

JARET: A Human Assistive A.I. Agent for Goal Review and Time Management

Andrew Schwabe^{1*} 

¹ Georgia Institute of Technology, Georgia, Amerika Serikat

*Corresponding author: aschwabe@gatech.edu

Abstrak

Banyak siswa tidak menetapkan tujuan atau merencanakan waktu mereka setiap minggu (karena kurangnya kemampuan, kesulitan yang dirasakan, dan alasan lainnya) yang mengakibatkan penundaan, stres, dan kinerja akademik yang lebih rendah. Makalah ini menyajikan metodologi desain dan pertimbangan untuk agen AI bantu manusia yang membantu siswa meninjau dan merencanakan tujuan belajar, mengurangi masalah abstrak besar menjadi serangkaian tugas peninjauan yang lebih sederhana. J.A.R.E.T. (Just A Recommender Engine for Time) menggunakan prinsip-prinsip kunci dari Self-Regulated Learning dan Cognitive Load Theory dalam sistem interaktif yang memandu siswa melalui tinjauan tujuan yang terfokus dan tugas-tugas perencanaan, kemudian menggunakan agen AI kepuasan kendala untuk menyusun jadwal kalender yang diusulkan yang dirancang untuk membantu mencapai tujuan siswa. Agen AI menggunakan batasan keras dan lunak dengan fungsi nilai yang dirancang dan mencari kecocokan terbaik yang mengikuti batasan sambil mencoba juga menyesuaikan preferensi siswa. Hasil menunjukkan bahwa desain mampu membangun solusi yang direkomendasikan dengan andal ketika kendala dan preferensi masuk akal dan tidak terlalu membatasi.

Kata kunci: Assistive AI, Manajemen Waktu, Self-Regulated Learning

Abstract

Many students do not set goals or plan their time weekly (due to lack of ability, perceived difficulty, and other reasons) resulting in procrastination, stress, and lower academic performance. This paper presents the design methodology and considerations for a human assistive AI agent that helps students review and plan for study goals, reducing a large abstract problem into a set of simpler review tasks. J.A.R.E.T. (Just A Recommender Engine for Time) uses key principles from Self-Regulated Learning and Cognitive Load Theory in an interactive system that guides students through focused goal review and planning tasks, then uses a constraint satisfaction AI agent to assemble a proposed calendar schedule designed to help achieve the student's goals. The AI agent uses hard and soft constraints with a value function designed and searches for a best fit that follows constraints while trying to also fit student preferences. Results show that the design is able to reliably build recommended solutions when constraints and preferences are reasonable and not overly restrictive.

Keywords: Assistive AI, Time Management, Self-Regulated Learning

History:

Received : August 23, 2021

Revised : August 28, 2021

Accepted : September 04, 2021

Published : October 25, 2021

Publisher: Undiksha Press

Licensed: This work is licensed under a Creative Commons Attribution 3.0 License



1. INTRODUCTION

Many students do not set goals or plan their time weekly (due to lack of ability, perceived difficulty, and other reasons) resulting in procrastination, stress, and lower academic performance (Basri et al., 2018; Steel, 2007). In this paper, you will learn how JARET was designed to address this problem. JARET changes the *complex* and *abstract* problem of building a personalized goal-oriented study schedule into a simplified set of activities, letting students more easily track their goals, increasing both confidence and satisfaction in their work, and thereby reducing avoidance behavior. In the results, you will see JARET's measured ability to quickly find quality approximate solutions for a personalized study schedule. This paper presents JARET (Just A Recommender Engine for Time) as a human assistive A.I. agent, designed to help students with self-regulating activities of goal planning and time management. The motivation for building this A.I. comes from years of personal observations of fellow students and coworkers who regularly

struggle with time management and goal planning. Inability (due to lacking the required skills), or avoidance of these planning activities have proven to have *negative* results on academic performance (Blunt & Pychyl, 2000).

A preliminary survey of 74 university students was conducted to help understand how JARET could best be applied to this problem and for its greatest potential benefits. The results show that many students (more than 55% in the study) had never received any formal education about goal setting and time management. More than 40% avoid planning because it is too hard, takes too much time, or they just don't know how. A majority of those surveyed said they would be willing to accept help with planning, and would use an app, and were comfortable or excited about receiving assistance from an A.I. agent. Originally, JARET was intended to solve one primary problem: the complex problem of assembling a schedule given requirements (constraints, and preferences). From the results of our preliminary survey and peer feedback, we discovered that there was just as much interest in reviewing goals and estimating required work effort as there was in the A.I. agent, so the scope of the application was expanded to include features to help with planning goals.

Research on *Self-Regulated Learning* (SRL), and particularly Zimmerman's *cyclical* model for SRL have identified time management and goal setting as critical *forethought* activities correlated to higher academic performance (DiBenedetto & Zimmerman, 2010; Panadero, 2017; Zimmerman & Moylan, 2009). The absence of these activities often results in procrastination, less satisfaction with academic quality, and overall lower academic performance (Broadbent, 2017; van Alten et al., 2020). The behavioral and emotional results may have additional consequences as well. Procrastination is known to result in stress and other health problems (Sirois & Pychyl, 2013). In a 2014 study with 96 undergraduate students, participants were offered assistance (as an intervention) to help plan goals and decompose a large task. Results show that in reaction, students allocated time more evenly to work toward their tasks, suggesting better self-regulatory behavior. It is noted, however, that these benefits were only seen on weeks when the intervention was provided, with other weeks reverting to less regulated behaviors and more erratic time allocation (Häfner et al., 2014).

Cognitive Load Theory, and particularly research on *working memory* limits in CLT helps explain how the human brain is able to assimilate new knowledge (Bentley & Sieben, 2019; Pollock et al., 2002). Several studies about designing CLT friendly teaching materials have shown that a *reduction of process complexity helps with more effective learning* (Van Merriënboer & Sweller, 2005). The reduction of complexity related to planning should reduce dependency on *working memory*, allowing students to focus more cognitive effort on domain specific work. Though many studies have been done about the benefits of Self-Regulated Learning activities, there were no studies found that attempted to address the problem directly with an assistive and interactive AI solution. Several studies discuss methods for optimizing the *curriculum timetabling* problem, which is similar in function to the schedule problem posed with JARET. Several studies have demonstrated the use of AI for finding approximate solutions (Lach & Lübbecke, 2012). JARET is different from other studies in that AI is applied to a previously human-driven process of time management intervention. No similar timetabling or scheduling problems were found to use a CSP based solution.

2. METHODS

JARET's A.I. agent was tested based on three key metrics: *error rate*, *quantity* of solutions, and *quality* of solutions. The *error rate* is defined as its ability to find a solution. As described above, the randomized nature of CSPs means that the search may not find a

solution within a given time limit due to large domain sets. The *quantity of solutions* is defined as how many solutions were found that satisfy the hard constraints (using the constraint function). The *quality of solutions* is defined as the mean value of the solutions found using the value function. To test JARET, a 5-day scheduling problem was created with 8 events between Monday and Friday and 7 tasks with event dependencies. JARET was asked to find a solution including meals for all days when, and if possible. JARET was tested for runtime limits between 1 and 30 seconds, running 50 epochs for each time limit. The mean number of solutions and mean quality (value function) was used for each time limit. This methodology was designed to test the viability and reliability of using this type of A.I. agent to solve this type of scheduling problem, and also to help inform a good choice of optimal time limit to use for end users.

3. RESULTS AND DISCUSSION

Results

The core application is built with Python 3, Flask framework, and the UI was built with Twitter Bootstrap and jQuery. The database selected was Postgres. The UI uses FullCalendar, Toastr and DataTables as visual components. *Goal Setting*, the primary view is a calendar with a single week visible. Students may add Events (aka *goals*) by simply clicking on an empty time slot (see Figure 1). Events are date/time entries when a student may be required to complete tasks. Lectures, for example, may require study for a test, or completion of homework. A “project due date” event may require final submission of a paper on or before the event date and time. *Goal Review and Task Workload Estimation*, the purpose of the goal review process is to let students focus on one goal at a time, decompose larger goals and estimate the work effort required for completion. A quiz in a class may only require a short review, while a draft of an academic paper may require hours of research, writing and review before submission. Each goal prompts for an estimate work effort (see Figure 2).

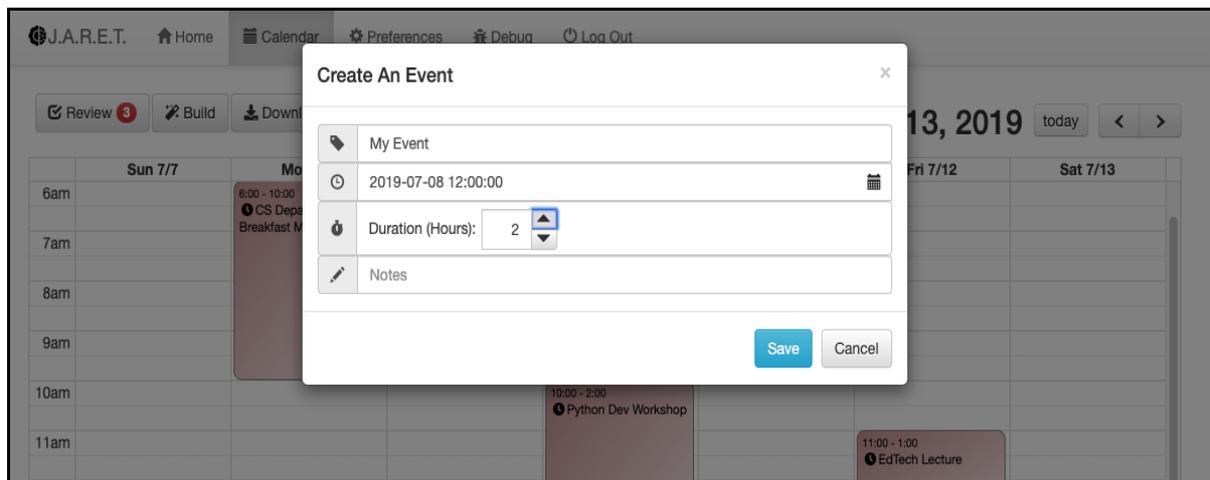


Figure 1. Clicking on an empty day/time slot allows easy creation of new Events

JARET presents the goal as a question to help the student focus on the requirements. By asking “what do you need to prepare,” the student can then review what they perceive is a correct amount of effort. In this version of JARET, there are two classes of estimate: a *review* activity, and an *estimated* number of hours for specific kind of work. *Review* activities are very specific tasks that attempt to allocate time very near the event during which the student would intent to do a short study session and review material. This is ideal for Events that

include tests, quizzes, or double checking the content of an assignment. *Estimate* activities have several subcategories of work, where a student may need time to work on an assignment, do research, general study or other activities. Unlike review activities, JARET attempts to schedule these activities as far in advance as possible of the Event to ensure enough time for completion and discourage procrastination. Students may choose any combination of a review activity and/or estimate activity, or may choose not to assign any activity. This would be consistent with any Events that do not require preparation, and could also include personal or social events. Upon completion of goal review, the student may then request for JARET to build the recommended calendar.

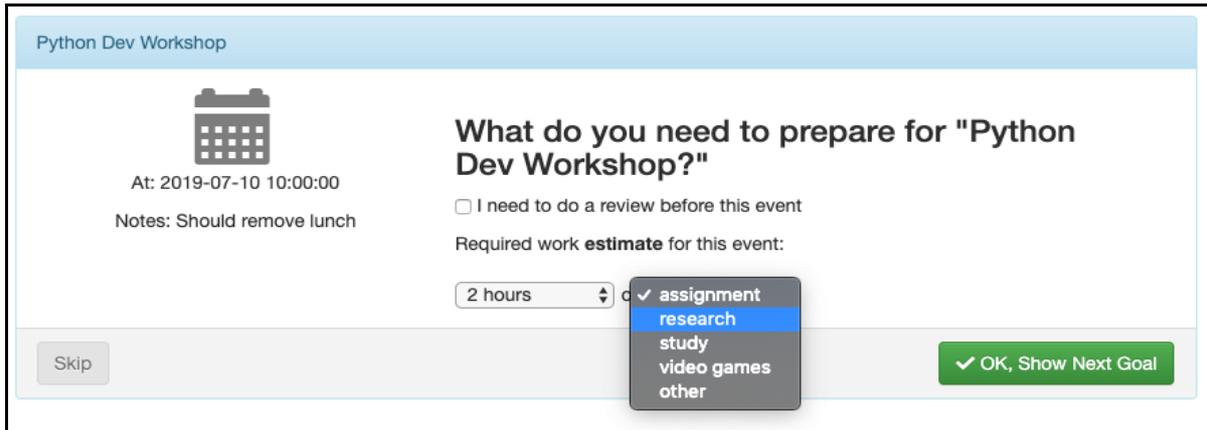


Figure 2. Reviewing an Event (goal)

Recommended Calendar Creation, when users click on the “Build” button, JARET analyzes the Events, Tasks and the user’s preferences, and builds an *approximate* calendar recommendation (see Figure 3). The resulting estimate solution attempts to fit meals and tasks in an optimal way around the Events on the calendar. It does so by using user preferences to assign times for tasks and meals in a way that maintains a sequence that supports completion of all the goals.

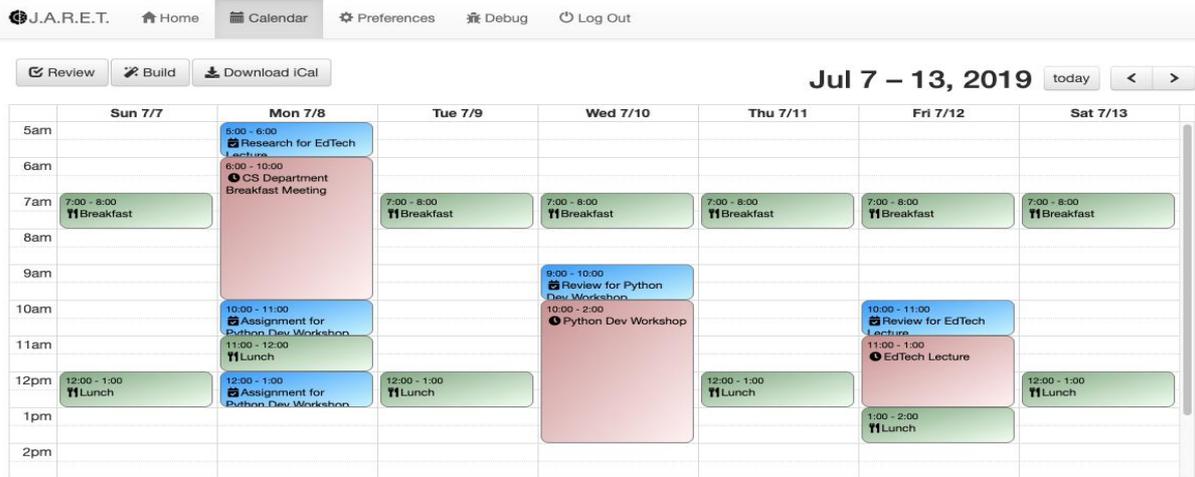


Figure 3. Approximate calendar recommendations

All task and meal events are recommendations, and are not final, by design. Depending on the user’s preferences, JARET may not always choose the best times for some activities. The UI permits the student to drag and drop tasks and meals in a way that they

prefer. In order to make sure that no constraints are broken, JARET will test every change made (using the constraint function via API) to ensure that tasks are still completed before their respective goals (Figure 4). As an *estimate* solution, JARET allows the student to adjust tasks and meals to their preference. As an *assistive* agent, JARET encourages students to customize the results to their liking, to increase the chances that the student will accept the solution rather than “being told” what to do (whether from a human or an A.I.).

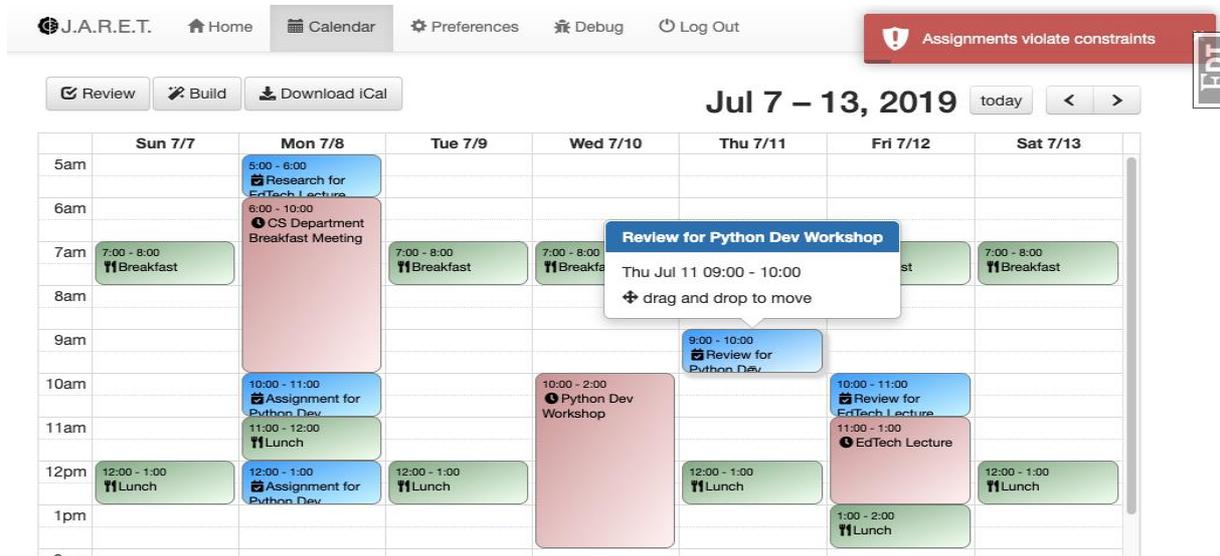


Figure 4. Changing a task that breaks constraints will warn the student

iCalendar Export, to further encourage ongoing use of the schedule, the customized calendar may be exported to an iCalendar file by clicking the “Download iCal” button. The events, tasks and meal schedule can then be imported into their own devices. *User Preferences*, a key component of personalizing the recommended schedule is for individual preferences. A view in JARET lets users adjust common settings to their preference, including acceptable times to assign tasks, preferred mealtimes, and preferred study hours (see Figure 5).

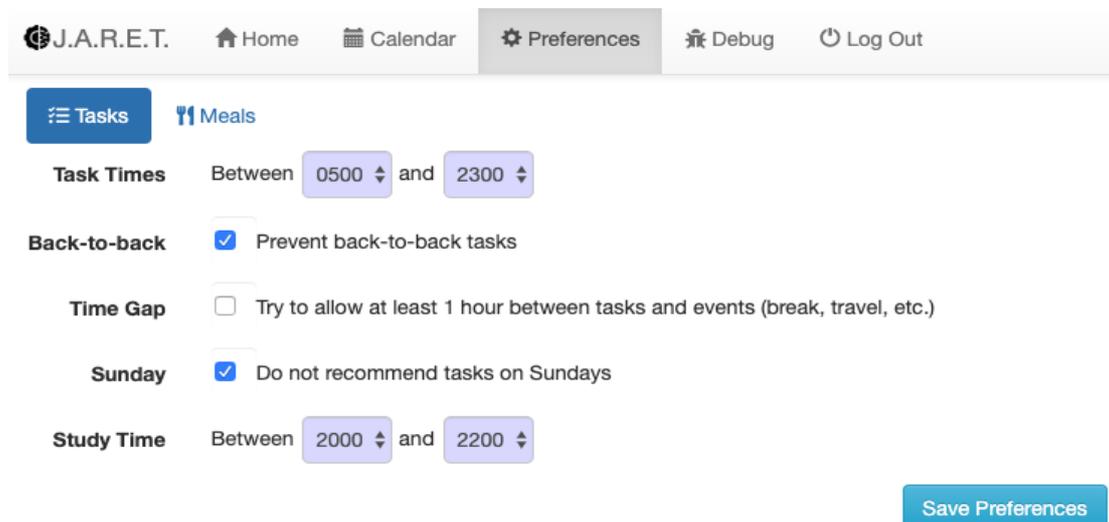


Figure 5. User Preferences Panel

Discussion

Rephrasing the Scheduling Problem as a CSP, once a structured collection of Events, Tasks and Meals is available, the task of assembling them into a schedule is presented. The problem itself can be compared to the well-known curriculum timetabling problem which is known to be very difficult to find an optimal solution. In JARET, we simplify this problem by only working with a single week's worth of data and rephrase the problem as a Constraint Satisfaction Problem (CSP) (Lach & Lübbecke, 2012; Russell & Norvig, 2010). In Russell and Norvig's CSP model, a fixed number of variables are established, and then a search algorithm finds values to assign for each variable such that all constraints are satisfied. In JARET, we use a two-stage CSP model, assigning time slots to calendar events using hard constraints to find viable solutions, and soft constraints (preferences) to measure the quality of each found solution. Mapping the schedule problem to a CSP requires encoding events, tasks, meals as variables, and encoding available day/time combinations as numeric *time slots*. By encoding time slots as integers, it becomes an efficient operation to compare assigned times of different events to determine its ordinarily.

Hard and Soft Constraints, a *constraint function* ensures that all basic rules are followed. No two scheduled items may occupy the same time slot. We also require all of the activities we planned to occur before their dependent events. Any assignment of a time slot to an event, meal or activity that violates these rules is not *consistent* and is rejected. For each candidate solution, a *value function* is used to measure its quality as defined by the user's preferences. A student may prefer, for example, to study after 8pm, so assigning a study activity to a time slot for 8pm would have a higher *value* than assigning that activity to 10am. JARET keeps the result with the highest value, returning that as the recommended solution.

Challenges with Using CSP for Scheduling

In general, CSP has been a good solution for this type of problem, especially considering the limited domain scope of one week. There were, however, still several challenges to using CSP with JARET's particular design. *Pruning Variables*, a classic CSP tries to find *complete* solutions (all variables assigned). In JARET, sometimes an Event may occupy all available time slots available for an activity, such as a meal. In this situation, the lesser priority activity may be removed, involving removal of a CSP variable (pruning) or else no solution can be found. JARET implements a user preference to allow removal of meals to avoid this. *Optimized Domain Selection*, many implementations of CSP use a random approach to value selection from a variable's domain. In JARET, picking random available *time slots* wastes processing time, and in fact, version 1 of the application was so poor at selecting values that the runtime was too long. To address this, the domain of time slot values was optimized based on the type of event. Meals first try to fill preferred times before trying less optimal times. Review activities first try time slots closest to their dependent events before trying less optimal times, etc. Preferences are dynamically applied to choose an optimized domain for each event.

Optimized Variable Selection, similar to domain selection, variable selection is often handled randomly in CSPs. After testing with version 1 of JARET, it was determined to be significantly faster to assign time slots to all *Tasks* before trying to assign time slots to *Meals*. *Runtime Limit*, JARET uses a backtracking search that chooses a task **at random** to assign, even though the domain of values is ordered. This presents a performance challenge, as sometimes early choices are problematic, blocking many other required assignments. In this scenario, the CSP will eventually find solutions, but potentially not before a long runtime since all available time slots for a week ends up with large domains of values. In version 1, JARET did not optimize the variable selection process, which yielded a much higher error

rate. In order to address this problem, JARET uses a runtime limit, and defines an *epoch* as one such run of the CSP search within that time limit. If the search yields no results, then JARET will run a second epoch with the same time limit. This limits runtime in the scenario where a poor early assignment causes no results to be found within the time limit. The results of testing showed that the version 2 agent is able to find solutions > 30% of the time even with small time limits, such as 1-2.5 seconds, and improves after 7.5 seconds. Version 1 had much higher error with short times (see Figure 6). The results show that between 5 and 9 seconds of runtime, a satisfactory quantity and quality of solutions may be found. These results also show that a longer runtime does not necessarily result in a higher quality solution (see Figure 7). These results show that the agent finds solutions quickly with similar quality.

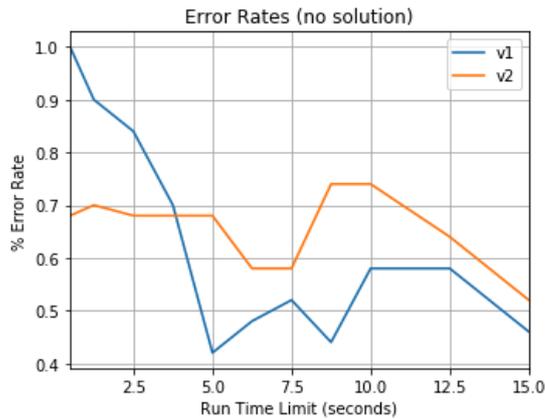


Figure 6. Error Rates by Runtime Limit (Lower is better)

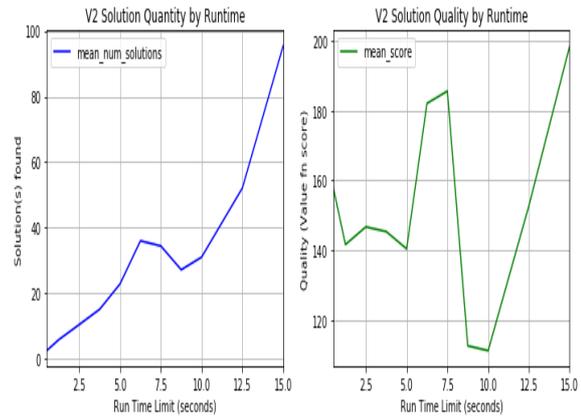


Figure 7. Solution Quantity and Quality by Runtime Limit (Higher is better)

JARET has several known limitations in its current version, namely: preferences only optimize *review* and *study* activities; Reviews of events only permit up to 2 types of activity selections; JARET only uses 1-hour time slots for activities; JARET is not aware of multi-week goals; more work is required to avoid “no solution” CSP situations. The full effectiveness of JARET to affect self-regulation behaviors (and reducing avoidance and procrastination) requires further study. There are several areas for the development of JARET that are intended: *Reflection* SRL activities would allow students to review their previous week’s work for quality, satisfaction and timeliness; More granular scheduling so that events use 15 or 30 minutes’ time slots; More granular preferences and value function tuning such that: students can identify days and times when they are most efficient; different types of work may be scheduled at different times; Extend the AI to learn from users’ drag and drop customization of final calendar solutions and adapt recommendation methods; Automatic decomposition of large tasks based on best practices and user preferences; Additional work to prevent review activities from causing “no solution” CSP situations, such as when a student needs more study hours than are available on the schedule.

4. CONCLUSION

Time management and goal setting activities are known to be positively correlated with academic performance, and the lack of skills or avoidance of these activities yields negative effects. Students often avoid these critical SRL activities for reasons including inability, perceived difficulty or believe that planning is a low value activity. JARET reduces the large, abstract problem of “figuring out a schedule to accomplish many tasks and goals” into a simple guided process of activities. The activities help students add known events to

their calendar, and easily estimate work effort required, then let the A.I. recommend a work/study schedule. The results show that, given events, tasks and preferences, JARET is capable of finding approximate solutions to this problem effectively in a short period of time. JARET can be further improved with better, more granular control over time slots, expanded preferences and optimized constraint and value functions.

5. REFERENCES

- Basri, W., Alandejani, J., & Almadani, F. (2018). ICT adoption impact on students' academic performance: Evidence from Saudi universities. *Education Research International*, 2018. <https://doi.org/10.1155/2018/1240197>.
- Bentley, B., & Sieben, R. (2019). Cognitive load theory: An adjunct to constructivist learning theory not an alternative. *Australian Educational Leader*, 41(1), 48–51. <https://doi.org/10.3316/INFORMIT.513437227836287>.
- Blunt, A. K., & Pychyl, T. A. (2000). Task aversiveness and procrastination: a multi-dimensional approach to task aversiveness across stages of personal projects. *Personality and Individual Differences*, 28(1), 153–167. <http://scinapse.io/papers/2066142246>.
- Broadbent, J. (2017). Comparing online and blended learner's self-regulated learning strategies and academic performance. *Internet and Higher Education*, 33, 24–32. <https://doi.org/10.1016/j.iheduc.2017.01.004>.
- DiBenedetto, M. K., & Zimmerman, B. J. (2010). Differences in self-regulatory processes among students studying science: A microanalytic investigation. *International Journal of Educational & Psychological Assessment*, 5(1). https://www.academia.edu/download/8333411/v5_tijepa.pdf#page=5.
- Häfner, A., Oberst, V., & Stock, A. (2014). Avoiding procrastination through time management: An experimental intervention study. *Educational Studies*, 40(3), 352–360. <https://doi.org/10.1080/03055698.2014.899487>.
- Lach, G., & Lübbecke, M. E. (2012). Curriculum based course timetabling: new solutions to Udine benchmark instances. *Annals of Operations Research*, 194(1), 255–272. <https://doi.org/10.1007/s10479-010-0700-7>
- Panadero, E. (2017). A review of self-regulated learning: Six models and four directions for research. *Frontiers in Psychology*, 8, 422. <https://doi.org/10.3389/fpsyg.2017.00422>.
- Pollock, E., Chandler, P., & Sweller, J. (2002). Assimilating complex information. *Learning and Instruction*, 12(1), 61–86. [https://doi.org/10.1016/S0959-4752\(01\)00016-0](https://doi.org/10.1016/S0959-4752(01)00016-0).
- Russell, S. J., & Norvig, P. (2010). *Artificial intelligence: A modern approach* (3rd ed.). Prentice Hall.
- Sirois, F., & Pychyl, T. (2013). Procrastination and the priority of short-term mood regulation: Consequences for future self. *Social and Personality Psychology Compass*, 7(2), 115–127. <https://doi.org/10.1111/spc3.12011>.
- Steel. (2007). The Nature Of Procrastination: A Meta- analutic an Theoretical Review of Quintessential Self- Regulatory. *Psychological Bulletin*, 133, 65–94.
- van Alten, D. C. D., Phielix, C., Janssen, J., & Kester, L. (2020). Self-regulated learning support in flipped learning videos enhances learning outcomes. *Computers and Education*, 158(July), 104000. <https://doi.org/10.1016/j.compedu.2020.104000>.
- Van Merriënboer, J. J., & Sweller, J. (2005). Cognitive load theory and complex learning: Recent developments and future directions. *Educational Psychology Review*, 17(2), 147–177. <https://doi.org/10.1007/s10648-005-3951-0>.
- Zimmerman, B. J., & Moylan, A. R. (2009). *Self-regulation: Where metacognition and motivation intersect*. In *Handbook of metacognition in education*. Routledge.