

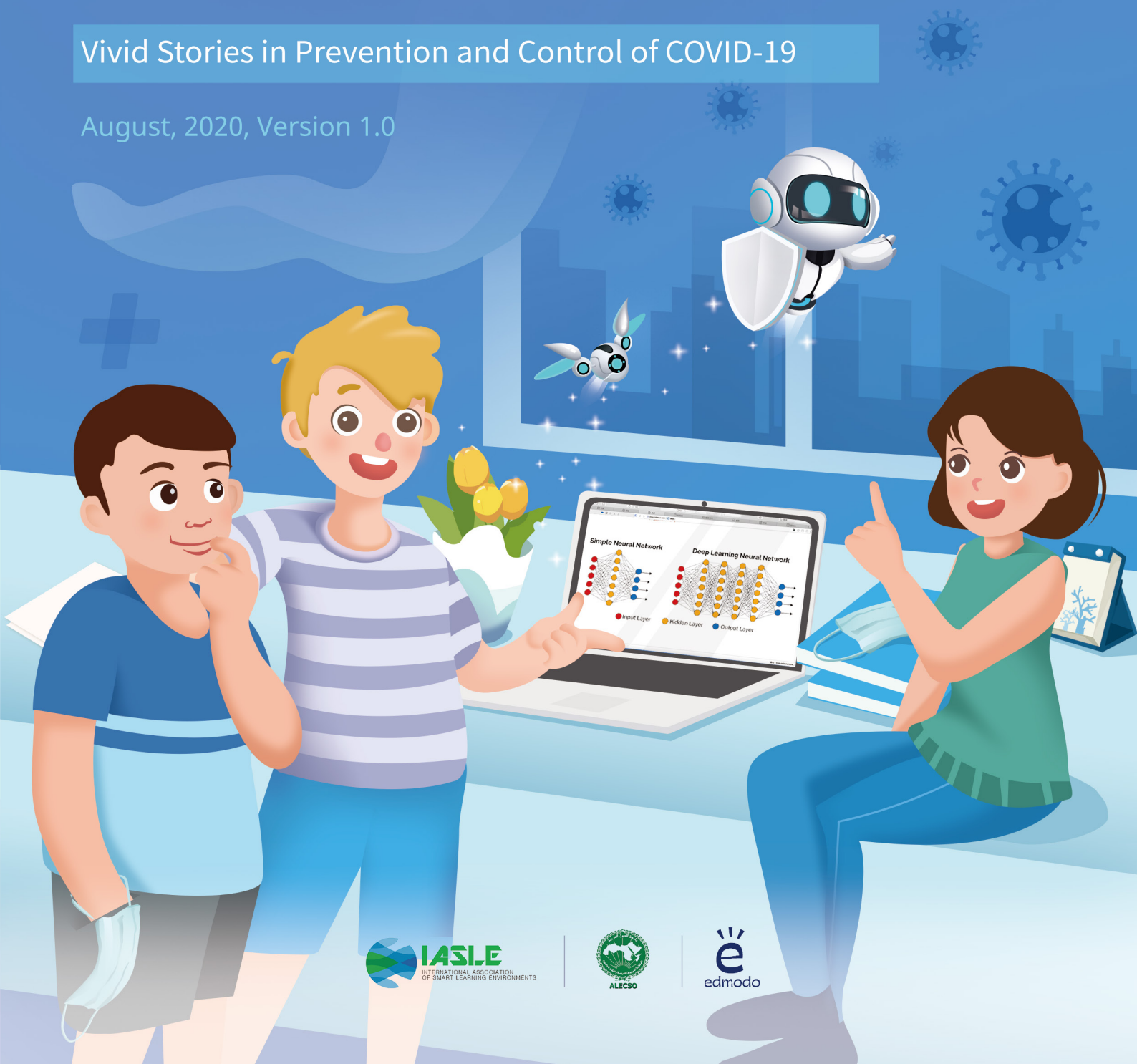


北京师范大学智慧学习研究院
Smart Learning Institute of Beijing Normal University

Interactive Book on Artificial Intelligence to Combat Pandemics

Vivid Stories in Prevention and Control of COVID-19

August, 2020, Version 1.0



Interactive Book on Artificial Intelligence to Combat Pandemics: Vivid Stories in Prevention and Control of COVID-19

© Smart Learning Institute of Beijing Normal University (SLIBNU), 2020

Rights and Permissions



This publication is available in Open Access under the Attribution-ShareAlike 3.0 IGO (CC-BY-SA 3.0 IGO) license (<http://creativecommons.org/licenses/by-sa/3.0/igo/>).

Please cite the work as follows:

Huang, R., Liu, D., Hu, X., Tlili, A., Own, C.M., Fan, L., Shubeck, K., Jemni, M. Zhang, X., Chen, H.Y. (2020). Interactive Book on Artificial Intelligence to Combat Pandemics: Vivid Stories in Prevention and Control of COVID-19. Beijing: Smart Learning Institute of Beijing Normal University.

Interactive Book on Artificial Intelligence to Combat Pandemics: Vivid Stories in Prevention and Control of COVID-19

The interactive and AI-based version of this handbook can be found on the Jupyter environment, where you can interact with the code, modify and download it. Please visit this link: <http://yuanzhuo.bnu.edu.cn/article/653>

Preface

Artificial Intelligence (AI) has been one of the emerging technologies that has been frequently used in our daily life in different domains, including, education, medicine, finance, etc. However, even though AI is frequently used and mentioned in the media, there is still lack of AI understanding. Technological literacy is also very important now, because children are growing up in smart environments where they interact with intelligent devices, like robots and tablets. As children gain exposure and understanding of AI technology, their reasoning about these devices becomes more thoughtful and nuanced. Currently, educators lack information on how to teach teenagers AI, but there is a tremendous need for this to happen rapidly given this pandemic.

Several AI solutions (e.g., using robotics, face recognition, etc.) have been developed to combat several crises, including the current COVID-19 pandemic. It is therefore important to prepare the young generation for the next crisis/pandemic in the era of AI. The more knowledgeable the upcoming generation is of AI technologies, the more likely they are to develop smart solutions that will serve for the well-being of humans in crisis and maintain several vital activities, including economical, educational and health activities.

Based on the above background, this book introduces AI to children, between the age of 9 and 15, in an easy and fun way. It also presents vivid stories about some AI-based solutions that were developed to combat the pandemic. Furthermore, this book presents a hands-on learning experience about AI by showing simple programming code that children could use and implement on the JupyterLab environment. JupyterLab is an open-source web-based interactive development environment that can support a wide range of workflows in data science, scientific computing, and machine learning. The book is structured according to the Five Big Ideas in AI developed by Computer Science Teachers Association (CSTA) for K-12 students, namely perception, representation and reasoning, learning, natural interaction and societal impact. The interactive AI-based version of this book, where readers can interact with the presented content and code (test, modify and download it), can be found on the Jupyter environment: <http://yuanzhuo.bnu.edu.cn/article/653>

Acknowledgement

Many people have helped us in finalizing this book. They have our great appreciation for the long hours and hard work they devoted to conducting research and developing the content. Without their incredible assistance, this book would not have been possible.

We would like to acknowledge the help of several researchers who worked on developing the contents and organizing the webinar for this book, namely Zhong Ren, Ting-Wen Chang, Rongxia Zhuang, Wei Zhou, Junxiu Wang, Bojun Gao, Yihong Shi, Zhonglin Zhao, Jiajia Liu, Jody Cockroft, Zhiqiang Cai and Zhenyu Cai. We would like also to acknowledge the contribution of multiple international partners, researchers, and staff who provided new ideas for this handbook during the organized webinar.

Thanks also go to those experts from the Smart Learning Institute of Beijing Normal University (SLIBNU), UNESCO International Research and Training Center for Rural Education (UNESCO INRULED), International Association of Smart Learning Environments (IASLE), Arab League's Educational, Cultural and Scientific Organization (ALECSO), The University of Memphis and Edmodo for their professional feedback and comments during the preparation of this book.

Reading guidelines of the book

This book introduces AI to children, between the age of 9 and 15, in an easy and fun way. It also presents vivid stories about some AI-based solutions that were developed to combat the pandemic. Furthermore, this book presents a hands-on learning experience about AI by showing simple programming code that children could use and implement on the JupyterLab environment.

Three age tags were used on the book content (chapter, section and story). Each tag denotes the age appropriateness of that content, as follow:

- **Age: 9+** tag highlights the sections that are appropriate for children of the age of nine and above.
- **Age: 11+** tag highlights the sections that are appropriate for children of the age of eleven and above.
- **Age: 13+** tag highlights the sections that are appropriate for children of the age of thirteen and above.

The age appropriateness for each of the book chapters, sections and stories was based on Coh-Metrix software and several AI teachers. Coh-Metrix is a computational tool that produces indices of the linguistic and discourse representations of a text (Graesser et al., 2004; McNamara et al., 2014). The produced output values can be used in many different ways to investigate the difficulty, readability and the coherence of the mental representation of the interactive book texts.

Additionally, a textbox definition was given to each “Bold” term in the text. This textbox aims to further explain that term for children. Furthermore, at the end of each section, an exercise was given in the form of a quiz or a reflection question, to help children practice the gained knowledge from that section.

It should be noted that this book can also be read by other people with different ages who may not have any background about AI. Parents and teachers can also use this book to introduce AI to their children or students respectively, and guide them to interact with the presented code online.

The interactive AI-based version of this book, where readers can interact with the presented content and code (test, modify and download it), can be found on the Jupyter environment: <http://yuanzhuo.bnu.edu.cn/article/653>

Contents

Chapter 1 What is Artificial Intelligence?	1
Chapter 2 Perception: How computers recognize the world like humans do?	4
2.1 Utilizing face recognition for tracking quarantine evaders during COVID-19	5
2.2 Utilizing speech recognition for automatic phone responses during COVID-19	9
2.3 Hands-on learning experience: Face detection	11
Chapter 3 Representation and reasoning: Can computers think and make decisions like humans?	15
3.1 Utilizing medical clinic diagnosis during COVID-19	16
3.2 Augmented Reality consultation during COVID-19	18
3.3 Hands-on learning experience: Reasoning using decision trees	19
3.4 Hands-on learning experience: Visualizing the Covid-19 data	23
Chapter 4 Learning: How do computers learn like humans?	26
4.1 Application of machine learning to predict the spread of COVID-19	29
4.2 Application of machine learning to accelerate healthcare solutions during COVID-19	30
4.3 Hands-on learning experience: training the machine to perform an action	31
Chapter 5 Natural interaction: Can computers understand humans and talk to them?	35
5.1 Smart chatbots to answer users' inquiries about COVID-19	36
5.2 Utilizing natural language processing to analyze and understand the online public's opinions and concerns towards COVID-19	39
5.3 Hands-on learning experience: Sentence tokenization	40
Chapter 6 Social Impact: How AI will impact your life?	44
Conclusion	46
Reference	47
Appendix: Youth Artificial Intelligence Innovation Initiative (YAIII)	48

Chapter 1 What is Artificial Intelligence?

Age: 9+

Artificial intelligence (AI) is not only about robots as shown in movies, for instance “I, Robot” movie (see Figure 1). It is much bigger than that; It has been a research area in computer science and engineering. Now, it is part of our lives. There are many different types of AI. AI is about making machines, such as computers, be smart and mimic human behaviors.

In fact, we are interacting and communicating with ‘machines’ all the time, they are like one of us. The field of AI that made this possible is called speech recognition. AI-enabled computers (we also call them machines, or in some countries, we simply call them electronic brain) can now read and write text like humans. The type of AI that allows this to happen is called Natural Language Processing (NLP). In addition to the ability of language processing, scientists have made computers do other things that are unique to humans. For example, AI can do things like ‘higher order cognitive abilities’, like thinking, reasoning, and decision making. We call ‘higher order cognitive abilities’ such as thinking, reasoning, and decision making. It has been shown that these abilities of the human brain are largely due to its special interaction style between neurons (the basic elements of human brain). Scientists enable computers with these higher-order cognitive capability by implementing something called ‘brain-style computation’ between neural **networks**. To implement these presented examples, AI relies on analyzing **data** using several **algorithms** to achieve its objective. Figure 2 presents examples of how AI technologies could mimic human actions and behaviors.

Artificial (/ ɑr təfɪ əl /)

Something made or produced by people, often as a copy of something natural.

Intelligence (/ ɪntel ɪ dʒəns /)

The ability to learn, think, understand, and make decisions that are based on reasoning.

Network (/ ˈnetwɜrk /)

A large system consisting of many similar parts, such as lines, tubes, nerves, etc. that are connected to each other and operate together.



Figure 1. AI representation in movies

Data (/ˈdeɪ tə, ˈdæt ə, ˈdɑ tə /)

Information, especially facts or numbers, collected to be examined and used to discover things or to make decisions. Data could be in different formats, such as text, picture or audio.

Algorithm (/ˈæl ɡərɪð əm /)

A list of instructions or a set of rules to be followed for solving a problem.

Coronavirus (/ kəˈrʊs nəvaɪ rəs /)

Coronaviruses are a large family of viruses which may cause illness in animals or humans. In humans, several coronaviruses are known to cause respiratory infections ranging from the common cold to more severe diseases such as Middle East Respiratory Syndrome (MERS) and Severe Acute Respiratory Syndrome (SARS). The most recently discovered coronavirus causes coronavirus disease COVID-19.

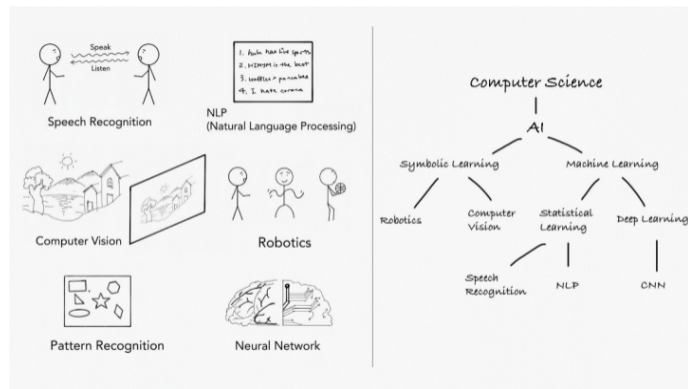


Figure 2. Examples of how AI technologies could mimic human actions

AI, consider omitting term in our language, and AI-enabled devices are playing critical role in our everyday life. They are used almost everywhere, maybe without you knowing about it. For instance, did you know that AI is used in many games that you are playing, to control Non-Player Characters (NPCs) or bots, in games like Fortnite? Did you know that AI is used in our smart-phones when we we talk to Siri? In fact, almost everything you see or do at any moment is enabled by AI to some degree. The book you are reading, yes, this book, has been a product of AI enabled computer programs. AI has been part of the entire life-cycle of every product of modern civilization. Feel free to point to anything you may see at this time, we are sure that AI-enabled technologies, environments, etc. have been part of the production process. These are not limited to the games you play, or smart phones you use.

When AI was first introduced it as to make our jobs more efficient. Now, more and more AI-enabled devices, AI-enabled processes and environments are making our lives easier and enjoyable. In this book, we give examples and facts about AI for humans. AI is actually “saving” human lives during crises and pandemics, during crises and pandemics, like this **COVID-19** pandemic. In this case, AI can be our super heroes that try to protect and keep us safe by being used in different ways. Therefore, to truly appreciate AI, we need to learn about it. It is also possible for you to be a super hero, like Sun Wukong (孙悟空) or Batman, and use AI to save lives in the future. However, as a first step, we need to understand AI together and learn how it can be a super hero. Therefore, this book attempts to explain the basics of AI in an easy and fun way. It also presents vivid stories from different countries about how AI is being used to combat the COVID-19 pandemic.

Let's finish reading this book and get ready to be a super hero in the future!



Story 1: The AI history

Age: 9+

Artificial Intelligence (AI) goes back to a time far, far away from this time, when there were no smart phones, no internet, and no fast cars. Philosophers started studying the human body and its structure, and described human thinking as a **symbolic system**. Ancient Greeks had myths about robots, while Egyptians built some automated systems (systems that work on their own), but back then, the term “Artificial Intelligence” did not exist. In 1952, Arthur Samuel, surprised the world, by making the first computer checkers-playing **program** and the first computer program to learn on its own. This computer was way bigger than computers we know nowadays (see Figure 3). After several years, specifically in 1956, the term “Artificial Intelligence” was finally coined at a conference at Dartmouth College in Hanover, New Hampshire-, United States. Three years later, the Massachusetts Institute of Technology (MIT) AI Lab was built. Later, with the help of newly invented technology, such as the Internet and smart chips, AI started to become more helpful and easier to use. Now, we can see AI with us wherever we go (like hospitals) and in things we use every day (like our smart phones).

Symbolic (/ sɪm'bɒl ɪk /) **System (/ 'sɪs təm /)**

A system that computes and communicates using symbols.

Program (/ 'prəʊ græm, -grəm /)

A set of coded instructions that make a computer perform an operation.



Figure 3. First computer checkers-playing program

Reflection exercise

Data mining is an essential part of AI. You almost can guess its meaning by the word “mining”. If you understand the process of Coal mining or Oil mining, try to come up with your OWN definition of Data mining and compare it to the textbook definition of data mining.

Chapter 2 Perception: How computers recognize the world like humans do?

Age: 9+

Sensor (/ˈsɛn sɔr, -sər /)

A sensor is a device that measures physical input from its environment and converts it into data that can be read by humans or machines. For instance, it is possible to find sensors that detect noises, heart beats, etc.

Perception is one of the most important abilities for human and some animals. It is the ability to help animals (humans included) be aware of physical environments. It has been primarily associated biological species. Different animals are equipped with different perception abilities. Human perception is the ability to see, hear, or become aware of your surroundings through the senses. As shown in Figure 4, while the human body perceives the world with its five senses (sight, sound, smell, taste and touch), computers mimic, or copy, human intelligence and use different sensors (cameras, sound detectors, etc.), similar to the human senses, to perceive the world. Making computers “see” and “hear” like humans is one of the important AI achievements.

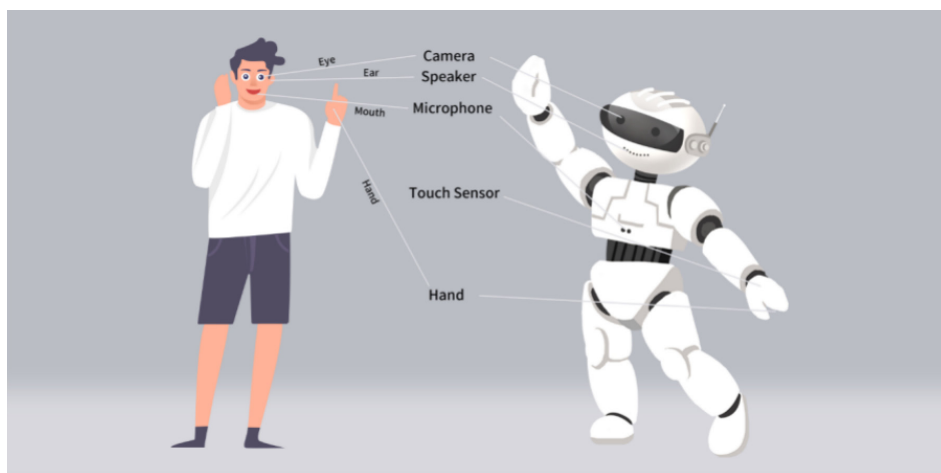


Figure 4. Examples of how humans and computers perceive the world

2.1 Utilizing face recognition for tracking quarantine evaders during COVID-19

Age: 9+

Recognizing faces is one of the first developed human instincts. Small babies need to recognize the faces of their parents from other faces so they can connect to them. In this context, babies start staring and seeing their parents' faces (for instance, during breastfeeding) until they memorize them. They can then start seeing the difference between their parents' faces and unfamiliar faces. Similarly, computers mimic human behavior and intelligence and, instead of using human eyes, they use sensors (cameras) to identify faces. Facial recognition is based on the **geometric characteristics** of the face (the shape of eyes, nose, and mouth) and the geometric relationship between them (the distance between them).

In Figure 5 we can see that facial recognition has four steps.

- **Face Detection:** Locate one or more faces in the image and mark with a bounding box.
- **Face Alignment:** Normalize the face to be consistent with the database, such as geometry and photometrics.
- **Feature Extraction:** Select features from the face that can be used for the recognition task.
- **Facial Recognition:** Compare the face to one or more known faces in a prepared database.

Geometric (/ ˌdʒi əˈmɛ trɪk /)
Characteristics (/ ˌkær ɪk təˈrɪs tɪks /)

They are visual characteristics that made up of shapes such as squares, triangles, or rectangles.

Facial (/ ˈfeɪ ʃəl /)
Recognition (/ ˌrɛk əɡnɪʃ ən /)

Facial recognition is a way of recognizing a human face through technology. A facial recognition system uses biometrics to map facial features from a photograph or video. It compares the information with a database of known faces to find a match.

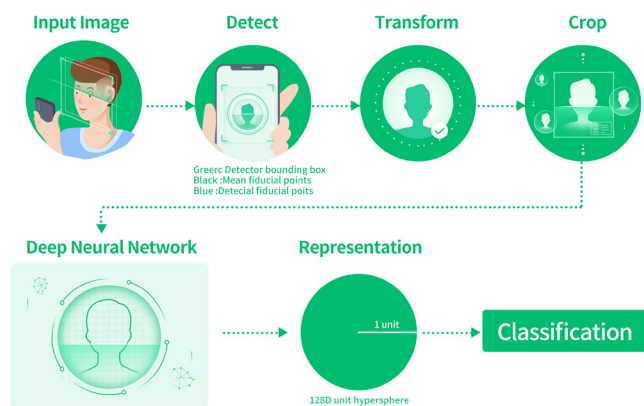


Figure 5. Face detection process



Story 2: Using robots for face mask detection during COVID-19

Age: 9+

Infrared (/ ɪn frəˈred /)

It is a type of light that cannot be seen with the naked eye, but can be felt in the form of heat.



Figure 6. A photo of AIMBOT being in use
Source: by baidu.com

"Your temperature is normal. Please wear a mask", says AIMBOT. AIMBOT is an intelligent patrol robot that reports and reminds people to wear a mask, as shown in Figure 6. When multiple people enter together, it can also conduct multiple face detections and mask identifications. Then it can remind everyone to keep practicing social distancing in order to reduce the risk of infection. How can AIMBOT do this? Even humans cannot measure their body temperature with only their eyes. It uses an **infrared** sensor in addition to its visible light binocular camera (like human eyes) to identify the visitor's mask on their faces and body temperature.



Story 3: Using face recognition for tracking quarantine evaders during COVID-19

Age: 9+

Incubation (/ ɪn kyəˈbeɪ ʃən, ɪn- /) Period (/ ˈpɪər i əd /)

The period of time between harmful bacteria or viruses entering a person's or animal's body, or entering a plant, and the symptoms of a disease appearing.

The COVID-19 virus is very tricky. After infecting people, it has an incubation period of about 14 days in the body. An "incubation period" means the amount of time it takes before a person starts showing symptoms, like a fever, or coughing. An **incubation period** of 14 days means a person can have the virus for two whole weeks before they realize it. You may be worried and asking yourself, "If they do not have any symptoms, how can I tell if I'm surrounded by infected people? How many people are in close contact with them?" Bingo! People's **mobile trajectory data** can tell us! The trajectory data can tell you where the patient has been and whether

Mobile (/ˈmoʊ bəl, -baɪl /)
Trajectory (/ trəˈdʒɛk tə ri /)
Data (/ˈdeɪ tə, ˈdæt ə, ˈdɑ tə /)

Data that show the curved path of somebody or something moving from one place to another.

you've been there. For example, the Beijing JianKang APP uses the camera to detect the person's face. It then shows the current status of that person. As shown in Figure 7, the green word indicates that the health of that person is good, while the yellow word indicates that the person should be home quarantined. Many supermarkets, parks and other public places in China used this APP to allow only those who are healthy to enter these places to protect more people.



Figure 7: Screenshot of JianKang application

The following challenges can make it difficult for computers to recognize a face.

- Pose **variations**: It is sometimes difficult for humans to recognize others, even their friends, when they are looking at them from different angles. Things like head movements, or the camera changing points of view, could lead to major changes in facial appearance. These can make automated fa-

Variation (/ ˌvɛər iˈeɪ ʃən /)

A change in the amount or level of something.

cial recognition a difficult task for computers.

- Facial expression changes: Even more differences could be caused by a person making different facial expressions. A person might change their face because they are in different emotional states (happy, angry, confused, etc.) like the ones in Figure 8. So, efficiently and automatically recognizing the different facial expressions is important for both the evaluation of emotional states and the automated face recognition.

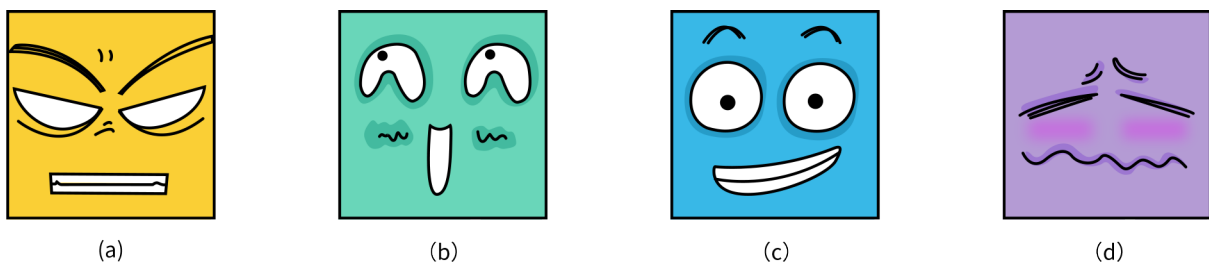


Figure 8. Face expressions based on different emotional states

- Aging of the face: Another reason the face's appearance can change is caused by the ageing of the human face, and could impact on the entire process if the time between each image capture is significant.
- Varying illumination conditions: It is sometimes difficult for humans to identify others when it is a bit dark (for instance at night). Similarly, large variations of light and dark could degrade the performance of recognizing a given face. Indeed, for low levels of lighting of the background or foreground, face detection and recognition are much harder to perform, since shadows could appear on the face and facial patterns could look the same, as shown in Figure 9.

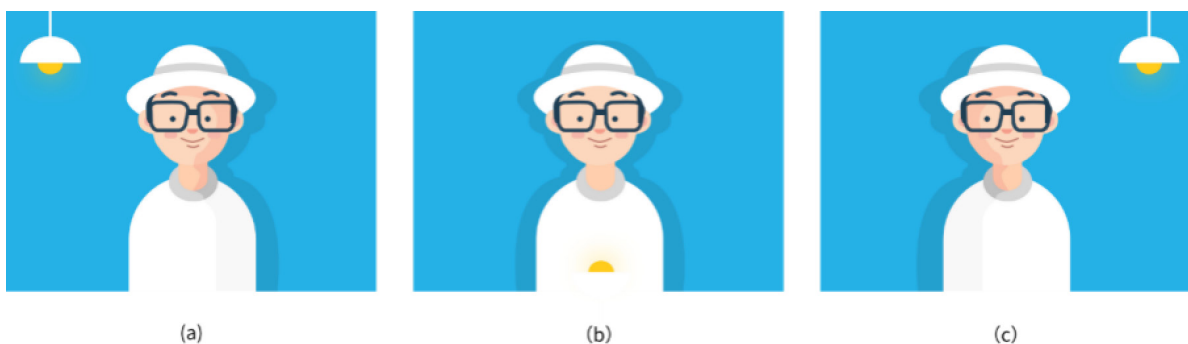


Figure 9. Shadow variation based on the lightning position

2.2 Utilizing speech recognition for automatic phone responses during COVID-19

Age: 11+

Similar to the examples in section 2.1, babies start listening at an early age to their parents' voices and recognize them. They can hear the difference from their voices and strangers' voices, so they might find more comfort from their parents' voices than a stranger's voice. However, unlike humans, computers do not have ears that let them hear sounds, or brains to understand that the sounds are words or phrases. Therefore, they use sensors, like voice detectors (just like the human ears), to detect speech. Specifically, there are three main stages involved in **speech recognition**:

- Preprocessing involves taking the speech sounds and turning it into something the computer can use. Specifically, when we speak, we create vibrations in the air. The computer then uses an **analog-to-digital converter (ADC)** to translate this analog wave into digital data that it can understand. To do this, it samples, or digitizes, the sound by taking precise measurements of the sound wave at frequent intervals (see Figure 10).
- During the recognition stage, the computer must identify what has been said. In this context, it starts analyzing the data also known as waves (obtained from the first step) and comparing it with already stored data of other words to identify what the user said.
- During the communication stage, the computer acts upon the translated input.

Speech (/ spɪtʃ /) Recognition (/ ˈrɛk əɡnɪʃ ən /)

Speech recognition develops methodologies and technologies that enable the recognition and translation of spoken language into text by computers. It allows, for instance, hands-free control of various devices and equipment and automatic voice translation.

ADC: Analog-to-Digital Converter (/ kənˈvɜː tər /)

An electronic integrated circuit that translate analog signals into digital values for use in processing and control systems, for example, when using the voice recorder of our mobile phones, the ADC will translate our voices as analog signals to digital data that can be recognized, saved and processed by our mobile phones.

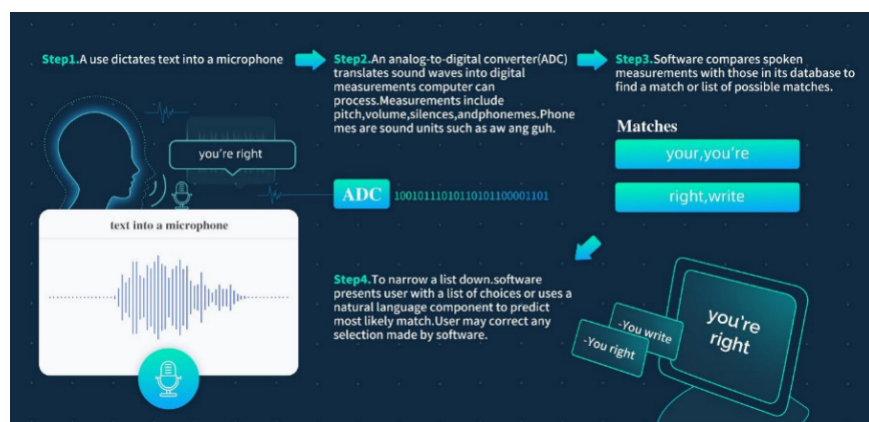


Figure 10: Speech recognition process



Story 4: AI Voice Responders Help Call Centers During COVID-19

Age: 9+

An AI Voice Responder, as shown in Figure 11, provides a fast and cost-effective way for organizations to better manage various call center needs and maintain a high level of customer service during the pandemic. The AI-enabled solution works by quickly answering the customer's call without making the customer wait. It can also automatically classify calls and resolve common problems without any manual intervention, and classify calls based on subject and urgency, so companies can prioritize follow-up issues to solve more complex issues.

As the measures to deal with the new coronavirus pandemic have led to the temporary closure of some mission-critical call centers around the world, consumers are facing long waits as companies struggle to cope with the surge in call volume. This is where the artificial intelligence voice reactor comes to the rescue and diverts high call volumes and answers common questions, quickly solving common support problems without a live agent.

For example, the artificial intelligence outbound services of Baidu and iFlytek, as leading providers of artificial intelligence services, can benefit as call centers that are under tremendous pressure due to the novel coronavirus pandemic. At present, the company's customer service dialogue AI platform can immediately solve problems over the phone, improve customer experiences and reduce costs.

Unlike previous generations of IVRs (Interactive Voice Responses), voice responders can understand natural language and enable callers to talk as if they were talking to a living agent. The IVR system accepts mixed voice phone input and key-key keyboard selection, and provides related responses in the form of voice, fax, call back, email and other contact methods. On the other hand, artificial intelligence voice responders are used to help users solve basic hardware problems of commercial products and replace the affected offshore call centers. In addition, voice AI solutions ask questions to identify problems, provide relevant solutions through text, and follow up if the problem is not resolved.

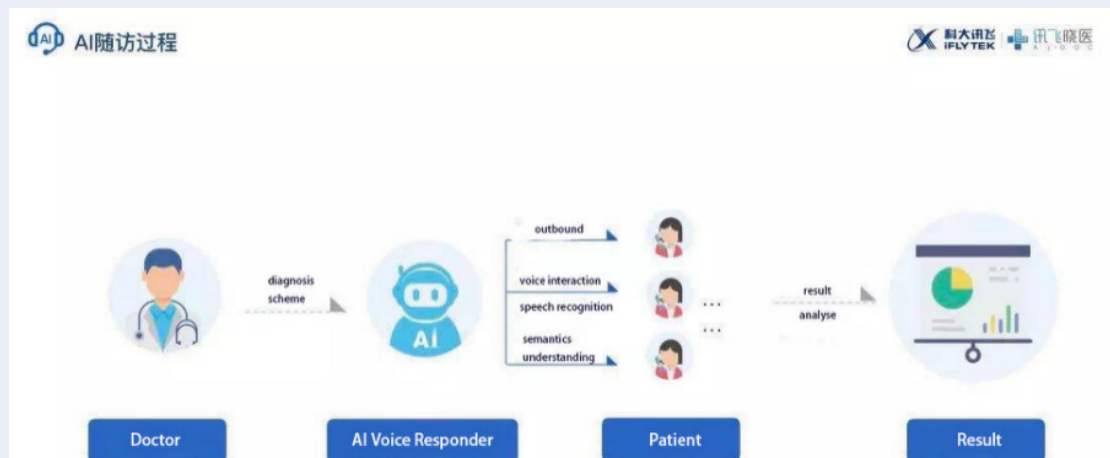


Figure 11. AI voice responder process



Story 5: Using psychological consultant robot during COVID-19

Age: 9+

Two middle school students Thomas and Lee are talking the news released by Carnegie Mellon University (See Figure 12). “Did you hear that Carnegie Mellon University is working on COVID-19 detection in human voice?” “Ah! Isn't that possible?” “Don't put limits on the technology. Imagine that if it works, it's certainly better to reduce the risk of infection when people go out to get tested.” “Yeah. Looking forward to it”. It determines the probability of a user being infected by analyzing his/her voice and comparing it with the voice characteristics of people who are infected with the COVID-19. Sure, until now this application is not a diagnostic system” and has not been approved by the FDA or CDC and therefore it should not be used as a substitute for medical examination by a health professional.

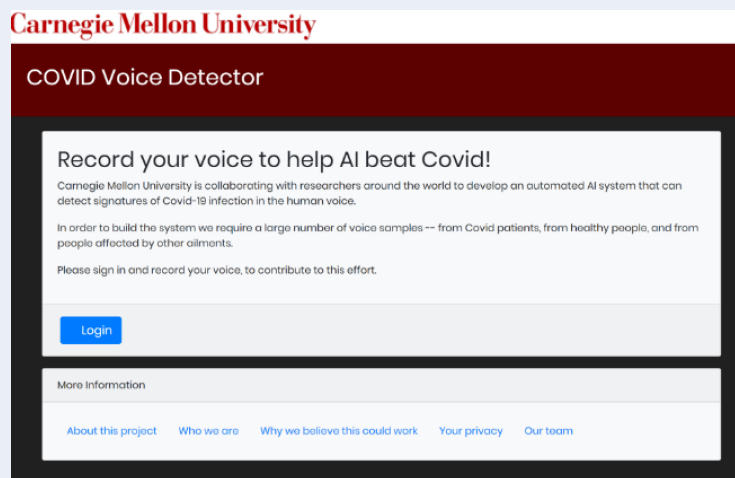


Figure 12. Graphic of Carnegie Mellon Voice detector

2.3 Hands-on learning experience: Face detection

Age: 9+

This section will use a widely popular face detection and manipulation library face-recognition by Adam Geitgey (). The images are customized with different colors and border shapes around the detected faces. First, import the useful libraries to our python code.

```
1 import PIL.Image
2 import PIL.ImageDraw
3 import face_recognition
```

Please select the image name SchoolKids.jpg (you can pick up any image) in the same folder as the code file or else give the proper path of the image. The selected image would load by the load_image_file() method from the face_recognition library which will convert it into a NumPy array of that image. We can then assign it to the variable name given_image. Then assign it to the variable name given_image.

```
1 given_image = face_recognition.load_image_file('SchoolKids.jpg')
```

Using face_locations() method from the same library, we will count the number of faces in the given_image and will print the length of total faces found in the image.

```
1 face_locations = face_recognition.face_locations(given_image)
2 number_of_faces = len(face_locations)
3 print("We found {} face(s) in this
    image.".format(number_of_faces))
```

Accordingly, to draw any shape on the image, we will convert the image to the Pillow library object using fromarray() method from PIL.Image.

```
1 pil_image = PIL.Image.fromarray(given_image)
```

Then, we will run a for-in loop to print four pixel locations such as top, left, bottom and right of the detected faces.

```
1 for face_location in face_locations:
2     top, left, bottom, right = face_location
3     print("A face is detected at pixel location Top: {}, Left: {},
    Bottom: {}, --Right:{}".format(top, left, bottom, right))
```

We will draw a green color rectangle box with width 10 around the faces.

```
1 draw = PIL.ImageDraw.Draw(pil_image)
2 draw.rectangle([left, top, right, bottom],outline="green",
    width=10)
```

Now, just use the variable `pil_image` to display our new image with detected faces using a rectangular border around them.

```
1 pil_image.show()
```

The new temporary image will be opened automatically on your computer (if everything is set up correctly). It should be something like Figure 13.



Figure 13. Example of face detection within a given picture

If you run the above file, and everything goes fine, you will get the below output at the console of your editor:

We found 5 face(s) in this image.

A face is detected at pixel location Top: 1454, Left: 3451, Bottom: 1775, Right: 3130

A face is detected at pixel location Top: 1784, Left: 1904, Bottom: 1939, Right: 1749

A face is detected at pixel location Top: 1818, Left: 2351, Bottom: 1973, Right: 2196

A face is detected at pixel location Top: 1830, Left: 1529, Bottom: 1959, Right: 1400

A face is detected at pixel location Top: 1878, Left: 2445, Bottom: 1967, Right: 2355



** Please test the code and practice it on our jupyter server **

<https://code.yuanzhuo.bnu.edu.cn>

Quiz: (Correct answers are: B and C)

Please select the incorrect statement:

- A. Sensors are able to detect noise, heart beats.
- B. AI enables speech recognition to be realized without converting voice input to something computer can use.
- C. Facial recognition and speech recognition are realized by AI.
- D. Speech recognition includes the recognition and translation of spoken language.

What does facial recognition do?

- A. Help doctors identify unknown symptoms of individuals.
- B. Match individuals' information without having a known database
- C. Identify individuals by their facial characteristics.
- D. Automatically operate plastic surgeries.

Chapter 3 Representation and reasoning: Can computers think and make decisions like humans?

Age: 11+

We, humans, learn from our experiences so we can start making decisions. For instance, to learn how to ride a bike, we start learning how to use the bicycle pedals, have balance and use the brakes when needed. After several tries, we can decide when to pedal fast and when to coast, when to apply the brakes and when not. With the advances of technology, researchers are now trying to make computers think, do reasoning and make decisions like humans. Knowledge representation and reasoning is the field of AI aimed to make computers intelligent by representing information about the world in a form that they can use to make reasoning and decisions like humans. For instance, we can give computers the location of several places (e.g., restaurants, shops, etc.) as a data point and then computers can do reasoning to select the shortest path from the current location to the location that the user wants to go to.

Three methods of reasoning are the deductive, inductive, and abductive approaches. Deductive reasoning starts with the assertion of general rule and proceeds from there to a guaranteed specific conclusion. Deductive reasoning moves from the general rule to the specific application: In deductive reasoning, if the original assertions are true, then the conclusion must also be true. For example, math is deductive:

If $x = 2$

And if $y = 5$

Then $3x + y = 11$

Inductive reasoning begins with observations that are specific and limited in scope, and proceeds to a generalized conclusion that is likely, but not certain, in light of accumulated evidence. You could say that inductive reasoning moves from the specific to the general. For example, Paul always leaves his home to go to school at 7:00 AM, and he is always on time. Paul then assumes that if he leaves at 7:00 AM for school today, he will be on time.

Abductive reasoning typically begins with an incomplete set of observations and proceeds to the likeliest possible explanation for the set. Abductive reasoning yields the kind of daily decision-making that does its best with the information at hand, which often is incomplete. For example:

Premise 1: You arrived home and were surprised that the front door was open (X)

Premise 2: However, if your son had arrived home before you, this would be unsurprising (If Y, then unsurprisingly X)

Premise 3: Therefore, it is reasonable to conclude that your son opened the door. (therefore, presumably Z).

3.1 Utilizing medical clinic diagnosis during COVID-19

Age: 11+

In the past, doctors needed long-term training to conduct professional diagnosis. Nowadays, researchers are making computers smart by making them do reasoning and make decisions, especially for medical diagnosis and treatment during COVID-19. Now, through the latest artificial intelligence identification system, we can diagnose the CT image of patients, and analyze the human cells infected by pathogens, so that in hospitals where medical experts and equipment are insufficient, we can also diagnose the epidemic situation in an emergency. First of all, AI is used to perform basic screening and judgment, and then doctors can do control and correction, to guarantee the safety of the patients and improve the time to treatment for the patients.



Story 6: Intelligent chest evaluation system of COVID-19

Age: 11+

The new chest computed tomography (CT) coronavirus pneumonia (see Figure 14) is an intelligent evaluation system. It can quickly screen suspected patients and assist medical institutions to quickly shunt patients. It includes the following features:

- Intelligent detection: the traditional method of detecting lesion areas requires doctors to manually outline the ROI, which is inefficient and difficult to promote. Using artificial intelligence can achieve rapid automatic detection of lesions, greatly improving efficiency.
- Intelligent analysis: The AI system enables dynamic 4D comparison of the whole lung lesion in CT, and quantitative analysis of the severity of pneumonia in 2-3 seconds.
- Intelligent follow-up: intelligent follow-up analysis of the patient's disease course, accurate matching of historical images, automatic analysis of disease metastasis and development.



Figure 14. CT coronavirus pneumonia



Story 7: AI-Based triage and monitoring system predicts spread of coronavirus

Age: 11+

Perhaps you're wondering if you could get a doctor's diagnosis and treatment recommendations at home to reduce the hospitals' overcrowding? Believe it or not, it is now possible, thanks to AI. Technology startup Diagnostic Robotics (Tel Aviv, Israel) has developed a triage and monitoring system (see Figure 15) to help healthcare providers, payers and government agencies in the fight against coronavirus (COVID-19). What kind of services can they provide? The solution includes remote patient progress monitoring, automated patient queries, provider-facing alerts about high-risk patients, and daily updates about the spread and progress of the disease at a community and regional level. Definitely, more improvement needs to be done to make it more efficient.

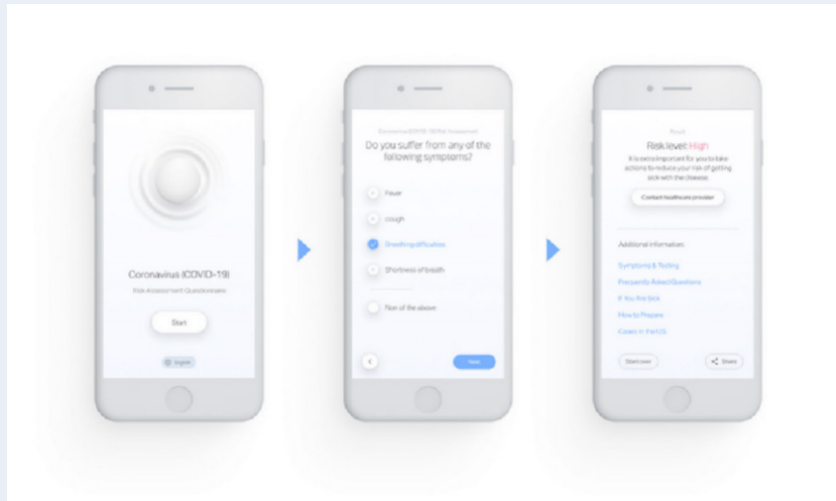


Figure 15. A monitoring system of patients during COVID-19



Story 8: AI tool to predict COVID-19 patients

Age: 11+

"The Patient in room number 3 is showing infection signs, available doctors, please come to this room." Such announcements are often heard in hospitals, especially during pandemics. Is it possible to predict if a patient will require additional interventions before they show symptoms? Can we use AI to carry out early prevention in order to save more lives? Some hospitals have begun to experiment with AI technologies, where these technologies learn from existing data to find similar patterns and then use those patterns to make predictions about the future. For instance, new algorithms are now being developed to determine which mildly ill patients are likely to become severely ill. In doing so, these algorithms also found some unexpected early clinical signs that predict severe cases of COVID-19.

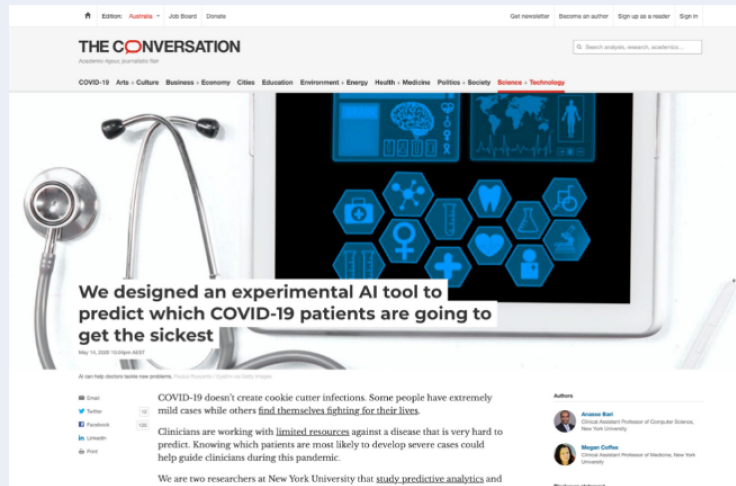


Figure 16. AI tool to predict COVID-19 infection

3.2 Augmented Reality consultation during COVID-19

Age: 11+

Augmented (/ ɔːɡmɛntɪd /) Reality (/ rɪəlɪti /)

Augmented Reality (AR) is computer-generated content overlaid on a real-world environment. AR hardware comes in many forms, including devices that you can carry, such as handheld displays, and devices you wear, such as headsets, and glasses.

Augmented reality and artificial intelligence are distinct technologies, but they can be used together to create unique experiences. In augmented reality, a 3D representation of the world must be constructed to allow digital objects to exist alongside physical ones. Visual data is used along with an accelerometer and gyroscopes to build a map of the world and track movement within it. Most of these tasks are still done using traditional computer vision techniques that make no use of machine learning.

Independently, however, AI models have gotten incredibly good at doing many of the things required to build immersive AR experiences. Deep neural networks can detect vertical and horizontal planes, estimate depth and segment images for realistic occlusion, and even infer 3D positions of objects in real-time. Because of these abilities, AI models are replacing some of the more traditional computer vision approaches underpinning AR experiences.



Story 9: How AI-Powered Augmented Reality Helps during Pandemics

Age: 11+

Do you know AR or **VR**? Have you experienced AR or VR? What do you think of these emerging technologies especially during this pandemic? AR and VR technologies are emerging as effective solutions in this time because they enable people to connect with others virtually. The pandemic has also brought opportunities for telemedicine or telehealth platforms that can offer both patients and care providers a different experience while delivering treatment. These technologies provide engagement for the users, patients and doctors. JD Health, a subsidiary of e-commerce giant JD.com, saw a tenfold growth in monthly consultations for its online health platform during the coronavirus epidemic. In addition, a provider of extended reality and therapeutic applications XRHealth is developing VR telehealth support in its response to prevent the spread of the virus. So, as businesses are suffering due to the spread of COVID-19, such immersive technologies are emerging as their rescue solutions helping them to sustain and improve their business profitability not only during the pandemic, but also post-crisis era. It is highly recommended that you try out these emerging technologies.

Virtual (/ 'vʌr tʃu əl /) Reality (/ rɪæl ɪ ti /)

Virtual Reality (VR) is the use of computer technology to create a simulated environment that is similar or different from the real world. Unlike traditional user interfaces where users see environments on the screen, users will use Head-mounted display (VR glasses) to be within the VR environment and interact with its components.

3.3 Hands-on learning experience: Reasoning using decision trees

Age: 11+

Herein, we provided the basic solution to process the data from data web site. The method is to make the decision by using a **decision tree**. What is a decision tree?

One of the reasons why decision trees are so powerful is that they can be easily visualized so that a human can understand what is going on. Imagine a flowchart, where each level is a question with a yes or no answer. Eventually an answer will give you a solution to the initial problem. That is a decision tree. Everybody subconsciously uses decision trees all the time for most menial tasks. Decision trees in machine learning take that ability and

Decision (/ dɪ'sɪʒ ən /) Tree (/ tri /)

Decision trees are one of the most popular and powerful machine learning algorithms. It is used to determine a set of actions or show a statistical probability. Each branch of the decision tree represents a possible decision, outcome, or reaction. The farthest branches on the tree represent the end results.

multiply it to be able to artificially perform complex decision-making tasks.

The decision tree analyzes a dataset in order to construct a set of rules, or questions, which are used to predict a class. Let us consider a dataset consisting of lots of different animals and some of their characteristics. These characteristics can be used to predict their class. If we take a falcon and a tiger, a question that would split these two animals would be 'Does this animal have feathers?' or perhaps 'Does this animal lay eggs?'. The answer no for either of these questions would lead to the classification of a tiger, whereas yes would be a falcon. These rules can be built up to create a model that can classify complex situations. To extend the animal classification example, consider the scenario of needing to classify a selection of animals into mammals, reptile, or insects. Look at the visualized decision tree (see Figure 17) to see how two simple questions can be used to split the data. These simple questions layered one after another, allow the classification of a wide range of animals. This is the power of decision trees. Now if we give the trained decision tree a new animal, for example a snake, it will classify it. Does a snake have vertebra? Yes. Is a snake cold-blooded? Yes. Therefore, the model will classify it as a reptile, the correct answer!

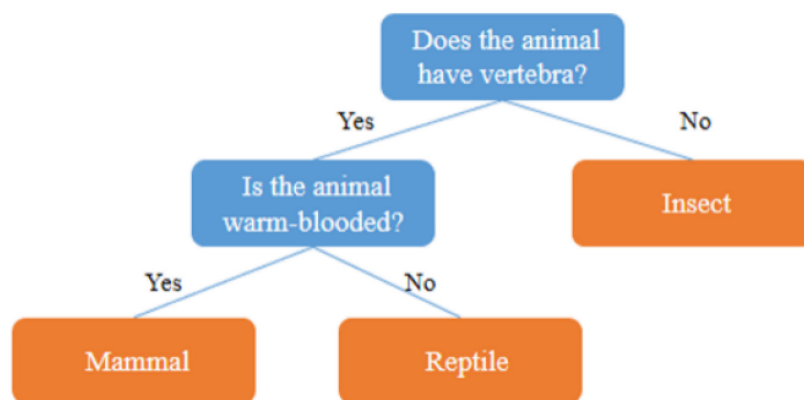


Figure 17. Decision tree to classify animals into mammals, reptile, or insects

When a human constructs a decision tree, the questions and answers are based on their logic and knowledge. In data science the creation of these rules is usually governed by an algorithm learning which questions to ask by analyzing the entire dataset. To put this into context we will come back to the animal example, the algorithm will look at all the animals to figure out that all animals that do not breathe air are all fish. This mathematically splits the dataset by its class. This creates powerful algorithms that can classify new data into classes in a way that any human can understand. Decision trees can become much more powerful when used as ensembles. Ensembles are clever ways of combining decision trees to create a more powerful model. These ensembles create state of the art machine learning algorithms that can outperform neural networks in some cases. The two most popular ensemble techniques are random forests and gradient boosting.

```

1 from sklearn.datasets import load_iris
2 from sklearn.tree import DecisionTreeClassifier
3 from sklearn.model_selection import train_test_split
4 from sklearn.metrics import confusion_matrix
5 from sklearn.tree import export_graphviz
6 from sklearn.externals.six import StringIO
7 from IPython.display import Image
8 from pydot import graph_from_dot_data
9 import pandas as pd
10 import numpy as np

```

Accordingly, let's take a look at how we could go about implementing a decision tree classifier in Python. For example, we'll be working with what has to be the most popular dataset in the field of machine learning, the iris dataset from UC Irvine Machine Learning Repository. The following source codes are parts of full version. Besides, we used the following library as the handy tool.

```

1 iris = load_iris()
2 X = pd.DataFrame(iris.data, columns=iris.feature_names)
3 y = pd.Categorical.from_codes(iris.target, iris.target_names)

```

In the proceeding section, we will attempt to build a decision tree classifier to determine the kind of flower given its dimensions.

```

1 X.head()

```

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
0	5.1	3.5	1.4	0.2
1	4.9	3.0	1.4	0.2
2	4.7	3.2	1.3	0.2
3	4.6	3.1	1.5	0.2
4	5.0	3.6	1.4	0.2

Although, decision trees can handle categorical data, we still encode the targets in terms of digits (i.e. setosa=0, versicolor=1, virginica=2) in order to create a confusion matrix at a later point. Fortunately, the pandas library provides a method for this very purpose.

```
1 y = pd.get_dummies(y)
```

We'll want to evaluate the performance of our model. Therefore, we set a quarter of the data aside for testing.

```
1 X_train, X_test, y_train, y_test = train_test_split(X, y,  
    random_state=1)
```

Next, we create and train an instance of the `DecisionTreeClassifier` class. We provide the `y` values because our model using a supervised machine learning algorithm.

```
1 dt = DecisionTreeClassifier()  
2 dt.fit(X_train, y_train)
```

We can view the actual decision tree produced by our model by running the following block of code.

```
1 dot_data = StringIO()  
2 export_graphviz(dt, out_file=dot_data,  
    feature_names=iris.feature_names) (graph, ) =  
    graph_from_dot_data(dot_data.getvalue())  
    Image(graph.create_png())
```

Notice how it provides the Gini impurity, the total number of samples, the classification criteria and the number of samples on the left/right sides.

Tip

Gini Impurity is a measurement of the likelihood of an incorrect classification of a new instance of a random variable, if that new instance were randomly classified according to the distribution of class labels from the data set. If you have interesting, please refer the following address, <https://bambielli.com/til/2017-10-29-gini-impurity/>

3.4 Hands-on learning experience: Visualizing the Covid-19 data

Age: 11+

Due to the characteristics of Python, many Internet masters have developed library functions to support various types of applications. In practicing machine learning algorithms, most of the data sources are available via the web, and this time the epidemic data is the same. Thus, the process of reading data by Python can be demonstrated simply as follows,

```
1 # pd.read_csv is the function to save the csv format file from web url.
2 pd.read_csv(' https://github.com/datasets/covid-
  19/blob/master/data/countries-aggregated.csv ', parse_dates=['Date'])
3 countries = ['Canada', 'Germany', 'United Kingdom', 'US', 'France', 'China']
```

where pd is the provided library, and the command function is read.csv (URL), i.e., go directly to the URL to capture the web archive (a comma-separated text file), the field name is date, and the countries classified are Canada, Germany, United Kingdom, United States, France and China.

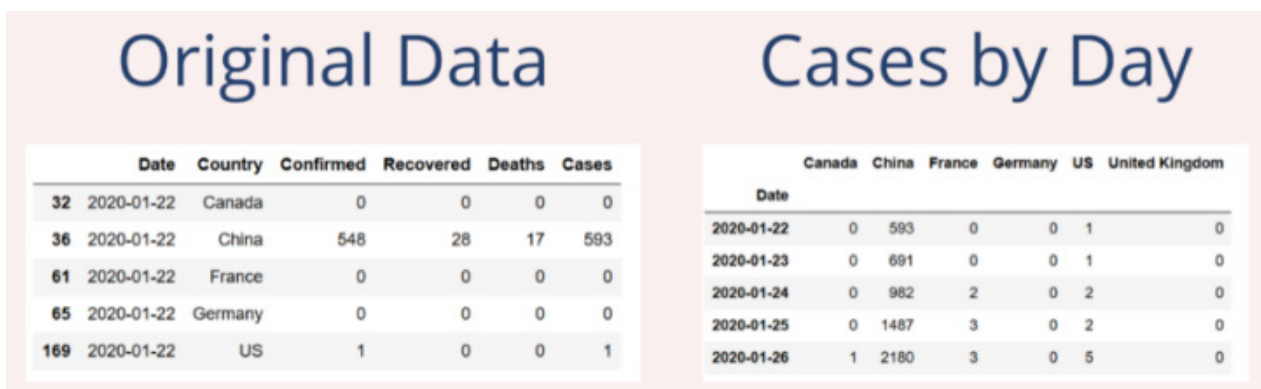


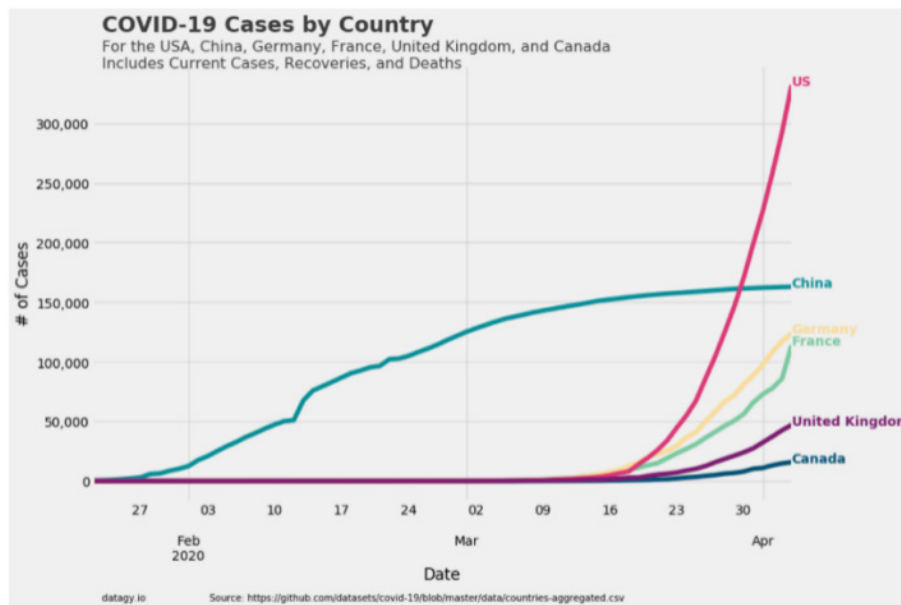
Figure 18. The left side is the raw data, the right side is the formatted data after reading, sorted by Date

```

1 # The definition of graphic colors and styles to present country data in
  different colors
2 colors = {'Canada':'#045275', 'China':'#089099', 'France':'#7CCBA2',
  'Germany':'#FCDE9C', 'US':'#DC3977', 'United Kingdom':'#7C1D6F'}
3 plt.style.use('fivethirtyeight')
4 # Create Visual Appearance, including graphic size, arrangement
5 plot = covid.plot(figsize=(12,8), color=list(colors.values()), linewidth=5,
  legend=False)

```

According to the description on web page, the Python library provided by the other party can only takes 5 minutes of program training, and the results are shown in Figure 19. The whole process is easy and clear. There are less than 20 lines of code, and less than three lines of codes are used for the data operation, and the rests are handling on the picture design.



Our First Visualization — Cases over Time by Country. Source: Nik Piepenbreier

Figure 19. Visualization of the data



** Please test the code and practice it on our jupyter server **

<https://code.yuanzhuo.bnu.edu.cn>

Quiz: (Correct answers are: A and A)

The approach of reasoning is _____ .

- A. inductive, deductive, and abductive
- B. deductive and abductive
- C. inductive and abductive
- D. inductive and deductive

How can AI be applied to medical clinic diagnosis during the COVID-19 pandemic?

- A. AR and VR technologies provide engagement for the users, patients and doctors during the treatments.
- B. AI can automatically detect and cure lesion areas of lungs.
- C. Develop algorithms to predict which healthy individual is likely to be infected based on their iris color.
- D. Generate optimal treatment for patients by solely collecting their personal history of diseases of respiratory system.

Chapter 4 Learning: How do computers learn like humans?

Age: 13+

While humans learn from their past experiences, computers can learn from data rather than through explicit programming. This is known as machine learning. Machine learning uses a variety of algorithms that iteratively learn from data to improve, describe data, and predict outcomes. For instance, as shown in Figure 20, in an online shopping website, the machine will learn the preferences or choices of a particular customer by recording the products that he/she has bought or went through. It will then start recommending to the customer based on these preferences.



Figure 20. A computer learning about a shopper from his data

In machine learning, four types of learning could be implemented:

1. Supervised learning: In this learning type, the machine learning model is taught some knowledge in advance so it can predict future instances. To simplify this definition, we want the machine to know if the student X is a hardworking, good or lazy student based on: (1) how many learning hours he/she spends per day; (2) how many hours he/she watches the TV. As we mentioned, we need to first teach the machine; in this case we have to give it previous knowledge (also known as labeled dataset) about other students, as shown in Table 1. As you can see in the knowledge given in Table 1, the type of student is known (hard working, good, lazy). The machine will first learn from this knowledge. It will then classify any new student (for instance, our student X) based on how many hours, he/she spent learning and watching TV.

Table 1. Example of knowledge information (also known as labeled dataset) to classify a student

Student	Learning hours	TV watching hours	Type of student
Adam	5	1	Hard working
Sarra	3	2	Good student
Charlie	2	3	Lazy student

2. Unsupervised learning: In this learning type, we do not teach the machine learning model (like in the first case). Hence, the machine will work on unlabeled data and it will classify or cluster this data based on similar patterns or features.
3. Reinforcement learning: In this learning type, an agent will be within an interactive environment and in a specific state. It will then learn from its own actions and experiences by trial and error using feedback from the environment. To simplify this, we take the example of “fetch the ball” game between the dog and its owner (See Figure 21). The dog (agent) will be in the garden with its owner (interactive environment) where the owner throws the ball (state). The dog (agent) has to perform an action which could be running after the ball or not. If the dog fetches the ball (action), its owner (interactive environment) will reward him by giving him food, in this way the dog (agent) will know that he is doing the right action and he should keep doing that. If the dog does not bring the ball (action), its owner will not give him food. In this way, the dog (agent) will know that it is doing the wrong action, and it should go bring the ball (correct action)

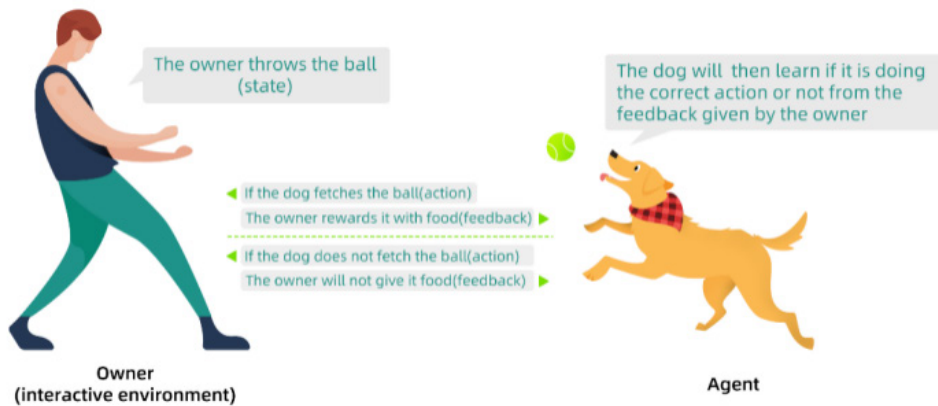


Figure 21. “Fetch the ball” game between the dog and its owner

4. Deep learning: Deep learning is a specific method of machine learning that incorporates neural networks in successive layers in order to learn from data in an iterative manner. Deep learning is especially useful when you are trying to learn patterns from unstructured data.

Just like we have seen above, machine learning can learn and make predictions based on the given data. Therefore, it is important to give the machine: (1) good quality data: which means accurate and correct data that can help the machine takes correct decisions; and, (2) big amount of data which can help the machine learns several information that can help it perform several actions and decisions. To simplify this, we can consider the machine as the “car” and the data as the “fuel” (See Figure 22). So, the car cannot work without the fuel, same for the machine, it cannot work without the data. Also, if we give too much good quality fuel to the car, the engine then will not be harmed and the car can take us on a long trip. Same for the machine, accurate data, will not harm the machine and make it take correct decisions, and too much data will make it take several decisions or predictions.

**Computer (/ kəmˈpyu tər /) data
(/ ˈdeɪ tə, ˈdæt ə, ˈda tə /)**

Computer data is information processed or stored by a computer. This information can be in different forms, such as text, audio, or image. Computer data is processed by the computer's Central Processing Unit (CPU) and can be stored on the computer's hard disk or online, on the cloud.

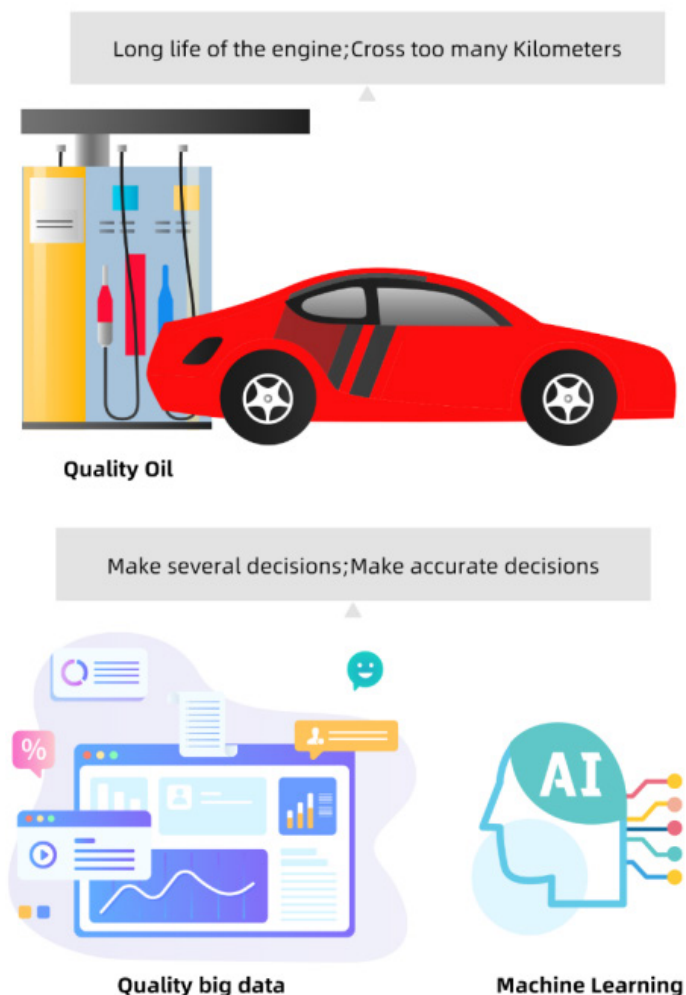


Figure 22. Importance of data for machine learning

There is an urgent need for hospitals and relevant departments at all levels to establish interactive information interfaces and connections, as well as a platform for all parties to collaborate and share information in real time. This way, information and data can be shared among medical institutions and between medical institutions and regional administrative departments in accordance with their mandates. This can provide an effective basis for management and decision-making. In this context, the most important thing is to provide a unified and structured data platform, so that doctors and researchers can easily access international and updated data. For instance, as shown in Figure 23, COVID-19 surveillance platform was established by Ningbo medical health committee, China in the early 2020, and it could be used to monitor the 4000 medical institutions in Ningbo. All of the data, such confirmed cases, age of infected patients, etc. come from the Yindu Cloud in China.



Figure 23. COVID-19 surveillance platform, established by Ningbo medical health committee

4.1 Application of machine learning to predict the spread of COVID-19

Age: 13+



Story 10: AI tool to predict COVID-19 patients

Age: 13+

“When will school reopen? What is the evolution of COVID-19? How long will COVID -19 last? How can we know the trend of this pandemic?” Many people are now asking these questions. To answer all these questions and predict how the virus will spread, Tuli et al. (2020) built a machine learning model to make a good advanced prediction of the number of new cases and the dates when the pandemic might end. For example, the model shown in Figure 24 shows curves corresponding to both new cases and deaths across the world.

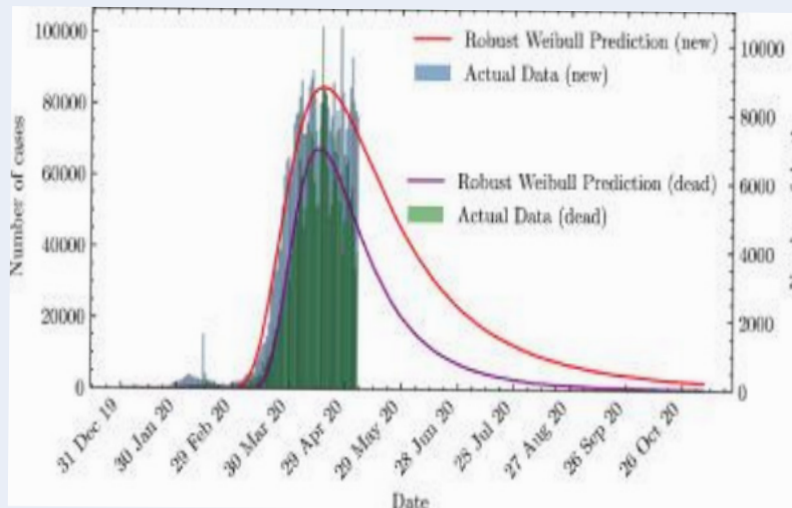


Figure 24. Prediction model about new cases and deaths across the world during COVID-19

4.2 Application of machine learning to accelerate healthcare solutions during COVID-19

Age: 13+

During pandemics, AI can be the best friend of doctors, where computers can be used to speed-up finding medical solutions or vaccines to some diseases. For instance, by using machine learning techniques, it is possible to do a quick screening of billions of chemical compounds quickly to find relevant medicine candidates.



Story 11: Application of machine learning to help accelerate the discovery of medicine for treating COVID-19

Age: 13+

Scientists can now apply machine learning to accelerate the process of finding medicine for COVID-19. For example, Oxford-based Exscientia is now using machine learning techniques to accelerate the process of finding effective medicine. Specifically, AI can learn the structure of DNA and the virus efficiently. It can then quickly analyze big data related to medicine information in order to identify those medicines which have the potential of treating COVID-19.

4.3 Hands-on learning experience: training the machine to perform an action

Age: 13+

Convolution Neural Network (CNNs) are largely applied in the domain of computer vision and have been highly successful in achieving state of the art performance on various test cases. The hidden layers in a CNN are generally convolution and pooling (downsampling) layers. In each convolution layer, we take a filter of a small size and move that filter across the image and perform convolution operations. Convolution operations are nothing but element-wise matrix multiplication between the filter values and the pixels in the image and the resultant values are summed.

CNN is also computationally efficient. It uses special convolution and pooling operations and performs parameter sharing. This

enables CNN models to run on any device, making them universally attractive. All of this sounds like pure magic. We are dealing with a very powerful and efficient model which performs automatic feature extraction to achieve superhuman accuracy. All CNN models follow a similar architecture, as shown in Figure 25.

CNN: Convolution (/ˌkɒn vəluːʃən/) Neural (/ˈnjuər əl/) Network (/ˈnetwɜːrk/)

CNN is a convolutional neural network is a feed-forward neural network that is generally used to analyze visual images by processing data with grid-like topology. It's also known as a ConvNet. Generally, a convolutional neural network is used to detect and classify objects in an image.

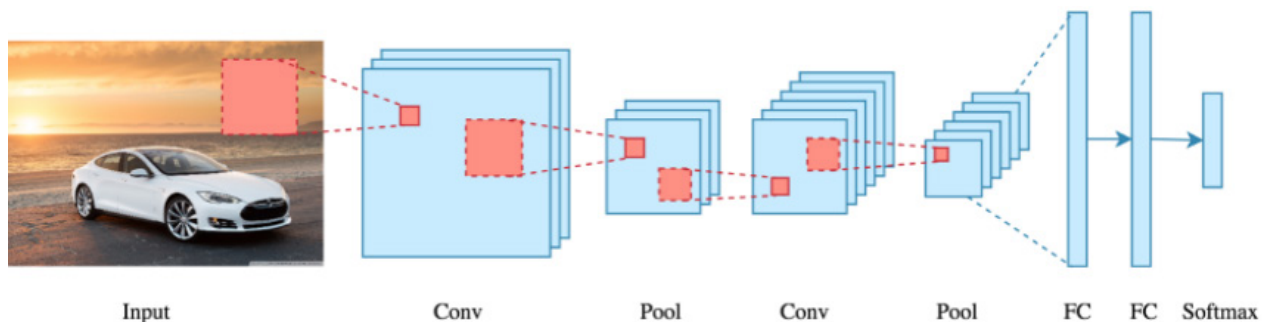


Figure 25. CNN architecture

Now, let's move on to pooling layers. Pooling layers are used to downsize the image. The image would contain a lot of pixel values and it is typically easy for the network to learn the features if the image size is progressively reduced. Pooling layers help in reducing the number of parameters required and hence, this reduces the computation required. Pooling also helps in avoiding overfitting. There are two types of pooling operation that could be done:

(a) Max Pooling — Selecting the maximum value

(b) Average Pooling — Sum all of the values and dividing it by the total number of values

Code:

Before we start coding, I would like to let you know that the dataset we are going to use is the MNIST digits dataset and we are going to use the Keras library with a Tensorflow backend for building the model. Ok, enough. Let's do some coding.

```
1 import keras
2 from keras.datasets import mnist
3 from keras.models import Sequential
4 from keras.layers import Dense, Dropout, Flatten
5 from keras.layers import Conv2D, MaxPooling2D
import numpy as np
```

First, let us do some necessary imports. The keras library helps us build our convolutional neural network. We need to download the mnist dataset through keras. Next we import a sequential model, which is a pre-built keras model where you can add the layers. After that we introduce the convolution and pooling layers. We also import dense layers as they are used to predict the labels. The dropout layer reduces overfitting and the flattening layer expands a three-dimensional vector into a one-dimensional vector. Finally, we import numpy for matrix operations.

```
1 batch_size = 128
2 num_classes = 10
3 epochs = 12
4 # input image dimensions
5 img_rows, img_cols = 28, 28
6 # the data, split between train and test sets
7 (x_train, y_train), (x_test, y_test) = mnist.load_data()
8 x_train = x_train.reshape(60000, 28, 28, 1)
9 x_test = x_test.reshape(10000, 28, 28, 1)
10 print('x_train shape:', x_train.shape)
11 print(x_train.shape[0], 'train samples')
12 print(x_test.shape[0], 'test samples')
13 # convert class vectors to binary class matrices
14 y_train = keras.utils.to_categorical(y_train, num_classes)
15 y_test = keras.utils.to_categorical(y_test, num_classes)
```

Most of the statements in the above code would be trivial, I would just explain some lines of the code. We reshape x_train and x_test because our CNN accepts only a four-dimensional vector. The value 60000 represents the number of images in the training data, 28 represents the image size and one represents the number of channels. The number of channels is set to one if the image is in grayscale and if the image is in

```

1 model = Sequential()
2 model.add(Conv2D(32, kernel_size=(3, 3), activation='relu', input_shape=
  (28,28,1)))
3 model.add(Conv2D(64, (3, 3), activation='relu'))
4 model.add(MaxPooling2D(pool_size=(2, 2)))
5 model.add(Dropout(0.25))
6 model.add(Flatten())
7 model.add(Dense(128, activation='relu'))
8 model.add(Dropout(0.5))
9 model.add(Dense(num_classes, activation='softmax'))

```

RGB format, the number of channels is set to three. We also convert our target values into binary class matrices. To know what binary class matrices look like take a look at the example below.

We then build a sequential model and add convolutional layers and max pooling layers to it. We also add dropout layers in between, dropout randomly switches off some neurons in the network which forces the data to find new paths. Therefore, this reduces overfitting. We then add dense layers at the end which are used for class prediction(0–9).

```

1 model.compile(loss=keras.losses.categorical_crossentropy,
  optimizer=keras.optimizers.Adadelta(), metrics=['accuracy'])
2 model.fit(x_train, y_train, batch_size=batch_size,
  epochs=epochs, verbose=1, validation_data=(x_test,
  y_test))
3 score = model.evaluate(x_test, y_test, verbose=0)
4 print('Test loss:', score[0])
5 print('Test accuracy:', score[1])

```

Now we will compile the model compile the model with a categorical cross entropy loss function, Adadelta optimizer and an accuracy metric. The we fit the dataset to the model, i.e we train the model for 12 epochs. After training the model, we will evaluate the loss and accuracy of the model on the test data and print it (see Figure 26).


```
Blog Code — -bash — 152x42
(TensorFlow) Rohiths-MacBook-Pro:TensorFlow rohith$ cd ..
(TensorFlow) Rohiths-MacBook-Pro:Documents rohith$ cd Blog\ Code/
(TensorFlow) Rohiths-MacBook-Pro:Blog Code rohith$ clear

(TensorFlow) Rohiths-MacBook-Pro:Blog Code rohith$ python MNIST_Tutorial.py
/Users/rohith/anaconda3/lib/python3.6/site-packages/h5py/_init_.py:36: FutureWarning: Conversion of the second argument of issubdtype from 'float' to
'np.floating' is deprecated. In future, it will be treated as 'np.float64 == np.dtype(float).type'.
  from ..conv import register_converters as _register_converters
Using TensorFlow backend.
x_train shape: (60000, 28, 28, 1)
60000 train samples
10000 test samples
Train on 60000 samples, validate on 10000 samples
Epoch 1/12
2018-05-19 01:19:57.918340: I tensorflow/core/platform/cpu_feature_guard.cc:140] Your CPU supports instructions that this TensorFlow binary was not comp
iled to use: AVX2 FMA
60000/60000 [=====] - 225s 4ms/step - loss: 4.5511 - acc: 0.6714 - val_loss: 0.0961 - val_acc: 0.9716
Epoch 2/12
60000/60000 [=====] - 227s 4ms/step - loss: 0.1543 - acc: 0.9563 - val_loss: 0.0530 - val_acc: 0.9835
Epoch 3/12
60000/60000 [=====] - 222s 4ms/step - loss: 0.0996 - acc: 0.9722 - val_loss: 0.0560 - val_acc: 0.9816
Epoch 4/12
60000/60000 [=====] - 232s 4ms/step - loss: 0.0769 - acc: 0.9786 - val_loss: 0.0445 - val_acc: 0.9854
Epoch 5/12
60000/60000 [=====] - 240s 4ms/step - loss: 0.0641 - acc: 0.9819 - val_loss: 0.0456 - val_acc: 0.9862
Epoch 6/12
60000/60000 [=====] - 239s 4ms/step - loss: 0.0551 - acc: 0.9844 - val_loss: 0.0349 - val_acc: 0.9900
Epoch 7/12
60000/60000 [=====] - 244s 4ms/step - loss: 0.0490 - acc: 0.9860 - val_loss: 0.0420 - val_acc: 0.9885
Epoch 8/12
60000/60000 [=====] - 254s 4ms/step - loss: 0.0462 - acc: 0.9870 - val_loss: 0.0353 - val_acc: 0.9905
Epoch 9/12
60000/60000 [=====] - 301s 5ms/step - loss: 0.0386 - acc: 0.9888 - val_loss: 0.0367 - val_acc: 0.9895
Epoch 10/12
60000/60000 [=====] - 227s 4ms/step - loss: 0.0374 - acc: 0.9894 - val_loss: 0.0336 - val_acc: 0.9900
Epoch 11/12
60000/60000 [=====] - 203s 3ms/step - loss: 0.0345 - acc: 0.9894 - val_loss: 0.0340 - val_acc: 0.9899
Epoch 12/12
60000/60000 [=====] - 204s 3ms/step - loss: 0.0325 - acc: 0.9903 - val_loss: 0.0296 - val_acc: 0.9907
Test loss: 0.029622057321942247
Test accuracy: 0.9907
(TensorFlow) Rohiths-MacBook-Pro:Blog Code rohith$
```

Figure 26. The obtained output results



**** Please test the code and practice it on our jupyter server ****

<https://code.yuanzhuo.bnu.edu.cn/hub/spawn-pending/ubuntu>

Quiz: (Correct answers are: D and C)

How does deep learning work?

- A. It predicts future instances only when it's taught some knowledge that has been labeled in advance.
- B. It predicts accurate patterns based on small amount of data.
- C. It uses unstructured data to generate new patterns only if it's within an interactive environment.
- D. It incorporates neural networks in successive layers in order to learn from data in an iterative manner.

Which statement about Convolution Neural Network (CNN) is not true?

- A. The hidden layers in a CNN are generally convolution and pooling (downsampling) layers.
- B. CNN is largely applied to analyzing visual imagery.
- C. The pooling layer of CNN is to augment samples of data.
- D. CNN assembles complex patterns using smaller and simpler patterns.

Chapter 5 Natural interaction: Can computers understand humans and talk to them?

Age: 11+

AI is now used to make machines intelligent and interact and communicate like humans. For example, we can mention the case of Siri which is a smart virtual assistant developed by Apple Inc (see several examples in Figure 27). It uses voice queries and a natural-language user interface to answer questions, make recommendations, and perform actions by delegating requests to a set of internet services.

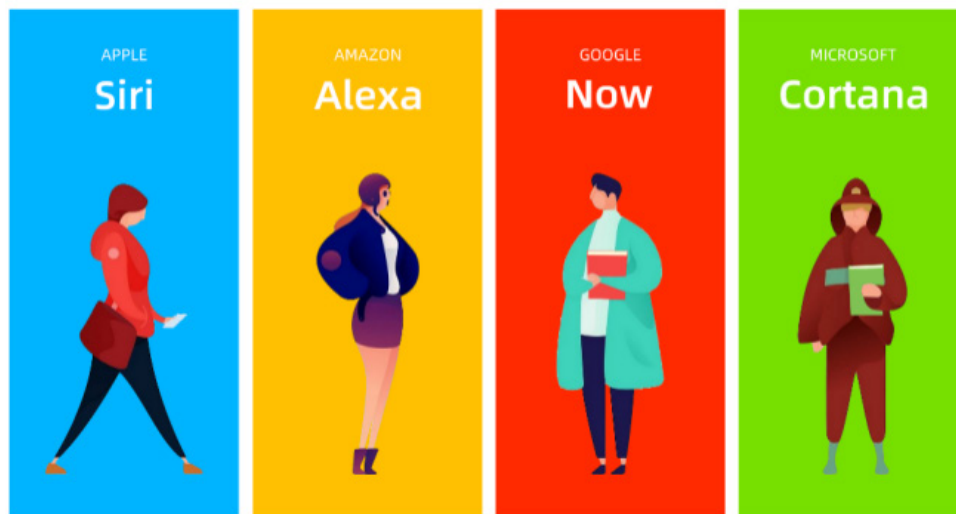


Figure 27. Examples of smart virtual assistants that can interact with humans

To conduct natural language interaction and communicate with humans, a machine conducts the following steps: (1) take the human input, such as speech, ; (2) analyze the input (speech) to understand what a person is trying to say; (3) do the reasoning process to provide an accurate answer or perform a task; and, (4) perform the action, by answering the user back or executing a specific task. To deliver an intelligent, humanlike experience, a machine should use a number of Natural Language Processing (NLP) principles and technologies. For instance, Figure 28 shows a communication scenario between a human and a virtual assistant to play a specific music.



Figure 28. A communication scenario between a human and a virtual assistant to play a specific music

NLP is defined as the application of computational techniques to the analysis and synthesis of natural language and speech. In other words: NLP is using different techniques from computer science (algorithms) to understand and manipulate human language and speech. NLP has the following two main components:

- Natural Language Understanding (NLU): It revolves around machine reading comprehension. This is an AI-hard problem. An NLU system needs the following components: (1) Lexicon, Parser, and Grammar rules; and, (2) Semantic theory to guide comprehension.
- Natural Language Generation (NLG) is concerned with generating natural language. It uses a machine representation system like a knowledge base or a logical form. You can think of it as a translator between data and natural language representation; this is the opposite of NLU. This involves three tasks: (1) Text Planning- To extract relevant content from the knowledge base; (2) Sentence Planning- To choose appropriate words, form meaningful phrases, and set sentence tone; and, (3) Text Realization- To map the sentence plan into sentence structure.

5.1 Smart chatbots to answer users' inquiries about COVID-19

Age: 11+

A chatbot is a computer program that allows humans to interact with technology using a variety of input methods such as voice, text, gesture and touch. It is known by several names, including as a conversational AI bot, AI chatbot, AI assistant, intelligent virtual assistant, virtual customer assistant, digital assistant, conversational agent, virtual agent, conversational interface and more, chatbots are growing in popularity. The chatbot uses NLP (see previous section) to analyze the text input, considers the best response and delivers that back to the user. For instance, it is possible to try the chatbot Elbot (<https://www.elbot.com/>). Smart chatbots, also known as AI chatbots, should have the following capabilities (Artificial Solutions, 2020).

- Intelligent Understanding is more than just correctly interpreting the user's request. It's about being able to instantly merge other pieces of information such as geolocation or previous preferences into the conversation to deliver a more complete answer.

- Memory allows a chatbot to remember pertinent details to reuse during a conversation or implicitly learn about a person to be reused later. For example, a mobile assistant might learn through previous requests and responses that the user clearly prefers Italian cuisine and so will use this information when asked for restaurant recommendations in future.
- Sentiment analysis enables a chatbot to understand the mood of the customer and the strength of that feeling. This is a particularly important in customer service type applications where it can be linked to complaint escalation flows, but also can be used in other more trivial ways such as choosing which songs to play upon request.
- Personality can make a huge difference to engagement and the trust users place in the chatbot. While some companies chose to reinforce it using avatars, personality can easily be conveyed in the conversation alone.
- Persistence allows people to pick up a conversation where they last left off, even if they switch devices, making for a more natural and seamless user experience.
- Topic switching enables the user to veer off onto another subject, such as asking about payment methods while enquiring if a product is in stock. The chatbot should also then be capable of bringing the user back on track if the primary intent is not reached.



Story 12: Human-Robot to clean hospitals during COVID-19

Age: 9+

Can you guess what the robot is doing (see Figure 29)? Yes, the robot is cleaning at the city state's Alexandra Hospital-Singapore. The robot can stop automatically when someone approaches from the opposite direction. It is popular at the hospital and some patients even wave at her. Remember to say “hello” to the robot when you see it.



Figure 29. A robot cleaner at hospitals during COVID-19



Have you thought about talking with a chatbot which can solve your health issues as well as other emergency questions? As shown in Figure 30, the Centers for Disease Control and Prevention (CDC) has developed the Chabot “Clara” to keep people safe during this pandemic.

CDC coronavirus chatbot

Coronavirus Self-Checker

Hi, I'm Clara. I'm here to guide you through the Coronavirus Self-Checker.

If you are experiencing a life-threatening emergency, please call 911 immediately.

This system does not replace the judgment of healthcare professionals or the performance of any clinical assessment.

To provide information on the right level of care, we are going to ask you a series of questions.

During the assessment, you can refresh the page if you need to start again

Ready? Let's get started.

Where are you located?

United States

Outside the US

CDC just now

Figure 30. The smart chatbot “Clara”

Age: 11+

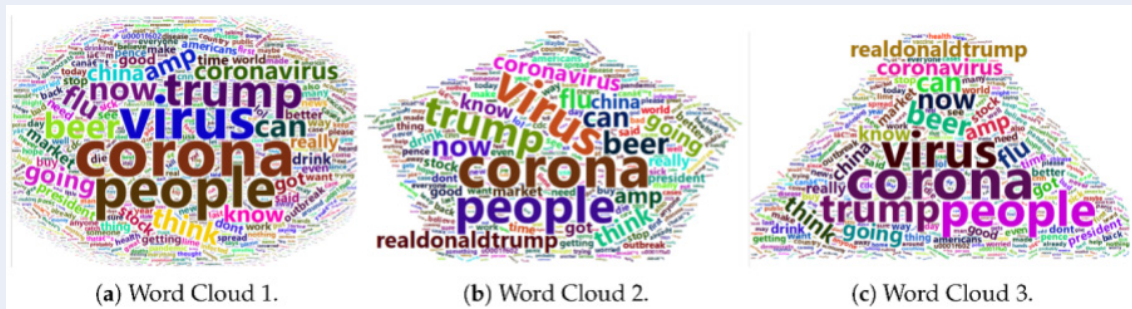


Figure 32: A couple of word cloud instances.

5.3 Hands-on learning experience: Sentence tokenization Age: 13+

When we talk about the natural language processing, we can first learn how to handle the sentence tokenization and formulate a document vector. Sentence tokenization (also called sentence segmentation) is the problem of dividing a string of written language into its component sentences. The idea here seems very simple. In English and some other languages, we can split apart the sentences whenever we see a punctuation mark. However, even in English, this problem is not trivial due to the use of full stop character for abbreviations. When processing plain text, tables of abbreviations that contain periods can help us to prevent incorrect assignment of sentence boundaries. In many cases, we use libraries to do that job for us, so don't worry too much about the details for now.

For example, let's look a piece of text about a famous board game called backgammon. It's *"Backgammon is one of the oldest known board games. Its history can be traced back nearly 5,000 years to archeological discoveries in the Middle East. It is a two-player game where each player has fifteen checkers which move between twenty-four points according to the roll of two dice."*

To apply a sentence tokenization with NLTK we can use the `nltk.sent_tokenize` function.

```
1 text = "Backgammon is one of the oldest known board games. Its history can
  be traced back nearly 5,000 years to archeological discoveries in the Middle
  East. It is a two-player game where each player has fifteen checkers which
  move between twenty-four points according to the roll of two dice."
2 sentences = nltk.sent_tokenize(text)
3 for sentence in sentences:
4     print(sentence)
5     print()
```

As an output, we get the three component sentences separately.

```
1 Backgammon is one of the oldest known board games.
2
3 Its history can be traced back nearly 5,000 years to archeological
  discoveries in the Middle East.
4
5 It is a two-player game where each player has fifteen checkers which move
  between twenty-four points according to the roll of two dice.
```

Accordingly, we can learn how to process bag-of-word and count the document vector. The bag-of-words model is a popular and simple feature extraction technique used when we work with text. It describes the occurrence of each word within a document.

To use this model, we need to:

Step 1: Design a vocabulary of known words (also called tokens)

Step 2: Choose a measure of the presence of known words

Any information about the order or structure of words is discarded. That's why it's called a bag of words. This model is trying to understand whether a known word occurs in a document, but we don't know where that word is in the document. The intuition is that similar documents have similar contents. Also, via the content, we can learn something about the meaning of the document. There are two steps,

1. Load the Data: we have a reviews.txt file, it is our data and we want to load it as an array. The file content is:

```
1 I like this movie, it's funny. I hate this movie. This was awesome! I like
  it. Nice one. I love it.
```

To achieve this, we can simply read the file and split it by lines.

```
1 with open("simple movie reviews.txt", "r") as file:
2     documents = file.read().splitlines()
3 print(documents)
```

The output is

```
1 ["I like this movie, it's funny.", 'I hate this movie.', 'This was awesome!
  I like it.', 'Nice one. I love it.']
```


2. Design the Vocabulary

Let's get all the unique words from the four loaded sentences ignoring the case, punctuation, and one-character tokens. These words will be our vocabulary (known words). Thus, we need to score the words in each document. The task here is to convert each raw text into a vector of numbers. The simplest scoring method is to mark the presence of words with one for present and zero for absence.

Now, let's see how we can create a bag-of-words model using the mentioned above CountVectorizer class.

```
1 # Import the libraries we need
2 from sklearn.feature_extraction.text
3 import CountVectorizerimport pandas as pd
4
5 # Step 2. Design the Vocabulary
6 # The default token pattern removes tokens of a single character. That's why
   we don't have the "I" and "s" tokens in the output
7 count_vectorizer = CountVectorizer()
8
9 # Step 3. Create the Bag-of-Words Model
10 bag_of_words = count_vectorizer.fit_transform(documents)
11
12 # Show the Bag-of-Words Model as a pandas DataFrame
13 feature_names = count_vectorizer.get_feature_names()
14 pd.DataFrame(bag_of_words.toarray(), columns = feature_names)
```

Our output is shown as follows,

	awesome	funny	hate	it	like	love	movie	nice	one	this	was
0	0	1	0	1	1	0	1	0	0	1	0
1	0	0	1	0	0	0	1	0	0	1	0
2	1	0	0	1	1	0	0	0	0	1	1
3	0	0	0	1	0	1	0	1	1	0	0

Finally, we can compare this to our text content. This was a basic introduction to natural language processing.

```
1 I like this movie, it's funny.I hate this movie.This was awesome! I like
   it.Nice one. I love it.
```



**** Please test the code and practice it on our jupyter server ****

<https://code.yuanzhuo.bnu.edu.cn/hub/spawn-pending/ubuntu>

Quiz: (Correct answers are: C and A)

Please select the appropriate example that utilizes Natural Language Process (NLP) in fighting COVID-19 pandemic?

- A. Administrators of Twitter are deleting tweets irrelevant to coronavirus.
- B. Alexa telling traffic conditions with voice inputs.
- C. Chatbots answering patients' inquiries with various input such as voice and text.
- D. AI technology diagnosing patients' symptoms and predicting how severe they are by collecting the data of their chest CT.

What are the main components of Natural Language Process (NLP)?

- A. Natural Language Understanding and Natural Language Generation
- B. Natural Language Pairing and Natural Language Converting
- C. Natural Language Understanding and Natural Language Pairing
- D. Natural Language Generation and Natural Language Converting

Chapter 6 Social Impact: How AI will impact your life?

Age: 9+

Artificial Intelligence (AI) may have good impact on societies, ecosystems, and human lives, including the human mind, in part because of the new ways in which it influences human thinking and decision-making, and affects education, science, culture, and communication and information. AI systems can be of great service to humanity, but also raises fundamental concerns. For instance, we can talk about artificial stupidity, racist robots, discrimination, job loss etc. The following AI examples can be found that someone should know and consider while developing AI-based solutions.

- Inaccurate data while training AI systems can lead to inaccurate or unfair decisions for some people. For instance, the system may not detect that someone is infected with COVID-19 when he/she is infected. This can put his/her life and the life of others in danger.
- AI systems should consider the privacy of persons (patients) while treating them in order to keep their information safe and protect their private lives.
- The data generated by AI systems can be stored in different servers (e.g. hospital servers, institution servers, etc.), therefore someone may ask if this data belongs to the person (patient) since the system used his/her data, or to the developer/company because it developed the system or to the third party (hospitals, institutions, etc.) since they provided the servers for data collection and processing. Data ownership is a big issue in the field of AI that should be considered.
- Transparency of how AI systems work, including algorithms, data collection and processing is another issue that should be considered.

As shown in this book, AI adoption, especially in health care, is very useful, but still complex due to the social and ethical issues presented above. Whether AI is good or bad can be examined from several perspectives. Therefore, it is crucial to keep in mind these issues while analyzing the broader societal issues at play. It is important to keep learning and stay informed in order to make good decisions for our future.

Furthermore, there is no doubt that AI and other emerging technologies, such as robotics, will take over several jobs in the future to make the work automated and faster (Internet of Business, n.d.). However, AI will also open up new opportunities for new jobs that did not exist before. These jobs will require new skills and new knowledge that were not needed long time ago, such as Python, React (web), Angular, machine learning, and Docker (Columbus, 2019). Several institutions and schools are now updating their curriculum to keep up with AI technologies. Therefore, it is very important to keep learning about AI and its related technologies for a better future career.

Quiz: (Correct answers are: D and B)

Please select the correct statement:

- A. AI does not have a bad side.
- B. The data generated by AI can only be saved in one server.
- C. Using inaccurate data to train AI systems can lead to inaccurate decisions.
- D. AI will lead to taking off some old job, but creating new opportunities for new ones instead.

What changes will not be brought by AI?

- A. Changes in social ethics
- B. Changes in human's facial characteristics
- C. Changes in international relations
- D. Changes in employment structure

Conclusion

AI is typically presented in Hollywood movies as a human-hating robot in a doomsday scenario. However, in the current COVID-19 pandemic, AI has emerged as a superhero that can save humanity from viruses and greatly reduce the number of deaths globally. AI researchers around the world have, over the past few years, engineered drastically new capabilities in health care, just in time to combat the novel coronavirus.

AI has been used to fight the virus in different stages, from screening and diagnosis to vaccine development. With its ability to learn quickly from data related to COVID-19, AI has saved human beings. Physical robots also come in handy. For instance, self-navigating robots have worked to disinfect hospitals, deliver food and medication, and check body temperatures. At airports and train stations, temperature sensors are used to scan crowds for high temperatures. They are also used with a face recognition system, which can pinpoint individuals with a high temperature and whether they are wearing a surgical mask or not. This system is also being used to ensure that citizens are obeying self-quarantine regulations. Particularly, those who broke the laws and left home would get a call from the authorities, presumably after being tracked by the facial recognition system.

Therefore, since severe crises are unlikely to disappear, it is important to prepare the young generation for the next pandemic especially in the era of AI. This could be ensured by preparing courses that could familiarize the young generation with an AI background as well as its possible tasks and functionalities. With this in mind, this book presented the basic solutions of AI in a fun and easy way, as well as vivid stories about how AI is used worldwide to prevent the spread of COVID-19. Finally, this book presented the AI challenges that should be considered in the future.

Reference

AIMBOT, <https://www.ubtrobot.com/cn/products/aimbot>

Artificial Solutions. (2020). Chatbots: The Definitive Guide.

Build Your Own Convolution Neural Network in 5 mins, HYPERLINK "<https://towardsdatascience.com/build-your-own-convolution-neural-network-in-5-mins-4217c2cf964f>" <https://towardsdatascience.com/build-your-own-convolution-neural-network-in-5-mins-4217c2cf964f>

Columbus, L. (2019) AI Skills Among The Most In-Demand For 2020. Accessed from: <https://www.forbes.com/sites/louiscolombus/2019/11/27/ai-skills-among-the-most-in-demand-for-2020/#77bbd7d86b44>

Decision tree in Python, <https://towardsdatascience.com/decision-tree-in-python-b433ae57fb93>

Face Detection in just 5 lines of code, <https://towardsdatascience.com/face-detection-in-just-5-lines-of-code-5cc6087cb1a9>

Graesser, A. C., McNamara, D. S., Louwerse, M. M., & Cai, Z. (2004). Coh-metrix: analysis of text on cohesion and language. Behavior Research Methods, Instruments, & Computers: A Journal of the Psychonomic Society, Inc, 36(2), 193–202.

Hygiene-obsessed Singapore deploys robots to keep coronavirus away in Asian Review, <https://asia.nikkei.com/Business/Technology/Hygiene-obsessed-Singapore-deploys-robots-to-keep-coronavirus-away>

<https://cvd.lti.cmu.edu>, the demonstration of CDC in Carnegie Mellon University

Internet of Business. (n.d.). Robotics, A.I will create 58 million jobs, decimate middle-class careers: World Economic Forum. Accessed from: <https://internetofbusiness.com/robotics-a-i-will-create-jobs-but-decimate-middle-class-careers-wef/>

McNamara, D. S., Graesser, A. C., McCarthy, P. M., & Cai, Z. (2014). Automated Evaluation of Text and Discourse with Coh-Metrix. Cambridge University Press.

Tuli, Shreshth, et al. "Predicting the Growth and Trend of COVID-19 Pandemic using Machine Learning and Cloud Computing." Internet of Things (2020): 100222.

The CT evaluation of Carnegie Mellon University, <https://www.yitutech.com/>

The demonstration of AI-Based Triage and Monitoring System, <http://www.diagnosticrobotics.com>

The AI perdition tool of the conversation, <http://theconversation.com>

The demonstration of AI-Based Triage and Monitoring System, <http://www.diagnosticrobotics.com>,

Visualizing COVID-19 Data Beautifully in Python by Nik Piepenbreier, HYPERLINK "<https://towardsdatascience.com/visualizing-covid-19-data-beautifully-in-python-in-5-minutes-or-less-affc361b2c6a>" <https://towardsdatascience.com/visualizing-covid-19-data-beautifully-in-python-in-5-minutes-or-less-affc361b2c6a>

Youth Artificial Intelligence Innovation Initiatives (YAIII)

The development and application of intelligent technology represented by artificial intelligence has brought changes in education content, mode, and environment. In order to cope with the opportunities and challenges of education in the era of artificial intelligence (AI), it is better to carry out AI education in primary and secondary schools, and cultivate innovative talents for the future.

In this context, Beijing Normal University (BNU) is actively advocating a lot of attention to the education of artificial intelligence and the growth of young people. In 2019, the National Engineering Laboratory for Cyber learning and Intelligent Technology (CIT) and Smart learning Institute of Beijing Normal University (SLI) have officially released the Youth Artificial Intelligence Innovation Initiative (YAIII).

YAIII aims to stimulate teenagers' curiosity and interest, and to raise the capability in developing original and innovative algorithms to solve authentic complex problems by creating the deep linkage between schools and all society. To prepare the young generation for the future of diverse, uncertain, and intelligent world with fundamental digital literacy and survival ability, YAIII intends to explore public service for students and teachers with advanced artificial intelligence in education by creating a collaborative mechanism among schools and all stakeholders.





北京师范大学智慧学习研究院
Smart Learning Institute of Beijing Normal University

Smart Learning Institute of Beijing Normal University (SLIBNU)

Beijing Normal University (BNU) grew out of the Education Department of Imperial University of Peking established in 1902, which initiated teacher training in China's higher education. After the development for over a century, BNU has become a comprehensive and research-intensive university with its main characteristics of basic disciplines in sciences and humanities, teacher education and educational science. Smart Learning Institute (SLI) is jointly established by Beijing Normal University and a global educational technology company NetDragon Websoft. SLI is a comprehensive experimental platform involving scientific research, technology development, and innovative instruction. SLI focuses on detecting learning patterns powered by ICT, creating smart learning environments and platforms for life-long and life-wide learning, as well as supporting diversified, personalized and differential learning needs for digital learners.



International association of smart learning environment (IASLE)

The International association of smart learning environments (IASLE) is a cutting-edge professional forum for researchers, academics, practitioners, and industry professionals interested and/or engaged in the reform of the ways of teaching and learning through advancing current learning environments towards smart learning environments. It provides opportunities for discussions and constructive dialogue among various stakeholders on the limitations of existing learning environments, need for reform, innovative uses of emerging pedagogical approaches and technologies, and sharing and promotion of best practices, leading to the evolution, design and implementation of smart learning environments.



Arab League's Educational, Cultural and Scientific Organization (ALECSO)

The Arab League Educational, Cultural and Scientific Organization (ALECSO) is a Tunis-based specialized institution working under the umbrella of the League of Arab States. It is essentially concerned with the development and coordination of the activities related to education, culture and sciences in the Arab World. It was established by virtue of Article 3 of the Arab Cultural Unity Charter, and was officially announced in Cairo on July 25, 1970. As stated in Article One of its Constitution, ALECSO was established with the aim of promoting Arab intellectual unity through education, culture and sciences, and enhancing the educational, cultural and scientific level in the Arab World so that it can positively contribute to universal civilization.



Edmodo

Edmodo is an educational technology company offering a communication, collaboration, and coaching platform to K-12 schools and teachers. The Edmodo network enables teachers to share content, distribute quizzes, assignments, and manage communication with students, colleagues, and parents. Edmodo is very teacher-centric in their design and philosophy: students and parents can only join Edmodo if invited to do so by a teacher. Teachers and students spend large amounts of time on the platform, both in and out of the classroom. Edmodo is free to use, but it also offers premium services.



北京师范大学智慧学习研究院
Smart Learning Institute of Beijing Normal University



IASLE
INTERNATIONAL ASSOCIATION
OF SMART LEARNING ENVIRONMENTS



edmodo

Website: <http://sli.bnu.edu.cn/en/>

Address: 12F, Block A, Jingshi Technology Building,
No. 12 Xueyuan South Road, Haidian
District, Beijing, China

Email: smartlearning@bnu.edu.cn

Phone: 8610-58807219

Postcode: 100082



[HTTP://SLI.BNU.EDU.CN/EN](http://sli.bnu.edu.cn/en)