FinAraT5: A text to text model for financial Arabic text understanding and generation.

Nadhem Zmandar and **Mahmoud El-Haj** and **Paul Rayson**
UCREL NLP group,
School of Computing and Communications,
Lancaster University, UK

Abstract

The financial industry generates a significant amount of multilingual data, and there is a pressing need for better multilingual NLP models for tasks such as summarisation, structure detection, and causal detection in the financial domain. However, there are currently no pre-trained finance-specific Arabic language models available. To address this need, we continue the pre-training of AraT5 to create FinAraT5, the first pre-trained Arabic language model specifically designed for financial use cases, trained on a large Arabic financial communication corpus consisting of annual and quarterly reports and press releases. We hypothesise that FinAraT5 would perform better than AraT5 on financial domain tasks. We demonstrate this through research on a publicly available discriminative task (translated from English), and a generative task from a novel summarisation dataset called FinAraSum. Our results show FinAraT5 is highly competitive with state-of-the-art models such as mT5, AraBART, BERT, and the original AraT5 on Arabic language understanding and generation tasks.

1 Introduction

Pre-trained language models are a hot topic in Natural Language Processing. Despite their success, most are trained on English or multilingual datasets. Leveraging the vast amount of unlabeled data available online, they provide an efficient way to pre-train continuous word representations that can be fine-tuned for a downstream task, along with their contextualization at the sentence level. Generally, pre-trained models are trained on massive corpora using GPUs or recently TPUs. Most follow the architecture proposed by (Vaswani et al., 2017). Sequence-to-sequence is the best architecture for abstractive models, and abstractive models are very efficient for news summarisation and text paraphrasing. Unlike extractive summarisation, abstractive approaches are not restricted to the input words (Rush et al., 2015; Chopra et al., 2016).

Arabic is a very rich language with few resources, and significantly fewer language models compared to English and other Latin languages. Arabic remains understudied in the Natural Language Processing (NLP) community. In addition, Arabic NLP and generation tasks have proven to be very challenging to tackle. Most Arabic language models are mainly encoder only and are not field-specific (Antoun et al., 2020).

The middle eastern stock exchanges have an increasing market cap motivated by oil and gas companies, real estate companies and especially investment companies (e.g. kingdom holding). Therefore, the middle eastern markets are gaining in popularity among western investors, especially with the evolution of jurisdiction in the UAE through the free trade zone and the flexibility of investment in a Gulf-listed company. In addition, the Tadawul Saudi Exchange is the ninth most significant stock market among the 67 members of the World Federation of Exchanges by market capitalization of listed companies (approximately US$2.6 trillion on 30 June 2021) and is the dominant market in the Gulf Cooperation Council (GCC). The successful IPO of Saudi Aramco demonstrates this. Tadawul is also included in the MSCI, FTSE Russell and S&P Emerging Market indices. It is the third largest stock market amongst its emerging market peers. It is an affiliate member of the International Organization of Securities Commissions (IOSCO), the World Federation of Exchanges (WFE), and the Arab Federation of Exchanges (AFE). These facts point to the increasing importance and scale of textual financial data in Arabic, which needs to be followed by an advance in Arabic NLP covering finance and investment-related tasks. Therefore, we propose the training of a monolingual Arabic T5 model customized for financial corpora.

We present FinAraT5, based on araT5, as a continuation of pre-training of araT5 on a large collected monolingual financial Arabic corpus. Unlike previously released Arabic BERT different versions, FinAraT5 is adapted for both generative and discriminative tasks. We evaluate the pre-trained model on financial sentiment analysis and financial news summarisation on a novel Arabic news summarisation dataset, FinAraSum, that we collected ourselves. This work aims to meet the need for a monolingual financial text-to-text model for the Arabic language since no previous public model existed. One issue with the past work targeting Arabic abstractive summarisation is the evaluation of such models on highly extractive datasets. The primary available Arabic extractive datasets are ANT Corpus (Chouigui et al., 2017) and KALIMAT (El-Haj and Koulali, 2013). Therefore in this study, we prepare our customized highly abstractive financial summarisation dataset to suit the financial model we created. Our contributions through this research paper are summarised as follows:

- We present the first pre-trained Arabic text-to-text financial language model pre-trained on financial narratives Arabic corpus. The model features 220 Million parameters and is trained on 25 GB of PDF text for 45 days using a google cloud TPU V3.8. The model is suitable for generative and discriminative tasks.

- We describe the steps to collect, convert, preprocess and clean a financial narratives corpus covering different middle eastern stock exchanges.

- We present the collection and creation of FinAraSum, a highly abstractive financial and economic news dataset which are an Arabic equivalent of OrangeSum (Kamal Eddine et al., 2021) and Xsum (Narayan et al., 2018)

- We evaluate FinAraT5 on discriminative and generative tasks and show that it produces promising results.

- We compare FinAraT5 with different versions of multilingual T5 to prove the importance of training monolingual language models.

- We show that FinAraT5 achieves state-of-the-art results on the small Arabic benchmark we created. It outperforms Bert based model, multilingual text-to-text models and some general-purpose Arabic models.

- All our models are integrated into a hugging face repository to facilitate replicability and reuse.

2 Background and Related work

2.1 T5 transformer

The T5 (Raffel et al., 2019a) text-to-text transformer is a sequence-to-sequence (encoder-decoder) language model pre-trained on a multi-task mixture of unsupervised and supervised tasks for which each task is converted into a text-to-text format. T5 works well on various tasks by prepending a different prefix to the input corresponding to each task (e.g., for translation: translate English to German; for summarisation: summarize:). It is configured for 4096 maximum input tokens. However, the model is based on relative position embeddings, which allows it to scale to longer input sequences. Because of the complexity O(n2) of the Transformer's self-attention mechanism, such scaling increases memory consumption exponentially. The idea of a unified Transformer framework for different tasks was introduced by (Raffel et al., 2019a). The T5 framework treats all generative and discriminative tasks as a text-to-text problem. This enabled a more efficient transfer learning approach. In addition, Google researchers recently extended the T5 model to multilingualism by releasing mT5 (Xue et al., 2021), a multilingual version of T5. In this work, we will also test the portability of mT5 to the Arabic language and explore its performance on Arabic financial tasks, for the first time.

Several models trained for seq2seq models were previously released. Seq2seq models connect the left encoder and the right decoder part of the transformer with attention to enable the model to produce output. A Seq2Seq model achieves this by using the following scheme: Input tokens-> embeddings-> encoder-> decoder-> output tokens. Among the commonly used seq2seq models is the BART model, which was pre-trained on several languages such as French (Kamal Eddine et al., 2021) and English (Lewis et al., 2020). In addition, there is a multilingual version of BART (Liu et al., 2020).

2.2 Arabic Pre-trained Language models

Since the emergence of transformer models, a number of Arabic LMs has been developed. **AraBERT** (Antoun et al., 2020) was trained with the same architecture as BERT (Vaswani et al.,

2017) and used the BERT Base configuration. AraBERT is trained on 23GB of Arabic text, making approximately 70M sentences and 3B words from Arabic Wikipedia, the Open Source International dataset (OSIAN) (Zeroual et al., 2019), and (El-Khair, 2016) Corpus (1.5B words). Antoun et al. compared the performance of AraBERT to multilingual BERT from Google and other state-of-the-art models. The results prove that araBERT achieves state-of-the-art performance on most tested Arabic NLP tasks.
ARBERT (Abdul-Mageed et al., 2021) is a large-scale pre-trained masked language model for Modern Standard Arabic. To train ARBERT, Abdul-Mageed et al. used the same architecture as BERT Base: 12 attention layers. It has approximately 163M parameters and was trained on a 61GB collection of Arabic datasets.
AraBART (Kamal Eddine et al., 2022b) is the first Arabic sequence-to-sequence model where the encoder and the decoder are trained end-to-end. It is based on BART. AraBART follows the architecture of BART Base which has 6 encoder and 6 decoder layers and 768 hidden dimensions. AraBART has 139M parameters and achieved state of art results on multiple abstractive summarisation datasets.
araT5 (Nagoudi et al., 2022) created the first Arabic text to text model (araT5). They released three powerful Arabic text-to-text Transformer versions. For evaluation, they used an existing benchmark for Arabic language understanding and introduced a new benchmark for Arabic language generation (ARGEN).
JABER and SABER: Junior and Senior Arabic BERT (Ghaddar et al., 2021) found that most of the released Arabic BERT models were under-trained and therefore developed JABER and SABER, Junior and Senior Arabic BERT models. Experimental results show that their models achieve state-of-the-art performances on ALUE, a new benchmark for Arabic Language Understanding Evaluation.

2.3 Financial pre-trained language models:

Finbert: is the first BERT model pre-trained on financial narrative text. It is trained on a 4.9B tokens corpus composed of Corporate Reports 10-K and 10-Q (2.5B tokens), Earnings Call Transcripts (1.3B tokens), and Analyst Reports (1.1B tokens). Finbert is fine-tuned for three use cases: a sentiment classification task, ESG classification task and forward-looking statement (FLS) FinBert. Their fine-tuned FinBERT models are available on Huggingface's transformers library[1]. This model achieves superior performance on financial sentiment classification tasks. (Yang et al., 2020)

3 Training Corpus Description

Training a transformer model needs a large corpus in plain text because of the large number of parameters in the model's architecture. There is no available public financial corpus covering financial statements in Arabic. Hence, we also created the training corpus ourselves. We aggregated two corpora of different orders of magnitude to train the models.

3.1 Financial Reports

In this section, we describe in detail our approach to collecting large-scale financial text in Arabic. The task is challenging, as financial reports are not readily available or centralised in one location.

Data Acquisition We collected several types of financial documents from different middle eastern markets: auditor reports, earning announcements, accounting documents, quarterly reports (Q1, Q2, Q3, Q4), annual reports and management board reports. A total of 30,000 PDF files were collected to form our source data. The total size of PDF files collected is around 25Gb.

We focused on major stock exchanges in the middle east to collect our corpus. Our data is collected from the following Arab markets: KSA exchange: TASI (Tadawul All Share Index) and NOMU (Saudi Parallel Market Growth parallel market), UAE (Dubai Financial Market (DFM), Abu Dhabi index), Kuwait (Boursa Kuwait), Oman (Muscat Stock Exchange), Qatar (Qatar Stock exchange) and Bahrain (Bahrain stock exchange).

The corpus is constituted as a diverse set of documents from different sectors and covers several categories. We have more than 35 categories in this corpus (E.g. financial services, Banking, insurance, telecommunication, oil and gas, energy, real estate, and utilities). We did not include the Egyptian financial disclosures since their data was not freely available. For other North African markets,

[1] https://huggingface.co/yiyanghkust/finbert-tone

such as Morocco and Tunisia, companies communicate mainly in French rather than Arabic.

Table 1 describes the corpus in detail by providing summary statistics about the different indexes used in this corpus.

3.2 PDF to Text process

A significant constraint is the nature of the documents which are scanned PDF, contain old Arabic fonts or a lot of noise. In addition, the use of Arabic numerals and a lot of tabular data made the task of converting to text files very complex.

We selected the pro version of the sejda app, but firstly used a PDF2Text algorithm to convert our PDF reports to plain text files. If the conversion did not work, we used their Arabic OCR solution. The Arabic OCR inverts the order of words from left to right, hence this has to be corrected. Among the 30,000 collected reports, 24,000 were used in the process. We passed them through a PDF-to-text script in several batches. Converting as PDF2text worked very well for many reports. The success rate was more than 40%. Some scanned docs were converted but generated ASCII code files, meaning the conversion script cannot detect the content.

For the others, we used the OCR tool of sejda[2]. On average, 10 PDF files took around one hour to be OCRed. The OCR operation took more than eight days in total, including the post-processing. Although the OCR solution of sejda is less efficient than we would like, it has an acceptable success rate given the poor quality of the report files. Finally, we performed a manual check to verify that all the files had the minimum required Arabic structure for our pre-training process. We manually deleted all the badly converted files. Further significant challenges during the data construction and data conversion process include the following aspects.

PDF2Text One of the common issues we observed from applying OCR on Arabic-written PDF files were repeated characters or additional spaces between the characters of one word (all the words are written with spaces) or concatenated words (not separated by spaces). This is reported to be a common issue for OCR in Arabic, especially if the quality of data is not good.

Memory Management Producing such a large-scale corpus is very time-consuming; hence we divided the whole task into small tasks. It took around three months to construct the corpus, from web scraping until the last cleaned and pre-processed files are used in training.

OCR Low success rate for Arabic and especially a very long processing time given there was no possibility for parallel execution.

3.3 Newswires

In addition to our financial and board reports corpus, we selected more than 30,000 financial and economic news items from a leading news Arabic website. This helps to make our training corpus more diverse and enables coverage of several topics and styles of writing. All the corpus text is written in Modern Standard Arabic.

3.4 Cleaning

Once converted from PDF to text, we cleaned the text in order to be ready for the training. We used farasa[3] for segmentation. We read files in chunks and applied our cleaning pipeline. This process started by removing all diacritics, HTML elements and their attributes, all special characters, and English alphabets and digits. We also removed tatweel characters, which are used regularly in Arabic writing. We reduced repeated characters to single characters, removed links and long words (longer than 15 chars). We used (Alyafeai and Saeed, 2020) to prepare our cleaning and preprocessing pipeline.

4 FinAraT5: Our financial text-to-text model

FinAraT5 is the first financial Arabic language model designed for text generation and text understanding. It is trained using a text-to-text approach. Our model is based on araT5 (Nagoudi et al., 2022), a pre-trained Arabic text-to-text model. It is the first financial Arabic model pre-trained in an encoder-decoder manner.

4.1 Architecture

We use the BASE architecture of T5 encoder-decoder (Raffel et al., 2019a), with 12 encoder layers and 12 decoder layers. Both the encoder and decoder have 12 attention heads and 768 hidden units. In total, therefore, FinAraT5 Base is an encoder-decoder with 220M parameters.

4.2 Vocabulary

Because we are continuing the pre-training of araT5, we opted for using the same vocabulary

[2] https://www.sejda.com/ocr-pdf

[3] https://farasa.qcri.org/segmentation/

Index	Tasi	Nomu	Dubai	AD	Kuwait	Qatar	Oman	Bahrain
# companies	223		178	73	163	47	111	42
MKT cap	3158.57		294.83		105.98	165.39	13.00	24.60
Time range	2003-2021		2009-2021		2012-2021	2010-2021	2015-2021	2014-2021
# reports	19651		3338		3192	2454	23	536
# sectors	21		11		13	7	2	6

Table 1: Statistics for the financial pre-training corpus. This table shows correct figures as at July 2022 from different sources such as statista.com. The columns represent the different indexes used. The rows describe the number of listed companies included in the report, market caps in US billion dollars, time range of the corpus, number of reports collected and the number of sectors included in the corpus. AD stands for Abu Dhabi stock exchange.

model used to train araT5 by Nagoudi et al., which was created using SentencePiece (Kudo and Richardson, 2018) which encodes text as WordPiece tokens (Bostrom and Durrett, 2020) with 110K WordPieces. Hence, our vocabulary model has a size of 110,000.

4.3 Training details

Pre-Training: We pre-train FinAraT5 on a TPU V-3.8 (with 8 cores) offered by Google cloud, with a learning rate of 0.001. We used the Adam optimizer (Kingma and Ba, 2014) and fix the batch size to 100,000 tokens. We set the maximum input and target sequence length to 512 sequences. We continued the training of the araT5 MSA base for additional 500,000 steps. We started from step 1 million, where the arat5 was stopped. In total, we pre-train FinAraT5 for 1.5 million steps[4]. The pre-training took around 40 days on the google cloud platform.

Pre-training TASK T5 was pre-trained on a mixture of supervised (mask language modelling) and unsupervised tasks. AraT5 was pre-trained using an unsupervised task. Therefore we use the same pre-training strategy as araT5, which is an unsupervised learning task trained on a raw plain text of financial qualitative data in Arabic. We cloned the architecture of T5 directly from the T5 repository[5]. We defined the task and performed the training using the t5 library[6], which enables us to perform the training using Tensorflow and get a Mesh TensorFlow Transformer.

5 FinAraBen: Financial Arabic benchmark

To evaluate any pre-trained models, we need to compare them against a benchmark task. Unfortunately, there are no public financial datasets in Arabic that could be used in this study. In fact, in the case of Arabic finance texts, labelled datasets are very scarce resources. Thus, we created a new benchmark for the financial Arabic language called FinAraBen which includes two datasets: financial text summarisation and financial sentiment analysis. The first was collected, cleaned and created by ourselves. The second was translated from a previously released dataset in English.

5.1 FinAraSum dataset

The FinAraSum dataset was inspired by the XSum dataset and OrangeSum dataset. It was created by scraping the “Arabyia asswak” website[7]. Alarabya is a large Saudi information media with 21.0M visitors per month. It publishes in Arabic and English, covering the MENA region. We decided to create our own Arabic financial news dataset to solve the issue of the need for more open sources of NLP datasets. The choice was to create a dataset adapted to abstractive summarisation, which is news headline generation. This enables testing the efficiency of the pretrained model by testing the generative component of the model, which is itself a challenging task in NLP.

Motivation: We followed the collection procedure described by (Narayan et al., 2018) and (Kamal Eddine et al., 2021) who presented Xsum and OrangeSum respectively, which are highly abstractive datasets. We present the financial Arabic version of Xsum, which is more abstractive.

[4] We note that the English T5Base (Raffel et al., 2019b) was trained only for 512K steps

[5] https://github.com/google-research/text-to-text-transfer-transformer

[6] https://pypi.org/project/t5/

[7] https://www.alarabiya.net/aswaq

Collection Process: We collected the newswires from “Al Arabyia Asswak” website[8]. The choice of this news source is motivated by the fact that it is the largest news website in the middle east, with 21M monthly visitors. Alarabya has specialized financial and economic journalists writing several articles daily covering the region’s financial news. They mainly use Modern Standard Arabic. The collected dataset covers seven categories: financial markets, economics, real estate, energy, economy, tourism and special stories. We collected all the available news articles covering a decade from 2012 to 2021.

Statistics about the FinAraSum: Table 2 compares FinAraSum with the previously released dataset such as CNN, DailyMail, NY Times, OrangeSum and XSum datasets. Our dataset is smaller than Xsum, CNN, NYT, and Daily Mail but larger than the OrangeSum title and OrangeSum abstract. Table 2 shows that our dataset comprises 44,900 newswires in the training split. The article body and the title are 238.3 and 9 words in length on average, respectively. The dataset was very clean and did not require any specific post-processing. Table 3 shows that our dataset is more abstractive than the previously released one, making it a very challenging task for our financial pretrained model. There are 37.8% novel unigrams in the FinAraSum Gold summaries, compared with 35.76% in Xsum, 26.54% in OrangeSum title, 30.03% in OrangeSum Abstract, 16.75% in CNN, 17.03% in DailyMail, and 22.64% in NY Times. Similar results are reported for Bigrams, Trigrams and 4-grams. This proves that FinAraSum is more abstractive than previously released datasets.

Split FinArasum train/val/testing We randomly split the dataset into train, validation, and test splits. The test set is composed of 2,500 news articles. The validation is composed of 1,500, with the remainder for training.

5.2 Financial Sentiment Analysis Dataset

Currently, to the best of our knowledge, there are no available financial sentiment analysis corpora in the Arabic language. For our experiments, we used the FinancialPhrase dataset[9]. The dataset was collected by (Malo et al., 2013). This release of the financial phrase bank covers a collection of 4,840 sentences. The selected collection of phrases was annotated by 16 people with adequate background knowledge of financial markets. We used sentences with more than 50 per cent agreement. To pre-process the classification dataset, we separated it into inputs and labels. The inputs are financial-related sentences, and the labels are sentiments (positive, neutral, negative). Then we encoded our labels as follows ‘positive’: 0, ‘neutral’:1, ‘negative’:2. We then split our dataset into training (80%) and testing (20%), and we ensured that our split respected a normal distribution of our labels. The training and testing datasets’ length are 3,876 and 970, respectively.

6 Experiments and Results

6.1 Financial Text Summarisation

The task of headline generation was addressed several times in past summarisation challenges, such as the Document Understanding Conferences (DUC) for 2002, 2003 and 2004.

Technical decision Usually, the summarisation script would set the loss function as the rouge score. In this study, we changed the loss function to the Bert score using the multilingual BERT checkpoint. Therefore, we could monitor the evolution of the Bertscore loss function in real time on the training and validation split using the Weights and Biases AI tool[10]. In addition, we used early stopping and took the best checkpoint on the validation split. We use the multilingual version of the BERT language model. This choice is justified by the highly abstractive nature of our dataset. Before this decision, we tried to train our models by minimizing the loss function of rouge and Bleu scores. However, Bertscore was the best choice and performed very well on the validation dataset. We used the original implementation of BertScore[11]. Bertscore calculates the similarity of the contextual embeddings of the system and reference summaries. We set our evaluation process to be executed at every step. For this work, we trained mT5 small, base, and large. We were unable to train the mT5 Xlarge due to memory limitations. We also trained arat5 small, arat5 base, araBart large and bert2bert base. For BERT2BERT, we followed the methodology proposed by Rothe et al.. We created a sequence-to-sequence model whose encoder

[8] https://www.alarabiya.net/aswaq

[9] https://huggingface.co/datasets/financial_phrasebank

[10] https://wandb.ai

[11] We use the official implementation https://github.com/Tiiiger/bert_score

Dataset	Train/Val/Test	Avg Doc Length		Avg Summary length		Vocab Size	
		words	Sentence	words	Sentence	Docs	Sum
CNN	90.3/1.22/1.09	760.50	33.98	45.70	3.58	34	89
Daily mail	197/12.15/10.40	653.33	29.33	54.65	3.86	564	180
NYT	590/32.73/32.73	800.04	35.55	45.54	2.44	1233	293
Xsum	204/11.33/11.33	431.07	19.77	23.26	1.00	399	81
Orangesum title	30.6/1.5/1.5	315.31	10.87	11.42	1.00	483	43
Orangesum Abstract	21.4/1.5/1.5	350	12.06	32.12	1.43	420	71
FinAraSum(ours)	44.90/1.5/2.5	238.3	10.15	9.0	1.0	492	46

Table 2: Sizes (column 2) are given in thousands of documents. Document and summary lengths are in words. Vocab sizes are in thousands of tokens as reported in (Kamal Eddine et al., 2021)

Dataset	% of novel n-grams in gold summary				LEAD		
	Unigrams	Bigrams	Trigrams	4-grams	R-1	R-2	R-L
CNN	16.75	54.33	72.42	80.37	29.15	11.13	25.95
Daily mail	17.03	53.78	72.14	80.28	40.68	18.36	37.25
NYT	22.64	55.59	71.93	80.16	31.85	15.86	23.75
Xsum	35.76	83.45	95.50	98.49	16.30	1.61	11.95
Orangesum title	26.54	66.70	84.18	91.12	19.84	08.11	16.13
Orangesum Abstract	30.03	67.15	81.94	88.3	22.21	07.00	15.48
FinAraSum(ours)	37.8	73.6	89.0	95.2	18.30	07.5	14.79

Table 3: Degree of abstractivity of FinAraSum compared with that of other datasets, as reported in (Narayan et al., 2018) and (Kamal Eddine et al., 2021). It can be observed that FinAraSum is more abstractive than XSum and OrangeSum and traditional summarisation datasets.

and decoder parameters are multilingual uncased Bert base model[12]. We will oblige the mbert model to work as an encoder and a decoder to generate the summary. To obtain the reported results, we fine-tuned all pretrained models for 22 epochs with train and validation data, and we used a learning rate that warmed up to 5e-5 with a batch size of 8. LEAD-1 baseline is included, a competitive extractive baseline for news summarisation by extracting the first sentence. We report BERTScore (Zhang et al., 2020), BARTScore (Yuan et al., 2021), Bleurt score (Sellam et al., 2020), Meteor (Banerjee and Lavie, 2005), Rouge (Lin, 2004), infolm score (Colombo et al., 2021) and Frugal score (Kamal Eddine et al., 2022a). Frugal 1 uses a tiny bert base mover scorer[13]. Frugal 2 uses a tiny deberta bertscore[14].

Table 4 benchmarks the performance of the models fine-tuned on the headline generation task. FinAraT5 shows very promising results compared to multilingual versions of mT5, especially with Base and Small models. It outperformed all the small and base models. This confirms the importance of pre-training monolingual models. Finally, all T5-based models outperform BERT2BERT by a significant margin.

Table 5 reports results of infoLM score (Colombo et al., 2021) on FinAraSum test split. This score calculates the mathematical distribution of the reference and candidate sentences then it calculates the mathematical distance between the two distributions. The less the distance is, the better the result is. We report different mathematical distances. The authors claim that regarding fluency and text structure, FisherRao distance works better.

We also report about rouge metrics. We report ROUGE-1, ROUGE-2 and ROUGE-L f1- scores (Lin, 2004). The original google implementation of rouge does not support the Arabic language. Instead, we used another implementation[18]. This table is for informational purposes only because

[12] https://huggingface.co/bert-base-multilingual-uncased

[13] https://huggingface.co/moussaKam/frugalscore_tiny_deberta_bert-score

[14] https://huggingface.co/moussaKam/frugalscore_tiny_deberta_bert-score

[18] https://github.com/ARBML/rouge_score_ar

	title generation					
	BE score	BA score	Frugal 1	Frugal 2	Bleurt	meteor
lead	72.66	44.51	85.10	86.30	-15.00	27.08
mT5 small	79.17	62.48	91.50	89.30	5.90	32.43
araT5 small	79.68	63.33	91.65	89.40	6.70	33.84
bert2bert base	75.50	56.27	91.26	89.20	-1.57	18.25
mT5 base	79.03	62.44	91.46	89.30	5.51	31.27
araT5 base	80.21	64.37	92.04	89.50	8.29	35.18
finaraT5 base(ours)	**80.46**	**64.66**	92.04	89.52	8.76	36.08
mT5 large	80.32	64.54	92.04	89.45	9.42	35.47
araBART Large	80.35	64.67	92.30	89.55	9.50	35.18

Table 4: Results on FinAraSum test split. BE Score stands for Bert score which uses uncased multilingual bert checkpoint. BA score stands for Bart score and uses the mbart checkpoint[15]. Macro F1 score averages are computed over all datasets. Frugal 1 uses a tiny bert base mover scorer[16]. Frugal 2 uses a tiny deberta bertscore[17]

	kl	alpha	beta	ab	renyi	l1	l2	l_infinity	fisher_rao
lead	-8.829	-4.252	6.993	9.256	2.206	1.893	0.285	0.134	2.887
mT5 small	-8.165	-4.090	6.705	8.258	2.053	1.861	0.292	0.144	2.832
mT5 base	-8.294	-4.120	6.830	8.387	2.086	1.867	0.295	0.145	2.842
mT5 large	-8.370	-4.123	6.880	8.462	2.089	1.867	0.297	0.147	2.845
araBART	-8.669	-4.157	7.125	8.777	2.136	1.870	0.300	0.147	2.858
araT5 small	-8.387	-4.104	6.858	8.484	2.067	1.863	0.297	0.149	2.840
araT5 base	-8.376	-4.093	6.809	8.501	2.059	1.859	0.296	0.147	2.835
finaraT5 base	-8.334	**-4.077**	6.789	8.408	**2.041**	**1.856**	0.295	0.146	**2.830**

Table 5: Reporting Results of infoLM (Colombo et al., 2021) on FinAraSum test split. The authors of InfoLM claim that it is a flexible metric and it can adapt to different criteria using different measures of information. KL stands for kl divergence between the reference and hypothesis distribution. alpha and beta stand for alpha and beta divergence between the reference and hypothesis distribution. Renyi stands for renyi divergence between the reference and hypothesis distribution. l1 and l2 and l_infinity stands for three versions of norm distances between the reference and hypothesis distribution. FisherRao is the distance between the reference and hypothesis distribution. Finally, the authors claim that regarding fluency and text structure, FisherRao distance works better

MODEL	rouge1	rouge2	rougeL
lead	23.21	9.55	21.02
mT5 small	37.91	20.02	35.93
araT5 small	39.31	21.33	37.24
bert2bert	24.34	9.10	23.08
mT5 base	37.35	19.45	35.31
araT5 base	40.91	22.49	38.71
finaraT5 base	**41.74**	**23.19**	**39.61**
mT5 large	41.17	23.14	38.99
araBART	41.38	23.19	39.34

Table 6: Results on FinAraSum in terms of ROUGE-1 (R1), ROUGE-2 (R2), ROUGE-L (RL)

rouge variants are based on n-gram-form matching and have no sense of semantic similarity (Kamal Eddine et al., 2021).

In Table 7, we report the degree of novel ngrams introduced per model on the generated summaries on the test dataset. We can see that FinAraT5 introduces on average 28.8% , 64.5% , 82.6% , 91.5% of novel unigrams, Bigrams, Trigrams and 4-grams respectively, in its summaries for the title generation task. These scores are superior to other models. We can deduce that FinAraT5 and araT5 base are more abstractive than other models, especially multilingual T5. Bert2Bert is an exception since it generates some random words. This may be justified by the fact that it is not a native encoder-decoder model.

We followed the method proposed by (Rothe et al., 2020). We calculated the percentage of repetition and the average length of the generated summary. The repetition rate is the rate of summaries including at least one word from the most frequent 400 words from the corpus. Results are detailed in Table 8. For repetitions, the less redundant models, closest to the ground truth, are araBART and MT5 large. The use of auto-generative models on abstractive datasets increases the risk of repetition. Our model FinAraT5 shows less repetition on this summarisation dataset than other models. This is a good sign of the quality and novelty of the generated text. Bert2Bert is the only model redundant with 15.76% of repetitions. The architecture of the model justifies this. In addition, this model generated more tokens on average. This is consistent with previous results. All the other models generate nine tokens coherently with the gold summaries' length.

6.2 Discriminitative task: Financial sentiment prediction:

In order to further test the model we performed training on a discriminative task. We can use either encoder-only models or encode-decoder models. In the second, the input sequence is passed to both the encoder and the decoder and we add a classification head to the representation of the sequence of tokens. text-to-Text models can perform discriminative tasks

Training details: we fine-tuned the models for 20 epochs with a learning rate of 2e-5. We set the batch size to be 32 and the max sequence length to 128.

Evaluation: Table 9 shows the results of the sentiment analysis task. We report only the models with a base architecture. FinAraT5 performed the best on the test split. We can conclude that the monolingual financial text model could perform well on generative and discriminative tasks.

6.3 Discussion

Multilingual vs. Monolingual Models The empirical results show the better performance of dedicated monolingual language models compared to multilingual models (multilingual T5 versions: 110 languages) of the same size (base). The FinAraT5 model benefits from the previously pretrained araT5 on a large Arabic MSA corpus. In addition, it specialises in the financial context by being trained on a large financial narrative corpus. This improved performance could be explained by the quality of the data collected from different financial reports and financial newswires in Arabic.
Transfer Learning: Multilingual models do not learn very well on some downstream tasks. Our monitoring of the evolution of bertscore using wandb.ai show that multilingual models do not improve significantly during training. They have a flat curve during the fine-tuning process compared to the monolingual models. mT5 models may suffer from capacity issues.
Abstractiveness: We manually evaluate our text-to-text models' ability to generate good quality financial context MSA text. Our qualitative analysis shows that the FinAraT5 is very powerful in summarising news and in generative tasks in general. It has a compelling ability to abstract and paraphrase the input. It introduces advanced grammatical Arabic structures, such as using question marks, exclamations, and oratorical questions. In addition,

Model	% of novel n-grams in system generated summary			
	Unigrams	Bigrams	Trigrams	4-grams
Gold	37.1	73.1	88.8	95.1
bert2bert	34.2	77.3	95.4	97.3
mT5 small	22.1	52.8	71.1	82.0
araT5 small	27.5	62.2	80.4	90.0
mT5 base	23.7	54.2	72.6	83.7
araT5 base	28.3	63.9	82.4	91.5
FinAraT5 base(ours)	28.8	64.5	82.6	91.5
mT5 large	26.3	60.8	79.5	88.8
araBART large	25.6	60.0	79.2	89.0

Table 7: Proportion of novel n-grams in the generated summaries on the test dataset using different models .

	Length	Repetition %
Gold	9.04	0.52
mT5_small	9.27	4.44
araT5_small	9.28	5.64
bert2bert	10.03	15.76
mT5_base	9.05	2.64
araT5_base.txt	9.08	3.64
finarat5_base	9.05	3.48
mT5_large	8.92	1.2
araBART large	8.71	1.04

Table 8: Summary statistics: Sequence length generated by models on the Test dataset and percentage of word repetition in the summary among the most common 400 words in the dataset

MODEL	arabert	mT5	araT5	finarat5
accuracy	0.9246	0.9246	0.9362	**0.9449**

Table 9: Sentiment analysis task on the test split

we see good use of commas, which is crucial in Arabic, enabling emphasis on some words. Finally, we can see that different versions of Arabic T5 generate content that has approximately the same meaning using different structures. In conclusion, we can see that our models are able to generate syntactically correct summaries in Arabic.

Evaluation methods: Three main types of metrics are used to measure the similarity between two sets of data: model-based, n-gram, and statistical-based. Model-based metrics use models to estimate the similarity between two sets of data. N-gram metrics measure the similarity between two data sets by counting the number of n-grams or phrases appearing in both data sets. Statistical-based metrics use statistical models to estimate the similarity between two data sets.

Grammatical: We manually analysed system summary generated examples. The generated text is syntactically correct, and the spelling is also correct. It is also in line with the general topic of the corpus. The method allowed the generation of coherent text and has succeeded in fully synthesising suitable Arabic financial text.

7 Conclusion And Future Work

We presented FinAraT5, a domain-specific skilled text-to-text model for financial Arabic text understanding and generation. We trained the model on a large dataset of Arabic financial texts which we collected and cleaned ourselves. Then we evaluated the model's performance on a new benchmark that we created. The results showed that FinAraT5 could model and generate coherent and accurate texts in the Arabic financial domain, outperforming strong baselines and demonstrating its ability to be a good benchmark as a language model for financial Arabic. Overall, we claim that FinAraT5 represents a significant step forward in the development of practical natural language processing tools for financial Arabic, which is at the moment still less well represented in previous research, and we believe it has the potential to be fine-tuned on several other downstream tasks (machine translation, summarisation, and information retrieval). Our next step is to perform a large-scale human evaluation task on Mechanical Turk.

8 Acknowledgements

We gratefully acknowledge support from Lancaster University to provide access to the high-end computing GPU cluster. We thank also the Google TensorFlow Research Cloud TFRC[19] program for the free access to Cloud TPUs V3.8 which was crucial for the pre-training process. In addition, we thank Google Cloud research team for the 1,000 USD GCP credits[20] to perform this research. We also acknowledge the AraT5 team for their help and for sharing their model checkpoints.

References

Muhammad Abdul-Mageed, AbdelRahim Elmadany, and El Moatez Billah Nagoudi. 2021. ARBERT & MARBERT: Deep Bidirectional Transformers for Arabic. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7088–7105, Online. Association for Computational Linguistics.

Zaid Alyafeai and Maged Saeed. 2020. tkseem: A preprocessing library for arabic. `https://github.com/ARBML/tnkeeh`.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. AraBERT: Transformer-based model for Arabic language understanding. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 9–15, Marseille, France. European Language Resource Association.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Kaj Bostrom and Greg Durrett. 2020. Byte pair encoding is suboptimal for language model pretraining.

Sumit Chopra, Michael Auli, and Alexander M. Rush. 2016. Abstractive sentence summarization with attentive recurrent neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 93–98, San Diego, California. Association for Computational Linguistics.

Amina Chouigui, Oussama Ben Khiroun, and Bilel Elayeb. 2017. Ant corpus: An arabic news text collection for textual classification. In *2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA)*, pages 135–142.

Pierre Colombo, Chloe Clave, and Pablo Piantanida. 2021. Infolm: A new metric to evaluate summarization & data2text generation. In *AAAI Conference on Artificial Intelligence*.

Mahmoud El-Haj and Rim Koulali. 2013. Kalimat a multipurpose arabic corpus.

Ibrahim Abu El-Khair. 2016. 1.5 billion words arabic corpus. *CoRR*, abs/1611.04033.

Abbas Ghaddar, Yimeng Wu, Ahmad Rashid, Khalil Bibi, Mehdi Rezagholizadeh, Chao Xing, Yasheng Wang, Duan Xinyu, Zhefeng Wang, Baoxing Huai, Xin Jiang, Qun Liu, and Philippe Langlais. 2021. JABER: junior arabic bert. *CoRR*, abs/2112.04329.

Moussa Kamal Eddine, Guokan Shang, Antoine Tixier, and Michalis Vazirgiannis. 2022a. FrugalScore: Learning cheaper, lighter and faster evaluation metrics for automatic text generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1305–1318, Dublin, Ireland. Association for Computational Linguistics.

Moussa Kamal Eddine, Antoine Tixier, and Michalis Vazirgiannis. 2021. BARThez: a skilled pretrained French sequence-to-sequence model. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9369–9390, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Moussa Kamal Eddine, Nadi Tomeh, Nizar Habash, Joseph Le Roux, and Michalis Vazirgiannis. 2022b. AraBART: a pretrained Arabic sequence-to-sequence model for abstractive summarization. In *Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP)*, pages 31–42, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization.

[19] `https://sites.research.google/trc/about/`

[20] `https://cloud.google.com/edu/researchers`

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Pekka Malo, Ankur Sinha, Pyry Takala, Pekka J. Korhonen, and Jyrki Wallenius. 2013. Good debt or bad debt: Detecting semantic orientations in economic texts. *CoRR*, abs/1307.5336.

El Moatez Billah Nagoudi, AbdelRahim Elmadany, and Muhammad Abdul-Mageed. 2022. AraT5: Text-to-text transformers for Arabic language generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 628–647, Dublin, Ireland. Association for Computational Linguistics.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019a. Exploring the limits of transfer learning with a unified text-to-text transformer. *CoRR*, abs/1910.10683.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019b. Exploring the limits of transfer learning with a unified text-to-text transformer.

Sascha Rothe, Shashi Narayan, and Aliaksei Severyn. 2020. Leveraging pre-trained checkpoints for sequence generation tasks. *Transactions of the Association for Computational Linguistics*, 8:264–280.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal. Association for Computational Linguistics.

Thibault Sellam, Dipanjan Das, and Ankur P Parikh. 2020. Bleurt: Learning robust metrics for text generation. In *Proceedings of ACL*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Yi Yang, Mark Christopher Siy UY, and Allen Huang. 2020. Finbert: A pretrained language model for financial communications.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation.

Imad Zeroual, Dirk Goldhahn, Thomas Eckart, and Abdelhak Lakhouaja. 2019. OSIAN: Open source international Arabic news corpus - preparation and integration into the CLARIN-infrastructure. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 175–182, Florence, Italy. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating Text Generation with BERT. *arXiv:1904.09675 [cs]*. ArXiv: 1904.09675.