Duck News Reporters: Automated fake news detection through contextual similarity comparison

COMP9491: Applied Artificial Intelligence — Project Report

Dhruv Agrawal
z5361800@unsw.edu.au

Duke Nguyen
z5398432@unsw.edu.au

Jim Tang
z5208565@unsw.edu.au

August 5, 2023

Todo list

1 Introduction

[Introduction] Describe the problem domain and aim of study, briefly introduce the developed methods and summarise your experimental findings

2 Related work

[Related work] **Dhruv**: Describe the current state-of-the-art or related literature in this problem domain

3 Methods

Figure 1 shows our mostly linear classification pipeline. After preprocessing and tokenization, we extract contextual articles which are fed into a similarity model to form our first feature. Additionally, non-latent features from raw text and BERT embeddings form the rest of our features. The concatenation of all the features are fed into our classification models which infers a binary classification label.

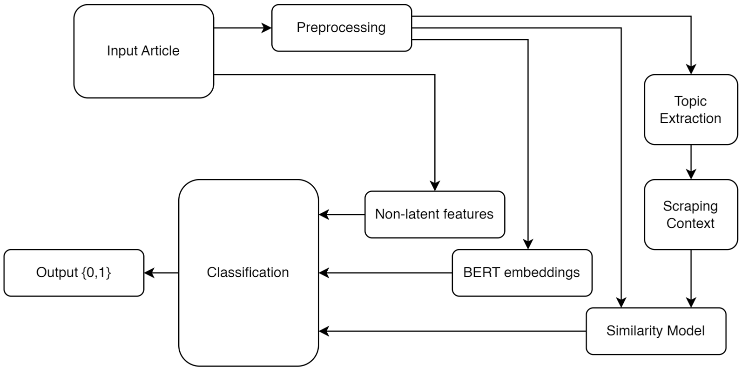


Figure 1: Our classification pipeline.

3.1 Preprocessing and tokenization

Before extracting any features, we will preprocess our input and convert the long form text into tokens. We perform the following preprocessing methods in order:

> **Remove non-ascii:** Our input articles contained unnecessary unicode tokens such as unicode double quotation marks. These can be removed safely since they do not add any extra semantics to the input articles and may confuse feature extraction.
>
> **Convert to lowercase:** In our research, we converted all text to lowercase. However upon further analysis, converting all text to lowercase hid acronyms such as "US" which could have affected the main themes of the text. Further, all proper nouns such as names and places were also hidden. We will discuss this limitation in Section 2.
>
> **Lemmatization:** We used the `nltk` [1] libaray to reduce words down to their lemma in the hopes of reducing the complexity within our text which may benefit feature extraction. This looks up the work in the WordNet corpus to get the lemma. Later in the research, we realised that this hypothesis may have not been accurate.
>
> Firstly the `nltk` library we were using does not automatically detect the part of speech and will by default, only lemmatize nouns. While it is arguably better for us to maintain the tense of nouns, we are technically not lemmatizing fully. Secondly, from more research, lemmatization may not be ideal for BERT embeddings since it removes some semantics that could be learnt by the BERT model. We will discuss these limitations further in Section 2.
>
> **Remove stopwords:** Stopwords were removed from the text in order to reduce complexity.

Apart from the above methods, we also tested removing punctuation. However, this was not used in the end since we added non-latent features to measure punctuation counts and also to maintain semantics for BERT.

After preprocessing, tokens are then generated based on any whitespace and punctuation in the remaining text. Table 1 shows samples of tokenized input articles.

3.2 Feature — Similarity model

One of the core aspects of our research was the ability to automatically gather articles that give some context to each input article. Our approach summarizes the input article so it can be used to find contextual articles. These articles can then be used for comparison to the input article.

ID	Article extract	Tokens
118_Real	FBI Director James Comey said Sunday that the bureau won’t change the conclusion it made in July after it examined newly revealed emails related to the Hillary Clinton probe. “Based on our review, we have not changed our conclusions that we expressed in July with respect to Secretary Clinton” Comey wrote in a letter to 16 members of Congress. [...]	[’fbi’, ’director’, ’james’, ’comey’, ’said’, ’sunday’, ’bureau’, ’change’, ’conclusion’, ’made’, ’july’, ’examined’, ’newly’, ’revealed’, ’email’, ’related’, ’hillary’, ’clinton’, ’probe’, ’.’, ’”’, ’based’, ’review’, ’,’, ’changed’, ’conclusion’, ’expressed’, ’july’, ’respect’, ’secretary’, ’clinton’,. . .]
15_Fake	After hearing about 200 Marines left stranded after returning home from Operation Desert Storm back in 1991, Donald J.Trump came to the aid of those Marines by sending one of his planes to Camp Lejuene, North Carolina to transport them back home to their families in Miami, Florida. Corporal Ryan Stickney was amongst the group that was stuck in North Carolina and could not make their way back to their homes. [...]	[’hearing’, ’200’, ’marines’, ’left’, ’stranded’, ’returning’, ’home’, ’operation’, ’desert’, ’storm’, ’back’, ’1991’, ’,’, ’donald’, ’j’, ’.’, ’trump’, ’came’, ’aid’, ’marines’, ’sending’, ’one’, ’plane’, ’camp’, ’lejuene’, ’,’, ’north’, ’carolina’, ’transport’, ’back’, ’home’, ’family’, ’miami’,. . .]

Table 1: Examples of preprocessing and tokenization extraction on items in dataset.

Summary extraction

To get the context articles, we need to summarize the main topic of our input article down to at most 10 keywords. We use the Python `gensim` [2] library which provides various topic modelling interfaces for text inputs. We use the `ldamodel` which implements Latent Dirichlet Allocation (LDA) to extract a single topic. LDA is a probabilistic model where the idea is you have a number of documents representing some latent topics characterized by a distribution over words. By feeding in the preprocessed sentences of our input article, we are able to get the main themes. We sort the output keywords by the probability they represent the topic then cap the amount of words to 10 at most.

For the scope of our research, we are able to perform manual validation of the summaries extracted to check the summary represented the article content well. Table 2 shows some samples of items in our dataset after applying LDA. We see that while the summaries extracted are not perfect, they still represent the general meaning of the article. Two common issues we saw were:

- Unordered words in the summary — words representing the topics seemed to be unordered. To a human reading the summary by itself, they might be able to see that the words are all keywords of the article but put together in a sentence, will not completely make sense. We hypothesize that this could have caused sub-optimal results when we started scraping articles using the summaries.
- Appearance of stop words and other meaningless non-topic words in the summary — As a flow on issue from our preprocessing, our summary was left with words such as “wa” (from “was”) or “ha” (from “has”). This would have impacted the meaning of our summary and later article scraping.

We will discuss the possibility of extracting better summaries using a more robust model in Section 6.1.

Article scraping

We feed the summary of the input article into Google News and collect the top three articles. We use Google News since it essentially provides a free PageRank algorithm which we can leverage to get the most popular articles during the time period. We will treat the articles we find as Real articles for purposes of comparison, i.e. an input article that is very different to our contextual article is likely to be Fake.

ID	Article extract	Summary
118_Real	FBI Director James Comey said Sunday that the bureau won't change the conclusion it made in July after it examined newly revealed emails related to the Hillary Clinton probe. "Based on our review, we have not changed our conclusions that we expressed in July with respect to Secretary Clinton" Comey wrote in a letter to 16 members of Congress. [...]	email review fbi clinton said july comey news new wa
15_Fake	After hearing about 200 Marines left stranded after returning home from Operation Desert Storm back in 1991, Donald J.Trump came to the aid of those Marines by sending one of his planes to Camp Lejuene, North Carolina to transport them back home to their families in Miami, Florida. Corporal Ryan Stickney was amongst the group that was stuck in North Carolina and could not make their way back to their homes. [...]	home marines trump wa stickney way north plane family

Table 2: Examples of summary extraction on items in dataset.

For our research, we will only manually feed in all summaries for our dataset. Our motivation for this research was to develop a tool that a user could potentially use to figure out if the current news they are reading contains misinformation. We acknowledge there exists APIs that provide either a wrapper around Google News or implement their own news search algorithm that we could have looked into. However, given the size of the dataset and our scope, this was not necessary to demonstrate our system.

SETUP: We use a virtual machine with a freshly installed latest version of Google Chrome. Searches are condicted in "Incognito Mode" tabs. We also use a VPN to the West coast of the US. These invariants serve the main purpose so that Google's does not give any personalized results based on a browser fingerprint or IP address. We chose the US as the VPN destination since our dataset articles were extracted from US news sources and we wanted to scrape for articles with a similar style of writing. If you were to use the tool in Australia, Google would usually return articles from local sources. We restrict our scope to specifically this dataset rather than train on a wide dataset from all sources.

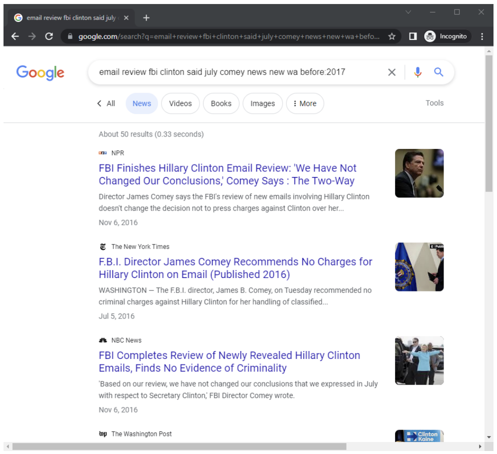
Another invariant we implement is to add a `before:2020` to our summary. This forces Google News to only find articles before this year so that the news we get won't be from recent news. A common discussion topic from our dataset was Donald Trump's 2016 election campaign and we know that the news regarding Trump in 2023 is much different to that of 2016. This makes sense as we are not using a very recent dataset so clamping the date we find contextual articles assumes that if were looking for fake articles at the time of reading the imput article, we wouldn't have too much future articles available.

PROCESS: We attempt to get the top three articles and save the URL for each input article. Not all summaries returned three articles so we perform scraping in three passes:

1. We enter the whole summary without any changes. This is the most ideal approach and most machine-replicable. This covered 70% of our dataset.
2. Still performing only generic actions, we remove any bad words or non-important connectives then searched again. This should still be machine-replicable with further work. This covered the next 20% of our dataset.

3. For the last 10% of our dataset, we had to manually look at the input article content and summary generated to figure out why we still received no results. Our hypothesis was that this was a combination of our non-tuned summary extraction and the fact that some *outrageous* Fake articles simply didn't have any similar articles that could be found. We will discuss this limitation in Section 6.1.

From the above passes, we were not able to find context articles for four input articles described in a table in Appendix B. Furthermore, we were only able to find one or two articles for some inputs but we can still continue with our similarity model.


email review fbi clinton said july
google.com/search?q=email+review+fbi+clinton+said+july+comey+news+new+wa+befo...
Incognito
Google
email review fbi clinton said july comey news new wa before:2017
All
News
Videos
Books
Images
More
Tools

About 50 results (0.33 seconds)

NPR

FBI Finishes Hillary Clinton Email Review: 'We Have Not Changed Our Conclusions,' Comey Says : The Two-Way

Director James Comey says the FBI's review of new emails involving Hillary Clinton doesn't change the decision not to press charges against Clinton over her...

Nov 6, 2016

The New York Times

F.B.I. Director James Comey Recommends No Charges for Hillary Clinton on Email (Published 2016)

WASHINGTON — The F.B.I. director, James B. Comey, on Tuesday recommended no criminal charges against Hillary Clinton for her handling of classified...

Jul 5, 2016

NBC News

FBI Completes Review of Newly Revealed Hillary Clinton Emails, Finds No Evidence of Criminality

'Based on our review, we have not changed our conclusions that we expressed in July with respect to Secretary Clinton,' FBI Director Comey wrote.

Nov 6, 2016

The Washington Post

Figure 2: Sample of articles found in Google after searching an article summary.

After gathering three URL links for each context article, we use the Python `newspaper3k` [3] library to download the article and automatically extract its title and content.

Similarity model

[Methods/Feature — Similarity model] **Duke**: Write

3.3 Feature — Non-latent features

[Methods/Feature — Non-latent features] **Duke**: Write

3.4 Feature — BERT embeddings

[Methods/Feature — BERT embeddings] **Duke**: Write

3.5 Normalization and scaling

[Methods/Normalization and scaling] **Dhruv**: Write

3.6 Model — Machine learning

We used four state of the art machine learning models (commonly used in fake news detection) to perform our classification. We chose Logistic Regression (LR), Support Vector Machines (SVM), Decision Trees

(DT), and XGBoost (XGB). Due to the small size of our dataset, we needed to tune the regularization hyperparameters to ensure our models didn't overfit. In particular, tree-based models such as DT and XGB should be able to fully segment our classes so we need to control the depth of the tree and splitting criteria. Models such as LR and SVM will need to control L2 regularization. In SVM, we will test the type of kernel used.

To find the best hyperparameters, we perform 5-Fold cross validation across 80% of our dataset and average the validation score. We pick the parameters with the best validation score and test our model with the remaining 20% of the dataset. The table in the Appendix C shows the hyperparameters tested for each model and a reason for why the ranges were selected.

3.7 Model — Neural networks

[Methods/Model — Neural networks] **Dhruv**: Write

4 Experimental setup

4.1 Dataset

For our research, we use the `FakeNewsData` dataset collated by Horne and Adali in [4] on research regarding fake news in the 2016 presidential elections. This dataset contains two subsets, *"Buzzfeed Political News"*, and *"Random Political News"*. We make use of the *Buzzfeed* subset since this contains long form text articles that are binary categorized in with Fake and Real labels. The *Random* subset contains an extra label, Satire, which is out of scope for our research.

The original dataset was collated by Craig Silverman (BuzzFeed News Editor) in an article [5] analyzing fake news. The analysis concentrates on the Facebook engagement on real and fake news articles shared to the social media website. Various keywords related to events during the election were searched and articles with highest engagement were collected. A ground truth was assigned by manual analysis using a list of known fake and hyperpartisan news sites. A details description of their process can be found in their article.

Following BuzzFeed's analysis, Horne extracted the content and title from the articles and formed the dataset. In total, there were 53 real and 48 fake articles. Notable events during the election such as Donald Trump's campaign and various Hillary Clinton scandals and rumors were features in the articles.

After extending this dataset with our novel context article scraping and similarity methods, we used a 60/20/20 train/validation/test set. This was stratified and randomized to ensure the best results. An example of items in our dataset can be found in Table 3.

4.2 Evaluation metrics

To evaluate our classification models, we will use accuracy and F1 score. These metrics are commonly used for binary classification problems as well as in the misinformation detection domain.

Accuracy is measured as the proportion of the total number of correctly classified samples over the total count of samples:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Our dataset is quite balanced so this will be a good general first step measure.

On the other hand, the F1 score is generally used on imbalanced datasets by looking at both precision and recall:

$$F_1 = 2\,\frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

Whilst our dataset is not imbalanced, we will still output this result for comparative purposes.

118_Real	**1_Fake**
FBI Completes Review of Newly Revealed Hillary Clinton Emails Finds No Evidence of Criminality	*5 Million Uncounted Sanders Ballots Found On Clinton's Email Server*
FBI Director James Comey said Sunday that the bureau won't change the conclusion it made in July after it examined newly revealed emails related to the Hillary Clinton probe. "Based on our review, we have not changed our conclusions that we expressed in July with respect to Secretary Clinton" Comey wrote in a letter to 16 members of Congress. [...]	Hillary in hot water over her email server, again. Sacramento, CA — Democratic nominee Hillary Clinton is in hot water again after nearly 5 million uncounted California electronic ballots were found on her email server by the F.B.I. The majority of those ballots cast were by Bernie Sanders supporters. [...]

Table 3: A sample of one fake and real article in our dataset. The article ID, title and content are shown in the rows. Both articles are regarding a scandal with Hillary Clinton using a private server to store emails. The fake article reports on an event that never happens whereas the real article reports the true event – that Clinton was exonerated from criminality.

5 Results and discussion

[Results and discussion] **Jim**: Machine learning

[Results and discussion] **Dhruv**: Neural nets

6 Conclusion

[Conclusion] Summarise the study and discuss directions for future improvement

6.1 Limitations

[Conclusion/Limitations] Convert list of limitations to subsubsections with discussion.

- Preprocessing and tokenization was a bit funny:
  1. Converting everything to lowercase destroyed acronyms such as "US"
  2. `nltk` lemmatizer required manually specifying the part of speech to work. We could have used a different library that extracted the POS automatically. Alternatively we could have investigated not lemmatizing at all to maintain proper structure for non-latent features and BERT embeddings.
- Summary extraction didn't produce perfect results
- Article scraping sometimes returned no results even after manually figuring out why
- Models trained on only US news, may be a problem
- Dataset is small and only contains articles related to political events surrounding the 2016 US election.

References

[1] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit.* "O'Reilly Media, Inc.", 2009.

[2] Radim Řehůřek and Petr Sojka. "Software Framework for Topic Modelling with Large Corpora". English. In: *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks.* Valletta, Malta: ELRA, May 2010, pp. 45–50.

[3] Lucas Ou-Yang. *newspaper3k.* 2013. URL: https://newspaper.readthedocs.io/en/latest/.

[4] Benjamin Horne and Sibel Adali. "This just in: Fake news packs a lot in title, uses simpler, repetitive content in text body, more similar to satire than real news". In: *Proceedings of the international AAAI conference on web and social media.* Vol. 11. 1. 2017, pp. 759–766.

[5] Craig Silverman. *This Analysis Shows How Viral Fake Election News Stories Outperformed Real News On Facebook.* Nov. 2016. URL: https://www.buzzfeednews.com/article/craigsilverman/viral-fake-election-news-outperformed-real-news-on-facebook.

A Individual contributions

Jim	Dhruv	Duke

A.1 Jim

[Individual contributions] **Jim**: ∼1pg detailing individual contributions

A.2 Dhruv

[Individual contributions] **Dhruv**: ∼1pg detailing individual contributions

A.3 Duke

[Individual contributions] **Duke**: ∼1pg detailing individual contributions

B Article scraping

ID	Article extract	Summary
128_Real	[...]I have a prediction. I know exactly what November 9 will bring. Another day of God's perfect sovereignty. He will still be in charge. His throne will still be occupied. He will still manage the affairs of the world. Never before has His providence depended on a king, president, or ruler. And it won't on November 9, 2016. "The LORD can control a king's mind as he controls a river; he can direct it as he pleases" (Proverbs 21:1 NCV). On one occasion the Lord turned the heart of the King of Assyria so that he aided them in the construction of the Temple. On another occasion, he stirred the heart of Cyrus to release the Jews to return to Jerusalem. [...]	god wa one never every king november still heart
2_Fake	Washington, D.C. – South African Billionaire, Femi Adenugame, has released a statement offering to help African-Americans leave the United States if Donald Trump is elected president. According to reports, he is offering $1 Million, a home and car to every Black family who wants to come to South Africa. Concerns about Donald Trump becoming president has prompted a South African billionaire to invest his fortune in helping African-Americans leave the United States to avoid further discrimination and inequality. [...]	ha adenugame africanamericans south femi united states africa president donald
10_Fake	The Internet is buzzing today after white supremacist presidential candidate Donald Trump was caught by hotel staff snorting cocaine. Maria Gonzalez an employee at the Folks INN & Suites Hotel in Phoenix brought room service to his room witnessed it all. "When I walked in I saw 3 naked prostitutes and maybe 100,000 in hundred dollars bills and a mountain of white powder on the table, I thought it was a dog on the floor sleep but it was his hair piece, he was bald and sweating like crazy." [...]	wa room hotel maria told employee gonzalez hit video get
34_Fake	It has been more than fifteen years since Rage Against The Machine have released new music. The members of the band have involved themselves in various other projects during their lengthy hiatus, but one pressing issue has forced the band to team up once again. In a statement posted online, Rage Against The Machine announced they would be releasing a brand new album aimed at spreading awareness about "how awful Donald Trump is". [...]	trump rage album machine band ha donald music outside year

Table 4: Articles we were not able to find context articles for.

C Machine learning

Model	Parameter	Selection	Reasoning
Logistic Regression	Inverse L2 coefficient	0.2:1.2:0.2	This is the main regularization parameter. We chose a range around the default 1.0 but shifted our range to be more biased towards higher regularization.
	Solver	lbfgs, liblinear	The liblinear solver was suggested by the documentation as an alternative for small datasets.
SVM	Inverse L2 coefficient	0.2:1.2:0.2	Same reasoning as LR regularization.
	Kernel	rbf, poly, sigmoid	Selecting the right kernel for a dataset will make our methods perform better.
	Kernel coefficient	$\frac{1}{\text{n\_features} \times var(X)}$, 0.01, 0.05	Same reason as above.
Decision Tree	Criterion	gini, entropy	To test different methods of measuring split quality on the node.
	Max depth	no limit, 3:9:2	Controls how complex the tree is. A less deeper tree is more regularized.
	Max features	$0.3 \times \text{n\_features}$, $\sqrt{\text{n\_features}}$, all features	Standard defaults suggested by documentation. Is a regularization control so not all features are considered at each split.
	Min samples for splitting node	2:4:1	Reduce the number of leafs with only one sample of representation to increase regularization.
XGBoost	Learning rate	0.1:0.5:0.1	Smaller learning rates reduce overfitting.
	Max depth	1:6:1	Same as max depth for DTs.
	L2 coefficient	0.8:1.6:0.2	Testing higher regularization. Default is 1.
	L1 coefficient	0:0.4:0.2	Testing higher regularization. Default is 0.0.

Table 5: Table of all the models chosen and the hyperparameters selected for each model. We describe a range of values in the format start:end:step, where start and end are inclusive.