Are chatbots taking over?

Chatbots in user assistance and multiple languages

Communicator

The Institute of Scientific and Technical Communicators
Spring 2019

Create dyslexia-friendly documents

Captivate content for the digital era

Stop being an imposter

Changing viewpoints over the decades

Captivating content for the digital era

Beyond the book. Andrew Mills looks at the possibilities for creating interactive content for the digital generation, and beyond.

As someone who enjoys learning through both direct interaction and visual learning formats, the captivating, immersive, and interactive worlds of Augmented Reality (AR) and Virtual Reality (VR) seem to directly hit both areas. This makes them highly attractive to me from a technical communicator's perspective.

I've watched numerous presentations over the last couple of years that show just how engaging AR and VR content can be in an educational context. One example that really stood out to me was an experiment in a primary school that involved eight-year olds walking across the Great Wall of China in VR.

There were no expectations set by the teachers on any learning. The children were simply told to have a look around and have fun. Throughout their VR journey, various historical facts appeared on their screens as hotspots (such as how many bricks were used in its construction).

Is there any value in using AR and VR for technical training?

You could not only hear the school children express their enthusiasm, but also see the huge smiles on their faces while they roamed around this great wonder of the world.

When the experiment was over, the teacher nonchalantly asked the children questions based on the various hotspots that were in the VR world. For every question, nearly all of the children raised their hands and the majority of them could enthusiastically recall the correct answers.

This, to me, seemed to prove that AR and VR had some amazing educational value. However, the real question for me was: *Is there any value in using AR and VR for technical training?*

I've come to discover that the answer is: most likely, but it will all be in the appropriate context.

Learning styles

One of the most important realisations I came to over the last few years is that while everyone learns differently, there's still a well-defined variety of delivery formats. These include:

- Text,
- Static Images,
- Audio,
- Video,
- Interactive.

As a technical communicator, I'm very keen to see if there's any way of incorporating all, or as many as possible, of these content delivery formats into a single workflow (ensuring maximum educational value, content, and asset reuse).

The benefits of visual learning formats

Visual context

As someone who prefers to learn visually, I've personally found that I can understand, and digest, the gist of what's being taught better as the visual aspect provides me with invaluable additional context. For example, when I want to learn new DIY skills, reading a book with static images only helps me so far.

So, I now automatically turn to YouTube and search for the best reviewed video tutorial for the task I want to complete. Immediately, the visual cues, along with high-quality commentary, gives me the confidence that I can replicate the task correctly.

Memorable

Personally, I find that when I'm reading just text, the information doesn't seem to 'stick' as well in my mind. Well-designed images and/or videos grab my attention much more easily.

I remember being shown a VR demonstration on the inside of a black taxi. This demo was designed for those who aren't familiar with how black taxis work. The demo allowed me to interact and inspect different parts of the inside of a virtual black cab and learn about each key area. There were animated sections that smoothly transitioned in the text into my view.

Higher levels of engagement

One of the biggest draws with AR and VR as visual formats is that they're both *extremely* interactive and, therefore, have the potential to be highly engaging. At their core AR, and especially VR, applications are designed to draw the user into the experience and immerse them fully in the moment.

The other beautiful thing about both AR and VR is that they can utilise the full-spectrum of delivery formats. Both AR and VR apps can be designed with text (via hotspots for example), audio narration, background music and sound effects, interactive 3D models, and even separate video clips.

Of course, not every AR and VR app *should* be filled to the brim with interactive sections. You may want to keep things much simpler and reduce the risk of things going wrong (such as software crashes for example).

A great example I saw of this simpler approach was an AR tutorial on how to change the ink cartridge on a printer. The screen was uncluttered and allowed the end-user to focus purely on the one task at hand.

However, it's fair to say that the more engaging and immersive you can make your AR or VR app, the higher the chances that the user will remember the information that's contained within, as they will be more focused on what you show them.

VR goals

I feel that it's worth seeing if the interactive nature of AR and VR can offer a knowledge retention value to users who are wanting to learn more about a subject in a visually stimulating manner.

There are several VR goals that I want to achieve during this experiment:

1. Make learning fun and really engaging,
2. Cater for visual and 'doing' learning styles,
3. Surprise the trainee,
4. Keep costs down with branded cardboard glasses (think Google Cardboard).

VR development challenges

One of the first key misconceptions to mention is that Virtual Reality doesn't *have* to involve using very expensive headsets hooked up to powerful PCs to work. In fact, using a phone app and placing it inside a cardboard viewer can work more than well enough.

I'll always remember the smile on my young daughter's face when she tried out VR for the first time (on my phone and in a cardboard viewer). She was completely amazed that in one minute she could be looking inside, and

Figure 1. My budget-friendly VR R&D kit

Figure 2. Gear 360 spherical camera. Brilliant for creating 4k 360° videos and images. © Samsung

interacting with, a Pharaoh's Egyptian tomb, and the next, she could be looking around the middle of outer space and taking a 360-degree close-up view of the various planets that make up our solar system.

Of course, dedicated VR set-ups (such as the Oculus Rift and Vive) offer much greater resolution and interaction options. But you can achieve some very creative and memorable results with almost *any* modern smartphone.

The value proposition

The main question that I need to keep in the back of my mind at all times is: "Where's the value in putting the time and effort into developing educational and training materials in AR and VR?" As I'm still firmly in the learning and experimental phase of this project, it's not a question I can fully answer yet. But, I'm confident that there's value to be had. The value may just be limited to more specific training scenarios (which I've yet to determine).

3D or 360 (that is the question)

VR isn't strictly tied to 'realistic' 3D worlds. There is in fact a *much* cheaper – and easier – option. 360° cameras (such as the 1st generation 4K Samsung Gear 360 camera) take both static images and full-motion videos in 360°.

Granted, while the level of interaction possible is greatly reduced via this route, prototype development time is also reduced. You can, however, still create immersive learning environments that work with cardboard VR headsets.

360° videos are fully supported in YouTube (as long as the correct metadata is present in the final video file), and this allows you to quickly generate test videos and try them in your headset within minutes.

If you want to move between multiple 360° videos, then there are existing websites that use JavaScript and HTML 5 to link these videos

Where's the value in putting the time and effort into developing training materials in AR and VR?

Figure 3. Testing prototype 360° YouTube videos on a VR headset

together (as well as allowing you to create information 'hotspots' should you want to increase the level of engagement even further).

There are a number of third-party websites that I have been using over the last couple of years to both simplify the development process of 360° videos, and also to help me understand how they work 'under the hood'.

These sites are:

- https://jaxry.github.io/panorama-to-cubemap (Panoramic to cubemap image conversion),
- www.360toolkit.co (Converts cubemaps to equi-rectangular images),
- www.adobe.com/uk/products/premiere.html (video editing, other editing tools may be OK),
- https://github.com/google/spatial-media/releases (Injects VR metadata into video files),
- www.marzipano.net (A tool for creating interactive 360° images).

The usual indemnity warning must be mentioned: while I've safely used these sites for working on my own development experiments before, please remember that you use them at your own risk.

Unity

While there are many different AR and VR third-party tools available on the market today (and they're maturing by the month), I opted to dig a little deeper and went with Unity as it's very versatile and flexible as a development tool.

As Unity is such a popular middleware tool, there's a thriving built-in marketplace that allows you to rapidly develop a prototype app with either no cost, or very little (depending on the complexity of your needs).

Unity is also a multiplatform development tool, and this therefore allows you to create Android, iOS, and even PC application bundles from the same project. Naturally, the more platforms you wish to cater for, the more complex your optimisations will become.

Finally, there are loads of tutorials available online on how to get the most from Unity. You *don't* have to be a coder to make a working prototype in Unity (although, it certainly wouldn't hurt if you have relevant coding skills). So, don't let a lack of coding experience put you off.

Assets

You won't have an experience to try without any assets to use. Sourcing 2D images, 3D models, audio, and text is vital if you haven't got the skills to create them yourself. Thankfully, there's a huge asset marketplace available both in Unity and online in general, suitable for all budget ranges (including free if you're just interested in creating a 'rough and ready' prototype.

The brain-eye connection and motion sickness

One of the most important issues with VR is the prevalence of motion sickness when using a headset. And the perceived connection your brain makes to this virtual world cannot be understated.

In one particularly memorable VR session, I was standing up in the middle of a swinging pirate ship ride. As soon as the ship started swinging backwards and forwards, my body involuntarily moved into a protective stance position in an attempt to counter the swinging momentum of the virtual ship.

Bear in mind that, in reality, I was simply standing in a small bedroom and not moving at all. My brain, on the other hand, freaked out and I eventually had to bail out of the virtual scenario as my legs were starting to tense up more and more as the height of the swinging intensified.

With regards to motion sickness, it is generally caused when there's a mismatch between what your eyes are seeing movement wise, and what signals your inner ears are sending to your brain. For example, if your eyes perceive you to be running and jumping around a scene, but your inner ears are registering you as sitting still, then there's an immediate disparity that can trigger motion sickness.

Mismatched frame rates and refresh rates are a common cause of motion sickness. For example, if the frame rate of your VR app constantly alternates between 40 and 50 frames per second (fps), but the display screen's refresh rate is set to 60Hz (which displays a maximum of 60fps). You don't want this.

A lack of a 'focal point' is another contributing factor. A lot of VR and first-person video games (such as Mirror's Edge) combat this using a small fixed dot in the centre of the screen. This 'focal point' allows your eyes to lock onto a fixed on-screen element that never changes location.

There's a lot more involved in this area, and it's one that gets very interesting, but also increasingly complex, the deeper down that rabbit hole you go. It's certainly something you must keep in the forefront of your mind at all times when developing for VR.

Mobile phone or dedicated headset

While dedicated headsets offer vastly superior performance benefits (thanks to more powerful and dedicated hardware), the number of people who have access to a PC powerful enough to run VR smoothly enough to prevent motion sickness is somewhat limited.

In comparison, the vast majority of smartphones these days contain an accelerometer and gyroscope and enough CPU and GPU power to hand in a respectable performance (especially if you keep 3D model optimisations and 'culling' as a key component of your development).

AR goals

There are several goals that I want to achieve with Augmented Reality:

1. Make learning fun and really engaging,
2. Learn how to blend virtual 3D models with physical printed models (such as 3D printing or cardboard),
3. Provide a tactile touch interface via the mobile device screen,
4. Online distribution options (potentially reducing off-site face-to-face training costs).

With that in mind, it's time to delve into some of the obstacles that need to be faced when looking to create an engaging, and fluid, AR app.

AR development challenges

Augmented Reality presents its own unique set of considerations and challenges. It also presents a great opportunity to reuse almost all of the assets created for a VR world (3D models, 2D images, audio, videos, and text can all be repurposed in an AR app – if designed accordingly in the beginning).

With that in mind, let's take a closer look at the challenges faced when looking to develop educational AR apps.

What AR development software to use

The only development software I've used for creating AR is already built-in to Unity and it is known as Vuforia. There are, however, plenty of other AR development options available online, and their level of complexity and functionality, usually, determines their cost.

Designing AR target images

A 'Target Image' is effectively an image that is designed to tell the AR app what specific model or image to display on screen when the camera is pointed at it. AR has advanced to the point where 'targetless' options exist, but they're an area I have yet to try.

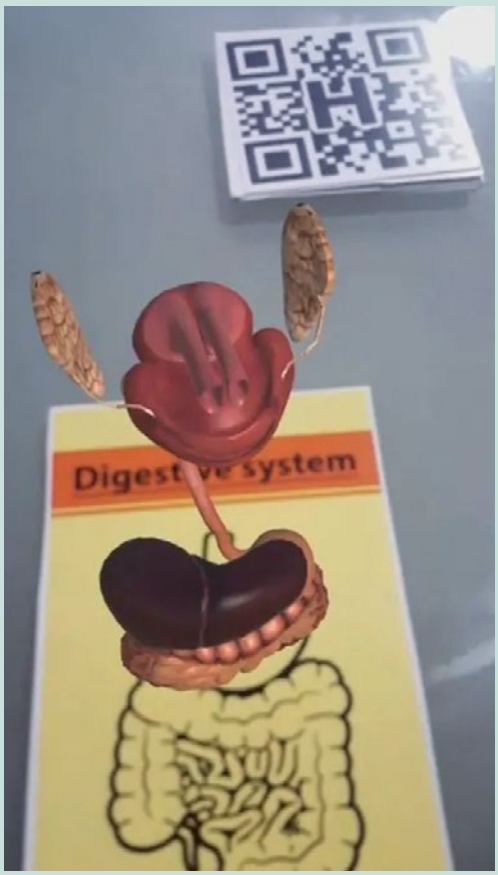

Figure 4. AR screenshot. Using AR to bring education to life. © Panther Studio (AR-3D Science app)

Creating an AR target image requires a lot of straight lines and/or right angles. This is due to the AR software needing to plot out as many unique data reference points as possible. Creating target images that are too similar can very easily confuse the AR software into displaying the wrong image on-screen.

Combining tactile and digital realms together

AR presents a great opportunity to superimpose virtual models on top of objects in the 'real world'. For example, you can construct a basic cube from cardboard, draw a unique target image on each cube face and then the AR app can change the displayed image immediately.

In fact, creating cardboard models is a really quick and cheap way to experiment on how to display information over the top (or beside) the cardboard model. If the cardboard is white, then you can stick it together and draw image targets on each required side.

You can then test to see if your interactive information hotspots, video clips, or any other information displays on-screen as intended. I'll let your mind do the rest but, suffice to say, that the creative options available to you here are only limited by your imagination.

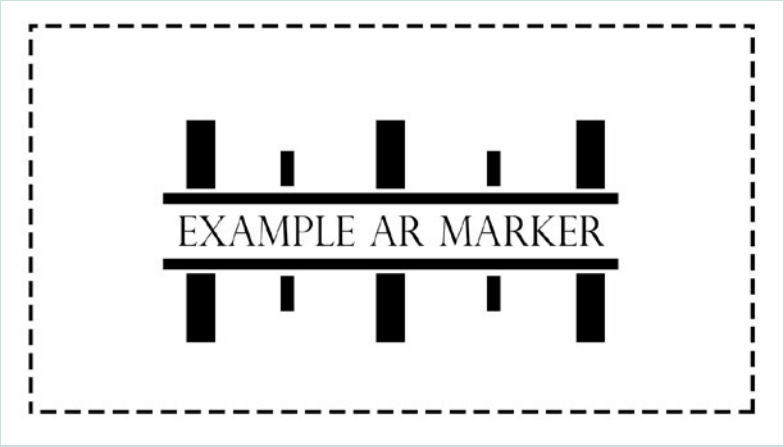

Figure 5. Use straight lines in your AR markers and keep the design simple

The production workflow

As I mentioned near the beginning of this article, one of my main goals as a technical communicator was to create a workflow that focuses very heavily on content reuse. Thankfully, as you can see in the following image, text, 2D visuals, voice overs, and 3D models get reused across all key content delivery formats.

Ideally, all content has been carefully created so a minimal amount of editing is required between different formats. For example, if the text was good enough for the script and subtitles in a video, then, in my opinion at least, it should be good enough to be used in a written format as is.

This is key as it ensures that no one is left out when it comes to their preferred learning method. For those who like to read, the text already exists. For those, like me, who really enjoy the more visual aspects of learning, then AR and VR opens up amazing opportunities to engage with this learning material in a way that resonates better with them.

My personal challenges

When it comes to delivering this project, there are a few key challenges that I always need to keep in the forefront of my mind. These include:

Acquiring stakeholders buy-in

Getting stakeholder buy-in is one of the trickier, and essential, aspects of any internal project. For this project to make it out of a prototyping stage, it's imperative that it gets the buy-in of key stakeholders across multiple different areas of the business (if you don't work for yourself of course).

While this aspect is still very much a 'Work-in-Progress', it's essential that you take a robust and reasonably interactive demo to those that have the power to integrate this further into the company's internal longer-term training plans.

This is both a blessing and a curse, as while AR and VR apps – by their very nature – are engaging formats for enjoying content (therefore making them great for demo-ing), they aren't as easy to develop to a high standard as say, videos, or images.

Therefore, any flakiness in the quality of the prototype will be more easily picked-up on (especially if you fail to ensure that no one gets any motion sickness in VR)!

Value measurement

Tied in very closely with my previous point, it's essential to be able to measure and quantify how much value these applications are providing the company commissioning their development.

From an educational standpoint, one potential measurement option is by asking trainees a series of questions based on the content they have been shown in the AR or VR experience. The final scores providing an indication to how effective the formats were at explaining complex concepts to trainees who aren't necessarily engineers.

It's still an area I'm investigating, and as soon as I come up with a reliable set of measurements that provide the answers I'm looking for, I'll make sure to tell you all.

How much to learn versus how much to outsource

One of our most valuable skills as technical communicators is our ability to learn new and complex information and software relatively quickly. And, as someone who loves to up my skills and learn new software, working out how to both create content in Unity and the assets that go along with it, is quite a tantalising challenge.

However, I am only one person, so a key consideration is in deciding just how much I need to learn, versus how much can be completed by finding those internally who already have the required skills.

The future

It's a very exciting time for AR and VR in the educational space. From what I've seen so far, there's an incredible amount of untapped potential waiting to be harnessed. As mobile and dedicated VR technology improves, so too will the visual fidelity and quality of the learning on offer.

Mixed-Reality (MR or XR for Cross-Reality) is an interesting hybrid that mixes together AR and VR into one world. Hand tools allow you to manipulate AR models in a VR space and they offer an additional way to interact directly with the content that's being displayed in front of you.

Packages, such as Zapbox, make this a much more affordable option as it provides an all-cardboard package (at the expense of reduced tracking accuracy).

Dedicated Mixed Reality headsets, such as the Magic Leap One, offer much better tracking for a far smoother end user experience, but come at a substantially greater monetary cost.

Either way, it's an amazing time to explore these areas as a technical communicator, and I truly hope your brain has conjured up some exciting, and creative ways to potentially deliver an engaging, and educational, visually-led experience to your audience. C

Andrew Mills FISTC. His first published guide was printed in March 1999. From 2003 onwards, he became the 'go-to' video game guide author for most of Future Publishing – Europe's largest magazine publisher.

Over 140 printed guides across different gaming publications were printed over the following seven years.

Andrew continued to create and self-publish game guides for digital platforms, such as iOS applications and e-books, before moving into the role of Technical Author in the business software sector in 2015.

Note: All views and experiences shared in this article are those solely of the author.
E: andrew@technicallyanauthor.com

Resources:

Third-party websites to simplify the development of 360° videos

Adobe. Video editing that's always a cut above' www.adobe.com/uk/products/premiere.html (accessed January 2019)

Alshaker O. '360 Tool Kit' www.360toolkit.co (accessed January 2019)

Crane L. 'Panorama to Cubemap' https://jaxry.github.io/panorama-to-cubemap (accessed January 2019)

Gorzel M (2018) 'Spatial Media Metadata Injector v2.1' https://github.com/google/spatial-media/releases (accessed January 2019)

Marzipano. www.marzipano.net (accessed January 2019)

Other resources

Google Cardboard. https://vr.google.com/cardboard (accessed January 2019)

Unity. https://unity3d.com (accessed January 2019)

Zapbox. www.zappar.com/zapbox (accessed January 2019)

Communicator articles

Görz J (2018) 'Scalable solutions for augmented reality' *Communicator*, Autumn 2018: 18-20

Judge K (2015) 'Virtual reality in games' *Communicator*, Spring 2015: 24

Luft T (2015) 'The future of virtual reality' *Communicator*, Summer 2015: 44-45

Patel F (2016) 'Amplifying content with augmented reality' *Communicator*, Spring 2016: 29-33

Ressaire T (2015) 'Virtual user engagement' *Communicator*, Spring 2015: 20-24