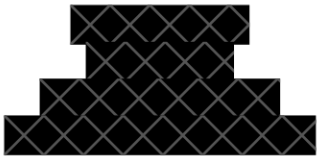
Using Similarity Scoring Techniques to Match Sustainability Topics with News Articles



1 Problem statement and Motivation

Building effective automated tools for tracking developments in sustainability will help navigate challenges related to climate change, inequality, poverty, shortage of resources, and economic development. In 2015, the United Nations member states adopted 17 sustainable development goals (SDG), setting out a 15-year pathway to solving the world's most prescient challenges.

Monitoring global progress on the SDG presents is tough. Although the UN has put in place robust methodology for tracking progress globally for each SDG, it could benefit from additional monitoring tools. Of particular interest are systems which could effectively track streams of global news. According to Czvetko et al. (2021) development of effective SDG monitoring tools able to identify activity in different SDG areas based on news articles could help to inform policy and strategy. Such tools would also help to identify the sustainability areas that are gaining attention and traction amongst the press, and those which are diminishing. Such systems could also act as a more immediate early warning for potential issues in certain areas, given articles can be constantly streamed and analysed, as opposed to the more formalized metrics which take months to collect and corroborate.

This paper will explore the effectiveness of contemporary semantic modelling approaches and propose a system which could automatically categorize news articles to SDG topics. Several tests will be used to assess the effectiveness of a range of semantic representation approaches at identifying alignment between texts. The approaches considered in this research are Word2Vec, SentBERT, Doc2Vec, and DistilBERT (Sent2Vec). The common theme between all these techniques is that they produce vector representation of text. Cosine similarity can be used to measure the similarity between the vector representations and produce similarity scores. Following an initial round of exploration and benchmarking the research will introduce a pipeline to evaluate the proximity of news articles to SDGs segmented into 6 topics.

The paper will utilise three datasets, two Hugging Face datasets and the text of the UNs SDGs. The first Hugging Face dataset holds BBC news articles each labelled by news category. The BBC news article dataset will be used to initially evaluate each of the modelling approach's ability to accurately associate news articles with the correct topics. The other Hugging Face dataset consists of news articles from the publications CNN and the Daily Mail. We'll use this dataset for our final evaluation of each model's ability to accurately associate news articles with the SDGs.

2 Research hypothesis

Given the task of text similarity matching, this paper aims to demonstrate that transformer models focused on text sequences of at least a sentence in length will outperform our baseline of a word2vec implementation.

3 Related work and background

For decades researchers have been exploring techniques related to semantic representations, to capture the meaning of text (Chersoni et al., 2021). Initially semantic representation began with lexical based methods. Lexical approaches use characters or words within a text. Examples of lexical approaches include Jaccard Similarity, which is computed by comparing the sets of unique tokens in two texts. Another early popular lexical approach to similarity scoring is called N-gram similarity, which compares sets of n-grams (sequences of n characters) from two texts and calculates their overlap. Jaccard similarity was

employed by Rocchio for text classification (1971), while Canvar et, al. used N-Gram similarity again for text classification (1994). However, lexical approaches only allow a very thin degree of semantic capture. Jaccard similarity is unable to track the order of words, while context capture in n-grams is limited to the length of the n-grams.

Researchers eventually landed on more performant approaches to text similarity, based on text vector embedding. Although vector embedding of text had been proposed in the 1970's by Salton, et, al. (1975) it would take it would take until the early 2000's for Bengio, et al (2003) to introduce using neural networks to learn distributed word representations as word embeddings, for embedding based approaches to overtake lexical approaches in popularity. Embedding models generate dense vector representation of text based on their semantic relationships within a corpus. Vector embedding allowed researchers to address a range of downstream NLP tasks related to knowledge representation, and crucially for text similarity applications as cosine similarity could be used to assess the similarity between two vector products. Initially researchers used shallow unsupervised neural networks to generate word embedding. Increasingly more sophisticated neural network approaches, namely transformer models (Vaswani, et, al. 2017), are employed for text embedding.

Model targets have also evolved. As Amigó et al. (2022) describes modern approaches to word vectorising were considered successful in applications related to prediction of the next suitable word in a sequence. However, they struggle with text representation in a semantic space. When vector embedding approaches use single words, the embedding space often becomes incoherent with their semantic meaning. More recent research has sort to overcome this issue by producing vectors based on longer text streams, like the entirety of a sentence or whole document. Research has shown vectors generated on longer word strings to be more accurate at capturing the semantic context of text and can improve performance in tasks related to similarity classification.

Following progression an increased interest in semantic textual similarity, benchmark challenges emerged (Marelli, et, al. 2014) (Agirre, et, al. 2012). Similarity benchmarks typically consist of hand labeled pairs of sentences, the label indicating the strength of semantic similarity.

The submission to these benchmarks use text embedding models to generate vectors of the sentence pairs and then use cosine similarity to generate a similarity score for each sentence pair (Yang, 2018). If the similarity score of the model roughly matches that of the existing label, then that counts as a correct classification. Our research will follow a similar methodology to the one seen in the benchmarks.

The top performing models for these benchmark, are all transformer models trained to generate sentence embeddings. One of the most popular, is an open-source model called SentBERT, developed by Reimers and Gurevych (2019). Other popular transform approaches include DistilBERT (Sent2Vec) (Pagliardini et al., 2018). Both SentBERT and DistilBERT will be considered in our experiment alongside Word2Vec, which will provide a baseline (Mikolov et al., 2013), and Doc2Vec (Le and Mikolov, 2014). Doc2Vec and Word2vec will also be the models we submit for parsing the test set because they can be readily manually trained.

The literature review has suggested both a potential architecture (one resembling that used in sentence similarity benchmarks) to address our task and that transformer models utilizing larger text sequences are likely to perform better than other approaches.

4 Accomplishments

The following list describes the task proposed and completed as part of the initial project proposal.

Data Related Tasks

- Load relevant data from hugging face and the UN, and preprocess text into list formats, preserving tokens in sentences, and their labels, group SDGs into topic labels – Completed.

Initial Evaluation of techniques

- Generate embeddings of an initial 100 article sample of the CNN Daily Mail dataset and compare the mean similarity scores between article 1 and the rest of the articles before quantitively assessing the outputs – completed.
- Using the labeled BBC news article dataset, test which techniques can

effectively classify articles to their correct news category – Completed.

Main experiment

- Using sample of the CNN Daily Mail dataset and the UN SDGs, develop an average similarity score for each SDG topic across all the articles in the dataset, before manually assessing the outputs for accuracy and errors – Completed.
- Hand label a subset of the CNN Daily Mail dataset with their most related SDG topic areas, before testing whether each embedding technique can classify the article with its correct SDG topic label – Completed.

5 Approach and Methodology

As hinted at in the accomplishment section, the methodology for this research occurs across two experimental phases. An initial phase of exploring and benchmarking the text embedding techniques. And a second phase of the main experiment testing whether the similarity scoring techniques can produce accurate SDG topic classifications. Both phases utilize the same data preprocessing pipeline alongside a mix of quantitative and qualitive assessment.

Data Processing

Data processing is intentionally limited to preserve as much the contextual information contained in the text of each article. No punctuation or stop words were removed or any stemming or lemmatizing performed on any dataset. After loading our datasets from hugging face into python as a list each sentence within a text was arranged so it would appear on a new line in the list. We also preserved the pre-defined labels of the articles and SDG text where they existed. Conveniently all the model tools used in the research accepted text data in a list format, meaning no distinct preprocessing steps were required for different tools.

Modeling Approaches

This section will provide an overview of the different text embedding approaches used in the course of the research to generate text similarity scores.

- **Word2Vec** – Is a neural network algorithm that uses either a Continuous Bag Of Word (CBOW) model or a Skip-Gram model, depending on the task, to produce word embeddings (Mikolov et al., 2013). The Word2Vec neural network has a single hidden layer with a non-linear activation function. Word2Vec requires manual training on a corpus of text before it can be used to develop embeddings.
- **SentBERT** – Is a pre-trained transformer model. A modification of the BERT architecture using Siamese and triplet networks to derive semantically meaningful sentence embeddings (Reimers and Gurevych, 2019). SentBERT's main drawback is that to apply transfer learning to it requires feeding it a large volume of manually labelled sentences pairs with corresponding similarity scores. Given the scope and time available to complete the project this was unachievable and should be left to future work.
- **Doc2Vec** – The neural network architecture of Doc2Vec is similar to that previously described in Word2Vec. However instead of simply learning the representation of words, Doc2Vec generates vector representations of entire documents. Doc2Vec like Word2Vec has to options for modelling both of which are similar to CBOW and Skip-Gram, however in both cases the context window is extended to the entirety of the document.
- **DistilBERT** (Sent2Vec) – DistilBERT is a version of BERT, trained on a smaller corpus of data, making it more useable. Deployed in Sent2Vec DistilBERT will generate sentence embeddings instead of single word embeddings. A drawback of using DistilBERT, is that there is no way of training the model to include the SDG training data.

Research Approach

Given the research aim, a pipeline was developed to categorize news articles based on their relation to SDG topics. We also conducted an initial evaluation of our text embedding models to gain an understanding of how we could build our final

pipeline and to benchmark their capabilities. The initial evaluation proved useful not only in suggesting an eventual test pipeline, but also in indicating how model and system architecture performance could be improved with adaptations in future research.

Initial Evaluation

The initial evaluation was divided into two stages. First an exploratory analysis of the different text similarity modelling techniques. Followed by a benchmark test reassembling the final test.

The initial evaluation won't be covered in detail. It involved using the text similarity models to identify articles similar to the first article in a further hundred articles from the CNN DailyMail test data. For each model, the top ten articles in terms of similarity score to the first article were printed and assessed for semantic accuracy. The evaluation phase also proved useful in indicating that a skip-gram approach for both the Word2Vec and Doc2Vec models could offer better performance over a CBOW implementation.

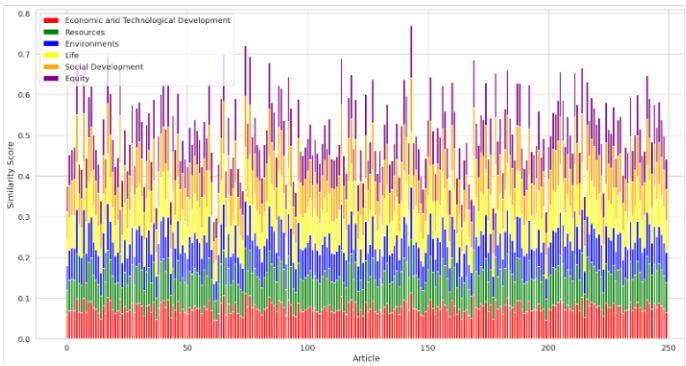
The second phase of the initial evaluation consisted of a test that would resemble our final experiment. The test used a dataset of BBC news articles from Hugging Face, each article already labeled with the category of news it was part of. For our task we randomly selected 3 articles from each news category and held them as a target set, the randomly selected articles grouped together by category. We then generated text embedding of both the target set and the remaining articles and used cosine similarity scoring to generate similarity scores between each of the grouped categories in the target set and the articles, allowing us to generate similarities scores for each article category. If the category with the highest cosine similarity score was the same as the labeled category for the article then this would be considered an accurate classification. The BBC article classification task provided insight into which models and optimization would likely perform best in the final experiment, as well as hinting at an architecture for our final pipeline. Later we will see that unlike in our final test, the BBC News article dataset presented the advantage of being of significant and balanced volume. For models which produce similarity scores based on sentences or words, we took the mean similarity score for each SDG topic to obtain a final similarity score.

Main Experiment

The main experiment tested a pipeline in which each model attempted to classify articles from the CNN Daily Mail dataset by their relevance to SDG topics. Analysis of model performance for this task would happen on both a qualitative and quantitative font. The pipeline deployed for classifying articles by their relevance to SDG topics remains consistent across both stages of analysis.

The pipeline began by grouping the content of each SDG into six topic labels, and associating each sentence to those topics, in effect resembling a series of text like an article. After which semantic modelling techniques are used to generate vector representations of both the SDG topics and the CNN Daily Mail article dataset. Cosine similarity was employed to generate similarity scores for each article vector and each SDG topic vector.

Figure 1: Word2Vec SDG Similarity Scores for 250 articles in the CNN Daily Mail Dataset

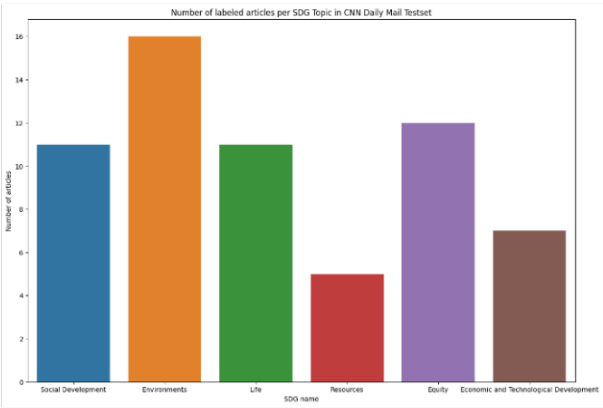


In the qualitative assessment of the our model output, the top 10 articles in terms of cumulative similarity score were printed and an assessment made of their relevance to SDGs in general, which is then considered as a score. The top 5 article for each SDG topic as identified by each model were printed and again an evaluation was made to their accuracy and collated into a score. By evaluating the model similarity score for articles at both a cumulative and topic level we're able to identify the type of articles that our models are effective at associating with a particular SDG topic and where they're making errors.

In the second phase of the main experiment the research aimed to generate more concrete accuracy metrics for each model. To do this, an unseen subsample of the CNN Daily Mail database was

given labels related to SDG topics they're associated with. The process of labelling the data suggested a potential underlying issue with the dataset. Only 57 of the 800 articles considered for a label can be fairly attributed to and SDG topic, indicating a weak prevalence of SDG relevant articles in general in the dataset, at around 1 in 14. Figure 2 shows the distribution of labels by SDG topic, 'Envrioments' being the most prevalent and 'Resources' being the least. After labelling, the labelled subset of articles were grouped into a separate dataset. The pipeline to classify articles by SDG topic was applied to the labelled dataset. If the topic with highest cosine similarity score matched that of the label, then this was considered an accurate classification by our pipeline. This process allowed us to generate some quantitative analysis of our models, and helped to identify which classes they were particularly bad at categorizing. We would expect a performant SDG classification model pipeline to achieve a relatively high degree of accuracy in associating articles with their correct labels.

Figure 2: SDG Topic Labelled CNN Daily Mail articles, by topic.



One of the issues particularly with the second phases of the experiment was the low accuracies achieved by our models. In most cases the models at least did manage to capture the full spectrum of possible output categories. The baseline model unoptimized Word2Vec performed especially poorly and as expected was unable to effectively map the semantic context of each article. Given that performance and model accuracy of our benchmark tests in the BBC test saw some models exceed 60% classification accuracy, it's reasonable to anticipate that the results for final pipeline experiment may have been better, and hints at the fact we may have issues with out datasets in the final experiment.

Libraries

Many libraries were used during the two phases of the project. To load, gather, visualize, and arrange data, a mixture of Hugging Face, Pandas, Numpy, Matplotlib and Seaborn were used. In terms of text processing we used NLTK, Genism and Re in different instances, to perform simple pre processing steps, mainly just ensuing tokens remained in the correct list formation, ready to be processed by our models. A range of libraries were used to import and train our models, Genism contained both Word2Vec and Doc2Vec, whereas SentBERT and DistilBERT were accessed through their own imported libraries. Scikit-Learn was used to produce cosine similarity scores for all models.

6 Datasets

The project employed three datasets. The BBC news article dataset from hugging face, used to initially evaluate the performance of our sentence embedding techniques. The two other datasets we'll use are the Hugging Face collection of CNN and Daily Mail articles, and the text of the UNs SDGs which was taken from the UN's website and has been manually compiled by topic. All three datasets were used as described in the methodology above.

Figure 3: Summary of key statistics related to each dataset use in the project

	Number of articles/topics	Mean sentence length per article/topic	Mean number of sentences per article/topic
BBC Test Articles	250	21.91	18.58
Sustainable Development Goals	6	19.88	106.83
SDG Labeled CNN DailyMail Articles	57	19.40	25.79
Unlabeled CNN Daily Mail Articles	250	19.68	31.39

The global dataset statistics suggest a major issue with the experiment. Limited alignment between CNN Daily Mail and the SDG datasets in terms of mean number of sentences per topic for each SDG could be impacting model outputs, as the SDG topics are obviously very different to the CNN Daily Mail articles in terms of the number of

sentences. We would anticipate Doc2Vec to be worst impacted by this misalignment, given its analyzing text on a document wide, as opposed to sentence level.

7 Dataset Preprocessing

As previously mentioned minimal data preprocessing was applied to our datasets. We did use Genism's simple_preprocess in the case of Word2Vec, which has the impact of lower casing all the words, removing all punctuation, and crucially for Word2Vec tokenizes strings of text data into individual words. Limited preprocessing was applied to all three datasets to preserve as much contextual information as possible. In further research it would make sense to see if introducing a greater degree of text preprocessing would lead to improved modelling performance.

8 Baseline

An unoptimized version of Word2Vec was used as a baseline throughout the experiment. As Amigó et al. (2022) suggested, generating vector embeddings based on single words was always likely to struggle with effectively mapping text to a semantic space as single word vector embeddings are likely to improve incoherent with their semantic meaning. Embedding techniques that use both more sophisticated neural network techniques or which generate vectors based on longer sequences of text should prove more performant than Word2Vec making it a sensible option for a baseline. If any of the other models perform significantly worse than Word2Vec we might safely assume that there was an issue with their implementation, or that they're very unsuited to the task at hand.

9 Results, error analysis

Model optimizations

Through each experiment we deploy a range of models each with differing architectures. For instance, the impact of using a CBOW or a Skip Gram approach is experimented for both Word2Vec and Doc2Vec models. We also experimented with the size of the corpus used to train the Word2Vec and Doc2Vec models. Although the experiment didn't explore a more traditional hyperparameter optimization related to vector size and epochs for the Word2Vec, and Doc2Vec models. It did take in a range of separate approaches, seven in total, to understand the most effective target vector, i.e. one based on a single words, single sentence or entire documents, and model architecture transformer or neural network.

Initial Evaluation

The results of the first qualitive evaluation, based on manually inspecting the top 10 aticles by similarity score to the first article, found that the DistilBERT model performed the best. Reflecting on the relevance of the test, all models performed relatively poorly, and the results shouldn't be considered as highly indicative of further performance.

Figure 4 Assessed relevance to Article 1 and the rest of the first 100 articles in the CNN DialyMail dataset by top 10 similarity score

	Relevance Score out of 10
Word2Vec CBOW	2
Word2Vec Skpi Gram	2
SentBERT	2
Doc2Vec CBOW	3
Doc2Vec Skip Gram	2
DistilBERT	4

The BBC news article test involved matching articles to their relevant news category using similarity scoring. This test saw Doc2Vec CBOW outperform all other models, Doc2Vec Skip Gram and SentBERT also performed well. Our baseline Word2Vec models failed to map the distribution of categories leading it to deliver a very poor score. The test indicated that transformer models and models focused on text streams longer than a sentence are likely to perform well in the final task.

Figure 5 Classification Metrics from BBC Article Classification task.

	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)
Word2Vec CBOW	16.00	3.20	20.00	5.52
Word2Vec Skip Gram	16.40	23.21	20.36	21.70
SentBERT	75.20	79.34	75.51	77.38
Doc2Vec CBOW	86.80	86.75	86.55	86.65
Doc2Vec Skip Gram	79.20	78.96	79.21	79.09
DistilBERT	40.00	44.39	41.14	42.70

Main Experiment

Similarity score outputs were inspected at both a cumulative score level, that is the combined similarity across all SDG topics for the top 10 most relevant articles, and at a topic category level for the top 5 scoring articles by category. SentBERT performed best across both tasks, followed closely by the Doc2Vec models. Across all models many

of the same articles appeared reputedly across each category. Models were best at highlighting Economic and Environmental related articles, although this could be a result of the weak relevance of SDG related articles in the dataset.

Figure 6: Assessed relevance score for 10 articles by cumulative similarity score, and top 5 articles per SDG category across 100 CNN Daily Mail Articles.

	Category Relevance (out of 30)	Cumlative top 10 Relevance	Total Correct (%)
Word2Vec Skip Gram	3	5	20.00%
SentBERT	9	6	37.50%
Doc2Vec CBOW	6	7	32.50%
Doc2Vec Skip Gram	6	7	32.50%
DistilBERT	0	1	2.50%

The process of manually parsing model outputs for accurate classifications highlighted that the main issue causing misclassifications was due to limited ability by the models to differentiate scoring by category. As a result, many of the same articles achieved high scores across all categories. Suggesting a weak prevalence of SDG relevant articles in the dataset. Demonstrates that the models are mapping each of our SDG topics into a very similar semantic space, so all SDG topics are in high proximity to all over topics, resulting in little differentiation in the list of articles with high similarity scores across the range of SDG topic categories.

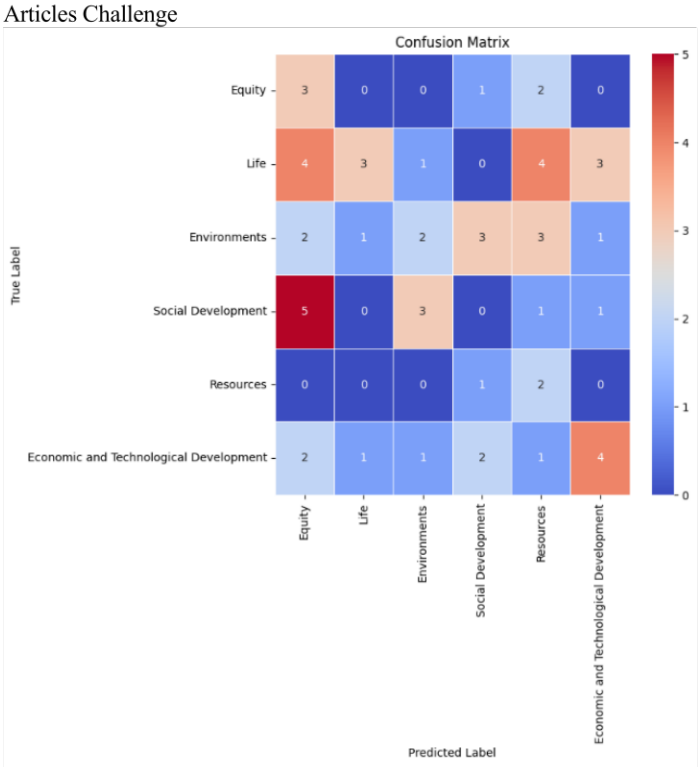
The final test involved using our SDG topic classification pipeline on the prelabeled CNN Daily Mail articles. Figure 7 indicates that the best performing model was Doc2Vec Skip Gram followed closely by SentBERT. The baseline Word2Vec performed particularly badly, failing to effectively map the distribution of classification possibilities, making all it's classification in just three of the six SDG topic categories. The difference in capability to map the full distribution of topics was the crucial difference between the baseline model and the best performing model. Although this same issue arose not only for our baseline but for several other approaches, as indicated by their low Recall and F1 Scores. Overall, all approaches performed poorly in the CNN DailyMail SDG topic labelled classification task, especially in comparison to the BBC article categorization task. Suggesting that there could be limited alignment between the targeted SDG Topic corpus and the CNN Daily Mail dataset.

Figure 7: Classification metrics for all models performing the Labeled Articles Challenge

	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)
Word2Vec Skip Gram	21.05	4.34	17.54	6.90
SentBERT	24.56	23.93	22.81	18.85
Doc2Vec CBOW	17.54	14.73	17.54	14.45
Doc2Vec Skip Gram	24.56	33.16	24.56	24.23
DistilBERT	14.04	5.43	12.28	4.65

Figure 8 is a classification matrix of the best performing model Doc2Vec Skip Gram, indicating that the model has a tendency to over classifying Equity and Resource related topic classes, while under classifying the Environment and Social Development classes.

Figure 8: Doc2Vec Skip Gram Classification Matrix Labeled Articles Challenge



10 Conclusions, Lessons Learned, and Further Research

Conclusion

The experiments show that text embedding models which generate vectors using longer text strings, perform better at semantic matching tasks. And that the performance advantage this enables can lead to less sophisticated shallow neural network approaches like Doc2Vec Skip Gram model outperforming transformer-based embedding approaches. This came as a surprise given the performance of SentBERT in relation to

other semantic similarity benchmarking tasks seen in the literature review. The experiment also indicates the importance of applying relevant training data. Heavily pre-trained models underperformed approaches which incorporate updates from directly relevant text samples.

Finally, the research failed to arrive at a tool accurate enough to be used for SDG news monitoring tool. Extensive further work building on this paper is still required to deliver a performant version of the monitoring tool.

Lessons Learned

The process of developing this research unearthed two key learnings.

Use text datasets which contain strong semantic signals. Although the CNN Daily Mail dataset is a realistic representation of a daily news feed, the weak prevalence of articles related to SDG topics made it challenging to highlight the performance of the models.

Target and test datasets in semantic representation tasks need to be well aligned. The discrepancy between classification accuracy of models in the BBC labelled article test in comparison to the final CNN Daily Mail test, suggests a major issue with data setup of the later experiment. Better performance may have been achieved by using a subset of articles already SDG topic labelled as the targets for similarity scoring instead of the SDG text.

Further Research

There is much room for further research building upon this project. Given more time and compute resources I would address the following:

- **Explore other modelling approaches**. Considering the strong performance of Doc2Vec it would be interesting to explore a BERT based model which generated it's embedding based on the entirety of a document. It would also be interesting to try BERT-based LDA topic modelling in this research to see if it would yield better accuracy performance in the tests.
- **Experiment with introducing more text preprocessing** steps to see their impact on model performance.
- **Test the models on a dataset with closer semantic proximity to the UNs SDG**. A deprecated version of the BBC dataset which excluded Entertainment and Sports articles, would be where I would start.
- **Introduce greater model optimization.** Hyperparameter optimization of Word2Vec and Doc2Vec models. And update pre-trained transformer models with SDG goals and a labelled subsample of articles related to the goals.

References

Agirre, E., Diab, M., Cer, D., & Gonzalez-Agirre, A. (2012). Semeval-2012 task 6: A pilot on semantic textual similarity. In Proceedings of the First Joint Conference on Lexical and Computational Semantics (Vol. 1 and 2), pages 385–393. Association for Computational Linguistics.

Amigó, E., Ariza-Casabona, A., Fresno, V., & Martí, M. A. (2022). Information Theory–based Compositional Distributional Semantics. Computational Linguistics, 48(4):907–948.

Augenstein, I., Rocktäschel, T., Vlachos, A., & Bontcheva, K. (2016). Stance detection with bidirectional conditional encoding. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 876–885, Austin, Texas. Association for Computational Linguistics.

Bengio, Y., Ducharme, R., Vincent, P., & Jauvin, C. (2003). A Neural Probabilistic Language Model. Journal of Machine Learning Research, 3(Feb), 1137-1155.

Cavnar, W. B., & Trenkle, J. M. (1994). N-Gram-Based Text Categorization. In Proceedings of the 3rd Annual Symposium on Document Analysis and Information Retrieval (SDAIR-94), pages 161-175.

Chersoni, E., Santus, E., Huang, C.-R., & Lenci, A. (2021). Decoding Word Embeddings with Brain-Based Semantic Features. Computational Linguistics, 47(3):663–698.

Le, Q., & Mikolov, T. (2014). Distributed Representations of Sentences and Documents. In Proceedings of the 31st International Conference on Machine Learning, PMLR 32(2), pages 1188-1196.

Marelli, M., et al. (2014). A SICK cure for the evaluation of compositional distributional semantic models. In Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14).

Mikolov, T., et al. (2013). Distributed Representations of Words and Phrases and their

Compositionality. In Advances in Neural Information Processing Systems, 26, pages 3111-3119.

Pagliardini, M., et al. (2018). An Efficient Framework for Learning Sentence Representations. In Proceedings of the 6th International Conference on Learning Representations (ICLR 2018), Vancouver, BC, Canada.

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP).

Rocchio, J. (1971). Relevance feedback in information retrieval. In The SMART Retrieval System: Experiments in Automatic Document Processing. Prentice Hall.

Salton, G., Wong, A., & Yang, C. S. (1975). A vector space model for automatic indexing. Communications of the ACM, 18(11): 613-620.

Vaswani, A., et al. (2017). Attention is All You Need. In Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

Yang, Y., & Pedersen, J. O. (1997). A comparative study on feature selection in text categorization. In Proceedings of the 14th International Conference on Machine Learning (ICML-97) pages 412-420.

Yang, Y., Yuan, S., Cer, D., Kong, S. Y., Constant, N., Pilar, P., Ge, H., Sung, Y. H., Strope, B., & Kurzweil, R. (2018). Learning semantic textual similarity from conversations. In Proceedings of the RepL4NLP Workshop at the 2018 Conference of the Association for Computational Linguistics (ACL) pages 135-143.