2018

Detecting Clickbait: Here's How to Do It

Christopher Brogly
Victoria L. Rubin

Detecting Clickbait: Here's How to Do It / Comment détecter les pièges à clic

Chris Brogly, Victoria L. Rubin

Canadian Journal of Information and Library Science, Volume 42, Numbers 3-4, September-December 2018, pp. 154-175 (Article)

Published by University of Toronto Press

Detecting Clickbait: Here's How to Do It

Comment détecter les pièges à clic

Chris Brogly
Faculty of Information and Media Studies, University of Western Ontario
cbrogly@uwo.ca

Victoria L. Rubin
Faculty of Information and Media Studies, University of Western Ontario
vrubin@uwo.ca

Abstract: Automatic clickbait detection is a relatively novel task in natural language processing (NLP) and machine learning (ML). "Clickbait" is a hyperlink created primarily to attract attention to its target content. This article introduces a binary classifier, the Language and Information Technology Research Lab (LiT.RL, pronounced "literal") Clickbait Detector, which automatically distinguishes clickbait from non-clickbait. We used NLP and ML for 38 textual features, contrasting clickbait with "headlinese." When tested on 11,000 hyperlinks, it achieves 94 per cent accuracy using a support vector machine. Integrated with the LiT.RL News Verification Browser, a downloadable stand-alone research tool, the Clickbait Detector user interface shows automated real-time colour-coded analysis of any news website.

Keywords: automated clickbait detection, "fake news," misinformation and disinformation, machine learning, natural language processing

Résumé : La détection automatique de pièges à clics est relativement nouvelle dans le traitement naturel du langage (TNL) et en apprentissage automatique (AO). Le piège à clics est un lien hypertexte créé surtout pour attirer l'attention vers son contenu cible. Cet article introduit un classifiant binaire, le détecteur de pièges à click Language and Information Technology Research Lab (LiT.RL), qui distingue automatiquement les pièges des non-pièges. Nous utilisons le TNL et l'AO sur 38 caractéristiques textuelles, contrastant les pièges à clics et de titres accrocheurs. Nous avons testé 11 000 liens hypertextes et avons obtenu un score de 94 pour cent d'exactitude en utilisant une machine à vecteurs de support. Intégrée au fureteur de vérification de nouvelles LiT.RL, un outil autonome de recherche téléchargeable, l'interface usager du détecteur de pièges à clics montre une analyse en temps réel qui utilise un code de couleur de tout site Web de nouvelles.

Mots clés : détection automatique de pièges à clics, fausses nouvelles, mésinformation et désinformation, apprentissage automatique, traitement naturel du langage Introduction

Clickbait

Writing attention-grabbing news headlines has long been viewed as an art form and a skill in journalism (Frampton 2015). Nowadays, the success of each headline in digital news is directly monetized via social media user engagement metrics such as clicks, shares, likes, or views, and such engagement drives web traffic and revenue for many digital news organizations. The business model of this kind incites content producers to be ever more creative with hyperlinks and the relationship between the headline hyperlink and the actual body of the article at the end of the clicked hyperlink. Many digital news organizations such as "*Quartz*, *The Huffington Post*, and *Upworthy* expend significant effort into crafting headlines that generate engagement" (Silverman 2015, 106). In the absence of original informative reporting, content producers resort to manipulative tactics of clickbaiting.

Clickbait, by definition, is content created for the primary purpose of attracting attention and encouraging visitors to click on the associated link; it contributes to the rapid spread of rumours and misinformation online (Chen, Conroy, and Rubin 2015). Clickbait formats vary from primitive listicles (for example, "five things you need to see") to more sophisticated attention-getting techniques (see Figure 1 for two contrasting examples of clickbait (1.a) and non-clickbait (1.b)). Clickbait topics often revolve around "soft news" about celebrities, pop culture, movies, style, weddings, and decor, to name a few. Trending topics beyond "soft news" can also be presented in a clickbaity format.

There has been increased interest in the topic of clickbait in the public, most notably brought to light by the 2016 US presidential campaign and proliferations of so-called "fake news." The Pew Research Center's 2016 survey of 1,002 US adults found that approximately two out of three respondents (64 per cent) said that fabricated news stories "cause a great deal of confusion" about the basic facts of current issues and events (Anderson and Rainie 2017, 2). By 2019, the problematic nature of intentional manipulative tactics (disinformation), as well as unintended errors (misinformation), have been widely acknowledged; they threaten to disrupt politics, business, culture (Jack 2017), and democracy (Owen 2017). The theme that unites most manipulative tactics in digital media is the intent to create a false belief or conclusion in the readers' minds, as per classical definitions of deception (Buller and Burgoon 1996; Zhou et al. 2004). Varieties of "fakes" proliferate in the news streams propagated via social media (outright falsifications, unconfirmed rumours, and satirical fake news), but of those "fakes" (Rubin, Chen, and Conroy 2015), clickbait is relatively distinct in its linguistic form and formulaic devices (Chen, Conroy, and Rubin 2015).

Clickbait promotion

Clickbait producers tend to present their content as "captivating," "engaging," and "meaningful for millennials" with overtly stated goals to "laugh, share and

(a) Clickbait	"Olsen twin or stylish senator? You decide"
(b) Non-clickbait	"Dominique Strauss-Kahn awaits verdict in 'aggravated pimping' trial"

Figure 1. Sample of clickbait (a) and non-clickbait (b) hyperlink headlines from the combined dataset used in this study

inspire" (Diply 2017). Unfortunately, clickbait often resonates with the audiences on the prowl for "light-weight" viral content. The process of clickbait creation is best described by those who are tasked with its generation as "taking something newsy and making it digestible"—in particular, borrowing trending ideas and "infusing them" with associated news content to "grab the feelings" of the audience (Diply 2017). Given the scale of clickbait proliferation, its self-branding, and its advertisement revenue-driven motivation to "engage users," we need to be actively considering accurate automated methods for detecting clickbait as countermeasures.

Automated clickbait detection

The natural language processing (NLP) and machine learning (ML) community has been looking for automated methods of identifying clickbait since around 2015, to the best of our knowledge (for example, Chen, Conroy, and Rubin 2015; Potthast et al. 2016; Gollub et al. 2017). Whether it is labelled "exciting" by its generators or simply "annoyingly overpromising" by its critiques, the content and delivery style of clickbait is unique enough that it is recognizable with the naked eye. It has also been proven to be sufficiently identifiable with ML and NLP techniques. The pioneers in automation of clickbait detection achieved 93 per cent accuracy in detecting clickbait and 89 per cent accuracy in blocking it with a built-in Google Chrome extension (Chakraborty et al. 2016).

A 2017 Clickbait Challenge produced a slew of accurately performing systems (for example, Elyashar, Bendahan, and Puzis 2017; Grigorev 2017; Indurthi and Oota 2017; Papadopoulou et al. 2017; Wei and Wan 2017; Zhou 2017). Other groups reported reaching 98.3 per cent accuracy with deep learning techniques trained on a large dataset of 1.67 million Facebook posts from 153 media organizations (Rony, Hassan, and Yousuf 2017). Our article further contributes primarily towards this body of literature focused on distinguishing clickbait headlines from non-clickbait headlines in automated ways by offering the NLP/ML community a robust set of predictive features and open-source code for further development.[1]

In this article, a clickbait is seen as a hyperlink in the context of automated clickbait detection. The clickbait hyperlink (typically expressed through text—for example, a news headline) is incomplete information about the target content of the hyperlink. Providing limited information about the linked content strongly promotes visiting the content as it exploits the psychological curiosity of readers. Such a curiosity is often driven (and adeptly directed) by a partial reveal or misleading subject in the clickbait hyperlink. Such techniques are manipulative, and options to expose and appropriately label such text ideally should be available to the public.

In the remainder of the article, the second section will briefly describe our dataset, implementation details, and methodology used for detecting clickbait hyperlinks. The third section shows overall system performance results with the 38 NLP features, while the fourth section goes into a detailed description of a subset of the best performing predictive features (in order of their effectiveness, from greatest to least). The fifth section discusses our results and limitations, raising concerns about how the concept of clickbait is operationalized in the literature and how certain assumptions may skew currently available training datasets.

Methods

Dataset and metrics

To develop the Clickbait Detector, the Language and Information Technology Research Lab (LiT.RL) examined a dataset of clickbait hyperlinked headlines (in textual form) for their prominent linguistic features and compared them to non-clickbait hyperlinked headlines. Informed by the previous literature on clickbait detection and our own observations, we used best predictors as input for ML techniques to train our classifier. Our dataset consisted of a combination of the 2017 Clickbait Challenge validation set of 21,000 hyperlink texts by Gollub et al. (2017) and Chakraborty et al.'s (2016) set of 30,000 hyperlink texts (Table 1).

The development started with the 2017 Clickbait Challenge training set of 2,100 hyperlink textual headlines. A larger crowd-sourced corpus was made available in the summer of 2017—the 2017 Clickbait Challenge validation set. The 2017 Clickbait Challenge validation set was compiled from Twitter and provides crowd-sourced rankings of individual hyperlink texts. Five human judges ranked each hyperlink text, based on a four-point scale, where 0 is not clickbait, 0.33 is slight clickbait, 0.66 is moderate clickbait, and 1 is clickbait, as per Gollub et al. (2017).

One of the measures in the released data was a mean score indicating the level of clickbait. The mean score is the average of the five human judge scores, and we used this metric to identify hyperlink texts that are clearly not clickbait (mean score < 0.1) and those that are likely to be clickbait (mean score > 0.6). In other words, when training the LiT.RL Clickbait Detector, any hyperlink text from the 2017 Clickbait Challenge validation set with a crowd-sourced mean score higher than 0.6 is classified as "clickbait," and any text with a mean

Table 1. Description of the datasets and metrics of "clickbaitiness"

Dataset	Number of clickbait	Number of non-clickbait	"Clickbaitiness" metrics
2017 Clickbait Challenge validation set (Gollub et al. 2017)	4,761	14,777	Interval mean score by 5 human judges (0–1.0): 0: non-clickbait; 0.33: slight clickbait; 0.66: moderate clickbait; 1: clickbait
2016 Chakroboty et al.'s (2016) set	15,999	16,001	Binary non-clickbait; clickbait
Our 2019 combined dataset	18,899	18,901	Binary mean score < 0.1: non-clickbait; mean score > 0.6: clickbait
Middle-ranked texts excluded for combined set from 2017 Clickbait Challenge Validation set	1,840	11,898	mean score > 0.1: non-clickbait; mean score < 0.6: clickbait

(a) How do dogs donate blood?	
(b) RT @BuzzFeedAnimals: What it's like to grow up with your best friend	

Figure 2. Examples of potentially ambiguous hyperlink text with means of approximately 0.5

score of less than 0.1 is classified as "not clickbait" (Table 1). The range of texts between 0.1 and 0.6 are excluded as they may be potentially ambiguous (as exemplified in Figure 2).

Chakraborty et al.'s (2016) dataset did not provide mean score metrics but contained two files that were pre-classified, drawn from *WikiNews* (for non-clickbait), and *Buzzfeed*, *Upworthy*, *ViralNova*, *Scoopwhoop*, and *ViralStories* (for clickbait). We used the two datasets in tandem as a combined set for our experiments, with 70 per cent of the combined set used for training and 30 per cent used for testing. The texts in each category are randomly selected for each new training of the detector, but the numbers remain the same at 70 per cent (training) and 30 per cent (testing). The histogram presented later in this article is based on the total combined set (as per Table 1).

Implementation and run-time performance

The LiT.RL Clickbait Detector is implemented in Python 2.7 using scikit-learn (Pedregosa et al. 2011) for all ML functionality. The pattern.en library provides most NLP, although some functions are from the Natural Language Toolkit (Bird, Klein, and Loper 2009). SciPy (Jones, Oliphant, and Peterson 2001) and NumPy (van der Walt, Colbert, and Varoquaux 2011) were also used. The detector has two main entry points, one for performing training/testing (producing the results shown here) and one for accepting a line of hyperlink text as input. The detector will classify the input text and output scores in the format (clickbait score, not-clickbait score). Optionally, the detector can output the class (0 = clickbait or 1 = not clickbait).

The run-time performance is presented in Table 2. Our system is not optimized for fast training, but classifying individual (previously unseen instances of) hyperlinks is fast, with many texts being processed in just one second.

Individual feature scores and descriptions are presented in Table 3, sorted from most to least accurate. Our research and development process was heavily influenced by Potthast et al.'s (2016, 2018) and Chakraborty et al.'s (2016) work, but it includes our own implementations of many features directly adopted from these works.

Table 2. Runtime for training stages of the support vector machine (SVM) classifier

Measure	Time, minutes
Dataset pre-processing	12.50
WordNet-based noun similarity	5.0
Miscellaneous	0.83
LiT.RL Clickbait Detector training overall	18.33

Table 3. The LiT.RL Clickbait Detector feature performance by accuracy using the SVM classifier

No.	Feature name	Precision	Recall	F1 score	Accuracy	New for LiT.RL detector	Feature description
1	*getPronounCount*	0.6658	0.9286	0.7756	0.7312		Number of pronouns
2	*getWord2GramsAvgLen*	0.6849	0.6235	0.6528	0.6683	Yes	Average length of bi-grams
3	*getDeterminers*	0.6286	0.8106	0.7081	0.6658		Number of determiners
4	*getWord3GramsAvgLen*	0.681	0.5974	0.6365	0.6588	Yes	Average length of tri-grams
5	*startsWithNumber*	0.5871	0.982	0.7349	0.6457		Begins with a number?
6	*getAdvpCount*	0.5862	0.8984	0.7095	0.6321		Number of adverbial phrases (ADVPs) part-of-speech (POS) tags
7	*getNPsCount*	0.6342	0.6017	0.6175	0.6273		Number of noun phrase (NP) POS tags
8	*containsTriggers*	0.6861	0.4394	0.5357	0.6192		Number of trigger words
9	*getNumbersSum*	0.5671	0.9378	0.7068	0.6109		Sum of numeric values in text
10	*getNNPLOCCount*	0.9123	0.231	0.3687	0.6044	Yes	Number of proper noun, singular, location – part of speech tags (NNP–LOCs)
11	*getFirstNNPPos*	0.581	0.7163	0.6416	0.5999	Yes	Position of first proper noun, singular (NNP) POS tags
12	*getWordCount*	0.6029	0.5779	0.5901	0.5986		Number of words
13	*getVerbCount*	0.5565	0.7189	0.6274	0.573		Number of verb POS tags
14	*containsOn*	0.5939	0.4467	0.5099	0.5706	Yes	is "on" in the sentence?
15	*maxDistToNNP*	0.5629	0.6054	0.5833	0.5676	Yes	Maximum distance to proper noun
16	*lenOfLongestWord*	0.5846	0.4394	0.5017	0.5635		Length of longest word
17	*minDistToOn*	0.5976	0.3661	0.454	0.5597	Yes	Number of characters to first "on" text
18	*avgDistToNNP*	0.5423	0.7357	0.6244	0.5574	Yes	Average distance to proper noun
19	*similarityOfNouns*	0.5793	0.416	0.4842	0.5569	Yes	Maximum similarity of NNPs and NPs
20	*getEmotiveness*	0.5467	0.6417	0.5904	0.5548		(Number of adjective phrases (ADJPs) + number of ADVPs divided by (number of NPs + number of verbs (VBs), base form VBs)
21	*getAdjpCount*	0.5624	0.4906	0.524	0.5544		Number of ADJP POS tags
22	*maxDistToQuote*	0.5288	0.9741	0.6854	0.5529		Number of characters to a quotation
23	*maxDistFromNPToNNP*	0.557	0.5013	0.5277	0.5513	Yes	Maximum distance from noun to proper noun
24	*getAcademicWordsCount*	0.8573	0.1187	0.2085	0.5494		Number of academic words (from pattern.en)
25	*minDistToNNP*	0.5229	0.8364	0.6435	0.5366		Minimum distance to a NNP
26	*getNNPPERSCount*	0.5086	0.9797	0.6696	0.5164	Yes	Number of NNP-person (PERS) POS tags
27	*NNPCountOverNPCount*	0.5106	0.5824	0.5442	0.512	Yes	Number of NNPs divided by number of NPs

(*continued on next page*)

Table 3. (*continued*)

No.	Feature name	Precision	Recall	F1 score	Accuracy	New for LiT.RL detector	Feature description
28	*getSwearCount*	*omitted*					Number of vulgar words
29	*npsPlusNNPsOverNumbersSum*	0.5086	0.6394	0.5666	0.5108	Yes	Number of NPs + NNPs divided sum of numbers in headline
30	*firstPartContainsColon*	0.7339	0.0321	0.0615	0.5102		Do the first 15 characters have a colon?
31	*getNNPCount*	0.5079	0.5918	0.5466	0.5091		Number of NNPs
32	*getTimeWordsCount*	0.5991	0.0464	0.0861	0.5076		Number of time-related words (from pattern.en)
33	*getQuestionMarks*	0.5034	0.9919	0.6679	0.5067		Number of question marks
34	*getAtMentions*	0.7568	0.0148	0.0291	0.505		Number of @ characters
35	*maxDistToAt*	0.7568	0.0148	0.0291	0.505		Distance to an @ character
36	*getHashTagsAndRTs*	0.5593	0.0175	0.0339	0.5018		Number of hash tags and re-tweet texts (RT)
37	*maxDistToHashTag*	0.5535	0.0155	0.0302	0.5015	Yes	Distance to a hash tag
38	*getCharLength*	0.4555	0.4237	0.4391	0.4586		Length of text

In Table 3, features that are marked as "yes" in the "New Feature" column were novel to this work, to the best of our knowledge, for the time of the detector implementation over the course of late 2017 and early 2018. The similarity to "headlinese" (Mårdh 1980) may be the key indicator of non-clickbait, and further work is needed to implement methods of parsing/detecting "headlinese." The importance of "headlinese" is discussed throughout the remainder of this article.

User interface

The LiT.RL Clickbait Detector is a key component of the LiT.RL News Verification Browser, which is available via GitHub at https://github.com/litrl/litrl_code/releases/tag/exp-0.14.0.1. The LiT.RL News Verification Browser is a research tool that allows the LiT.RL lab's deception detectors to be easily used through a graphical web browser. Figure 3 demonstrates the appearance of the LiT.RL Clickbait Detector's user interface, as applied to three news websites. The websites are arranged in the decreasing degree of "clickbaitiness": from a social media news site (92 per cent "clickbaitiness," 9 September 2019) and a more traditional "legacy" news site (37 per cent "clickbaitiness," 9 September 2019) to a generally non-clickbaity university news webpage (20 per cent "clickbaitiness," 9 September 2019).

As a key component of the LiT.RL News Verification Browser, the LiT.RL Clickbait Detector is, to the best of our knowledge, the only clickbait detector with a straightforward desktop user interface that can run fully on a local machine. No communication with a remote server is required. Links are colour-coded according to the severity of the clickbait, by analogy to a traffic stoplight

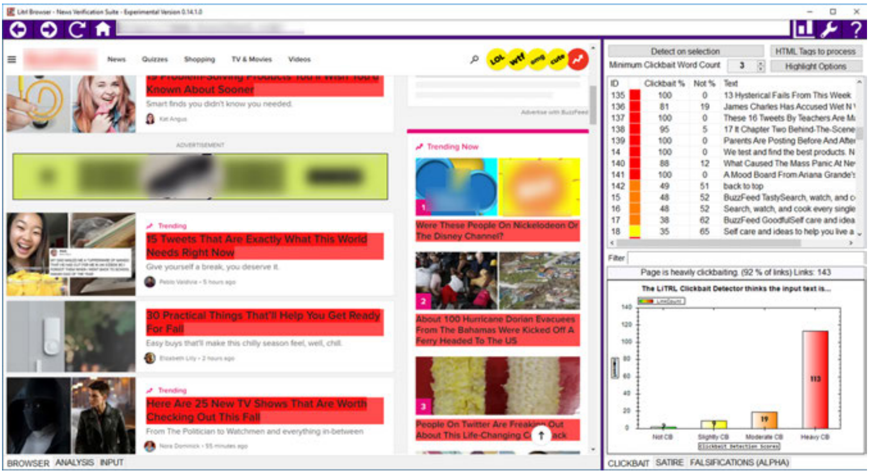
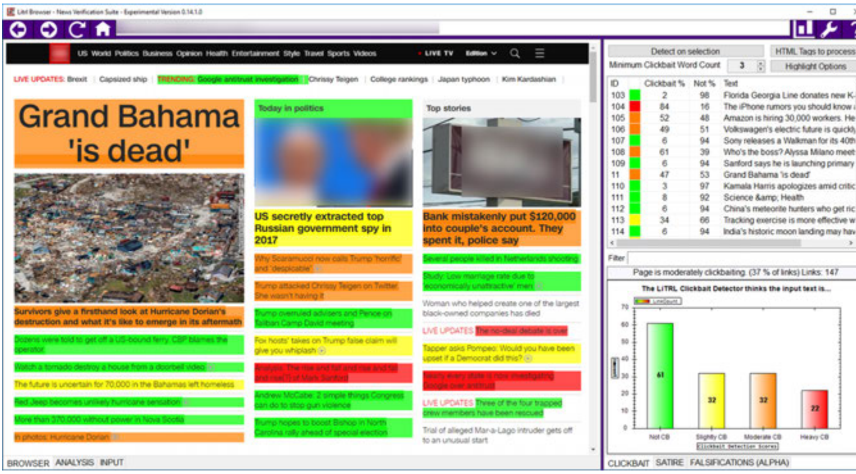


Figure 3. Screenshots of three websites in decreasing degree of "clickbaitiness" seen in the LiT.RL Clickbait Detector's user interface, which is part of the LiT.RL News Verification Browser: (1) a social media news site (92 per cent "clickbaitiness," 9 September 2019); (2) a traditional news site (37 per cent "clickbaitiness," 9 September 2019); and (3) a webpage of university news (20 per cent "clickbaitiness," 9 September 2019)



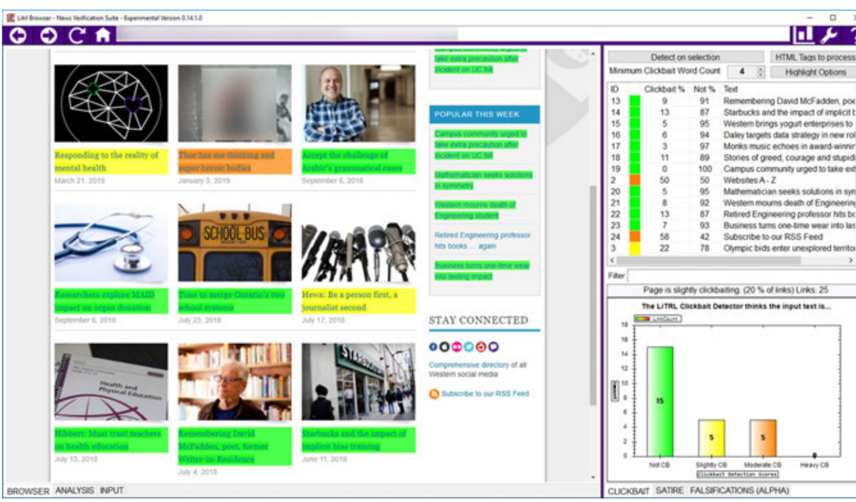


Figure 3 (Continued).

(shown in gray from left to right on the bar graphs in the right hand corner of images from figure 3; green = non-clickbait; yellow = slightly clickbaity; orange = moderately clickbaity; and red = heavily clickbaity). Users can save the results of the detection to a standard SQLite database, which includes the individual 38 feature scores, the HTML tags processed, the URL of the page, and the detector's overall clickbait score per analyzed news webpage. Using the detector as part of this graphical program has shown that high test set accuracy scores are not a perfect indicator of real-world performance, as the effectiveness of the detector tends to fluctuate noticeably more when applied to real-world internet browsing. The browser was developed, in part, to observe this trend.

Results

In the binary text classification task (clickbait/non-clickbait), the LiT.RL Clickbait Detector achieves 94 per cent accuracy with a support vector machine (Linear SVC from sklearn) on a test set of 5,670 clickbait texts and 5,671 non-clickbait texts. A copy of this detector was included in the 0.14.0.0 public release of the LiT.RL News Verification Browser on GitHub (Rubin et al. 2018). Parameters in the code are shown in Figure 4.

The training set size for clickbait was 13,229 texts, while the training set size for non-clickbait was 13,230 texts (Table 4). As already mentioned, the training/test sets were created by using a 70/30 split on the combined dataset. Scaling was not performed.

We used 38 NLP features to distinguish clickbait hyperlink text from non-clickbait ones. Table 3 presents the metrics of individual feature performance based on the test set, a randomly selected 30 per cent of the combined corpus (Table 1).

Key clickbait predictors

This section is a detailed explanation of the key NLP features indicative of clickbait, from most to least effective in their individual performance (in the order presented in Table 3). Several features are illustrated with a low scoring and high scoring text example. Understanding the between-group differences is crucial for automated binary classification and the ability to predict the binary label (clickbait or not clickbait) for a new previously unseen hyperlink.

Pronoun frequencies

The best performing indicator of the set of 38 features is the *getPronounCount* function, which achieves a 73.1 per cent accuracy in binary classification (clickbait or not clickbait). We found that the number of texts without pronouns is almost doubled in non-clickbait, and the number of clickbait with one pronoun is almost seven times the number of non-clickbait single pronoun hyperlinks. Figure 5 contains two examples differentiated by the prevalence of pronouns in the clickbait sample picked up by this feature.

```
linearclf = svm.LinearSVC(class_weight=None,verbose=1,max_iter=2000)
self.classifier_linear = CalibratedClassifierCV(linearclf)
self.classifier_linear.fit(Xtrain, Ytrain)
```

Figure 4. Parameters to the instance of linear SVC from sklearn

Table 4. Number of clickbait and non-clickbait texts used, train/test split, and test set classification results

	Total	Train (70%)	Test (30%)	Test (30%) Correct	Test (30%) Incorrect
Clickbait	18,899	13,229	5,670	5,221	449
Not clickbait	18,901	13,230	5,671	5,431	240

(1) Low scoring example (of a likely non-clickbait hyperlink text)	"Here's the cold @LinkedIn message that prompted a CEO to give the sender a job:"
(2) high scoring example (of a likely clickbait hyperlink text)	"**My** friend got with **her** boyfriend after **he** cheated on **my** sister should **I** snub **their** wedding?"

Figure 5. The *getPronounCount* feature contrasting examples that illustrate binary classification (clickbait/not clickbait; pronouns are in bold).

Chen, Conroy, and Rubin (2015) has previously suggested that unresolved pronouns occur frequently in clickbait. This result also confirms Chakraborty et al.'s (2016) empirical observation. Clickbait also tends to have at least one pronoun and often no nouns.

Average length of n-grams

Average length is defined as the sum of the character length of each *n*-gram divided by the number of *n*-grams. Bi-gram average length (*getWord2GramsAvgLen*, 65.9 per cent) was the second most individually accurate feature. Clickbait bi-grams are generally shorter than non-clickbait bi-grams. Tri-gram average length (*getWord3GramsAvgLen*, 65.88 per cent) was slightly less accurate (ranked fourth). Initially, we predicted the total character length of a hyperlink text to be one of the better indicators of clickbait as the feature had been used previously in Potthast et al. (2016, 2018). In fact, such a feature (*getCharLength*) is not as accurate (thirty-eighth), but it is still somewhat effective. On an individual basis, measuring average *n*-gram length performs considerably better (see Figures 6 and 7 for examples).

Determiner frequencies

We found one or more determiners in the dataset (for example, the, a, some, most, every, no, which) that were used more often in clickbait texts than in non-clickbait texts (Figure 8). Counting determiners on their own is an effective measure of clickbait, as per Chakraborty et al. (2016), and our *getDeterminers* feature was the third most accurate. Unfortunately, like a few other features, such as NNP–LOC part-of-speech tags, determiners occur only in a limited number of clickbait hyperlinks.

Sentence-initial numerals and other numeric terms

"A listicle" is textual content arranged in a numbered or bullet form. Listicles are prominently marked with a cardinal number at the beginning of the headline. The numeral forward references, which are the counted items in the linked article, still appear to be in use by some clickbait writers, even though the audience is generally aware that such a format is a good example of clickbait patterns (Figure 9). The *startsWithNumber* feature was the fifth most individually accurate (64.6 per cent).

We also used the *pattern.en* NLP framework to parse any words in a hyperlink text that represented a numeric value. Numeric terms of clickbait resulted

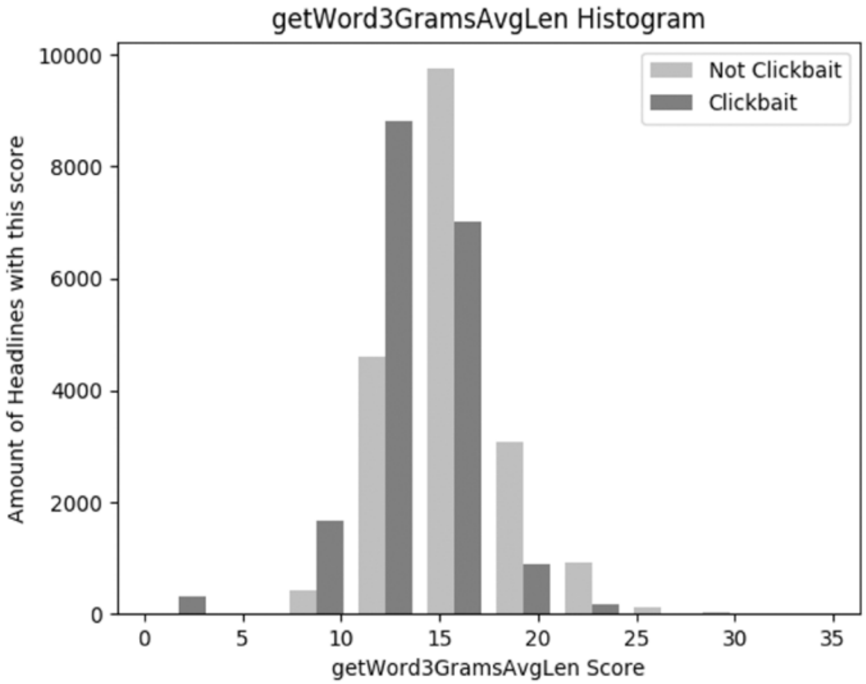


Figure 6. Histogram illustrating the shorter average length of tri-grams found in clickbait

(1) Low scoring example	"Caught tech-handed"
(2) High scoring example	"Watch A Weatherman Flawlessly Pronounce llanfairpwllgwyngyllgogerychwyrndrobwllllantysiliogogogoch"

Figure 7. The *getWord3GramsAvgLen* feature exemplified by two contrasting examples

(1) Low scoring example	"Sebastian Gorka likes to be called" Dr. Gorka. "He gets his way only in conservative media."
(2) High scoring example	"There's a story behind **the** video of **a** man punching **a** kangaroo **the** head"

Figure 8. The *getDeterminers* feature exemplified by two contrasting examples (determiners are in bold)

14 strangely satisfying videos of melting cheese
10 ways to study you didn't know about
3 things you need to know

Figure 9. Examples of "listicles" or clickbait that start with numbers

(1) Low scoring example	"this is … unexpected"
(2) High scoring example	"It **Really Really Really Really Really** Sucks In New York City **Right Now**"

Figure 10. The *getAdvpCount* feature exemplified by two contrasting examples

in a larger sum than those of non-clickbait. The feature, called *getNumbersSum*, was the ninth most individually accurate feature (61.1 per cent).

ADVP frequencies

ADVPs occur in clickbait significantly more often than in non-clickbait. The *getAdvpCount* feature had an individual classification accuracy of 63.2 per cent and was the sixth most individually accurate (see Figure 10).

Common nouns and named entities

Clickbait contains distinctly fewer noun phrases than non-clickbait does. Lengthier non-clickbait allows for more descriptive information including factual references to entities in the world. The *getNPsCount* feature has the seventh highest individual feature classification accuracy of 62.7 per cent. Clickbait tends to omit proper nouns, often replaced by curiosity-triggering pronouns and determiners. We measured NNP-PERS (proper noun, person) part-of-speech tags with our *getNNPPERSCount* feature. References to people occurred slightly more frequently in clickbait than in non-clickbait.

Geographic named entities (NNP-LOC), exemplified in Figure 11(2), are more common in non-clickbait, possibly because they are more likely to occur in "headlinese"; *getNNPLOCCount* was the tenth most individually accurate feature (60.4 per cent).

Clickbaity trigger words versus "headlinese"

In general, clickbait differentiates itself from legitimate hyperlinks through an absence of "headlinese" register or the presence of certain expressions. Clickbaity "trigger words" do not normally appear in legitimate news headlines. We compiled a list of 19 words, representative of this definition regardless of their POS tag: a, an, everything, here, here's, heres [sic], how, meet, people, spot, that, there's, they, this, video, watch, ways, why, you (see a clickbait example in Figure 12(2)). The *containsTriggers* feature was the eighth most individually accurate feature (61.9 per cent).

(1) Low scoring example	"Fly healthier: this celebrity chef wants to change the way you eat in business class"
(2) High scoring example	"The **US, South Korea, Japan** start military drills off **North Korea**"

Figure 11. The *getNNPLOCCount* feature exemplified by two contrasting examples

(1) Low scoring example	"Impressive. Most impressive."
(2) High scoring example	"**There's a** right way to fall and it can save you a lot of grief if **you** know **how** to do it"

Figure 12. Samples containing no trigger words in contrast with those that have several trigger words

We were only able to incorporate a very limited number of "headlinese"-related features in our model. One feature, *containsOn*, was a simple condition that checks for the presence of the text "on" in the hyperlink text, as suggested in Mårdh (1980). This feature was fourteenth in accuracy but rare in the data. Mårdh's (1980) syntactic analysis of a corpus of headlines and the taxonomy of headlines, complete with parse trees, may find further applicability in manual feature engineering for the identification of clickbait and other deceptive strategies in digital media.

Distance measures by first position in string

We examined where singular proper nouns tend to occur in the hyperlink by measuring distance to the first singular proper noun from the start of the hyperlink. "Distance" is defined here as the number of part-of-speech tags before the first occurrence of an NNP part-of-speech tag. Three distance-based features were included in our classifier, but *getFirstNNPPos* is the most accurate individually of these three at 60 per cent or eleventh overall. To the best of our knowledge, such distance-related measures have not yet been mentioned in the clickbait identification literature.

Maximum distance to a proper noun (*maxDistToNNP*, which is illustrated in Figure 13) was initially expected to be a key feature based on initial work with the smaller 2017 Clickbait Challenge training set. Its effectiveness decreased when the LiT.RL Clickbait Detector was trained using the 2017 Clickbait Challenge validation set and, eventually, the combined set. The individual classification accuracy is 56.8 per cent or fifteenth overall. Similar features in our model include: *maxDistFromNPToNNP* (twenty-third in distance from a general noun to a proper noun), *getFirstNNPPos* (eleventh), and *avgDistToNNP* (eighteenth in average distance to a proper noun).

Word counts

Counting the number of words in a hyperlink can be an effective measure of clickbait compared to non-clickbait. Traditional headlines in newspapers are restricted by the print space and tend to be short, clipped to sound almost like telegraphic speech. We found that non-clickbait headlines tend to use around

(1) Low scoring example	"Cheap, widely available drug could stop thousands of mothers bleeding to death"
(2) High scoring example	"Wow, I was about to reveal something from **Season 7** and thought, What am I doing?, he was quoted as saying. #GoT"

Figure 13. Maximum distance to a proper noun feature illustrated

(1) Low scoring example	"thx 4 clearing that up 4 us"
(2) High scoring example	"4 things Hollywood gets wrong about archaeologists and 2 things it gets right"

Figure 14. The *getWordCount* feature exemplified by two contrasting examples

five words, while the pattern is almost reversed for the 10+ word headlines that are primarily clickbaity (Figure 14). This was the twelfth most individually accurate feature.

WordNet-based lexical similarity

Juxtaposed words may contribute to the level of sensationalism in clickbait. To experiment with this idea, we used a *WordNet* lexical similarity function from *pattern.en* for each general noun and proper noun to determine if they were similar in terms of their *WordNet* synsets (De Smedt and Daelemans 2012). Nouns in clickbait hyperlinks tend to be more dissimilar than nouns in non-clickbait. It is promising, but the use of WordNet slows down the classifier (in pre-processing, before training) by about five minutes. Further research is needed for other parts-of-speech similarity.

Vulgarity, crudeness, and swearing

Clickbait may overuse vulgar terms, including swearing and sexual terms that legitimate news sources would normally avoid. The combined dataset did not contain many of those terms so the effectiveness of this feature was lower than anticipated. The feature *getSwearCount* scores have been omitted from Table 3, as the feature underperformed and our source of vulgar and swear words changed frequently during the development of the Clickbait Detector and the News Verification Browser.

Discussion and related literature

Several research groups offer nuanced distinctions of clickbaiting techniques that are worth mentioning. We see them as techniques in which the quality of the news-like content is compromised, in one way or another, in terms of headline–article relevance, congruity, veracity, and informativeness.

Compromise in headline–article relevance (stance detection)

Bourgonje, Schneider, and Rehm (2017) consider relevance between a headline and its corresponding article body. They argue that knowing whether a headline is related to its article body (or not) is a first step in detection of clickbait and possibly false news. This nuanced task is phrased as "stance detection."

Compromise in headline–article congruity (incongruence detection)

Chesney et al. (2017) are interested in a similar task of "incongruence detection." They define "incongruence" as misleading the reader "by overstating the claim made later in the article," often significantly misrepresenting or exaggerating the findings reported in the article (Chesney et al. 2017). This conceptualization of headline–article mismatch is promising for clickbait detection and possibly for a broader inventory of disinformation strategies. Other fine-grained forms of logical fallacies that are not immediately apparent also warrant further consideration as predictive features in NLP-based ML-enabled detection tools. Such common forms of violations of reasoning in argumentation include, to name a few, the "straw man," "slippery slope," or "moral equivalence" tactics, which can potentially be identified with NLP techniques at the lexico-semantic, syntactic, and pragmatic levels and further boot-strapped with ML.

Interpersonal deception theory (ITD) distinguishes several kinds of deceptive strategies that have not yet been addressed at the fine-grain individual level beyond the general sense of deceptiveness that a text emits, based on leaked cues (Burgoon et al. 1994). Those strategies were specifically elaborated for the context of communication by Burgoon et al. (1994) and are widely accepted in deception detection, computer-mediated communication, and interpersonal psychology communities. The IDT's three-fold classification of deception varieties differentiates them using seven features (amount and sufficiency of information, degree of truthfulness, clarity, relevance, ownership, and intent). Falsification (lying or describing "preferred reality") is the most deceptive and least readily detected since it is most prevalent and practised most often; it is followed by concealment (omitting material facts) and equivocation (dodging, skirting issues by changing the subject, or offering indirect responses) (Burgoon et al. 1994). The latter is most readily detected since it offers the least amount of clarity, completeness, directness, and often induces suspicion (Burgoon et al. 1994). (For overviews of deceptive strategies and information manipulation tactics, see Rubin and Chen (2012) for information science literature; Rubin (2017a) for social media research methods literature; and Rubin (2018) for journalism literature in French). More refined cues of other deceptive strategies are yet to be explored for their usefulness beyond a general sense of deceptiveness conveyed by incongruity.

Compromise in veracity (deception detection)

In the early clickbait detection literature, clickbait was conceptualized as a compromise in veracity that needed to be addressed with automated measures. Clickbait was intentionally misleading content that compromised the quest for truth and interfered with sense making based on facts (Anderson and Rainie 2017). This perspective clearly emphasized the contrast between traditional legitimate news writing and sensationalized tabloids, as the roots of clickbaiting practices are very pervasive in current digital media. In fact, the primary danger posed by tabloidization is not that "hard news" topics (e.g., politics, science, economics) will be replaced by "soft news" (e.g., entertainment, sports, gossip), according to Reinemann et al.'s (2011) definitions) but, rather, that the focus on attention-grabbing, shareable reporting has led to "the willful blurring of lines between fact and fiction" (O'Neil 2013, cited in Anderson and Rainie 2017). Such a perspective provides a clear contrast and a path for detecting clickbait as a variety of misleading strategies inspired by sensationalisms in news and advertisement-revenue rewards. The LiT.RL Clickbait Detector was developed with this model in mind and, specifically, as part of the broader agenda for automatic identification of varieties of "fakes" in other forms of deceptive strategies to manipulate digital content such as falsifications and satirical "fakes" (Rubin, Chen, and Conroy 2015). The detector can be broadly applied through the use of the LiT.RL News Verification Browser.[1]

Compromise in informativeness (ad detection)

Potthast et al. (2016) frame clickbait as "web content advertisement." Clickbaiting is then seen as a marketing technique for attracting readers, even in the absence of interesting content (see earlier discussion on how clickbait content is promoted). From this perspective, NLP-based insights into clickbait could be gleaned from further works on understanding features of "sponsored content in disguise," also known as native advertisements (see, for instance, Cornwell and Rubin 2017, 2019).

For Papadopoulou et al. (2017, 1), clickbait is neither about news nor ads, it is rather "a short post in a social network platform" that "manages to attract traffic but the content fails to deliver." Such a perspective is further removed from crafting attention-grabbing tabloid headlines, and, correspondingly, the datasets used for binary distinctions become murkier. For instance, the 2017 Clickbait Challenge dataset that we used draws on Twitter data and seems to include tweets or re-tweets that could not be clearly marked as clickbait or not (Figure 2). Another research group broadly studied Facebook posts, and their massive data included "unreliable media" freely mixing conspiracy, satire, and junk science articles with clickbait (Rony, Hassan, and Yousuf 2017). The legitimacy of broadening the definition of the clickbait phenomenon will be verified with time, as more research on clickbait perceptions emerges.

Limitations and contributions

Having gone through the process of supervised ML for clickbait detection with two datasets, we observed the power of the large combined datasets as well as their limitations. Unclear operationalization of the concept of "clickbaitiness" resulted in the presence of ambiguous data, and, therefore, a large chunk of data had to be excluded from training (0.1–0.6 mean scores). In addition, the dataset was heavy on typical social media personal comments, posts, tweets, and re-tweets which broadens the boundaries of non-clickbait to any post on social media (for example, Figures 5(1) and 14(1)). While research is catching up on users' perceptions of what is and what is not clickbait (for example, Chen and Rubin 2017), we call for greater awareness of users' subjectivity in the interpretation of the phenomenon and for more clarity in data collection criteria by developers. Additional efforts to create a well-curated "gold standard" are needed for the use of NLP/ML techniques in the task of automated clickbait detection.

Our current contribution is a unique combination of predictive features with an accurate clickbait detection system. The detector is lightweight with respect to system resource usage on a relatively modern personal computer (as of 2019), fast at its binary classification task, and is available via GitHub as an open source code.[2] The LiT.RL Clickbait Detector is a key component of the LiT.RL News Verification Browser, a suite of analytical tools aiming for automated detection of three types of disinformation (Rubin 2017b; Rubin et al. 2019): news falsifications (Asubiaro and Rubin 2018); satirical news "fakes" (Rubin et al. 2016); and clickbait.

Conclusions and future work

This article describes a newly developed LiT.RL Clickbait Detector (pronounced "literal"), a binary classifier that uses 38 NLP-based features that distinguish clickbait headlines from non-clickbait. Much attention in this article is given to the description of the nature of clickbait, its frequent patterns and trigger words as opposed to more standard news and "headlinese" register. Despite relatively high success rates in binary classification results with our methods (94 per cent accuracy) and with those leading in the field (such as 98 per cent, shown in Rony, Hassan, and Yousuf (2017)), we caution the community against rushing to conclusions that clickbait detection is a solved problem. Upon the manual review of the training data, we identified ambiguities in data labelling and differences in how researchers conceptualize the phenomenon. The use of the LiT.RL Clickbait Detector as part of the LiT.RL News Verification Browser indicates that this class of detectors likely offer fluctuating real-world performance when applied directly to the Internet news website, calling into question the meaning of high test set accuracy scores. This concern may be investigated using larger qualitative studies with more user involvement, which we plan to conduct in the near future. We further express concern about the potential variance in subjective interpretations of the phenomenon by crowd-sourced judges. The "murky middle" within the training data is problematic. Additional research on what people generally agree to call "clickbait" is needed. We also call for better curated "gold standard" datasets.

Clickbait detection itself is an important effort in assisting users in revealing manipulative behaviours online. The task should be incorporated into a broader set of measures for news verification to label or filter out a variety of misleading or deceptive content, including outright falsification in news, which is also known as "fake news" (Rubin 2017b). The LiT.RL News Verification Browser represents such an attempt, and it is a working "proof of concept" emphasizing the necessity to have a suite of tools to combat misinformation and disinformation online. To be successful at identifying deceptive phenomena, the NLP/ML research community needs to constantly monitor new and creative developments in online content generation "tricks." We need to see through euphemisms such as "captivating," "engaging," and "meaningful content for millennials" and call it what it is: a psychological manipulative trick through language use and "candy for the brain." We also need to be aware of how tempting it must be for traditional journalists to resort to using clickbait techniques to gain visibility for their content and bring new audiences to their "think pieces" (for example, the traditional news site homepage reveals 37 per cent "clickbaitiness") (see Figure 3(2)). "The clickbait creep" may be affecting the profession in profound ways, and it is unclear how long the legitimate news outlets will strongly resist it. As digital journalism is undergoing significant changes in its models for how news (or "newsy" content) is being produced, disseminated, and funded, more research is clearly needed to understand and monitor the content generators' practices, the accuracy of prediction models, and the evolving perceptions of clickbait.

Acknowledgements

This research has been funded by the Social Sciences and Humanities Research Council Insight Grant (no. 435-2015-0065) awarded to Victoria Rubin for the project Digital Deception Detection: Identifying Deliberate Misinformation in Online News. Many thanks to Yimin Chen, Sarah Cornwell, and Toluwase Asubiaro, doctoral students and graduate research assistants at the Language and Information Technology Research Lab (LiT.RL) at Western University, for their helpful suggestions, linguistic observations, and insights on formulaic clickbait tactics that were glimpsed from other clickbait perception studies, run prior and in parallel with the Clickbait Detector research and development.

Notes

1 See 'Litrl Browser Experimental 0.14.0.1 Public,' *Github*, https://github.com/litrl/litrl_code/releases/tag/exp-0.14.0.1 (accessed 9 September 2019).
2 See https://github.com/litrl/litrl_code (accessed 9 September 2019).

References

Anderson, Janna, and Lee Rainie. 2017. "The Future of Truth and Misinformation Online." *Pew Research Center: Internet and Technology*. Accessed September 9, 2019. http://www.pewinternet.org/2017/10/19/the-future-of-truth-and-misinformation-online/.

Asubiaro, Toluwase, and Victoria Rubin. 2018. "Comparing Features of Fabricated and Legitimate Political News in Digital Environments (2016–2017)." In *Proceedings of the American Society for Information Science and Technology Annual Meeting (ASIST2018)*, 747–50. Accessed September 9, 2019. https://www.asist.org/wp-content/uploads/Final-81st-Annual-Meeting-Proceedings-1.pdf.

Bird, Steven, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit.* Sebastopol, CA: O'Reilly Media.

Bourgonje, Peter, Julian Moreno Schneider, and Georg Rehm. 2017. "From Clickbait to Fake News Detection: An Approach Based on Detecting the Stance of Headlines to Articles." In *Proceedings of the 2017 EMNLP Workshop on Natural Language Processing Meets Journalism*, 84–89. Copenhagen, Denmark: Association for Computational Linguistics.

Buller, David B., and Judee K. Burgoon. 1996. "Interpersonal Deception Theory." *Communication Theory* 6 (3): 203–42. https://doi.org/10.1111/j.1468-2885.1996.tb00127.x.

Burgoon, Judee K., David B. Buller, Amy S. Ebesu, and Patricia Rockwell. 1994. "Interpersonal Deception: v. Accuracy in Deception Detection." *Communication Monographs* 61 (4): 303–25. https://doi.org/10.1080/03637759409376340.

Chakraborty, Abhijnan, Bhargavi Paranjape, Sourya Kakarla, and Niloy Ganguly. 2016. "Stop Clickbait: Detecting and Preventing Clickbaits in Online News Media." In *Proceedings of the IEEE-ACM International Conference on Advances in Social*

Networks Analysis and Mining (ASONAM), 9–16. Davis, CA: IEEE Press Piscataway. https://doi.org/10.1109/ASONAM.2016.7752207.

Chen, Yimin, Niall J. Conroy, and Victoria L. Rubin. 2015. "Misleading Online Content: Recognizing Clickbait as 'False News'." In *Proceedings of the ACM WMDD-ICMI Workshop on Multimodal Deception Detection (WMDD2015), Joint with the International Conference on Multimodal Interaction (ICMI2015)*, 15–19. Seattle, WA: ACM Press. https://doi.org/10.1145/2823465.2823467.

Chen, Yimin, and Victoria L. Rubin. 2017. "Perceptions of Clickbait: A Q-Methodology Approach." In *Proceedings of the 45th Annual Conference of the Canadian Association of Information Science*, n.p. Toronto, ON: Canadian Association for Information Science.

Chesney, Sophie, Maria Liakata, Massimo Poesio, and Matthew Purver. 2017. "Incongruent Headlines: Yet Another Way to Mislead Your Readers." In *Proceedings of the 2017 EMNLP Workshop on Natural Language Processing meets Journalism*, 56–61. Copenhagen: Association for Computational Linguistics.

Cornwell, Sarah E. L., and Victoria L. Rubin. 2017. "Native Advertising: Ads in Disguise as Editorials." In *Proceedings of the 45th Annual Conference of the Canadian Association of Information Science*, n.p. Toronto: Canadian Association for Information Science.

Cornwell, Sarah E. L., and Victoria L. Rubin. 2019. "What Am I Reading?: Article-style Native Advertisements in Canadian Newspapers." In *Proceedings of the Annual Hawaii International Conference on System Sciences (HICSS-52)*, n.p. Maui, HI: IEEE Computer Society. Accessed January 27, 2019. http://hicss.hawaii.edu/resources/proceedings/.

De Smedt, Tom, and Walter Daelemans. 2012. "Pattern for Python." *Journal of Machine Learning Research* 13: 2063–67.

Diply. 2017. "About Us." Accessed January 27, 2019. https://diply.com/static/about.

Elyashar, Aviad, Jorge Bendahan, and Rami Puzis. 2017. "Detecting Clickbait in Online Social Media: You Won't Believe How We Did It." *arXiv.org*. Accessed January 27, 2019. https://arxiv.org/abs/1710.06699v1.

Frampton, Ben. 2015. "Clickbait: The Changing Face of Online Journalism." *BBC News*, September 14. Accessed March 28, 2018. http://www.bbc.com/news/uk-wales-34213693.

Gollub, Tim, Martin Potthast, Matthias Hagen, and Benno Stein. 2017. "Clickbait Challenge 2017." Accessed January 27, 2019. http://www.clickbait-challenge.org.

Grigorev, Alexey. 2017. "Identifying Clickbait Posts on Social Media with an Ensemble of Linear Models." *arXiv.org*. Accessed January 27, 2019. https://arxiv.org/abs/1710.00399.

Indurthi, Vijayasaradhi, and Subba Reddy Oota. 2017. "Clickbait Detection Using Word Embeddings." *arXiv.org*. https://arxiv.org/abs/1710.02861.

Jack, Caroline. 2017. *Lexicon of Lies: Terms for Problematic Information*. New York: Data and Society Research Institute.

Jones, Eric, Travis Oliphant, and Pearu Peterson. 2001. "SciPy: Open Source Scientific Tools for Python." *SciPy*. Accessed January 27, 2019. http://www.scipy.org.

Mårdh, Ingrid. 1980. *Headlinese: On the Grammar of English Front Page Headlines*. Lund: CWK Gleerup.

O'Neil, Luke. 2013. "The Year We Broke the Internet." *Esquire*, December 23.

Owen, Taylor. 2017. "Dr. Taylor Owen on How Internet Monopolies Threaten Democracy." 2017 CBC Dalton Camp Lectures. *CBC Radio*. Accessed January 27,

2019. http://www.cbc.ca/radio/ideas/dr-taylor-owen-on-how-internet-monopolies-threaten-democracy-the-2017-dalton-camp-lecture-1.4451101.

Papadopoulou, Olga, Markos Zampoglou, Symeon Papadopoulos, and Ioannis Kompatsiaris. 2017. "A Two-Level Classification Approach for Detecting Clickbait Posts Using Text-Based Features." *arXiv.org*. Accessed January 27, 2019. https://arxiv.org/abs/1710.08528.

Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, and Vincent Dubourg. 2011. "Scikit-learn: Machine Learning in Python." *Journal of Machine Learning Research* 12: 2825–30.

Potthast, Martin, Tim Gollub, Kristof Komlossy, Sebastian Schuster, Matti Wiegmann, Erika Patricia Garces Fernandez, Matthias Hagen, and Benno Stein. 2018. "Crowdsourcing a Large Corpus of Clickbait on Twitter." In *Proceedings of the 27th International Conference on Computational Linguistics (COLING 2018)*, 1498–507. Accessed January 27, 2019. https://webis.de/data/webis-clickbait-17.html.

Potthast, Martin, Sebastian Köpsel, Benno Stein, and Mattias Hagen. 2016. "Clickbait Detection." In *Advances in Information Retrieval. 38th European Conference on IR Research (ECIR 16)*, 810–17. Berlin: Springer. https://doi.org/10.1007/978-3-319-30671-1_72.

Reinemann, Carsten, James Stanyer, Sebastian Scherr, and Guido Legnante. 2011. "Hard and Soft News: A Review of Concepts, Operationalizations and Key Findings." *Journalism* 13 (2): 1–19. https://doi.org/10.1177/1464884911427803.

Rony, Md Main Uddin, Naeemul Hassan, and Mahammad Yousuf. 2017. "Diving Deep into Clickbaits: Who Use Them to What Extents in Which Topics with What Effects?" *arXiv.org*. Accessed January 27, 2019. https://arxiv.org/abs/1703.09400v1.

Rubin, Victoria L. 2017a. "Chapter 21. Deception Detection and Rumor: Debunking for Social Media." In *The SAGE Handbook of Social Media Research Methods*, ed. Luke Sloan and Anabel Quan-Haase, n.p. Los Angeles, CA: Sage Publications.

———. 2017b. "News Verification Suite: Towards System Design to Supplement Reporters' and Editors' Judgements." In *Proceedings of the 45th Annual Conference of the Canadian Association of Information Science*, n.p. Toronto: Canadian Association for Information Science.

———. 2018. "La détection des nouvelles trompeuses et fallacieuses dans les contenus en ligne" / "Detecting Deceptive and Misleading News in Online Content." In *Les fausses nouvelles, nouveaux visages, nouveaux défis. Comment déterminer la valeur de l'information dans les sociétés démocratiques?* ed. Florian Sauvageau, Simon Thibault, and Pierre Trudel, 153–72. Quebec: Presses de l'Université Laval. Accessed January 27, 2019. https://www.pulaval.com/produit/les-fausses-nouvelles-nouveaux-visages-nouveaux-defis-comment-determiner-la-valeur-de-l-information-dans-les-societes-democratiques.

Rubin, Victoria L., Chris Brogly, Nadia Conroy, Yimin Chen, Sarah E. L. Cornwell, and Toluwase V. Asubiaro. 2018. "litrl/litrl_code: LiT.RL Browser Experimental." *Github*. Accessed January 27, 2019. https://github.com/litrl/litrl_code/releases/tag/exp-0.14.0.1.

Rubin, Victoria L., Chris Brogly, Nadia Conroy, Yimin Chen, Sarah E. L. Cornwell, and Toluwase V. Asubiaro. 2019. "A News Verification Browser for the Detection of Clickbait, Satire, and Falsified News." *Journal of Open Source Software* 4 (35): 1208. https://doi.org/10.21105/joss.01208.

Rubin, Victoria L., and Yimin Chen. 2012. "Information Manipulation Classification Theory for LIS and NLP." In *Proceedings of the American Society for Information Science and Technology Annual Meeting (ASIST2012)*, n.p. St. Louis, MO:

American Society for Information Science and Technology. https://doi.org/10.1002/meet.14504901353

Rubin, Victoria L., Yimin Chen, and Niall J Conroy. 2015. "Deception Detection for News: Three Types of Fakes." In *Proceedings of the Association for Information Science and Technology Annual Meeting (ASIST2015)*, n.p. St. Louis, MO: American Society for Information Science and Technology.

Rubin, Victoria L., Niall J. Conroy, Yimin Chen, and Sarah Cornwell. 2016. "Fake News or Truth? Using Satirical Cues to Detect Potentially Misleading News." In *Proceedings of the Workshop on Computational Approaches to Deception Detection at the 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-CADD2016)*, n.p. San Diego, CA: Association for Computational Linguistics.

Silverman, Craig. 2015. *Lies, Damn Lies, and Viral Content: How News Websites Spread (and Debunk) Online Rumors, Unverified Claims and Misinformation*. New York: Tow Center for Digital Journalism.

van der Walt, Stéfan, S. Chris Colbert, and Gaël Varoquaux. 2011. "The NumPy Array: A Structure for Efficient Numerical Computation." *Computing in Science and Engineering* 3 (2): 22–30. https://doi.org/10.1109/MCSE.2011.37.

Wei, Wei, and Xiaojun Wan. 2017. "Learning to Identify Ambiguous and Misleading News Headlines." *arXiv.org*. Accessed January 27, 2019. https://arxiv.org/pdf/1705.06031.pdf.

Zhou, Lina, Judee K. Burgoon, Jay F. Nunamaker, and Doug Twitchell. 2004. "Automating Linguistics-Based Cues for Detecting Deception in Text-Based Asynchronous Computer-Mediated Communications." *Group Decision and Negotiation* 13 (1): 81–106. https://doi.org/10.1023/b:grup.0000011944.62889.6f.

Zhou, Yiwei. 2017. "Clickbait Detection in Tweets Using Self-attentive Network." *arXiv.org*. Accessed January 27, 2019. http://arXiv.org/abs/1710.05364v1.