A story about a tiny bot

Marco Antonio Andrade Barrera

RSS 2019 Conference, Belfast, Northern Ireland

Speech and slides

I would like to begin by telling you the origin of the story about a tiny bot, which is the topic of the article I wrote for the S*ignificance* writing competition. Last April, I was on holidays at home. I was very sad because my crush thinks that I'm a nice guy but just that — the dreaded friend zone situation. I was anxious and very stressed. I couldn't stop thinking about it. But suddenly a saying came to my mind: "don't worry, get busy". I'm not sure if the saying was applied to the right context, but I started working on the competition. But what story to write?

Well, I started remembering a situation when I was in my classroom after a Bayesian inference class, and two friends were talking about some artificial intelligence concerns, such as improving human abilities and simulating complex brain processes like feeling fear or love, and the consciousness.

I was thinking about it and suddenly I became aware of my awareness and a question was raised regarding my current emotional state. Could the kind of situations, where adversity starts a spark of creativity, be simulated using technology?

We can find optimist researchers who think that it could happen in decades, though others, like Dr. Rodney Brooks[1] from MIT, estimate hundreds of years.

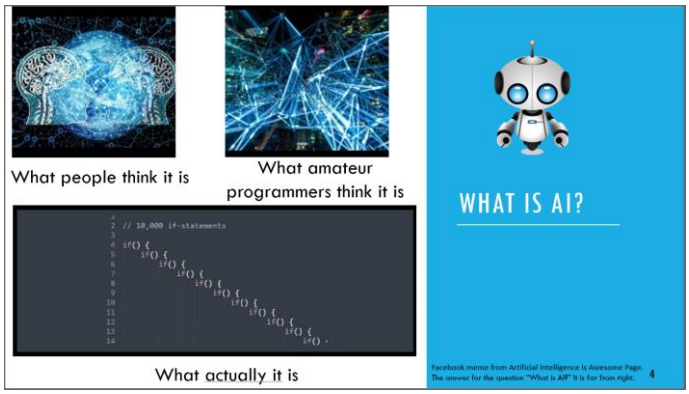
No one can be completely sure about this. However, one thing is sure right now: AI is a very powerful tool that, little by little, is a more intimate part of our lives than ever. Today, we can find AIs capable of driving cars, detecting diseases or recognizing objects, and I like to think about these AIs, and technology in general, as tools that can be attached to humans to make us better at least in some ways. I mean, It allows us to cross the Atlantic in a few hours or save and analyze thousands of GB of data. This kinds of things are not natural, we can achieve them because of these tools.

In this regard, I thought "I will create a bot and I will write a story about it". The purpose is to show that it's becoming increasingly easier to create bots and that there is a logical path to creating them in which the use of data and statistics is the core.

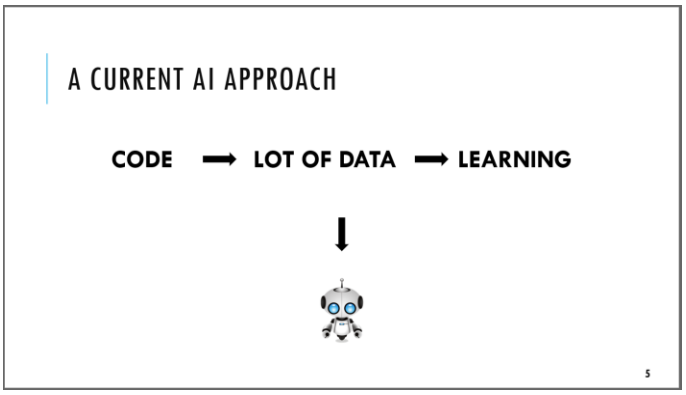
When I wrote the story, I tried to do it in a thought-provoking way. I didn't do it only making a description of the work. I also tried to write a story in a way where the bot would have consciousness, which of course now it's considered sci-fi. However it could possibly be real in the future. Hopefully, the article will be published, and soon you will be able to read it online.

But today I'm not here to talk about sci-fi. I would like to talk about the real deal. And what is that? The real part is that AI can provide us with abilities to do very complex processes or at least speed up time-consuming processes.

[1] Brooks, 2019. Robots, AI, and other stuff. https://rodneybrooks.com/agi-has-been-delayed/.



The other day I saw a Facebook meme which describes AI. What people think it is, what amateur programmers think it is, and what it actually is. So, it states that AI is just "if statements". And it's funny because this answer to the question "What is AI" it's completely wrong. Since there are also "else" statements. Of course I'm joking.

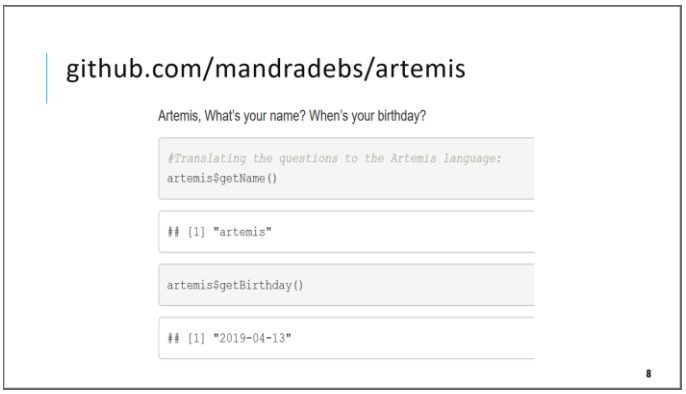


A current approach when we talk about an AI that emulates specialized human thinking or behavior is simple and there are not a lot of "if statements". We write a few lines of code and we call this set of code, for example, a bot. Afterward, with the bot ready, we just give it lots of data about some specific subject. Then, it will start learning from this data and derives its own rules and, if you want, its own "if statements" to achieve some specific goals. It's like the idea of raising a child. You don't teach kids a rule for every situation, instead, they have positive or negative experiences and then they figure it out on their own. I am calling these experiences "data".

Now I will talk about my bot. It has 355 lines of R code, it's very short. After creating the bot, I gave it a bunch of data and now he's capable of identifying stock chart patterns. His name is Artemis, which is also the name of my cat.

The bot's structure is based on object-oriented programming. From a simple point of view, it means that Artemis has features — like name or birthday — and abilities — like print an article or download some data from the Internet. You can have just one Artemis but you can also make copies of him. So, you can have a group or army of these bots.

Artemis can be downloaded from my GitHub account. Here is the link. You also will find on the same page, a short guide to install and start using him. Artemis doesn't have a friendly interface to interact with humans yet, but it's very easy to talk with him.



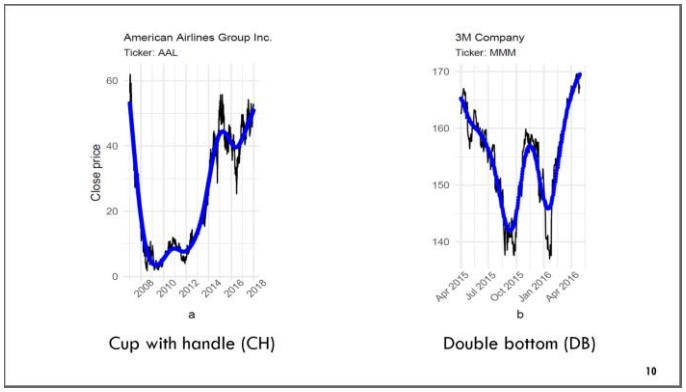
For example, we can send messages to him and he will answer back. To ask his name, we just write this line on the R console: Artemis + dollar sign + getName and he will answer with his name. The same applies for asking for his date of birth.

Someone who feels familiar with object-oriented programming could say, that's not AI, it's just an object with some very simple methods. And that's true, but it's the beginning of this bot and as we'll see at the end, he will be able to perform a very specialized task.

A couple of hours after the birth of Artemis, I gave him a new ability. I called it "generate study material". It means that someone provides a list of tickers to Artemis. A ticker is an arrangement of characters representing securities. For example, AMZN is for the Amazon Company and AAL is for the American Airlines Company. Then

Artemis browses the Yahoo Website to download the related historical prices. This work is done using an R package called *quantmod*. I gave him 503 tickers and he downloaded the price time series for all of them, from 2007 to his date of birth. All this data is saved in a feature called “study material”. We can think about this as the books used by a kid when he goes to school. It’s something similar for Artemis.

A few days later, I added a new ability to Artemis. I call it *learn* and it allows to Artemis to gather examples of chart patterns. Until now, I have taught him two patterns:



Cup with handle and double bottom. You can see examples of them in the figure. Cup with handle pattern looks like a cup profile with the handle at the right, just like Figure A. In Figure B, we can see a double bottom which looks like a W.

The learning ability does the following: It takes a time series representing a pattern, for example, a cup with handle like Figure A. Afterwards, Artemis takes his study material and splits every time series into subsets to determine possible candidates for the pattern. Then he compares each candidate with Figure A using a similarity measure, for example the Euclidean Distance. Here comes an interesting thing, since it’s how Artemis gets his knowledge.

So, until now, Artemis has a similarity measure for each candidate, which is just a number representing how similar is a candidate to the cup with handle pattern. After that, Artemis sorts the candidates from high to low then displays a plot for each candidate and asks, “Does the plot contain a true cup with handle pattern?” A human should answer, and in this case, that human was me. All answers are saved in an Artemis’ feature which I named *knowledge*.

With the *knowledge* ready, a new ability is required since the knowledge isn’t enough to test how well Artemis has learned the patterns. A training process is needed. Practice makes perfect.

I named this new ability *train*. It basically uses two R packages for machine learning, *caret* and *h2o*. I asked Artemis to use 5 methods: 1) k-nearest neighbors, 2) neural networks, 3) random forest, 4) support vector machines, and 5) logistic regression. To train the 5 methods, Artemis splits his knowledge into two subsets, one of them is used to training the models and select its best parameters. The other subset allows testing how good are the selected models on new data.

Table 1: Performance statistics by method and pattern name.

		Testing dataset		
Pattern	Method	Accuracy	Sensitivity	Specificity
DB	**deepNet**	**0.83**	**0.83**	**0.82**
DB	**svmRadial**	**0.87**	**0.81**	**0.91**
DB	knn	0.83	0.68	0.92
DB	rf	0.82	0.69	0.90
DB	logistic	0.63	0.63	0.63
CH	**deepNet**	**0.92**	**0.68**	**0.97**
CH	**svmRadial**	**0.93**	**0.62**	**0.99**
CH	rf	0.90	0.53	0.98
CH	knn	0.89	0.35	0.99
CH	logistic	0.84	0.50	0.90

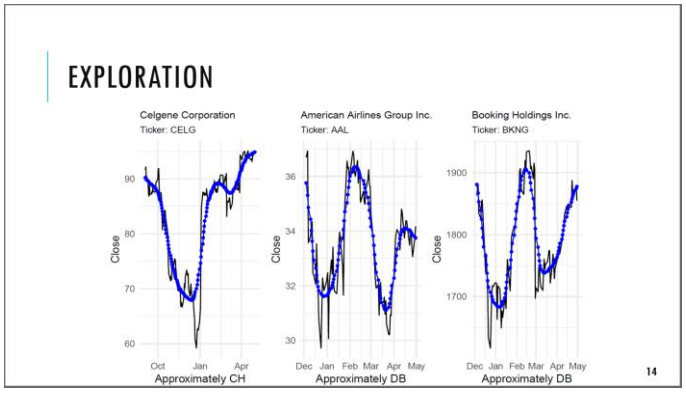
12

Here are the performance statistics by method and pattern name. Look just at the bold numbers. Considering the accuracy, the third column, neural networks and support vector machines were the best training methods. They got around 0.8 and 0.9, meaning that for each 10 guessing trials Artemis failed roughly once and twice. Looking at specificity, the last column, which is the proportion of true negatives correctly identified, something similar occurred, but almost without fail on the cup with handle patterns. And looking at sensitivity, the proportion of true positives correctly identified, Artemis failed to recognize approximately 2 out of 10 double bottoms and 3 out of 10 cups with handle. Of course, the results were not perfect, with the worst results in the sensitivity, but Artemis has another great ability to take advantage of these models.

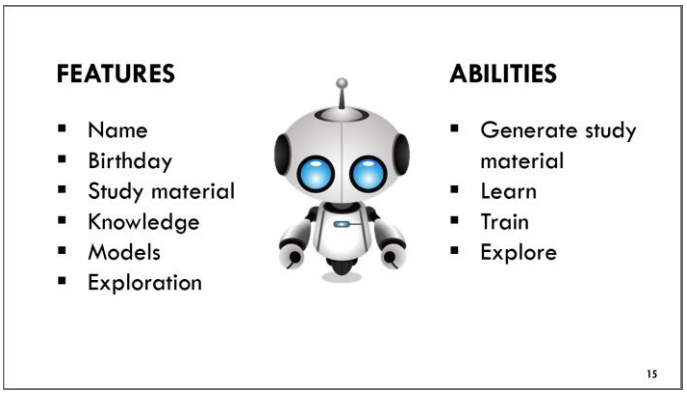
I named this last ability *explore*. Artemis just needs a list of tickers to start, then he goes to the Yahoo Website and downloads the daily

close prices for each ticker. Afterwards, he applies his trained models to determine the presence of cups with handle and double bottoms. I'm happy with this because Artemis can explore one or one hundred tickers and unlike me, his performance is not affected by the amount of work. He never feels tired, disappointed, bored or sad like humans. He's free from all these limitations so, the job is always done at the same level.

To test this last ability, I gave Artemis a list of 100 tickers and he began the exploration. In the best case scenario, a manual exploration would take me upwards of two hours.



For Artemis, it took 19 minutes on a mainstream laptop. At the beginning, there were around 1,000 candidates and he could reduce the list to 55 possible patterns. Then he printed a plot for each of the 55 patterns and it took me around 5 minutes to scroll through the plot and pick up just the three presented in the figure. The first one looks like a cup with handle while the remaining two are double bottoms.



In summary, Artemis has features like a name, date of birth, study material, knowledge, models which are the result of his training, and exploration where he saves the exploration results. Also, he has abilities like generate a study material, learn, train and explore.

That's all. He is Artemis and he can perform identification tasks at a much higher rate than me. I am not sure yet if he is better than me in his accuracy, but time will tell. Indeed, there is a lot of things to be done. For example, we can teach Artemis lots of other patterns and improve his knowledge and learning ability by considering other similarity measures, not just the Euclidean Distance and exploring other machine learning algorithms. It's even possible to add a bold ability to Artemis — he could be able to buy and sell stocks on his own, and this is not something new. There are books about creating automatic trading systems.

As a final aside, I confess that Artemis didn't help me too much with my emotional issues, but it brought me here. I like the financial world and I am a raw investor. But, I am completely sure that this kinds of tools are wonderful at least when working on repetitive stuff which can be done by bots and it's relatively easy to create them. It provides us with free time and we can spend it on more valuable things, like thinking, writing and telling stories.

I would like to thank David Juárez, Dario Díaz, Arturo Alvarado and Carlos Octavio Pérez for their help and excellent comments when I was working on this speech.