**LEVEL 0 SUMMARY**

* **Name of student:** Reckia Jiffard
* **Name of your Level 1:** Mohamed Anissat
* **Source (e.g. scholars.google.com):** Google scholar
* **Paper title:** Hybrid business process modeling for the optimization of outcome data
* **Keywords specific to the paper:** “business process modeling” “deep learning”
* **Summary of the main contributions (Use text paragraphs, tables and if necessary, figures):**

Introduction Business processes can be specified using either imperative or declarative modeling languages. Imperative languages explicitly specify the allowed execution sequences of activities at design time while declarative languages describe properties that must hold without specifying how. However, existing declarative languages have limitations when it comes to:

* Only specifying activity ordering
* Optimizing only execution time/resources
* Lack of executable models in BPMS.

This paper aims to address these limitations by formalizing a hybrid model oriented towards optimizing outcome data and automatically generating an executable imperative model. The hybrid model combines a data-oriented declarative specification with an imperative control-flow specification. It is applied to a "trip planner" case study where the goal is finding input data that optimizes the total price outcome based on customer criteria. Formalizing the Hybrid Model A new Data-Oriented Optimization Declarative (DOOPT-DEC) language is introduced to describe the model. It is divided into a subprocess description specifying activities/gateways imperatively and subprocess relationships specifying data relationships declaratively.

The description defines activity input/output data sets, individual activity input/output data, and the union of all activities’ data forming the total input/output data sets. Pre/post-conditions describe how data is modified by activities. The relationships specify constraints between data attributes flowing between activities. Constraint programming (CP) is used to represent the declarative part as it allows modeling with variables and constraints. Generating an Imperative ModelAn entirely imperative model is automatically created by including a constraint optimization problem (COP) as a script activity.

The COP contains all variables, domains, constraints from the declarative part plus an objective function. It is solved at runtime to find optimal input data, avoiding executing "white-box" activities where outputs can be directly obtained from inputs/constraints. The COP is integrated into a standard process model by configuring the COP-containing script activity for the chosen BPMS. Application to a Trip Planner The paper explains the trip planner case study modeled in DOOPT-DEC, presenting activities for flight, hotel, and car rental searching. The total price is minimized as the objective function. Two possible resulting imperative models are described - a "white-box" approach using the COP if all activities are like brochures precisely defining prices, and otherwise a "black-box" approach executing all activities to find optimized outputs.

The approach was tested on the trip planner using the Choco CP solver configured as a Bonita Open Solution script task. Memory/time tests showed efficiency scaling linearly with problem size as expected for constraint reasoning. The paper formalizes a hybrid model approach combining declarative data optimization specifications and imperative control-flow for problems with unknown inputs needing runtime optimization. An entirely executable imperative model is automatically generated from this specification. The approach addresses limitations of existing declarative languages.

* **AI model used (e.g. Neural network, etc.)**

This paper focus on the development of a new Data-Oriented Optimization Declarative (DOOPT-DEC) language.

* **Introduce the AI models**

The Data-Oriented Optimization Declarative (DOOPT-DEC) language is a key focus of the paper. It is introduced as a formalism to enrich imperative business process models by incorporating a declarative description of the relationships between data. The language is designed to address the limitations of existing declarative languages by providing a means to specify data-oriented optimization problems within business processes. DOOPT-DEC is divided into subprocess description and subprocess relationships, allowing for the detailed description of components associated with imperative activities and the declarative description of relationships between components through data-flow. The language aims to enable the formalization of a hybrid model that represents the outcome data optimization requirements in a declarative way, which can be included in an imperative model representing the control-flow requirements. Additionally, DOOPT-DEC facilitates the automatic creation of an imperative model from a declarative description, thereby providing a comprehensive alternative for the specification of declarative requirements within imperative descriptions for data-oriented optimization problems in business processes.

* **How do they contribute the idea proposed by the paper?**

The Data-Oriented Optimization Declarative (DOOPT-DEC) language contributes significantly to the idea proposed by the paper by addressing the limitations of existing declarative languages and providing a formalism to enrich imperative business process models with a declarative description of data relationships. Specifically, the language enables the formalization of a hybrid model that represents the outcome data optimization requirements in a declarative way, which can be included in an imperative model representing the control-flow requirements. This addresses the need to describe unknown input values of data, the objective function according to the data output of activities, and how input values in an activity can affect other activities.

Furthermore, DOOPT-DEC facilitates the automatic creation of an imperative model from a declarative description, providing a comprehensive alternative for the specification of declarative requirements within imperative descriptions for data-oriented optimization problems in business processes. By introducing this language, the paper aims to provide a new point of view of declarative languages focused on data, thereby contributing to the development of a more comprehensive approach to business process modeling and optimization.

**Supported by a software application? (If yes, provide more details)**

It is not currently supported by any software application.