three classes: agree, disagree, discuss.

4. Experiments and evaluation environment

The proposed approach was applied and evaluated in the context of the Fake News Challenge FNC-1 whose goal was to determine a headline’s stance by classifying it in relation to its body text into 4 classes: unrelated, agree, disagree, and discuss. In this section, we first describe the corpus provided in this challenge. Second, we explain the experiments performed with different configurations of our system. Finally, the evaluation metrics used are outlined. The results obtained will be presented, discussed and compared to the other participating systems in the challenge in subsequent sections.

4.1. Fake News Challenge Dataset

The experimentation is conducted over the FNC-1 dataset whose instances are labeled as agree, disagree, discuss and unrelated.

The dataset was split into a training set (66.3%) and a testing set (33.7%), where neither the headlines nor the body text overlapped. The distribution of documents (bodies and headlines) is presented in Table 1.

As the distribution of the classes indicates in Table 2, there is a significant imbalance for both the training and testing sets where the instances of the unrelated class alone (over 70%) are greater than the sum of the remaining classes. At the other extreme, the disagree class is remarkably lower compared to the others.

4.2. Experiments

To measure our system’s performance, a set of experiments was conducted as follows, the results of which will be shown and discussed in Section 5. Our experiments can be replicated at Github [11]:

- Relatedness Stage Validation: The aim of this experiment is to assess the performance of this classification stage, where related or unrelated headlines are initially identified. First, we analyze and compare the performance of the classifier when either summaries or the full body is employed. Second, we conduct an ablation study to verify whether the relatedness features used for the classifier make a positive contribution.

---


Table 3

<table>
<thead>
<tr>
<th>System</th>
<th>Related F1 Score</th>
<th>Unrelated F1 Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relatedness Stage FNC-1-Summary</td>
<td>98.38</td>
<td>99.40</td>
</tr>
<tr>
<td>Relatedness Stage FNC-1-Body-text</td>
<td>98.36</td>
<td>99.37</td>
</tr>
</tbody>
</table>

- **Stance Stage Validation**: The goal of this experiment is to determine how accurate the Stance Stage is when the errors produced by the Relatedness Stage are avoided, thereby using an ideal input for this stage, i.e., the gold-standard headlines annotated as related in the FNC-1 corpus. By this means, we can measure the effectiveness of this stage in isolation. Furthermore, to validate the extent to which our proposed stance detection model can be generalized, we apply it to a different headline stance detection dataset, i.e., Emergent dataset [17].

- **HeadlineStanceChecker Validation**: In this last experiment, the entire system (integrating the Relatedness and Stance classifiers as a two-step classifier and using summaries as input for the whole process instead of the full text) is tested. Its performance is then compared to other configurations of the model as well as to competitive state-of-the-art systems.

In addition, we also investigate the system performance considering two different inputs: summaries and full body. This experiment and its further analysis is detailed in Section 5.4.

### 4.3. Evaluation metrics

Originally, the organizers of the FNC-1 challenge proposed the Relative Score metric, which assigned a higher weight to examples correctly classified, as long as they belonged to a different class from the unrelated one. The rationale behind this metric was to address the highly imbalanced distribution of the classes caused by the over-representation of the unrelated.

However, as pointed out in [102], the inner imbalance among the three related classes (agree, disagree, and discuss) was not addressed. Therefore, following [102], this study incorporates, in addition to the FNC-1 relative score, both a measure of $F_1$ class-wise and a macro-averaged $F_1 (F_1 m)$ as the mean of those per-class $F$ scores so as to address the imbalance among the less represented classes. The advantage of this measure rests in it not being affected by the size of the majority class. Additionally, average accuracy is also obtained.

### 5. Results and discussion

This section presents the results obtained in each of the experiments described in Section 4.2. The values are expressed in percentage mode.

#### 5.1. Relatedness Stage Validation

Our first experiment was designed to evaluate the first module as an isolated element of the system, acting as a binary classifier. In this case, we were not evaluating the system to detect agree, disagree or discuss examples, but to perform related versus unrelated classification. We carried out an analysis of the classification results and also an ablation study that considered the following involved features: cosine similarity; PLM Salience Score; and, overlap coefficient.

The performance of the relatedness classifier was first validated by analyzing whether the use of summaries had a positive impact on the output compared to using the whole document. The results are shown in Table 3. Both approaches used the three features previously described in Sections 3.1.1 and 3.1.2.

**Relatedness Stage FNC-1-Summary** refers to an experiment that uses summaries both to calculate features and to enter the classification model, whereas the Relatedness Stage FNC-1-Body-text approach uses the body text instead of the summary as input to the relatedness stage to classify the headline. The results validate the use of summaries as a useful approach to the stance detection problem as even if some information is excluded, the findings indicate a slight improvement when using the summarized text.

The approach that uses summaries throughout the process is able to improve the related class, which is the minority class. Figs. 3 and 4 show each confusion matrix of the two approaches with minimal variation in the classification, thus showing that the use of summaries does not harm the results of this classifier.

These results show that, by using the PLMs to condense the relevant information from a piece of news, the resulting summaries offer an attractive substitute for the full news text, enabling a reduction of the computational load for the classifiers, which increases when dealing with longer texts.

Furthermore, to evaluate the influence of the added features in the relatedness stage, an ablation study of the features extracted from the summary has been conducted. Each feature (Cosine similarity, PLM Salience Score and Overlap coefficient) has been removed and an experiment has been designed that will return the results of the classification without the incidence of the removed feature. To the extent that the classification result
reading and comparison, we have also included the non-ablation results. Ablation study results for the features used in the Relatedness Stage. To facilitate Table 4

<table>
<thead>
<tr>
<th>Removed feature</th>
<th>$F_1$ Score Related</th>
<th>$F_1$ Score Unrelated</th>
<th>$F_{1m}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cosine similarity</td>
<td>98.24</td>
<td>99.32</td>
<td>98.78</td>
</tr>
<tr>
<td>PLM Salience Score</td>
<td>98.00</td>
<td>99.23</td>
<td>98.61</td>
</tr>
<tr>
<td>Overlap coefficient</td>
<td>98.10</td>
<td>99.27</td>
<td>98.68</td>
</tr>
<tr>
<td>Non-ablated results</td>
<td>98.38</td>
<td>99.40</td>
<td>98.89</td>
</tr>
</tbody>
</table>

Fig. 4. Confusion matrix resulting from the Relatedness Stage FNC-1-body.

Table 4 Ablation study results for the features used in the Relatedness Stage. To facilitate reading and comparison, we have also included the non-ablation results.

is worse, this would imply that the eliminated feature has a great influence on improving the classification results. The most influential feature for the classification was observed to be the PLM Salience Score as the experiment that does not use the PLM score obtains the worst results, followed by the one that does not use overlap coefficient and, finally, by the one that uses cosine similarity. Table 4 shows the ablation study.

5.2. Stance Stage Validation

This experiment was designed to determine the validity of the Stance Stage. This task can be tackled as a double question, since two fundamental issues arise: (i) the validity of the Stance Stage as a general proposal; and (ii) the effectiveness of the Stance Stage performance within an ideal case.

As for the first issue, this experiment aims to demonstrate that the approach is not an ad-hoc solution but a general one. For this purpose, the Stance Stage was applied to a different stance dataset called Emergent dataset [17]. For this dataset, each example results from a combination of one article and its headline, and one claim. There are three different options for describing how a claim can be related to a piece of news. Specifically, each example was manually labeled by a journalist with one of the following tags: for, if the article states that the claim is true; against, if it states that the claim is false and observing, when the claim is reported in the article, but without assessment of its veracity. The dataset is composed of 2,595 examples, derived from the combination of 2,571 news, 2,536 headlines and 300 claims (see Table 5 for further details of the dataset).

To replicate our experimental environment with this dataset, the equivalence between labels in both datasets regarding their meaning is for ≃ agree, against ≃ disagree and observing ≃ discuss. Therefore, to validate the generalization of the approach, Table 6 includes the following performance results:

- **Emergent Upper Bound**: This experiment is performed as an upper bound by using a human-written headline created by a journalist, and considering it as a perfect summary that comprises the main information of the news body text. Nevertheless, this upper bound is only applicable to the Emergent dataset since in the case of FNC-1 no journalist-written headline is provided, as occurs in the case of the Emergent dataset.
- **Stance Stage using Emergent Dataset**: Our model is trained with the Emergent dataset and the Stance Stage is applied to it.
- **Stance Stage tested with Emergent, but trained with FNC-1**: This performance uses the Emergent dataset to test the Stance Stage but with the model trained on the FNC-1 so as to demonstrate the extent to which our proposal can be generalized.

The second aspect that needs to be addressed relates to the appropriateness of this second stage and its performance by isolating this stage from the rest of the system. The strategy here is focused on avoiding the errors inherited from the previous stage. To achieve this, only the examples tagged as related from the FNC-1 Gold-Standard are used and evaluated. The results of this performance correspond to Stance Stage FNC-1 row in Table 6.

The analysis of the results obtained in this stage regarding the comparison of the performance using Emergent dataset are very promising considering that this model is using automatic summaries. The results are very close to the upper bound obtained by using human-made summaries. Analyzing per class, using the Emergent dataset for a training and testing task, the disagree class is even better classified by using automatic summaries. Additionally, when the approach is trained on the FNC-1 dataset and the test is carried out on the Emergent dataset, the discuss class surpasses the upper bound.

Regarding the performance of the Stance Stage in isolation, i.e., without considering the Relatedness Stage, the results present a slightly better performance than the whole approach with an increase of 2 percentage points (see Table 7). This was to be expected since errors derived from the Relatedness Stage are avoided. To conclude, these figures demonstrate that the approach, apart from potentially being a general solution, also demonstrates that using summarization of the body text as input is useful for the stance detection task, since the performance is very close to the upper bound proposed at Emergent.

5.3. HeadlineStanceChecker validation

The results of the HeadlineStanceChecker are shown in Table 7. This approach integrates the Relatedness and Stance classifiers and only uses automatic summaries for these two classifiers (but for the Relatedness classifier, the external features are included). This table contains the performance for the class-wise $F_1$, macro-average $F_{1m}$, accuracy (Acc.) and the relative score (Rel. Score). Moreover, it also provides the results obtained by competitive state-of-the-art systems together with additional configurations that were also tested.

The 3 first rows are the top-3 best systems that participated in the FNC-1 challenge. The results for each of the evaluation metrics

---

were calculated using the confusion matrices and results were published [47] or made available by the authors.\(^{13,14}\)

The fourth row corresponds to the Human Upper Bound, and is the result of conducting the FNC-1 stance detection task manually. This upper bound was defined by [46]. Five human annotators were asked to manually label 200 random instances, obtaining an overall inter-annotator agreement of Fleiss’k of 0.686. Due to the fact that there is no upper bound reported in the FNC-1 data, we also used these values as reference for comparison purposes.

Next, the fifth and sixth rows include the results of recent approaches [48,49] that also addressed the headline stance detection task using the FNC-1 dataset, but did not take part in the challenge. Since there was no public code available, these results were also calculated from the confusion matrices provided in the papers.

The seventh row indicates the results for our HeadlineStanceChecker approach but configured only with a single classifier. We have called this approach HeadlineStanceChecker-1stage. Finally, the last row belongs to our HeadlineStanceChecker approach, using our proposed two-stage classification. For clarity purposes, in the table we will refer to this approach as HeadlineStanceChecker-2stages. Regardless of whether the classification is conducted in 1 or 2 stages, both approaches have used just the automatic summaries created from the full body text during the whole process.

As can be seen in Table 7, HeadlineStanceChecker-2stages is competitive enough with respect to the other systems, given that it only uses short summaries for the classification process, and not the full body text as the other systems use, so the information reduction does not imply a high loss in the results obtained, being better than the FNC-1 participants, and the human upper bound. Furthermore, the results also validate the fact that the divide-and-conquer strategy applied for dividing the classification into two stages is beneficial and yields better performance when using our proposed model with a single classifier (rows 7th and 8th). This is especially the case for detecting disagreement between the headline and the news article.

Furthermore, the most remarkable improvement for HeadlineStanceChecker-2stages is achieved in the discuss category, over performing all the remaining approaches. The \(F_1\) improves by around 2 points compared to the second-best approach, i.e., [49], and 13 points over the lowest-performance system [47] in this category. By achieving competitive values in the other classes as well, HeadlineStanceChecker-2stages obtains a final macro-F1 value of 80.39%, being only beaten by the system proposed in [48], which takes advantage of a considerable number of external features beyond similarity to enrich the neural model. It is worth highlighting that in terms of accuracy and relative score, our approach (i.e., HeadlineStanceChecker-2stages) obtains the best result among the automatic systems in both cases, achieving 94.31% and 91.02%, respectively.

Focusing on the results obtained by the participants in the FNC-1 competition, when these results are analyzed independently for each of the classes, it can be seen that except for the classification of unrelated headlines (whose results are close to 100% in F1 measure, and this happens also for the remaining approaches as well) for the remaining classes, the results are very limited. The systems that participated in the FNC-1 competition have a very reduced performance especially in detecting the disagree stance, whereas the detection of agree is around 50% in F1 measure and for discuss around 75% for the best approach.
Outside the FNC-1 competition, the performance increases in all categories, being the disagree category one of the most challenging to classify, in which only the approach proposed in [48] obtains surprisingly high results for this category compared to the remaining methods.

After having established that HeadlineStanceChecker-2stages performs adequately (correctly detecting 94.31% of the test set classes), the confusion matrix presented in Fig. 5 provides more detail on the actual performance of the system for each stance class. From this information, we can observe that per class, the major classification problems occur with disagree and discuss classes. The data reflects that 22.5% of the total number of disagree stances are being classified as discuss, whereas as 23.6% of the total number of agree stances are classified as discuss. However, only 7.2% of the total number of discuss stances are being classified as agree and 17.9% of the total number of disagree stances are being classified as agree.

5.4. Summary versus body text analysis

Finally, in order to test the convenience of using the summary or the body text as input to our whole system, a final analysis was performed by an experiment designed to allow us to compare the results in both cases. To determine how this would be accomplished, we considered the singularities of our system, since the use of RoBERTa implies certain constraints that affect the input processing.

RoBERTa, as a classification model, allows a maximum input length of 512 tokens, called maximum sequence length. Information that exceeds such a length is truncated. Our configuration takes as input both the headline and the text to which it refers, body text or summary, but it is relevant to remark that, in this case, the sequence length includes the tokens of the headline plus the tokens of that text. Since the headline must remain complete for the classification process, if it is necessary to truncate, it is the information in the body text which is lost.

In relation to the aforementioned issue, the previous work described in [49] focused on the analysis of the length of the body text for classification purposes, showing that for the examples in which the input sequence is greater than 512 tokens, the accuracy of the classification decreases considerably with respect to smaller sequences.

Against this backdrop, our hypothesis states that applying summarization to the text before classification implies an improvement in the results. In order to prove it, we first create two subsets from the FNC-1 dataset according to the news story length: subset>512 and subset<512. Table 8 shows the class distribution for both subsets.

Next, we trained and tested the system with both subsets twice: first with the bodies as input, and then with the summaries. The results in Table 9 show that for long news stories (subset>512), the system performs better with summaries as input than truncating the text of the full article. This could happen because reducing the input by simply cutting text at the end of the document results in relevant information being lost, whereas when making a summary, it is the relevant information that prevails in a more concise mode.

Similarly, results for subset<512, the shortest news stories, are reported in Table 10. The system was again trained and tested, taking the body and the summary as inputs. In this case, results are better when using the full body text, which could indicate that all the information needed for a proper classification is present by considering the whole text (an unfeasible scenario with longer texts).

There exist no explicit rules that determine what the length of a news article should be, but there is instead certain evidence supporting that news tend to be longer than 512 tokens. In Table 11 we have gathered statistics from the most popular news datasets that are being used in language processing tasks. All together, they contain more than 2 million articles from different sources, with an average length superior to 512 tokens. The relevance of our approach is made clear by these figures, which indicate that, in most cases, using news summarization would be the right strategy.

Table 8: Class distribution for FNC-1 subset>512 and FNC-1 subset<512.

<table>
<thead>
<tr>
<th>Class</th>
<th>FNC-1 subset&gt;512</th>
<th>FNC-1 subset&lt;512</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agree</td>
<td>1,112</td>
<td>2,566</td>
</tr>
<tr>
<td>Disagree</td>
<td>314</td>
<td>526</td>
</tr>
<tr>
<td>Discuss</td>
<td>3,536</td>
<td>5,373</td>
</tr>
<tr>
<td>Unrelated</td>
<td>12,886</td>
<td>23,659</td>
</tr>
<tr>
<td>Total</td>
<td>17,575</td>
<td>38,242</td>
</tr>
</tbody>
</table>

Table 9: HeadlineStanceChecker-2stages results for subset>512 with different inputs: news body and news summary.

<table>
<thead>
<tr>
<th>Input</th>
<th>F₁ Score</th>
<th>F₁im</th>
</tr>
</thead>
<tbody>
<tr>
<td>News body</td>
<td>54.45</td>
<td>61.40</td>
</tr>
<tr>
<td>News summary</td>
<td>59.61</td>
<td>66.96</td>
</tr>
</tbody>
</table>

Table 10: HeadlineStanceChecker-2stages results for subset<512 with different inputs: news body and news summary.

<table>
<thead>
<tr>
<th>Input</th>
<th>F₁ Score</th>
<th>F₁im</th>
</tr>
</thead>
<tbody>
<tr>
<td>News body</td>
<td>78.64</td>
<td>84.35</td>
</tr>
<tr>
<td>News summary</td>
<td>74.17</td>
<td>80.03</td>
</tr>
</tbody>
</table>
6. Conclusions and future work

HeadlineStanceChecker has been demonstrated to be an effective approach for detecting misinformation in news, specifically when a headline has to be compared to its body text. The novelty of our approach rests on two key premises: (i) the adoption of a divide-and-conquer strategy, thus tackling the stance classification problem by means of a 2-stage neural architecture; and (ii) the use of extractive semantic summarization instead of the full news body text for the whole classification, in addition to a salience and two similarity features that will help to determine the relatedness of the headline with respect to the news article.

To show the appropriateness of HeadlineStanceChecker, different experiments were carried out in the context of an existing task (Fake News Challenge FNC-1), where the stance of a headline had to be classified into one of the following classes: unrelated, agree, disagree, and discuss. The experiments involved validating each of the proposed classification stages in isolation together with the whole approach, as well as a comparison with respect the state of the art in this task. Furthermore, additional experiments with another corpus for headline stance detection (i.e., Emergent dataset) were also performed to verify the generalization of our approach. The results obtained by our system were very competitive compared to other SOTA systems obtaining 94.31% Accuracy, as well as the highest result in FNC-1 relative score compared with the state of the art (91.02%).

The unbalanced nature of the FNC-1 dataset leads to existing systems being more capable of learning how to detect unrelated headlines, but are less accurate when it comes to the remaining classes. Even so, the results obtained by HeadlineStanceChecker for the different categories with less examples, agree, disagree and discuss are fair enough and promising, which indicates that the chosen approach is appropriate for the task.

Future work will aim to improve the results of agree and disagree classification by extending the system to take into consideration Sentiment Analysis features. Furthermore, as reported speech is recently being introduced to determine bias and document stance, it could be very useful for determining the stance of headlines and news articles. Some reporting events are neutral, for example, by using reported or said, whereas some others introduce a stance, for instance, ‘deny’ implies disagreement or ‘confirm’ indicates agreement.

Besides, another interesting aspect to focus on would be to investigate the relation of the stance detection classes (agree, disagree, discuss and unrelated) with the “incongruent” and “congruent” classification to determine whether this relation can provide some insights for different scenarios.

Finally, as a future goal that contributes to investigating the problem of fake news detection, we expect to apply HeadlineStanceChecker to a real world scenario to detect when headlines introduce mis- or disinformation to readers. Our contribution to improving the current research in the field, by means of new learning strategies and discourse aware techniques, will help to combat online fake news, a societal problem that requires concerted action.

### References

3. S. Tavenissen, As fake news spreads lies, more readers shrug at the truth, N.Y. Times (2019).


B.S. Andreas Hanelowski, F. Caspelherr, Description of the system developed by team athene in the FNC-1, 2017.


Z. Zhou, J. Lin, X. Liu, Condensed convolution neural network by attention over self-attention for stance detection in twitter, in: International Joint Conference on Neural Networks (IJCNN), IEEE, 2019, pp. 1–8.


[87] B. Li, L. Han, Distance weighted cosine similarity measure for text classification, in: Proceedings of the 14th International Conference on Intelligent Data Engineering and Automated Learning, Springer-Verlag, 2013, pp. 611–618.


