GPT-3 for Decision Logic Modeling

Alexandre Goossens[1,3,*,†], Simon Vandevelde[2,3,4,†], Jan Vanthienen[1,3] and Joost Vennekens[2,3,4]

[1]KU Leuven, FEB, Research Centre for Management Informatics (LIRIS)
[2]KU Leuven, De Nayer Campus, Dept. of Computer Science, Belgium
[3]Leuven.AI – KU Leuven Institute for AI, B-3000 Leuven, Belgium
[4]Flanders Make – DTAI-FET

Abstract
Operational decisions are an important part of knowledge-intensive organizations, as these are taken in a high volume on a daily basis. For this purpose, the Decision Model and Notation (DMN) standard describes an intuitive and user-friendly notation to model, communicate and execute business decisions. However, manually modeling the decisions in DMN remains a time-consuming task, as various textual sources need to be analyzed. As such, automation of decision modeling is beneficial for the decision process. This work investigates an automated approach to generating decision tables from natural language based on the GPT-3 large language model. Through a total of 72 experiments over six problem descriptions, this work evaluates GPT-3's decision logic modeling and reasoning capabilities. While GPT-3 demonstrates promising abilities in extracting decision context and identifying relevant variables from natural language, further enhancements are needed to improve its decision table capabilities for efficient automation of DMN modeling.

Keywords
Decision Modeling, DMN, GPT-3, Large Language Models

1. Introduction

Organizations have to abide by internal and external rules to execute various operational decisions, such as deciding loan eligibility or approving driving licences. These decision rules are often stored in various texts distributed across internal and sometimes external documents [1]. As such, a lot of time and effort is put into managing such decision rules by creating manuals and guidelines. However, these manuals and guidelines are often too complex for new stakeholders. To overcome this latter issue, the Object Management Group (OMG) publishes the Decision Model and Notation (DMN) standard. This standard aims to assist at modeling, executing and communicating decisions in a user-friendly manner without losing relevant information about the operational decisions [2]. Containing both the decision structure and decision logic, DMN

RuleML+RR'23: 17th International Rule Challenge and 7th Doctoral Consortium, September 18–20, 2023, Oslo, Norway
*Corresponding author.
†These authors contributed equally.
alexandre.goossens@kuleuven.be (A. Goossens); s.vandevelde@kuleuven.be (S. Vandevelde); jan.vanthienen@kuleuven.be (J. Vanthienen); joost.vennekens@kuleuven.be (J. Vennekens)
0000-0001-8907-330X (A. Goossens); 0000-0001-7312-3675 (S. Vandevelde); 0000-0002-3967-7055 (J. Vanthienen); 0000-0002-0791-0176 (J. Vennekens)

models can be employed in numerous manners, ranging from communicating rules to automatic decision execution.

Decision logic in DMN is mainly stored in decision tables, which offer an intuitive representation. Such tables can also straightforwardly be verified w.r.t. their consistency and completeness [3, 4, 5]. However, manually converting decision rules from a textual format to decision tables is a time-consuming and arduous process, which could benefit from a (semi-)automatic approach. While this has been done in the past for structured documents [6] and single sentences [7], extracting decision tables from realistic textual descriptions remains challenging. It is here that novel Natural Language Processing (NLP) technologies such as Generative Pre-trained Transformers-3 (GPT-3) [8] might be the key to closing this gap, as they have already proven themselves to perform excellently on a plethora of other NLP tasks [8].

This paper investigates to what extent GPT-3 lends itself to automatically discovering decision logic from textual descriptions, formulating it into comprehensive decision tables and reasoning on such decision tables. To evaluate this, a dataset of various decision descriptions has been collected and provided to GPT-3. Various questions were then asked to GPT-3 in relation to decision logic and decision tables. In total, 72 experiments were conducted on this dataset.

This paper is structured as follows: Section 2 introduces DMN and GPT-3, followed by Section 3 which deals with the related work. Next, Section 4 elaborates on research questions and the set-up of the experiments whilst Section 5 reports and discusses the results. Lastly, Sections 6 and 7 respectively discuss the limitations and conclude the paper with future work and final remarks

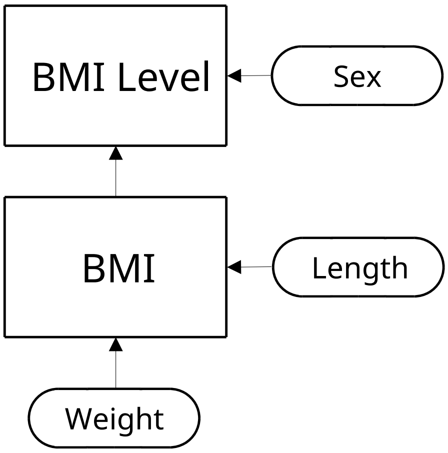
2. Preliminaries

2.1. Decision Model and Notation

DMN [2] is a standard for (business) decision modeling published and maintained by the OMG group. It is designed "to be used by anyone involved in the decision modeling process", and therefore aims to be as user-friendly and intuitive as possible. A DMN model consists of two main components: the Decision Requirements Diagram (DRD), and decision tables.

The DRD is a graph representing a DMN model's structure. It shows which decisions should be made, what input information is required, where the data comes from, and more. Fig. 1a shows an example of a DRD with two decisions (as represented by the rectangles) and three inputs (as represented by the ovals). Decisions follow the direction of the arrows: here, the model first decides the *BMI* based on *Length* and *Weight*, followed by *BMI Level* based on *BMI* and *Sex*.

The second component of DMN, decision tables (as part of the Friendly Enough Expression Language (FEEL), are a structured representation of decisions. As an example, Fig 1b shows the decision tables for *BMI* and *BMI Level*. Such a table defines the value of the output variable (right, in blue) based on the value of the input variable(s) (left, in green). Each row of the table represents a decision rule, which is *applicable* when the values in the input cells correspond to those of the input variables. For example, if *BMI = 22* and *Sex = Female* then the fourth rule would be applicable, leading to a *BMI Level* of *Normal*. When multiple rules are applicable, the behavior of a table is defined by its *hit policy*. In this paper, we limit ourselves to the U(nique)



(a) Example DRD

BMI			
U	Weight	Length	BMI
1	—	—	Weight/(Length*Length)

BMI Level			
U	BMI	Sex	BMI Level
1	< 18.5	Male	Severely Underweight
2	< 18.5	Female	Underweight
3	$[18.5..25)$	Male	Underweight
4	$[18.5..25)$	Female	Normal
5	$[25..30)$	Male	Normal
6	$[25..30)$	Female	Overweight
7	> 30	—	Obese

(b) Example decision tables

Figure 1: Example of a DMN model to decide a BMI Level.

hit policy. This policy denotes that rules must be exclusive and may not overlap, thereby preventing multiple rows from being applicable for the same set of input values. Furthermore, it is recommended that a decision table is *complete*, in the sense that it has an applicable rule for each possible set of input values.

2.2. GPT-3

GPT-3 [8] is a *Large Language Model* developed by OpenAI, aimed at NLP-applications with sequential text. It is the third iteration of OpenAI's *Generative Pre-trained Transformers* (GPT), which are neural networks that use a transformer architecture [9][1] and have been trained on a large corpus of text. GPT-3 contains an enormous 175 billion parameters, which have been tweaked by training on a large data set consisting of internet texts, two book datasets, Wikipedia, and more.

GPT-3's target use is *text completion*, in which it is given a string (also known as prompt) and tasked to generate a relevant continuation of it. For example, if we give GPT-3 the string "A summary of the Decision Model and Notation standard in 3 sentences is as follows:", it generates the following.

> "The Decision Model and Notation (DMN) is a modeling language and notation designed to represent and analyze business decisions. It provides a graphical representation of decision logic that is easily understandable by both technical and non-technical stakeholders. DMN is based on the Unified Modeling Language (UML) and Business Process Model and Notation (BPMN) standards, making it easy to integrate with other business modeling tools."

[1]Technically speaking, they only use the *decoder* part of a transformer[10]

The generation of words is dictated by probabilities, in the sense that GPT-3 tries to predict the word that is the most likely to appear next to the string. This process of predicting words is repeated until it predicts a *string end*. Due to this probabilistic approach, GPT-3 and other LLMs are sometimes referred to as *stochastic parrots* [11]: they predict the most probable word, but likely do not *understand* their meaning. This leads to problems such as *hallucinations* [12], in which the network "hallucinates" false information without any grasp of reality.

The success of GPT has sparked a rise in research on its applications, ranging from textual operations such as text summarization [13], generating stories [14] and generating patents [15] to seemingly unrelated operations such as statutory reasoning [16]. This burst of interest is caused by the phenomenon of *emergent intelligence*, which dictates that neural networks trained on very large amounts of data can "suddenly" be capable of tasks previously thought impossible.

The most well-known example of GPT-3 currently in use is ChatGPT [17], a chatbot which has taken the world by storm. It is the fastest-growing application in history, with already 100 million active users two months after launch [18]. It has shown impressive capabilities as a chatbot by having realistic conversations (up to an extend). Another example of an LLM in use is Copilot [19], a network trained specifically to generate code which, depending on the programming language, has a correctness of 27%-57% on small coding puzzles [20].

3. Related work

Automatically extracting decision rules from text has already been investigated in the past with the first approaches focusing on a syntactic analysis to extract decision rules from well-structured documents [21, 22]. Next, the focus shifted from a syntax analysis to using dependency trees, as these are more robust against sentence variations and can handle less structured documents [23]. For example, in [24] the authors report how they use such dependency trees to extract Unified Modeling Language (UML) diagrams from use cases. Legal documents are often also considered prime candidates for rule extraction. The authors of [25] used Convolutional Neural Networks (CNNs) to automatically classify legal rules into classes either describing legal definitions, prohibitions, obligations or permissions. Next, the automatic retrieval of rules from legal documents using patterns and deep learning has been investigated in [26] and [27] respectively. The recent introduction of GPT-3 has also seen a plethora of applications to support users with managing, generating and understanding decision rules with LawGPT assisting in answering legal questions and generating legal documents [28] or being used to generate captions on radiology images [29].

It is important to note that the previous approaches did not focus on the relation between decision rules and their interaction. The next paragraph focuses on works extracting rules in a decision table format or DMN format allowing for a better overview of all the rules. The authors of [6] investigated the extraction of decision tables from Semantics Of Business Vocabulary And Business Rules (SBVR) documents [30] which are structured textual documents. Extracting decision tables from unstructured single sentences has been investigated in [7]. With the use of Natural Language Processing (NLP) and the identification of recurring patterns, the extraction of DRDs and DMN models has been investigated in [31] and [32], both use an artificial dataset. In [33], the use of deep learning models has been investigated to automatically extract DRDs and logical statements from a textual description. This approach uses a pipeline combining sentence classifiers and BERT models finetuned for Named Entity Recognition (NER) tasks to find the relevant parts of a sentence. The approach does not find complete decision rules but rather relevant parts describing logic, and as such, does not generate DMN models directly.

4. Research questions and methodology

This section elaborates on the methodology for this study. The general aim of the experiments is to look at three aspects:

1. Can GPT-3 identify the relevant information items of a decision together with their values?
2. Can GPT-3 construct correct, complete and overlap-free decision tables?
3. Is GPT-3 able to correctly reason with decision tables?

This will be investigated by prompting GPT-3 with a set of questions related to decision logic and decision table construction. In total, six problem descriptions will be given to GPT-3. This section introduces the dataset used for the evaluation followed by the experimental set-up and prompts given to GPT-3. Next, the settings of GPT-3 are explained, after which the evaluation metrics are introduced.

4.1. Dataset

Table 1 summarizes the decision descriptions used for the experiments, together with their word length. The BMI-level description was taken from [31], and the holidays problem was taken from a Decision Management Community challenge[2]. The remaining descriptions are synthetic but realistic, and are in a similar vein as the BMI-level and the holidays problems. A collection of all descriptions is available online[3].

As an example, the BMI textual description reads as follows:

> "The BMI of a person can be calculated based on weight in kgs and length of a person in meters by using the following formula: weight/(length*length). If the BMI value is above 30, then the BMI-level is considered Obese. If you are a male and the BMI-value is under 18.5 then the BMI-level is severely underweight and if you are female, then you are considered underweight with the same bmi-value. If the BMI-value is between 18.5 and 25 (without 25), then the BMI-level is underweight for a male and normal for a female. Lastly, if the BMI-value is between 25 and 30 and you are a Male then the BMI-level is normal but if you are a female then BMI-level is overweight."

[2]https://dmcommunity.org/challenge/challenge-jan-2016/
[3]https://gitlab.com/EAVISE/sva/GPT-DMN

Table 1
Dataset Description

Problem	**Topic**	**Length**
BMI-level	Describes how BMI should be calculated based on length, weight and gender.	123
Pet	Describes which pet a family will take in depending on allergies, kids, garden and free time.	191
Driver's License	Describes when a person receives their driver's license based on age, nationality and a practical test.	100
Holidays	Describes how many holidays a person is entitled to based on age and years of service.	113
Scholarship Eligibility	Describes scholarship eligibility based on grades, annual income and previous scholarships.	88
Transportation mode	Describes which mode of transport should be used depending on the weather, distance and whether a person is in a hurry.	225

4.2. Questions

In the experiments, GPT-3 is given a decision description and asked to answer the following nine questions:

Q1 *Below is a decision description. What does this description decide?* **«Problem Description»**

Q2 **«Test1»**

Q3 *What are the variables that influence this decision?*

Q4 *For each input and output, give me an overview of their data type and their possible values.*

Q5 *What are the relevant values of the numerical variables?*

Q6 *Could you generate a DMN decision table for this description? Make sure the table can be read horizontally: the column headers contain the inputs and output.*

Q7 *Make the rules mutually exclusive.*

Q8 *Is this table complete? (I.e., is there an applicable rule for each set of inputs?) If it is incomplete, can you find an example for which no rule would be applicable?*

Q9 *According to your table, answer the following question.* **«Test2»**

Q1, Q3, Q4 and **Q5** aim to measure whether GPT-3 understands what a decision is about, which variables are relevant and what values influence the decision. **Q6, Q7** and **Q8** ask GPT-3 to construct a decision table, make its rules mutually exclusive (free from overlap) and verify if the table is complete. **Q2** and **Q9** query GPT-3 to make a decision based on the problem description (Q2) or its own generated decision table (Q9), as indicated by **Test**. These tests have been created manually for each prompt. For example, the following questions are asked for the BMI description:

Test1: What is the BMI-level of a male of 1.76m weighing 68kg?

Test2: What is the BMI-level of a girl of 1.4m weighing 42kg?

Note that **Q2** asks GPT-3 to answer the question using the description whilst **Q9** asks GPT-3 to answer the question using the decision table GPT-3 generated.

The motivation behind this structured question-and-answer approach is two-fold. Firstly, carefully formulated questions ensure GPT-3 will produce desirable answers with a higher frequency [34, 35]. Secondly, the approach allows for a clear difference between the experiment questions and the answers generated by GPT-3. Having this split is important, as the previous questions and answers are always included in a single string as context when asking a new question, and GPT-3 needs to be able to interpret it correctly.

4.3. GPT-3 settings

The experiments in this work are performed by a Python program[4] which communicates with the official GPT-3 API. To tweak GPT-3's behaviour, this API supports setting multiple parameters, such as *model*, *temperature*, and *top_p*. As these parameters can greatly influence the output of the system, it is important that they are set correctly. On their websites, OpenAI include the optimal settings for a number of example applications, including for "creating tables from long-form text"[5]. By using these values, the experiments should allow for a correct representation of GPT-3's decision table capabilities.

Table 2 shows an overview of the settings used in this work. Note that there is a deviation from the suggested values for two parameters: *max_tokens* and *temperature*. The paramater *max_tokens* denotes the number of *tokens* the model is allowed to generate. To ensure that GPT-3's output does not cut-off in the middle of an answer, this parameter was set to 2048. The *temperature* setting refers to the model's sampling temperature, i.e., how much freedom is allowed when generating answers. A higher temperature makes the output less deterministic, with more room for "creativity". To gauge the effect of this creativity on GPT-3's decision capabilities, the experiments are performed using four different temperature values: 0, 0.3, 0.7, and 1.

4.4. Evaluation metrics

Following the completion of all 72 experiments, comprising of six examples assessed across three iterations for each of the four temperatures, each answer to a question was evaluated on a

[4] **https://gitlab.com/EAVISE/sva/GPT-DMN**

[5] https://platform.openai.com/examples/default-parse-data – though this has recently been modified. At the time of the experiments, the page suggested a presence penalty of 0.6. However, we expect its impact to be minimal.

Table 2
GPT-3 settings

Parameter	Value
model	text-davinci-003
max_tokens	2048
temperature	{0, 0.3, 0.7, 1}
top_p	1
frequency penalty	0.0
presence penalty	0.6

pass/fail basis based on the ground truth. Only answers that are completely correct are counted as "pass". If an answer is partially correct, such as listing some but not all variables for Q2, the answer is counted as incorrect. The rationale for this "strict" grading is to make the evaluation more straightforward, and to remove doubts about what counts as correct.

As an exception, Question 6 underwent a more in-depth evaluation. Here, we evaluated the table on four dimensions of correctness:

1. Does the table contain the correct input variables?
2. Does the table contain the correct output variable?
3. Does the table contain at least one correct rule?
4. Is the table a correct decision table for the problem?

To ensure the accuracy and reliability of the evaluation process, both first authors independently evaluated all responses, and cross-evaluated their findings afterwards. In instances where potential conflicts arose, a consensus was reached through discussions between the two authors. This approach ensured that the evaluation process was comprehensive and rigorous, thus, enhancing the reliability of the findings.

5. Results & discussion

This section presents the results of our experiments. These results are two-fold: first, the generated scores are presented, after which some interesting observations from the experiments are laid out.

5.1. Scores

Table 3 and Table 4 show the average evaluation scores and standard deviation, sorted by temperature. There are a few interesting take-aways from this data. To begin with, querying GPT-3 with a temperature of 1 resulted in the highest accuracy on questions Q2-Q4, and the second-highest accuracy on Q5. On the other hand, lower temperatures seem to perform better on the last three questions. A possible explanation for this observation is related to the nature of the questions: these first questions require a certain degree of interpretation in the

Table 3
Accuracy and standard deviation for Q1-Q5, Q7-Q9

Temp	Q1	Q2	Q3	Q4
0.0	**1.00** $\pm$ 0.00	0.78 $\pm$ 0.40	0.67 $\pm$ 0.52	0.17 $\pm$ 0.41
0.3	**1.00** $\pm$ 0.00	0.83 $\pm$ 0.41	0.72 $\pm$ 0.44	0.28 $\pm$ 0.25
0.7	**1.00** $\pm$ 0.00	0.78 $\pm$ 0.40	0.83 $\pm$ 0.28	0.44 $\pm$ 0.27
1.0	**1.00** $\pm$ 0.00	**0.89** $\pm$ 0.17	**0.89** $\pm$ 0.27	**0.50** $\pm$ 0.41

Temp	Q5	Q7	Q8	Q9
0.0	0.33 $\pm$ 0.52	**0.67** $\pm$ 0.42	**0.67** $\pm$ 0.42	0.44 $\pm$ 0.50
0.3	0.44 $\pm$ 0.46	0.56 $\pm$ 0.34	0.50 $\pm$ 0.35	**0.50** $\pm$ 0.35
0.7	**0.56** $\pm$ 0.40	0.33 $\pm$ 0.42	0.56 $\pm$ 0.40	0.33 $\pm$ 0.37
1.0	0.50 $\pm$ 0.41	0.50 $\pm$ 0.35	0.39 $\pm$ 0.25	0.39 $\pm$ 0.33

Table 4
Accuracy and standard deviation for Q6.1-Q6.4

Temp	Q6.1	Q6.2	Q6.3	Q6.4
0.0	0.56 $\pm$ 0.50	0.89 $\pm$ 0.27	0.39 $\pm$ 0.49	0.00 $\pm$ 0.00
0.3	0.67 $\pm$ 0.42	**0.94** $\pm$ 0.14	**0.61** $\pm$ 0.39	**0.06** $\pm$ 0.14
0.7	0.72 $\pm$ 0.44	0.89 $\pm$ 0.27	**0.61** $\pm$ 0.44	**0.06** $\pm$ 0.14
1.0	**0.78** $\pm$ 0.34	0.89 $\pm$ 0.17	0.56 $\pm$ 0.40	**0.06** $\pm$ 0.14

identification of variables and their values, so a higher temperature might perform better. In contrast, answering Q7-Q8 does not require creativity, but rather a precise format for soundness and completeness.

Regardless of temperature, the results are a bit underwhelming, but show potential. While GPT-3 always correctly answers Q1, the correctness for the other questions varies wildly. In fact, of all 72 experiments, it only managed to create a correct decision table in three cases, once for each temperature between 0.3 and 1.0 (see Q6.4, Table 4). Overall, GPT-3 reaches the best results in the questions about the variables present in a decision model (Q3, Q4, Q6.1-2).

Another interesting result is that the standard deviation for *t=1* is typically lower in our results. While this might seem counter-intuitive, as a higher temperature leads to more non-determinism in the output of GPT-3, it is important to remember that the evaluation deals with the deviation on the *correctness* of the answers. Two answers that are formatted differently may still both be correct. A higher temperature results in more consistent correctness, whereas a lower temperature can have more varied results across different prompts.

An example of a complete "run" of all questions and their answers is shown at the end of this paper. All generated output files and the evaluation spreadsheet are also available online[6].

[6]https://gitlab.com/EAVISE/sva/GPT-DMN

5.2. Discussion

Throughout the 72 experiments, GPT-3 demonstrated some interesting behaviour that is worth highlighting.

Maths For the "BMI" description, GPT-3 was asked to derive a BMI level based on weight and height, requiring the calculation of the BMI. In all cases, this calculation was incorrect. Moreover, GPT-3 also had difficulties with comparisons. For example, in one experiment it output the following for a height of 1.4m and a weight of 42kg (BMI = 21.4):

> "The BMI-Level would be severely underweight, since the BMI value would be 30.3, which is below 18.5"

Own definition of concepts It is very difficult to know what GPT-3 exactly knows about DMN and whether all that knowledge is actually correct. It is known that GPT-3 has been trained on data prior to June 2021[7], but which information exactly has been fed to GPT-3 regarding DMN and whether all this information is correct, is unknown.

This is also the case for concepts introduced in the descriptions. For example, the definition of BMI level in our experiments is purposefully incorrect in our example prompt: e.g., men with a BMI between 18.5 and 25 are considered to be underweight instead of having a normal BMI level. However, in almost all examples, GPT-3 still states that a BMI between 18.5 and 25 is considered normal for men. Here, it is most likely using information it learned from previous data, instead of using the description that it was given. This behaviour is risky, as it is not clear when GPT-3 will use its own definition of a concept instead of the one its given by a user.

Hallucinating missing rules When presented with inputs for which no rules match, GPT-3 will hallucinate a possible output. In trying to explain this output, it will make up a rule on the spot.

Completeness When asked if a table is complete (Q8), GPT-3 answered *Yes* in 70 out of 72 cases. In reality however, the generated tables were only complete about 42% of the time, and even this was often only due to an abundant usage of *any value* (—) in the cells.

In one of the two cases in which GPT-3 correctly responded that the table was incomplete, it then proceeded to give an example for which the table did actually have an applicable row.

Inferring implicit values The "Scholarship" example states that "a student is only eligible if their grades are excellent or good". Based on this, a human modeler can infer that there are at least three values for a grade: "excellent", "good", and "bad". Yet, GPT-3 never included a third value at temperatures 0 and 0.3, and only included it half of the time at temperatures 0.7 and 1.

[7] https://platform.openai.com/docs/models/gpt-3-5

Misjudging relevance In "Transport", Q2 asks for the best method of travel for a distance of 15 kilometres. In 8 out of 12 answers, this caused GPT-3 to later list 15km as a relevant value in Q4-Q6, even though it does not appear in the description itself.

General Remarks An interesting observation is that GPT-3 seems to perform well on answering questions using the textual description (**Q2**) whilst at the same time performing poorly on the identification of relevant input and output variables (**Q4, Q5, Q6**). This is a good indication that GPT-3 does not really understand the reasoning driving a decision but is simply using probabilities to create sentences that make sense using the provided description. This is why GPT-3 has a tendency to reformulate answers in the same way as the provided description, since that formulation is the one with the highest probability. Another important side note is that the provided textual descriptions are pretty small and not comparable to large organizational or legal decisions which encompass many more pages. Therefore, the encouraging GPT-3 results might not be valid for larger texts.

6. Limitations

While the methodology of this paper was carefully designed, it is not without its limitations. Firstly, GPT-3 was evaluated with zero-shot learning, i.e., without being given any examples. Because GPT-3 already performs well on various other NLP tasks with zero-shot learning [8], this could reasonably have been expected to work well. However, GPT-3's poor performance on this task with zero-shot learning does not mean that it might not perform better in a different setting.

Secondly, the output generation of GPT-3 is inherently random: even with a temperature of zero, the output can still vary greatly. In this work, this randomness was addressed by running each description and temperature three times. Moreover, to prevent any interpretation errors, the results were cross-validated making the results sufficiently reliable despite the non-deterministic output.

Thirdly, the newest iteration of GPT (GPT-4) has recently come out, which is reported to have many improvements over GPT-3. Unfortunately, the GPT-4 API is not publicly accessible at the time of writing.

7. Conclusion and future work

This paper started with the observation that operational rules are often captured in textual documents whilst decision tables are an effective method for managing and ensuring the correctness and completeness of operational rules. However, creating decision tables can be a time-consuming process. This paper explored the potential of GPT-3 for understanding, modeling, and reasoning with decision tables based on decision descriptions. The results show that GPT-3 can understand the decision context, identify the input and output variables, and provide template decision tables for problem-solving. Nevertheless, GPT-3 cannot create complete and correct decision tables as it lacks an understanding of critical concepts such as mutual exclusivity and completeness.

Future work in this area could include exploring the potential of GPT-3 for constructing complete DMN models and investigating whether fine-tuning GPT-3 for decision table construction would improve performance. Further experimentation with various settings would be interesting to provide an insight into which values provide the best results. Another very interesting research path would be to conduct a more in depth study about prompt optimization for the specific problem of decision logic analysis and reasoning since the various formulations and provided contexts can greatly influence the results achieved with GPT [34, 35]. It would also be valuable to examine whether few-shot learning techniques would perform better than existing models, as well as evaluate the potential of GPT-4[8] for decision table construction.

8. Full example of queries

To illustrate the approach of this work, this section shows one full run for the vacation description at temperature = 1. It has been edited slightly for readability's sake, including converting GPT-3's markdown table output into LaTeX. This output goes as follows:

We will ask you a series of questions on Decision Model and Notation tables. Each question starts with "Q:", and each response should start with "A:". Below is a decision description, between quotation marks.

"Every employee receives at least 22 days of vacation. Additional days are provided according to the following criteria: only employees who are younger than 18 or at least 60 years old, or employees with at least 30 years of service will receive 5 extra days. Employees with at least 30 years of service and also employees of age 60 or more, receive 3 extra days, on top of possible additional days already given. If an employee has at least 15 but less than 30 years of service, 2 extra days are given. These 2 days are also provided for employees of age 45 or more. These 2 extra days can not be combined with the extra 5 days. "

Q: What does this description decide?

A: *This description decides how many vacation days each employee will receive based on their age, with additional days given depending on the employees' age and years of service.*

Q: How many holidays does a 32 year old receive if they have 6 years of service?

A: *The 32 year old would receive 28 days of vacation, as they do not meet the criteria for additional holidays.*

Q: What are the variables that influence this decision?

A: *The variables that influence this decision are the employee's age and years of service.*

Q: For each input and output, give me an overview of their data type and their possible values.

A: *The input data type is "integer" and the possible values are any number between 0 and 60 (age) or 15 and 30 (years of service). The output data type is "integer" and the possible values range from 22 to 33 (number of vacation days).*

Q: What are the relevant values of the numerical variables?

A: *The relevant values of the numerical variables are the age (less than 18 or 60 or more) and the years of service (at least 30).*

Q: Could you generate a DMN decision table for this description? Make sure the table can be read horizontally: the column headers contain the inputs and output.

U	Age	Years of Service	Days
1	< 18	> 30	5
2	≥ 60	> 30	8
3	≥ 45	[15..30)	2
4	$-$	< 15	22

Q: Make the rules mutually exclusive.

[8] https://openai.com/product/gpt-4

U	Age	Years of Service	Days
1	< 18	> 30	5
2	≥ 60	> 30	8
3	≥ 45	$[15..30)$	2
4	$[15..60)$	< 15	22

Q: Is this table complete? (I.e., is there an applicable rule for each set of inputs?) If it is incomplete, can you find an example for which no rule would be applicable?

A*: This table is complete, as there is an applicable rule for every set of inputs.*

Q: According to your table, answer the following question. How many holidays does a 64 year old receive if they have 32 years of service?

A*: According to the table, the 64 year old would receive 11 days of vacation.*

Acknowledgements

This research received funding from the Flemish Government under the "Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen" programme. This work was also supported by the Fund for Scientific Research Flanders (project G079519N) and KU Leuven Internal Funds (project C14/19/082).

References

[1] J. Vanthienen, Decisions, advice and explanation: an overview and research agenda, in: A Research Agenda for Knowledge Management and Analytics, Edward Elgar Publishing, 2021, pp. 149–169.

[2] Object Management Group, Decision model and notation v1.3, 2021-02, 2021.

[3] J. Vanthienen, C. Mues, A. Aerts, An illustration of verification and validation in the modelling phase of kbs development, Data & Knowledge Engineering (1998).

[4] J. Huysmans, K. Dejaeger, C. Mues, J. Vanthienen, B. Baesens, An empirical evaluation of the comprehensibility of decision table, tree and rule based predictive models, Decision Support Systems 51 (2011) 141–154.

[5] S. Vandevelde, B. Callewaert, J. Vennekens, Context-aware verification of DMN, Proceedings of the 55th Hawaii International Conference on System Sciences, 2021.

[6] K. Kluza, K. Honkisz, From SBVR to BPMN and DMN models. proposal of translation from rules to process and decision models, in: International Conference on Artificial Intelligence and Soft Computing, Springer, 2016, pp. 453–462.

[7] L. Arco, G. Nápoles, F. Vanhoenshoven, A. L. Lara, G. Casas, K. Vanhoof, Natural language techniques supporting decision modelers, Data Mining and Knowledge Discovery 35 (2021) 290–320.

[8] T. Brown, et al., Language models are few-shot learners, Advances in neural information processing systems 33 (2020) 1877–1901.

[9] A. Vaswani, et al., Attention is all you need, in: Advances in Neural Information Processing Systems, volume 30, Curran Associates, Inc., 2017.

[10] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, et al., Improving language understanding by generative pre-training (2018).

[11] E. M. Bender, T. Gebru, A. McMillan-Major, S. Shmitchell, On the dangers of stochastic parrots: Can language models be too big?, in: ACM Conference on Fairness, Accountability, and Transparency, ACM, New York, 2021, pp. 610–623.

[12] J. Maynez, S. Narayan, B. Bohnet, R. T. Mcdonald, On faithfulness and factuality in abstractive summarization, in: 58th Annual Meeting of the ACL, 2020.

[13] V. Kieuvongngam, B. Tan, Y. Niu, Automatic text summarization of covid-19 medical research articles using bert and gpt-2, 2020. arXiv:2006.01997.

[14] J. Guan, F. Huang, Z. Zhao, X. Zhu, M. Huang, A Knowledge-Enhanced Pretraining Model for Commonsense Story Generation, Transactions of the Association for Computational Linguistics 8 (2020) 93–108.

[15] J.-S. Lee, J. Hsiang, Patent claim generation by fine-tuning OpenAI GPT-2, World Patent Information 62 (2020) 101983.

[16] A. Blair-Stanek, N. Holzenberger, B. V. Durme, Can GPT-3 perform statutory reasoning?, 2023. arXiv:2302.06100.
[17] OpenAI, Chatgpt, https://github.com/openai/gpt-3, 2020.
[18] A. R. Chow, How ChatGPT managed to grow faster than TikTok or instagram, TIME (2023).
[19] GitHub, GitHub Copilot: Your AI pair programmer, 2021.
[20] N. Nguyen, S. Nadi, An empirical evaluation of GitHub copilot's code suggestions, in: Proceedings of the 19th International Conference on Mining Software Repositories, MSR '22, ACM, New York, NY, USA, 2022, pp. 1–5.
[21] E. Riloff, Automatically generating extraction patterns from untagged text, in: Proceedings of the national conference on AI, 1996, pp. 1044–1049.
[22] S. Soderland, Learning information extraction rules for semi-structured and free text, Machine learning 34 (1999) 233–272.
[23] D. Lin, P. Pantel, Discovery of inference rules for question-answering, Natural Language Engineering 7 (2001) 343–360.
[24] P. Danenas, T. Skersys, R. Butleris, Natural language processing-enhanced extraction of sbvr business vocabularies and business rules from uml use case diagrams, Data & Knowledge Engineering 128 (2020) 101822.
[25] M. Michel, D. Djurica, J. Mendling, Identification of decision rules from legislative documents using machine learning and natural language processing., in: HICSS, 2022, pp. 1–10.
[26] M. Dragoni, S. Villata, W. Rizzi, G. Governatori, Combining NLP Approaches for Rule Extraction from Legal Documents, in: 1st Workshop on MIning and REasoning with Legal texts (MIREL 2016), Sophia Antipolis, France, 2016.
[27] C. Sansone, G. Sperlí, Legal information retrieval systems: State-of-the-art and open issues, Information Systems (2021) 101967.
[28] H.-T. Nguyen, A brief report on lawgpt 1.0: A virtual legal assistant based on gpt-3, arXiv preprint arXiv:2302.05729 (2023).
[29] A. Lecler, L. Duron, P. Soyer, Revolutionizing radiology with gpt-based models: Current applications, future possibilities and limitations of chatgpt, Diagnostic and Interventional Imaging (2023).
[30] OMG, Omg: Semantics of business vocabulary and rules (2008), 2008.
[31] V. Etikala, Z. Van Veldhoven, J. Vanthienen, Text2dec: Extracting decision dependencies from natural language text for automated dmn decision modelling, in: International Conference on BPM, Springer, 2020, pp. 367–379.
[32] L. Quishpi, J. Carmona, L. Padró, Extracting decision models from textual descriptions of processes, in: Conference on BPM, Springer, 2021, pp. 85–102.
[33] A. Goossens, J. De Smedt, J. Vanthienen, Extracting dmn models from text using deep learning techniques, Expert Systems with Applications 211 (2023).
[34] A. Jojic, Z. Wang, N. Jojic, Gpt is becoming a turing machine: Here are some ways to program it, arXiv preprint arXiv:2303.14310 (2023).
[35] J. Wei, X. Wang, D. Schuurmans, M. Bosma, E. Chi, Q. Le, D. Zhou, Chain of thought prompting elicits reasoning in large language models (2022).