Proceedings of the 4th Financial Narrative Processing Workshop @ LREC 2022, pages 79–82
Marseille, 24 June 2022
© European Language Resources Association (ELRA), licensed under CC-BY-NC-4.0

Financial narrative Summarisation Using a Hybrid TF-IDF and Clustering summariser: AO-Lancs System at FNS 2022

Andrew Ogden and Mahmoud El-Haj
UCREL NLP Group, Lancaster University
{a.g.ogden1,m.el-haj}@lancaster.ac.uk

Abstract

This paper describes the HTAC system submitted to the Financial Narrative Summarization Shared Task (FNS-2022). A methodology implementing Financial narrative Processing (FNP) to summarise financial annual reports, named Hybrid TF-IDF and Clustering (HTAC). This involves a hybrid approach combining TF-IDF sentence ranking as an NLP tool with a state-of-the-art Clustering Machine learning model to produce short 1000-word summaries of long financial annual reports. These Annual Reports are a legal responsibility of public companies and are in excess of 50,000 words. The model extracts the crucial information from these documents, discarding the extraneous content, leaving only the crucial information in a shorter, non-redundant summary. Producing summaries that are more effective than summaries produced by two pre-existing generic summarisers.

Keywords: FNS, Summarization, English, Spanish, Greek, NLP

1. Introduction

Each financial year companies release an annual report, these reports serve to describe their current financial state as well as their financial state throughout the previous year. These reports vary in length and composition but are often dozens of pages in length and contain numerous different sections such as statements from the company's Chief Executive Officer (CEO), Chief Financial Officer (CFO) and President, as well as many others. The reports also contain the financial statements from the past year such as Balance sheets, income statements and cash flow statements[1]. These documents must be summarized effectively allowing readers to ignore any superfluous information, making the process of parsing the reports much faster. To effectively summarise lengthy and complex documents such as financial annual reports, as much information as possible must be collated to determine the sentence rankings, providing them as much weight as possible. To this point, hybrid summarizers can be implemented, which take the information produced by several Natural Language Processing (NLP) and machine learning techniques and combine them to produce new sentence rankings, using all of the available information. This paper covers a hybrid TF-IDF and Clustering summarizer combining the base NLP technique of TF-IDF with the results of a K-Means Clustering model, using state of the art Word2Vec Embeddings, intending to improve upon the individual results of each.

2. Background

NLP summarisation has been a well-researched topic for the past decade as researchers have recognised the benefits of automatically generating summaries of large blocks of text. The purpose of automatic text summarisation is to produce a condensed, non-redundant summary text from either a single or multiple input texts (Nenkova and McKeown, 2011). The field of automatic test summarisation branched into two distinct approaches; Extractive summarisation in which sentences from the initial document compose the summary (Gupta and Lehal, 2010) and Abstractive summarisation where the summary text is entirely generated by the summariser but based on the contents of the input document (Moratanch and Chitrakala, 2016). The summariser in this project utilises extractive techniques. Extractive summarisation uses a variety of statistical methods to score parts of the original document (i.e., sentences and phrases) based on their perceived importance. This incorporates a number of different feature extraction/engineering methods and evaluations. The methods used in this Hybrid summariser are TF-IDF sentence Ranking (Luhn, 1958) and a Clustering Machine Learning Model (Radev et al., 2004; Liu and Lindroos, 2006; El-Haj et al., 2011). TF-IDF sentence ranking is a statistical technique utilising word frequencies to score the importance of certain words, it is a seminal concept in automatic text summarisation (Nenkova and McKeown, 2011). Clustering, specifically K-Means clustering is a machine learning model that clusters numerical data points around a set of iteratively generated centroids (Kanungo et al., 2002), which can be used after the input text has been converted to numerical vectors. To conclude, research into the greater area of this paper, namely NLP summarisation has been taking place for over half a century, with the seminal piece of research taking place in 1958 (Luhn, 1958). Since the publication of Luhns paper, research into NPL summarisation has come a long way with the discipline splitting up into Abstractive and Ex-

[1] Corporate Finance Institute(CFI) 2022 `https://corporatefinanceinstitute.com/resources/knowledge/finance/annual-report/`

tractive approaches and the use of machine learning to enhance results. However, despite the many new techniques published since that time, the core of Luhn's research using statistical analysis of words and word counts remains important and widely used.

3. Methodology

This paper describes the HTAC system submitted to the Financial Narrative Summarization Shared Task (FNS-2022) (Zmandar et al., 2022). The shared task has been running since 2020 (El-Haj et al., 2020b; Zmandar et al., 2021) as part of the Financial Narrative Processing (FNP) workshop series (El-Haj et al., 2022; El-Haj et al., 2021; El-Haj et al., 2020a; El-Haj et al., 2019; El-Haj et al., 2018).

The Summariser uses an extractive hybrid approach, using a statistical method to combine the TF-IDF scores of each sentence with the Euclidean distance to the centre point of its cluster. These new scores determine the final sentence rankings, the highest-scoring sentences are added to the final summary until the summary reaches the 1000 word limit. The summariser was developed using a highly modular approach, this allows each part to be changed and run separately. Overall, this meant that each part of the process could be changed, and so long as the output format remained consistent, all other parts would continue to function effectively with the new data. Consequently, a lot of time was saved as both summarisers took a significant time to run and now only needed to be re-run when they were altered. With this modularity, it became possible to test different changes to individual parts of the hybrid summariser easily, allowing for different weights when combining the data. The 4 main components are the TF-IDF and Clustering results generators, the range normaliser, and the combination summariser.

3.1. TF-IDF

TF-IDF sentence ranking provides a good base for initial summaries, being a core pillar of NLP. TF-IDF is a statistical technique using word counts to ascribe importance to certain words based on their perceived relevance to the document as whole. Thus sentences scores can be calculated when the scores for each word in the sentence are summed. The training documents in the dataset were used to create a large dictionary of word counts, which was concatenated to the top 20000 entries, removing very low-frequency words and improving computation time later on. This dictionary can then be used to create the TF-IDF word scores. The training data was used instead of a per-document basis as this allows the words of the input document to be compared against a larger cross-section of financial annual reports, not just within the context of its own content. This provided more weight to each of the frequencies and thus TF-IDF scores as they are representative of a greater dataset. For each input document, all words were added to the word dictionary, ensuring every word in the document is present in the dictionary. This dictionary is then used to create a TF-IDF score dictionary contain every discrete word in the input document, and its TF-IDF score. These scores are then added up for each word in a sentence, creating a new dictionary with every sentence and its TF-IDF score. This dictionary is then saved and later accessed by the hybrid summariser.

3.2. Clustering

This component utilises Machine learning to cluster each sentence around a set of cluster points, this then allows us to calculate the Euclidean distance between each point and the centroid of the nearest cluster. The clusters can be interpreted as a group of semantically identical sentences, that carry similar information. Thus sentences with the lowest distance, are more likely to be important as they are closer to a central concept or notion of the input document, which in the abstract is what the clusters represent. This component produces results that are more difficult to utilise in simple sentence ranking. While the TF-IDF scores provide a simple and easy to sort metric, the Euclidean distance between points can be harder to utilise, as the output of the clustering model is the coordinates of the sentences and cluster centroids in an abstract space. This means that for each point we must determine the distance between its location and the centroid of each cluster to find the distance to the nearest cluster and then record it. The summarisation in this version was implemented using K-Means clustering via SciKit Learn (Pedregosa et al., 2011), a machine learning technique that aims to cluster data points around a set of iteratively generated points. Clustering requires the input data to be in a numerical form. To do this the text of each report was converted into mathematical vector representation using Word2Vec from Gensim (Rehurek and Sojka, 2010). Word2Vec was chosen as it is a state of the art and effective way of converting text into a numerical vector representation, thus preferable to older techniques, such as bag-of-words. Word2Vec uses a low-level neural network, implementing both skip-grams and continuous-bag-of-words to return the word embeddings for the input text, producing results that are more accurate and information-dense than older models i.e., bag-of-words (Rehurek and Sojka, 2010). The Word2Vec model was created and then trained on the entire training dataset. This trained model was then loaded into the summariser and for each input document, its vocabulary was updated with every word in the input report. The model then undergoes further training with the input text. The word vectors were then extracted, combined into the appropriate sentence vectors, and passed to the K-Means Clustering model (Pedregosa et al., 2011), using 9 clusters. The number for clusters used was selected using an iterative methodology, the summariser was ran with every cluster number between 4 and 11, the resulting data showd that 9 was the optimal number. The data points were then extracted from the completed model and the Euclidean distance between each point and the centroid of its cluster was calculated. This information was then used to create a dictionary of sentences and their Euclidean distances, which was then saved allowing the combination hybrid summariser to access later.

3.3. Range Normaliser

To combine the results of the two summarisers, they were first normalised ensuring that there are in the same range. The original data points were mapped to the range of 0-100, this was chosen as it is a simple range that is both easy to visualise and to alter, allowing the weightings to be change into several commonly used increments. To normalise the sentence scores produced by each summariser, it takes the original values and replaces them with the corresponding value in the new range (0-100), this is done using the following equation. $N_{val} = \frac{(O_{val} - O_{min})N_{max}}{O_{max}} + N_{min}$ The same process is then applied to the results of the Clustering summariser and then each new value is taken from 100, to reflect that the lower the clustering score, the higher the sentence should be ranked. Once completed, the new datasets will now have the same data as the original, only represented in the new range. The new datasets will be saved and later accessed by the hybrid summariser for it to use accordingly.

3.4. Hybrid TF-IDF and Clustering (HTAC)

To combine the sentence scores, the weightings, 40/60 in favour of clustering were used, this was determined by testing each combination of weightings between 10/90 and 90/10 in 9 evenly spaced increments, and comparing results. For each summary, the two sentence scores are combined and added to a new data frame once the appropriate weights have been applied. The new data frame containing the combined sentence scores is then sorted, from highest to lowest. This data frame will then be iterated through, starting with the highest scored sentence. Each sentence, after passing several quality checks discussed below, will be added to two lists -

- The first, contains the sentences for the final summary.
- The second, contains the sentences index in the original document.

Once no more sentences will fit in the list without exceeding the word limit, the two new lists will be used to create a new data frame where the first column contains the sentences, and the second contains their corresponding index in the original report. This data frame will then be sorted by the indexes, resulting in the sentences being in their original order as found in the input report. The sorted sentences will then be joined to create a String containing the final summary.

3.5. Quality checks

Before a sentence is added to the final summary, several checks are applied to make sure that it is not redundant.

- The first and simplest is a length check, ensuring that only sentences over 10 characters are added. This removes any issues with the tokenization process which often produces some single word sentences and while these may have a high score due to their low length, they tend to have a low value in a summary as they have little of their original context remaining.
- The second check is to ensure that a very similar sentence has not already been added. This check involves comparing each new sentence to all sentences in the current summary using their Rouge-1 score. Sentences which score 0.7 or above indicates that they are 70 per cent alike, and so the sentence is not added, allowing different, more unique sentence can take its place.

Rouge Variant	**English**	**Spanish**	**Greek**
Rouge-1/F	0.381	0.402	0.336
Rouge-L/F	0.297	0.165	0.250
Rouge-SU4/F	0.200	0.192	0.178
Rouge-2/F	0.141	0.123	0.129

Table 1: Results on Official Validation

Table 1 shows the average Rouge score produced for each language dataset, using several Rouge variants. The Rouge-2 / F scores are similar across the 3 languages, with the English set expectedly producing the best results, being a far larger dataset.

Rouge Variant	**English**	**Spanish**	**Greek**
Rouge-1/F	0.317	0.448	0.334
Rouge-L/F	0.257	0.164	0.252
Rouge-SU4/F	0.185	0.211	0.182
Rouge-2/F	0.143	0.134	0.131

Table 2: Results on Official Testing Set

Table 2 shows the results of the FNS 2022 Task for this System, generated using the testing datasets. The results from the Validation and training data sets were similar, with a small increase in each of the Rouge-2 / F scores across the 3 languages in the Testing results. Once again, The English language set has the highest scoring results.

4. Conclusion

To conclude, the extractive HTAC (Hybrid TF-IDF And Clustering) system discussed in this paper constitutes an effective method of summarising financial annual reports, combining the sentence scores produced by multiple individual methodologies into final sentence rankings using statistical techniques. The results were consistent across both the validation and training datasets as well as all 3 languages. This is a positive result, with a higher level of consistency across the 3 languages than most other participants in the FNS 2022 shared task. Future work could be undertaken to determine why the system presented in this paper maintained such high levels of consistency across the 3 languages. As this could be combined with learning from the other systems that outperformed on the English dataset, whilst suffering quality drops in within the Greek and Spanish datasets. This could allow the strengths of each system to create improvements, consistently across multiple languages.

5. Bibliographical References

El-Haj, M., Kruschwitz, U., and Fox, C. (2011). Exploring clustering for multi-document arabic summarisation. In *Asia Information Retrieval Symposium*, pages 550–561. Springer.

El-Haj, M., Rayson, P., and Moore, A. (2018). The first financial narrative processing workshop (fnp 2018). In *Proceedings of the LREC 2018 Workshop*.

Mahmoud El-Haj, et al., editors. (2019). *Proceedings of the Second Financial Narrative Processing Workshop (FNP 2019)*, Turku, Finland, September. Linköping University Electronic Press.

Dr Mahmoud El-Haj, et al., editors. (2020a). *Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation*, Barcelona, Spain (Online), December. COLING.

El-Haj, M., AbuRa'ed, A., Litvak, M., Pittaras, N., and Giannakopoulos, G. (2020b). The financial narrative summarisation shared task (FNS 2020). In *Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation*, pages 1–12, Barcelona, Spain (Online), December. COLING.

Mahmoud El-Haj, et al., editors. (2021). *Proceedings of the 3rd Financial Narrative Processing Workshop*, Lancaster, United Kingdom, 15-16 September. Association for Computational Linguistics.

Mahmoud El-Haj, et al., editors. (2022). *Proceedings of the 4th Financial Narrative Processing Workshop*, Marseille, France, 24 June. The 13th Language Resources and Evaluation Conference, LREC 2022.

Gupta, V. and Lehal, G. S. (2010). A survey of text summarization extractive techniques. *Journal of emerging technologies in web intelligence*, 2(3):258–268.

Kanungo, T., Mount, D. M., Netanyahu, N. S., Piatko, C. D., Silverman, R., and Wu, A. Y. (2002). An efficient k-means clustering algorithm: Analysis and implementation. *IEEE transactions on pattern analysis and machine intelligence*, 24(7):881–892.

Liu, S. and Lindroos, J. (2006). Experiences from automatic summarization of imf staff reports. *Practical Data Mining: Applications, Experiences and Challenges*, page 43.

Luhn, H. P. (1958). The automatic creation of literature abstracts. *IBM Journal of research and development*, 2(2):159–165.

Moratanch, N. and Chitrakala, S. (2016). A survey on abstractive text summarization. In *2016 International Conference on Circuit, power and computing technologies (ICCPCT)*, pages 1–7. IEEE.

Nenkova, A. and McKeown, K. (2011). *Automatic summarization*. Now Publishers Inc.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830.

Radev, D. R., Jing, H., Styś, M., and Tam, D. (2004). Centroid-based summarization of multiple documents. *Information Processing & Management*, 40(6):919–938.

Rehurek, R. and Sojka, P. (2010). Software framework for topic modelling with large corpora. In *In Proceedings of the LREC 2010 workshop on new challenges for NLP frameworks*. Citeseer.

Zmandar, N., El-Haj, M., Rayson, P., Abura'Ed, A., Litvak, M., Giannakopoulos, G., and Pittaras, N. (2021). The financial narrative summarisation shared task FNS 2021. In *Proceedings of the 3rd Financial Narrative Processing Workshop*, pages 120–125, Lancaster, United Kingdom, 15-16 September. Association for Computational Linguistics.

Zmandar, N., El-Haj, M., Rayson, P., Abura'Ed, A., Litvak, M., Giannakopoulos, G., Pittaras, N., Carbajo-Coronado, B., and Moreno-Sandoval, A. (2022). The financial narrative summarisation shared task (fns 2022). In *Proceedings of the 4th Financial Narrative Processing Workshop*, Marseille, France, 24June. The 13th Language Resources and Evaluation Conference, LREC 2022.