'You won't believe what happened next!' Building a News Article Validation Tool with NLP

K Iwasaki **Sohag Desai** **Phat Doan**

Abstract

Reading news has become incredibly more convenient with the rise of the Internet, as readers can quickly access in real-time quality journalism at a push of a button. However, as the rate of information grows exponentially, our human brain is frequently overwhelmed by too many choices. This, in turn, leads us to find ourselves looking for patterns, trying to simplify issues and giving precedence to opinions that reinforce what we already believe to be true. Our project aims to leverage Natural Language Processing techniques to create a tool that enables news consumers to quickly identify non-clickbait news. We break down our proposed tool into two separate classifiers: clickbait detection at the title level, and text summarization for clickbait classification at article content level.

1 Introduction

Clickbait news articles are endemic for online readers. These articles often contain misleading titles or purposely withholding information required to understand the content. The term also describe a news headline which will tempt a user to follow by using provocative and catchy wording.Their main purpose is to entice users to click on the articles to generate revenues, the content on the landing page is usually of low quality. At the same time, the content of these articles is often of low quality. A few examples of clickbait titles extracted from our dataset are: "*Should I Get Bings?*", "*Which TV Female Friend Group Do You Belong In?*". Non-clickbait titles that we identified are: "*White House Announces International Meetings to Address Energy and Climate Issues*", "*France Approves Crackdown on Internet Piracy*".

As reading news has become incredibly more convenient with the rise of the Internet and the rate of information grows exponentially, our human brain is frequently overwhelmed by too many choices. At the same time, readers are often frustrated with news articles often not living up to the expectation due to misleading titles and exaggerating the content on the landing page. Our project aims to leverage Natural Language Processing techniques to create a tool that enables news consumers to quickly identify non-clickbait news that are of their interests. Given an article title and its content, we will classify it into one of two classes: clickbait or not clickbait. We divide our problem into 2 tasks: classification at title-level and summarize news articles for title comparison. To evaluate the efficacy of our model, we conduct experiments on two datasets. First dataset is provided by a team memberâĂŹs company that consist of 15,999 clickbait and 16,000 non-clickbait headlines. Second dataset is scraped from Reddit /r/savedyouaclick for clickbait articles and /r/news for non-clickbait articles.

Our intended contributions to the current research: 1/ build an open-source clickbait dataset that contains article contents in addition to the headlines. 2/ propose a new approach to clickbait classification by applying text summarization technique to article contents and comparing the summarization results to article title. Our team firmly believes that while title-level clickbait classification is superior at the moment, untrustworthy content generators will eventually outsmart these classifications. Therefore, it would be important to forge ahead and contribute to current research of content-level clickbait classification.

2 Background

Much research has been done on clickbait classification in the recent years. Most notable works are Anand et. al. 2016, Biyani et. al. 2016,

and Chakraborty et al (2016). Chakraborty utilizes SVM, decision tree, and random forest to predict clickbait with great results of 0.97 ROC-AUC. Anand explored several neural network models such as Recurrent neural network (RNN) architectures, Gated Recurrent Units (GRU) and standard RNNs and achieved 0.99 ROC-AUC with word embedding LSTM neural network. Both teams utilize... dataset for their models. However, these research focus solely on clickbait classification at the title level. Biyani et. al. 2016 expanded the research to include title, url, and article body as input for the classification. Biyani and team used novel informality features for clickbait classification, which have not been previously used, and showed that these features are the most important. For the body, Biyani applied textual similarity algorithm between titles and top sentences of the body to detect clickbait. Through an ensemble classifier, engineered features, and 7677 word features, Biyani achieved 0.755 precision and 0.760 recall on the test set. While this is not as strong a classifier as title-only classification, its approach is arguably more holistic by taking consideration of both title and article content.

Text summarization is to concisely recapitulate a text. Current research indicates two approaches to summarization: extractive and abstractive. Extractive methods build summaries exclusively using phrases and/or sentences from the source text. This can be formulated using algorithms. Abstractive methods, on the other hand, utilize different words and phrases not appeared in the source text for their summarization. This is inherently more difficult to build from a machine learning algorithm perspective. Most research in the past has focus on extractive (Kupiec et al., 1995; Paice, 1990; Saggion and Poibeau, 2013). On the abstractive front, recent research has been successful with the use of sequence-to-sequence models such as recurrent neural networks (RNNs) to both ingest, digest, and freely generate text (Chopra et al., 2016; Nallapati et al., 2016; Rush et al., 2015; Zeng et al., 2016). From our observation, See et. al. 2017 created a model using Pointer-Gen + Coverage algorithm that outperformed all previously researched model. See 2017 evaluated their models with the standard ROUGE metric (Lin, 2004b), reporting the F1 scores for ROUGE1, ROUGE-2 and ROUGE-L. These metrics respectively measure the word-overlap, bigram-overlap, and longest common sequence between the reference summary and the summary to be evaluated.

3 Method

3.1 Proposed work

Given an article title and its content, classify it into one of two classes: clickbait or not clickbait. We divide our problem into two tasks: classification at title-level and summarize news articles' major topics for document representation. Our work expands previous work to include title, abstract, and news content as input for our detection algorithms. We use word embeddings CNN neural network and hyperparameter tuning as the main methodology for our clickbait classifier at the title level. For text summarization, we use See et. al. 2017 as the foundation of our work to form the hypothetical title, which is then compared with the actual article title using textual similarity techniques.

To evaluate the efficacy of our model, we conduct experiments on two datasets. First dataset is provided by a team memberâĂŹs company that consist of 15,999 clickbait and 16,000 non-clickbait headlines. Second dataset is scraped from Reddit /r/savedyouaclick for clickbait articles and /r/news for non-clickbait articles. We build a custom scraper and parser to get the headlines and article content from Reddit through its json source code. The two mentioned subreddits provide a naturally curated place for clickbait and non-clickbait articles. /r/savedyouaclick is moderated by Reddit moderator as a forum for sharing interesting clickbait articles. Redditors would share a clickbait article in this general format: "*5 Reasons Tinder Is Dead And Is Never Coming Back | It isn't, the writer just deleted it from their phone*". The first part before "|" contains the actual article title, while the second part is the Redditor's summarization of the article content. This setting automatically provides us a great training data set for our clickbait classification at the content level. The /r/news, on the other hand, provides moderated balance-viewed, non-political news from around the world and will serve as the validated non-clickbait training set.

Using this Reddit-scraped dataset, we will build a news summarization that we then can compare with the actual article's title to determine if it is a clickbait or not.

4 Exploratory Data Analysis

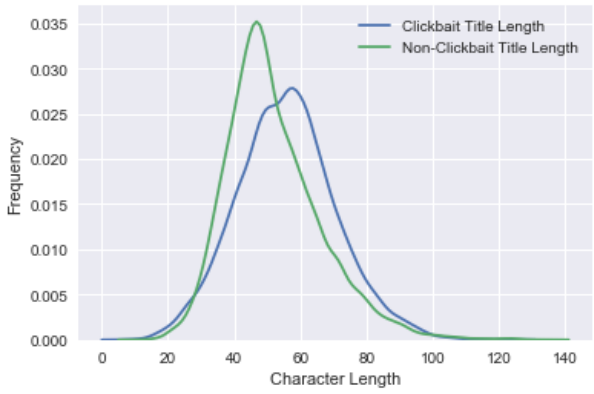
Conducting Exploratory Data Analysis on the first dataset, we performed the following operations. First, we prepared the data by splitting the dataset into test and train sets with an 80/20 split. In the first dataset, we have a total of 32,000 training samples, of which 15,999 are labeled clickbait and 16,001 are non-clickbait. The train set comprises 25,600 samples and the test set comprises 6,400.

Our next step was to query the data. We examined the following aspects of the data in order to identify features that we could use to classify the data:

- Character count of title
- Word count of title
- Common words used
- Most frequent punctuation characters used

We observed some interesting patterns which can be used towards feature engineering.

This graph shows a plot of character count distribution for clickbait and non-clickbait samples.

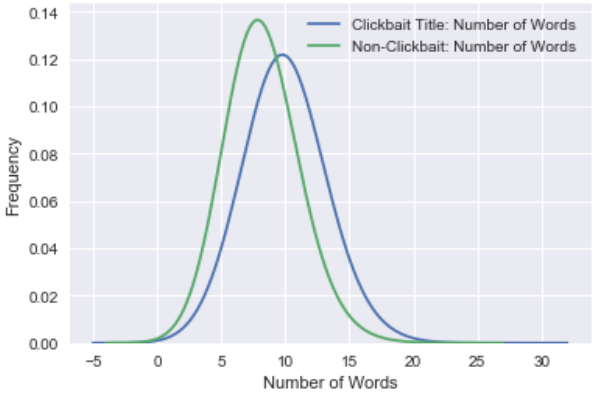


The following table shows the top 25 words used in clickbait and non-clickbait headlines:

Rank	Clickbait		Non-Clickbait	
	Word	Frequency	Word	Frequency
1	You	4805	in	4354
2	The	4723	to	3394
3	To	3231	of	2662
4	A	2600	for	1691
5	Your	2536	the	1246
6	Of	2422	on	1236
7	Are	1938	and	1165
8	In	1910	'a	1118
9	This	1804	at	955
10	That	1776	New	607
11	And	1639	US	596
12	Is	1624	by	532
13	On	1449	U.S.	523
14	For	1393	after	512
15	Will	1242	as	474
16	What	1123	Is	414
17	About	1052	from	345
18	Things	995	with	344
19	With	958	dies	336
20	Who	946	With	306
21	People	909	over	297
22	How	899	dead	294
23	Which	893	killed	289
24	From	825	UK	282
25	We	806	Australian	270

Table 1: Top 25 Words

This graph shows a plot of word count distribution for clickbait and non-clickbait samples.



The following table shows the top 25 punctuation characters used in clickbait and non-clickbait headlines:

Rank	Clickbait		Non-Clickbait	
	Punctuation	Frequency	Punctuation	Frequency
1	SPACE	143074	SPACE	11512
2	"	4751	,	3504
3	""	4198	.	2687
4	-	1151	'	1671
5	,	737	-	1604
6	.	390	:	711
7	:	362	"	390
8	(	59	;	228
9	)	59	$	202
10	#	57	?	119
11	$	57	%	117
12	&	49	/	60
13	?	43	&	50
14	!	32	'	38
15	*	29	(	32
16	%	27	)	32
17	/	21	£	22
18	~	20	–	20
19	@	6	—	16
20	+	5	€	10
21	|	5	‘	9
22	=	2	!	8
23	;	2	+	5
24	[	1	“	2
25	]	1	”	2

Table 2: Top 25 Punctuations

We observed the following:

- Title lengths for both word count as well as character count for non-clickbait tend to be larger
- "You" and "Your" figure as the top five most frequent words for clickbait; prepositions "in", "to", "of", "for", etc. figure more frequently for non-clickbait
- Quotation marks ("") are frequent and figure in the top five in clickbait and not so frequent in non-clickbait
- Euro (€) and Pound (£) characters are in the top 25 for non-clickbait, whereas they don't figure in the top 25 for clickbait

5 Completed Work

When evaluating the approaches to tackle the clickbait classification problem, we reviewed prior work in applying Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) to this and related NLP problems. CNN is typically used for image recognition problem and is not commonly applied to NLP.

However, Yin et al (2017) and Vu et al (2016) have done work in comparing the use of CNN versus RNN for classification tasks. They report higher performance of CNN over RNN and give evidence that CNN and RNN provide complementary information. RNN computes a weighted combination of all words in the sentence; CNN extracts the most informative ngrams for the relation and only consider their resulting activations. Both Wen et al. (2016) and Adel and Schutze (2017) support CNN over GRU/LSTM for classification of long sentences.

In addition, Yin et al. (2016) achieve better performance of attention-based CNN than attention-based LSTM for answer selection. Dauphin et al. (2016) further argue that a fine-tuned gated CNN can also model long context dependency, getting new state-of-the-art in language modeling above all RNN competitors.

This motivated our approach to use CNN for the clickbait classification problem. Our strategy involved preserving the structural integrity of the text and allowing the CNN to do feature engineering.

We used the following hyperparameters:

- Filter sizes: 3,4,5
- Number of filters: 512
- Dropout: 0.5
- Embedding dimension: 100
- Number of epochs: 5

We trained on 23040 samples and validated on 2560 samples. The following are the other parameters:

- Sequence length: 25,600
- Number of classes: 2
- Batch size: 32

For learning the embedding, we considered two approaches: *word2vec* and *GloVe*. The former is a predictive approach and the latter a count-based approach. Predictive models learn their vectors in order to improve their predictive ability of the loss of predicting the target words from the context words given the vector representations. Count-based models learn their vectors by essentially doing dimensionality reduction on the co-occurrence counts matrix.

GloVe has been extensively used and proven to be very effective. It is an unsupervised learning algorithm for obtaining vector representations for words. Training is performed on aggregated global word-word co-occurrence statistics from a corpus, and the resulting representations showcase interesting linear substructures of the word vector space. It has been trained on Wikipedia 2014 (6B words) and pre-trained word vectors are available for use. This motivated the use of *GloVe* for our embedding layer.

For our optimizer, we considered *Adagrad* and *Adam*. *Adagrad* is an algorithm for gradient-based optimization that adapts the learning rate to the parameters, performing larger updates for infrequent and smaller updates for frequent parameters. For this reason, it is well-suited for dealing with sparse data. Adaptive Moment Estimation (*Adam*) is another method that computes adaptive learning rates for each parameter. In addition to storing an exponentially decaying average of past squared gradients, Adam also keeps an exponentially decaying average of past gradients. Since our data is not sparse we selected *Adam* over *Adagrad*.

We used the following values for optimization:

- Optimizer: adam
- Learning rate: 0.0001
- Exponential decay rate for the 1st moment estimates (beta_1): 0.9
- Exponential decay rate for the 2nd moment estimates (beta_2): 0.999
- Constant for numerical stability (epsilon): 1e-08
- Learning rate decay over each update cycle: 0.0001
- Loss function: binary cross-entropy
- Metrics: accuracy

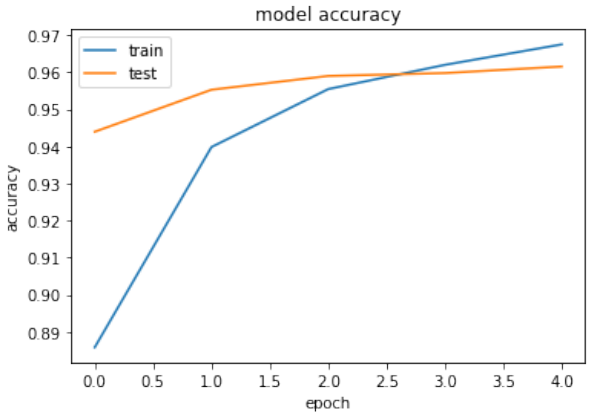
6 Results and Discussion

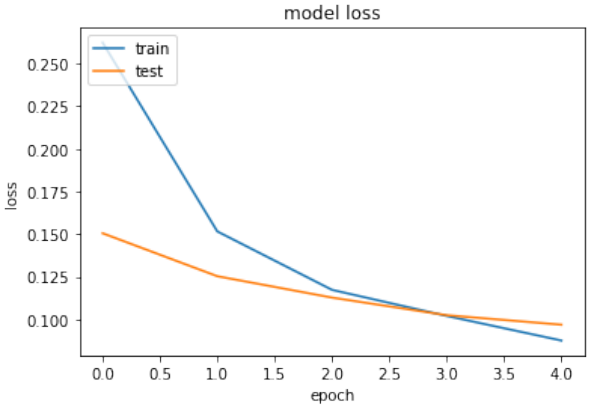
6.1 Results

The following are the results after our initial training and running the model:

Actual/Predicted	Clickbait	Non-Clickbait
Clickbait	3097	97
Non-clickbait	130	3076

Table 3: Confusion Matrix





ID	title	category	clickbait_score
392	8 Things I Would Love to Give My Kids This Christmas	EN_US_LIFESTYLE	0.999997
1769	5 Things You Need To Know About Interacting With Me, a Fat Girl Who Wears Revealing Clothes	EN_US_LIFESTYLE	0.999984
1002	53 Fascinating Facts You Probably Didn't Know About Disney Films	EN_US_ENTERTAINMENT	0.999982
242	16 Things You Think He Cares About But He Really Doesnt	EN_US_LIFESTYLE	0.999976
1587	16 Adorable Animals Who Are Too Tired To Animal	EN_US_LIFESTYLE	0.999928
1718	9 Books You Need To Read This Summer	EN_US_LIFESTYLE	0.999912
353	24 Kids Who Are Wise Beyond Their Years	EN_US_LIFESTYLE	0.999910
368	Victor Martinez's Sexy Dance Is Very Sexy	EN_US_ENTERTAINMENT	0.999858
730	6 Bizarre Side Effects of Foods You Eat Every Day	EN_US_LIFESTYLE	0.999851
605	5 things you need to know Monday	EN_US_NATIONAL	0.999817

Table 4: Sample Accuracy

	precision	recall	f1-score	support
0	0.96	0.97	0.96	3194
1	0.97	0.96	0.96	3206
avg / total	0.96	0.96	0.96	6400

Table 5: Classification Report

ID	Original Text	Reverted Text	Prediction	Actual
10	2 Ex-Timesmen Say They Had a Tip on Watergate First	2 ex say they had a tip on watergate first	Clickbait	Non-clickbait
81	The Online Ad That Knows Where Your Friends Shop	the online ad that knows where your friends shop	Clickbait	Non-clickbait
103	Hangover Cures The Internet Swears By	hangover cures the internet swears by	Non-clickbait	Clickbait
105	Drake Got A New Tattoo In Honor Of Toronto	drake got a new tattoo in honor of toronto	Non-clickbait	Clickbait
171	This Artist Painted Lin-Manuel Miranda's "Hamilton" On A Ten-Dollar Bill	this artist painted lin manuel miranda 's hamilton on a ten dollar bill	Non-clickbait	Clickbait
176	Publication date for last Harry Potter book announced	publication date for last harry potter book announced	Clickbait	Non-clickbait
196	How To Have A Mouthgasm At Taco Bell	how to have a at taco bell	Non-clickbait	Clickbait
222	Non-Scientists React To Science Things	non scientists react to science things	Non-clickbait	Clickbait
256	The Strange Season of David Wright	the strange season of david wright	Clickbait	Non-clickbait
268	Curvy Targeting Is Banned For Gmail Advertisers	curvy targeting is banned for gmail advertisers	Non-clickbait	Clickbait
293	This Optimistic Man Tried To Sneak 14 Bottles Of Liquor Into Saudi Arabia In His Underwear	this optimistic man tried to 14 bottles of liquor into saudi arabia in his underwear	Non-clickbait	Clickbait
302	Accounts, People, Miscellany	accounts people miscellany	Clickbait	Non-clickbait
308	Dating Struggles For Extroverted Ladies	dating struggles for ladies	Non-clickbait	Clickbait
345	A Black Body On Trial: The Conviction Of HIV-Positive "Tiger Mandingo"	a black body on trial the conviction of hiv positive tiger	Non-clickbait	Clickbait
362	Modifying Mortgages Can Be Tricky	modifying mortgages can be tricky	Clickbait	Non-clickbait
416	When It Rains In L.A	when it rains in l a	Non-clickbait	Clickbait
424	Ben Stiller Made A Super Bowl Commercial For Female Viagra	ben made a super bowl commercial for female	Non-clickbait	Clickbait
426	Don Vito Has Died At 59	don has died at 59	Non-clickbait	Clickbait
446	When the Mall Looks More Like Main Street	when the mall looks more like main street	Clickbait	Non-clickbait
460	Helpline: Do you know this pianist?	do you know this	Clickbait	Non-clickbait
575	These Could Be The Last Days Of The Messiest Party On Earth	these could be the last days of the party on earth	Non-clickbait	Clickbait
694	Losing Yourself in HDTV Is a Few Tweaks Away	losing yourself in is a few away	Clickbait	Non-clickbait
696	Finding the Right Point-and-Shoot Camera	finding the right point and shoot camera	Clickbait	Non-clickbait
708	Jennifer Lawrence And Amy Schumer Add Chris Pratt And Aziz Ansari To Their Squad	jennifer lawrence and amy schumer add chris pratt and aziz ansari to their squad	Non-clickbait	Clickbait
712	Those Who Lost Savings Find Little Comfort	those who lost savings find little comfort	Clickbait	Non-clickbait
727	Outlander Just Made A Big Casting Announcement	outlander just made a big casting announcement	Non-clickbait	Clickbait
746	Vin Diesel Shut Down Body Shamers With A Single Instagram	vin diesel shut down body shamers with a single instagram	Non-clickbait	Clickbait
775	Baby Eats Bacon For The First Time, Reaches Peak Satisfaction	baby eats bacon for the first time reaches peak satisfaction	Non-clickbait	Clickbait
790	This Man Is Giving His Software Away For Free To Help People With Disabilities	this man is giving his software away for free to help people with disabilities	Non-clickbait	Clickbait
804	Kanye West Has Revealed His Album Title Is "The Life Of Pablo"	kanye west has revealed his album title is the life of pablo	Non-clickbait	Clickbait

Table 6: Example Errors

6.2 Discussion

6.2.1 Confusion Matrix

We see that the confusion matrix shows results that are quite reasonable with a 1.5% false negative rate (97/6400) and a 2% false positive rate (130/6400).

6.2.2 Model Accuracy and Loss

Running the model on the test data shows an improvement on model accuracy that starts from approximately 0.94 at the first epoch, till it reaches 0.96 on the fifth epoch. As can be expected, the accuracy on training starts below 0.88 but approaches 0.97 by the fifth epoch Conversely, the loss on training data starts above 0.25 and converges to below 0.05 by the fifth epoch; and on

test data, it starts at 0.15 and flattens to 0.10 by the fifth epoch.

6.2.3 Classification Report

All the measures for the classification report precision, recall and f1-score show good performance of the classifier with all these measures above or equal to 0.96.

6.2.4 Example Errors

The error examples reveal some deficiencies in the model. Certain words and acronyms that appear in the test set are not available in the training set, so these are missing in the pre-processed data, thus confusing the classifier. For example, the samples with ID 10 ("Timesman"), 196 ("Mouthgasm"), 308 ("Extroverted"), 345 ("Mandingo"), 460 ("Helpline", "pianist"), 694 ("HDTV"), provide examples of these types of errors. Absence of these key words and acronyms in the reverted text cause the classifier to misclassify these samples as non-clickbait when in fact they are clickbait. Other error sources could include words and punctuations that are typically used in clickbait (examples: ID 81 ("knows"), 256 ("strange"), 302 (","), 362 ("tricky")), thus causing the model to misclassify these sample as clickbait when in fact they aren't.

7 Next Steps

For the rest of the semester, our team will focus on building clickbait classification at the content level engine using text summarization and textual similarity algorithms. We are developing a Reddit scraper and parser for the two subreddits /r/saveyouaclick and /r/news. This will allow us to collect a curated dataset of clickbait vs. non-clickbait at the content-level, with /r/saveyouaclick for clickbait articles and /r/news for non-clickbait articles. Next, we will use the research presented in See (2017) as our basis for building a text summarization engine with a pointer-generator network algorithm. Last, we will classify clickbait by comparing generated text summary with the actual article titles.

References

Anand, A., Chakraborty, T., and Park, N 2016, December 05. *We used Neural Networks to Detect Clickbaits: You won't believe what happened Next!*. https://arxiv.org/abs/1612.01340.

Biyani, P., Tsioutsiouliklis, K., and Blackmer, J. 2016, February 21. *"8 Amazing Secrets for Getting More Clicks": Detecting Clickbaits in News Streams Using Article Informality*. https://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/1180
C.

Chakraborty, A., Paranjape, B., Kakarla, S. and N. Ganguly. 2016. *Stop Clickbait: Detecting and preventing clickbaits in online news media*, IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), San Francisco, CA, 2016, pp. 9-16. http://ieeexplore.ieee.org/document/7752207/?reload=true.

Chopra, S., Auli, M., and Rush, A. 2016. *Abstractive sentence summarization with attentive recurrent neural networks*. In North American Chapter of the Association for Computational Linguistics.

Kupiec, J., Pedersen, J., and Chen, F. 1995. *A trainable document summarizer*. In International ACM SIGIR conference on Research and development in information retrievale.

Nallapati, R., Zhai, F.,and Zhou, B. 2017. *SummaRuNNer: A recurrent neural network based sequence model for extractive summarization of documents*. In Association for the Advancement of Artificial Intelligence.

Nallapati, R., Zhou, B., Santos, CÂÿand Xiang, B. 2016. *Abstractive text summarization using sequence-to-sequence RNNs and beyond*. https://arxiv.org/abs/1602.06023e.

Saggion, H., and Poibeau, T. 2013. *Automatic text summarization: Past, present and future*. In Multi-source, Multilingual Information Extraction and Summarization, Springer, pages 3âĂŞ21.

See, A., Liu, P., and Manning, P. 2017. *Get To The Point: Summarization with Pointer-Generator Networks*. https://arxiv.org/pdf/1704.04368v2.pdf.

Yin, W., Kann, K., Yu, M., and Schutze, H. 2017. *Comparative Study of CNN and RNN for Natural Language Processing*. https://arxiv.org/pdf/1702.01923.pdf.