Stakeholder-specific Jargon-based Representation of Multimodal Data within Business Process

Aleksandar Gavric, Dominik Bork and Henderik A. Proper

Business Informatics, TU Wien, Favoritenstrasse 9-11/194-3, 1040 Vienna, Austria

Abstract

Stakeholders can struggle to understand and engage with process models due to a mismatch between the technical language used and their own domain-specific jargon and personal communication styles. The paper explores the application of transformer-based architectures to enhance the representation of process models and additional multimodal process data by tailoring them to the language of stakeholders. We present an approach that personalizes process model representations through two types of paraphrasers: one that aligns with domain-specific jargon and another that adapts to individual stakeholder styles. We developed a golden dataset from process model-stakeholder interaction simulation and a silver dataset using large language models to train and validate our approach. Initial findings suggest that these methods could enhance stakeholder engagement and contribute to better teaching of process mining and procedural thinking.

Keywords

Process Models, Transformer models, Multimodal Evidence, Process Representation

1. Introduction

Process mining focuses on extracting insights from event logs to discover, monitor, and improve actual processes by analyzing the flow of activities within an organization [1]. Beerepoot et al. [2] have highlighted that the struggle between human involvement and task automation in managing work processes points to the significant impact that resolving these challenges will have on knowledge-intensive work. Despite its potential, one of the significant challenges in process mining is effectively communicating the insights gained from these analyses to stakeholders [3], who often come from diverse backgrounds with varying levels of familiarity with the technical and domain-specific language. The precision required in process models leads to the use of jargon, which, while transparent to domain experts, can be confusing or opaque to others. This communication barrier can hold back the adoption of process mining insights, limiting their impact on decision-making and process improvement.

For instance, let's imagine a project manager, Sarah, who oversees the implementation of a new digital healthcare system in a large hospital. Sarah has extensive experience in project management, but her familiarity with healthcare-specific jargon is limited. She works closely with a team of doctors, nurses, and IT specialists, each fluent in their domain language. During meetings, the medical professionals often discuss processes in terms that are second nature to them—terms like "EHR integration," "clinical workflows," and "patient pathway optimization." To them, these phrases precisely capture the complexity of the processes involved in patient care. However, to Sarah, who lacks a clinical background, these discussions often feel like a foreign language, making it difficult for her to grasp the important details of the system she's managing. Now, let's imagine another scenario where Sarah is working on a similar project, but this time, the communication has been tailored to her level of understanding. Instead of using healthcare-specific jargon, the process models are described in more general project management terms. For instance, rather than discussing "EHR integration," the conversation revolves around "aligning the digital system with existing hospital processes." Instead of "clinical workflows," they talk about

Companion Proceedings of the 17th IFIP WG 8.1 Working Conference on the Practice of Enterprise Modeling Forum, M4S, FACETE, AEM, Tools and Demos co-located with PoEM 2024, Stockholm, Sweden, December 3-5, 2024

aleksandar.gavric@tuwien.ac.at (A. Gavric); dominik.bork@tuwien.ac.at (D. Bork); henderik.proper@tuwien.ac.at (H. A. Proper)

0009-0005-1243-7722 (A. Gavric); 0000-0001-8259-2297 (D. Bork); 0000-0002-7318-2496 (H. A. Proper)

"task sequences in patient care." In this scenario, Sarah could feel more confident and engaged in the project because the information is presented in a way that resonates with her background and expertise. This contrast between jargon-heavy communication and language tailored to the listener's experience is not just a hypothetical situation—it's an identified challenge in many industries [4, 5], especially those that rely on complex processes and specialized knowledge, such as healthcare, finance, manufacturing, or education. The problem of jargon-laden communication is intensified in process mining, where the interpretation and representation of process models are essential for understanding and improving organizational workflows. Process mining involves extracting knowledge from event logs to visualize and analyze processes. However, when these process models are presented in technical or domain-specific language that stakeholders may not understand, the benefits of process mining can be significantly diminished. Existing methods have made progress in addressing similar challenges by processing language [6, 7] or using visual aids [8, 9] to make process models more accessible. Natural language processing (NLP) technologies, particularly those based on transformer architectures, have also shown promise in generating more understandable text by leveraging vast amounts of contextual information. These approaches [10], however, often remain one-size-fits-all solutions, lacking the personalization needed to engage stakeholders who may have varying levels of familiarity with the subject matter.

This paper proposes an approach that leverages transformer-based architectures to create personalized representations of process models. Our goal is to bridge the communication gap by developing two types of paraphrasing: one that aligns with the *domain-specific jargon* used by experts and another that adapts to the *individual communication styles* of different stakeholders. But language is only part of the equation. In complex fields like healthcare, education, and industrial operations, process models can benefit from integrating multimodal evidence—combining text, images, data visualizations, and even video to provide a comprehensive understanding of the processes involved. Integrating this multimodal data into personalized process models adds another layer of complexity and offers an opportunity to enhance the discovery of learning patterns within process mining [11]. By understanding how different stakeholders interact with these multimodal representations, we can gain insights into their learning processes, which can improve how we teach and implement process mining techniques [10].

In the sections that follow, we will discuss the related work (Section 2), detail our methodology for developing (Section 3) and the personalized paraphrases, explore the implications of our findings for the future of process mining and stakeholder communication (Section 4), and conclude with closing remarks (Section 5). Through this work, we hope to contribute to the ongoing efforts to make complex processes more accessible, understandable, and actionable for all stakeholders involved.

2. Related Work

This section provides an overview of the most relevant research in process modeling and machine learning techniques, particularly in the domain of paraphrasing and semantic transformation of process-related representations, contextualizing our work within the broader landscape of NLP-based process model management.

2.1. Paraphrasing and Semantic Transformation in Process Models

One key challenge in business process model management is ensuring that models are interpretable and usable by various stakeholders, each with varying expertise and domain-specific knowledge. Early work by Leopold et al. [12] addressed this issue by introducing automated techniques for transforming business process models into natural language descriptions. Their approach laid the groundwork for subsequent research by demonstrating that computerized tools could effectively bridge the gap between formal process models and natural language, albeit with limited adaptability to different domains or stakeholder needs. Recent advancements in transformer-based models, such as BERT [13] and GPT [14], have opened new avenues for paraphrasing and semantic transformation tasks. These models have been applied to various domains, including text summarization, translation, and paraphrasing, but their application to process models is still an emerging field.

In a recent work [15], Kourani et al. (2024) leverage the capabilities of Large Language Models (LLMs) to represent process models in the context of Business Process Management (BPM). This study introduces a novel framework that harnesses LLMs to enhance the interpretability of complex process models, addressing challenges that arise as organizations scale and processes become increasingly intricate.

Our work builds on these advancements by applying transformer models specifically designed to paraphrase process models across multiple domains and stakeholder profiles, addressing the limitations of earlier approaches in handling domain-specific jargon and personalized communication.

2.2. Domain-Specific Language Models

The use of domain-specific language models has been explored in several contexts, particularly in medical [16] and legal [17] domains, where the accuracy of language processing is critical due to the specialized terminology involved. These studies highlight the importance of tailoring language models to specific domains to improve performance. For instance, Lee et al. [16] developed BioBERT, a variant of BERT pre-trained on biomedical text, which significantly outperformed general-purpose models on tasks like named entity recognition and relation extraction in the medical domain. Our work draws inspiration from these works by training custom transformers on process model representations specific to different domains. This approach allows our models to effectively generate paraphrases that are accurate and contextually relevant, aligning with each domain's specialized terminology and communication styles.

2.3. Hybrid Approaches to Process Model Translation

The literature has also explored hybrid approaches that combine rule-based methods with machine-learning techniques. For example, Friedrich et al. [18] developed a hybrid approach to automatically generate textual descriptions of process models by combining rule-based transformations with statistical methods. While effective, these approaches often require extensive domain knowledge to implement and are less adaptable to new or evolving domains. Our work diverges from these traditional hybrid methods by leveraging entirely data-driven transformer models, which learn the nuances of process model paraphrasing directly from training data. Zerbato et al. (2023) develop methodological guidance [19] to assist novice analysts during their analysis and build an empirical basis for process mining, laying the foundation for the development of user-centered support. Our work aims to contribute to question development in process mining and interactive modeling, addressing areas where support is still lacking.

Overall, our work extends the existing literature on process model paraphrasing by introducing transformer-based models explicitly tailored to domain-specific jargon and stakeholder communication styles.

3. Adaptive Transformer-Based Framework for Process Model Communication

We propose a transformer-based architecture to address the challenges of effectively communicating process models to stakeholders with varying levels of domain expertise (see Fig. 1). This architecture consists of two core components: a jargon-specific paraphraser (`proc2jargon`) and a personalized paraphraser (`proc2ownw`); both are designed to translate process models and multimodal data into text that is accessible and meaningful to different audiences.

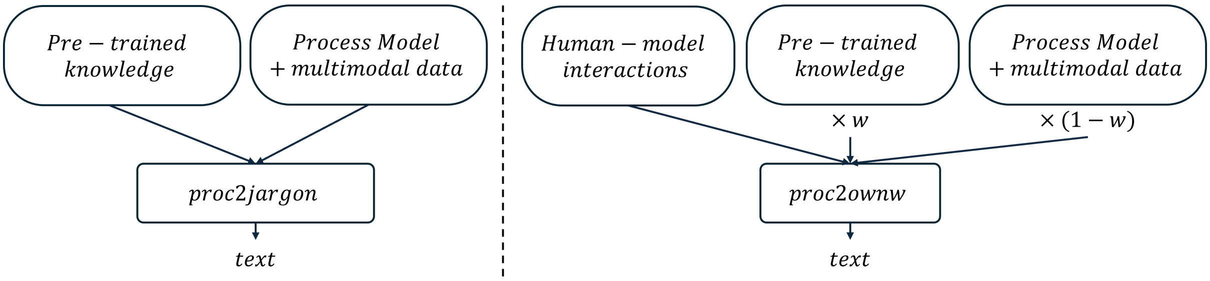


Figure 1: Illustration of our solution that includes (left) a jargon-specific paraphraser (`proc2jargon`) for domain-specific language and (right) a personalized paraphraser (`proc2ownw`) tailored to individual stakeholder preferences.

3.1. Jargon-Specific Paraphraser (`proc2jargon`)

`proc2jargon` generates text that aligns with the specialized domain jargon. This model leverages pre-trained knowledge combined with the specifics of the process model and any associated multimodal data to produce outputs that maintain the technical rigor and precision expected by domain experts.

Let D_j represent the domain-specific jargon dictionary, and $\mathbf{X}_{pm}$ denote the input process model, which includes both the textual and multimodal data features. The model's task is to generate a sequence $\mathbf{Y}_j = (y_{j1}, y_{j2}, \dots, y_{jn})$ where each $y_{ji} \in D_j$. The architecture of `proc2jargon` can be represented as

$$\mathbf{Y}_j = \text{Transformer}_{\text{jargon}}(\mathbf{E}_{ptk}, \mathbf{E}_{pm})$$

where $\mathbf{E}_{ptk}$ is the embedding of pre-trained knowledge, and $\mathbf{E}_{pm}$ is the embedding of the process model and multimodal data.

This model aims to maximize the conditional probability $P(\mathbf{Y}_j \mid \mathbf{X}_{pm}, \mathbf{E}_{ptk})$, such that:

$$P(\mathbf{Y}_j \mid \mathbf{X}_{pm}, \mathbf{E}_{ptk}) = \prod_{i=1}^{n} P(y_{ji} \mid \mathbf{X}_{pm}, \mathbf{E}_{ptk}, y_{j1}, \dots, y_{j(i-1)})$$

The conditional probability $P(\mathbf{Y}_j \mid \mathbf{X}_{pm}, \mathbf{E}_{ptk})$ represents the likelihood of generating the sequence of jargon terms $\mathbf{Y}_j$ given the input process model $\mathbf{X}_{pm}$ and the embedding of pre-trained knowledge $\mathbf{E}_{ptk}$. In essence, the model generates each jargon term one by one, ensuring that each term is not only contextually appropriate based on the process model and pre-trained knowledge but also coherent with the previously generated terms in the sequence.

3.2. Personalized Paraphraser (`proc2ownw`)

`proc2ownw` produces text personalized to individual stakeholders' communication styles and language preferences. This model balances integrating human-model interaction data with pre-trained knowledge and process model inputs to generate outputs that resonate with non-expert stakeholders.

Let H represent the set of human-model interaction embeddings, which encode personalized communication preferences, and let w be the weight that determines the influence of these interactions on the model's output. The text sequence $\mathbf{Y}_p = (y_{p1}, y_{p2}, \dots, y_{pn})$ generated by `proc2ownw` is computed as:

$$\mathbf{Y}_p = \text{Transformer}_{\text{personal}}(w \cdot \mathbf{E}_H + (1 - w) \cdot \mathbf{E}_{ptk}, \mathbf{E}_{pm})$$

where $\mathbf{E}_H$ is the embedding of human-model interactions, and $\mathbf{E}_{ptk}$ and $\mathbf{E}_{pm}$ are as defined previously. The model optimizes the conditional probability $P(\mathbf{Y}_p \mid H, \mathbf{X}_{pm}, \mathbf{E}_{ptk})$, expressed as:

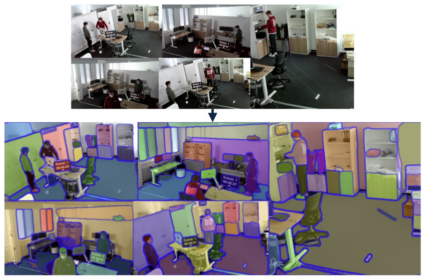
$$P(\mathbf{Y}_p \mid H, \mathbf{X}_{pm}, \mathbf{E}_{ptk}) = \prod_{i=1}^{n} P(y_{pi} \mid H, \mathbf{X}_{pm}, \mathbf{E}_{ptk}, y_{p1}, \dots, y_{p(i-1)})$$

The weights w and $1 - w$ are determined based on historical interaction data, allowing the model to adapt over time and refine its outputs for each stakeholder. This formulation allows for dynamic adaptation to the stakeholder's language preferences, producing outputs that are accurate in content and tailored in style.

Ⓐ *Object − Centric Event Log*

Step	Begin	End	Activity	Humans	Assets	Locations
1	08:48:32.113	08:48:37.040	Enter room	C1		door
2	08:48:39.585	08:48:48.719	Chat	A1, C1		it_working_desk
3	08:48:49.404	08:49:05.479	Pick asset from warehouse	A1	L1	laptop_shelf

Ⓑ *Multimodal Evidence*



Ⓒ *Process model (oc − Petri Net)*

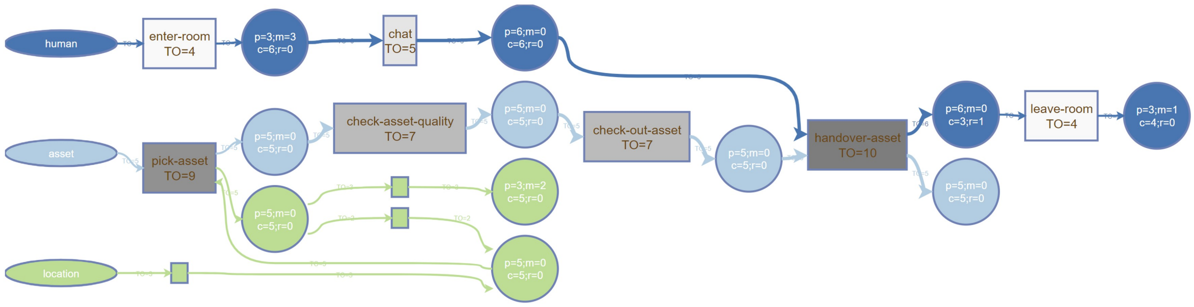


Figure 2: The Solve4X [21] process data used for training of our framework. We explore different inputs to our framework, in particular: (A) an Object-Centric Event Log (OCEL), (B) multimodal evidence, and (C) a process model example. (The labels in the model example are illustrative and should be read according to [22])

3.3. Multimodal Data Handling

To incorporate multimodal data, including textual descriptions, images, videos, and structured data (e.g., event logs), our models process input by embedding these different data types into a unified representation space. The multimodal embeddings are combined with the process model embeddings through a fusion function [20] that integrates these various data modalities. We use the unified representation space to input pre-trained multimodal embeddings as tokens.

3.4. Training `proc2jargon` and `proc2ownw` models

The training process begins with the preparation of the dataset. In our case, the dataset includes detailed process instances related to asset disbursement, Solve4X [21], where each instance provides a multimodal source of information for the model to learn from. The process model outlines several key activities: the IT staff using an asset management system to manage the issuance of items, performing quality checks, and the eventual handover of assets to clients. Alongside these textual descriptions, the dataset includes multimodal data such as sensor readings. We take the event log and multimodal evidence from Solve4X and create oc-DFG (object-centric Directly Follows Graphs), oc-Petri Net and oc-BPMN (Business Process Model and Notation) model using OC-PM (process mining) tool [22]. The data is illustrated in Fig. 2.

We utilized a supervised learning approach to train the two transformer-based models. The transformer model's attention mechanism plays a crucial role in this process. Multi-head self-attention allows the model to focus on different parts of the input sequence, capturing complex relationships between the elements of the process description. The attention mechanism calculates the weighted sum of values based on the similarity between queries and keys, enabling the model to attend to relevant

Table 1
Summary of Profile Background Knowledge

Profile	Background Knowledge
Profile A	Advanced technical background in engineering
Profile B	Background in project management with moderate technical knowledge
Profile C	Executive-level knowledge, low technical detail required

information from different subspaces. This is implemented using PyTorch and NanoGPT[1], simplifying the creation of these attention layers. The overall architecture, which includes both encoder and decoder components, is trained on a single NVIDIA A40 GPU, optimizing the model using the Adam optimizer. We pre-trained our model on the openwebtext [23] dataset and trained on custom Solve4X-based data with a micro-batch size of 12, using gradient accumulation steps of 40, a block size of 1024, 12 layers with 12 attention heads and an embedding size of 768, no dropout during pretraining, no bias in LayerNorm and Linear layers, an AdamW optimizer with a max learning rate of 6e-4, weight decay of 0.1, beta values of 0.9 and 0.95, gradient clipping at 1.0, and a learning rate decay enabled over 600,000 iterations, including 2,000 warmup steps, down to a minimum learning rate of 6e-5.

4. Evaluation

To guide our exploration of personalized process model representation, we formulate our research question (RQ) as: **how can transformer-based architectures be effectively utilized to generate paraphrased process data that adapt to the diverse linguistic preferences of stakeholders?** This question addresses the challenge of adapting technical language to improve stakeholder engagement without losing the precision required for accurate decision-making.

In this section, we provide details on the evaluation of the custom transformer models `proc2ownw` and `proc2jargon` by creating simulated stakeholder profiles with distinct personalities and generating domain-specific jargon corpora for five different fields. The evaluation aims to assess the ability of `proc2ownw` to tailor communication to specific stakeholders and `proc2jargon` to generate domain-specific language accurately. The experiments use simulated conversations parameterized to cover realistic scenarios.

We developed three distinct stakeholder personality profiles to test the personalization capabilities of `proc2ownw`. Each profile has unique communication preferences and background knowledge, which influence how they interact with process models. The profiles were designed to reflect common stakeholders in various professional environments, as detailed in Table 1. These profiles were used to generate simulated conversations, which were then fed into the model to evaluate its ability to adapt its output to different communication styles.

4.1. Interviewing

We interview each profile using a set of evaluation questions, with examples provided in the following. Questions follow the discovered steps taken during an exploratory data analysis in a process-oriented experiment [24] conducted by Daele and Janssenswillen (2023). For the test scenarios, we created questions both from the same domain as the training data [21] and from other domains to assess cross-domain generalization in relation to the domain of the process being taught. For the other domain, we chose data that originate from a large multinational company operating from The Netherlands in the area of coatings and paints [25].

Profile A (Analytical):

Human: "Can you explain the process of asset disbursement in detail?"

Model: "Certainly. The process begins with IT staff utilizing an asset management system to track each asset. Every step is logged, from retrieval to quality control, ensuring compliance with technical standards. Detailed logs and sensor data verify the accuracy at each stage."

[1]https://github.com/karpathy/nanoGPT

Profile B (Collaborative):

Human: "I think the team needs a clear outline of the steps involved."

Model: "Sure! The process is straightforward: IT staff check out the item in the system, perform a quality check, and hand it over to the client. Everyone on the team can see these steps and collaborate to ensure nothing is missed."

Profile C (Strategic):

Human: "Give me the high-level overview of asset disbursement."

Model: "The key steps involve tracking inventory in the asset management system, verifying quality, and issuing assets to clients. It's crucial for maintaining operational efficiency and reducing errors."

To evaluate `proc2jargon`, we generated domain-specific jargon corpora for the medical, police, software development, logistics, and accounting domains. These jargon corpora were created using large language model (LLM) [7] to simulate the typical language used in these domains. This evaluation aimed to test how well `proc2jargon` could adapt its output to match the specialized language of different fields. We used LLM to generate heavy jargon content for each domain by crafting specific prompts. These prompts were designed to evoke responses filled with domain-specific language, which was then used to build the jargon corpus for `proc2jargon`.

4.2. Examples

In the following, we provide examples of a tailored explanation for the processes of asset management systems for different professionals.

For *a Medical Doctor (MD):* The IT team manages new equipment by logging its details into a management system, conducting thorough quality inspections, tagging each item, and storing it in designated areas. This process ensures accurate tracking and integration into the inventory, like how patient records are carefully maintained and monitored for precise medical management.

For *a Police Officer:* IT staff handle new equipment by documenting its details in a system, inspecting it for any issues, tagging it with a barcode, and storing it in secure locations. Every action is logged, akin to how evidence is handled and recorded to ensure proper tracking and security.

For *a Software Engineer:* New IT assets are processed by entering their specifications into a management system, verifying their condition through quality checks, tagging them with barcodes, and updating their storage locations. This ensures that all inventory data is accurately synchronized and traceable, much like managing code versions and deployments.

For *a Logistics Representative:* IT staff process new assets by recording their details in a digital system, performing quality inspections, labeling the items, and placing them in specified warehouse locations. This organized approach to inventory management ensures precise tracking and efficient storage, similar to coordinating shipments and warehouse operations.

For *an Accountant:* The IT department handles new assets by entering their details into an asset management system, verifying their condition, tagging them for identification, and logging their storage locations. This careful documentation and tracking are crucial for maintaining accurate financial records and ensuring proper asset management, similar to tracking expenses and inventory in accounting.

4.3. Results

The evaluation reports the effectiveness of both `proc2ownw` and `proc2jargon` in adapting to stakeholder preferences and domain-specific language.

The results demonstrate that both `proc2ownw` and `proc2jargon` can generate customized outputs that meet the specific needs of stakeholders and domains. The findings also suggest that while pre-existing social network data can provide a useful foundation, exposure to relevant process models is essential for achieving the highest levels of personalization and accuracy.

Table 2 summarizes the results of training custom models (`proc2jargon` and `proc2ownw`) to paraphrase various types of process models. The evaluation involved multiple process model representations, including Object-Centric Event Logs (OCEL), and multimodal evidence (`mmevd`), Directly Follows Graphs

Table 2
Results of Model Training Across Different Process Model Types, Domains, and Personalities.

Inputs			*Domain*					*Average*
Process Inter. Type		***Prof.***	***Medical***	***Police***	***Soft. Dev.***	***Logistics***	***Account.***	
proc2jargon	*ocel2jargon*	*ABC*	90.55%	87.06%	84.01%	90.30%	87.17%	87.82%
	mmevd2jargon	*ABC*	86.92%	91.77%	85.37%	87.83%	92.80%	88.94%
	dfg2jargon	*ABC*	87.55%	85.01%	85.16%	89.10%	87.88%	86.94%
	petri2jargon	*ABC*	91.02%	85.86%	87.59%	87.57%	89.85%	88.38%
	bpmn2jargon	*ABC*	91.04%	85.47%	90.60%	90.78%	85.22%	88.62%
proc2ownw	*ocel2ownw*	*A*	84.95%	90.88%	88.01%	91.08%	88.70%	
		B	90.00%	91.16%	88.08%	91.65%	86.11%	*88.67%*
		C	89.21%	87.99%	86.56%	90.32%	85.40%	
	mmevd2ownw	*A*	88.81%	89.94%	87.00%	86.32%	84.68%	
		B	91.10%	86.97%	92.25%	89.38%	87.81%	*89.12%*
		C	86.20%	91.73%	91.86%	91.64%	91.07%	
	dfg2ownw	*A*	88.24%	89.60%	91.79%	88.10%	84.54%	
		B	85.06%	84.24%	85.47%	89.46%	88.40%	*87.42%*
		C	86.23%	84.47%	92.45%	87.46%	85.84%	
	petri2ownw	*A*	92.73%	91.41%	89.73%	92.46%	90.05%	
		B	92.05%	90.09%	91.92%	90.13%	84.23%	*90.55%*
		C	89.65%	90.16%	91.07%	91.18%	91.37%	
	bpmn2ownw	*A*	88.95%	84.52%	85.28%	92.26%	88.15%	
		B	88.03%	85.94%	84.67%	92.77%	90.13%	*88.42%*
		C	92.60%	87.24%	84.86%	92.25%	88.60%	
Average			89.04%	88.08%	88.19%	90.10%	87.90%	***88.49%***

(DFG), Petri nets, and Business Process Model and Notation (BPMN). The models were trained and tested across five domains (medical, police, software development, logistics, and accounting) and three distinct stakeholder personalities, as described earlier.

When it comes to process models representation, Table 2 shows the percentage accuracy in predicting the next token in the test conversation sets across different process model types, domains, and stakeholder personalities, indicating that the `proc2ownw` approach generally achieves higher accuracy than `proc2jargon`, particularly when representing Petri Net model, with an average accuracy of 90.55%. Accuracy refers to the percentage of correct predictions made by the model when forecasting the next token (sub-word unit) in the sequences of the test conversation sets. This accuracy was measured by comparing the predicted process-relevant token against the actual token that appeared next in the sequence. The higher the percentage, the more frequently the model correctly predicted the next token in the conversation. The relevance of each token is manually annotated based on common sense and process data description.

Beyond process models, when it comes to representation of sub-model process data (event log and multimodal evidence), Table 2 shows that models trained with multimodal evidence data (`mmevd2{jargon,ownw}`) generally perform better, achieving a higher average accuracy compared to those trained with OCEL data (`ocel2{jargon,ownw}`). This suggests that multimodal data, which includes various forms of input like images and visualizations, enhances the model's ability to adapt to stakeholder preferences and predict the next token more effectively than using OCEL alone.

Overall, the models demonstrate a robust ability to generate domain-specific, personalized outputs, with an average accuracy of 88.49% across all categories.

5. Conclusion

In this study, we explored how transformer-based architectures can be leveraged to generate paraphrased process data that adapt to the diverse linguistic preferences of stakeholders across various domains. We focused on two custom transformer models, `proc2ownw` and `proc2jargon`, to assess their ability to tailor communication effectively to different stakeholder profiles and domain-specific languages. The paper also proposes an approach that extends beyond linguistic personalization to integrate multimodal evidence—combining text and images into process models.

The evaluation results demonstrate that `proc2ownw` is particularly effective in adapting to stakeholder communication preferences, especially when tested across distinct personality profiles. On the other hand, `proc2jargon` successfully generated domain-specific language, as evaluated in fields such as medical, police, software development, logistics, and accounting.

Overall, our findings suggest that transformer-based models can improve communication in process management by tailoring outputs to both the stakeholder's linguistic preferences and the specific jargon of their domain. Additionally, integrating multimodal data into personalized process models adds another layer of complexity and opens up new opportunities for enhancing stakeholder understanding and engagement. Future work will involve further refining these models to handle more complex communication scenarios, expanding the scope of evaluation to include additional domains and stakeholder profiles, and exploring the full potential of multimodal evidence in process mining, while conducting studies and experiments with stakeholders.

References

[1] W. M. P. van der Aalst, J. Carmona (Eds.), Process Mining Handbook, volume 448 of *Lecture Notes in Business Information Processing*, Springer, 2022.

[2] Beerepoot, I. et al., The biggest business process management problems to solve before we die, Computers in Industry 146 (2023) 103837. doi:10.1016/j.compind.2022.103837.

[3] W. M. Van der Aalst, Process-aware information systems: Lessons to be learned from process mining, in: Transactions on Petri Nets and Other Models of Concurrency II, 2009, pp. 1–26.

[4] T. Fassier, E. Azoulay, Conflicts and communication gaps in the intensive care unit, Current Opinion in Critical Care 16 (2010) 654–665. doi:10.1097/MCC.0b013e32834044f0.

[5] J. D. Peters, The gaps of which communication is made, Critical Studies in Media Communication 11 (1994) 117–140.

[6] K. Busch, A. Rochlitzer, D. Sola, H. Leopold, Just tell me: Prompt engineering in business process management, in: Enterprise, Business-Process and Information Systems Modeling, Springer Nature Switzerland, Cham, 2023, pp. 3–11.

[7] OpenAI, Chatgpt, 2024. URL: https://chat.openai.com, accessed: 2024-08-01.

[8] A. Alman, A. Arleo, I. Beerepoot, A. Burattin, C. Di Ciccio, M. Resinas, Tiramisù: Making sense of multi-faceted process information through time and space, Journal of Intelligent Information Systems (2024). doi:10.1007/s10844-024-00875-8.

[9] I. Bider, E. Perjons, D. Bork, Towards on-the-fly creation of modeling language jargons, in: 17th International Conference on ICT in Education, Research and Industrial Applications, 2021, pp. 142–157.

[10] S. Yin, C. Fu, S. Zhao, K. Li, X. Sun, T. Xu, E. Chen, A survey on multimodal large language models, ArXiv abs/2306.13549 (2023).

[11] A. Gavric, D. Bork, H. Proper, Multimodal process mining, in: 26th International Conference on Business Informatics, IEEE, 2024. URL: https://model-engineering.info/publications/papers/CBI-MultiModalProcessMining.pdf, in press.

[12] H. Leopold, J. Mendling, A. Polyvyanyy, Generating natural language texts from business process models, in: Advanced Information Systems Engineering - 24th International Conference, 2012, pp. 64–79.

[13] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers

for language understanding, in: Conference of the North American Chapter of the Association for Computational Linguistics, 2019, pp. 4171–4186.

[14] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, Improving language understanding by generative pre-training, OpenAI (2018). https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf.

[15] H. Kourani, A. Berti, J. Henrich, W. Kratsch, R. Weidlich, C.-Y. Li, A. Arslan, D. Schuster, W. M. P. van der Aalst, Leveraging large language models for enhanced process model comprehension, 2024. URL: https://arxiv.org/abs/2408.08892. arXiv:2408.08892.

[16] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, J. Kang, Biobert: a pre-trained biomedical language representation model for biomedical text mining, Bioinformatics 36 (2020) 1234–1240.

[17] I. Chalkidis, M. Fergadiotis, P. Malakasiotis, N. Aletras, I. Androutsopoulos, Legal-bert: The muppets straight out of law school, arXiv preprint arXiv:2010.02559 (2020).

[18] F. Friedrich, J. Mendling, F. Puhlmann, Process model generation from natural language text, in: Advanced Information Systems Engineering: 23rd International Conference, CAiSE 2011, Springer, 2011, pp. 482–496.

[19] F. Zerbato, L. Zimmermann, H. Völzer, B. Weber, Promise: Process mining support for end-users., in: CAiSE Research Projects Exhibition, 2023, pp. 60–67.

[20] R. Girdhar, A. El-Nouby, Z. Liu, M. Singh, K. V. Alwala, A. Joulin, I. Misra, Imagebind: One embedding space to bind them all, in: CVPR, 2023.

[21] T. Fehrer, A. Egger, D. Chvirova, J. Wittmann, N. Wördehoff, W. Kratsch, M. Röglinger, Business Processes in IT Asset Management Multimedia Event Log, 2024. doi:10.6084/m9.figshare.25246291.

[22] A. Berti, W. M. P. van der Aalst, OC-PM: analyzing object-centric event logs and process models, International Journal on Software Tools for Technology Transfer 25 (2023) 1–17. doi:10.1007/s10009-022-00668-w.

[23] A. Gokaslan, V. Cohen, Openwebtext corpus, http://Skylion007.github.io/OpenWebTextCorpus, 2019.

[24] S. V. Daele, G. Janssenswillen, Identifying the steps in an exploratory data analysis: A process-oriented approach, in: Process Mining Workshops, 2023, pp. 526–538.

[25] S. Esser, D. Fahland, Multi-dimensional event data in graph databases, 2021. URL: https://arxiv.org/abs/2005.14552. arXiv:2005.14552.