The Financial Narrative Summarisation Shared Task (FNS 2020)

Mahmoud El-Haj[1] **Ahmed AbuRa'ed**[2] **Marina Litvak**[3]
Nikiforos Pittaras[4] **and** **George Giannakopoulos**[4]

[1]Lancaster University, UK, [2]UPF, Spain, [3]SCE, Israel, [4]IIT Demokritos, Greece
[1]m.el-haj@lancaster.ac.uk, [2]ahmed.aburaed@upf.edu,
[3]litvak.marina@gmail.com
[4]{ggianna,pittarasnikif}@iit.demokritos.gr

Abstract

This paper presents the results and findings of the Financial Narrative Summarisation shared task (FNS 2020) on summarising UK annual reports. The shared task was organised as part of the 1st Financial Narrative Processing and Financial Narrative Summarisation Workshop (FNP-FNS 2020). The shared task included one main task which is the use of either abstractive or extractive summarisation methodologies and techniques to automatically summarise UK financial annual reports. FNS summarisation shared task is the first to target financial annual reports. The data for the shared task was created and collected from publicly available UK annual reports published by firms listed on the London Stock Exchange (LSE). A total number of 24 systems from 9 different teams participated in the shared task. In addition we had 2 baseline summarisers and additional 2 topline summarisers to help evaluate and compare against the results of the participants.

1 Introduction

Companies around the world produce a variety of reports containing both narrative and numerical information at various times during their financial year. Such reports are referred to as financial disclosures and usually include quarterly reports, preliminary earnings announcements, conference calls, press releases financial annual reports (El-Haj et al., 2018a). This creates a vast financial information environment which can be impossible to keep track of (Salzedo et al., 2014; El-Haj et al., 2014a; El Haj et al., 2018b; Athanasakou et al., 2019).

The same set of information can be crucial for a number of different reasons. It can help highlight company achievements and gain support from shareholders in the stock exchange. It can identify risks and opportunities that investors need to take into account. Financial reporting is also strongly related to due diligence processes during mergers and acquisitions, as well as during auditing processes. All the above uses, many of which can be critical during a company life-cycle, show the vital need for automatic summarisers, in order to reduce the amount of time and effort required by stakeholders - be they shareholders investors or other parties - to read and analyse those documents.

The financial narrative summarisation (FNS) shared task focuses on annual reports produced by UK firms listed on the London Stock Exchange (LSE). In the UK and elsewhere, annual reports structure is much less rigid than those produced in the US elhaj2019, . Companies usually produce glossy brochures with a much looser structure that is usually disseminated in PDF file format. This makes automatic summarisation of narratives in UK annual reports a challenging task, since the structure of those documents needs to be extracted first in order to summarise the narrative sections of the annual reports. This can be done by detecting narrative sections that usually include the management disclosures (financial narratives) rather than the financial statements of the annual reports (El-Haj et al., 2016). Previously, the 1st and 2nd Financial Narrative Processing Workshops (FNP 2018 and FNP 2019) focused on the process of extracting and analysing financial narratives from multilingual financial statements written in languages such as English, French and Spanish (El-Haj et al., 2018a; El-Haj et al., 2019c).

Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation, pages 1–12
Barcelona, Spain (Online), December 12, 2020.

In this task we ask participants to generate automatic summaries for lengthy UK annual reports (each with more than 60,000 words on average) by focusing on the financial narratives of the reports and producing a summary of no more than 1000 words for each annual report.

This paper presents the results and findings of the Finanical Narrative Summarization shared task, as follows. We begin with an overview of related work in Section 2. We then describe the data, elaborating on the task (Section 3), and continue with the baseline and topline system descriptions (Section 4). After this complete picture of the task and setting, we overview the submitted systems, in section 5. We conclude the paper with the task evaluation and related results as well as an appropriate short discussion of the findings (Section 6).

2 Related Work

The increased availability of financial report data has been met with research interest for applying automatic summarisation methods. The task of automatic text summarisation aims to produce a condensed, informative and non-redundant summary from a single or multiple input texts (Nenkova and McKeown, 2011). This is achieved by either identifying and ranking subsets of the input text (i.e. extractive approaches ((Gupta and Lehal, 2010)), or by generating the summary from scratch (i.e. abstractive methods (Moratanch and Chitrakala, 2016)).

Extractive summarisation methods have received far higher attention than their abstractive counterpart methods. This is mainly due to their relative simple approach when compared to the comparatively high requirements of the abstractive methods, especially when it comes to computational resources and data availability.

Extractive summarisation utilises scoring approaches to identify and reorder parts of the input (e.g. sentences, phrases and/or passages), using a variety of feature extraction/engineering and evaluation methods (Luhn, 1958; Baxendale, 1958; Edmundson, 1969; Mori, 2002; McCargar, 2004; El-Haj, 2012; Giannakopoulos et al., 2008; Koulali et al., 2013). Where adequate data is available, machine learning methods have been employed, such as Hidden Markov Models (Fung and Ngai, 2006), topic-based modelling (Aries et al., 2015), clustering methods (Radev et al., 2000; Liu and Lindroos, 2006; Kruengkrai and Jaruskulchai, 2003), deep neural network classification (Nallapati et al., 2017) and language models (Liu, 2019).

The application of summarisation and natural language processing techniques in general has promising applications in the financial domain (El-Haj et al., 2019b). Recently, statistical features with heuristic approaches have been used to summarise financial disclosure texts (Cardinaels et al., 2019), generating summaries with reduced positive bias and leading to more conservative valuation judgements by investors that receive them.

Furthermore, the financial narrative summarisation task (El-Haj, 2019) of the Multiling 2019 workshop (Giannakopoulos, 2019) involved the generation of structured summaries from financial narrative disclosures. The SummariserPort system (de Oliveira et al., 2002) has been used to produce summaries for financial news. It utilises lexical cohesion (Flowerdew and Mahlberg, 2009), using sentence linkage heuristics to generate the output summary. A summarisation system of financial news was proposed in (Filippova et al., 2009), generating query-based and company-tailored summaries, via unsupervised sentence ranking using simple frequency-based features.

3 Data Description

The financial narrative summarisation (FNS) shared task focuses on annual reports produced by UK firms listed on The London Stock Exchange (LSE). The produced annual reports are written in Corporate language English, which comprises the words and visuals a company uses to communicate internally and externally. This influences corporate communication as a whole, from internal messaging to web content, press releases and including annual reports, which in turn could affect the communication between the corporate and stakeholders (Dawkins, 2004).

In the UK and elsewhere, many registrants publish a glossy report containing graphics, photographs and supplementary narratives such as the letter to shareholders (Dikolli et al. 2017). These documents are

typically provided as a digital PDF file and outside the U.S. they represent the primary annual reporting vehicle. This results in barriers to large-scale automated analysis nevertheless mean that little is known about this ubiquitous reporting channel.

3.1 Data Creation

Previous work on analysing UK annual reports provided a methodological through developing, describing and evaluating an automated procedure for retrieving and classifying the narrative component of glossy annual reports presented as digital PDF files (El-Haj et al., 2020). The developed tool, CFIE-FRSE[1], has helped in creating large corpora of by-section text extracted from thousands of UK annual reports[2], which has primarily facilitated the availability and generation of a summarisation dataset that is specific to UK annual reports' financial narratives(El-Haj, 2019).

For the FNS 2020 Shared task we use around 4,000 UK annual reports for firms listed on LSE covering the period between 2002 and 2017 (El-Haj et al., 2014b; El-Haj et al., 2019a). The annual reports have been indirectly summarised by the firms' chairwoman/chairman, the chief executive officer (CEO) and the firm's management. The summaries have been used int he FNS shared task as gold-standard summaries. In addition, those summaries include the financial highlights reported by each firm at the beginning of their annual report. The summaries were extracted from the annual reports using the CFIE-FRSE tool and were then manually converted into a standard summarisation dataset through providing a document and a number of 2 to 3 gold-standard summaries.

3.2 FNS Shared Task Dataset

We divided the annual reports‘ full text into *training*, *testing* and *validation* sets providing both the full text of each annual report along with the gold-standard summaries.

In total there are 3,863 annual reports divided into training, testing and validation sets. Table 3.2 shows the dataset details.

Data Type	Training	Validation	Testing	Total
Report full text	3,000	363	500	3,863
Gold summaries	9,873	1,250	1,673	12,796

Table 1: FNS 2020 Shared Task Dataset

3.3 Data Availability

The FNS summarisation dataset was delivered to the participating teams at different stages. We provided the training and validation sets first, this included the full text of each annual report along with its gold-standard summaries. On average there are at least 3 gold-standard summaries for each annual report with some reports containing up to 7 gold-standard summaries.

The testing set was provided at a later stage so participants can test their summarisers on unseen test set. We did not provide the gold-standard summaries for the testing set.

The training, testing and validation sets all came in UTF-8 plain text (.txt) file format as shown in Section 3.4.

3.4 Data Sample

Figure 1 shows the structure of the Financial Narrative Summarisation dataset. We provided participants with two sets of directories "training" and "validation". Each contained the full text of the annual reports (*_annual_reports) and the gold standard summaries (*_gold_summaries).

The data was provided in plain text file format in a directory structure similar to the one shown in Figure 1.

Each annual report have a unique identifier (ID) which is used across the datasets in order to link annual reports' full text to their gold-standard summaries.

[1]https://github.com/drelhaj/CFIE-FRSE

[2]https://doi.org/10.17635/lancaster/researchdata/271

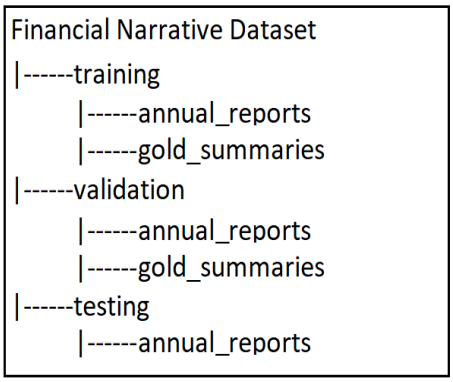

Figure 1: Dataset Structure

For example: The *training/annual_reports* directory contains a file called **19.txt** where 19 is a unique ID and can be used to locate this report's gold standard summaries in the *training_gold_summaries* directory (e.g. **19_1.txt** to **19_3.txt**.

3.5 Task Description

For the purpose of this task we asked each participating team to produce one summary for each annual report. The summary length should not exceed **1000** words[3]. Participants were advised that the summary to be generated/extracted based on the narrative sections of the annual reports, which they could do through training their summarisers to detect narrative sections before creating the summaries.

3.5.1 System Summaries Output

For the output summary we asked each team to produce a no more than 1000 words summary for each annual report in the testing set. Only one summary is allowed for each report, but participating teams are welcome to participate with more than one methodology, each methodology to be evaluated as a separate participating system. The participants were asked to follow a standard file naming process. In the future we aim to provide the human and system summaries free for research purposes. The naming pattern they were asked to follow is: **ID_summary.txt**. Example: 25082_summary.txt. For standardisation and consistency all output summary files should be in UTF-8 file format.

3.5.2 Evaluation

To evaluate the generated system summaries against the human gold-standard summaries we used the JRouge[4] package for ROUGE, using multiple variants (i.e. ROUGE-1, ROUGE-2, ROUGE-L and ROUGE-SU4) (Ganesan, 2015; Litvak et al., 2016).

4 Baseline and Topline Summarisers

All participating systems were evaluated and compared against 2 top performing (topline) systems and 2 baseline systems.

4.1 Baselines

The simplicity and the frequent use of TextRank and LexRank in literature and summarisation tasks make them ideal baselines for our shared task.

4.1.1 TextRank

Rada Mihalcea and Paul Tarau (2004) introduced TextRank as the first graph-based automated text summarisation algorithm. TextRank is a simple application of the PageRank algorithm (Brin and Page, 1998). In order to find the most relevant sentences in text, a graph is constructed where the vertices of the graph

[3]We used regex white space delimiter to detect word boundaries.
[4]https://github.com/kavgan/ROUGE-2.0

represent each sentence in a document and the edges between sentences are based on content overlap, namely by calculating the number of words that two sentences have in common.

In order to find relevant keywords, the TextRank algorithm constructs a word network. This network is constructed by looking which words follow one another. A link is set up between two words if they follow one another, the link gets a higher weight if these two words occur more frequently next to each other in the text.

Based on this network of sentences, the sentences are fed into the Pagerank algorithm which identifies the most important sentences.

4.1.2 LexRank

LexRank is another graph-based algorithm for automated text summarisation (Erkan and Radev, 2004). A cluster of documents can be viewed as a network of sentences that are related to each other. Some sentences are more similar to each other while some others may share only a little information with the rest of the sentences. Like TextRank (Section 4.1.1), LexRank too uses the PageRank algorithm for extracting top keywords. The key difference between the two baselines is the weighting function used for assigning weights to the edges of the graph. While TextRank simply assumes all weights to be unit weights and computes ranks like a typical PageRank execution, LexRank uses degrees of similarity between words and phrases and computes the centrality of the sentences to assign weights. (Erkan and Radev, 2004)

4.2 Toplines

To make the shared task more challenging, two topline summarisation algorithms have been used, MUSE and POLY.

4.2.1 MUSE

MUSE is a language-independent approach for extractive summarisation based on the linear optimisation of several sentence ranking metrics using a Genetic Algorithm (GA). We applied the original set of 31 sentence metrics[5], described in (Litvak et al., 2010). The metrics are divided into three main categories—*structure*-, *vector*-, and *graph*-based—according to the text representation model they are based on. Their best combination for the given corpus is calculated by a GA. A typical GA requires (1) a genetic representation of the solution, and (2) a fitness function to evaluate the solution quality. MUSE represents solution as a vector of weights for a linear combination of sentence metrics, and starts with the randomly initialized real values. We applied ROUGE-1 Recall (Lin, 2004) as a fitness function[6], which is maximized during the optimisation procedure. MUSE computation time is directly affected by the number of words in a summarized document. Moreover, its training time is proportional to the number of GA iterations multiplied by the number of individuals in a population times the fitness evaluation (ROUGE) time. As such, training MUSE on the entire training set of FNS 2020 dataset (3000 lengthy files, each with more than 60,000 words on average) is a very time and memory-consuming task. Therefore, we trained MUSE's model on 30 randomly selected reports from the training set and applied it on entire testing set (500 files). All files, from both training and testing sets, were pre-processed before MUSE application. Financial reports usually contain multiple sections, figures, and tables. Because the text files in the FNS-2020 dataset were obtained by converting PDF files into plain text file format, these text files contain a lot of "noise" caused by broken tables and meta-data such as section and page numbers. We cleaned the noise by measuring the ratio between text and numbers and ratio between number of words and white-spaces. Lines with low ratio were removed. Then, regular expressions were applied to find and mark such entities as URLs, phone numbers, dates, time, emails. Finally, non-Unicode characters were filtered out. MUSE is a multilingual summariser which was evaluated on multiple languages[7] and outperformed other systems in multiple MultiLing contests (Litvak and Last, 2013; Litvak et al., 2016). Therefore, it was selected as a topline system.

[5]MUSE can be configured with different number of metrics.

[6]MUSE can be configured with different ROUGE metrics.

[7]English, Hebrew, Arabic, and Persian

4.2.2 POLY

POLY (Litvak and Vanetik, 2013a) is unsupervised approach based on linear programming. POLY represent the document as a set of intersecting hyperplanes–polytope. The summary is described by the objective function–hyperplane–and is considered best if the optimal value of objective function is preserved during summarisation. As such, POLY translates the summarisation problem into a problem of finding a point on a convex polytope which is the closest to the hyperplane describing the "ideal" summary. POLY can be run with multiple objective functions describing the distance between a summary (a point on a convex polytope) and the best summary (the hyperplane). We applied POLY with Maximal Weighted Term Sum ($OBJ_1^{POS\_EQ}$ in (Litvak and Vanetik, 2013a)) objective function, which maximizes the information coverage as a term sum, with the same weight for all terms, regardless the term's frequency and position. Because POLY is unsupervised, it was directly (after pre-processing) applied on 500 files from the testing FNS set. All files were pre-processed in the same manner as for MUSE. POLY was selected as a baseline due to its polynomial run-time, no need in training, and comparatively good performance on English and other languages (Hebrew and Arabic), according to MultiLing results from 2013 (Litvak and Vanetik, 2013b) and 2015 (Vanetik and Litvak, 2015). POLY has been chosen as a competitive topline to MUSE and both have been used to assess the quality of the system summaries submitted by the participating summarisation systems.

5 Participating Teams and Systems

A total number of 9 teams participated in the FNS 2020 shared task with a total of 24 system submissions. Table 2 presents the names of the participating teams and their affiliations. In addition we report results from the topline and baseline algorithms (see Section 4).

Team	**Affiliation**
SRIB2020	Samsung
SUMTO	Politecnico di Torino
HULAT	Universidad Carlos III de Madrid
AMEX-Al Labs	American Express AI Labs, Bangalore
FORTIA	Fortia Financial Solutions
CIST@BUPT	Beijing University of Posts and Telecommunications
SUMSUM	Cornell University
KG-SUMMAR	IIT Bombay, India
SCE	Shamoon college of engineering (SCE)

Table 2: List of the 11 teams that participated in the FNS 2020 Shared Task

Table 5 summarizes the approaches adopted by each team. In the table, ML refers to any non-neural machine learning technique such as multinomial naive Bayes (MNB) and support vector machines (SVM). Neural refers to any neural network based model such as bidirectional long short-term memory (BiLSTM), or convolutional neural network (CNN). In terms of features, word and character ngram features. Language-model based features were also used a lot. A few participants used pre-trained embeddings. Table 3 shows the approaches (techniques and features) adopted by the participating teams. ML refers to any non-neural machine learning technique such as MNB, SVM, etc. Neural refers to any neural network (deep learning) based model such as BILSTM, CNN, GRUs, etc. LM refers to language-model based features. WC corresponds to word and character features. We ordered the teams by Rouge-2 F-measure. Tables 4 and 5 show the results in details for all of the 4 variations of ROUGE scores.

6 Results and Discussion

A number of 24 summarisation systems by 9 different teams have participated and submitted their system summaries to FNS 2020. In addition we report the results of the 4 topline and baseline summarisers (MUSE, POLY, TextRank and LexRank respectively) are reported in Tables 4 and 5.

		Techniques			Features		
Team	**F1**	ML	Neural	Ensemble	Structure	LM	Embeddings
SUMSUM	0.306		X			X	X
SRIB2020	0.289		X			X	X
FORTIA	0.274		X				X
HULAT	0.261	X			X		X
SUMTO	0.249		X				X
CIST@BUPT	0.248	X					
KG-SUMMAR	0.247	X	X				X
AMEX-Al Labs	0.214		X	X			
SCE	0.138				X		

Table 3: Techniques and Features used by the participating systems

The participating systems used a variety of techniques and methods ranging from rule based extraction methods and more towards high performing deep learning models and word embeddings. In addition the participating teams used methods to investigate the hierarchy of the annual reports to try and detect the structure of the report and extract the narrative sections. The majority of the applied techniques were extractive applying methods such as Determinantal Point Processes (DPPs) sampling algorithm and a combination of Pointer Network and T-5 (Test-to-text transfer Transformer) algorithms. Other extractive summarisers used word embeddings such word2vec, BERT and using CBOW & skip grams. An end-to-end training method using Deep NLP techniques, and a hierarchical summary that visualises as a tree with summaries under different discourse topics and an ensemble based model have also been reported.

This variety of techniques shows the interdisciplinary notion of FNS 2020 and the fact that it attracted such a range of methods, making it a gateway for researchers and practitioners working on summarising lengthy annual reports.

Some of the challenges and limitations reported by the participants is the fact that the average length of those annual reports is 60,000 words, this makes the training process difficult in term of time and performance, which is a problem that we are aware of and it is what prompted us to introduce such a challenging task. In addition, participants explained that it is difficult to detect structure of such reports due to the fact that they come originally in PDF format and extracting information from such files results in a lot of noise. This is a problem that we have been working on since 2012 and though we understand it is challenging, we believe it opens up an interesting research problem that is worth investigating more in the future.

Tables 4 and 5 show the results of the participating systems using ROUGE-1, ROUGE-2, ROUGE-L and ROUGE-SU4 respectively. The results have been sorted in descending order according to the highest F-score. The topline (MUSE and POLY) and baseline (TextRank and LexRank) systems are highlighted in bold font. The results show that the majority of the participating systems produced results that are better on average than our two baselines and POLY topline. On the other hand, our topline MUSE results show a challenging notion making it hard to beat, but even though we are happy to see that many participating systems have managed to produce results that are significantly better than MUSE. Such results will be used as a comparison line in the future through creating a venue of results and techniques for researchers working on financial text summarisation.

System / Metric	R-1 / R	R-1 / P	R-1 / F	R-2 / R	R-2 / P	R-2 / F
SRIB2020-3	0.61	0.39	0.47	0.45	0.22	0.29
SRIB2020-2	0.61	0.39	0.47	0.45	0.22	0.29
SUMSUM-BASE	0.49	0.48	0.46	0.40	0.26	0.29
SUMSUM-BERT	0.45	0.53	0.46	0.37	0.30	0.31
KG-SUMMAR-NN	0.57	0.38	0.44	0.40	0.18	0.25
SUMSUM-1	0.45	0.51	0.44	0.36	0.28	0.29
HULAT-1	0.54	0.39	0.44	0.41	0.20	0.26
KG-SUMMAR-SVM	0.49	0.42	0.44	0.36	0.20	0.25
KG-SUMMAR-S-LSTM	0.51	0.41	0.44	0.36	0.19	0.24
MUSE	0.48	0.41	0.43	0.31	0.20	0.23
CIST-BUPT-3	0.43	0.45	0.43	0.29	0.23	0.25
SUMTO-3	0.45	0.43	0.42	0.30	0.23	0.25
SUMTO-2	0.44	0.43	0.42	0.28	0.22	0.23
SUMTO-1	0.43	0.44	0.42	0.27	0.23	0.24
CIST-BUPT-2	0.42	0.44	0.42	0.27	0.22	0.24
AMEX-ENSEMBLE	0.44	0.41	0.41	0.26	0.19	0.21
AMEX-BILSTM	0.44	0.41	0.41	0.26	0.19	0.21
FORTIA-1	0.43	0.43	0.41	0.30	0.28	0.27
HULAT-2	0.50	0.35	0.40	0.37	0.18	0.23
CIST-BUPT-1	0.40	0.42	0.40	0.26	0.21	0.22
FORTIA-2	0.39	0.41	0.38	0.25	0.26	0.24
FORTIA-3	0.37	0.37	0.35	0.21	0.22	0.20
SCE	0.29	0.40	0.30	0.16	0.16	0.14
AMEX-TEXTRANK	0.35	0.27	0.29	0.18	0.10	0.12
SRIB2020-1	0.24	0.38	0.28	0.11	0.14	0.12
POLY	0.32	0.25	0.27	0.15	0.09	0.11
LEXRANK	0.34	0.27	0.26	0.19	0.11	0.12
TEXTRANK	0.41	0.12	0.17	0.23	0.04	0.07

Table 4: ROUGE-1 and ROUGE-2 Recall, Precision and F-measure scores

System / Metric	R-L / R	R-L / P	R-L / F	R-SU4 / R	R-SU4 / P	R-SU4 / F
SRIB2020-3	0.61	0.38	0.46	0.51	0.21	0.29
SRIB2020-2	0.60	0.38	0.46	0.51	0.21	0.29
MUSE	0.47	0.37	0.41	0.37	0.20	0.25
SUMTO-3	0.41	0.40	0.39	0.35	0.22	0.26
SUMTO-1	0.41	0.39	0.39	0.33	0.22	0.25
HULAT-1	0.44	0.36	0.39	0.46	0.19	0.26
SUMTO-2	0.40	0.38	0.38	0.34	0.21	0.25
FORTIA-1	0.40	0.40	0.38	0.34	0.33	0.32
AMEX-ENSEMBLE	0.41	0.37	0.38	0.33	0.19	0.24
AMEX-BILSTM	0.40	0.36	0.37	0.32	0.19	0.23
HULAT-2	0.39	0.36	0.36	0.43	0.17	0.24
FORTIA-2	0.37	0.37	0.36	0.30	0.31	0.29
FORTIA-3	0.34	0.34	0.33	0.26	0.27	0.25
CIST-BUPT-3	0.32	0.35	0.33	0.35	0.21	0.25
SUMSUM-BASE	0.33	0.35	0.32	0.44	0.24	0.29
CIST-BUPT-2	0.31	0.35	0.32	0.33	0.20	0.24
SUMSUM-BERT	0.30	0.39	0.32	0.41	0.27	0.30
KG-SUMMAR-NN	0.39	0.28	0.32	0.46	0.17	0.24
KG-SUMMAR-S-LSTM	0.34	0.31	0.32	0.42	0.18	0.25
CIST-BUPT-1	0.29	0.36	0.32	0.32	0.19	0.23
SUMSUM-1	0.30	0.38	0.31	0.40	0.25	0.28
KG-SUMMAR-SVM	0.34	0.30	0.31	0.41	0.19	0.25
AMEX-TEXTRANK	0.25	0.24	0.24	0.25	0.11	0.14
SRIB2020-1	0.21	0.25	0.22	0.17	0.15	0.15
SCE	0.22	0.29	0.22	0.21	0.16	0.16
LEXRANK	0.21	0.26	0.22	0.25	0.12	0.14
TEXTRANK	0.24	0.20	0.21	0.30	0.05	0.08
POLY	0.26	0.18	0.20	0.21	0.11	0.13

Table 5: ROUGE-L and ROUGE-SU4 Recall, Precision and F-measure scores

References

Abdelkrime Aries, Djamel Eddine Zegour, and Khaled Walid Hidouci. 2015. Allsummarizer system at multiling 2015: Multilingual single and multi-document summarization. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 237–244. The Association for Computer Linguistics.

Vasiliki Athanasakou, Mahmoud El-Haj, Paul Rayson, Martin Walker, and Steven Young. 2019. Annual report management commentary articulating strategy and business model: Measurement and impact.

Phyllis B Baxendale. 1958. Machine-made index for technical literature—an experiment. *IBM Journal of research and development*, 2(4):354–361.

Sergey Brin and Lawrence Page. 1998. The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*.

Eddy Cardinaels, Stephan Hollander, and Brian J White. 2019. Automatic summarization of earnings releases: attributes and effects on investors' judgments. *Review of Accounting Studies*, 24(3):860–890.

Jenny Dawkins. 2004. Corporate responsibility: The communication challenge. *Journal of communication management*.

Paulo Cesar Fernandes de Oliveira, Khurshid Ahmad, and Lee Gillam. 2002. A financial news summarization system based on lexical cohesion. In *Proceedings of the International Conference on Terminology and Knowledge Engineering, Nancy, France*.

Harold P Edmundson. 1969. New methods in automatic extracting. *Journal of the ACM (JACM)*, 16(2):264–285.

Mahmoud El-Haj, Vasiliki Athanasakou, Paul Rayson, Steven Young, and Martin Walker. 2014a. Computer-based analysis of the strategic content of uk annual report narratives. In *2014 American Accounting Association Annual Meeting Global Engagement and Perspectives*.

Mahmoud El-Haj, Paul Rayson, Steven Young, and Martin Walker. 2014b. Detecting document structure in a very large corpus of uk financial reports. *European Language Resources Association (ELRA)*.

Mahmoud El-Haj, Paul Edward Rayson, Steven Eric Young, Martin Walker, Andrew Moore, Vasiliki Athanasakou, and Thomas Schleicher. 2016. Learning tone and attribution for financial text mining. In *The 10th edition of the Language Resources and Evaluation Conference (LREC'16). Portoroz, Slovenia.* European Language Resources Association (ELRA).

Mahmoud El-Haj, Paul Rayson, and Andrew Moore. 2018a. The first financial narrative processing workshop (fnp 2018). *LREC 2018*.

Mahmoud El Haj, Paul Edward Rayson, Paulo Alves, and Steven Eric Young. 2018b. Towards a multilingual financial narrative processing system. In *FNP 2018 Workshot at the 11th edition of the Language Resources and Evaluation Conference (LREC'18)*.

Mahmoud El-Haj, Paul Rayson, Paulo Alves, Carlos Herrero-Zorita, and Steven Young. 2019a. Multilingual financial narrative processing: Analysing annual reports in english, spanish and portuguese. *World Scientific Publishing*.

Mahmoud El-Haj, Paul Rayson, Martin Walker, Steven Young, and Vasiliki Simaki. 2019b. In search of meaning: Lessons, resources and next steps for computational analysis of financial discourse. *Journal of Business Finance & Accounting*, 46(3-4):265–306.

Mahmoud El-Haj, Paul Rayson, Steve Young, Houda Bouamor, and Sira Ferradans. 2019c. Proceedings of the second financial narrative processing workshop (fnp 2019). In *Proceedings of the Second Financial Narrative Processing Workshop (FNP 2019)*.

Mahmoud El-Haj, Paulo Alves, Paul Rayson, Martin Walker, and Steven Young. 2020. Retrieving, classifying and analysing narrative commentary in unstructured (glossy) annual reports published as pdf files. *Accounting and Business Research*, 50(1):6–34.

Mahmoud El-Haj. 2012. *Multi-document Arabic Text Summarisation*. Ph.D. thesis, University of Essex.

Mahmoud El-Haj. 2019. MultiLing 2019: Financial narrative summarisation. In *Proceedings of the Workshop MultiLing 2019: Summarization Across Languages, Genres and Sources*, pages 6–10, Varna, Bulgaria, September. RANLP.

Günes Erkan and Dragomir R Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of artificial intelligence research*, 22:457–479.

Katja Filippova, Mihai Surdeanu, Massimiliano Ciaramita, and Hugo Zaragoza. 2009. Company-oriented extractive summarization of financial news. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 246–254.

John Flowerdew and Michaela Mahlberg. 2009. *Lexical cohesion and corpus linguistics*, volume 17. John Benjamins Publishing.

Pascale Fung and Grace Ngai. 2006. One story, one flow: Hidden markov story models for multilingual multidocument summarization. *ACM Transactions on Speech and Language Processing (TSLP)*, 3(2):1–16.

Kavita Ganesan. 2015. Rouge 2.0: Updated and improved measures for evaluation of summarization tasks. *arXiv preprint arXiv:1803.01937*.

George Giannakopoulos, Vangelis Karkaletsis, George Vouros, and Panagiotis Stamatopoulos. 2008. Summarization system evaluation revisited: N-gram graphs. *ACM Transactions on Speech and Language Processing (TSLP)*, 5(3):1–39.

George Giannakopoulos. 2019. Proceedings of the workshop multiling 2019: Summarization across languages, genres and sources. In *Proceedings of the Workshop MultiLing 2019: Summarization Across Languages, Genres and Sources*.

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

Rim Koulali, Mahmoud El-Haj, and Abdelouafi Meziane. 2013. Arabic topic detection using automatic text summarisation. In *2013 ACS International Conference on Computer Systems and Applications (AICCSA)*, pages 1–4. IEEE.

Canasai Kruengkrai and Chuleerat Jaruskulchai. 2003. Generic text summarization using local and global properties of sentences. In *Proceedings IEEE/WIC International Conference on Web Intelligence (WI 2003)*, pages 201–206. IEEE.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Marina Litvak and Mark Last. 2013. Multilingual single-document summarization with muse. In *Proceedings of the MultiLing 2013 Workshop on Multilingual Multi-document Summarization*, pages 77–81.

Marina Litvak and Natalia Vanetik. 2013a. Mining the gaps: Towards polynomial summarization. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 655–660.

Marina Litvak and Natalia Vanetik. 2013b. Multilingual multi-document summarization with poly2. In *Proceedings of the MultiLing 2013 Workshop on Multilingual Multi-document Summarization*, pages 45–49.

Marina Litvak, Mark Last, and Menahem Friedman. 2010. A new approach to improving multilingual summarization using a genetic algorithm. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 927–936.

Marina Litvak, Natalia Vanetik, Mark Last, and Elena Churkin. 2016. Museec: A multilingual text summarization tool. In *Proceedings of ACL-2016 System Demonstrations*, pages 73–78.

Shuhua Liu and Johnny Lindroos. 2006. Experiences from automatic summarization of imf staff reports. *Practical Data Mining: Applications, Experiences and Challenges*, page 43.

Yang Liu. 2019. Fine-tune bert for extractive summarization. *arXiv preprint arXiv:1903.10318*.

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

Victoria McCargar. 2004. Statistical approaches to automatic text summarization. *Bulletin of the American Society for Information Science and Technology*, 30(4):21–25.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 404–411.

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

Tatsunori Mori. 2002. Information gain ratio as term weight: the case of summarization of ir results. In *COLING 2002: The 19th International Conference on Computational Linguistics*.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: a recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 3075–3081.

Ani Nenkova and Kathleen McKeown. 2011. *Automatic summarization*. Now Publishers Inc.

Dragomir R Radev, Hongyan Jing, and Malgorzata Budzikowska. 2000. Centroid-based summarization of multiple documents: Clustering, sentence extraction, and evaluation. In *Proceedings of the ANLP/NAACL-2000 Workshop on Summarization*.

Catherine Salzedo, Steven Young, and Mahmoud El-Haj. 2014. Does equity analyst research lack rigor and objectivity? evidence from conference call questions and research notes. In *Evidence from Conference Call Questions and Research Notes (May 5, 2014)*.

Natalia Vanetik and Marina Litvak. 2015. Multilingual summarization with polytope model. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 227–231.