In a 1950 paper, Alan Turing, who is considered by many to be the founder of the modern computer, posed the question "Can computers think?" The answers to versions of this question have been widely debated over the last 70 years (Turing 433). Some believe artificial general intelligence, where AI is able to outperform us in every task, is only a few decades away, while others believe that this is obviously impossible. But everyone who has an iOS device with Siri or a device that supports Alexa can agree that we are not there yet: personal assistants like these make dismal conversation partners. They often misunderstand us, and they simply feel glaringly unnatural—one reason they fall so short of the mark is their complete lack of ability to produce and interpret humor. This usually isn't implemented in today's personal assistants because it's quite difficult to do in any capacity, and we're not anywhere close to the technology we need to make humor generation and perception in AI that matches the ability of even an average person. But of course, those aiming to develop artificial general intelligence that can match us in all our skills will need to overcome this obstacle, because humor is unquestionably a way in which we express ourselves and display intelligence. So today, I'd like to talk about the progress that has been made in the production of humor by machines, which is known as computational humor, and what these efforts show about the field and about humor more generally. Since this is a field that's quite new, more progress is being made in more rudimentary areas like pun generators, while much less progress is being made in more complex and contextual humor like you would hear in a stand-up routine. These different levels of progress show the path ahead to developing more complex models in this field, and they highlight which areas of humor are more concrete and which require more knowledge and context to produce and appreciate.

So I think the first natural question to ask is: is it even possible for AI to generate humor? And just to be clear, I'm talking about jokes that are actually written by a computer, so if you ask Siri or Alexa to tell you a joke, that wouldn't count because that's just pulling from jokes written by a human. It makes sense to be skeptical that this is even possible, because we do often think of humor as being extremely human and it's usually part of our self-constructed identity. So as to whether computers can create humor on par or funnier than that of humans, the short answer is "not yet", because this is still a field in its early stages. While we've already seen a number of programs that can write novel puns and the like that at least make sense and are passable as jokes, even the best ones would be unlikely to get a reaction out of most people. But to say this means it isn't possible would be to ignore the trend we've seen over and over again with other areas in AI: it starts off terrible compared to humans, and then eventually surpasses us. For example, computers designed to play chess

started off making moves almost at random when they were first designed, and now even a mediocre chess engine can beat the best human players fairly easily. In the same way that machine learning was used for chess, language and humor models will likely keep improving as well as the creativity of AI with regards to humor. While we can't say for sure that AI-generated humor will one day be able to match us in terms of making people laugh, based on current trends it certainly seems plausible and maybe even likely. Some may respond to this by saying that AI-generated humor "doesn't count" because the computer doesn't actually understand the jokes or isn't able to appreciate them, and I'll touch on this later, but this turns out to be quite a subjective idea that's difficult to discuss meaningfully in a concrete way.

Another question that naturally arises is why we would want to develop artificial intelligence that can be funny in the first place. One reason is one that I've already touched on—if we want to develop artificial general intelligence, then it'll need to be fully proficient in humor by definition (because AGI is as good or better at every human task than us). But of course, the natural response to this is "Why then do we want to develop AGI?" So one reason is that it'll allow more seamless communication between AI and humans. We use humor constantly when talking with each other and writing, so AI will need a model of humor to be able to understand us, and to communicate with us in a way that feels understandable and natural. As Anton Nijholt (a leading researcher in computational humor) said in his 2006 paper, "Humor helps to regulate a conversation and can help to establish common ground between conversational partners. It makes conversation enjoyable and supports interpersonal attraction" (Nijholt 62). So if we want to use AI as a tool, we need to improve its communication, and implementing humor is one way in which we can. A lot of the papers in this topic discussed ways it could be implemented into personal assistants and user interfaces, as computers could use humor to diffuse frustration from users and make products more engaging. One example of this Graeme Ritchie discusses is with education—students using AI programs to learn, as is already becoming more commonplace, would likely be more engaged if the program utilized humor effectively (Ritchie 77).

This leads quite nicely into one of the really interesting projects that has already been done with computational humor. It was a project in 2008 by a team of researchers in this field called STANDUP, and their results were published in a peer-reviewed journal called Applied Artificial Intelligence. So STANDUP stands for "System To Augment Non-speakers' Dialogue Using Puns", and it was basically a program designed by these researchers to help children with cerebral palsy (which usually involves an impaired speaking and reading ability). Since these children wouldn't have

had as many opportunities to develop language skills and engage in typical language play with peers, the idea was to create a virtual playground of sorts that would allow these children to "explore sounds and meanings by making up jokes, with computer assistance" (Manurung 3). It was more of a proof of concept than a product they were selling, but these researchers really did make an effort to make a program that would be engaging and helpful to the kids.

They hired a graphic designer to design this interface (Manurung 31), so it's supposed to be intuitive for anyone to use. And since these children had lower reading abilities, it would run the jokes it generated through a text-to-speech so they could listen to them. The user could pick which type of joke they want, which included typical structures such as "What do you get when you cross a blank with a blank?", "What do you call a blank with a blank?", and "What's the difference between blank and blank?". So these structures are obviously human ones, but the main part of the project was being able to algorithmically fill in the blanks and create novel jokes within these structures.

One example the authors give to demonstrate how these jokes are generated is "What do you call a shout with a window? A computer scream". Basically, the program has access to a lexical database (similar to a dictionary), and to a list of words that sound similar, or homophones. With this one, the pattern it was using looks for homophones and then combines them in this way. So it found that "scream" and "screen" were homophones, and then looks for synonyms for those words and finds "shout" and "window" as synonyms, and then essentially just fills these components into a formula. The humor, if it's funny, stems from the ambiguity between the homophones that almost sound the same but have different meanings (Manurung 10).

I've included some of the other jokes that the program created that the authors included in the paper, and as you can see, the quality varies pretty widely (Manurung 29-30). These first few sound comparable to what you might find on something like a laffy-taffy wrapper. They make sense, and while they're arguably not that funny, a lot of people don't find puns particularly funny in general—that's the reason it's often socially acceptable to groan at puns rather than laugh. And then this last joke really doesn't even make sense, and it's one they presented as an especially bad joke that the program generated. And that was one of the issues they discussed: it would sometimes pick nouns that were too vague and weren't funny as a result. But overall, the researchers actually had a lot of success—they didn't do a super formal evaluation of the children's reactions, but based on anecdotal evidence, the kids enjoyed using the program and used its feature to share some of the jokes they liked with family and friends, and their families often reported that their communication seemed to improve

after using STANDUP. So this is a great example of how even simple implementations of computational humor can be effective and actually useful.

There have been a number of other studies that have also done some really interesting things with computational humor, although they weren't leveraged to benefit people quite as directly as STANDUP. One is HAHAcronym (Stock and Strapparava), which essentially takes in already existing acronyms and algorithmically generates a new meaning for the acronym that's supposed to be funny. And the way it tries to make them funny is by having the new meaning opposed in one sense to the original one while maintaining similarity in other senses, which is known as a comparison axis. So one example they give that the program generated was it took MIT as an input, which stands for Massachusetts Institute of Technology, and the acronym it came up with was the Mythical Institute of Theology. The opposition here is that of technology versus religion (Stock and Strapparava 143), since technology is exclusively rooted in the scientific method while religion is usually rooted in other things as well like faith. So the program saw "Technology" in MIT and replaced it with "Theology", which sounds similar because it rhymes but they're opposed based on the schema of technology vs. religion. And to add to this opposition, it replaces "Massachusetts" with "Mythical" because that adds to the opposition with technology. I've also included some of the other examples here. And many of the acronyms scored as funny with a group that's described in the paper, even when they didn't know they were written by an AI (Stock and Strapparava 143).

One more study I want to touch on was by Tinholt and Nijholt 2007 that uses a similar technique of opposition. The idea was that it would take in sentences that use ambiguous pronouns, and then if it deemed the ambiguity to be humorous, it would make a comment based off of that. So, to use their example from their paper, if you have the sentence "The cops arrested the demonstrators because they were violent," the "they" could technically refer to the demonstrators or the cops. And so you get 2 different scripts, where the implied one is "The demonstrators were violent, and as a result they were arrested by the cops", but the other one is "The cops were violent, and as a result they arrested the demonstrators". And in this case, this has the same sort of comparison axis as we saw in HAHAcronym where the opposition could be something like rowdy vs. orderly, so the program might generate a response like asking "The cops were violent?" (Tinholt and Nijholt 479). So the program looks to see if the ambiguous antecedent in the given sentence could reference two things that are opposed in some way. Another example the paper gives is if you input "Our lawyers put your money in little bags, then we have trained dogs bury them around town", it would determine that "lawyers" and "bags" are opposed (as one is inanimate and the other is living), and as a

result outputs "Do they bury the bags or the lawyers?" And importantly, the bags being buried, which is the implied script, comes before the "funny" interpretation of the lawyers being buried, because the humor comes from the normal script being overwritten by the incongruent one.

There have been other studies in computational humor, but those are some of the more well-known and more-developed ones in this field. Each of these studies is reasonably successful in their stated goal: they create at least some jokes that are arguably funny, and STANDUP appears to actually have had a positive impact on its users' language development. And this makes sense, because the goals were fairly limited. STANDUP's users were children who hadn't had as much experience with humor as other children their age, so they were unlikely to be very cynical about the quality of the jokes. And the other two studies were very narrow applications of script opposition, so they didn't have quite as many ways in which the jokes could fall short. Another thing to note is that the outputs that the researchers presented in the paper were likely to be some of the better jokes that were generated, and they mention that not all of the humor produced was of the same quality. But overall, these projects did stand up to some external judging and produced some humor that's arguably of good quality, and they're definitely interesting proofs of concept that can be extended. So in light of these findings, I'd like to close by discussing what extensions of these projects in computational humor can show about humor, as well as how they'll affect our relationship with AI.

These findings, and ones that may come in the future, are an interesting way to discuss the complexity of humor, because it'll be much harder for a computer to generate and interpret humor if it's more complex and layered than if it's simple. The jokes produced in these studies were extremely structured, and the computer was simply looking for words that fit the pattern described by the programmer. They also weren't using much cultural context—the script oppositions showed in HAHAcronym and the referential humor used a bit of this context, but they didn't have a great deal of information about each word and concept and how it fits into our culture. One example from a paper by Bergen and Colson that would be hard for a computer to interpret is the joke "Everyone had so much fun diving from the tree into the swimming pool, we decided to put in a little water". The humor from this comes from the script we initially have of people having fun diving into a swimming pool full of water, and expecting to hear something like "put in a diving board", and then having to reevaluate that script and realize people were jumping off a tree and landing, probably 15 feet below them, on a concrete pool floor. That joke makes sense to us because we implicitly understand what features pools and trees usually have, but it's not so easy to turn that into a

model a computer can understand (Bergen and Colson 59). And finally, the jokes in these findings are very isolated, and it'll be more difficult for a computer to naturally and effectively use humor in a longer conversation, or to develop a stand-up routine that connects jokes smoothly. So all of this is to say that the structured, isolated jokes that AI is already able to create are on the more simplistic side of humor, and how difficult it is for a computer to create or understand a joke may be a way we can talk about the complexity of humor in a somewhat concrete way.

Finally, I want to touch on the point I raised at the beginning about whether any of this actually "counts" as humor, since the computer doesn't necessarily understand the humor in the same way that we do. And this is something that Turing addresses in the paper from 1950 I mentioned, which introduces the idea of a Turing test. So the basic idea of a Turing test is that you have a judge that's communicating through text to a human and to an AI without knowing which one is which, and the AI passes the test if the judge isn't able to determine which entity is human. Turing's answer to whether or not a computer can "think" is whether or not it can pass a Turing test, because if it isn't distinguishable from a human in communication, he argued it must have some human-like cognitive qualities. Shah and Warwick actually conducted a series of Turing tests in 2017 where the judge didn't always correctly identify the human, and this was in many cases because the AI attempted to use humor in conversation, even if it didn't go perfectly, because using humor feels human to us and the authors believe that's why they passed the test (Shah and Warwick 554). Of course, the judges would have eventually been able to tell, and misidentified the AI because they were only allowed to spend 5 minutes asking questions. But if we get to a point where computer-generated responses are truly indistinguishable from human ones, Turing argues that it's difficult to prove that the computer doesn't understand the humor. One might argue that since a computer is composed entirely of simple binary bits, it's absurd that it could truly understand humor, but one could argue that our brains are composed of simple neurons that also don't understand anything on their own, but we still can appreciate humor subjectively (Turing 446). It's difficult for us to prove to each other that we're having a subjective experience, so if a computer tells us that it's having a subjective experience, we shouldn't be so quick to dismiss it. Computational humor, when taken to its full extent, raises many fascinating questions like these, and it'll hopefully offer many benefits like I've already discussed. It will be exciting to see how it keeps playing out over the next century.

Thank you.

Note

Each paragraph corresponds to a slide, so the 13th paragraph is the note for the 13th slide.

References

Shah, Huma & Warwick, Kevin. 2017. Machine humour: examples from Turing test experiments. *AI & Society* 32(4). 553-561.

Turing, Alan. 1950. Computing Machinery and Intelligence. *Mind: A Quarterly Review of Psychology and Philosophy* 59(4). 443-460.

Bergen, Benjamin & Coulson, Seana. Frame-Shifting Humor in Simulation-Based Language Understanding. *IEEE Intelligent Systems* 21(2). 59–62.

Nijholt, Anton. Embodied Conversational Agents: "A Little Humor Too". *IEEE Intelligent Systems* 21(2). 59–62.

Ritchie, Graeme. 2009. Can computers create humor? *AI Magazine* 30(3). 71-81.

Manurung, Ruli; Ritchie, Graeme; Pain, Helen; Waller, Annalu; O'Mara, Dave; & Black, Rolf. The construction of a pun generator for language skills development. *Applied Artificial Intelligence* 22(9). 841–869.

Stock, Oliviero & Strapparava, Carlo. The act of creating humorous acronyms. *Applied Artificial Intelligence* 19(2). 137–151.