Detecting Incongruous News Articles

Jacob Barrow - 40337360

Submitted in partial fulfilment of
the requirements of Edinburgh Napier University
for the Degree of
BSc (Hons) Computing Science

School of Computing

March 31, 2021

Abstract

Contents

1 Introduction

Fake news, disinformation and misleading articles have been the subject of much debate and controversy. With the decline of traditional print media, more people turn to online sources for news; publishing has a much lower entry barrier, making it an easier target to spread false information.

Incongruous headlines misrepresent an article's content and, intentional or not, can lead to the formation of unfounded opinions and misconstrued facts. Existing articles should be analysed to ascertain the extent of the problem, and a system created whereby unseen articles' congruence can be determined.

Considering the rate at which publishers output news articles, it would be impractical for a human-scale operation to undertake this task. Therefore, this dissertation seeks to create a classifier capable of calculating a measure of a headline's incongruence by utilising Natural Language Processing (NLP) to analyze the components of a news article.

As the laws of natural language - the spoken and written means by which humans communicate with each other - are uncodified, blurry and change from generation to generation, analysing the meaning of text poses a considerable challenge. For instance, while the sentiment of a headline may be entirely at odds with an article's content, this could potentially be a sarcastic or ironic technique used by the publisher.

Overcoming these obstacles requires a thorough understanding of NLP and the range of approaches and techniques used to identify and extract meaning from text.

A considerable set of articles will need to be created, with both a longitudinal aspect spanning an extensive period and a cross-section of current online news articles. Additionally, to provide both a baseline and a method of training a classifier, a portion of this dataset will need to be labelled with absolute values of congruence.

1.1 Aims

The primary aim of this dissertation is to create an incongruence classifier. Additionally, a large dataset of news articles will be created, and a subsection of them labelled.

1.2 Research Questions

In order to judge the success of the project and to provide a direction for research, this dissertation seeks to answer the following questions:

- To what extent do incongruent articles exist in current contemporary online news?
- How, if at all, has the language of news articles changed over the past decade?
- How useful is statistical NLP in determining the congruence of a news article's headline?

1.3 Project Outline

The project will consist of a range of deliverables and milestones.

Background Research

A literature review will be undertaken to provide context to the project and define its scope

Data collection

To provide the algorithm with articles to classify, a dataset is required. It will be sourced from several publishers and ideally cover a large timescale to allow trends to be identified.

Data Labelling

A raw dataset can be made more valuable by creating a training set from it. The set will be a small subsection of the articles that volunteers will label with their incongruence judgement.

Algorithm Creation

Once a dataset is generated and labelled, the algorithm to classify the articles can be implemented.

Visualisation

The breadth of the raw dataset will make it unwieldy to work with, so a platform will be created to aid with the visualisation and management of it.

Analysis

The algorithm will create a range of data points that can be analysed. Trends will be spotted in the dataset used, and comparisons made with different implementations of the algorithm.

2 Background Research

2.1 Introduction

This study aims to create a method of detecting incongruence in news articles. Before the implementation begins, it is essential to review the existing literature to give the study context.

This literature review begins by defining different several types of incongruence and specifies the bounds applicable to this study.

It then goes on to evaluate several existing approaches, which are discussed to give a clearer picture of the field as it stands, and the gaps present in the research concerning incongruence detection. These sources are also analysed to gain some insight into a possible approaches to tackle the problem at hand.

Natural language processing (NLP) is then defined, and different features and approaches reviewed and discussed.

2.2 Types of Incongruence

In the scope of media, incongruence is a broad term that covers many different forms of deception and misleading information. Chesney, Liakata, Poesio, and Purver (2017) classifies three different types of incongruent news articles: clickbait, fake news, and sensationalism.

Clickbait

Potthast, Köpsel, Stein, and Hagen (2016) define clickbait as a kind of "web content [...] designed to entice its readers into clicking an accompanying link". Clickbait uses exaggerated language, outright fake information and can be accompanied by graphics designed to entice a reader. Figure 1 shows an example of clickbait, sourced from a Natural Health website [1].

Mahoney (2015) terms a collection of clickbait stories as a 'chum boxes' - chum being dead fish used to bait other fish. Mahoney goes on to examine how clickbait uses psychological methods to manipulate and how they can have an unconscious effect on an individual.

Fake News

Allcott and Gentzkow (2017) defines fake news as "news articles that are intentionally and verifiably false, and could mislead readers". For example, a fake news conspiracy theory claimed that a pizzeria, Comet Ping Pong,

[1] `https://naturalon.com/`

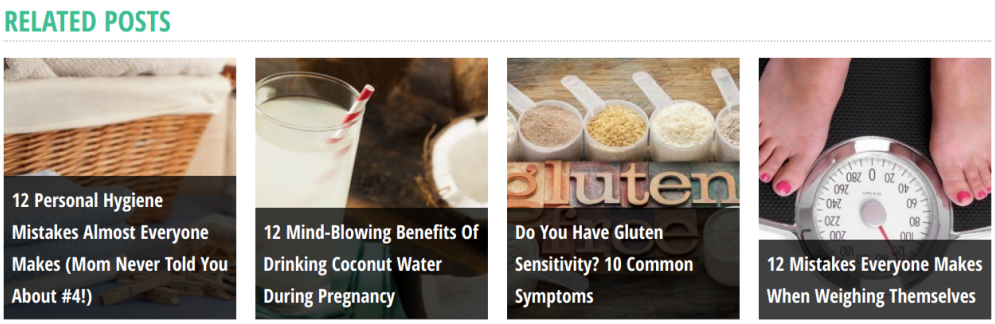


Figure 1: Several clickbait articles in a 'chum box'

in Washington ran a child sex ring in its basement. Figure 2 shows a news article from 2016 from Your News Wire[2] (now News Punch).

Figure 2: A fake news story

This conspiracy theory, perpetuated by fake news, lead to a man walking into the Comet Ping Pong pizzeria with an assault rifle and firing several shots. The restaurant's owner and staff also received several death threats (Lopez, 2016).

Allcott and Gentzkow go further in their definition and give the following sub-categories for fake news: satire, parody, fabrication, manipulation,

[2] https://archive.is/YTk3n

advertising and propaganda. While satire and parody's intention is not to deceive but to criticise, the other classifications have more subversive aims, such as misinforming people or gaining as many clicks as possible.

Sensationalism

Molek-Kozakowska (2013) defines sensationalism as "a specific discourse strategy aimed at channeling audience's attention, which may well be resorted to by both popular and quality outlet". They suggest that media fails to provide important and valuable news, preferring superficial and quick-paced content.

Below are examples of sensationalised headlines sourced from The Sun:

- DOOMSDAY DISEASE FEARS Terrorists could turn 'sniff and die' virus that kills victims in 24 hours into a BIO-WEAPON
- SPICE UP YOUR LIFE Chilli and ginger 'slash the risk of cancer – stopping tumours growing'
- JAB DEBATE As Melinda Messenger slams the HPV jab the parents of two teenagers blame their daughters' 'paralysis on vaccine'
- 'I KNOW WHO KILLED JONBENET' Juror from the JonBenet Ramsey case gives sensational interview revealing he 'knows who killed six-year-old'

These headlines use dramatic language ('slams', 'sensational', 'slash') to evoke a sense of urgency and excitement in the reader, urging them to click through to the rest of the article. Unlike clickbait headlines, information is not withheld but rather dramatised - while the aim is still to get as many clicks as possible; this is achieved through different means.

This sensationalism is intended to provoke and entertain, at times at the expense of accuracy (Chesney et al., 2017). The below headline, again from The Sun, illustrates this well:

> UK lockdown: March 29 rules loophole means you CAN go inside your pal or family's home as restrictions ease [3]

This headline suggests that people will be able to socially gather inside someone else's house as a result of the ease of coronavirus restrictions, however, this is not the case. The article's body reveals that you can only enter a friend's house to use the toilet or if you're walking through to access an

[3] https://www.thesun.co.uk/news/14453072/can-visit-garden-lockdown-eases-indoors-loo/

outdoor space such as a garden, and that meeting indoors is still banned. This example shows the danger of sensational headlines - by describing a deliberate, common-sense exception to a law as a loophole, readers may be under the false impression that it is acceptable to subvert restrictions and mix socially indoors.

2.2.1 Project scope

Fake news is not the focus of this project - by its nature, the entirety of a fake news article will be false, not just the headline. Therefore, to determine whether an article is fake, external sources would have to be consulted. Creating an algorithm for truth, while an open problem in computer science[4], is considered out of scope.

Instead, this study will seek to evaluate to what extent a headline represents an article's body. It aims to identify sensationalism, exaggerated news stories and potentially some types of clickbait.

2.3 Impact of Incongruence

Karlsson and Strömbäck (2010) characterise online news both by its immediacy and interactivity, which has both shortened the news cycle and increased the competitivity between publishers. Therefore, publishers have to make the news more appealing to potential consumers, and employ deceptive tactics to draw readers in.

In the information-overload arena of online news reporting, a news article's body is less read than the headline (Gabielkov, Ramachandran, Chaintreau, & Legout, 2016). This means that an incongruent headline will be taken at face value, as readers will not have read the body that refutes its claim.

For those that read beyond the headlines, incongruent articles can still be problematic; it's a well-established theory in psychology that first opinions matter (Digirolamo & Hintzman, 1997). Ecker, Lewandowsky, Chang, and Pillai (2014) ran a study that investigated how headlines affect the processing of the facts in the news: "Information that is initially accepted as valid but is later found to be incorrect can have a persistent influence on people's memory and reasoning". Publishers can seek to sway individuals by using choice phrases to influence their mindset, which means that the same content could be interpreted in many ways depending on its headline (Reis et al., 2015). This demonstrates that if a headline is incongruent, even if the individual

[4] `https://www.youtube.com/watch?v=leX541Dr2rU`

reads the whole article there's a real possibility they will be left with a false impression of the facts.

2.4 Existing Approaches

Manjesh, Kanakagiri, Vaishak, Chettiar, and Shobha (2017) used a range of different techniques to identify clickbait and were able to achieve a 98% accuracy with a deep learning approach. However, they only analysed the article's headline and disregarded the body text. They found that clickbait headlines tend to have elaborate sentences with various linguistic nuances, such as "21 Pics Of Celebs Photoshopped In The Best Way Ever. These Are EPIC". There's also a statement at the end to further strengthen the central claim of the headline.

Park, Kim, Yoon, Cha, and Jung (2020) used a deep learning approach to create a web interface for detecting incongruent articles and managed to gain an accuracy of 86%. However, for their dataset, they generated incongruent articles by swapping a different article's text in for the original. For example, Headline A would have a section of Article B's body. They then considered congruent headlines to be those with the original body text in place. This method could lead to false positives, and as the manufactured dataset does not reflect the incongruence in real-world articles, their algorithm's output lacks validity.

The first Fake News Challenge (FNC1) was held by Pomerleau and Rao (2017). The challenge supplied a dataset of articles and encouraged contestants to create a classifier capable of detecting fake news. While there have not been subsequent challenges, 50 teams competed and produced a wide range of different approaches.

2.4.1 Exisiting Labelled Datasets

A labelled dataset is a set of data points tagged with some information that identifies that data's characteristic. For example, a labelled dataset may consist of several articles that have been tagged with a level of congruence. Labelled datasets can be used to inform and train an algorithm to correctly tag unlabelled data or be used as a 'gold-standard' to determine a system's performance.

Chesney et al. (2017) reviews the current datasets available to detect incongruent articles and concludes that while many are available and have some potential use, none are a good fit for the task.

Source	Labels	Size
Clickbait Challenge	Discrete (Not/Slightly/Considerably-/Heavily Clickbaiting)	2495
Piotrkowicz, Dimitrova, and Markert (2017)	Continuous (prominence, sentiment, superlativeness, proximity, surprise, and uniqueness)	11980
FakeNews Challenge	Discrete (Agree, Disagree, Discuss, Unrelated)	50000

Table 1: An overview of existing labelled datasets

2.5 Natural Language Processing

Natural language processing (NLP) is a method of extracting information from a spoken or written language. 'Natural' here means the freer and less well defined human language, as opposed to strictly interpreted programming and mathematical notation. (Jackson, 2002)

As natural language is filled with a range of nuances, assumptions and relies heavily on context, codifying it into a standardised, programmatic output poses a range of difficulties. For example, consider the following two sentences:

- Apple's shares fell by 10% in the last quarter
- An apple a day keeps the doctor away

The word 'apple' appears in both sentences, but in one it refers to a multinational company, and in the other a tasty fruit. Only by using the context clues in the surrounding sentence can the word's meaning be deduced.

Various approaches seek to tackle the problems inherent in determining the meaning and sentiment of natural language, each with their own characteristics, strengths and limitations.

2.5.1 Statistical NLP

Statistical NLP creates metadata from a sentence and aims to extract meaning using statistical inference (Manning & Schutze, 1999). Several techniques can be used to create and interpret the metadata.

Tokens

A token is typically an alphanumeric string or a punctuation mark. For instance, the sentence "Is this the way to Amarillo?" could be tokenised

(represented as a list of tokens) like so: `"Is", "this", "the", "way", "to", "Amarillo", "?"`.

n-grams

An n-gram is a subsection of a tokenised sentence, where `n` represents the number of tokens in a subsection. An n-gram of length 3 (also known as a tri-gram) of the above sentence could be `"way", "to", "Amarillo"` The location of these n-grams, their frequency and their composure all provide data points that can provide insights into the meaning of a body of text (Banerjee & Pedersen, 2003).

Colocations

Manning and Schutze describe a colocation as "an expression consisting of two or more words that correspond to some conventional way of saying things". For example, "around about", "stark naked", and "stiff upper lip" are all colocations. In a colocation, the subsequent parts make up a whole and lose some of their independent meaning - "fool hearted" makes sense to an English speaker's ear, but "idiot hearted" could sound offensive or cause a misunderstanding. One way of identifying colocations is to count the frequency of bigrams in a body of text - a high number of two words occurring next to each other could indicate a colocation.

By themselves, these techniques would be insufficient to extract any substantiative meaning or measure of congruence from an article. However, they will be essential in forming a foundation for a classifier, and knowledge of them will be crucial in understanding the more advanced and complex NLP approaches.

2.5.2 Sentiment Analysis

Sentiment analysis is a branch of NLP that could potentially aid the detection of an incongruous article. It is a relatively new development (no substantial research had been conducted before 2000) that aims to extract opinions from text and speech (Liu, 2012).

Liu (2012) identifies a fundamental problem with sentiment analysis: sentiment is a very subjective concept; calculating an absolute sentiment score for a sentence is fraught with potential difficulties. For instance, the phrase "I really enjoy writing in an academic style" could be interpreted as a very positive remark or perceived with sarcastic overtones and classified as a negative sentiment.

Sentiment analysis could be used to approach this project - both the headline and the body could be analysed, and the results compared. The difference in sentiment could then be used as an incongruence score.

However, two pieces of text can share the same sentiment yet disagree with each other. For instance, consider the following two phrases:

- I love Daniel Craig's work - he is the best Bond
- Sean Connery is by far the best Bond, and he is a great actor

While both have a very positive sentiment regarding each actor's ability, they are entirely opposed in opinion. Likewise, it is also possible for two texts to have opposing sentiments and yet be congruent in meaning - a headline could be a positive sentiment about wind turbines, and the body could contain very negative sentiments about coal-fuelled power plants.

Because of these issues, sentiment analysis by itself would not provide a significant metric by which an article's congruence can be determined. However, it could prove beneficial to integrate it alongside other NLP techniques.

2.5.3 Named Entity Recognition

Named Entity Recognition (NER) is the process of extracting and locating references to real-world objects from text. These named entities can represent a vast number of 'information units', such as people, organisations, locations and numeric expressions (Nadeau & Sekine, 2007).

There are several approaches to NER, two of which are covered here.

One Hot Encoding

One hot encoding represents each word in a phrase as a binary string, with the length of the string being the number of unique tokens in the phrase (Bommana, 2019). For example, consider the following sentence "I've got to go to France!", which can be tokenised as:

```
"I've", "got", "to", "go", "to", "France", "!"
```

I've	100000
got	010000
to	001000
go	000100
France	000010
!	000001

This phrase can be encoded using the one hot method, using a binary string of length 7, as on the left.

These binary representations can then be consumed by a neural network, which can be trained to identify named entities by detecting patterns in the encoded string's formation.

Word Vectorisation

The act of converting a word to a vector (word2vec) is a simple but powerful concept. As well as being used to identify words with similar meanings, it can also identify analogous pairs and connections between words. The most famous of these analogies is "Man is to king as woman is to x", where word2vec can give x as "queen". (Church, 2017)

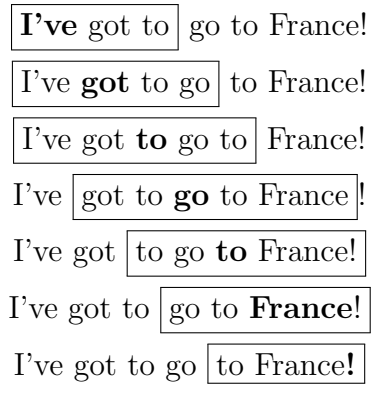


Figure 3: Context windows

These core concept of these connections are 'context windows' - a set of words that surround a target. For instance, using a window of size 2 (2 tokens on either side of the target), the example sentence would be analysed as in figure 3 (the target token is emboldened).
Bigrams (pairs of tokens) can then be taken for each window, and the collection of bigrams then used to train a neural network to generate a vector.

These vectors take the form of an array of floats used to represent a word - for instance, "cat" could be represented as `[0.023, 0.131, 0.001, 0.415, 0.901]`. The array's length is determined by the number of neurons in the neural network's hidden layers, as each neuron is responsible for calculating a single float (Bommana, 2019). These vectors can be thought of as a point in a multi-dimensional space, and connections can be made by traversing the dimensions to find new words.

In terms of this project's scope, word vectorisation could provide some useful insight into the headline's relatedness and the body text. Again, this may not be sufficient as a standalone technique, but when used as part of a more extensive pipeline, it has the potential to yield some good results.

2.5.4 Lexical Overlap

Lexical overlap, also known as lexical similarity or textual entailment, is a measure of similarity between two elements of text (Adams, 2006). This overlap can either be character-based (similar text) or statement-based (similar meaning).

Pradhan, Gyanchandani, and Wadhvani (2015) reviewed several different approaches to obtaining a measure of lexical overlap. Statement-based lexical overlap uses the distance between word vectors to determine the difference

in sentiment. Cosine similarity can be used to calculate the distance between two words by taking into account the angle created between the vectors and the origin (Qian, Sural, Gu, & Pramanik, 2004). Alternatively, the difference between a single word and a set of words can be calculated using centroid-based similarity. This is the measure of distance from one point in a vector to the geometrical centre (the arithmetic mean) of a set of points (Awrejcewicz, 2012).

Character-based lexical overlap is used to determine how similar words are with respect to their composition. For example, 'witches' has a considerable character overlap with 'britches', but minimal overlap with 'broomstick'. The Levenshtein distance can be used to calculate overlap and is loosely defined as 'the minimum number of [operations] to make two strings equal' (Navarro, 2001). Other distance measures include the Hamming distance (number of replacements), the Episode distance (number of additions) and the Longest Common Subsequence distance (number of additions and deletions).

2.5.5 tf-idf

Term Frequency times Inverse Document Frequency (tf-idf) is a measure used to determine a search query's relevance to a given document (Rajaraman & Ullman, 2011). Its core mechanic uses the inverse frequency of a word in a set of documents to determine relevance, which can be used to discount superfluous words from the query. For instance, given the search phrase 'The best apricots in Australia', the word 'The' is not important to the query. As 'The' will appear in many documents (if not all of them), the inverse of its occurrence can be used to give it a low weighting.

Given a set of N documents, where f_{ij} is the number of times a word i appears in document j, Rajaraman and Ullman define the term frequency TF_{ij} to be:

$$TF_{ij} = \frac{f_{ij}}{\max_k f_{kj}}$$

This augments f_{ij}, as it divides it by the maximum number of occurrences of *any* term in the document (the most frequent word). This means the word with the most occurrences will have a TF of 1, and the least frequent word will have the lowest TF. The inverse document frequency IDF_i is defined as $\log_2(N/n_i)$, where n_i is the count of documents that a term i appears in. This gives a measure of how common a word is with regards to the whole set of documents - the rarer the word, the higher its IDF.

With both these elements in place, the overall tf-idf is calculated using $TF_{ij} \times IDF_i$. The higher the output, the more likely it is the term i represents

the content of the document j.

tf-idf could potentially be used to tackle the problem of identifying incongruence by substituting the search term for the article's headline and the document for the article's body. Its ability to detect relevance could then be used to provide a measure of similarity between the two. However, this approach may only be of use for articles with a substantial disparity between headline and body; it is unlikely to identify subtle differences in text as it is designed to obtain a broad measure of relevance.

2.5.6 Text Summarisation

Text summarisation is the process of distilling a body of text into a shorter format by identifying the essential facts and disregarding the rest. Techniques can be described as either *extractive* or *abstractive* (Maybury, 1999) (Tan, Wan, & Xiao, 2017).

Extractive Summarisation

Extractive approaches to summarisation search within the text for key information and then reproduce it word for word (Tan et al., 2017). For instance, in the below paragraph, the key information has been emboldened.

On a rainy Sunday, ***Felix went to the supermarket****. While he was there, he wandered through the aisles and eventually he* ***bought a jar of honey****, which was on sale. When he got home he spread it with wild abandon onto* ***some toast****.*

A summary can be constructed using these core facts to represent the gist of the text, but remove the superfluous information like so:

Felix went to the supermarket and bought a jar of honey for some toast

To rank the importance of phrases and determine the key facts, a sentence score is calculated using various indicators, employing machine learning to weight each indicator. Depending on the approach used, the best sentences or an optimised set of sentences are selected and used to generate the summary.

Abstractive Summarisation

Conversely, abstractive approaches present the text's information in a new way, using different words and phrases to construct a more concise and natural summarisation (Allahyari et al., 2017). Synthesising language in this way is not trivial; it requires an understanding of each phrase's context, relevance, and importance, meaning classical NLP approaches are not best suited to the task. Several advanced approaches have been used for abstractive summarisation. Xu, Yang, and Lau created a system that finds key terms from the

text on Wikipedia and then use the technique of keyword clustering to determine replacements and construct a summary. Banko, Mittal, and Witbrock view the task of summarisation from the point of view of text translation instead of text comprehension. They use a statistical approach to machine translation to translate the text from a verbose language to a concise one, and in doing so managed to introduce new text into the system - for instance, it correctly generated "soybean grain prices lower" from the phrase "Corn, Wheat Prices Fall".

By using summarisation to generate a single sentence from an article of text (known as extreme summarisation), a headline can be generated. Abstractive summarisation is best suited for headline generation, as an understanding of the content of the article is required to distil it down so heavily (Hayashi & Yanagimoto, 2018). Additionally, the nuances of headlines (for instance, using puns and sarcasm) may mean that the phrases in them do not appear in the main body, even though they are relevant to it. As this project focusses on classical NLP methods, as opposed to machine-learning approaches, abstractive summarisation is out of scope.

3 Methodology

3.1 Data Collection

To create an algorithm to detect incongruence in news articles, news articles first have to be collected and structured, and their suitability to the project determined. While there are already existing datasets, they tend to be incomplete, out of date, or in inaccessible formats, as they have been made to tackle different problems. For instance, while the Internet Archive has a catalogue of online news from around the world[5], it is heavily fractured over thousands of files and is difficult to parse. Additionally, several datasets used to identify misleading and deceptive articles fabricate incongruence by pairing one article's headline with the body of another, such as in Park et al. (2020). As mentioned in section 2.4, this does not reflect the more subtle incongruity in real-world articles, so is not suited for this project.

To overcome these obstacles, a bespoke dataset needs to be created that can be classified and analysised by an NLP algorithm.

3.1.1 Attributes

Before collecting the data, it is essential to decide what form it will take and what attributes will be stored.

As the project aims to identify incongruence between an article's headline and body, these two attributes will be included in the dataset. To identify trends and allow for further analysis, the article's date of publication and the publisher (e.g. BBC, The Guardian, etc.) will also be stored.

The collection could have gone further and retained the articles category (e.g. 'politics', 'sport'), but different publishers categorise articles in different ways - for instance, the BBC has a combined 'Science and Environment' category, whereas The Guardian splits these into two distinct categories. Additionally, similar news articles can be filed under different categories, depending on the publisher. As the focus is on the article's content and not categorisation, it can be considered out of scope to investigate the interplay between different publisher's approach to categorising articles.

3.1.2 Sources

The Independent is one of the only online publishers to make available their entire archives. Using the methods mentioned in Section 3.1.3, 344,858 articles were collected, from 2011 to 2018. This 7-year period should prove a

[5] `https://archive.org/details/ArchiveIt-Collection-11171`

useful dataset to analyse a potentially changing landscape in news headlines' congruity.

The BBC has an 'On This Day' page[6] that has a very select archive from 1950-2005, and analysing these articles could produce some interesting results. However, each of these articles will have been hand-picked (as evidenced by the 'In Context' notes alongside each article) and only represent historic world news events. Therefore, these articles will not provide an unbiased representation of the period they encompass.

As well as archives, current news was also collected from a range of publishers. A varied range of UK publishers were selected to create a representative cross-section of the nation's media.

Table 2 shows the full list of data sources collected, as well as the time range they cover and the total records obtained.

Publisher	**Earliest**	**Latest**	**Raw**
BBC On This Day	1950-01-21	2005-12-11	1857
The Independent	2011-01-01	2018-05-28	317,135
BBC	2020-09-17	2021-01-01	4142
Daily Mail	2020-09-26	2021-01-02	14726
The Guardian	2020-09-17	2021-01-01	4078
Huffington Post	2020-10-09	2021-01-01	2920

Table 2: Extents of the data sources collected

3.1.3 Obtaining the Data

Several Python scripts were created to collect the data. For the daily news, the publishers' various RSS feeds were consulted, and for the archives, a more customised approach was taken.

These scripts utilise the BeautifulSoup library to parse each article's webpage and scrape them for the headline, date and body text. Each script had to be tailor-made to fit the page structure as each publisher builds their websites using different design patterns and different technologies. All the scripts used are available in this project's GitHub repository[7].

Additionally, some sites implemented a strict rate-limit on requests - to make a copy of The Independent's archive took around six months to complete, scraping one article every 15 seconds.

[6] `http://news.bbc.co.uk/onthisday`

[7] `https://github.com/jacobbarrow/honours/tree/master/data-collection`

These scripts ran on a Raspberry Pi for around 100 days, from 2020-09-18 to 2020-01-02. Except for the Independent archive script, which ran continuously, a cron job was used to run each script once per day.

3.1.4 Ethics and Copyright

Across a variety of datasets, 345 thousand articles were collected for analysis, which is a substantial amount of data and represents many individual journalists and news publishers' work.

While automated techniques were used to collect the data, everything collected was publically accessible. In addition, it is legal to make a digital copy of copyrighted data for non-commercial research [8]. Even so, care still needs to be taken in the obtainment of the data in order to avoid overloading or altering the regular service of these archives. As mentioned above, requests were rate-limited to avoid inadvertent denial of service attack and spread out over a long period of time. Additionally, the rolling news was only collected once per day, at 1 a.m., to minimise the scraping's impact.

3.1.5 Cleanliness

While a bespoke scraper was created for each site, on some articles, publishers used different page structures or included certain elements (such as infographics) that the scraper did not know how to handle. As a result, several collected articles have erroneous text in them, such as unformatted lists of tweets or social media comments.

To obtain a measure of cleanliness, a subset of 300 articles was created. From this small sample, 36 (12%) were corrupt or included content not part of the article's body text. Extrapolating this to the rest of the dataset means approximately 41,000 of the collected articles are 'dirty'.

Cleaning the dataset is out of this project's scope - the erroneous content does not follow a set pattern and would be non-trivial to remove. Either human intervention or a well-trained neural network could be used to clean the dataset, or potentially a combination of both.

3.1.6 Compilation

Once collected, the different sources were compiled into a single SQLite database using a Python script [9]. The database structure is described in

[8] `https://www.gov.uk/guidance/exceptions-to-copyright#text-and-data-mining-for-non-commercial-research`

[9] `https://github.com/jacobbarrow/honours/blob/master/data-collection/compile.py`

table 3.

Field	Description
`id`	An auto-incrementing numerical id
`source`	The publisher the article was sourced from
`headline`	The article's headline
`body`	The article's body
`date`	The date the article was published

Table 3: Database structure of compiled articles

3.2 Data Labelling

Chesney et al. (2017) reviews the current datasets available to detect incongruent articles and concludes that while many are available and have some potential use, none are a good fit for the task. With a lack of a well suited labelled dataset, a bespoke one was created.

3.2.1 Generating a Dataset

To create a subset to be labelled, 150 articles were randomly selected from the collected articles' primary dataset. As this subset was taken before collection had been completed and the dataset finalised, the articles selected will not represent the most recent in the dataset. However, as the time frame excluded (around two months) is relatively insubstantial, this should have a trivial effect on the data quality.

3.2.2 Design and Implementation of Labelling Site

The site used to collect ratings for articles was bespoke - existing off-the-shelf survey solutions did not meet this approach's needs or would have to be heavily modified to suit the data.

The labelling site's design was kept minimalist and clean, ensuring that the volunteers would not find themselves hindered and could concentrate on the task at hand.

Two pages were created, one to show the consent and briefing information, the other to show an article and allow a user to select a rating. Appendix A.C shows the form controls used to rate each article, and a full view of both the page's contents can be seen on the project's GitHub repository [10].

[10] `https://github.com/jacobbarrow/honours/tree/master/data-labelling/views`

As well as the rating (codified as an integer from 0-6), a javascript file measured the time someone took to read and rate the article. This will allow for a more extensive analysis of the data and aid in compiling the final labelled set.

The site used Flask, a Python framework, to serve the pages and interact with the SQLite database.

3.2.3 Ethics

While no personal information is collected, and it is impossible to identify an individual from the data, the site asks volunteers to participate in research. Therefore, ethical approval must be acquired.

Appendix A.A shows the participant information sheet shown to all participants before they proceeded to the rating.

Appendix A.B is the ethical approval form that details the extent of the information collected and how it will be processed and retained.

3.2.4 Analysis of Labelled Data

The site was distributed through several channels (friends, online forums), and 159 ratings were received. Unfortunately, this represents roughly one rating per article, which is not enough to form an accurate picture of incongruence - multiple ratings would be required so an average could be taken. This means the labelled dataset cannot be used in the training and analysis of a classifier.

3.3 Congruence Experimentation

To create the classifier, the efficacy of individual approaches needs to be evaluated. The general structure of each experiment run takes the following form:

1. Implement the classifier approach under test
2. Compile and load the dataset according to the parameters discussed in section 3.3.1
3. Run the algorithm on each article in the dataset, measuring the resultant classification
4. Record the accuracy of the classifier using the dataset as a baseline truth

5. Calculate the significance values of the outcome, and optionally plot a chart to represent the results, if appropriate

3.3.1 Experimentation Dataset

At the time of experimentation, the bespoke labelled dataset was not yet complete, as not enough ratings had been given to calculate an average rating per article. In lieu of this, the FNC-1 dataset was used as a gold standard against which to measure the preliminary experiments. The dataset contains articles labelled as `agree`, `disagree`, `discuss`, or `unrelated`. Articles labelled as `discuss` were discarded from the set before experimentation - if an article discusses the content, it could ultimately either agree or disagree with the headline, so it provides no meaningful information considering the scope of this project. Articles tagged as `unrelated` were also discarded, as they comprised of body texts matched with headlines from a different article, which will not be present in the articles collected. Once the FNC-1 dataset was cleaned, 843 labelled articles remained.

3.3.2 Sentiment Analysis

The VADER sentiment analyser[11] was used for experimentation - it would not be the best use of time to design a bespoke analyser if no trend was discovered in the dataset. The code used to perform this experiment is displayed in appendix B.A.

VADER produces 3 measures of sentiment - negativity, positivity and neutrality - as well as a compound value of all of them. To visualise the dataset, the percentage difference of the headline and the body text's measures was plotted. The results are shown in figure 4. Table 4 shows the results of a significance test; the data had a non-Guassian distribution so a Mann-Whitney U test was used.

Sentiment	**p-value**
Positive	0.20988
Negative	0.0667
Neutral	0.37534
Compound	0.08266

Table 4: Sentiment analysis significance

No clear trend in the difference between articles that agree and disagree is visible in the plot. Additionally, as none of the p-values is below 0.05, the results lack significance and must be rejected. This quite strongly shows there is no correlation between the difference in the sentiment of headlines

[11] `https://github.com/cjhutto/vaderSentiment`

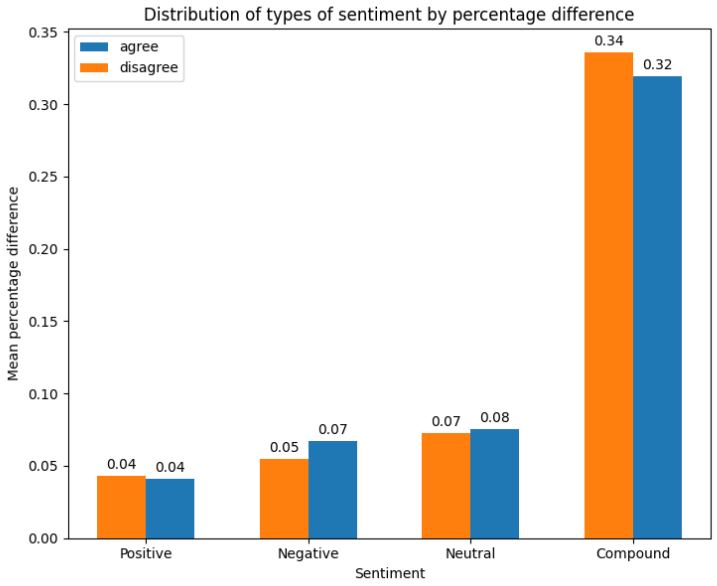


Figure 4: Distribution of sentiment types in the FNC-1 dataset

and articles, and as a result, sentiment analysis will not provide a meaningful use in the creation of the classifier.

This result is not surprising - the difference between sentiment and congruence was covered in section 2.5.2. It is possible for two pieces of text to have different sentiments and be in concurrence, or to have similar sentiments and be opposed in meaning, which means sentiment analysis is not suited for this approach.

3.3.3 Word Vectorisation

Another approach tested was word vectorisation - codifying a token as a vector comprised of numbers representing a certain quality of the word's meaning, using the context windows discussed in section 2.5.3.

Gensim's Python implementation of word2vec[12] was trained on articles from the raw dataset. As the training data is only used to give word2vec an understanding of how words relate to each other and not used to determine congruence, the data does not need to be labelled. Ten iterations of learning and 250,000 articles were used to train a model with 300 nodes.

The vectorisation model created is of good quality, and able to perform

[12] `https://radimrehurek.com/gensim/models/word2vec.html`

mathematical operations on words, such as the often-used example:

$$\text{Woman} + \text{King} - \text{Man} = \text{Queen}$$

With the model trained, it can be used to evaluate the labelled FNC dataset. For each article in the database, an average vector was calculated for both the headline and the body. This was achieved by summing up each word's vector and dividing by the number of words in the text. The similarity of these vectors was then computed using the `cosine_similarites` function in Gensim, which calculates the cosine difference of two vectors. The average difference for both articles that agree and disagree was obtained. The code used to perform this experiment is displayed in appendix B.B.

For articles labelled as agree, the average vector difference was 0.5175, and those labelled disagree had a difference of 0.5050, with a p-value of 0.19308. Like with the sentiment analysis experiment, the lack of significance and the negligible difference between each average shows that this is an ineffective way to determine congruence. This could be because only a few of the words in an article will point to incongruence - by obtaining an average of all of them, the text's meaning will be lost in the noise.

3.3.4 tf-idf

To ascertain the practicality of using tf-idf to detect incongruence, the algorithm set out in section 2.5.5 was implemented in Python. To get an informal understanding of the implementation's efficacy, it was used to determine the relevance of a range of keywords to an article about a house fire caused by fireworks. The results are shown in table 5. As the table shows, the implementation is working as expected - more relevant words, such as 'ablaze', have higher tf-idf values, indicating a higher relevance, and terms such as 'and' and 'football' have a much lower, negligible value.

Term	**tf-idf value**
ablaze	421.500
fireworks	168.600
explosive	93.667
the	0.089
and	0.058
football	0.000

Table 5: tf-idf values for a range of terms

To apply the algorithm to the problem, tf-idf was used to calculate each word's relevance in an article's headline to its body text. A mean was then taken of those values, as shown below. A full code listing is available in the Github repository.[13]

[13] `https://github.com/jacobbarrow/honours/tree/master/experiments/tfidf.py`

```
for article in articles.values():
    tfidfs = []
    for word in article['headline']:
        tfidfs.append(tfidf(word, article['body'], domain))

    mean = numpy.mean(tfidfs)
```

The mean will give an overall relevance between the headline and body of an article. The means were classified according to the article's labelled stance (agree/disagree) and then analysed. For articles labelled as agree, the average tf-idf mean of their headlines was 0.1167. For those labelled disagree, the average was 0.0718. While this difference may initially seem promising, with a p-value of 0.26281, the results must be rejected.

This result is disappointing but expected; tf-idf is used to determine the relevance of a word to a document and does not account for the document's overall sentiment. For instance, consider the following two statements:

- Mauve is a trendy colour to paint a house
- The worst colour to paint a house is mauve

The term 'mauve' appears the same number of times for both of them, so according to tf-idf will have identical relevance. However, each statement's sentiments are opposed, something unable to be determined by just the relevance alone.

Additionally, using the mean of all headline values may be 'muddying the waters' of the relevancy, as happened in the word vectorisation experiment. However, this may not be as strong a factor, as the difference between a relevant word and an irrelevant word is large, as seen in table 5, and so would have a more significant effect on the mean.

3.4 Visualisation

Creating a platform to visualise and identify trends using the NLP approaches discusses and used in section 3.3 will allow for ease of analysis, demonstrate areas for further work, and aid other researchers with the management of the dataset.

In order to be effective, the visualiser should:

- Allow for a downloadable export of the analysed data
- Be able to generate graphs, using a range of NLP approaches to create data points based on the dataset

- Have a non-obtrusive, simple design
- Allow a user to identify trends in the dataset

3.4.1 Design and Frontend Implementation

To keep things simple, the design of the visualiser was kept to a one-page layout. Appendix C.A shows a rudimentary wireframe of this layout.

The front end was written in HTML and CSS, with accessibility as a key concern. Semantic HTML5 elements were used, enhancing screen reader navigation and comprehension, and the form has a logical structure. Additionally, changes to the dynamic element of the form (changing the analysis technique used) were broadcast using the ARIA standard[14]. These ARIA regions were updated using JavaScript.

The bar chart was generated using chart.js[15], an open-source JavaScript library that allows for real-time updates, which was important - in order to identify trends, a moving average was used, which can be adjusted without reanalysing the data.

Appendix C.B shows the front-end implementation of the visualiser.

3.4.2 Backend Implementation

The backend of the visualisation site was created using Python, with Flask to serve the content. This allowed for quick prototyping and easy database interaction - the articles were stored in an SQLite database (compiled in section 3.1.6).

Sentiment analysis was the only approach technique implemented in the visualisation. This decision was made as each analysis method would require a bespoke approach to chart the data, which would not have been practical to produce in the interests of project management. Additionally, sentiment analysis produced the most robust and informative trends out of the approaches tested from informal experimentation.

To pass data back to the front-end, a `/data` route was created. When queried via an AJAX request, it returns a JSON representation of the last analysis that was run. While this means further modification would require regular use at scale, it allowed for an efficient front-end development, as a new analysis did not have to be conducted for each minor change. This same concept allows a user to save the data - a `/download` route returns the JSON with a header that forces the browser to download it to a file.

[14] `https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/ARIA_Live_Regions`

[15] `https://www.chartjs.org/`

3.4.3 Analysis

Several improvements could be made to the visualiser. For instance, the only implemented approach was sentiment analysis, as mentioned above. However, the code has been written to be extensible, and further methods can be added with relative ease.

The visualiser is relatively slow - for larger analyses, with tens of thousands of articles, it can take several minutes, during which time the browser hangs. This makes for a poor user experience, as there is no feedback on progress. To overcome this, a queue system could be implemented, where each analysis forms a job, which a user can check in on.

However, the visualiser meets the majority of the specification. It allows for the analysed data to be downloaded, has a simple, understandable design, and lets users manipulate the data to find trends.

4 Analysis and Discussion

5 Analysis of Data

The visualiser created in section ?? was used to analyse the trend in sentiment from different perspectives.

5.1 Longitudinal analysis

The Independent archive scraped contains articles from 2011 to 2018, offering just over seven years of news. When the headline and body's sentiment were initially analysed, there was a high noise level present in the data, as shown in appendix C.C. However, a rolling average was applied using the smoothing feature of the visualiser, and a more apparent trend appeared, shown in figure 5.

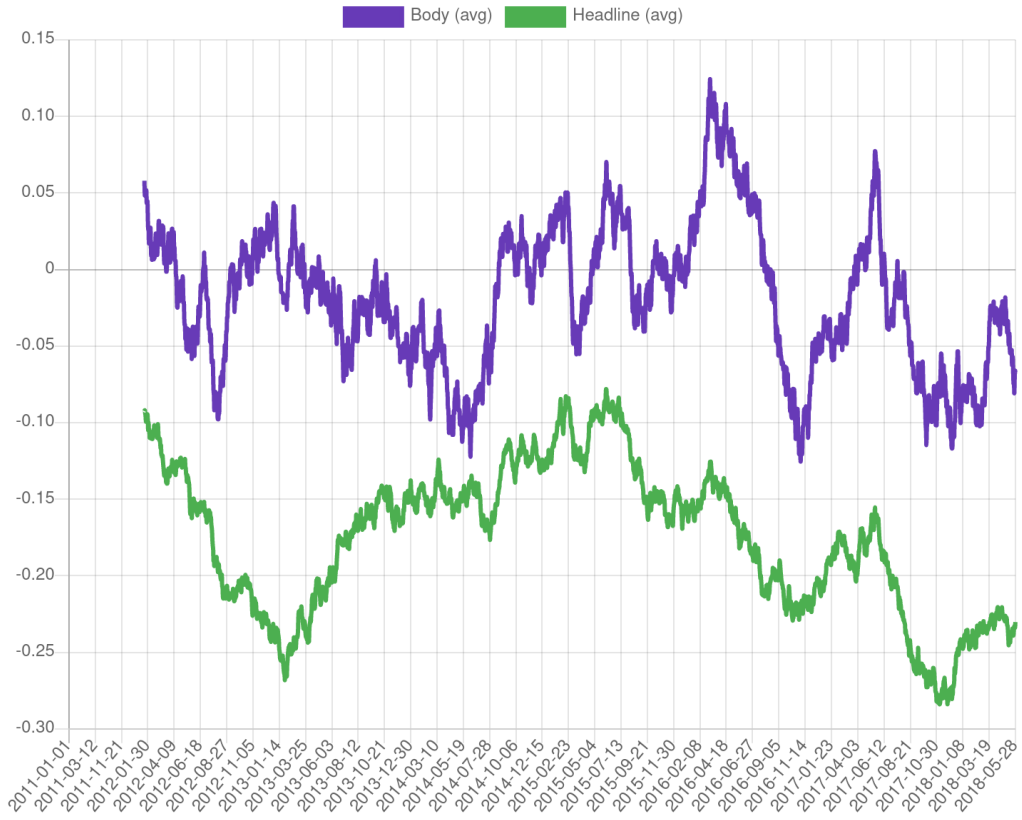


Figure 5: A chart to show the difference between headline and body sentiment of the Independent archive.

The first thing of note in the above chart is that the headline is, on average, consistently more negative than the body of a news article. This trend supports the study conducted by Arango-Kure, Garz, and Rott (2014), who determined that bad news sells more magazines than good news. This finding means that a news story's negative aspects are more likely to be exaggerated and played up in the article's headline to increase revenue.

Although an average article's headline is has a more negative sentiment than the article's content, there is a direct relationship between the two - for instance, when the sentiment of the body peaks in June 2017, the headline's sentiment mirrors this with a peak of its own. An exception to this is the dip of headline sentiment in 2013, which is not reflected in the body's trend.

The peaks and troughs in the data can also be linked to national and global events. The UK voted to leave the European Union on the 23rd of June 2016, followed by a period of political and economic instability. On the chart, a steep decline in the sentiment of articles bodies can be seen in June. As The Independent is a paper with a pro-market stance, it makes sense that their reporting's negativity mimics the strong downward trend of the FTSE. However, due to the rolling average in place, the data points cannot be tied down to a specific date, so any connection between the trends shown and real-life events is spurious.

5.2 Cross sectional analysis

Articles from a cross-section of news sources had been collected from a 3-month time period. As with the previous chart, when the headline sentiment of each of these sources was analysed, it produced a very incoherent and noisy output, as shown in appendix C.D. However, some clear trends surfaced when a rolling average was applied, shown below in figure 6. Note that while the visualiser was used to generate the raw data, the resultant graph was created using a separate script [16]. This is because the visualiser has been designed to compare different analysis approaches instead of different data sources, so it is unable to plot different sources on the same chart.

One of the trends that appears in the chart is that the Huffington Post has the most positive headlines, whereas the Daily Mail has the most negative headlines. A possible explanation for this is that the Daily Mail typically manufactures outrage, whereas the Huffington Post generally produces list-based articles and stories not generally linked to current affairs (e.g., 'Comic

[16] `https://github.com/jacobbarrow/honours/blob/master/visualisation/comparison.py`

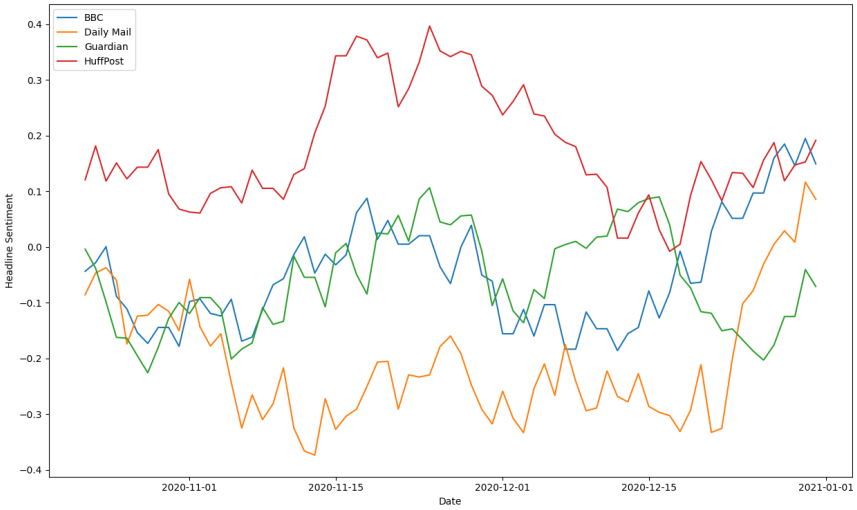


Figure 6: A chart to show the difference in headline sentiment between several news outlets

Relief: 7 Highlights From This Year's Red Nose Day Telethon'[17] and 'The Bread Cutting Hack You Never Knew You Kneaded'[18]).

Another interesting trend is the peak in the average sentiment of all sources towards the end of the year. This could be because of the Christmas holidays, or even because the general sentiment at the time is that the new year would mark a change in the tide of the pandemic [19,20].

[17] https://www.huffingtonpost.co.uk/entry/red-nose-day-2021-best-bits-funniest-highlights_uk_6055be48c5b6f12839d47dfe

[18] https://www.huffingtonpost.co.uk/entry/bread-cutting-hack_uk_60547026c5b6d6c2a2a663b1

[19] https://www.independent.co.uk/news/uk/home-news/covid-vaccine-climate-change-remote-working-b1768098.html

[20] https://www.aarp.org/health/conditions-treatments/info-2020/coronavirus-2021.html

6 Conclusion

From the experiments conducted in section 3.3 and the review of the literature in section 2, it is clear the statistical natural language processing methods are not sufficient to identify the nuances incongruence.

While classical approaches can identify the general gist of a piece of text, articles can become incongruent through a subtle remark or a seeming offhand turn of phrase, which cannot be detected using the methods investigated in the project.

6.1 Future Work

While this project has failed to create a classifier capable of detecting incongruent news headlines, in every other respect, it has been a success; the groundwork has been laid for future research and development in many forms.

Firstly, several approaches have been considered and discounted through empirical testing, and a framework for experimentation has been proposed. This will save future researchers the time and effort of starting from scratch and provide a baseline standard against which to compare their results. It has been suggested that a machine learning approach, utilising a neural network, could be an appropriate next step.

Although a substantial labelled data source was not created, the framework for labelling and a website to allow individuals to rate articles has been implemented. With either a wide network of willing volunteers or a small amount of funding, a large labelled dataset can be created, which can then be used to train a classifier or act as a gold standard of truth.

Additionally, a visualiser has been created. This proof-of-concept website has been made with future development in mind, and while only one form of experiment has been implemented (sentiment analysis), it can be easily extended. This will allow future researchers to analyse their results without having to create a bespoke graphing platform.

Finally, a vast dataset of 345k articles has been produced spanning several decades and from various sources. Included within these articles is a snapshot of the UK media as it reported on the most deadly pandemic to impact the nation since influenza in 1919. This wealth of data will hold a plethora of trends and insights waiting to be uncovered.

6.2 Final Thoughts

Throughout this project it has become clear that statistical natural language techniques are not capable of detecting the twists and turns that news editors

employ, intentionally or not, to mislead and decieve their readers. There's just too much nuance involved, and subtle misdirection is at play that conceals the true meaning of an article. This is a non-trivial problem that, if solveable, will require a great deal of research and computational power to overcome.

Nonetheless, the project has explored the breadth of NLP, as well as looking at classical techniques in depth. It has produced a range of outputs, from tools to help others collect and visualise data, to a substantial set of articles that span over half a century of national news. It is hoped that this work will be continued and future researchers uncover more trends and produce a classifier capable of detecting incongruent articles.

References

Adams, R. (2006). Textual entailment through extended lexical overlap. In *Proceedings of the second pascal challenges workshop on recognising textual entailment* (pp. 128–133).

Allahyari, M., Pouriyeh, S., Assefi, M., Safaei, S., Trippe, E. D., Gutierrez, J. B., & Kochut, K. (2017). Text summarization techniques: a brief survey. *arXiv preprint arXiv:1707.02268*.

Allcott, H., & Gentzkow, M. (2017, May). Social media and fake news in the 2016 election. *Journal of Economic Perspectives*, *31*(2), 211–236. Retrieved from `https://doi.org/10.1257/jep.31.2.211` doi: 10.1257/jep.31.2.211

Arango-Kure, M., Garz, M., & Rott, A. (2014). Bad news sells: The demand for news magazines and the tone of their covers. *Journal of Media Economics*, *27*(4), 199–214.

Awrejcewicz, J. (2012). Geometry of masses. In *Advances in mechanics and mathematics* (pp. 140–145). Springer New York. Retrieved from `https://doi.org/10.1007/978-1-4614-3791-8_3` doi: 10.1007/978-1-4614-3791-8_3

Banerjee, S., & Pedersen, T. (2003). The design, implementation, and use of the ngram statistics package. In *International conference on intelligent text processing and computational linguistics* (pp. 370–381).

Banko, M., Mittal, V. O., & Witbrock, M. J. (2000). Headline generation based on statistical translation. In *Proceedings of the 38th annual meeting of the association for computational linguistics* (pp. 318–325).

Bommana, H. (2019). *Deep nlp: Word vectors with word2vec.* Retrieved from `https://medium.com/deep-learning-demystified/deep-nlp-word-vectors-with-word2vec-d62cb29b40b3`

Chesney, S., Liakata, M., Poesio, M., & Purver, M. (2017). Incongruent headlines: Yet another way to mislead your readers. In *Proceedings of the 2017 emnlp workshop: Natural language processing meets journalism* (pp. 56–61).

Church, K. W. (2017). Word2vec. *Natural Language Engineering*, *23*(1), 155–162.

Digirolamo, G. J., & Hintzman, D. L. (1997). First impressions are lasting impressions: A primacy effect in memory for repetitions. *Psychonomic Bulletin & Review*, *4*(1), 121–124.

Ecker, U. K., Lewandowsky, S., Chang, E. P., & Pillai, R. (2014). The effects of subtle misinformation in news headlines. *Journal of experimental psychology: applied*, *20*(4), 323.

Gabielkov, M., Ramachandran, A., Chaintreau, A., & Legout, A. (2016). Social clicks: What and who gets read on twitter? In *Proceedings of the 2016 acm sigmetrics international conference on measurement and modeling of computer science* (pp. 179–192).

Hayashi, Y., & Yanagimoto, H. (2018). Headline generation with recurrent neural network. In *New trends in e-service and smart computing* (pp. 81–96). Springer.

Jackson, P. (2002). *Natural language processing for online applications : text retrieval, extraction, and categorization*. Amsterdam Philadelphia, PA: John Benjamins Pub.

Karlsson, M., & Strömbäck, J. (2010). Freezing the flow of online news: Exploring approaches to the study of the liquidity of online news. *Journalism Studies*, *11*(1), 2–19.

Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, *5*(1), 1–167.

Lopez, G. (2016, Dec). *A man fired shots at a dc pizzeria while "investigating" a bizarre fake news conspiracy theory.* Vox. Retrieved from `https://www.vox.com/policy-and-politics/2016/12/5/13839178/comet-fake-news-pizzagate-gunman`

Mahoney, J. (2015, Jun). *A complete taxonomy of internet chum.* Retrieved from `https://www.theawl.com/2015/06/a-complete-taxonomy-of-internet-chum/`

Manjesh, S., Kanakagiri, T., Vaishak, P., Chettiar, V., & Shobha, G. (2017). Clickbait pattern detection and classification of news headlines using natural language processing. In *2017 2nd international conference on computational systems and information technology for sustainable solution (csitss)* (p. 1-5).

Manning, C., & Schutze, H. (1999). *Foundations of statistical natural language processing*. MIT press.

Maybury, M. (1999). *Advances in automatic text summarization*. MIT press.

Molek-Kozakowska, K. (2013). Towards a pragma-linguistic framework for the study of sensationalism in news headlines. *Discourse & Communication*, *7*(2), 173–197.

Nadeau, D., & Sekine, S. (2007). A survey of named entity recognition and classification. *Lingvisticae Investigationes*, *30*(1), 3–26.

Navarro, G. (2001). A guided tour to approximate string matching. *ACM computing surveys (CSUR)*, *33*(1), 31–88.

Park, K., Kim, T., Yoon, S., Cha, M., & Jung, K. (2020). Baitwatcher: A lightweight web interface for the detection of incongruent news headlines. *arXiv preprint arXiv:2003.11459*.

Piotrkowicz, A., Dimitrova, V., & Markert, K. (2017). Automatic extrac-

tion of news values from headline text. In *Proceedings of the student research workshop at the 15th conference of the european chapter of the association for computational linguistics (eacl srw 2017)* (pp. 64–74).

Pomerleau, D., & Rao, D. (2017). *Fake news challenge stage 1 (fnc-i): Stance detection.* Retrieved from `http://www.fakenewschallenge.org/`

Potthast, M., Köpsel, S., Stein, B., & Hagen, M. (2016). Clickbait detection. In *European conference on information retrieval* (pp. 810–817).

Pradhan, N., Gyanchandani, M., & Wadhvani, R. (2015). A review on text similarity technique used in ir and its application. *International Journal of Computer Applications*, *120*(9).

Qian, G., Sural, S., Gu, Y., & Pramanik, S. (2004). Similarity between euclidean and cosine angle distance for nearest neighbor queries. In *Proceedings of the 2004 acm symposium on applied computing* (pp. 1232–1237).

Rajaraman, A., & Ullman, J. D. (2011). Data mining. In *Mining of massive datasets* (pp. 1–17). Cambridge University Press. Retrieved from `https://doi.org/10.1017/cbo9781139058452.002` doi: 10.1017/cbo9781139058452.002

Reis, J., Benevenuto, F., de Melo, P. O., Prates, R., Kwak, H., & An, J. (2015). Breaking the news: First impressions matter on online news. *arXiv preprint arXiv:1503.07921*.

Tan, J., Wan, X., & Xiao, J. (2017). From neural sentence summarization to headline generation: A coarse-to-fine approach. In *Ijcai* (Vol. 17, pp. 4109–4115).

Xu, S., Yang, S., & Lau, F. (2010). Keyword extraction and headline generation using novel word features. In *Proceedings of the aaai conference on artificial intelligence* (Vol. 24).

Appendices

A Data Labelling Ethics

A.A Participant Information Sheet

An investigation into incongruous headlines

Participant Information

I'm working on creating an algorithm that can detect if a news article's headline matches its content. To do this, I need a collection of articles that have already been labelled with a rating of incongruence, both to determine if the algorithm is working, and also to train it to be more accurate.

In this study, you will be shown a news article, then be asked to read through it and rate how well you think the headline matches the content.

You can rate as many articles as you want, and stop participating at any time.

The only information collected when rating an article is the rating you give and the length of time taken to reach that rating. You can take as long as you feel necessary to read through and rate each article.

The information collected from your participation will be made publically available. No identifying information regarding your participation is collected, so any rating you provide will be completely anonymous.

If you are below the age of 18, please do not participate in this study.

Informed Consent

By taking part in the study, you confirm you agree with the following:

- I freely and voluntarily consent to be a participant in this research to be conducted by Jacob Barrow, who is an undergraduate student in the Edinburgh Napier School of Computing.
- I have been informed of the broad goal of this research study. I have been told what is expected of me and that I can spend as long as I feel comfortable participating
- I have been told that my responses will be anonymised. My name will not be linked with the research materials, and I will not be identified or identifiable in any report subsequently produced by the researcher. I have been told that these data may be made publically available.
- I also understand that if at any time during the survey I am free to leave. That is, my participation in this study is completely voluntary, and I may withdraw from it at any time without negative consequences.
- In addition, should I not wish to rate a particular article, I am free to decline.
- I understand that I can contact Jacob Barrow at 40337360@live.napier.ac.uk if I have any questions regarding the survey.
- I have read and understood the above and consent to participate in this study. My participation is not a waiver of any legal rights. Furthermore, I understand that I will be able to keep a copy of this consent form for my records.

A.B RI Approval

Application for Cross-University Ethical Approval

1. Research Details

Name:	Jacob Barrow
School or Professional service department:	School of Computing
Email:	40337360@live.napier.ac.uk
Contact number:	07437767306
Project Title:	An investigation into news articles' incongruence
Start Date:	17/09/2021
Duration of Project:	6 months
Type of Research: UG	

2. Screening Questions

Please answer the following questions to identify the level of risk in the proposed project:

If you answer 'No' to all questions, please complete Section 3a only.
If you have answered 'Yes' to any of the questions 5-14 please complete Section 3a and 3b.
If you have answered 'Yes to any of the questions 1-4, complete all of Section 3.

	You Must Answer All Questions	**Yes**	**No**
1.	Is the research clinical in nature?	☐	X
2.	Is the research investigating socially or culturally 'controversial' topics (for example pornography, extremist politics, or illegal activities)?	☐	X
3.	Will any covert research method be used?	☐	X
4.	Will the research involve deliberately misleading participants (deception) in any way?	☐	X
5.	Does the Research involve staff or students within the University?	☐	X
6.	Does the Research involve vulnerable people? (For example people under 18 or over 70 years of age, disabled (either physically or mentally), those with learning difficulties, people in custody, migrants etc).	☐	X
7.	Is the information gathered from participants of a sensitive or personal nature?	☐	X
8.	Is there any realistic risk of any participants experiencing either physical or psychological distress or discomfort?	☐	X
9.	Have you identified any potential risks to the researcher in carrying out the research? (for example physical/emotional/social/economic risks?)	☐	X
10.	Are there implications from a current or previous professional relationship i.e. staff/student/line manager/managerial position that would affect the voluntary nature of the participation?	☐	X
11.	Will the research require the use of assumed consent rather than informed consent? (For example when it may be impossible to obtain informed consent due to the setting for the research – e.g. observational studies/videoing/photography within a public space)	☐	X
12.	Is there any risk to respondents' anonymity in any report/thesis/publication from the research, even if real names are not used?	☐	X

13.	Will any payment or reward be made to participants, beyond reimbursement or out-of-pocket expenses?	☐	X
14.	Does the research require external ethics clearance? (For example from the NHS or another institution)	☐	X
15.	Does the research involve the use of secondary data?	X	☐

3A. Details of Project

In this section please provide details of your project and outline data collection methods, how participant consent will be given as well as details of storage and dissemination.

Please give a 300 word overview of the research project	
In order to train and evaluate an algorithmic approach to detecting incongruence between an article's headline and it's body, participants will be asked to read a news article and provide a rating of how well the headline matches the body text. As well as the rating, the time a participant takes to reach that rating will also be recorded, to allow for further analysis of the results and provide legitimacy – if someone has taken more time to read the article, this could provide some insight into the variations in ratings.	
Data Collection	
1.	**Who will be the participants in the research?**
	Anonymous participants, sourced from various locations (friendship groups, online forums)
2.	**How will you collect and analyse the research data? (please outline all methods e.g. questionnaires/focus groups/internet searches/literature searches/interviews/observation)**
	The data will be collected using a custom-made website on a secured server
3.	**Where will the data will be gathered (e.g. in the classroom/on the street/telephone/on-line)**
	Online
4.	**Please describe your selection criteria for inclusion of participants in the study**
	Adults above the age of 18
5.	**If your research is based on secondary data, please outline the source, validity and reliability of the data set**
	The news articles participants will be reviewing have been collected from a range of online news sites (BBC, The Daily Mail, The Independent, Huffington Post).
Consent and Participant Information	
7.	**How will you invite research participants to take part in the study? (e.g. letter/email/asked in lecture)**
	Posts made on online forums and facebook groups

8.	**How will you explain the nature and purpose of the research to participants?**
	Before they proceed to the rating section of the site, participants will be shown a description of the study, the nature of data to be collected, and how it will be stored and made available
9.	**How will you record obtaining informed consent from your participants?**
	A participant will not be able to access the collection aspect of the website without ticking a checkbox confirming they understand and agree to the terms of the study.
Data storage and Dissemination	
10.	**How and in what format will data be stored? And what steps will be taken to ensure data is stored securely?**
	The data collected will only be a number value that represents the participant's rating, and the length of time (in seconds) taken to arrive at that rating.
11.	**Who will have access to the data?**
	After the study, the dataset will be made public
12.	**Will the data be anonymised so that files contain no information that could be linked to any participant?**
	Yes – no identifiable data will be stored
13.	**How long will the data be kept?**
	Indefinitely
14.	**What will be done with the data at the end of the project?**
	It will be made publicly accessible
15.	**How will the findings be disseminated?**
	The findings will be reported in the dissertation of my honour project
16.	**Will any individual be identifiable in the findings?**
	No

3B. Identification and Mitigation of Potential risks

This section is designed to identify any realistic risks to the participants and how you propose to deal with it.

1. **Does this research project involve working with potentially vulnerable individuals?**

Group	Yes	NO	Details (for example programme student

			enrolled on, or details of children's age/care situation, disability)
Students at Napier	☐	☐	
Staff at ENU	☐	☐	
Children under 18	☐	☐	
Elderly (over 70)	☐	☐	
Disabled	☐	☐	
Migrant workers	☐	☐	
Prisoners / people in custody	☐	☐	
Learning difficulties	☐	☐	

2. **If you are recruiting children (under 18 years) or people who are otherwise unable to give informed consent, please give full details of how you will obtain consent from parents, guardians, carers etc.**

3. **Please describe any identified risks to participants or the researcher as a result of this research being carried out**

4. **Please describe what steps have been taken to reduce these identified risks? (for example providing contact details for appropriate support services (e.g. University Counselling, Samaritans), reminding participants of their right to withdraw and/or not answering questions, or providing a full debriefing to participants)**

5. **If you plan to use assumed consent rather than informed consent please outline why this is necessary**

6. **If payment or reward will be made to participants please justify that the amount and type are appropriate (for example the amount should not be so high that participants would be financially coerced into taking part, or that the type of reward is appropriate to the research topic).**

3C. Justification of High Risk Projects

If you answered 'Yes' to the screening questions 1-4 this section asks for justification on the choice of research topic and methodology.

1. **If you have answered yes to question 1 please give a full description of all medical procedures to be used within the research and provide evidence that the project has obtained NHS ethical approval.**

2. **If you have answered yes to questions 2 (research into a controversial topic) please provide a justification for your choice of research topic, and describe how you would deal with any potential issues arising from researching that topic.**

3. **If you have answered yes to questions 3 or 4 (use of deception or covert research methods) please provide a justification for your choice of methodology, and state how you will mitigate the risks associated with these approaches**.

Declaration	
☐	I consider that this project has no significant ethical implications to be brought to the attention of Research Integrity Committee
☐	I consider that this project may have significant ethical implications to be brought to the attention of the Research Integrity Committee
Researcher Signature:	**Date:**
Director of Studies/Supervisor/Principal Investigator Signature:	**Date:**

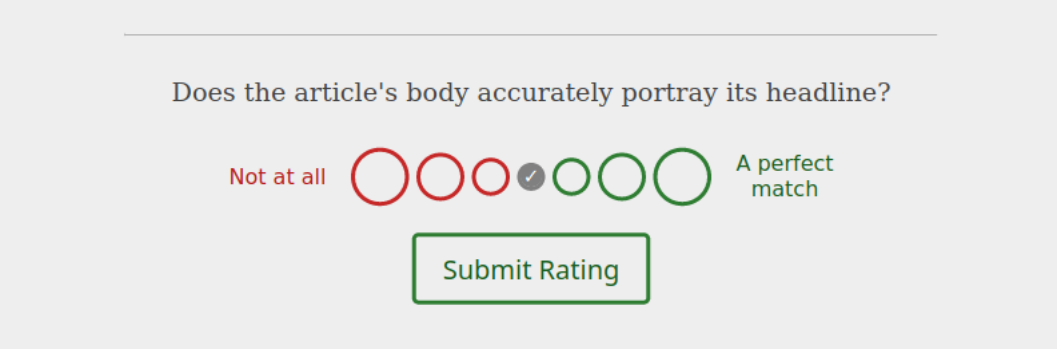
Checklist

All applications require the following to be submitted with the application form

Participant Information Sheet	X
Informed Consent Form	X
Interview/Survey Questions	X

A PDF information sheet and consent form are attached – a text version will appear on the website before participants are shown articles to rate, as well as a download link to the PDF file.

A.C Example Rating Form



B Code Listings

B.A Sentiment Analysis Experiment

```
import numpy

import matplotlib.pyplot as plt

from vaderSentiment.vaderSentiment import
    SentimentIntensityAnalyzer

import utils
from fnc import load

analyzer = SentimentIntensityAnalyzer()
articles = load.FNC()

STANCES = ['agree', 'disagree']
KEYS = ['pos', 'neg', 'neu', 'compound']
LABELS = ['Positive', 'Negative', 'Neutral', 'Compound']

# Set up an empty 2d dictionary
differences = {}
for stance in STANCES:
    differences[stance] = {}
    for key in KEYS:
        differences[stance][key] = []


def _autolabel(ax, bars):
    """Attach a text label above each bar in *rects*,
    displaying its height."""
    for bar in bars:
        height = bar.get_height()
        ax.annotate('{}'.format(round(height, 2)),
                    xy=(bar.get_x() + bar.get_width() / 2,
    height),
                    xytext=(0, 3),  # 3 points vertical
    offset
                    textcoords="offset points",
                    ha='center', va='bottom')


def generateVis(limit=None):
    for i, article in enumerate(articles.values()):
        headline_sentiment = analyzer.polarity_scores(article
    ['headline'])
        body_sentiment = analyzer.polarity_scores(article['
```

```
    body'])

        for key in KEYS:
            # Sentiment values range from -1 to 1, so take half the difference
            percentage = abs(headline_sentiment[key] - body_sentiment[key])/2
            differences[article['stance']][key].append(percentage)

        if limit and i > limit:
            break

    _, ax = plt.subplots()

    label_locations = numpy.arange(len(LABELS))
    bar_width = 0.4

    means = {}
    for index, stance in enumerate(STANCES):
        means[stance] = []
        for key in KEYS:
            means[stance].append(numpy.mean(differences[stance][key]))

        bar_location = label_locations + 0.2 - (bar_width * index)
        bars = ax.bar(bar_location, means[stance], bar_width, label=stance)
        _autolabel(ax, bars)

    for key in KEYS:
        significance = utils.calcSignificance(differences['agree'][key],
                                              differences['disagree'][key])
        p_value = significance['p']
        distribution = significance['distribution']

        print(f'{key}: {round(p_value, 5)} ({distribution})')

    ax.set_title('Distribution of types of sentiment by percentage difference')
    ax.set_ylabel('Mean difference')
    ax.set_xlabel('Sentiment')
    ax.set_xticks(label_locations)
    ax.set_xticklabels(LABELS)
    ax.legend()
    plt.show()
```

```
79
80
81 if __name__ == '__main__':
82     generateVis()
```

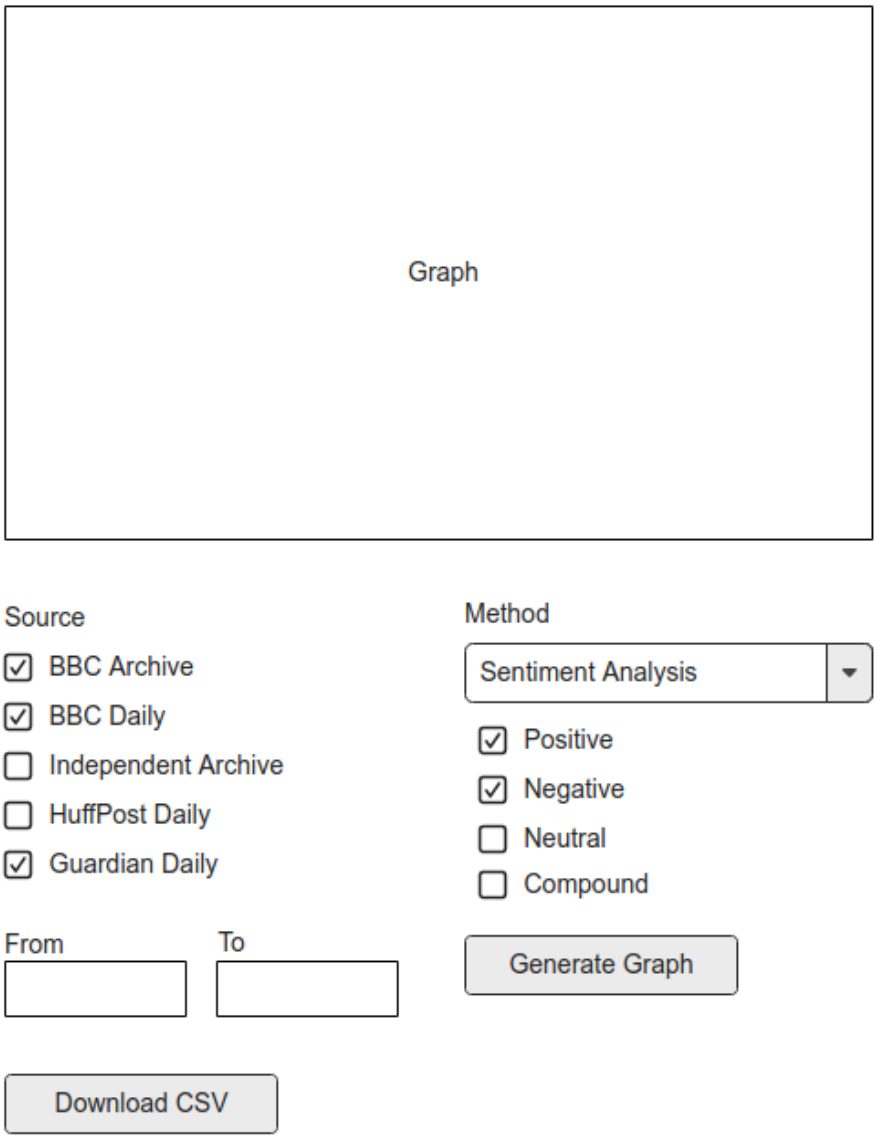
B.B Word Vectorisation Experiment

```
1  import numpy
2
3  from gensim.models import Word2Vec
4
5  import utils
6  from fnc import load
7
8
9  def getAverageVector(words):
10     total = 0
11     for word in words.split(' '):
12         try:
13             total += model.wv[word.lower()]
14         except KeyError:
15             pass
16
17     return total
18
19
20 articles = load.FNC()
21 model = Word2Vec.load('vectorisation_model/vector.model')
22
23 agree_differences = []
24 disagree_differences = []
25
26 for i, article in enumerate(articles.values()):
27     try:
28         headline_vec = getAverageVector(article['headline'])
29         body_vec = getAverageVector(article['body'])
30
31         difference = model.wv.cosine_similarities(
   headline_vec, [body_vec])[0]
32
33         if(article['stance'] == 'agree'):
34             agree_differences.append(difference)
35
36         else:
37             disagree_differences.append(difference)
38
39     except numpy.AxisError:
40         pass
41
```

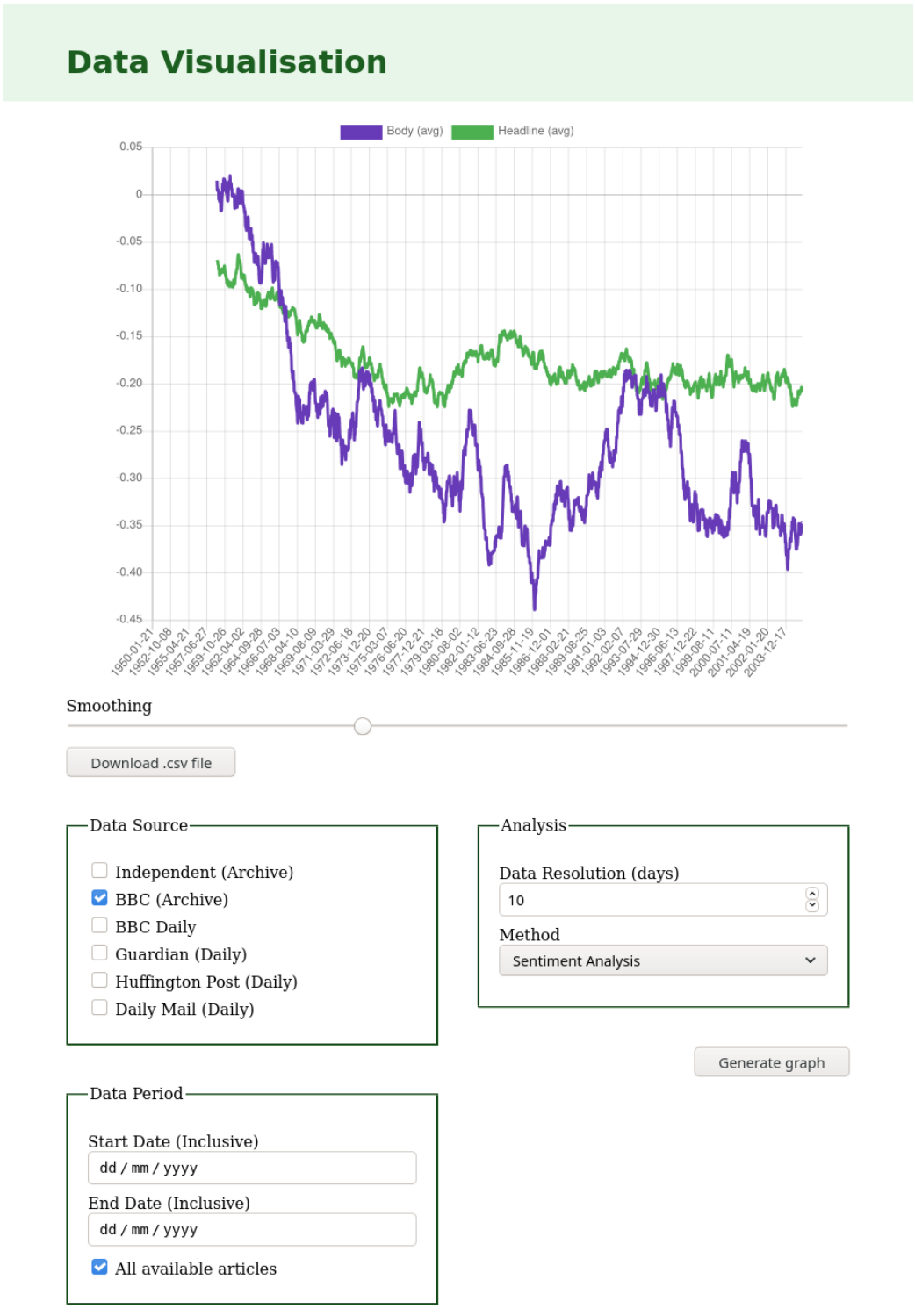
```
42 significance = utils.calcSignificance(agree_differences,
       disagree_differences)
43
44 p_value = significance['p']
45 distribution = significance['distribution']
46
47 print(f'Agree avg      {numpy.mean(agree_differences)}')
48 print(f'Disagree avg   {numpy.mean(disagree_differences)}')
49 print(f'Significance   {round(p_value, 5)} ({distribution})')
```

C Visualiser

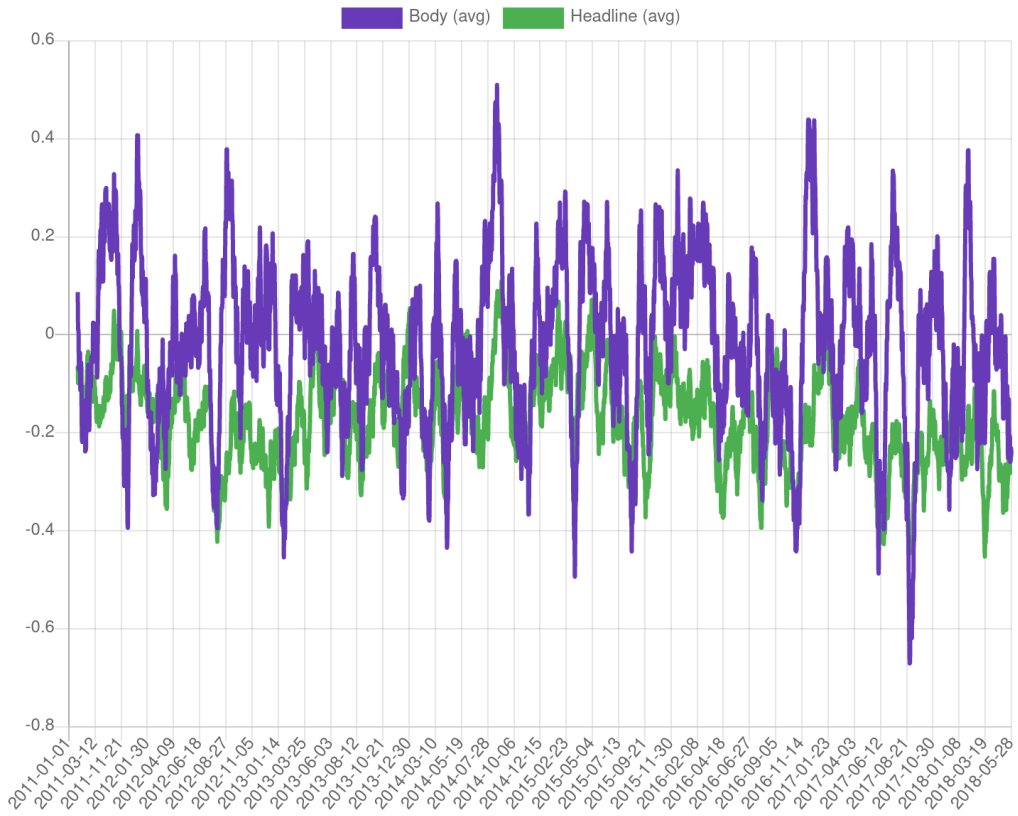
C.A Wireframe



C.B Implementation



C.C Independent Archive Raw Analysis



C.D Comparison of all sources Raw Analysis