Sadia Karim 23386320

FIT5212 ASSIGNMENT 1 DISCUSSION REPORT

It is clear from the results in part 1 that the given dataset was of excellent quality for text classification with great representations and an even balance of both classes. This made it very easy to for statistical models to learn and produce very accurate predictions. The dataset was not a great fit for topic modelling, and this may be due to the length of the texts, the fact that many terms were shared amongst the different topics which made it difficult to discern exactly what the individual topics are, or different types of language patterns used which led to one topic being dominant over all others.

PART 1: TEXT CLASSIFICATION

The 3 algorithms I compared were:

- Logistic Regression (LogRegCV)
- Random Forest Classifier (RFC)
- Recurrent Neural Network (RNN)

From the results, it was clear to see that out of the 3 algorithms, the LogRegCV with stopwords on the full dataset performed the best on the test set with a precision, recall, and F1-score of 0.99

TEXT PREPROCESSING: STOPWORDS VS NO STOPWORDS

The two text pre-processing configurations I explored was keeping stopwords vs. removing stopwords.

I focused on the statistical models first and tried to find the best preprocessing technique to increase their performance. I performed all the standard text preprocessing steps to get a baseline performance of the algorithms on the data using the following baseline configuration:

- SpaCy Lemmatizer
- TF-IDF vectorizer
- Removing stopwords
- Converting to lowercase
- Unigrams

Using this baseline performance, I was able to see the effect of each parameter and select accordingly.

There were several variations that I explored when trying to find the best pre-processing configurations:

- Spacy Lemmatizer vs. WordNet Lemmatizer
- TfidfVectorizer (TV) vs. CountVectorizer (CV)
- Removing stopwords vs. Keeping stopwords
- Lowercase vs. Original case
- ngram(1,1) vs. ngram(1,2)

The best combination for all three algorithms was given by:

- WordNet Lemmatizer
- Keeping stopwords
- TfidfVectorizer (TV)

- Original text case
- ngram(1,2)

The SpaCy Lemmatizer performed slightly worse than the WordNet Lemmatizer (WNL) in all my test cases with the baseline configuration. This was surprising since in all my research, SpaCy is touted as the industry standard for tokenization and is supposed to perform faster. WNL performed ~1% better in comparison on the full dataset (99% WNL vs. 98% SpaCy), and ~5% better on the smaller dataset (98% WNL vs. 93% SpaCy).

I initially expected TV to perform better than CV as it preserves the importance of words, which I believe would be important when looking at clickbait headers since usually the language used between the classes would be different - clickbait text may be more extreme and dramatic, compared to non-clickbait. However, with my baseline configuration, CV outperformed TV by almost 5% every time with both datasets. I found this very interesting, and it led me to believe that one of the other text features like stopwords, or original text case was probably important to the different classes. I then did some research to try and understand in which case CV would outperform TV, it appears that there are some cases where TF-IDF may hurt accuracy instead of improving it:

1. *"When there is class imbalance. If you have more instances in one class, the good word features of the frequent class risk having lower IDF, thus their best features will have a lower weight"*
2. *"When you have words with high frequency that are very predictive of one of the classes (words found in most documents of that class)"* [1]

I believe there was strong evidence of point 2 in my dataset – when keeping the original text case with stopwords, there was a very clear difference in the use of language between the two classes. I confirmed this by first keeping the stopwords and seeing an improvement.

I then looked at the text case. The models performed at around 95% accuracy when the text was lower-cased, but when I compared it against the original text case in the data, the models performed significantly better. The reason was very clear when I did a quick search (cells can be found in section 5 of the code) on a few common stopwords with whitespace on either side, indicating that I am looking for the term in the middle of a sentence, and on punctuation terms. The results were very telling:

Table 1: "Stopword in the middle of sentences and punctuation" exploration

	Clickbait	Non-clickbait
" The "	2419	30
" Of "	1467	1
" To "	1915	4
" In "	1118	9
" the "	4	744
" of "	9	1599
" to "	6	2033
" in "	2	2673

[1] Source: https://stackoverflow.com/a/39413780

%	34	118
!	21	8
‘	2361	735
#	36	0

Title case in the middle of sentences is significantly higher in clickbait articles, than in news articles, where stopwords tend to be lowercase, and punctuation is also important in both classes.
It’s clear that title case and stop words are two very strong indicators of class, and all this valuable information is lost if the text is converted to lower case or stopwords removed.

Keeping stopwords, punctuation and text case improved the results of the TFIDF vectorizer, to the point where it outperformed the Count vectorizer as I initially expected.

I further improved the models with the introduction of unigrams and bigrams. Since text case was such a strong indicator of class, I found there to be a big improvement in accuracy when introducing bigrams, which included combinations of lower and title case.

MODEL RESULTS & DATA SIZE

The 3 algorithms had quite a range of performance on the smaller dataset:

Table 2: Small dataset Results

SMALL DATASET	RESULTS
LogReg + Stopwords	Precision: 0.9779 Recall: 0.9780 F1-Score: 0.9779 ----- Confusion Matrix: [[2450 51] [62 2557]]
RFC + Stopwords	Precision: 0.9371 Recall: 0.9362 F1-Score: 0.9353 ----- Confusion Matrix: [[2430 71] [260 2359]]
RNN + Stopwords	Precision: 0.9283 Recall: 0.9285 F1-Score: 0.9283 ----- Confusion Matrix: [[2342 159] [208 2411]]

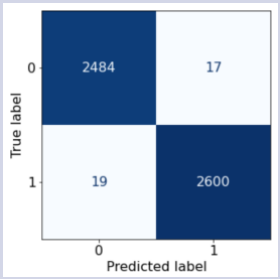
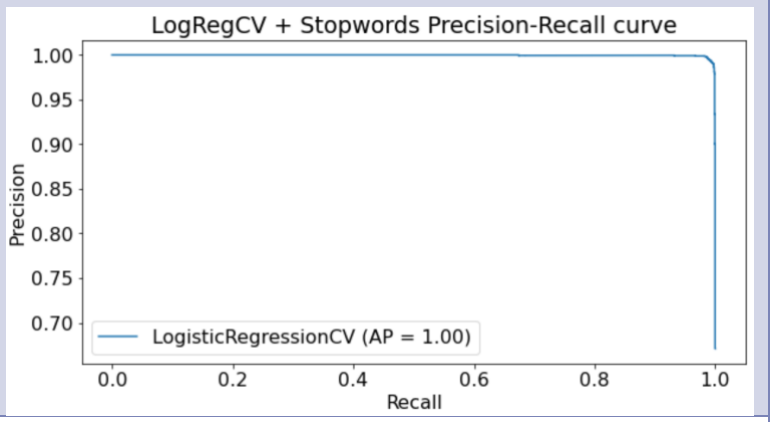
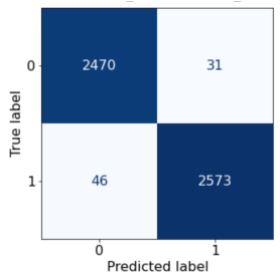
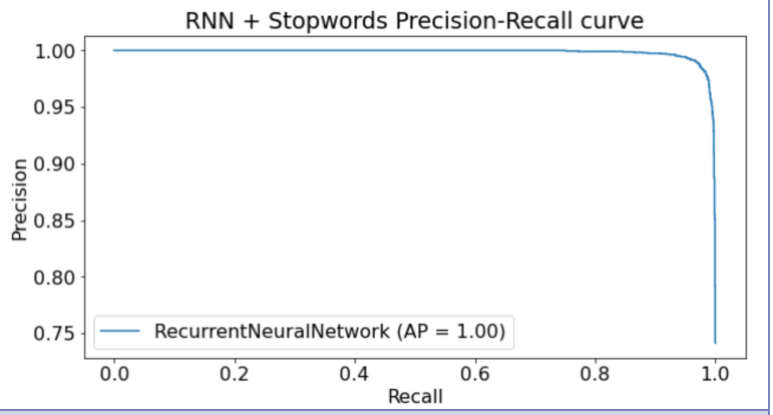
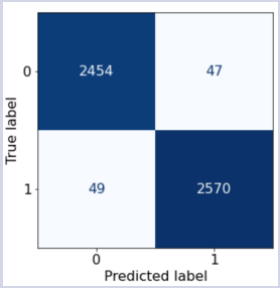
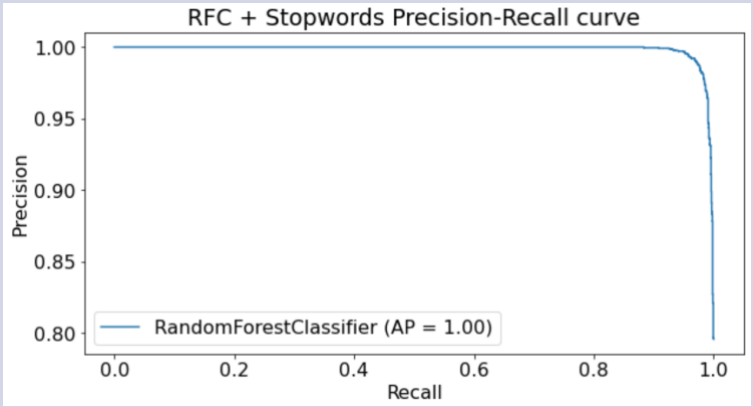
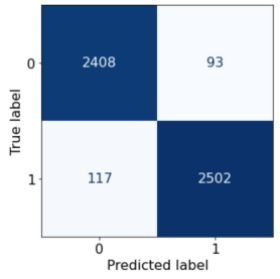
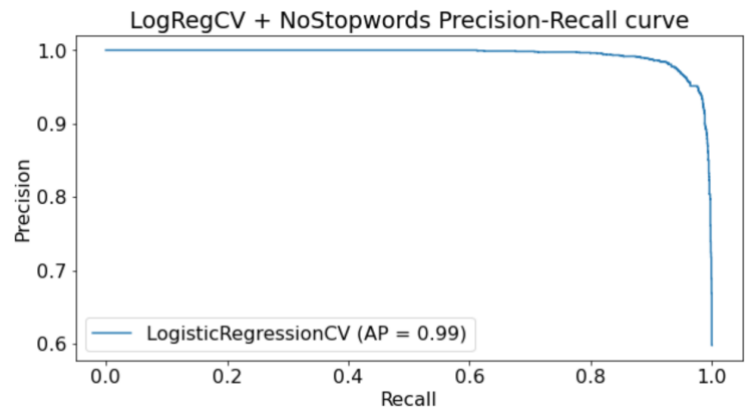
LogReg + NoStopwords	Precision: 0.8712 Recall: 0.8686 F1-Score: 0.8672 ----- Confusion Matrix: [[2302 199] [480 2139]]
RFC + NoStopwords	Precision: 0.8330 Recall: 0.8197 F1-Score: 0.8157 ----- Confusion Matrix: [[2320 181] [755 1864]]
RNN + NoStopwords	Precision: 0.7919 Recall: 0.7519 F1-Score: 0.7396 ----- Confusion Matrix: [[2359 142] [1151 1468]]

The dataset size was very important for the algorithms to produce accurate results. Although LogReg performed the best, it didn't come close to its performance on the full dataset (99%+). There was also a big difference in the stopwords vs. no stopwords datasets. All the algorithms performed better on the stopwords dataset which was to be expected, and they had similar precision and recall values, meaning that the algorithm returns the relevant results most of the time. In the "NoStopwords" results, we can see that the precision tends to be slightly higher than recall, meaning that the model is missing true positives possibly due to having trouble identifying the indicators from such a small dataset. This makes sense as stopwords were identified as being key indicators of the classes.

All the models tended to have higher rates of False Negatives especially in the NoStopwords dataset. RNN also performed significantly worse out of all 3. This could be because LogReg and RFC have a more simple method of identifying the indicators of a positive label, and were able to identify positive cases easier than RNN. The fact that the RNN had a much higher rate of False Negatives shows that its learning method was more complicated and its learned definition of "True Positive" was too restrictive, and so didn't have enough data to learn what a True Negative looks like.

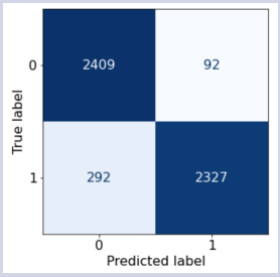
We see a much bigger improvement in all 3 algorithms with the larger dataset where RNN performs just as well as LogReg and even better than RFC on the stopwords dataset, indicating that datasize is very important for an RNN, but LogReg and RFC can get away with smaller datasets and still produce very accurate results, provided that the input data is of good quality.

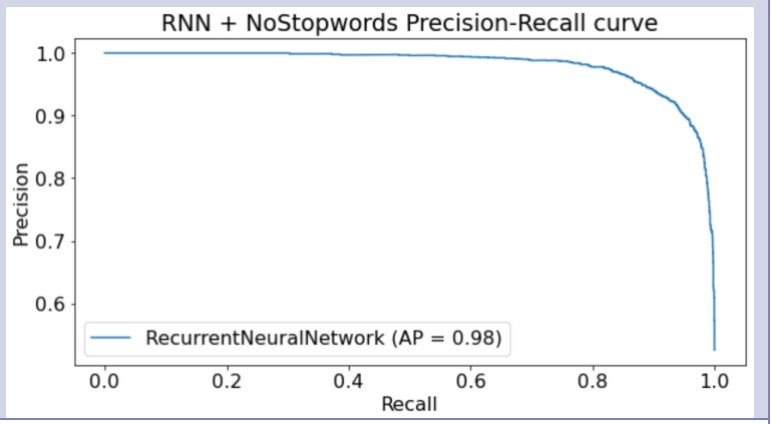
Full Dataset Results Ranked from Best to Worst

	Confusion Matrix	Precision-Recall Graph
LogReg + Stopwords Precision: 0.9930 Recall: 0.9930 F1-Score: 0.9930		
RNN + Stopwords Precision: 0.9849 Recall: 0.9850 F1-Score: 0.9850		
RFC + Stopwords Precision: 0.9812 Recall: 0.9812 F1-Score: 0.9812		
LogReg + NoStopwords Precision: 0.9589 Recall: 0.9591 F1-Score: 0.9590		

RNN + NoStopwords

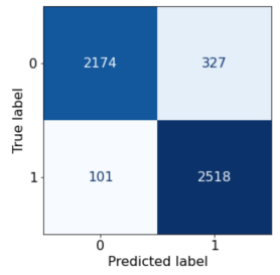
```
Precision: 0.9269
Recall: 0.9259
F1-Score: 0.9250
```

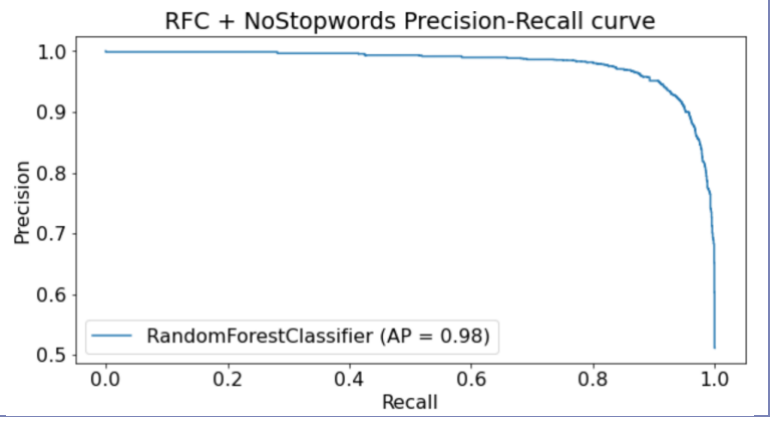




RFC + NoStopwords

```
Precision: 0.9203
Recall: 0.9153
F1-Score: 0.9160
```





All three algorithms performed significantly better **with Stopwords** on the **full dataset**. LogReg performed the best out of all 3 algorithms, with RNN and RFC performing almost identically on both datasets, however RFC and RNN performed worse on the NoStopwords full dataset, than the Stopwords small dataset, showing that the quality of the input data is just as important as the quantity.

If the data does not have enough identifying features, then the model will require a lot more data to learn the class features more accurately. RFC also had higher rates of False Positives than the other models in the NoStopwords dataset, indicating that some features have been incorrectly learned as positive during training.

All this aside, the F1 scores of all the models are relatively high, meaning that precision and recall are both high, so the models are still great at predicting. The Precision-Recall curves of all models have very high average precision, indicating that the models still perform very well at different recall thresholds.

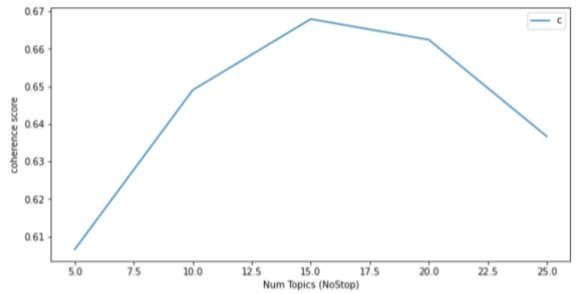
The overall output of this task has shown the importance of having good class-specific features when training text classification models. In our case it was keeping stopwords that made a significant difference to our models. When we removed stopwords, we can see that the models struggled to identify the classes correctly and started learning and attributing incorrect features to the positive class which resulted in high rates of False Negatives.

It was interesting to see that LogReg, being the simplest of algorithms out of all 3, performed the best. It also performed the best on the Kaggle dataset, producing an accuracy of 0.9923. This goes to show that binary classification on this specific dataset was probably too simple for more complex algorithms like RFC and RNN, which are generally used on much bigger datasets, or multi-label classification.

PART 2: TOPIC MODELLING

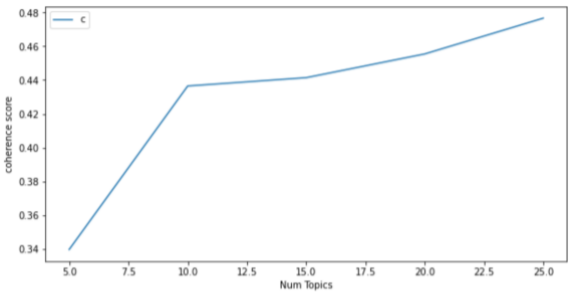
I used the same text preprocessing technique as part 1: Stopwords vs. NoStopwords. Because stopwords were so important when classifying the text, I was interested to see the effect they had in topic modelling. Prior to exploring, I tried to hypothesise what the result would look like, and I suspected that the dataset **with stopwords** would be full of noise and the true topics would not be very clear. Due to this reason, I focused on maximising the results from the **nostopwords** dataset.

I modelled out the coherence scores (measuring the degree of semantic similarity between high scoring words in the topic) of a range of topic numbers and found that K=25 had the highest coherence value of 0.4767 for data with stopwords but all values were less than the coherence values of nostopwords, K=15 had the highest coherence value of **0.6679 for NoStopwords**. For this reason, I have chosen to explore K=5 and K=15 in Part 2.




```
Num Topics = 5  has Coherence Value of 0.6066
Num Topics = 10  has Coherence Value of 0.649
Num Topics = 15  has Coherence Value of 0.6679
Num Topics = 20  has Coherence Value of 0.6624
Num Topics = 25  has Coherence Value of 0.6367
```

Figure 1: NoStopwords "coherence vs. num topics"




```
Num Topics (stop) = 5  has Coherence Value of 0.3395
Num Topics (stop) = 10  has Coherence Value of 0.4365
Num Topics (stop) = 15  has Coherence Value of 0.4414
Num Topics (stop) = 20  has Coherence Value of 0.4555
Num Topics (stop) = 25  has Coherence Value of 0.4767
```

Figure 2: Stopwords "coherence vs. num topics"

I also plotted the word frequencies of stopwords and nonstopwords to determine the parameters for my dictionaries – trying to remove overly frequent words and very uncommon words.

I ran LDA over the datasets with K=5 and K=15, also determined the coherence score and perplexity of each model. The following are the wordclouds of the top 30 terms in each model, and the keywords and most representative document of each topic:

K=5, No Stopwords

Topic	Top 10 Keywords	Most Representative Doc
Topic 0	christmas, day, australian, favorite, china, game, pictures, south, test, girls	Justin Bieber Called Bette Midler "Britt Meddler" And Hopefully This Is The Start Of An Amazing Feud
Topic 1	u.s, based, life, real, british, ways, women, photos, zodiac, instagram	Which Donald Trump Quote Are You Based On Your Zodiac Sign

Topic 2	time, reasons, thing, halloween, man, kids, american, north, video, totally	13 Reasons Why Taylor Swift's 1989 World Tour Was The Best Part Of 2015
Topic 3	people, love, star, tweets, guess, movie, harry, india, disney, house	I Saw "The Force Awakens" Without Seeing Any Other "Star Wars" Movie And This Is What Happened
Topic 4	things, times, obama, perfect, president, year, understand, canadian, city, indian	Car Bomb Kills Police Official in Spain

K=15, No Stopwords

Topic	Top 10 Keywords	Most Representative Doc
Topic 0	based, favorite, zodiac, guess, sign, based_zodiac, fans, characters, white, justin	Can We Guess Your Favorite Disney Movie Based On Your Zodiac
Topic 1	south, man, house, pakistan, internet, court, england, relationship, awkward, supreme	Recriminations and Regrets Follow Suicide of South Korean
Topic 2	people, times, british, year, week, united, california, confessions, states, bank	United States anti-drug efforts in Latin America criticized by WOLA report
Topic 3	u.s, instagram, canadian, australia, state, lyrics, taylor, case, easy, swift	46 Taylor Swift Lyrics For When You Need An Instagram Caption
Topic 4	totally, remember, guy, good, recipes, girl, chief, adorable, afghanistan, perfectly	Which Hogwarts Houses Do These "Gossip Girl" Characters Belong In

Topic 5	real, tweets, pictures, girls, sex, laugh, literally, guaranteed, fan, funny	27 Tweets About Donald Trump That'll Actually Make You Laugh
Topic 6	women, halloween, test, london, movie, west, school, high, chinese, delicious	I Went To Lauren Conrad's Fashion Show And Lived Out My High School Fantasy
Topic 7	life, reasons, hilarious, watch, disney, food, woman, makeup, gifts, wedding	For Everyone Who Became Completely Obsessed With Kendall Jenner In 2015
Topic 8	day, harry, york, potter, harry_potter, american, france, hair, moments, florida	Here's How Much Fun I Had On The "Harry Potter" Studio Tour
Topic 9	star, perfect, wars, star_wars, character, questions, party, fall, worst, person	75 Thoughts I Had While Watching "Star Wars: The Force Awakens"
Topic 10	things, time, australian, india, video, happened, wikinews, work, music, happen	Fremantle defeat Sydney, qualify for 2013 Australian Football League Grand Final
Topic 11	love, movies, live, iran, feel, canada, facebook, michael, plan, weird	A.M.A. Opposes Government-Sponsored Health Plan
Topic 12	obama, kids, friends, family, north, bush, texas, iconic, cast, song	Former US Vice President Dick Cheney: 'Barack Obama is a one-term President'
Topic 13	ways, photos, iraq, understand, things, indian, big, loss, single, true	15 Indian Beauty Secrets The Whole World Should Know
Topic 14	christmas, thing, china, game, city, president, years, black, men, dog	This Dog Begging For His Human's Forgiveness Is The Cutest Thing You'll See Today

K=5, Stopwords

Topic	Top 10 Keywords	Most Representative Doc
Topic 0	you, your, are, what, how, which, can, know, should, these	We Know Your Favorite "Game Of Thrones" Character Based On Your Favorite "Harry Potter" Character
Topic 1	the, this, and, from, with, u.s, times, was, for, their	A 3-Year-Old Girl With Cancer Sang "Let It Go" From Her Bed With Some Help From Her Nurses
Topic 2	that, will, who, things, for, people, about, make, are, that_will	21 Pictures That Are Way Too Real For People Who Are Always Cold
Topic 3	new, here, like, most, need, first, time, what, the_most, ways	Taylor Swift Donated 25,000 Books To NYC Schools In Need
Topic 4	world, has, british, over, women, perfect, try, india, south, its	E.U. Trade Deal Is Close, South Korean Leader Says

K=15, Stopwords

Topic	Top 10 Keywords	Most Representative Doc
Topic 0	for, actually, reasons, get, christmas, love, have, pictures, why, and	21 Terrifyingly Perfect Gifts For Shark-Loving Kids

Topic 1	times, real, way, too, were, life, that, pakistan, family, made	27 "Real Life" Magazine Headlines That Went Way, Way Too Far
Topic 2	british, obama, game, says, canadian, u.s, court, france, former, state	Former Alabama Governor Loses Final Round in Federal Appellate Court
Topic 3	who, people, u.s, their, for, people_who, not, man, date, case	62 People Who Dressed As Matt Bellassai For Halloween
Topic 4	the, about, this, and, one, most, his, out, has, the_most	17 Of The Most Iconic "American Idol" Performances Of All Time
Topic 5	and, her, over, ways, this, was, tell, into, she, west	A Daughter Is Turning To Social Media After She Realized Her Mom Was Lonely
Topic 6	china, harry, potter, harry_potter, its, friends, back, chief, old, during	Publication date for last Harry Potter book announced
Topic 7	with, things, women, try, only, all, south, understand, will, than	27 Things Everyone Obsessed With Ikea Will Understand
Topic 8	that, will, you, make, your, that_will, every, really, things, london	15 Crucial Holiday Storage Hacks That Will Make Your Life Easier
Topic 9	what, this, you, here, like, when, for, would, look, did	Here's What Top Professional Models Look Like Without Makeup
Topic 10	new, how, world, you, well, new_york, york, india, how_well, two	2007 Rugby World Cup: New Zealand, Australia and Ireland win
Topic 11	your, you, know, based, need, which, based_your, should, what, favorite	We Know Your Favorite Disney Channel Original Movie Based On Your Zodiac Sign
Topic 12	are, you, that, more, and, just, want, after, photos, halloween	7 Easy Fall Dinners That Are Actually Satisfying
Topic 13	you, can, these, which, should, are, star, guess, wars, disney	Can You Guess These Classic "Star Wars" Creatures

Topic 14	from, the, best, first, the_best, day, time, movie, california, may	Bacon Lovers Meet Tiny Piglets For The First Time

First looking at the stopwords data, I can see that stopwords are a hindrance when it comes to topic modelling. They create a lot of noise in the data and don't allow the important themes to emerge. There are no useful terms identified at all, in both K=5 and K=15, and when looking at the LDA visualisation in the notebook, I can see that none of the clusters are overlapping indicating that they are all very different and individual clusters, but the keywords that are identified in the topics actually share majority of the terms so I am curious to know how the model has identified these documents as part of the given cluster.

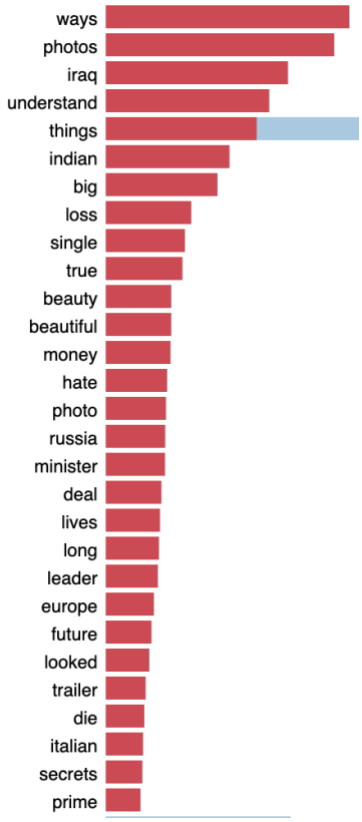
That said, I think that the outputs of the K=5 cluster actually make more sense than K=15, because even though K=15 has a higher coherence value (0.4623 vs. 0.4061), K=5 has a slightly lower perplexity – meaning the model was slightly less confused about how to cluster the documents overall. Looking at the keywords identified in each topic, this makes sense. There is very little overlap between the keywords so the model would have an easier time allocating the documents to the topics, however these topics are nonsense and although it may be easier for the model, the outputs are not useful for human interpretation.

This can easily be seen when looking at the topmost representative document for each topic in both K=5 and K=15 clusters – there is no tie between the keywords and the document. It would be impossible for a human to be able to allocate the documents to the topics given only the keywords. Therefore, stopwords must be removed from the dataset when performing topic modelling in order to allow the most important terms to surface and be visible to the model for learning.

Looking to the nostopwords dataset, there are some much better topics and keywords identified. We can see in K=5, there is a very strong topic about politics (topic 4), and all other topics are quite heavily geared towards pop culture as can be seen by the representative document. In the LDA visualisation there is an almost complete overlap between topics 0 and 4 which when looking at the terms, there is clearly some mix of political topics in topic 0, even if the top 10 keywords don't obviously indicate as such. All the other topics seem to be about popular culture, even if there is some allusion to politics (eg. Which Donald Trump Quote Are You Based On Your Zodiac Sign), the model has still identified this as being more pop culture than politics, which I would agree with.

Although the terms in this dataset are much clearer, there are again some terms which appear "noisy" such as day, big, things, red etc. which don't make much sense as topic terms, but again this could be due to the fact that the model is trying to force these documents into a smaller number of topics than is ideal, so it is looking for as many semantically similar terms as it can which may result in some seemingly unimportant words being identified as being significant.

Looking at the K=15 topics, this model has performed the best and the topics identified seem to be much more coherent and understandable, but even then there is a very strong leaning towards topics of pop culture than any other possible topics. There are no clear topics identified, and most of them seem to be a mix of terms. For example, I cannot clearly identify any topic that is explicitly about news, health or sports. I also am unable to determine what Topic 2 is talking about with its key terms "people, times, british, year, week, united, california, confessions, states, bank". When looking at the LDA visualisation in the notebook, I can see that the political news has been buried lower within the cluster terms. For example, in topic 14:



I can see that there are terms that are clearly news or politics related but they are so far down the relevancy list that they are unable to play any real significance to the cluster or even stand as their own cluster. It appears that most of the clusters are determined by clickbait articles, and all other documents are arranged around them in the model.

In trying to understand why the results have appeared this way, I have looked deeper into the dataset itself and think that the reason the model has learned this way is because clickbait articles tend to use similar types of language – numbers (eg. 27 things, 65 reasons, etc.), positive verbs (love, feel, etc.), plural words, and similar topics (eg. Harry Potter, Star Wars, Taylor Swift, etc.). It is easy to see how these words would be semantically similar and would make the bulk of the topic terms, and the model identifies these pop culture terms extremely well.

It is much harder to identify these sorts of patterns and characteristics in news articles which tend to use less of these types of words, and more country names, leadership position names, and the topics tend to be quite different – the topics and terms tend to be quite different based on the year, or event. It would be harder to connect such terms, at least on this current dataset.

Looking at the data, I can also see how difficult it would be for a model to identify a sports topics given that the following are examples of sports:

“FIFA announce Russia to host 2018 World Cup, Qatar to host 2022 World Cup”

“Lewis Hamilton wins 2008 Australian Grand Prix”

All of these are examples of sport but “Grand Prix” is only mentioned 33 times in the entire dataset, and there is no indication that it is a sport. Similarly, “World Cup” has no indication that this is sports related and only appears in the dataset 38 times.

There are also many pop culture articles about politics – eg. Which Donald Trump Quote Are You Based On Your Zodiac Sign, 27 Tweets About Donald Trump That'll Actually Make You Laugh

It is easy to see how these terms get lost within the more dominant topic terms of “zodiac sign”, and less about Donald Trump, when there are more documents talking about tweeting and zodiac signs with Donald Trump in them, than there are news articles. News articles also don’t tend to use similar language consistently like articles about pop culture do, which is why I believe the model identifies these topics easier.

I think the model has extracted as many topics and terms as it can based on the given dataset. I don’t think the dataset is a good one for topic modelling – there aren’t enough clear-cut terms used for the different topics that exist in the data, and the text is probably not enough for the data to learn from. All of the topics identified appear to be pop-culture heavy, and I think this is due to the fact that clickbait articles, which tend to be more pop-culture related, tend to use similar language that easily form patterns and relations with other terms, than other topic articles which uses different types of language that is not as easily identifiable and not common enough for the model to identify it separately.