

CASE STUDY

An accessible maths journey

Lilian Joy, Digital Education Team, University of York, York, England. Email: lilian.joy@york.ac.uk
Natalie Curran, Maths Department, University of York, York, England. Email: nsec500@york.ac.uk
Cordelia Webb, Maths Department, University of York, York, England. Email: cordelia.webb@york.ac.uk
Maciej Capinski, Maths Department, University of York, York, England. Email: mc545@york.ac.uk

Abstract

The four narratives that follow bring together the stories from the lived experiences of a post-graduate blind maths student, her tutor, her transcriber and a learning technologist over the course of five years. It provides an insight into what is needed to help one student with their own way of learning maths. It also demonstrates how pulling at one thread can help to unravel and reveal the many lenses through which accessible maths needs to be approached.

Keywords: Accessible, Disabled, Blind, Technology, Narrative.

1. Introduction

Stories can help people to come to a shared understanding and to connect with one another (Connelly and Clandinin, 1990). The four narratives that follow bring together the stories from the lived experiences of a blind maths student, her tutor, her transcriber and a learning technologist. We started a 'story conga', a way to capture the lived experience of each person connected to a particular student and letting each other's stories sink in and change our own perspectives of our connection in the story. Each person wrote their own experience and after reading each other's narratives, they edited their story to reflect further insights.

2. Our stories

2.1. A blind maths student's story - Natalie

I lost my sight when I was 8/9 years old. The school I was at did their best to support me, but when it reached the time to start choosing my Standard Grade (GCSE) subjects, we all agreed that it would be in my best academic interests to switch to a school with a specialised 'Visual Impairment Unit'.

One of the main reasons we thought this was a good idea was that 'they'll know how to do the maths properly and how to do different languages in braille and stuff.' Well, they didn't. It's true that they definitely had more time and resources than my previous school, but their approach to braille in the sciences and languages was no more advanced than our own. In other words, write it in braille and then have someone sighted translate the gibberish. The only difference was that they used an electronic braille device for everything rather than swapping out for the mechanical Perkins Braille as I did for Maths.

Perhaps this is a good place to stop and give a quick explanation as to what I am talking about.

- Perkins Braille: a large, clunky, extremely heavy mechanical braille device greatly resembling an old-fashioned type-writer; paper is physically rolled into it and rows of tiny needles are pushed through the paper in the formation the user types, thereby producing lines of braille.

- Braille Note Apex: an electronic device, much smaller, lighter and hence more portable than the aforementioned Perkins. It has much the same capacity as a PC from about 15/20 years ago, an in-built voice function and a strip of 32 (or 18) braille cells which form an electronic braille display in place of a screen.

Apart from the pros and cons already mentioned, the key advantage of the Apex over the Perkins when in text-based classes was the ability to listen to what I'd previously written which is much faster than reading the physical braille. On the other hand, the primary benefit of the Perkins compared to the Apex when in science-based classes was the ability to quickly and easily scan up and down previous lines of working. The bonus of the voice provided by the Apex was almost entirely lost by the fact that the Apex wasn't smart enough - and braille rules were too complicated - for it to be able to tell the difference between English and mathematical constructions. For example, the Apex would read (as in speak and print) the equation: $2+3=5$, as: "2 ;to 3 ;(5" [ignoring the various quotation marks]. I was informed by my new school that I would be expected to type my maths in such a form - i.e. one which only made sense when physically reading the braille - and then pass it to a specialist assistant teacher for translation, before it could then be handed in to the class teacher.

I thought this was a rubbish idea.

I decided there must be a better way. I'd already started lightly exploring some of the deeper, more technical aspects of my Apex, particularly as the thought of maybe wanting to study something mathematical at university had started to flicker across the edges of my imagination. So, I went home and experimented with every kind of file type the Apex had to offer. I read various sections in the user guide and eventually settled on the `computer braille table` file type. I discovered that if I used a different set of braille commands, I could produce braille that felt right to me, sounded right when the Apex read it back and looked right to my class teacher without it having to go through any sort of transcription process! This was computer braille, computer maths and Unicode.

The next day, I went into class and told the assistant teacher about this. I think it's fair to say she was impressed but cautious. We agreed to give my new discovery a go, with the understanding that I'd revert to their original method if we found it wasn't working. That was eleven years ago now, and I am still using the same method.

I would be the first to admit that my system is far from perfect. My Apex does not have access to the entire set of unicode tables, hence there are some symbols which it cannot interpret and we have to work round. There are many symbols which sound and look correct but are displayed in braille as a random combination of dots, or perhaps even worse, written out in words. For example: "left-brace" appears on my display whenever the "{" character is used; far from ideal when dealing with set notation. I also have to spend a bit of time undoing all the quirks which appear when reading a file sent to me because, for some reason, the Apex does some strange things such as swapping all Greek characters for normal letters of the alphabet but with different accents added to them. e.g. "α" becomes "á". And then there's the obvious fact that most people do not use computer maths for extended amounts of mathematical text. It is regarded as somewhat crude compared to the more advanced approaches such as LaTeX.

Nevertheless, these disadvantages are far outweighed by the benefits:

1. I can produce maths myself which is then immediately legible and understandable to sighted readers;
2. I can use the speech feature on my Apex fairly effectively;
3. I can assign shortcuts and commands to the symbols I use most frequently;

4. My materials can be made accessible to me by anyone at a similar level of mathematical understanding or higher, without any need for specialist training or knowledge of braille;
5. I can email my lecturers questions quickly and directly, and read their replies, without the need for any transcription;
6. If my Apex were to suddenly break, I would be able to continue my studies, though not as easily, with a laptop and braille display.

It has even proved useful in other subjects, such as Philosophy - when learning about logical notation; and French/Spanish - because it gave me access to all the different accents used in these languages.

In short, whilst unicode, computer maths and computer braille create nothing like an ideal system, I am yet to find anything better.

Since working with Lilian (learning technologist), I have been made aware that it may be possible to read mathematical material on websites or online journal articles that are not written in computer maths. If we can find a way of doing this easily and effectively, then it could give me access to a whole host of content which has hitherto been closed off to me, thereby bringing me that tiny bit closer to an equal playing field with my peers.

We managed to get JAWS (on my pc), combined with the Mathjax¹ menu, and Voiceover (on my iPhone) to successfully read equations aloud. However, no one in their right minds would learn maths simply by listening to it, and the braille output each was able to present had its own problems.

The other most obvious issue with this solution is that it is purely web-based. It does not address the problems of how I save this information; edit it in order to make my own notes or comments; or create and output my own maths, which is clearly legible to both myself and my tutors. Even if we allow for my own system neutralising the final negative here, the first two elements are still significant barriers in their own right.

2.2. A tutor's story – Maciej

I first came across Natalie when I first started marking her exams. This was a bit challenging as the format was quite different: her solutions were submitted in plain text so were less readable than the LaTeX or Lyx used by other students, and the text was also more scattered making it harder to navigate. Otherwise, it was a standard student submission. There were issues when the exercise was to produce a graph in Excel but instead, Natalie made tables with the data. Geometric exercises were also much harder for Natalie as in some areas, much can be read from a picture which posed problems for her. It is also really important that Natalie has the transcription as it would be impossible to do anything without them.

Knowing it would be a challenge to teach her, I talked to my father (also an instructor on the course) who would make an effort to reply quickly when Natalie would post questions in text format. Mathematics is a peculiar subject as it can take weeks to understand a text but the same information can be conveyed within an hour's seminar. There is some magic involved in talking over real-time mathematics. With COVID, tools have been developed to teach online but it always seems to work if you start with a blank file and create the maths in real-time. This approach also allows the students

¹ Mathjax is "a JavaScript display engine for mathematics that works in all browsers". When it is implemented, people can right-click (or select) the maths (and open the Applications menu) to see extra accessibility functions and they can set these for the whole page.

to ask questions when they feel something is wrong: often the maths is correct and they don't understand something but this allows for discussion and further explanation. It was therefore important to find a format that allowed me to explain things to Natalie in real time that was not just limited to text. In higher mathematics, you need to discuss arguments and this is much harder if you cannot "see" the equations.

After some trial and error, we found Framapad¹ which is a web page where I can type text and Natalie can read it on her braille reader in real time. If you copy and paste mathematical text symbols such as \int or $\sqrt{\quad}$, these appear as required. This still requires simplification as very long formulas are less readable but this compactification is common in mathematics: we understand F to be $F(x, t)$ which saves writing the latter every time. I was astonished and surprised by the technology limitations that Natalie must cope with: her screen reader has under 20 characters that can be read in a single line. I had imagined that they were like A4 paper but instead she can only read one line at a time and that line cannot be very long. Frequently, Natalie would have to jump between lines to see the text that I was typing. This system also works for the coding aspects of the course compared to other sites that required Natalie to jump between windows.

Lecturer's notes for students.

Lemma. (Ito) If $X(t)$ is an Ito process:

$$dX(t) = a(t)dt + b(t)dW(t)$$

and $F(t, x)$ is a $C^{1,2}$ function. Then $F(t, X(t))$ is an Ito process; moreover

$$dF(t, X(t)) = \frac{\partial F}{\partial t}(t, X(t))dt + \frac{\partial F}{\partial x}(t, X(t))a(t)dt + \frac{\partial F}{\partial x}(t, X(t))b(t)dW(t) + \frac{1}{2} \frac{\partial^2 F}{\partial x^2}(t, X(t))b^2(t)dt.$$

Figure 4 The original set of lecture notes used for sighted students.

Lecturer's notes for Natalie.

Ito lemma.
 If $X(t)$ is an Ito process:
 $dX(t)=a(t)dt+b(t)dW(t)$
 and $F(t,x)$ is a $C^{(1,2)}$ function. Then $F(t,X(t))$ is an Ito process, moreover
 $dF(t,X(t)) = F_t(t,X(t))dt + F_x(t,X(t)) a(t) dt + F_x(t,X(t)) b(t) dW(t) + 0.5*F_{xx}(t,X(t)) b^2(t) dt.$

Figure 5 The compactified version produced with the online text editor for Natalie.

For Excel documents, we use Google Spreadsheets as we can work on the same spreadsheet, and I could see where her cursor is so I can direct her to the next cell. I could add text which she could read in real-time. This works well but is much much slower than when meeting with other students.

¹ Framapad (<https://framapad.org/abc/en/>) is an installation of Etherpad (<https://etherpad.org/>), a simple open source collaborative online text editor.

To be honest, I'm amazed at how she is able to work with big and complicated spreadsheets with only 20 characters at a time. It's mind-blowing!

Key tips for working with Natalie

Having shorter meetings on a regular basis: We have a system where we have an hour-long meeting with each student concentrating on 3 different modules. This is an efficient and effective system for most students but, for Natalie, everything takes much more time. It is better if she sends the questions upfront so the answers can be prepared to suit her. For most students, I can just share my screen, highlight the relevant parts of the lecture notes for example and maybe write some more explanation if needed. For Natalie, there's no point in sharing my screen and I need to think about the best way to write the particular part in a compact form, find an explanation which avoids technical derivations or construct a simple example to sell the main point. Doing three modules in one meeting is therefore difficult so it is better to meet more regularly.

Understand her limitations: I was surprised by the limitations of the technology that Natalie has: there is a real need for technological advancement in the tools that blind students have access to and there should be readers which cover more. Mathematical equations are often one after the other and you need a grasp of, say 3 lines simultaneously. Some proofs are very technical and a sequence of equations in a PDF can take two pages. In this case, having an overview is different to being able to scroll line by line. This is an engineering problem and providing help to maths students is not quite enough of an incentive but there should be a "braille ipad" which would help a wider audience.

2.3.A transcriber's story – Cordelia

My first step to becoming a transcriber was replying to a general email sent around the maths department to find someone to help transcribe diagrams for Natalie. This was in March 2020 and, one lockdown later, I was employed by the Learning Support Centre (LSC) who coordinated the support work for students with disabilities at the University of York.

I've since helped students in subjects including Maths, Physics and Electronic Engineering. These are very different subjects both in the type of technical content they have and how it is taught: this means note taking is varied and needs to be tailored to suit both the subject and the individual student.

Notetaking: how the lecturers help (or don't!)

From a notetaking perspective, the easiest lectures for me to transcribe are those where the lecturer handwrites (clearly!) on a board. The handwriting ensures the pace of the lecture is reasonable and helps clarify what information is most important.

With COVID and the subsequent shift to more online content, I have notetaken almost exclusively online, remotely and asynchronously. This has enabled me to help more students but also made me reliant on Panopto, the lecture capture system used in York.

It is much easier to read and make accurate notes when handwritten lectures are done digitally or through a visualiser. The only downside to this is that content is lost from view at the "end of the page" rather than on chalkboards where, when used systematically, content can remain visible for a much longer period. Generally chalkboard lectures are hard to read with lecture capture as cameras are typically at the back of the room and don't always successfully follow the lecturer across the boards.

For most of these subjects, slides are not extensively used. When they are, the pace is often faster and the content often more quantitative: Maths almost never uses slides while the less calculation-based Physics content has many more slide-based course notes.

Lecturers mostly engage positively with recording lectures although some have the opinion that students should be attending in person. While it is important that students fully engage with their courses, it is also important to remember that not everyone watching the lecture capture is having another hour in bed: there are a range of reasons students watch the lecture capture and this may include students who attended the original lecture in person or, like me, support workers. There were times where I was timetabled to be supporting 4 students simultaneously. It is only possible for these 4 students to have the support work they require if I can work from the lecture recordings.

Notetaking: what I do

I've supported students with a range of different educational support needs and have made notes mostly in Word as this was what the original training suggested and what most students requested as accessible and useful for them. However, there is now more debate about this training for science and maths subjects as well as more discussion about encouraging students to embrace other software and technologies.

Recently, I have been using MathType¹ to input more mathematical content. This did slow me down at first as it took time to find a way to use this efficiently and smoothly among non-technical content. However, now I am more used to the technology, it is faster and makes it easier to output alternative formats.

Transcription: what I did and what I do now

For Natalie, the bulk of the transcription work normally occurs over the summer before term begins. I receive the material as PDF documents in a Google Drive which I transcribe into Rich Text Format (RTF) documents.

Previously, I copied and pasted the PDF content to a Word document. However:

- line breaks occurred after each line from the PDF;
- within the text, certain letter combinations (e.g. ff, fi) were interpreted as symbols rather than letters so needed to be corrected;
- mathematical content lost many symbols (e.g. σ , Σ) and formatting (1/2 pastes as 1 2 with a line break);
- punctuation was incorrectly copied (, as ; for example)

All of these had to be corrected manually which was time consuming and error prone. Find and Replace was the most important shortcut to help with this and it could also be used to help meet Natalie's particular needs and preferences. For example, Natalie requires no space after an = sign which can occur hundreds of times within a document.

¹ MathType (<https://www.wiris.com/en/mathtype/>) is a formula editor and equation writer by the company, Design Science. Also known as Wiris MathType, it is the Maths editor found in the Blackboard VLE. It can also be bought as an add-on for Microsoft Office.

After meeting Lilian (learning technologist), I found out about Equatio¹ and used that to screenshot and copy mathematical content into LaTeX. I later found out that I could get the LaTeX file from the lecturers. This made my life easier as nearly all the issues from above were fixed. However, there were downsides:

- additional LaTeX formatting and comments had to be removed;
- mathematical content had to be translated from LaTeX.

Again, Find and Replace went a long way to addressing these problems: I could easily and unambiguously replace `\sigma` with σ but LaTeX has a lot of formatting syntax so more needed to be stripped out for Natalie. For example, `\frac{\sigma^2}{2}` can be simplified to $\sigma^2/2$. Lecturers also need to supply the LaTeX file which is not always comfortable, especially if the file is not “tidy” and contains lots of comments.

In terms of technologies that can help, MathPix² is the fastest alternative we have found to date. I can take the PDF supplied for the whole course and output a Word document where the text appears as required; there are no additional line breaks and mathematical content is displayed using the Word equation editor. With Mathtype installed, I can quickly translate all the maths to plain text with a keyboard shortcut which is much closer to what Natalie needs. There are also no additional commands and text is translated accurately and with no additional line breaks. Although I still have to go through the document for particular quirks, it has more than halved the transcribing time and reduced the errors that can slip through.

Lecture notes formatted with LaTeX.

In the proof we shall need the following lemma, a version of the Bayes formula.

Lemma 7.2

If $Q(A) = \mathbb{E}_P(\mathbb{I}_A f)$, $f \geq 0$, $\mathbb{E}_P(f|X) < \infty$, $\mathbb{E}_Q(|X|) < \infty$ and $\mathcal{G} \subset \mathcal{F}$, then

$$\mathbb{E}_P(fX|\mathcal{G}) = \mathbb{E}_Q(X|\mathcal{G})\mathbb{E}_P(f|\mathcal{G}).$$

This lemma was proved in the Black–Scholes module.

Proof of Proposition 7.1

\implies) Suppose that $\frac{X(t)}{S_1(t)}$ is P_1 -martingale. Then, by using Lemma 7.2 with $P = P_1$,

Figure 6 A section of the PDF lecture notes shows an equation in a call out box.

¹ Equatio (<https://www.texthelp.com/en-gb/products/equatio/>) is a tool by Texthelp that allows users to create equations, formulas, mathematical and scientific notation, through voice, screenshot, handwriting or typing at a keyboard.

² Mathpix (<http://mathpix.com>) is a conversion tool that can convert handwritten maths, images and PDFs to a range of outputs including Markdown, LaTeX (including straight to Overleaf), DOCX, Excel or even ChemDraw. It also has a markdown editor.

Lecture notes for Natalie with formatting removed.

In the proof we shall need the following lemma, a version of the Bayes formula.

Lemma 7.2

If $Q(A) = \mathbb{E}_P(J_A * f)$, $f \geq 0$, $\mathbb{E}_P(f * |X|) < \infty$, $\mathbb{E}_Q(|X|) < \infty$ and $\mathcal{G} \subset \mathcal{F}$, then $\mathbb{E}_P(f * X | \mathcal{G}) = \mathbb{E}_Q(X | \mathcal{G}) * \mathbb{E}_P(f | \mathcal{G})$.

This lemma was proved in the Black-Scholes module.

Proof of Proposition 7.1

\Rightarrow Suppose that $X(t)/S_1(t)$ is P_1 -martingale. Then, by using Lemma 7.2 with $P = P_1$, $Q = P_2$, $f = M_1^2(t)$, $Y = X(t)/S_2(t)$, and $\mathcal{G} = \mathcal{F}_s$, we get

Figure 7 The transcribed version of Fig 3 for Natalie has formatting removed and the equations are linear.

General advice to lecturers

1. **Be aware of the limitations of the technology you are using:** lecture capture does not always accurately follow you and PDF documents are not always the best output for students
2. **Embrace the technology:** use lecture capture as it benefits many students in numerous ways and using a visualiser or tablet to write content is easier to read than a blackboard but still retains that “chalk and talk” feel.
3. **Help students help themselves:** releasing material in alternative formats such as HTML lets students be more independent and gives them a range of ways to engage with content. Moreover, students may not be aware how they can best use these different formats so consider pointing them towards different software and assistive technologies.
4. **Lead by example:** by making your content more accessible, you can also encourage students to make their content accessible which benefits them but may also make it easier for lecturers to mark their work too! Lecturers and markers have accessibility needs too!

2.4.A learning technologist's story - Lilian

I work as an education adviser and learning technologist with a specialist area of digital accessibility at the University. I first started on my accessible maths journey in 2019 after spending a day with Natalie, who was then an undergraduate maths student. I wanted to know how materials were being converted for her and how she was learning higher level maths. She did her best to explain the challenges and why her own maths method worked for her, but I didn't yet have the baseline knowledge about braille, maths and the various formats and tools to do her justice. It led me to start my research journey into accessible maths.

My colleague and I organised a discussion about accessible STEM which made us realise that most lecturers produce their notes using LaTeX output as Beamer¹ slides or PDF, followed by Microsoft Equation Editor or Mathtype in Word or PowerPoint and, to a lesser extent, R. There was also a lot

¹ Beamer is a LaTeX editor for creating slides featuring overlays and animation. It is an alternative to PowerPoint and provides support for more complex mathematical notation.

of handwritten content. As I delved more into the world of accessible maths, I realised that I had opened a can of worms - there was a lot of material on the virtual learning environment (VLE) that had mathematical or STEM content. These could not easily be converted into other formats by the accessibility tool, Ally¹, that we had acquired for our VLE, unless the maths was accessible in the first place. I set about looking at each format and how we could make it more accessible. The university funded several projects with student interns so that we could work out what needed to be done and the best workflows for us. Through the research on these projects, we found some tools that were useful to help with the creation of alternative formats when necessary.

We acquired licences for MathType to encourage people away from using Microsoft equation editor in Word and PowerPoint. Microsoft's equation editor has its own ecosystem for users but doesn't allow us to output alternative formats easily. MathType however can be output to a range of formats including HTML with MathJax. MathType is also the maths editor within Blackboard, which is our VLE at the University.

We acquired Texthelp Equatio as this enabled us to capture handwritten maths and convert it into digital formats. Instructors could also input maths easily by handwriting, typing or using LaTeX, thereby encouraging a digital-first workflow. Equatio could be used with Microsoft Office documents, Google Documents as well as on the VLE. It was also a valuable tool that enabled students to help themselves. We developed a worksheet to help students to learn the tool and we would run workshops with departments to help students with it.

LaTeX took me a while to grasp but four things really helped:

1. Jim Tyson from University College London (UCL) had kindly shared his notes on creating accessible files from LaTeX. I sent this to lecturers who reported back that they were able to follow Jim's methods and produce nice web pages or Word documents from their LaTeX files. However, it didn't always produce the kind of referenced equations that maths lecturers like.
2. Having my own digital maths intern who could talk me through the basics of LaTeX and experiment with me to work out a template that could be useful to lecturers, providing them with accessibility advice and a format they could use.
3. Having access to Overleaf² meant I could dabble with LaTeX without too much effort.
4. Getting Pandoc³ made available through our software centre meant lecturers could easily add this to their managed computers. Pandoc can be used to convert documents from one format to another.

We produced a LaTeX template with accessibility guidance built-in. Although it uses an accessibility package that is no longer supported (since 2022), it provides us with a way to output PDFs that pass

¹ Ally (<https://www.anthology.com/products/teaching-and-learning/learning-effectiveness/anthology-ally>) is an add-on to the Blackboard, Moodle or Canvas VLE (Ally for web is a sister-product for other types of websites). Ally provides guidance and feedback to instructors/page owners on the accessibility of their online information, and it allows the students/readers to download documents in a variety of accessible formats.

² Overleaf (<https://www.overleaf.com/>) is an online collaborative LaTeX editor.

³ Pandoc (<https://pandoc.org/index.html>) is a tool run on a PC that can convert documents between a wide range of file types.

the accessibility checkers while we're waiting for LaTeX to become more accessible by design. We are currently working on advice around using Mathpix for this process too.

Web-based materials are seen as the gold-standard for digital accessibility, but students are still keen to get their learning materials as PDFs (Hughes, 2019). If the lecturers can easily generate different formats, students can help themselves to the format that works best for them. So far, this is the main challenge for some lecturers, especially those who use LaTeX to produce more complicated content. Some lecturers have successfully pivoted to using alternative systems like R Markdown to produce their notes (Dias, 2022). However, students also need to be aware of the benefits of using web-based versions and they need access to the right tools to help them use PDFs more flexibly for notetaking and revision.

Both Mathpix and EquatIO are tools that we feel can really help staff and students to create accessible documents for themselves. They can convert handwritten materials to digital, depending on how clear the original handwritten materials were in the first place! Editing the digital conversion is quite straightforward in both software.

Handwritten maths notes.

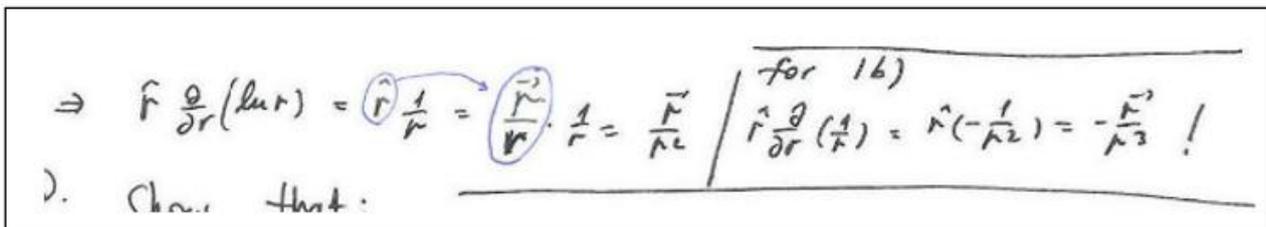


Figure 8 Some handwritten annotations that Mathpix struggled to understand and therefore imported as an image

Equatio's screenshot editor in action.

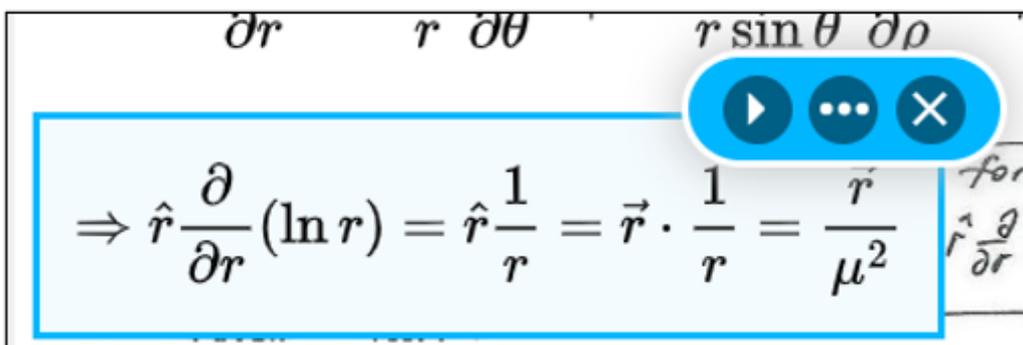


Figure 9 EquatIO provides a first stab at converting the handwritten maths to digital maths.

Students already have access to EquatIO but we're considering whether to provide Mathpix too as it would allow them to convert PDFs into more editable formats so they can make notes and learn some LaTeX too.

Our journey towards digital-first accessible maths is still on-going, but we now have several students, instructors and departments working together as part of a maths accessibility working group. We output everything we learn to our Accessible Equations web site¹. We help to chair the Jisc Accessible Maths working group² where several members are working together to pool our knowledge and skills to progress accessible maths across the sector.

My journey was started by one conversation with Natalie, one attempt to understand things from her perspective that launched me into a whole chain of events, creating further opportunities to work with her and to bring others on board. Natalie has driven my passion for user research and the ambition to make mathematical materials more accessible at the University. This 'story conga' has revealed to me how much more there is to be aware of as I marvel at Maciej's inventive approach to teaching Natalie online, and Cordelia's persistence with different technologies to work out the most efficient approach to transcription. By working together, we have built an infrastructure and mapped a way towards accessible maths (Webb, 2022) that benefits all staff and students at the University.

Advice for learning technologists:

- If you're a learning technologist interested in progressing accessible maths at your organisation, you'll have to collaborate with many people on several projects to make it happen. See Accessible Maths blog post³.
- There isn't a one size fits all and learning technologists and organisations need to engage with a diverse range of tools and workflows to enable all staff and all students to produce accessible maths.
- Start by speaking to
 - one student about their workflow and their needs and go from there.
 - one lecturer who has had to teach a visually impaired student.
 - transcribers who work with and support the students.
 - your disability team to find out what they recommend and who they work with.

3. References

Connelly, F. M. and Clandinin, D. J. (1990) 'Stories of Experience and Narrative Inquiry', *Educational Researcher*, 19(5), pp.2–14. Available at: <https://doi.org/10.2307/1176100>.

Dias, A. (2022) 'Developing digital accessibility using R Markdown', *Forum Magazine*, 49, p.19. Available at <https://bit.ly/adiasMarkdown>

Hughes, C. (2019) 'Making lecture notes more accessible', unpublished.

Webb, C. (2022) 'Mapping the way to accessible mathematics', *Forum Magazine*, 50, p.25. Available at <https://bit.ly/webbCMapping>

¹ Accessible Equations web site for the University of York, <http://bit.ly/eaccess-equations2>.

² Jisc Accessible Maths working group have a web site at <https://github.com/A11yMaths>.

³ Accessible maths blog post (<https://elearningyork.wpcostaging.com/2021/09/27/accessible-maths/>)