phase 4 project

March 22, 2022

By James Pheby

Text classification model for the Guardian newspaper to automatically archive articles

Summary

The Guardian newspaper is looking to implement a system to automatically archive articles into their correct subject categories, saving journalists having to manually enter metadata when moving urgent copy

To do this, they have asked us to assess the viability of producing a model that will take in every news article published to its internal system and immediately tag it with one of its 24 subject category labels

If the model is successful, the newspaper will provide further funding to improve the model and potentially market it to other news outlets

They have also asked us to present them any data picked up during the modelling process that could provide insights into hidden trends and reporting standards

Main recommendations

- The best model predicted the correct classification label almost 80 percent of the time on unseen articles, suggesting further development could produce a commercially viable model

- The key to improved results is providing more storage space and processing power in order to tune the models and develop techniques to separate categories most often mislabelled

Outline

- Business Problem
- Data Collection, Analysis and Modeling
- Results
- Conclusion

Business Problem

- To produce an accurate model that will automate the process of tagging news article with their correct classification label. The model will allow urgent copy to be sent quicker, and could potentially be marketed to other media outlets

- The newspaper is interested in finding any hidden trends in each category and in articles that generate the most social media interest

Data and Methods

Data

- To generate the models, we used a dataset of every article stored in the Guardian's archives since 2013, a total of 735,920. These articles were pre-tagged with their category and were accessed through the Guardian's free-to-use API

- We filtered the dataset to only include articles in the 24 largest categories, and kept only metadata detailing publication date, headline, byline and wordcount along with the article text

Methods

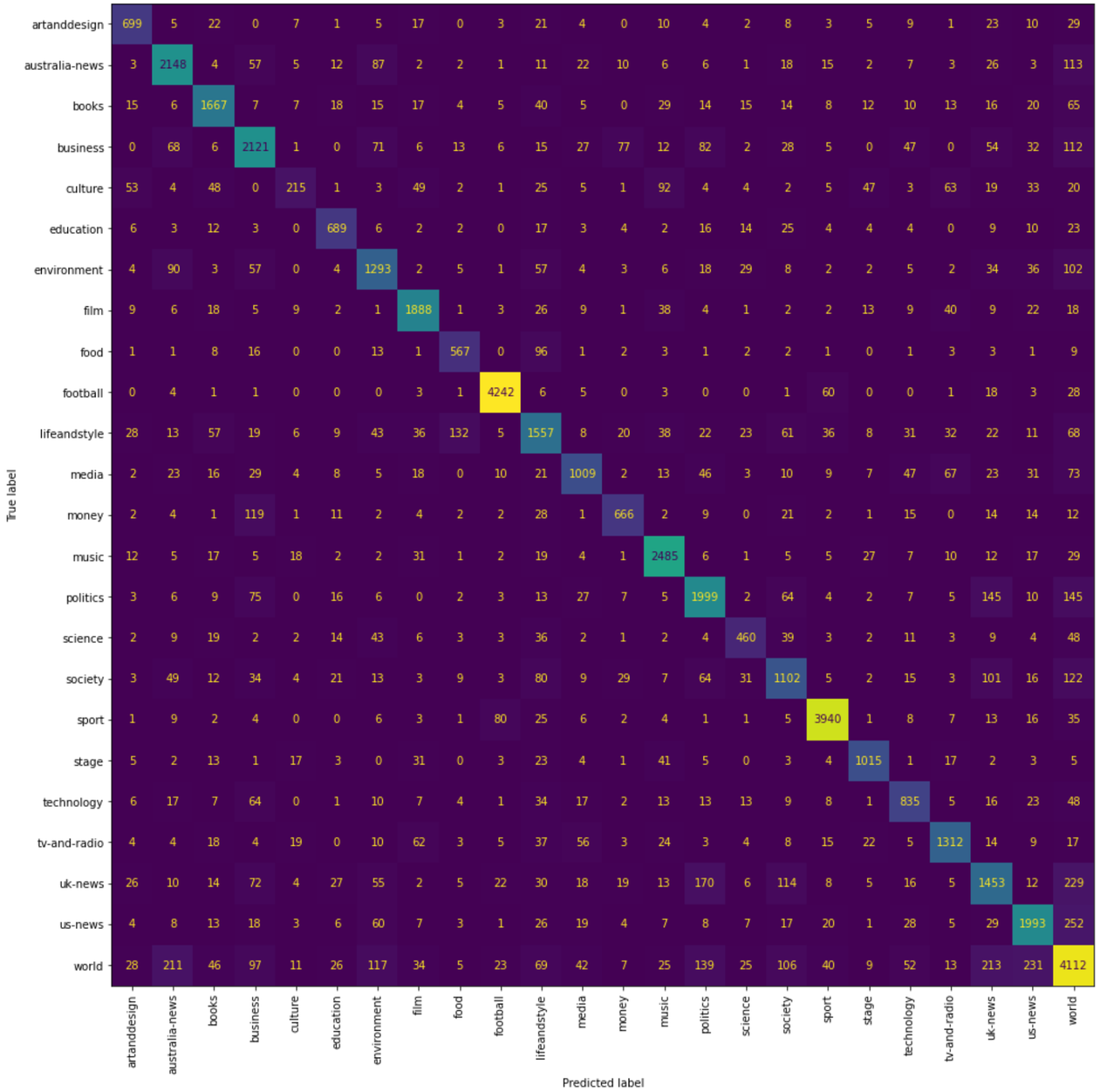
- We trained various models using the text and category label of the articles. We also used different techniques for processing the words and for representing them in the model. Unseen articles were fed into the models, and their performances evaluated and compared

- We analyzed the workings of the best performing model to see which words were most important in deciding which category each article belonged in, potentially revealing hidden insights

Results

	SampleSet	VectorizationType	Model	ProcessedVector	Accuracy (%)
16	FullSet	tf/idf	Logistic	StopStemTuned	78.9
22	FullSet	WordEmbedding	NeuralNet	W2V100d	78
17	FullSet	tf/idf	GradientBoost	StopStemTuned	75.3
23	FullSet	WordEmbedding	NeuralNet	SelfLearning	74.6
18	FullSet	tf/idf	RandomForest	StopStemTuned	73
15	FullSet	tf/idf	Naive-Bayes	StopStemTuned	72.2
7	Sample10k	tf/idf	Logistic	SpecialWords	71.9
21	FullSet	WordEmbedding	GradientBoost	W2V50d	71.9
9	Sample10k	tf/idf	SupportVector	SpecialWords	71.1
19	FullSet	WordEmbedding	RandomForest	W2V50d	70.2
20	FullSet	WordEmbedding	Logistic	W2V50d	69.7
10	Sample10k	tf/idf	GradientBoost	SpecialWords	69
5	Sample10k	tf/idf	Naive-Bayes	SpecialWords	66.9
13	Sample10k	WordEmbedding	Logistic	W2V50d	66
2	Sample10k	tf/idf	Naive-Bayes	Stemmed	65.7
4	Sample10k	tf/idf	Naive-Bayes	TunedMaxFeatures	65.6
3	Sample10k	tf/idf	Naive-Bayes	NumSent	65.5
8	Sample10k	tf/idf	RandomForest	SpecialWords	64.7
1	Sample10k	tf/idf	Naive-Bayes	Stopword	64.7
14	Sample10k	WordEmbedding	GradientBoost	W2V50d	64.1
12	Sample10k	WordEmbedding	SupportVector	W2V50d	63.4
11	Sample10k	WordEmbedding	RandomForest	W2V50d	63.4
0	Sample10k	tf/idf	Naive-Bayes	Baseline	54
6	Sample10k	tf/idf	Naive-Bayes	Bigrams	53

- **The two best performing models, out of the 24 created, predicted the correct labels almost 80 percent of the time**



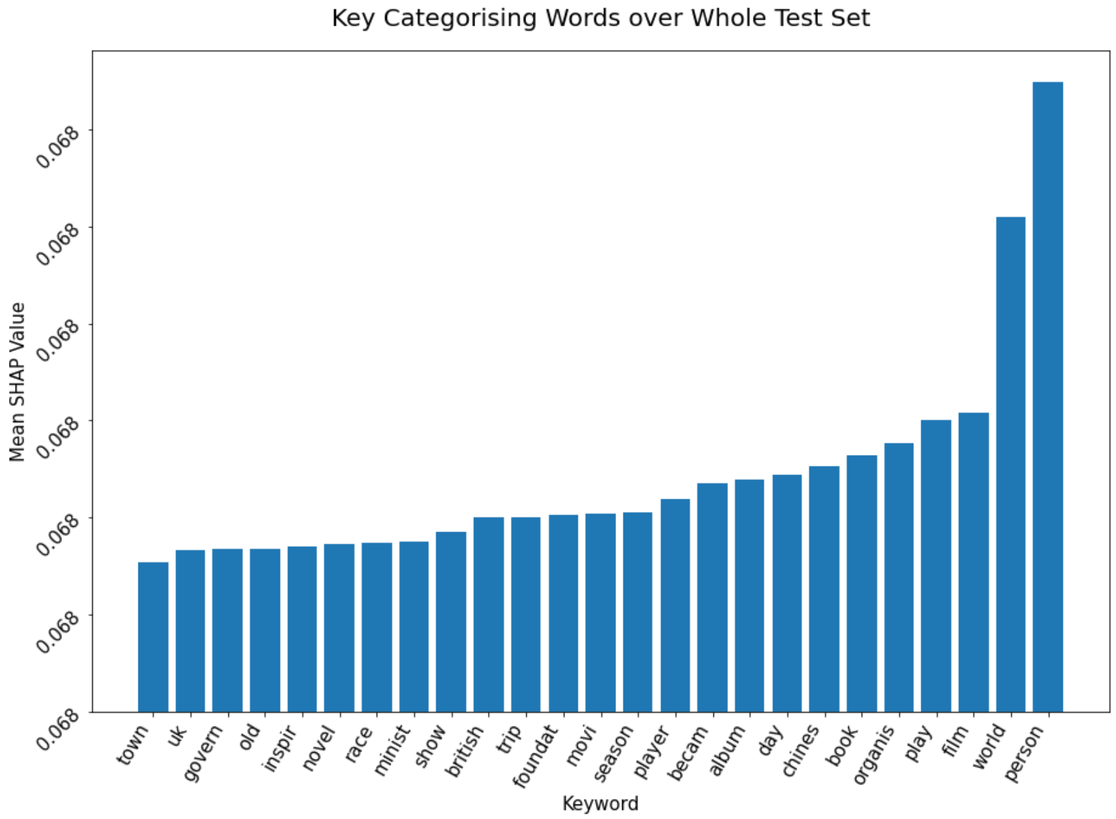
The best-performing model struggled most with stories in the world-news, business and lifestyle categories as these tend to be more loosely defined and overlap with other categories

Data processed during modelling can easily be extracted to give clients visualisations of which words are featured most by category, date or across the whole Guardian corpus

World News WordCloud

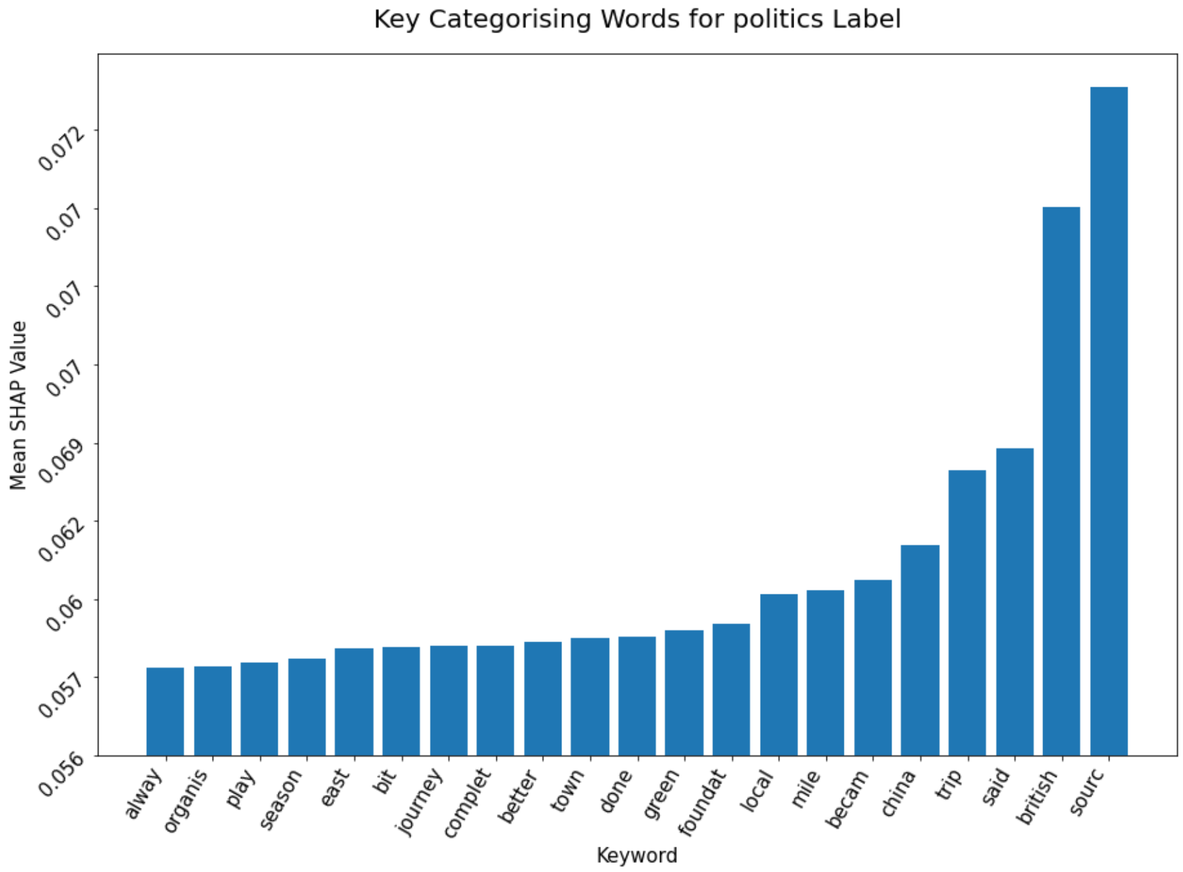
Music Stories WordCloud

- **The model's internal decision making process was analyzed to reveal which words were the most important in predicting category labels, whether across the whole corpus, each category or individual articles**



- **Across the whole dataset, we can see some of the most inportant words in deciding the label include China, film, book, album and player. This is perhaps unsurprising as they tend to be words that occur regularly only in a small number of categories**

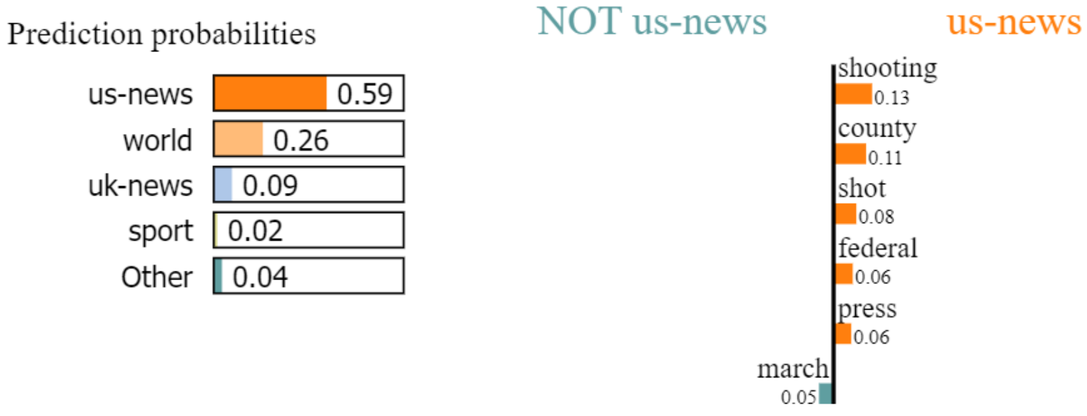
- **More interesting insights can be gleaned by looking on a more granular level**

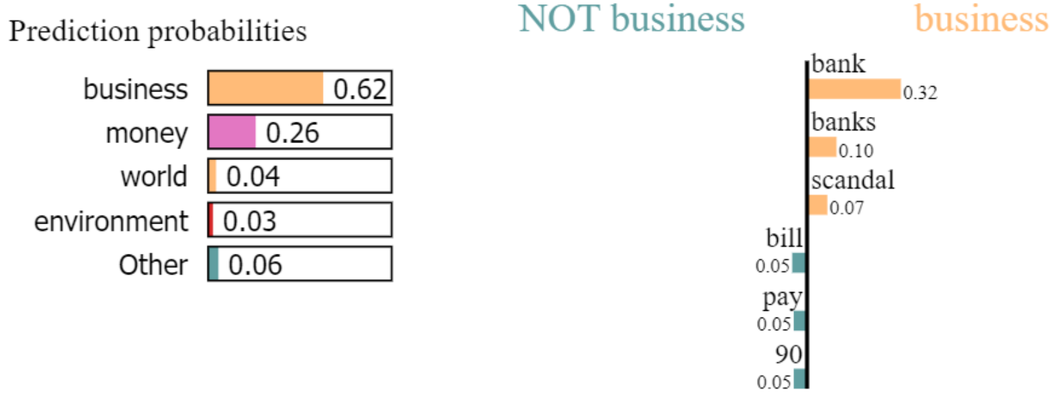


- **For instance, the model reveals the word 'source' as a key indicator of a politics story.**

- **The word is generally used when attributing a quote to an unnamed person, so are we using too many anonymous sources to stand up our politics stories?**

Looking at how the model decides individual articles is also revealing. Here we look at random stories from the US news and business categories



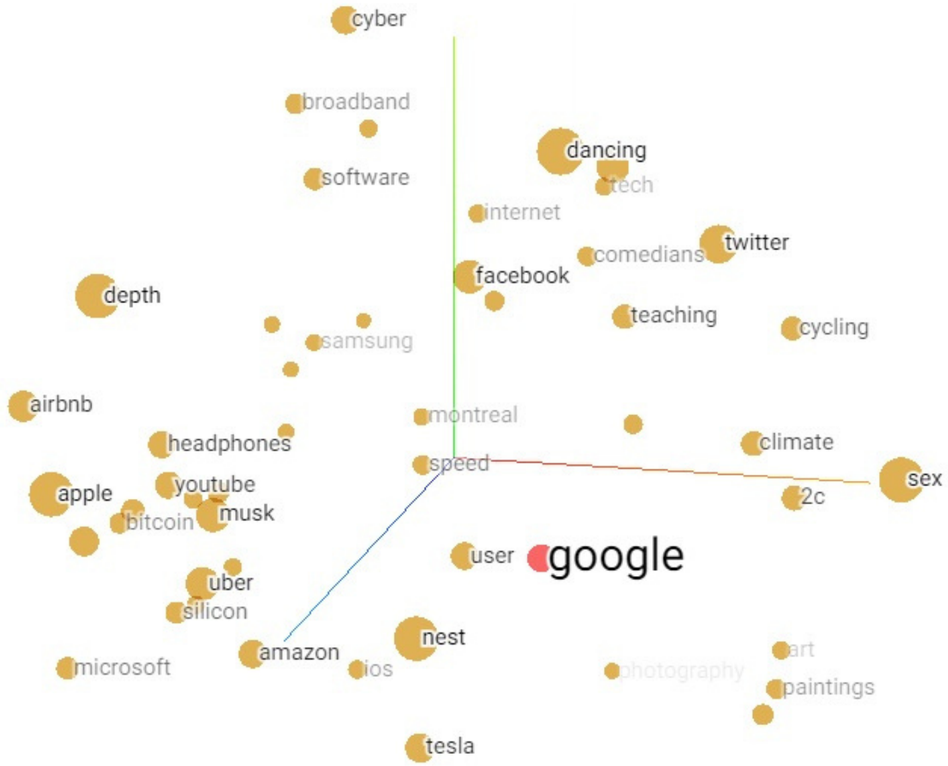


- **The word shot/shooting is key indicator of this US story. While not surprising in this instance, we can easily see how hidden insights will reveal themselves, potentially helping with story ideas**

- **The word scandal is a key indicator in this business story. Again, maybe not an altogether suprising insight!**

The neural network model's working also reveal the similarity of words used in our copy. Again, this can provide insights into the subjects themselves, but could also be compared to other outlets to find any structural differences in vocabularies used across the media

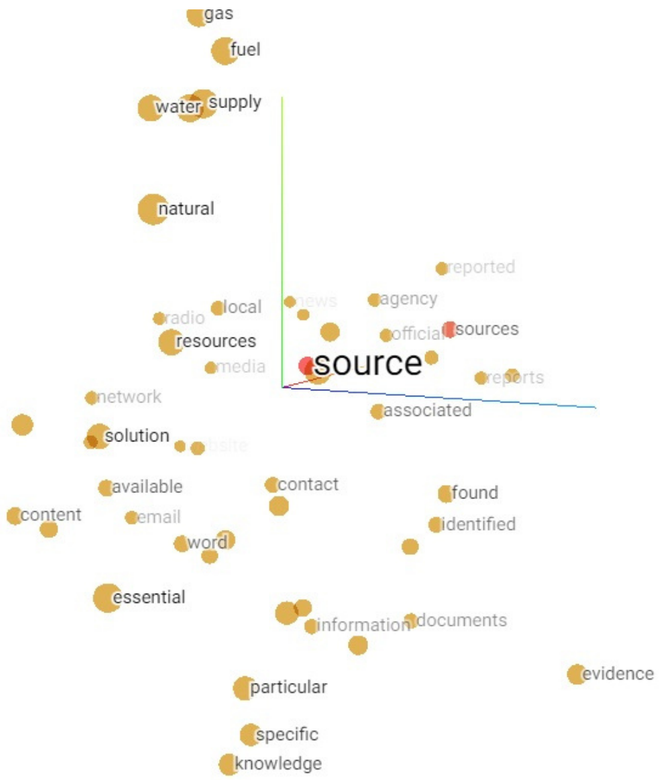
Linguistic position of 'google' as learned on Guardian articles



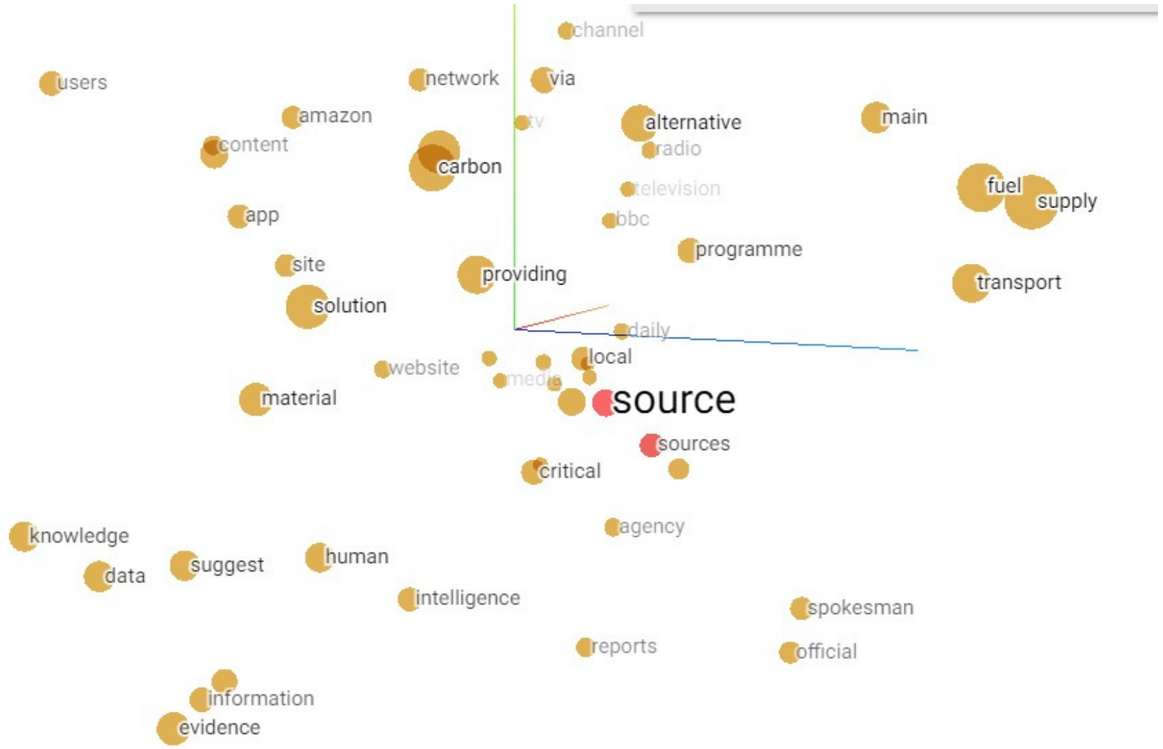
- **Here we use a fairly basic example with the word 'google'. as learned entirely on Guardian articles. The words shown are the 50 most similar learned by the model**
- **We can see how it clusters similar organisations, while other linked words are spread more widely**
- **This is intended as a clear-cut conceptual example, but we can analyze any word, again potentially revealing rich and less obvious insights**

These two plots show linguistic similarities learned over the entire Wikipedia corpus, and the same values then updated by learning on the Guardian articles. Using 'source' in this example, we can then see how Guardian use of words is different from general usage. We could also compare to other media outlets to reveal different editorial practices.

'Source' trained on Wikipedia



'Source' trained on Wikipedia, updated on Guardian



Conclusion

- The model is definitely accurate enough to suggest it could become a viable product, both internally and commercially, with more resources
- It also generates useful insights, potentially revealing hidden threads that run through categories and generating story angles, and also in analysing our own copy for possible biases and adherence to reporting standards

Next Steps

- **Provide funds for more storage space and processing power to enable the models to be better 'tuned' and to allow more advanced processing techniques to help them discern between overlapping categories**

- **Channel more resources into analyzing the workings of the model to find hidden insights. The model can easily be linked to the Twitter API, or to other metadata tags such as byline to glean distinctive features of successful writers or stories that play well on social media**

Contact

Email: jpheby93@gmail.com

GitHub: https://github.com/jimp93

Medium: https://medium.com/@jpheby93/