2023 IEEE Symposium Series on Computational Intelligence (SSCI)
Mexico City, Mexico. December 5-8, 2023

Domain-Specific Large Language Model Finetuning using a Model Assistant for Financial Text Summarization

Loukia Avramelou[1], Nikolaos Passalis[1], Grigorios Tsoumakas[2], and Anastasios Tefas[1],
[1]*Computational Intelligence and Deep Learning Group, AIIA Lab*
[2]*Intelligent Systems Laboratory*
Department of Informatics, Aristotle University of Thessaloniki
Thessaloniki, Greece
{avramell, passalis, greg, tefas}@csd.auth.gr

***Abstract*—The financial market and public opinion are correlated. This means that changes in the financial market can result in changes to public opinion and changes to public opinion can result in changes to the financial market. Accordingly, it is essential for understanding and interacting with the financial market to gather text content from online sources and process it. As a result of the rapid growth of social media and other online sources, we have seen an exponential rise in data, particularly textual data, in recent years. It can be difficult for a person to read, let alone process, the massive volumes of data generated every day. This indicates that we need automated methods for processing textual data and extracting useful information. Automated text summarization is a method of shortening huge amounts of text without losing essential information. Transformers, which can efficiently manage and analyze textual data, are state-of-the-art text summarization models. However, developing such an automated text summarization model specialized in a domain (e.g. finance) can be challenging since we lack necessary domain-specific summarization datasets. In this work, we propose a pipeline for fully automating the finetuning of a text summarization model in a specific domain, namely cryptocurrency domain, without the involvement of human annotators. To this end, we introduce a novel method for self-improvement of text summarization models which relies on a model assistant which encodes domain knowledge, enabling finetuning text summarization models in specific domains in which we lack specific-domain summarization datasets. The proposed method is evaluated on a cryptocurrency-related text summarization problem and three well-known Large Language Models (LLMs) used for text summarization.**

***Index Terms*—Natural Languange Processing, Text Summarization, Finetuning Large Language Models**

I. Introduction

Large Language Models (LLMs) with powerful natural language processing capabilities that enable real-time understanding and extraction of insights from the constant flow of data, facilitating more efficient and insightful decision-making processes, play an important role in large-scale data stream applications (e.g., financial forecasting). As a result, it is critical to investigate ways to adapt them in large-scale data stream applications and advantage of them due to their ability to rapidly process data streams from a variety of domains. LLMs can be used in order to solve a variety of tasks, such as sentiment analysis, machine translation, and automated text summarization.

Automated Text Summarization is a Natural Language Processing (NLP) task that aims to shorten large text into text that contains the key content and information. Although summarization is a challenging task, it is crucial since we can create shorter text that is more accessible and manageable, especially in these years when an enormous amount of data are generated every day. There is a large number of automated text summarization methods in the literature that vary from simple methods, like TextRank [1] and LexRank [2] to complex neural network models, such as Sequence-to-Sequence models [3] and Transformers [4]. Pretrained models, such as PEGASUS [5] and BART [6], are used for text summarization and have a wide range of applications in a variety of fields. These models are designed on the Transformers architecture and use an attention mechanism in order to analyze text based on its context.

In the case of models that focus on a specific domain, it is important to handle each domain differently because of the different terminology and vocabulary of each domain. This means that we should develop a different model for summarizing medical documents [7] and a different model for summarizing financial documents [8] because of their specific vocabulary. Therefore, when we try to train a specific domain model, we need to finetune the model by considering the terms typically used in the corresponding domain. One example of finetuned PEGASUS model is the Financial Summarization Pegasus model [8]. This model is finetuned on a financial news dataset that contains 2,000 articles from Bloomberg, on topics such as stocks, markets, and currencies and is able to generate quality summaries from text with financial content.

Extending this idea, we aim to develop a model for text summarization that focuses on cryptocurrency-related text. However, there are no datasets that contain cryptocurrency-related texts along with their corresponding summaries as ground truth for training a text summarization model. On one hand, if we have text with cryptocurrency-related content, we can generate summaries using trained text summarization models. On the other hand, these summaries do not encode efficiently domain knowledge since they are not finetuned in this specific domain. Thus, there is a need for an automated way to encode this knowledge in order to train a cryptocurrency-related text summarization model. This leads us to the main research question of this paper: Can we

978-0-7381-4408-5/23/$31.00 ©2023 IEEE

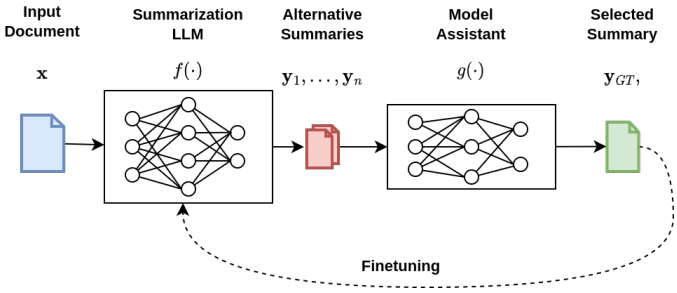


Fig. 1: Model assistant-based self-improvement of LLMs. First, we create a number of alternative summaries using a baseline text summarization model. After that, we use a model assistant (text classification model) in order to select the best summary as ground truth. Finally, we compile the final finetuning dataset that consists of the original text and the selected summary.

develop a pipeline that employs a model that encodes domain knowledge for a task different from summarization (e.g., classification) and use it as a "model assistant" during the finetuning process? Such a model assistant should encode specific domain knowledge without necessarily being a text summarization model and help to finetune a domain specific text summarization model. This would enable us to create a summarization dataset focused on a specific domain, i.e., in our case cryptocurrencies, and then finetune a text summarization model without the involvement of human annotators which will cost time and effort.

The main contribution of this work is the implementation of a model assistant-based pipeline for finetuning LLMs in order to develop a domain specific summarization model that can summarize text about cryptocurrency assets, enabling models to self-improve. In more detail, we finetuned three different pre-trained summarization models using cryptocurrency-related documents. We extracted these documents from a variety of online sources, such as online articles or social media content and then we finetuned the corresponding model using the proposed pipeline. Finally, we evaluated generated summaries based on their ability to predict the financial market, while also examining the effect of the finetuning process on models' behavior.

The rest of the paper has the following structure. In Section II, we describe the proposed pipeline that we followed in order to finetune automated text summarization models. Then, in Section III, we provide the results of the conducted experiments and we evaluate the impact of the finetuning process in baseline models. Finally, in Section IV, we present our conclusion and discuss future work.

II. Proposed Method

In this section, we describe the proposed pipeline for finetuning text summarization models in order to develop models that are able to handle cryptocurrency-related content and improve financial trading.

In order to develop a cryptocurrency-related text summarization model, we use a baseline text summarization model. The proposed pipeline consists of 4 phases as follows: a) data collection and preprocessing, b) text classifier training, c) summarization model finetuning, and d) evaluation. We describe each of these phases in detail in the following sections.

A. Data Collection and Preprocessing

The data collection and preprocessing process is divided into two steps. In the first step, we collected text data, such as text documents from online sources that are related to cryptocurrency content. For each of these text documents, we generated a number of summaries using the baseline model by varying the decoding process employed by the model. In the second step, we collected price data on an hourly base regarding the same time period when the text documents are published. We transformed continuous price data p_t for each time step t into a categorical label l_t as:

$$l_t = \begin{cases} 1 \text{ (up class)}, & \text{if } \frac{p_t - p_{t-1}}{p_{t-1}} > T \\ -1 \text{ (down class)}, & \text{if } \frac{p_t - p_{t-1}}{p_{t-1}} < -T \\ 0 \text{ (same class)}, & \text{otherwise} \end{cases}, \quad (1)$$

where p_t is the price in the current time-step and p_{t-1} is the price in the previous time-step.

We chose $T = 0.25$ as a threshold in order to transform continuous data into categorical data since this threshold results in equal samples of data in each class. Next, we merged text samples with price data based on date. Thus, each sample of our initial dataset contains the current date/time step t, the original text $\mathbf{x}_t$, and the corresponding price movement l_t. Finally, we divided the whole dataset into 3 disjoint subsets in order to use them in each of the following steps. In more detail, we used subset X_c for text classifier training, X_f for summarization model finetuning and X_e for evaluation.

B. Text Classifier Training

As mentioned before, our most challenging task in finetuning a text summarization model on a specific domain is to create a dataset that contains both the text documents with their corresponding summaries. The use of human annotators has a very large cost in time and effort that is impossible to use. In this work, we propose using another model, called model assistant, to this end. Given a number of generated summaries, the model assistant evaluates them based on how well they encode the domain knowledge. Since we focus on the financial domain, we employ a model assistant that aims to classify price movements based on the generated summaries. Therefore, we employ the classification model to make predictions for a summary $\mathbf{y}_i$ as $o_i = g(\mathbf{y}_i)$. Thus, we train the model $g(\cdot)$ to perform classification, based on the corresponding price movement l_t. Note that we might have multiple documents that can be associated with the same time-stamp t based on the employed sub-sampling.

To further improve the ability of the model assistant to introduce domain knowledge and have a more accurate market representation, we train a text classifier model $g(\cdot)$ that can predict the price movements at the past 24 hours, at the current hour, and the future 24 hours having as input summaries generated from the baseline text summarization models. In other words, we developed a model that is able to provide a prediction for the financial market in the past,

present, and future based on a given summary $\mathbf{y}_t$. This model acts as an *assistant* in the self-improvement process of the LLM since it enables it to choose the best summary that should be used as ground truth from the set of generated summaries. It is worth mentioning that this model can be used not only for creating the finetuning dataset but also independently for predicting the financial market.

C. Summarization Model Finetuning

In this step, we finetuned the baseline summarization model $f(\cdot)$ using our finetuned dataset that was created using the text classifier model $g(\cdot)$ of the previous step. As mentioned before, for each one of the extracted text documents of subset $\mathbf{x}_i \in X_f$ we generated a number of summaries $y_1, y_2, ..., y_n$ using the baseline summarization model $f(\cdot)$. This is possible by altering the next token selection in the employed decoding strategy. Then, we employ the classification model to make predictions for each summary as:

$$o_i = g(\mathbf{y}_i). \tag{2}$$

In order to select the best summary, we calculate the loss L used for training the classification model for each generated summary as:

$$e_i = L(o_i, l_i). \tag{3}$$

Then, we can select the best summary $\mathbf{y}_{GT} = \mathbf{y}_k$ where

$$k = \arg\min_i e_i. \tag{4}$$

Therefore, we feed the generated summaries to the text classifier and the summary with the least loss value y_k is chosen as the ground truth summary in the finetuning dataset. So, the finetuning dataset consists of text documents along with their corresponding summaries.

In this way, we can generate a finetuning dataset completely automatically without human involvement making the process faster and more affordable. After creating the finetuning dataset, we finetuned the baseline text summarization model using this dataset in order to develop a text summarization model that is focused on cryptocurrency-related content.

D. Evaluation

This is the last step of the proposed pipeline. In this step, we evaluated the quality of our generated summaries based on their ability to predict various aspects of the financial market in comparison to summaries generated by a baseline (not finetuned) model. To this end, we employ a text classification model, as described in Section II, trained to predict price movements from the generated summaries. It is worth mentioning that we do not employ typical summary measures such as ROUGE [9] or BLEU [10] to evaluate model performance since we are interested in improving ground truth summaries. We, also, observed the differences between the generated summaries in comparison to baseline generated summaries in order to spot the effect of finetuning on the model's behavior.

III. Experimental Evaluation

Datasets and Experimental Setup As we described in Section II, the proposed pipeline requires three different datasets, i.e., X_c for training the text classifier model, X_f for creating the finetuning dataset, and X_e for evaluation purposes. We managed to collect a total of 7,710 text documents that are related to the Bitcoin cryptocurrency. We chose the Bitcoin cryptocurrency since it is the most popular and most expensive cryptocurrency. The datasets for training the text classifier and for finetuning the text summarization models consist of 3,605 samples each, while we used the rest 500 samples for the evaluation dataset. We also collected BTC-USDT pair price data as mentioned in Section II for the corresponding time period. We used a BERT model for text classification [11]. This model's architecture consists of a BERT model with a sequence classification head on top that predict 9 labels (3 labels for each of past, present, and future price movements). As for text summarization models we used three variations of PEGASUS-based models, as explained below.

Experimental Evaluation We conducted experiments using 3 variations of PEGASUS-based models. More specifically, we finetuned: a) a generic text summarization model, namely the PEGASUS-XSUM model, b) a specialized text summarization model, namely the Financial PEGASUS model, and a generic large language model, the PEGASUS-LARGE.

A. Finetuning a Generic Text Summarization model

In the case of PEGASUS-XSUM, we followed the proposed pipeline, as described before, in order to finetune it using cryptocurrency-related text. First, we generated three summaries for each text document using the PEGASUS-XSUM model and we created the three data subsets. It is worth mentioning that PEGASUS-XSUM is a text summarization model that is trained in news articles from BBC and covers a variety of domains, such as News, Politics, Sports, and more. This means that it is not focused on the finance domain. After generating the summaries, we trained the BERT text classifier in 8 epochs with a learning rate set to 2×10^{-6} and as loss, we used the cross entropy loss. In Fig. 2, we can observe the accuracy and loss curves during the training of the BERT text classifier. In this case, the training accuracy is greater than 68%, and in the validation set the accuracy is close to 48%. These accuracy values are explained by the fact that PEGASUS-XSUM is not focused on the financial domain and this has as a result to have not accurate summaries. Despite this, note that the model still provides predictions better than a random classifier, since it predicts three equally probable classes (random chance classification accuracy 33.33%).

Next, we finetuned the PEGASUS-XSUM model using the proposed pipeline. In Fig. 4a), we can see the training loss during the finetuning process. Here, we can see that the loss values in the first steps of finetuning are higher than the loss values in the case of the financial summarization model (see Fig. 4b)). As mentioned before, PEGASUS-XSUM is not focused on the financial domain and probably failed to generate cryptocurrency-related summaries in the first steps

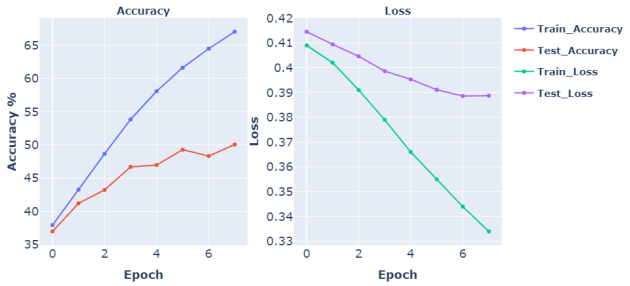


Fig. 2: Accuracy and loss curves during BERT text classifier training using PEGASUS-XSUM model.

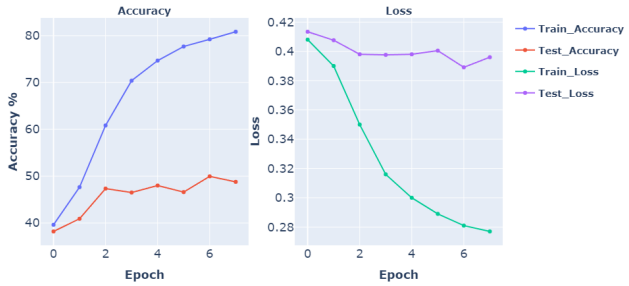


Fig. 3: Accuracy and loss curves during BERT text classifier training using Financial Summarization model.

TABLE I: Price movement forecasting results for baseline and finetuned summarization model in the case of PEGASUS-XSUM model.

	BASELINE	FINETUNED	DIFF %
ACCURACY - PAST	47.17%	**47.83%**	**+1.4 %**
ACCURACY - NOW	47.39%	**49.13%**	**+3.67%**
ACCURACY - FUTURE	47.83%	**48.48%**	**+1.36%**

TABLE III: Accuracy results for baseline and finetuned summaries in the case of Financial Summarization model.

	BASELINE	FINETUNED	DIFF %
ACCURACY - PAST	49.27%	**51.36%**	**+4.24 %**
ACCURACY - NOW	52.41%	**53.04%**	**+1.2 %**
ACCURACY - FUTURE	50.73%	**51.36%**	**+1.24 %**

of finetuning. However, as the finetuning continues the model was trained and generated cryptocurrency-related summaries since the training loss is reduced significantly.

Finally, we evaluated the model's performance based on the generated summaries. First, we used the BERT text classifier to determine if using the finetuned summaries can result in more accurate financial market predictions. In Table I, we can see that using the finetuned summaries resulted in more accurate financial market predictions. This indicates that the finetuning process improved the model's ability to generate cryptocurrency-related summaries.

We, also, evaluated finetuned model's performance regarding the vocabulary and how it changed in comparison to the baseline vocabulary. In Table II, we can observe the top-5 most used words in the summaries of the evaluation dataset. We can see that in the case of the finetuned summaries the most used words are all related to cryptocurrency vocabulary. However, in the case of the most used words in the baseline summaries, we can see that there are words that are not correlated with the cryptocurrency domain.

B. Finetuning a Specialized Text Summarization model

Similarly, in the case of the Financial Summarization PEGASUS model, for each of the text documents we generated three summaries using this model. After that, we trained the BERT text classifier for eight epochs with a learning rate set to 10^{-5}. The accuracy and loss curves during training are provided in Fig. 3. The training accuracy in past, present, and future price movements is greater than 80%, but in the validation set is close to 49%. In this case, we observed better results, especially in the training accuracy since this baseline model is focused on the financial domain and is able to summarize cryptocurrency-related content more efficiently than PEGASUS-XSUM.

TABLE II: Top-5 most used words in the original text, summaries generated by baseline model, and summaries generated by the finetuned model in the case of PEGASUS-XSUM model.

	ORIGINAL TEXT	BASELINE	FINETUNED
1.	bitcoin	copyrighted	bitcoin
2.	btc	images	nakamoto
3.	xbt	bitcoin	cryptocurrency
4.	crypto	transactions	ethereum
5.	sats	nakamoto	satoshi

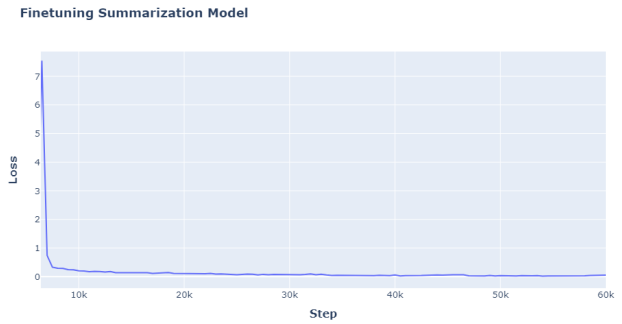
Next, we finetuned the financial summarization model using the proposed method, as described before. We trained the text summarization model for 16 epochs. In Fig. 4b), we can observe the training loss during the finetuning process. The training loss is reduced as excepted during the finetuning. This means that the model is trained to generate cryptocurrency-related summaries. In comparison to the PEGASUS-XSUM finetuning loss, we observed that the loss in the early steps of the financial text summarization model is less than the loss of PEGASUS-XSUM in the same steps. This could be further explained by the fact that the financial summarization model is fine-tuned in the financial domain, which is correlated to cryptocurrency-related content, and hence the loss is significantly lower in this case.

Finally, we evaluated the model's performance based on the generated summaries. First, we used the BERT text classifier in order to determine if using the finetuned summaries can result in more accurate financial market predictions. In Table III, we can see that using the finetuned summaries resulted in more accurate financial market predictions. This indicates that the finetuning process improved the model's ability to generate cryptocurrency-related summaries that can improve financial market prediction.

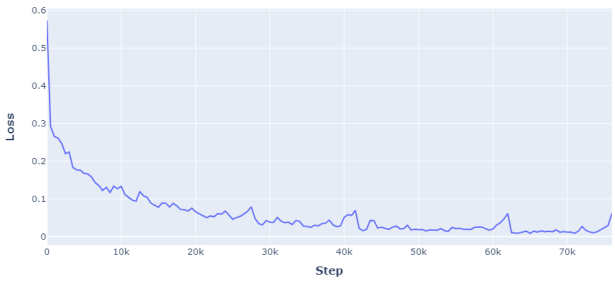
Apart from that, we evaluated finetuned model's performance regarding the vocabulary and how it changed in comparison to the baseline vocabulary. In Table IV, we can observe the top-5 most used words in the summaries of the evaluation dataset. We can see that in the case of finetuned summaries the words like *bitcoin* and *blockchain* are used most of the time. In baseline summaries, we see that these

TABLE IV: Top-5 most used words in original text, summaries generated by baseline model and summaries generating by finetuned model in the case of Financial Summarization model.

	ORIGINAL TEXT	BASELINE	FINETUNED
1.	bitcoin	check	bitcoin
2.	btc	bitcoin	check
3.	xbt	metaverse	blockchain
4.	crypto	news	metaverse
5.	sats	blockchain	news



(a) PEGASUS-XSUM



(b) Financial PEGASUS model

Fig. 4: Cross entropy loss during finetuning Text Summarization models.

words are also used but they are not the most used ones. Moreover, in Table VII, we can see an example of generated summaries by the baseline and the fine-tuned model.

C. Finetuning a Large language model

Lastly, we implemented the proposed pipeline using the PEGASUS-LARGE model. PEGASUS-LARGE is a language model that is trained on C4 and HugeNews datasets and can be used in order to train text summarization models. In order to implement the proposed pipeline in this model, we used the finetuning dataset that we used in the case of the financial summarization model since we can not generate summaries using the PEGASUS-LARGE model without finetuning it to perform summarization first. Similar to the PEGASUS-XSUM model, the training loss values are also higher in the first steps of finetuning but after some steps, it is reduced dramatically indicating that the model is able to generate cryptocurrency-related summaries.

Finally, we evaluated the model's performance using the BERT text classification that we used in the case of the fine-

TABLE V: Accuracy results for finetuned summaries in the case of PEGASUS-LARGE model.

	FINETUNED
ACCURACY - PAST	52.2%
ACCURACY - NOW	51.78%
ACCURACY - FUTURE	49.9%

TABLE VI: Top-5 most used words in original text and summaries generating by the PEGASUS-LARGE finetuned model. Note that the model was not already trained to perform summarization.

	ORIGINAL TEXT	FINETUNED
1.	bitcoin	nfts
2.	btc	bitcoin
3.	xbt	check
4.	crypto	market
5.	sats	cap

tuning of the financial summarization model. In Table V, we can see the accuracy obtained from the BERT text classifier using finetuned PEGASUS-LARGE generated summaries. We observed that accuracy levels are similar to previous results despite the fact that the model is not pre-trained to perform text summarization tasks.

In order to evaluate the generated summaries in terms of generated text and vocabulary, we extracted the top-5 most used words of generated summaries of the evaluation dataset, as seen in Table VI. We can see that again the most used words are related to cryptocurrency and financial vocabulary. This indicates that using the proposed pipeline in the PEGASUS-LARGE model results in a finetuned text summarization model that is able to handle cryptocurrency-related content.

IV. Conclusion

This paper presented a methodology for finetuning text summarization models in a fully automated way without the involvement of human annotators, enabling models to self-improve, while adapting to a new domain using a model

TABLE VII: Example of generated summaries using Financial PEGASUS finetuned model.

Original Text	The cryptocurrency market has oscillated between support and resistance levels in the past few months. While it appears that Bitcoin now can consistently remain above $20,000, the market's biggest asset has had trouble breaking past the $30,000 mark.1 However, other crypto assets are showing consistent growth over the past few weeks. This week, we examine XRP (XRP), The Sandbox (SAND), XDC Net- work (XDC), and Lido (LDO). In selecting these assets, we have considered several factors, including positive technical devel- opments, significant news events, and no- ticeable changes in price. Terra Classic (LUNC) was the largest gainer by percent- age last week but was removed because it is worth much less than 1 cent.
Baseline Model	This week, we look at noticeable, The Sandbox, XDC Network, and Lido.
Finetuned Model	We look at the best-performing crypto assets of the past week. Terra Classic was the largest gainer by percentage last week.

assistant. This pipeline overcomes the most challenging aspect of text summarization training which is the lacking of summarization datasets and generates summarization datasets in an automated way using a different assistant model, e.g., a BERT text classifier. Experimental results in three different models demonstrate the effectiveness of the proposed pipeline in finetuning text summarization models. The results indicate that the proposed method holds significant potential in the field of text summarization finetuning. This work has been also an inspiration for future research. First, it is critical to investigate the effect of the proposed pipeline in other domains and how we can implement this pipeline in different types of text summarization methods or different models, such as GPT models. Moreover, it is interesting to examine the use of reinforcement learning in the finetuning of text summarization models for maximizing the ability of these models to generate cryptocurrency-related summaries that also improve financial market predictions and overcome issues that could potentially arise from the smaller variety due to the employed decoding strategies. Lastly, it is crucial to investigate the opportunities of data stream analysis using LLMs in different applications and explore the ways to adapt LLMs in data stream applications [12].

ACKNOWLEDGMENT

This research was funded by the project "SEMANTIC ANNOTATION AND METADATA ENRICHMENT OF OPEN VIDEO STREAMS USING DEEP LEARNING" (Project code: KMP6-0079092) that was implemented under the framework of the Action "Investment Plans of Innovation" of the Operational Program "Central Macedonia 2014 2020", that is co-funded by the European Regional Development Fund and Greece.

REFERENCES

[1] R. Mihalcea and P. Tarau, "TextRank: Bringing order into text," in *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, (Barcelona, Spain), pp. 404–411, Association for Computational Linguistics, July 2004.

[2] G. Erkan and D. R. Radev, "Lexrank: Graph-based lexical centrality as salience in text summarization," *Journal of artificial intelligence research*, vol. 22, pp. 457–479, 2004.

[3] R. Nallapati, B. Zhou, C. dos Santos, Ç. Gulçehre, and B. Xiang, "Abstractive text summarization using sequence-to-sequence RNNs and beyond," in *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, (Berlin, Germany), pp. 280–290, Association for Computational Linguistics, Aug. 2016.

[4] Y. Liu and M. Lapata, "Text summarization with pretrained encoders," *arXiv preprint arXiv:1908.08345*, 2019.

[5] J. Zhang, Y. Zhao, M. Saleh, and P. Liu, "Pegasus: Pre-training with extracted gap-sentences for abstractive summarization," in *International Conference on Machine Learning*, pp. 11328–11339, PMLR, 2020.

[6] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," *arXiv preprint arXiv:1910.13461*, 2019.

[7] J. L. D. Nunna, V. Hanuman Turaga, and S. Chebrolu, "Extractive and abstractive text summarization model fine-tuned based on bertsum and bio-bert on covid-19 open research articles," in *International Conference on Machine Learning and Big Data Analytics*, pp. 213–223, Springer, 2022.

[8] T. Passali, A. Gidiotis, E. Chatzikyriakidis, and G. Tsoumakas, "Towards human-centered summarization: A case study on financial news," in *Proceedings of the First Workshop on Bridging Human–Computer Interaction and Natural Language Processing*, pp. 21–27, 2021.

[9] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Text Summarization Branches Out*, (Barcelona, Spain), pp. 74–81, Association for Computational Linguistics, July 2004.

[10] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, (Philadelphia, Pennsylvania, USA), pp. 311–318, Association for Computational Linguistics, July 2002.

[11] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[12] N. Passalis, M. Tzelepi, P. Charitidis, S. Doropoulos, S. Vologiannidis, and A. Tefas, "Deep video stream information analysis and retrieval: Challenges and opportunities," in *2022 IEEE 5th International Conference on Multimedia Information Processing and Retrieval (MIPR)*, pp. 336–341, IEEE, 2022.