The 17th International Conference on Space Operations, Dubai, United Arab Emirates, 6 - 10 March 2023.
Copyright © 2023 by the authors on behalf of SpaceOps. All rights reserved.
SpaceOps-2023, ID # 349

Natural Language Processing for Explainable Satellite Scheduling

Cheyenne Powell[1, a], **Audrey Berquand**[2, b], **Annalisa Riccardi**[1, c]

[1] *Department of Mechanical and Aerospace Engineering, University of Strathclyde, 75 Montrose Street, Glasgow G1 1XJ, United Kingdom.*
[2] *European Space Agency, Keplerlaan 1, 2201 AZ Noordwijk, The Netherlands*
[a] *cheyenne.powell@strath.ac.uk.*
[b] *audrey.berquand@esa.int.*
[c] *annalisa.riccardi@strath.ac.uk.*

Abstract

Facilitating the interactions between humans and Artificial Intelligence (AI) in automated systems is becoming central with the advancements in technology and their more widespread adoption in practical applications. Mathematical programming scheduling techniques are a driving factor to assist ground station operators both on board the satellite, for autonomous decision making, and on ground, for supporting mid-term operations scheduling. When communication to ground is limited, scheduling algorithms require a level of autonomy and robustness able to respond to issues arising on board the satellite in the absence of communication with a ground operator. Moreover, explanations must be generated, along side schedules, for the operator to build and gain trust in the autonomous system.

Explainable Artificial Intelligence (XAI) is an emerging topic in AI. Explanations are a necessary layer to effectively deploy autonomous trustworthy systems in practical applications. Queries may arise from operators such as why, what, when and how the scheduled actions were selected autonomously on board for a specific time. Explanations are provided based on the definition of the problem with its respective constraints.

Autonomous decision making algorithms can be explained in several ways. Computational Argumentation (CA) and Natural Language Processing (NLP)) are some techniques, belonging to the domains of formal logic and machine learning, that can be used to generate explanations and communicate them back to the user in the form of textual output. An Argumentation Framework (AF) was created to assist in answering questions raised by the end user. The AF encodes, in its lower level, all the necessary information on when conflicts may occur between actions, as well as, environmental conditions inhibiting the occurrence of the actions within a schedule. This database of information is used to construct arguments in support or negation of user submitted queries or to provide an explanation of the complete derived schedule. NLP is then used as a bridge to communicate the relevant arguments to the user.

The queries received revolved around three main areas: the subject, the time of interest and the intent. Following the interpretation, the queries were mapped to the AF database, returning a list of conflicts, agreements and neutral outcomes. The chosen NLP method for this architecture, GPT-3 was used to then deduce the answer to the query and justify it with a textual explanation.
Keywords: Explainable Artificial Intelligence (XAI), Natural Language Processing (NLP), GPT-3, Satellite Scheduling, Abstract Argumentation (AA), Language Model (LM)

1 Introduction

The growth in number of satellite scheduling systems and the increase in their complexity has made it necessary for computational systems to be developed to replace the manual calculation of schedules, expanding the use of AI systems [1]. An example recently presented in [1] involved a Reinforcement Learning (RL) technique based on SatNet data, applied to maximise the total number of scheduled hours for 12 oversubscribed antennas. The results show that the RL method consistently increased scheduled hours compared to the Mixed Integer Linear Programming (MILP) baselines. Another study, [2], combined Deep Learning (DL) and RL techniques, also called Deep Reinforcement Learning (DRL), to assess the effectiveness of value-based Monte Carlo Tree Search (MCTS) and policy-based Proximal Policy Optimization (PPO) algorithms in maximising the amount of scientific data collected by an Earth Observation (EO) satellite. The study demonstrated that the policy-based algorithm was particularly more effective in both execution time and performance.

To the author's knowledge, there are no existing infrastructures able to explain and explore the output and decisions made by AI systems in the field of spacecraft operations. Fundamentally, XAI aims to make AI decisions more transparent to help users understand the internal logic of the system that would otherwise be hidden from the user. XAI has three main purposes: (i) *Justification*, to defend the systems decisions, (ii) *Control*, to identify and prevent inaccuracies, and (iii) *Improvement*, to fine-tune the system for better results. Each of these purposes can be achieved singularly, but combined together they enhance the overall capability of an AI system and ultimately build trust with users [3]. The meaning of trust between humans and AI systems was explored in [4], establishing that a system is trustworthy if it maintains a contract with the user(s), meaning if it performs as expected or anticipated. As soon as a system performs unexpectedly, the contract is broken and trust is lost. XAI was identified as a key means of evaluating and improving the trustworthiness of AI systems, by increasing trust in systems operating as expected and creating distrust in those that do not [4].

There are various approaches used by XAI methods to process input data. In this study we will focus on combining two approaches, firstly, by using Computational Argumentation [5] and secondly, by using NLP-based methods [6]. Argumentation techniques involve using logical reasoning to analyse data. These techniques are used to identify and evaluate arguments within data [7]. The aim is to discover relationships between data points, identify patterns and draw conclusions. There are different types of frameworks created around argumentation, some of which are Abstract Argumentation Framework (AAF), Bipolar Argumentation Framework (BAF), Tripolar Argumentation Framework (TAF) [5, 8, 9], Incomplete Argumentation Framework (iAF) [10], Structured Argumentation Framework (SAF) [11]. However, this study focuses on an Abstract Argumentation (AA) which is extracted from an AAF, a subset of AF, where binary attack relations in the form of conflicts between two or more arguments occur [12, 13]. This means any entity that negatively impacts another, is considered an attack $'-'$, while an entity that positively influences another is considered a support $'+'$.

NLP-based methods use Machine Learning (ML) algorithms to identify and understand the language data, such as relationships between data points, by extracting information from text, and interpret the context in which text is written. Language Models (LMs) are a key component of this process. They can predict the likelihood of a given sequence of words occurring in a given context [14]. Modern LMs are based on DL techniques such as Recurrent Neural Networks (RNNs) [15] and transformer networks that are trained on large corpora of text, such as web-text data and news articles [16, 17]. By identifying which words are most likely to appear in a sentence or phrase, LMs generate human-like text. The most recent Large Language Models (LLMs) such as GPT-3 [18], BLOOM [19], and Pathways Language Model (PaLM) [20] can now generate clear textual explanations, enhancing the trust in the system or informing any corrective action [6, 21, 22].

This research paper explores the potential of combining AA and NLP for answering queries related to an EO satellite schedule. Specifically, we present a novel framework for XAI incorporating AA and NLP to answer queries received from an end user and to explain the decisions made by a satellite scheduler, taking into account the various constraints and parameters.

This paper briefly reviews the existing literature in Section 2 on AA and NLP. It includes a report on the experiments in Section 3 with a real-world satellite scheduling dataset, where the model is evaluated and analysed, followed by the results and discussion in Section 4 entailing the implications of our findings for the conclusion and future applications of XAI in Section 5.

2 Background

2.1 Satellite Schedule Problem

A simple satellite scheduling problem was previously derived in [23] using an existing EO satellite with an optical payload in sun synchronous orbit. Using the Ansys STK tool, the coordinates of the spacecraft were recorded for a duration of 6 months. The optimiser then predicted the spacecraft's actions, choosing between imaging a_p, processing a_r, down-linking a_d or idling a_e, based on the spacecraft's position and the following constraints:

- Taking of images a_p: Each image requires 2.688 GB of the available on-board memory. Imaging can occur when there is sufficient light exposure and the area of interest (land) is visible.
- Processing of images a_r: Once an unprocessed image is available, processing may occur any time with an assumed processing rate of 50 MB/s at every 5 second interval. The original image will not be removed until

down-linking has been executed.

- Down-linking of images a_d: When a processed image is available in memory, down-linking can only happen with communication access to ground with a data rate of 280 MB/s per 5 second intervals.
- Idle time a_e: Is only created when no other actions are executed.
- Memory m: The overall memory at any point in time must not be exceeded, while m_{tn} represents the memory across each specific time interval.

These actions and constraints were used as a baseline for the AA layer to extract logic to prompt to the NLP layer in generating explanations for XAI.

2.2 XAI for Satellite Scheduling

XAI for satellite scheduling is an area that is currently being researched to assist Ground Station Operators (GSO) to better understand what is happening on-board a satellite. To begin, the work done by [24], was initiated to explore a technique to analyse a schedule created using the constraints such as environmental conditions of sunlight, ground communications, land visibility, and memory availability related to each action. Additionally, if action a_p was successfully replaced, and this image includes an area of interest, [25] investigated the possibilities to re-plan the retaking of an image due to a stochastic failure. AA was used in this scenario, where any action, excluding idle time (a_e), can attack and replace an existing scheduled action, on the condition that the constraints are satisfied. This means that if a_e is scheduled, it can be attacked by the other actions a_p ,a_r, and a_d respectively but cannot attack another. Likewise, a_p can be attacked by a_r, and a_d; a_r by a_p and a_d and finally a_d by a_p and a_r; where attacks are represented as $'-'$ and supports are represented as $'+'$ shown in Figure 1, with their respective environmental and memory constraints.

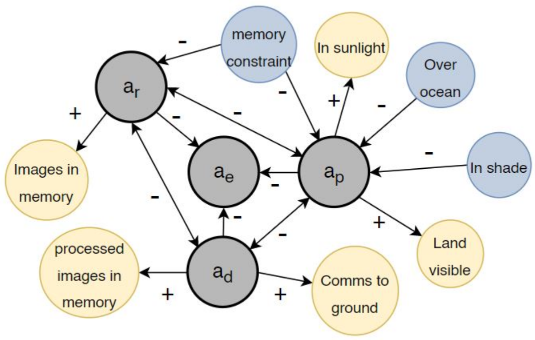

Figure 1: Conditions involved with supports and attacks for each action based on their constraints (environment and memory) of each action. [24].

This is where AA was used within a concept known as Single Exchange Property (SEP) to assist in providing explanations to the end user. SEP is defined as a singular exchange of any critical job with another job, between machines to improve an existing schedule by [26]. This definition was adapted for the satellite scheduling problem which states that any action replaced by another action (excluding itself) satisfies SEP if across any time instance within a schedule, the maximum memory has not been breached following this change at any point in time [24]. This concept checks and analyses each time interval for all possible attacks on every action scheduled followed by whether or not the exchange is deemed feasible based on the cascade of memory change throughout the rest of the schedule. Note, it is only feasible when there is no breach of the maximum memory. This design was then recognised for its potential use for providing feedback to queries through a Language Model (LM).

2.3 Prompting Language Models

A LM is a probability distribution trained on a wide corpus of documents and able to predict word sequences. LMs based on the Transformer neural network architecture have dramatically altered the NLP landscape over the past 5 years [27, 28]. This type of model architecture enables Transfer Learning, where a model first learns from an initial training objective, and is then applied to a different target objective. A standard approach for using these models on specific downstream applications usually consisted in fine-tuning the pre-trained models on small task-specific datasets. However, the GPT-3 model from OpenAI [18], an autoregressive LM, achieved higher performances using few-shot prompts, thus paving the way for the novel concept of *prompt engineering*. As defined by [29], prompt engineering consists in finding the prompt achieving the highest performances. A prompt is a text to be completed, submitted to the model. Prompt engineering is thus similar to the original training objective of LMs. The prompt includes a context indicating the task to be performed. With zero-shot, one-shot and few-shot learning, the prompt will respectively contain zero, one or several examples of the expected output. The weights of the pre-trained model are not modified with this approach [18].

In this study, chain-of-thought prompting is used to elicit a reasoning and a written explanation from a LM. Chain-of-thought prompting is defined by [30] as a sequence of reasoning steps in natural language leading to a final output. The prompt format consists of a triple of shape $<$*input, chain-of-thought, output*$>$. In the context of this study, the $input$ would be a satellite scheduling question, assessing the feasibility of exchanging two actions at a time t. The *chain-of-thought* on which the reasoning is based consists in bullets points, or arguments, summarising the status of the environmental and memory constraints. Finally, the $output$ would be the feasibility answer and the written explanation. The authors of [30] provide evidence that this prompting format yields better performances with models above 100B parameters. This constraint will be taken into consideration for the LM selection. For instance, the recent 540B parameters model presented by Google Research in [20], PaLM, achieved breakthrough performances, notably relying on chain-of-though for generating explanations for reasoning tasks.

3 Methodology

3.1 Approach

A schedule was previously generated by [23], using the principles and constraints, including the actions, as stated in section 2.1, where a day was selected at random, excluding the first day, as all variables were initialised on the first day. An argumentation layer was created for the generated schedule and the concept of SEP was applied as in [24]. Figure 2 represents the use of NLP within the flow of communication between the end user and the system. A query is passed to the argumentation layer that extracts the necessary data from the scheduler. A list of arguments is then generated and prompted, along with the query, to the LM which produces the explanation.

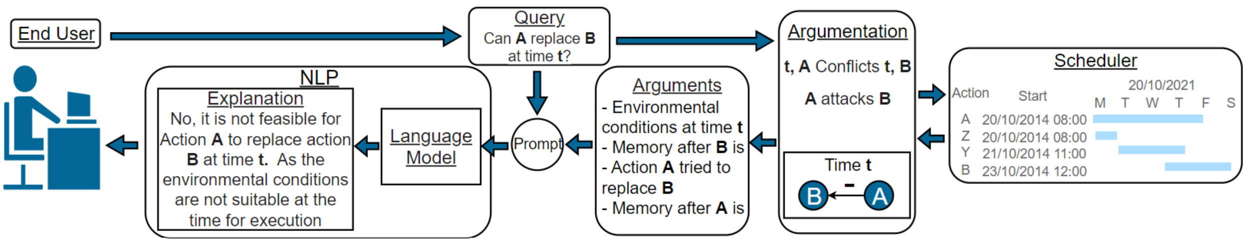

Figure 2: The use of NLP in the system.

3.2 SEP Generation

The concept of SEP, as stated in Section 2.2 that was defined by [24] and modified from the definition by [26], was applied in this paper. Using the argumentation concept from Figure 1; Figure 3 represents an overview of the memory at 3 different time intervals (m_{tn}, m_{tn+1}, m_{tn+2}) in a schedule, where green represents the scheduled actions and grey represents the other actions that can attack the scheduled action based on the constraints at that point in time.

In Figure 3, at time t_n, if a query was asked about whether any action can replace the scheduled action a_r, the possible attacks would be a_p and a_d based on the conditions: (i) there is enough memory available, and the satellite is within sunlight and over land for a_p to replace a_r; and (ii) there is ground station access and processed images in memory for a_d to replace a_r. Furthermore, the attack is only deemed feasible if the memory alteration at m_{tn} does not impact the memory during the execution of the other scheduled actions on branch 'a' for memories m_{tan+1}, m_{tan+2}. While at time t_{n+1}, action a_e can be attacked by the other 3 actions a_d, a_r and a_p only if the constraints are satisfied for this time and the future memory based on the schedule is not affected (branch 'b' for memories m_{tbn+2}, m_{tbn+3}).

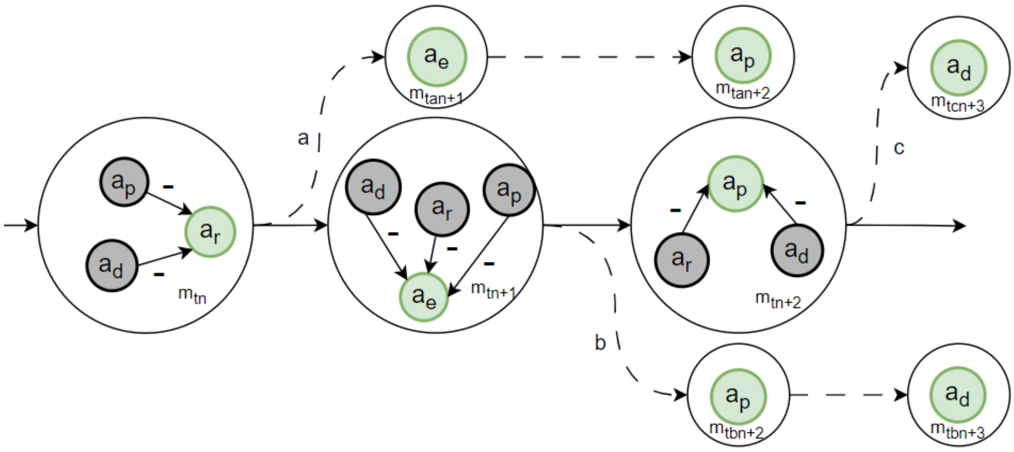

Figure 3: The functionality of SEP applied in 3 scenarios resulting in 3 different memory branches with the initially scheduled actions (a, b and c) on the conditions the attacks are successful.

Similarly, if a_p was attacked by a_d at time t_{n+2}, the exchange would only be possible if the ground station was visible, if there was at least one processed image to be down-linked, and if the action switch does not result in a memory overflow at a later stage (branch 'c' for memories m_{tcn+3} etc.). In the scenario where a_p is replaced, the coordinates of the satellite are extracted and a methodology derived in [25] was used to determine the next opportunity to retake the image. Therefore, if queried, the system would inform the end user on the next imaging opportunity.

3.3 LM selection and hyper-parameters

Several LMs were investigated, such as PaLM, BLOOM, and GPT-3. Preliminary results suggested that the GPT-3 model yielded the most promising results, hence was chosen for this study.

The parameters for GPT-3 were:

- $model =' text - davinci - 003'$
- $temperature = 0.7$
- $max - tokens = 256$
- $top_p = 1$
- $frequency - penalty = 0$
- $presence - penalty = 0$

3.4 Prompting approach

As mentioned in Section 2.3, the prompt is the text to be completed and submitted to the model. All prompts of this study include a question of format *Can action A replace action B at time T?*, followed by a list of arguments based on the environmental and memory constraints presented in Section 2.1. Figure 4 summarises how the arguments are generated using SEP and AA. The User query is passed to the argumentation layer. The environmental conditions are first verified, then the memory is checked, to eventually determine if the switch of action is feasible. If all initial constraints are fulfilled, an additional verification is done to check that the switch does not trigger a memory breach at a later stage of the schedule. Table 1 displays a few prompt examples including a user query and its list of arguments.

A query and a list of arguments is the baseline prompt applied in this study. This would correspond to a so-called Zero-Shot Learning (ZSL) approach where a prompt with one query and its set of arguments are submitted to the LM. The model then assesses whether a switch is feasible or not and justifies its output with a written explanation. Additional methods are explored where the model is also shown a few examples with the format <*query, arguments, explanation*>. These examples are manually generated. In this study, explanations are thus generated with the zero-shot, and the few-shot with 3 examples and 10 example approaches.

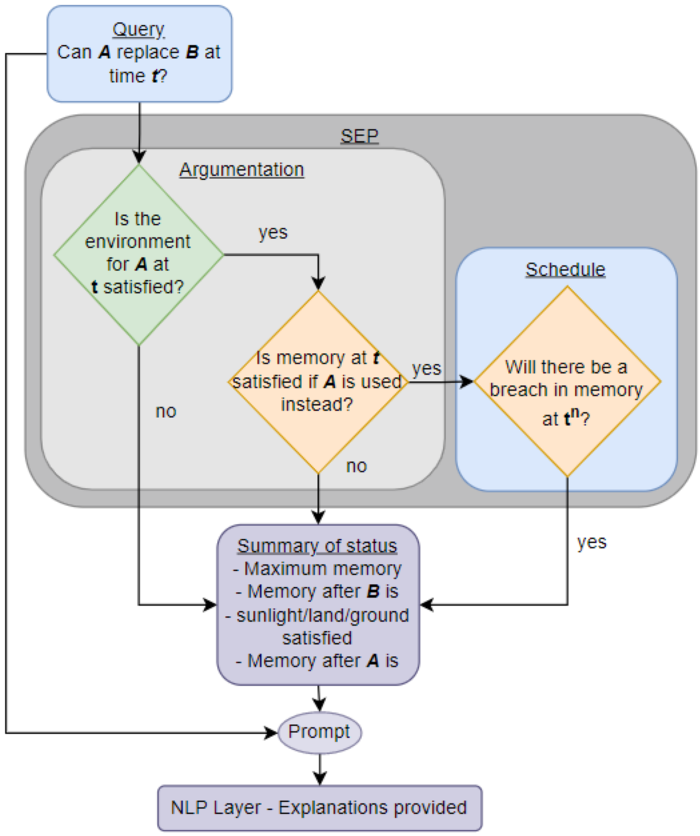

Figure 4: Decision flow based on the constraints and environmental conditions for generating prompts from the AA layer.

3.5 Dataset and Evaluation Metrics

A dataset with 50 queries and their respective arguments sets is created for this study. For each query, an answer is manually generated. This manual deduction is used as ground truth to evaluate the LM outputs. 10 additional queries with their arguments and manual answers are created to serve as examples for the Few-Shot Learning (FSL) approach.

Five metrics are applied to evaluate the explanation generated by the LM. First, the common metrics of Bilingual Evaluation Understudy (BLEU) and Recall-Oriented Understudy for Gisting Evaluation (ROUGE) scores are computed. Both scores assess different variations of n-gram overlaps between the target text and the generated text. The BLEU score, [31], is a precision oriented score while ROUGE, introduced in [32], is a recall oriented score. These scores focus on syntactic matches, failing to grasp semantic similarities. To evaluate the generated text from a semantic perspective, two transformer-based measures are applied, respectively based on token-level and sentence-level embeddings. Presented in [33], the BERTScore evaluates the similarity of two sentences by embedding each sentences' tokens and summing their cosine similarities. Sentence-BERT (SBERT) [34] provides sentence-level embeddings. The human and model generated sentences are embedded with SBERT before being compared with cosine similarity.

The scores obtained with the above automatic metrics are finally correlated with a human manual evaluation provided by 3 annotators. Human evaluation enables to look more in depth into the quality of the generated text. In [35], 4 annotators rated the factual accuracy of 30 generated facts. The mean human scores were then correlated to the ROUGE score and an in-house evaluation metric. In [36], human annotators scored the coherence of objections (to a claim) automatically generated. The rating scale, proposed in Table 2, is based on a set of best practice for human evaluation presented in [37], and tailored to the use case of this study. The annotators must evaluate the quality of the generated text based on (i) the accuracy of the deduction provided by the model, and (ii) the quality of its deduction. For instance, if the model correctly infers that a switch of actions is feasible, and its justification correctly leverages the environmental parameters and the memory status, then the generated text will receive the highest score.

Table 1: Prompt Examples

Query	Arguments
Can the Image taking action replace the Processing action at time 74701?	- The Processing action was initially scheduled - Processing image number 2928.29 - Memory for the scheduled action is 1804098 MB - Image taking action tried to replace Processing action - Land is satisfied - Sunlight is satisfied - The memory after Image taking at time 74701 would be 1806536 MB - The memory would have exceeded at time 79106 with a value of 1920094 MB - Maximum memory is 1920000 MB
Can the Down-linking action replace the Processing action at time 81752?	- The Processing action was initially scheduled - Processing image number 3012.56 - Memory for the scheduled action is 1391638 MB - Down-linking action tried to replace Processing action - No access to the ground station - Maximum memory is 1920000 MB
Can the Image taking action replace the Processing action at time 27646?	- The Processing action was initially scheduled - Processing image number 2216.8 - Memory for the scheduled action is 1232350 MB - Image taking action tried to replace Processing action - Land is satisfied - Sunlight is satisfied - The memory after Image taking at time 27646 would be 1234788 MB - The memory would have exceeded at time 79106 with a value of 1920094 MB - Maximum memory is 1920000 MB

Table 2: Scoring scale (end result: normalize average score

Points	Justification
3	The model's deduction is correct and its explanation leverages all relevant environmental and memory parameters.
2	The model's deduction is correct but its justification is partial and lacks to reference all the relevant parameters.
1	The model's deduction is correct but its justification is wrong, it either lacks to reference the relevant parameters or makes a faulty usage of them.
0	The model's deduction is incorrect.

4 Results and Discussions

Table 3 displays the scores obtained for the three different prompting approaches: ZSL, few-shot with 3 (FSL-3) and 10 examples (FSL-10). Overall the best performances are achieved with the FSL-10 approach. The manual evaluation metric, labeled as '*Human*' reveals a drastic improvement between the ZSL and the FSL learning approaches. The BERTScore and the SBERT scores find those approaches to perform similarly, and therefore do not reflect as well as the manual metric the quality spectrum of the explanations generated. The syntactic evaluation metrics, the BLEU and ROUGE scores, indicate lower performances as the text generated by the LM often uses synonyms or different words than the ones found in the target text. Table 4 displays the explanations automatically generated with FSL-10, and their respective scores, for the prompts presented in Table 1. As shown in this table, the syntactic metrics reflect poorly on the explanations quality as their scores depend on the tokens contained in the target text, and not on the meaning of the target explanation. The semantics metrics based on BERTScore and SBERT yield a more trustworthy evaluation, in agreement with the manual evaluation. Finally, the manual evaluation, although more time-consuming, is the most reliable metric.

It should be noted that the accuracy of the model's answers is high for all approaches. For 90% of the queries, GPT-3 with ZSL correctly infers whether the switch of actions is feasible or not. The accuracy raises to 96% for the FSL-10 method. The quality of the explanation then varies between the approaches. However, the model, even

Table 3: Average scores and standard deviation obtained for each evaluation metric and prompting approach

Approach	**BLEU**	**ROUGE**	**BERTScore**	**SBERT**	**Human**
ZSL	$0.56_{0.14}$	$0.37_{0.11}$	$0.83_{0.03}$	$\mathbf{0.73}_{0.1}$	$0.53_{0.08}$
FSL-3	$0.68_{0.15}$	$0.41_{0.12}$	$0.85_{0.03}$	$0.72_{0.12}$	$0.83_{0.12}$
FSL-10	$\mathbf{0.74}_{0.12}$	$\mathbf{0.43}_{0.09}$	$\mathbf{0.86}_{0.02}$	$\mathbf{0.73}_{0.1}$	$\mathbf{0.84}_{0.09}$

Table 4: Examples of explanations generated by the GPT-3 model with the few-shot learning 10 examples approach

Query	**Target Text**	**Generated Text**	**Scores**
Can the Image taking action replace the Processing action at time 74701?	The processing action was attempted to be replaced by image taking, resulting in a violation of the memory constraint with a saturation at time 79106, thus meaning processing was critical for this time. The 2 constraints at the time of attack were initially satisfied, but the exchange was infeasible.	No, the image taking action cannot replace processing at time 74701 because the memory after the action would exceed the maximum memory at a later time. Therefore, the processing action is not feasible to be replaced.	BLEU: 0.65 ROUGE: 0.34 BERTScore: 0.83 SBERT: 0.5 Human: 0.78
Can the Down-linking action replace the Processing action at time 81752?	No, it is not feasible to replace the Processing action at time 81752 with a Down-linking action. The Processing action was initially scheduled for Processing image number 3012.56 and with a scheduled memory of 1391638 MB. With no access to the ground station and no change to the scheduled memory, it would not be possible to replace the Processing action with a Down-linking action.	No, the down-linking action cannot replace the processing action at time 81752 as there is no access to the ground station. Therefore the processing action will continue to process image number 3012.56.	BLEU: 0.40 ROUGE: 0.51 BERTScore: 0.88 SBERT: 0.90 Human: 1
Can the Image taking action replace the Processing action at time 27646?	It is not feasible for image taking to replace processing at time 27646. It is initially possible with both sunlight and land satisfied, however, due to the memory being exceeded later at time 79106, it is no longer possible for this action to take place	At time 27646, the memory after the image aking action would be 1234788 which is below the maximum memory. However, later at time 79106, the memory would be exceeded resulting in an infeasible result which means, image taking cannot replace the processing action at time 27646.	BLEU: 0.81 ROUGE: 0.42 BERTScore: 0.84 SBERT: 0.74 Human: 0.89

with ZSL, demonstrates an impressive understanding of domain-specific technical language. Appendix A displays several examples of generated texts with the different learning methods. The following comments are based on the observations of the 3 annotators. With the FSL-10 method, the model is able to better grasp the concept of time, whether the memory will be exceeded at a later time or not. It also better formulates the concept of revisit time and delays in rescheduling an imaging. The ZSL and FSL-3 learning methods have a tendency to reuse the full arguments in their explanation while the FSL-10 approach is more efficient at reformulating the arguments with more complex vocabulary and turn of phrases. The ZSL approach often yields confused explanations related to the processing of an image, mixing it with the allocated memory. Tables 5, 6, and 7 display the average scores obtained for each learning method per type of queries. Queries related to a switch of the *Processing* action with the *Down-linking* action scored the lowest for the ZSL and FSL-3 methods. As shown in the fourth example in Appendix A, this is likely due to the confusion around the memory triggered in the model by the word *processing*.

Table 5: Average score per user query type with the Zero-Shot approach

Replace \ With	Imaging	Down-linking	Processing
Imaging		2.2	2.3
Down-linking	1		1.9
Processing	1.6	0.9	
Idle	1	1	2.9

Table 6: Average score per user query type with the Few-Shot (3 examples) approach

Replace \ With	Imaging	Down-linking	Processing
Imaging		2.8	2.7
Down-linking	2.9		2.9
Processing	2.5	1.9	
Idle	2.7	2.7	3

Table 7: Average score per user query type with the Few-Shot (10 examples) approach

Replace \ With	Imaging	Down-linking	Processing
Imaging		2.1	2.1
Down-linking	2.6		2.1
Processing	2.7	2.6	
Idle	2.7	3	2.8

5 Conclusion

The alliance of AA and NLP for XAI of an EO satellite schedule has shown great potential. First, for a given user query, a list of arguments summarising the schedule status and constraints was generated by AA utilizing SEP. This information was prompted to GPT-3, a LM within the NLP layer, either following a ZSL or FSL approach. For each query, the model generated an answer and an explanation. Even though GPT-3 is trained on a general corpus of texts, it produced promising results for space systems related queries. In comparing the results, the FSL-3 and FSL-10 generated more accurate explanations than ZSL, with FSL-10 often producing more complex sentences. However, all approaches achieved a high accuracy in determining whether the action switch was feasible or not.

Overall, the use of NLP for explainable satellite scheduling has demonstrated the effectiveness in increasing transparency in how actions were selected for a satellite schedule. This has opened a new avenue of research for satellite scheduling and other decision making processes. Some of which are: to explore more argumentation concepts and their applicability to XAI for satellite schedules; to expand the complexity, size and type of query datasets; and also to provide a more detailed analysis of the average scores per query types.

Acknowledgements

This study was half-funded by ESA under the OSIP Co-Sponsored PhD activity: “Robust and Explainable Mission Planning and Scheduling (REMPS)” No. 4000132894/20/NL/MH/hm. The authors would also like to acknowledge the support of ESA through the Research Fellow program. Finally, they wish to warmly thank Dimitris Kardaris (ESA) and Simone Frattini (Solenix) for all the insightful discussion about mission operations constraints and opportunity for explainability.

Appendix A: Examples of text generated by the GPT-3 model with different learning approaches.

Query	Arguments	Approach	Outputs
Can the Down-linking action replace Image taking action at time 56661?	- The image taking action was initially scheduled - Land satisfied for the scheduled action - Sunlight satisfied for the scheduled action - Memory for the scheduled action is 729698 MB - Down-linking action tried to replace Image taking action - Ground station access - The memory after Down-linking at time 56661 would be 724210 MB - The image number that would be down-linked is 2540.63 - The satellite coordinates are: latitude 67.027671, longitude -60.361479', 'image retake times are:'('day:6', 'time: 57251'), ('day:6', 'time: 57256'). - Maximum memory is 1920000 MB	**Target:**	The down-linking action can replace image taking at time 56661. The ground station is accessible, thus enabling down-linking of image 254063 to take place. If down-linking was to taking place instead, the image retake times would be on day 6 at 57251 and 57256 to retrieve the image that was replaced.
		ZSL:	Yes, the down-linking action can replace the image taking action at time 56661 as the memory would not be saturated, and ground station access is available. The image number that would be down-linked is 2540.63 and the next opportunity for image retake at latitude 67.027671, longitude -60.361479 is on day 6 at time 57251.
		FSL 3:	Yes, the Down-linking action can replace the Image taking action at time 56661. The memory after Down-linking at time 56661 would be 724210 MB and the image number that would be down-linked is 2540.63. The satellite coordinates are latitude 67.027671, longitude -60.361479 and the image retake times are day:6 time: 57251 and day:6 time: 57256. The maximum memory available is 1920000 MB.
		FSL 10:	It is feasible for down-linking to replace image taking, as the memory would remain below the maximum memory. The next opportunity for this image to be retaken is on day 6 at time 57251.
Can the Image taking action replace the Processing action at time 33816?	- The Processing action was initially scheduled - Processing image number 2297.53 - Memory for the scheduled action is 1198358 MB - Image taking action tried to replace Processing action - Land is satisfied - Sunlight is satisfied - The memory after Image taking at time 33816 would be 1200796 MB - The memory would have exceeded at time 79106 with a value of 1920094 MB - Maximum memory is 1920000 MB	**Target:**	The exchange was initially possible at time 33816 but the memory would later be exceeded at time 79106 therefore making it infeasible to do the exchange.
		ZSL:	No, the Image taking action cannot replace the Processing action at time 33816. The memory after Image taking at time 33816 would exceed 1920000 MB after 79106, which exceeds the maximum memory allowed.
		FSL 3:	No, the image taking action cannot replace processing at time 33816 because the memory would exceed the maximum allowed at time 79106. Therefore, it is not feasible for the image taking action to replace the processing action
		FSL 10:	At time 33816, the memory after the image taking action would be 1200796 which is below the maximum memory. However, later at time 79106, the memory would be exceeded resulting in an infeasible result which means, image taking cannot replace the processing action at time 33816.

Query	Arguments	Approach	Outputs
Can the Image taking action replace the Processing action at time 7416?	- The Processing action was initially scheduled - Processing image number 2000.28 - Memory for the scheduled action is 1674702 MB - Image taking action tried to replace Processing action - Land is satisfied - Sunlight is not satisfied - Maximum memory is 1920000 MB	**Target:**	Image taking cannot replace processing at time 7416. This is due to sunlight not being satisfied, as a result, leaves the memory is the same and no changes made to the scheduled action
		ZSL:	No, the Image taking action cannot replace the Processing action at time 7416. The Processing action was initially scheduled and the memory for the scheduled action is 1674702 MB, which is greater than the maximum memory of 1920000 MB. Additionally, although the land is satisfied, the sunlight is not, which may prevent the Image taking action from replacing the Processing action.
		FSL 3:	No, the image taking action cannot replace processing at time 7416 because sunlight is not satisfied which is a requirement for the action to be executed. Therefore memory is the same.
		FSL 10:	No, the image taking action cannot replace the processing action at time 7416, because even though land is satisfied, sunlight is not which is a requirement for the action to be executed. Therefore memory will be unaffected.
Can the Down-linking action replace the Processing action at time 7271?	- The Processing action was initially scheduled - Processing image number 1997.58 - Memory for the scheduled action is 1667452 MB - Down-linking action tried to replace Processing action - No access to the ground station - Maximum memory is 1920000 MB	**Target:**	It is not feasible to replace the processing action at time 7271 with the down-linking action as there is no access to the ground station. Therefore the memory remains the same and the initial action will proceed.
		ZSL:	No, the Down-linking action cannot replace the Processing action at time 7271. The Processing action requires 1667452 MB of memory, which is greater than the maximum memory available (1920000 MB). Additionally, the Down-linking action cannot be performed without access to the ground station.
		FSL 3:	No, the Down-linking action cannot replace the Processing action at time 7271 because there is no access to the ground station and the memory is already saturated. This means that the image will not be able to be down-linked and the memory will remain the same.
		FSL 10:	No, the down-linking action cannot replace the processing action at time 7271 as there is no access to the ground station. Therefore, image number 1997.58 will proceed to be processed.

References

[1] E. Goh, H. S. Venkataram, B. Balaji, M. D. Johnston, and B. Wilson, "Satnet: A benchmark for satellite scheduling optimization," 2022.

[2] A. Herrmann and H. Schaub, "A comparison of deep reinforcement learning algorithms for earth-observing satellite scheduling," in *AAS Spaceflight Mechanics Meeting*, Austin, TX, Jan. 15–19 2023, paper No. AAS 23-116.

[3] P. Banerjee and R. P. Barnwal, *Methods and Metrics for Explaining Artificial Intelligence Models: A Review*. Cham: Springer International Publishing, 2023, pp. 61–88. [Online]. Available: https://doi.org/10.1007/978-3-031-12807-3_4

[4] A. Jacovi, A. Marasović, T. Miller, and Y. Goldberg, "Formalizing trust in artificial intelligence: Prerequisites, causes and goals of human trust in ai," in *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, 2021, pp. 624–635.

[5] K. Čyras, A. Rago, E. Albini, P. Baroni, and F. Toni, "Argumentative xai: A survey," 5 2021. [Online]. Available: http://arxiv.org/abs/2105.11266

[6] E. Balkir, S. Kiritchenko, I. Nejadgholi, and K. C. Fraser, "Challenges in applying explainability methods to improve the fairness of nlp models," 6 2022. [Online]. Available: http://arxiv.org/abs/2206.03945

[7] A. Vassiliades, N. Bassiliades, and T. Patkos, "Argumentation and explainable artificial intelligence: A survey," *Knowledge Engineering Review*, vol. 36, 2021.

[8] D. M. Gabbay, "Logical foundations for bipolar and tripolar argumentation networks: preliminary results," *Journal of Logic and Computation*, vol. 26, no. 1, pp. 247–292, 2016. [Online]. Available: https://browzine.com/articles/47422557

[9] A. Rago, O. Cocarascu, C. Bechlivanidis, D. Lagnado, and F. Toni, "Argumentative explanations for interactive recommendations," *Artificial Intelligence*, vol. 296, p. 103506, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0004370221000576

[10] G. Alfano, S. Greco, F. Parisi, and I. Trubitsyna, "Incomplete argumentation frameworks: Properties and complexity," 2022. [Online]. Available: www.aaai.org

[11] S. Pandzić, "A logic of defeasible argumentation: Constructing arguments in justification logic," *Argument and Computation*, vol. 13, pp. 3–47, 2022.

[12] P. M. Dung, "On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games," *Artificial Intelligence*, vol. 77, no. 2, pp. 321–357, 1995. [Online]. Available: https://www.sciencedirect.com/science/article/pii/000437029400041X

[13] P. Besnard, C. Cayrol, and M. C. Lagasquie-Schiex, "Logical theories and abstract argumentation: A survey of existing works," pp. 41–102, 2020.

[14] R. Jozefowicz, O. Vinyals, M. Schuster, N. Shazeer, and Y. Wu, "Exploring the limits of language modeling," 2016. [Online]. Available: https://arxiv.org/pdf/1602.02410.pdf

[15] K. Fiok, W. Karwowski, E. Gutierrez, and M. Reza-Davahli, "Comparing the quality and speed of sentence classification with modern language models," *Applied Sciences*, vol. 10, no. 10, 2020. [Online]. Available: https://www.mdpi.com/2076-3417/10/10/3386

[16] L. Floridi and M. Chiriatti, "Gpt-3: Its nature, scope, limits, and consequences," *Minds and Machines*, vol. 30, no. 4, pp. 681–694, 2020.

[17] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[18] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess,

J. Clark, C. Berner, S. Mccandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," 2020, pp. 1877–1901. [Online]. Available: https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf

[19] T. L. Scao, A. Fan, C. Akiki, E. Pavlick, S. Ilić, D. Hesslow, R. Castagné, A. S. Luccioni, F. Yvon, M. Gallé *et al.*, "Bloom: A 176b-parameter open-access multilingual language model," *arXiv preprint arXiv:2211.05100*, 2022.

[20] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, and N. Fiedel, "Palm: Scaling language modeling with pathways," April 2022. [Online]. Available: http://arxiv.org/abs/2204.02311

[21] M. Danilevsky, K. Qian, R. Aharonov, Y. Katsis, B. Kawas, and P. Sen, "A survey of the state of explainable ai for natural language processing," *arXiv preprint arXiv:2010.00711*, 2020.

[22] K. Chowdhary, "Natural language processing," *Fundamentals of artificial intelligence*, pp. 603–649, 2020.

[23] C. Powell and A. Riccardi, "Towards explainability of on-board satellite scheduling for end user interactions," vol. IAC-21, pp. 1–10, 2021. [Online]. Available: https://iafastro.directory/iac/paper/id/63954/abstract-pdf/IAC-21,B6,2,8,x63954.brief.pdf?2021-03-29.16:07:05

[24] C. Powell and A. Riccardi, "Abstract argumentation for explainable satellite scheduling," in *2022 IEEE 9th International Conference on Data Science and Advanced Analytics (DSAA)*, 2022, pp. 1–10.

[25] C. Powell and A. Riccardi, "On-board re-planning of an earth observation satellite for maximisation of observation campaign goals," Sep. 2022, 73rd International Astronautical Congress (IAC), IAC2022 ; Conference date: 18-09-2022 Through 22-10-2022. [Online]. Available: https://www.iafastro.org/events/iac/iac-2022/

[26] K. Čyras, D. Letsios, R. Misener, and F. Toni, "Argumentation for explainable scheduling (Full Paper with Proofs)," *arXiv*, 2018.

[27] A. Vaswani, G. Brain, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Łukasz Kaiser, and I. Polosukhin, "Attention is all you need," 2017. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[28] J. Devlin, M.-W. Chang, K. Lee, K. T. Google, and A. I. Language, "Bert: Pre-training of deep bidirectional transformers for language understanding." Association for Computational Linguistics, 2019, pp. 4171–4186.

[29] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and G. Neubig, "Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing," *ACM Computing Surveys*, 9 2022. [Online]. Available: https://doi.org/10.1145/3560815

[30] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le, and D. Zhou, "Chain of thought prompting elicits reasoning in large language models," January 2022. [Online]. Available: http://arxiv.org/abs/2201.11903

[31] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation." Association for Computational Linguistics, July 2002, pp. 311–318.

[32] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries." Association for Computational Linguistics, July 2004, pp. 74–81.

[33] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, "Bertscore: Evaluating text generation with bert," April 2019. [Online]. Available: http://arxiv.org/abs/1904.09675

[34] N. Reimers and I. Gurevych, “Sentence-bert: Sentence embeddings using siamese bert-networks,” August 2019. [Online]. Available: http://arxiv.org/abs/1908.10084

[35] B. Goodrich, V. Rao, P. J. Liu, and M. Saleh, “Assessing the factual accuracy of generated text.” Association for Computing Machinery, July 2019, pp. 166–175.

[36] L. Thorburn and A. Kruger, “Optimizing language models for argumentative reasoning,” September 2022. [Online]. Available: https://lukethorburn.com/

[37] C. van der Lee, A. Gatt, E. van Miltenburg, and E. Krahmer, “Human evaluation of automatically generated text: Current trends and best practice guidelines,” *Computer Speech and Language*, vol. 67, 5 2021.