NodeToJoy: Understanding Probabilistic State through Interactive Visualisation

Charlotte Cox
Computer Laboratory
University of Cambridge
cc2207@cam.ac.uk

Alan F. Blackwell
Computer Laboratory
University of Cambridge
afb21@cam.ac.uk

Abstract
As probabilistic machine learning systems are deployed increasingly widely, computational literacy will require better understanding of the fundamental distinctions between conventional deterministic models of computation, and probabilistic behaviours. This may be especially important in cases that seem superficially similar, such as text string generation via sequence prediction in LLMs. We introduce NodeToJoy[1], a visualisation that juxtaposes a Finite State Machine visualisation with a Markov Chain visualisation, both adapted to the exploration of large and complex datasets by interactive hierarchical semantic zooming (including a novel extension of Harel's Statecharts to Markov Chains). Our paper reports classroom studies demonstrating that lessons including interaction with NodeToJoy significantly advance student understanding of these fundamental principles.

1. Introduction
The way computers appear to 'understand' what users mean remains a mystery to many young students. Traditionally, a computer's apparent 'understanding' of language could be explained through lexical analysis and parsing - deterministic, rule-governed systems, where Finite State Machines (FSMs) play a significant role. Large language models (LLMs) might appear superficially similar effect to non-technical users, in the sense that they interpret a sequence of input text. Yet, the underlying probabilistic model is formally distinct from the rule-governed manipulation of state that has historically been central to the mathematics of computation (Vidal, Thollard, de la Higuera, Casacuberta, & Carrasco, 2005), and would be better described in mathematical terms as a Markov Chain (MC). Emphasising the difference between rule-based and probabilistic-inference machines can shape a more informed engagement with such tools, and potentially empower students to think critically about interactions with LLMs.

Despite the computational distinction, both FSM and MC models are state-based formalisms. Both can be represented diagrammatically in node-link form, where states correspond to notes, and links represent either transition likelihood, or allowable transitions. FSMs are generally introduced in this way, where the metaphor of state as a visual location in (a "state space") aids spatial intuition. MCs are often taught through algebraic notation or matrix representation of transition probabilities, but graphical rendering significantly improves user response times (Keller, Eckert, & Clarkson, 2006), and visualisation has been shown to enhance learning (Kim & Jin, 2022; Presmeg, 2006; Gates, 2018; Elliott, Hudson, & O'Reilly, 2000; Wang, Han, & Chou, 2018).

We start from this diagrammatic motivation to explore an urgent educational need. Although UK students studying Computer Science at high-school (A Level, corresponding to 11th and 12th grades in the USA) are introduced to Finite State Machines (AQA, n.d.), no existing course touches on probabilistic models of computation. Following recent developments elsewhere, reforms are now planned for the UK national curriculum with a focus on probabilistic literacy and real-world applications (Royal Statistical Society, 2024). As a teaching strategy, the Royal Statistical Society already recommends greater use of visualisation software for Data Science and AI (Royal Statistical Society, 2024),(Mathematics in Education and Industry (MEI), n.d.).

[1] A reviewer has asked us to explain our choice of name. For decades, students of theoretical computer science have looked upon Finite State Machines and Markov Chains as cold, unfeeling arrays of circles and arrows. Just as Beethoven shattered classical boundaries by infusing his final symphony with vocal passion, NodeToJoy transforms static state diagrams into a living chorus.

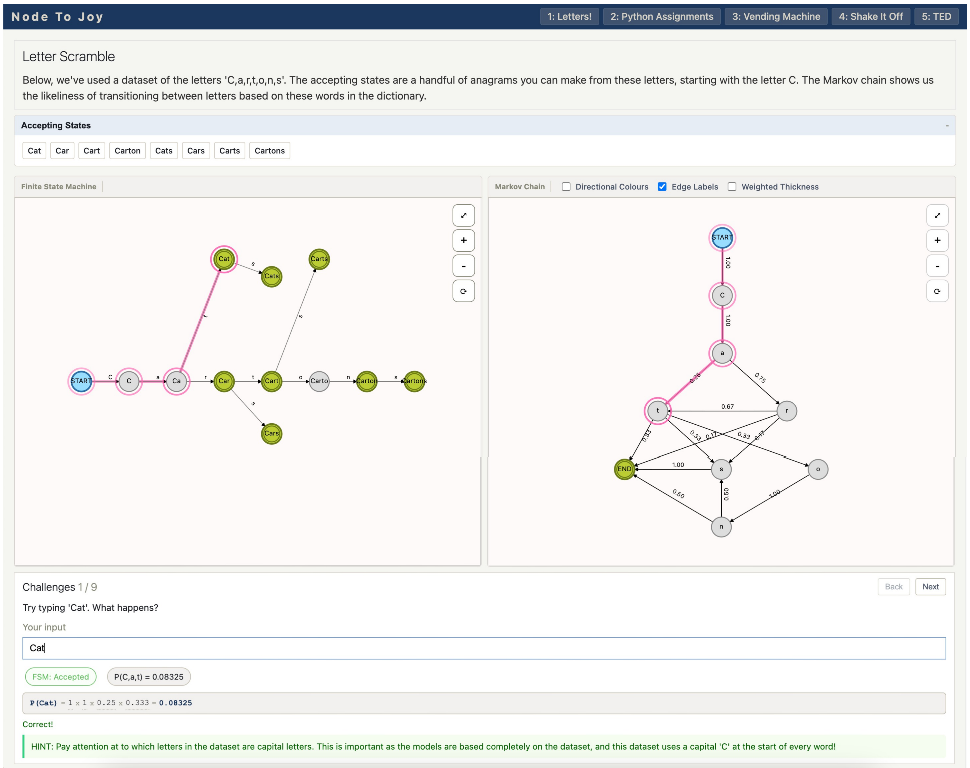


Figure 1 – The first page of NODETOJOY, *featuring a simple FSM, left, and MC, right.*

A foundational question for AI literacy in this turn toward probabilistic system behaviour is for students to gain a critical understanding of the difference between computational concepts of *validity* and of *likelihood*. Our system NODETOJOY seen in figure 1, allows students to explore and contrast these concepts through interactive parallel visualisations of Finite State Machines and Markov Chains.

1.1. Previous FSM and MC visualisations

Literature on educational FSM and MC visualisation addresses the two models in isolation: no existing platform explores their juxtaposition for novice learners. The most popular educational tool for visualising FSMs is JFLAP, a visual simulation platform (Rodger, 2006). The expressive power of JFLAP is valuable for undergraduate students, but less appropriate to novices in their first-time encounter with automata. Other studies have explored methods around gamification and narrative to make FSMs more approachable with interactivity and visualisation. An identified lack of K-12 resources led to Castro's game-building tool using FSM blocks (Castro, 2022), whereas introducing FSMs through storytelling improved approachability and memory retention for graduate students (Iudean & Vescan, 2025). Dengel combined both strategies, suggesting a Virtual Reality scenario of acting within a FSM, emphasising the immersive 'unplugged' benefits to engagement (Dengel, 2018).

Markov chain education is often centred on transition matrices and probabilistic calculation, rather than visual tools. Pfannkuch and Budgett addressed this gap through an interactive tool linking transition matrices with their equivalent diagram in real time, finding that visualisation aided students' conceptual understanding (Pfannkuch & Budgett, 2016). In a similar domain of Markov decision processes, Mutua found that graphical visualisations employing narrative were more effective than textbook learning for undergraduates in Kenya (Mutua & Blackwell, 2023). This project aims to work at the intersection of these interactive and playful strategies.

1.2. Contributions

We have created an interactive educational platform to juxtapose two foundational but contrasting models of computation: finite state machines and Markov chains. The goal is to help students who may be unfamiliar with these models to develop foundational understanding of both. Importantly, this en-

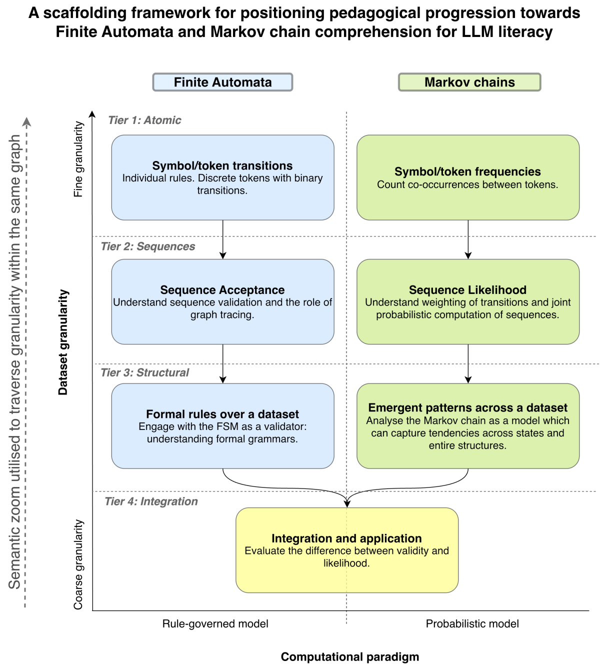


Figure 2 – Framework for describing the range of datasets that may be used in NODETOJOY, *for reference in justifying design choices. Dataset scale across these positions can range from toy illustrative to pedagogically reinforcing to statistical, large corpora.*

ables dialogue around the difference between validity and likelihood, questions that are critical to the stochastic behaviour of LLMs. NODETOJOY makes a number of key contributions towards extending computational model visualisations in education. It creates robust node-link graph visualisations for both FSMs and MCs across a wide range of scale and complexity, with integrated interactive techniques such as semantic zooming, node manipulation, and sequence tracing to promote active engagement with the models.

Most notably, this project draws on David Harel's statecharts to extend both the finite state machine and Markov chain visualisations through hierarchical structure (Harel, 1987). Whilst statecharts have become an important component of the UML framework, research on *Markov statecharts* - an extension identified, but not pursued, by Harel in the 1987 paper "On Statecharts" - has been sporadic across disciplines and underexplored. The use of hierarchy is invaluable for pedagogical scaffolding, abstraction, and greater expressiveness (Melero, Hernández-Leo, & Blat, 2012). It facilitates modelling real-world applications through information hiding, reducing educators' reliance on contrived datasets for visual simplicity. We describe a novel platform and visualisation framework, which we believe is the first to combine hierarchical FSMs and Markov chains, and also the first to utilise semantic zoom to traverse levels of depth in statecharts. The project's effectiveness was evaluated through two in-school user trials with the target audience of secondary school students in year 12 of the English school system (also called Lower Sixth, equivalent to USA 11th Grade or High School Junior). These provide evidence that NODETOJOY is highly effective in increasing students' understanding of finite state machines, Markov chains, sequence processing, and the probabilistic nature of LLMs.

2. A framework for computational state literacy

Our central claim is that the conceptual distance between rule-governed and probabilistic models of computation can be bridged through traversable interaction design. To provide a principled basis for the design decisions made throughout implementation, this section provides a framework to characterise the conceptual space this platform navigates. The main axes of concern are granularity, scale, and computational paradigm.

Granularity describes the level of abstraction at which a student engages with the model. Following Bloom's taxonomy (Forehand, 2007), learners' progression of comprehension skills map onto three tiers, illustrated by figure 2. **Atomic (Tier 1)** granularity focuses on individual symbol transitions, before composing these to parse **sequences (Tier 2)**. As students understand the models' functionality, states can be abstracted into **structures (Tier 3)**, reflecting abstraction. This dissertation implemented support for all tiers: clear node layout to ease graph tracing in atomic granularity, path-highlighted sequence computation for sequential granularity, and statecharts to support structural granularity. Semantic zoom is the mechanism by which students can navigate **between tiers of granularity within a singular graph visualisation**, supporting the integration of knowledge for Tier 4.

Independent of granularity, corpus scale was also considered throughout development. As students begin learning, toy datasets are often used by educators for illustration, before progressing to pedagogically-scoped, scale-bounded, reinforcement datasets, eventually to large, statistical corpora of real-world data. Clear layout for all dataset scales was considered throughout development: layout-wrapping and viewport culling ensure smooth rendering performance for large corpora. To verify that datasets of the expected JSON format could be integrated in the project without bespoke changes, additional speech text corpora containing 500+ nodes were parsed, tested, and successfully laid out clearly without dataset-specific changes.

We selected teaching datasets at different positions within this framework, providing a scaffolded progression from fine-granularity detail to coarse-granularity exploration(Melero et al., 2012). The lesson plan presentation of NODETOJOY consists of four themed pages in which an FSM and MC for each of these datasets are presented in side-by-side panels. A fifth themed page presents just the Markov chain for a very large corpus, with fifty TED Talks parsed into fine-granularity bigrams that allow students to engage in higher-order, exploratory learning.

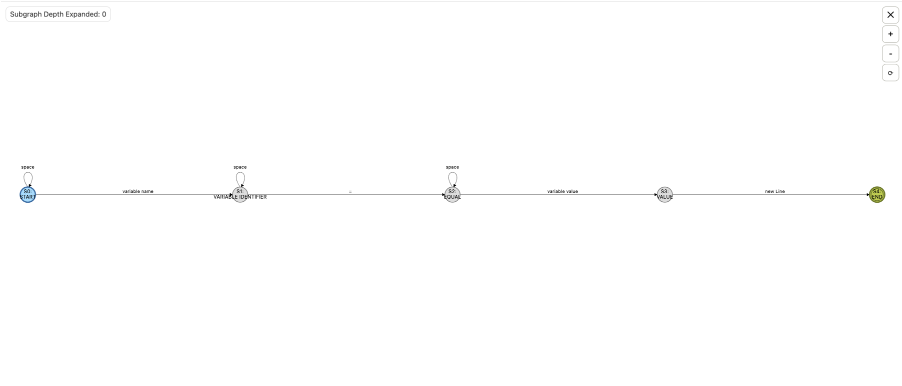
3. Visual interaction strategies

Our design employs a number of graph visualisation and interaction strategies to ease the cognitive load of novice users exploring potentially very large graphs (Sweller, 2016). We make extensive use of semantic zooming, adding detail as the user zooms in, to progressively reveal levels of statechart hierarchy (Cockburn, Karlson, & Bederson, 2009). A focus+context strategy is employed to ensure that revealed detail can be understood within a graph's wider context (Furnas & Bederson, 1995). In the Markov chain viewer, this is enacted through viewport-sensitive edge-culling; only rendering the edges between visible nodes when a user zooms in, which preserves surrounding context without visual noise. Baudisch and Rosenholtz's 'halos' are also used to visualise off-screen nodes: rectangles are nested around subgraphs to preserve the context of expanded objects (Baudisch & Rosenholtz, 2003).

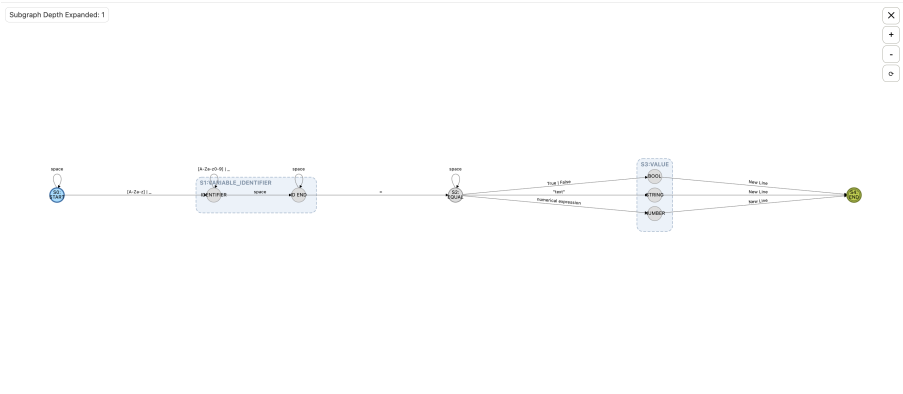
Combining hierarchical graph nesting with semantic zoom offers a novel interactive traversal mechanism between fine and coarse grained datasets. When a user zooms past a semantic zoom threshold (a depth of granularity in the dataset) a parent's node and edges disappear and are replaced by its child subgraph in the correct position. On collapse, the reverse must occur without corrupting node positions that the user may have manually adjusted. When a parent node is expanded, its incident edges are left dangling with no semantic meaning. To maintain the sequential flow of edges in a FSM, these transitions must be rerouted to preserve the visual continuity between the parent's subgraph and the surrounding graph.

3.1. Markov chain hierarchical subgraphs

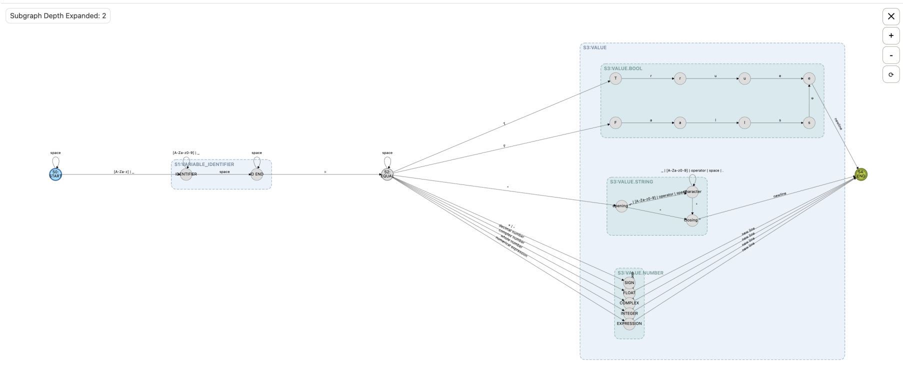
A coarsely-grained MC dataset presupposes probabilistic transitions between structural states represented as nodes. Within each structural state may reside a finer-grained subgraph, meaning that a coarsely-grained state can contain its own probability distribution over a finer-grained state space. Hierarchical MCs thus reflect the goal of granularity traversal through semantic zooming while preserving structural probability. Each subgraph is rendered within its halo, with coarse-level, inter-structure transitions drawn as arrows against each bounding rectangle using parametric ray-intersection. This preserves the user's understanding that each coarse state's subgraph is locally isolated but globally situated within a probabilistic state space. Inter-structure transition arrows are rendered as dashed Bezier curves to



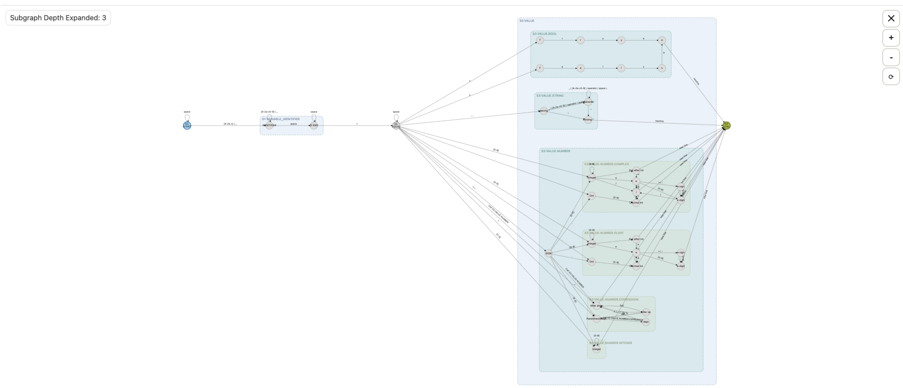
(a) Python hierarchical FSM before subgraph expansion



(b) Hierarchical FSM expanded to depth 1



(c) Hierarchical FSM expanded to depth 2



(d) Hierarchical FSM expanded to depth 3: letter-level parser

Figure 3 – Expansion progression of a hierarchical FSM across 3 nested depths.

Dataset	Granularity tier	Scale	Models Implemented	Pedagogical Reasoning
Subset of anagrams of "Cartons"	Fine-grained: character-level atoms	Illustrative toy	Flat FSM (Partial DFA) + flat MC	A simple letter level parser to familiarise students with both models. Connects FSM structure to compiler tokenisation.
Python assignments	FSM: Coarse-grained abstract structures → fine-grained character-level atoms MC: fine-grained character-level atoms	Pedagogical reinforcement	Hierarchical FSM (NFA with 3 levels of subgraph nesting) + flat MC	Hierarchical FSM across 4 depth levels walking students from complete abstraction to regex-like letter parsing. MC dataset of 1000 real world assignments.
30p Vending Machine	Structural whole: state represents cumulative currency	Illustrative toy	2x Flat FSM (DFA) + Flat MC	Demonstrates state explosion by adding a new valid coin to dataset 2. Illustrates FSM/MC uses beyond language parsing to construct an informed mental model.
Shake It Off lyrics, by Taylor Swift	FSM: Structural whole (suffix automaton) MC: Coarse-grained song structure → fine-grained word-level tokens	Exploratory analysis of statistically-sized dataset	Flat FSM (Partial DFA) + Hierarchical MC (1 level of hierarchy)	Demonstrates unsuitability of FSM for some real-world problems, alongside hierarchical MC traversal of an engaging context.
TED Talks (×50)	Structural whole: word-level tokens	Exploratory analysis of statistically sized dataset	50x Flat MCs	Student-directed exploration at scale (500+ nodes). Supports higher-order thinking about emergent patterns in natural language.

Table 1 – Datasets used in NODETOJOY, positioned within the framework for comparative computational literacy.

differentiate them from solid intra-state edges, with obstacle avoidance to prevent intersections with subgraphs. Edge thickness is linked to the probability of transition, providing a visual dual-coding of probability seen in figure 4.

Given that a subgraph may nest arbitrarily many times, there must be a graphical notion of sub-subgraphs to visually support user's understanding of hierarchy. This is addressed by encapsulating each subgraph in a rectangular "halo" container (Baudisch & Rosenholtz, 2003), of which sub-subgraphs reside in their parent subgraph's halos - visible in figures 3c and 3d.

3.2. Secondary notations

Interactive engagement is core to our pedagogical design aim; active learning through manipulation is encouraged, as guided by constructivism (Ben-Ari, 1998). Accordingly, interaction must be low-stakes, reversible, and consistent with the graph's underlying structure. For example, the default graph layout may not match a student's mental model (Ben-Ari, 1998), so we support dragging for users to organise the graph to reflect their own understanding. Semantic zooming is core to hierarchy traversal, but deep zoom levels risk contextual disorientation (Touya, Gruget, & Muehlenhaus, 2023). Geometric (rather than semantic) zoom buttons let users pan out to view the current subgraph within the graph's global context.

3.3. Focus+context filtering

Large Markov chain graphs present a density problem specific to natural language datasets, which are key to this project's LLM literacy aim. Following Zipf's law (Piantadosi, 2014), some words will occur with disproportionately high frequency in corpora, producing densely connected nodes whose edges

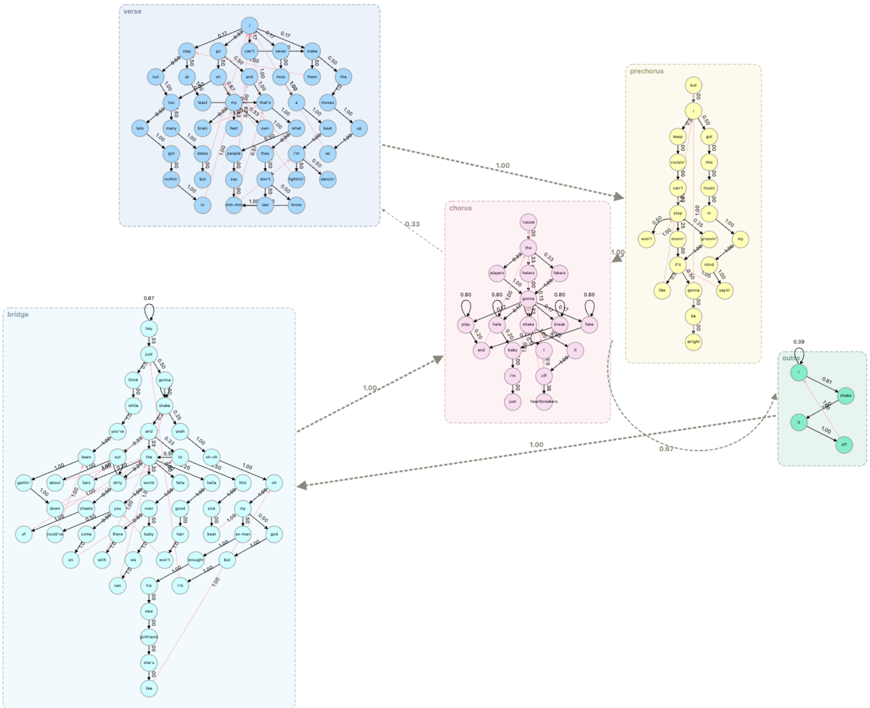


Figure 4 – Hierarchical Markov statechart with subgraphs expanded

become visually occluded. A focus+context approach (Furnas & Bederson, 1995) is taken from 3 filtering angles in the Markov viewer to address this: viewport culling to suppress edges when not zoomed in, per-node focus fades all non-incident edges and nodes to opacity 0.15, and dataset-item focus to highlight the subgraph for a specific piece of input data as in figure 5.

3.4. Sequence inspection

For guided lesson engagement, NODETOJOY also provides a panel of interactive challenges. This allows students to test hypotheses about the graph: inputting sequences and receiving immediate feedback on probability/validity computation. As a student inputs a sequence, the platform highlights the path taken by their sequence on flat graphs. This is important for drawing cross-form connections between tracing the graph and linking to their real-world purposes by seeing live validity/probability computation.

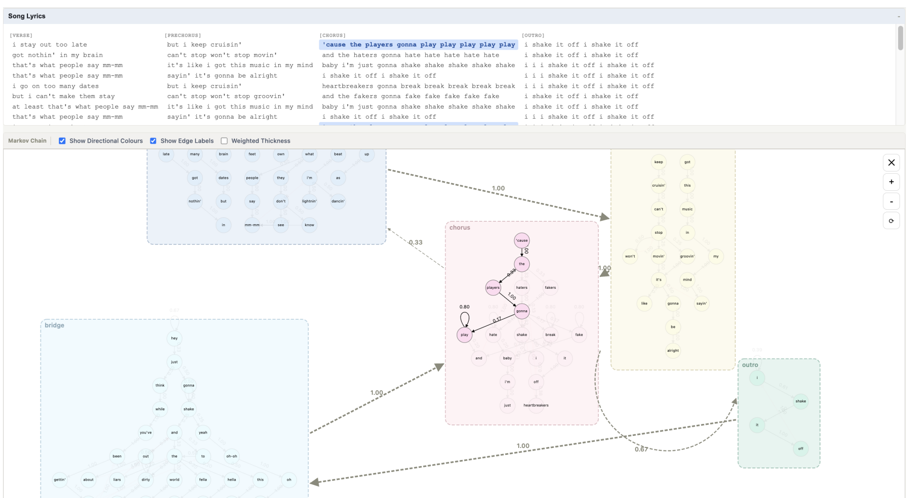


Figure 5 – Dataset item selected and highlighted in Markov chain subgraph

4. Classroom evaluation

We have evaluated NODETOJOY in two in-school trials. This study was approved by the Ethics Committee of the University of Cambridge Department of Computer Science and Technology. The lessons were delivered by the first author who has appropriate DBS-certification. Students were assigned an animal name as a pseudo-anonymous identifier to connects Likert-scale ratings, open-ended question answers and logged platform interactions.

4.1. Method

Each of the two lessons followed the same 1 hour structure:

1. Students completed an open response question to gauge their pre-lesson understanding: "The use of Large Language Models in platforms like ChatGPT have made computers increasingly able to write convincingly human-like text. How is the way a computer reads code different to the way it may read an essay or poetry?"

2. The researcher/teacher led a short presentation of the intuition behind FSMs/MCs.

3. Students spent the majority of the lesson (over 40 minutes) engaging with NODETOJOY, with interaction events logged anonymously.

4. The open question from the start of the lesson was repeated, for post-lesson evaluation of gained understanding.

5. The lesson closed with a Likert questionnaire. Students assessed their level of understanding before and after using the platform across key learning objectives.

Responses to the pre- and post-lesson question, were classified into four categories: (1) incorrect or no answer, (2) some evidence of computational thinking, (3) some understanding of FSMs or Markov chains, (4) comparative discussion of FSMs against Markov chains. Within-subjects comparison used a Wilcoxon signed-rank test. Students also self-assessed their confidence of each concept on a scale of 1 (no understanding) to 7 (confident understanding). To reduce bias when measuring program effectiveness, Nimon, Zigarmi & Allen's work informed the questionnaire structure (Nimon, Zigarmi, & Allen, 2011). Interaction logs recorded 18 event types, covering graph manipulation, dataset navigation, sequence input, and question progression.

4.2. Results summary

In both lessons, results found a consistent, significant gain in student understanding. A pooled repeated-measures multivariate test was computed with time (pre-lesson versus post-lesson) as the within-subject factor. Across all 24 students and 6 concepts, the test yielded $T^2 = 119.50$, $F(6,18) = 15.59$, $p = 0.000003$, providing strong evidence of effect on learning. 22 students increased their Likert score assessment across every concept and both cohorts converged to a post-lesson median confidence of 6/7 in every concept (except AI explanation at All Saints' with a median of 5.5). This consistency across both groups is particularly encouraging, suggesting that the platform successfully scaffolds these complex topics to support learning from a range of prior knowledge baselines.

4.3. Year 12 non-CS class

The first lesson took place at Hills Road Sixth Form College, Cambridge, as part of an extra-curricular Data Science class with eighteen year 12 students (one left early). Most students in the class studied A Level mathematics; two studied A Level computer science, so had learned FSMs as part of the AQA curriculum (AQA, n.d.). A small group expressed an extra-curricular interest in Markov chains from YouTube videos, although were unfamiliar with the model's formalisms. Most students accessed NODETOJOY on a laptop or iPad, with some opting to use desktop computers and one using their mobile phone. The platform worked well across all devices, with the graphs using dynamic layout resizing and pinch-to-zoom functionality for devices without a trackpad or mouse.

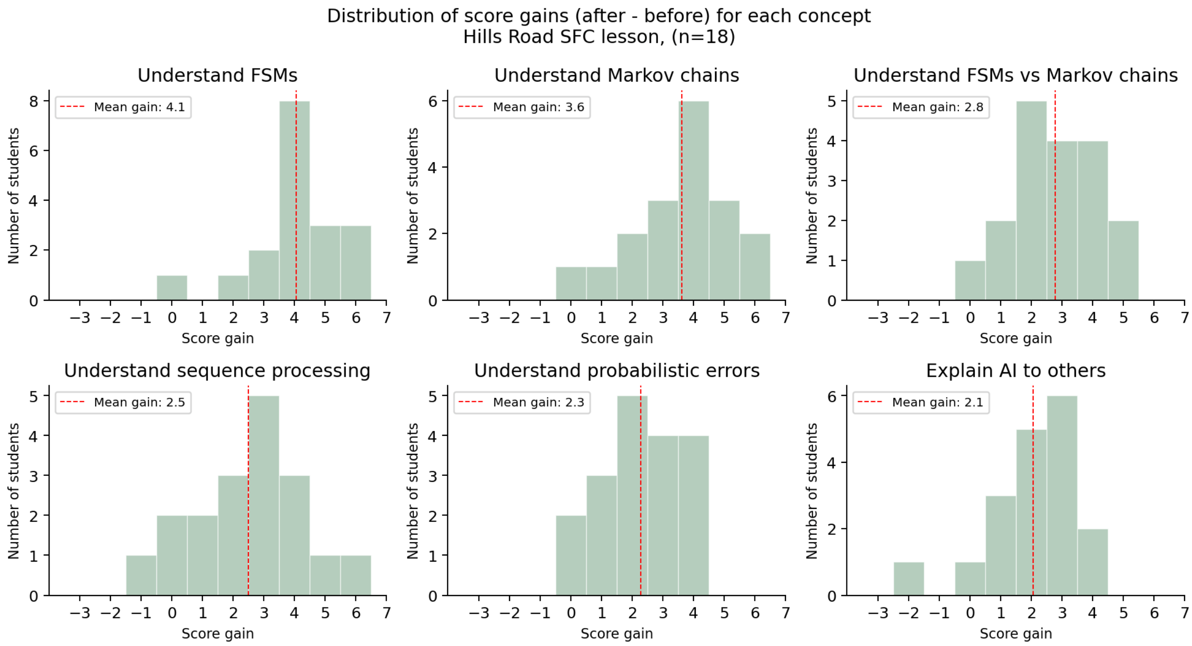


Figure 6 – Distribution of score gains per Likert scale question at Hills Road

NODETOJOY proved accessible against the wide range of knowledge backgrounds. Students new to the concepts often engaged by dragging and investigating the graphs, alongside answering structured exploration questions. More confident students often progressed to the later datasets; they conversed with peers about patterns they noticed across different datasets, as intended with the exploratory, statistically-sized corpora. Many students responded positively about their interest in the use of the models, with their teacher noting the high-level of engagement amongst students.

Students showed highly significant gains in comprehension across all six categories, with an average gain of 2.88. A repeated-measures multivariate test for the 18 students across 6 concepts at Hills Road showed a significant overall effect, with $T^2 = 156.28$, $F(6,12) = 18.39$, $p = 0.000021$.

The largest mean gains were in FSM and MC understanding, consistent with most students being unfamiliar with these concepts before the lesson. Students' pre-lesson understanding of more applied topics (probabilistic errors and AI) was high (medians of 3.5 and 3.0), reflecting general awareness of LLM behaviour. Responses converged to the top of the scale on every criterion, and all six comparisons were highly significant (p<0.001), providing strong evidence that a single lesson with NODETOJOY led to a meaningful increase in self-reported understanding across all target concepts. Individualised score changes can be seen in figure 6.

Figure 7 highlights the shift in open response categorisation across the lesson. The progression is clear, from a majority of students showing no/limited evidence of computational thinking, to sixteen students discussing FSMs and MCs explicitly. Before the lesson, many students applied general computing knowledge to the question without any formal terminology. One student "Chipmunk" gave long and relatively sophisticated answers, demonstrating prior knowledge. Post-lesson, their understanding was explicitly linked to the datasets explored, discussing the use of FSMs as Python syntax validators. This suggests that NODETOJOY can help solidify and encourage application of knowledge for particularly able students.

Students with no prior knowledge also demonstrated improved understanding. "Alligator" scored only 1 pre-lesson, but gave an elegant category post-lesson response discussing FSMs as syntax validators versus AI using probabilities to "generate a prediction of the statistically most likely outcome". A further interesting response from "Parrot" included an unprompted, insightful comparison of FSM and MC

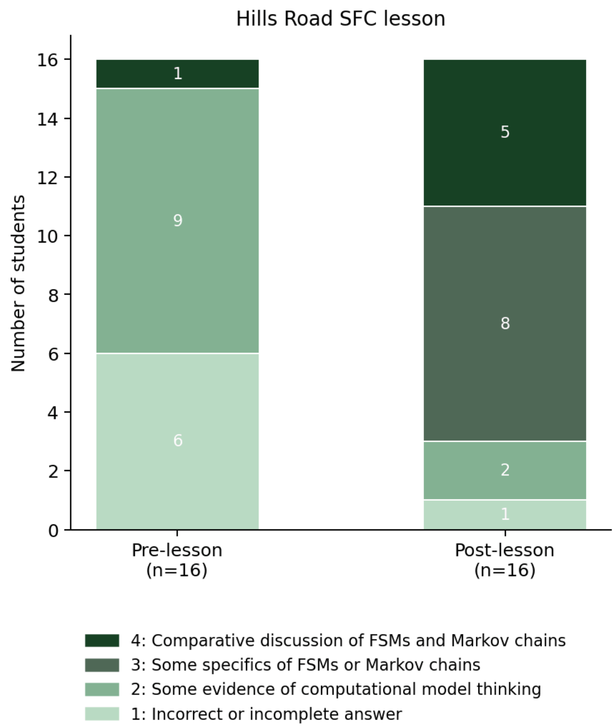


Figure 7 – Comparison of pre-lesson and post-lesson open-ended question answers at Hills Road

similarities, suggesting that the side-by-side design was effective in supporting higher-order thinking skills (Forehand, 2007). More broadly, answers across the cohort shifted towards contrasting structure and probability, reflecting our core aim to emphasise the difference between rule-based and probabilistic inference models.

4.4. Blended year 12/13 CS class

The second lesson took place at All Saints' Catholic Voluntary Academy Sixth Form, Mansfield, with a group of six A Level computer science students, spanning year 12 and 13 (two withdrew before completion). Most had not taken GCSE computer science, so were finding A-Level content challenging. The school faces socioeconomic challenges, reflected in classroom behaviour issues and lower levels of prior computer science knowledge. None of the students were familiar with finite state machines. The lesson was cut slightly short due to technical delays, reducing the time students had to explore later datasets. Throughout the lesson, students were engaged and enjoyed working with a partner to progress through questions. The score breakdown by student can be seen in figure 8, where the data is more spread than at the first school. One student's reducing Likert scores were inconsistent with open questions, suggesting possible non-compliance, misunderstanding, or possibly inflated pre-lesson confidence.

Many students were able to discuss the uses of FSMs or MCs in relation to language and code parsing. One student, "Polarbear", increased from a pre-lesson score of 1 to 3. Their response evolved from discussing "structure" and logical differences between essays and code, to an explanation of probabilistic next word prediction in LLMs, and the risk of LLMs sounding articulate without factually understanding information. "Otter" was able to make a similar jump from discussing machine code to directly identifying the use-cases of FSMs and MCs in relation to the question. "Barracuda" jumped from a classification of 1 to 2, a small gain despite spending 94% of their time actively, answering questions. This could be due to spending over 75% of their time on the first dataset, suggesting that the shortened lesson length prevented deep exploration of the platform.

All pre-lesson responses fell into categories 1 or 2, with post-lesson responses falling between categories 2 or 3. The shift was smaller than at Hills Road, although most students did increase their categorisa-

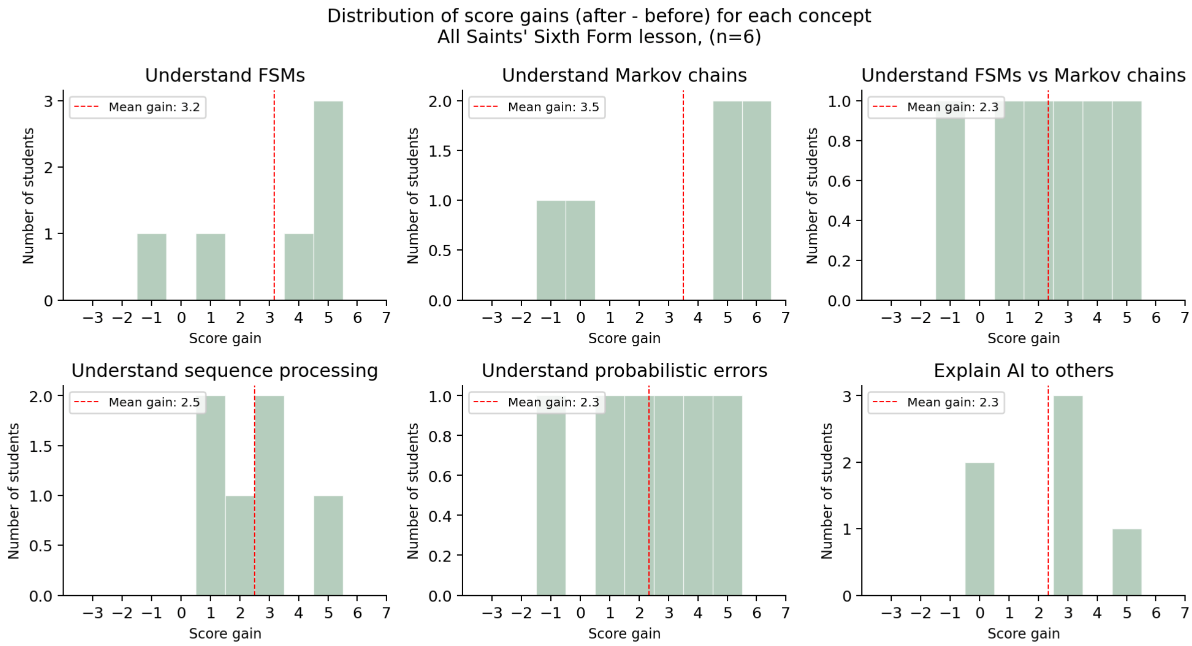


Figure 8 – Distribution of score gains per Likert scale question at All Saints'

tion score by at least 1, perhaps reflecting the reduced time available for self-directed exploration of statistically-sized datasets or a lower baseline of prior knowledge.

4.5. Comparative findings

The pattern of average score gains across concepts is consistent with Bloom's taxonomy, such that lower-order knowledge gains (understanding what FSMs and MCs are) saw the largest increase, whereas higher-order gains (transferring this knowledge to sequence processing more generally, and explaining the link to generative AI language models) saw more modest increases (Forehand, 2007). This is to be expected within a single 1 hour lesson, and suggests that extended use of the platform, or a follow-up lesson, could consolidate these higher-order skills.

The quantitative and qualitative results together support our design and learning goal of developing improved intuitive understanding of likelihood and validity. Across both cohorts, the pooled repeated-measures test found a highly significant overall effect on learning, and 22 students increased their long-form response categorisation scores. Even for students in post-test category 2, there was a shift towards references to sequence processing, rule-governed validation and probabilistic prediction. This supports our aim of critically informed LLM usage.

5. Conclusion

NODETOJOY is, to the best of our knowledge, the first educational platform contrasting rule-based and probabilistic models of computation using interactive visualisation. This comes at a time national curriculum plans call for greater emphasis on probabilistic literacy (Mathematics in Education and Industry (MEI), n.d.), (Royal Statistical Society, 2024). We have created visualisation pipelines to support hierarchically nested subgraphs. Our robust layout algorithms and pipeline can render up to 11,000 nodes. Most significantly, semantic zoom is applied as a novel mechanism for traversing levels of dataset granularity: navigating continuously from coarse, structurally-encapsulating nodes to fine-grained, character-level subgraphs. This allows non-trivial datasets to be abstracted and used approachably in educational contexts. We have introduced a hierarchical Markov chain viewer, exploring a gap that Harel identified in 1987 but did not pursue (Harel, 1987). In-school trials with our target audience of 16/17 year olds demonstrated significant learning gains, among cohorts with a wide range of socioeconomic and prior-knowledge backgrounds.

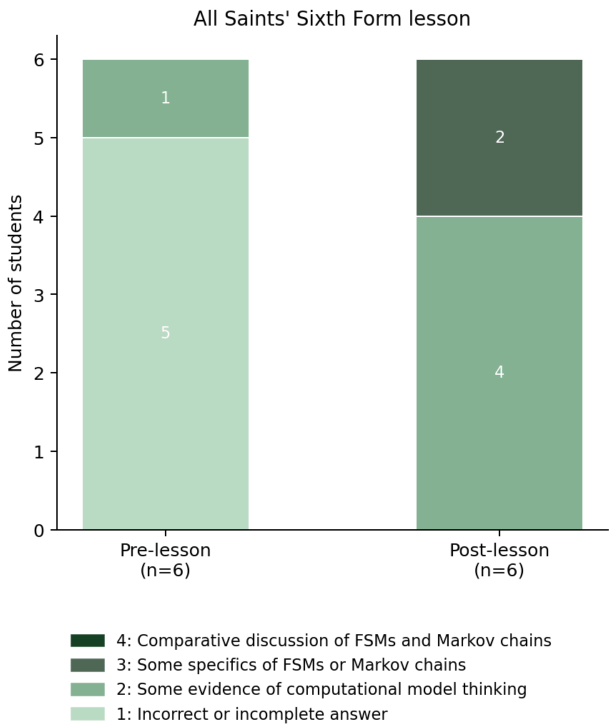


Figure 9 – Comparison of pre-lesson and post-lesson open-ended question answers at All Saints'

6. References

AQA. (n.d.). *A-level Computer Science 7517 Specification.* Retrieved 2026-04-15, from `https://www.aqa.org.uk/subjects/computer-science/a-level/computer-science-7517/specification/specification-at-a-glance`

Baudisch, P., & Rosenholtz, R. (2003). Halo: a technique for visualizing off-screen objects. In *Proceedings of the sigchi conference on human factors in computing systems* (p. 481–488). doi: 10.1145/642611.642695

Ben-Ari, M. (1998). Constructivism in computer science education. In *Proceedings of the twenty-ninth sigcse technical symposium on computer science education* (p. 257–261). doi: 10.1145/273133.274308

Castro, F. E. V. (2022). Exploring the use of finite-state machines and game creation to teach computational thinking in middle schools. In *Proceedings of the 27th ACM Conference on Innovation and Technology in Computer Science Education Vol. 2* (p. 618). doi: 10.1145/3502717.3532137

Cockburn, A., Karlson, A., & Bederson, B. B. (2009, January). A review of overview+detail, zooming, and focus+context interfaces. *ACM Computing Surveys*, *41*(1), 1-31. doi: 10.1145/1456650.1456652

Dengel, A. (2018). Seeking the treasures of theoretical computer science education: Towards educational virtual reality for the visualization of finite state machines. In *2018 ieee international conference on teaching, assessment, and learning for engineering (tale)* (p. 1107-1112). doi: 10.1109/TALE.2018.8615288

Elliott, S., Hudson, B., & O'Reilly, D. (2000, 4). Visualisation and the influence of technology in 'A' level mathematics: A classroom investigation. *Research in Mathematics Education*, *2*(1), 151-168. doi: 10.1080/14794800008520074

Forehand, M. (2007). Bloom's taxonomy. In *Emerging perspectives on learning, teaching, and technology.* Retrieved from `https://resources.finalsite.net/images/v1741805303/d41org/icn48k4luekuivu9f4vm/bloomstaxonomy.pdf`

Furnas, G. W., & Bederson, B. B. (1995). Space-scale diagrams: understanding multiscale interfaces. In

Proceedings of the sigchi conference on human factors in computing systems (p. 234–241). doi: 10.1145/223904.223934

Gates, P. (2018). The importance of diagrams, graphics and other visual representations in STEM teaching. In R. Jorgensen & K. Larkin (Eds.), *STEM Education in the Junior Secondary* (pp. 169–196). Singapore: Springer.

Harel, D. (1987, 6). Statecharts: a visual formalism for complex systems. *Science of Computer Programming*, *8*(3), 231-274. doi: 10.1016/0167-6423(87)90035-9

Iudean, B., & Vescan, A. (2025). Narrative-driven learning: Teaching finite state machines through storytelling. In *Proceedings of the 17th international conference on computer supported education - volume 2: Csedu* (p. 627-638). doi: 10.5220/0013471600003932

Keller, R., Eckert, C. M., & Clarkson, P. J. (2006, March). Matrices or node-link diagrams: which visual representation is better for visualising connectivity models? *Information Visualization*, *5*(1), 62–76. doi: 10.1057/palgrave.ivs.9500116

Kim, M., & Jin, Q. (2022). Studies on visualisation in science classrooms: a systematic literature review. *International Journal of Science Education*, *44*(17), 2613-2631. doi: 10.1080/09500693.2022.2140020

Mathematics in Education and Industry (MEI). (n.d.). *Developing a Level 3 Qualification in data science and AI.* Retrieved 2026-04-15, from `https://mei.org.uk/insight/developing-a-level-3-qualification-in-data-science-and-ai/`

Melero, J., Hernández-Leo, D., & Blat, J. (2012). A review of constructivist learning methods with supporting tooling in ICT higher education: Defining different types of scaffolding. *Journal of Universal Computer Science*, *18*(16), 2334–2360.

Mutua, B. M., & Blackwell, A. F. (2023). Interactive narrative visualization for learning Markov decision process. In *Proceedings of the 34th annual workshop of the psychology of programming interest group* (p. 96-106).

Nimon, K., Zigarmi, D., & Allen, J. (2011). Measures of program effectiveness based on retrospective pretest data: Are all created equal? *American Journal of Evaluation*, *32*(1), 8-28. doi: 10.1177/1098214010378354

Pfannkuch, M., & Budgett, S. (2016). Markov processes: Exploring the use of dynamic visualizations to enhance student understanding. *Journal of Statistics Education*, *24*(2), 63-73. doi: 10.1080/10691898.2016.1207404

Piantadosi, S. T. (2014, 10). Zipf's word frequency law in natural language: A critical review and future directions. *Psychonomic bulletin & review*, *21*(5), 1112. doi: 10.3758/S13423-014-0585-6

Presmeg, N. (2006). Research on visualization in learning and teaching mathematics. In *Handbook of Research on the Psychology of Mathematics Education* (p. 205-235). Leiden, The Netherlands: Brill.

Rodger, S. (2006, 6). Learning automata and formal languages interactively with JFLAP. *ACM SIGCSE Bulletin*, *38*(3), 360-360. doi: 10.1145/1140123.1140270

Royal Statistical Society. (2024). *Key recommendations for the statistics curriculum in the UK.* Retrieved 2026-04-15, from `https://rss.org.uk/RSS/media/File-library/Policy/2024/Recommendations-for-statistics-curriculum-in-the-UK-full-paper-final.pdf?ext=.pdf`

Sweller, J. (2016). Cognitive load theory and computer science education. In *Proceedings of the 47th acm technical symposium on computing science education* (p. 1). doi: 10.1145/2839509.2844549

Touya, G., Gruget, M., & Muehlenhaus, I. (2023, 2). Where am I now? modelling disorientation in pan-scalar maps. *ISPRS International Journal of Geo-Information*, *12*(2). doi: 10.3390/IJGI12020062

Vidal, E., Thollard, F., de la Higuera, C., Casacuberta, F., & Carrasco, R. C. (2005, July). Probabilistic finite-state machines-Part I. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *27*(7), 1013–1025. doi: 10.1109/TPAMI.2005.147

Wang, C. H., Han, D., & Chou, W. K. (2018). Overview of information visualisation in science education. *International Journal of Computational Science and Engineering*, *17*(1), 118-125. doi: 10.1504/IJCSE.2018.094423