Extended Intelligences - Repurposing food waste
Carlotta, Flora, Jorge and Manuja

Why is repurposing food waste positively important?

Repurposing food waste is a crucial aspect of sustainable waste management with far-reaching environmental, social, and economic implications. Here's a detailed exploration of the importance of repurposing food waste:

1. Environmental Impact:

a. Reduced Methane Emissions: When food waste ends up in landfills, it decomposes anaerobically, producing methane—a potent greenhouse gas that contributes to climate change. By repurposing food waste, particularly through composting, you can significantly reduce methane emissions.

b. Conservation of Resources: Food production requires substantial resources such as water, energy, and land. When food is wasted, these resources are essentially squandered. Repurposing food waste into useful products like plant fertilizer can close the loop, contributing to a more sustainable and circular economy.

c. Soil Enrichment: Using food waste as plant fertilizer enhances soil fertility and structure. This, in turn, reduces the need for synthetic fertilizers, which can have negative environmental impacts such as water pollution and habitat destruction.

2. Economic Benefits:

a. Cost Savings for Businesses: Businesses that generate large quantities of food waste, such as restaurants or food processing units, can experience cost savings by implementing effective food waste repurposing strategies. This includes reduced waste disposal fees and potential revenue from selling or using the repurposed products.

b. Job Creation: Industries related to the repurposing of food waste, such as composting facilities or biogas plants, can create job opportunities. This contributes positively to the economy and helps build a more sustainable workforce.

3. Social Implications:

a. Addressing Hunger and Poverty: In some cases, edible food is discarded due to cosmetic imperfections or overproduction. By redirecting this food to those in need, repurposing food waste can contribute to addressing issues of hunger and poverty.

b. Educational Opportunities: Initiatives focused on repurposing food waste provide opportunities for educational programs. Educating communities about the importance of reducing food waste and its environmental impacts can lead to more responsible consumption patterns.

4. Prevention of Excess Food Waste:

a. Supply Chain Optimization: Implementing better inventory management and distribution systems can help prevent overproduction and excess food waste. This optimization can lead to cost savings for businesses and reduce the overall environmental footprint of the food industry.

b. Consumer Awareness: Education campaigns aimed at consumers can raise awareness about the impact of food waste and encourage responsible purchasing and consumption habits. Smaller portion sizes, better storage practices, and creative cooking with leftovers can all contribute to reducing excess food waste at the consumer level.

In summary, repurposing food waste is a multifaceted solution that addresses environmental concerns, promotes economic efficiency, and contributes to social well-being. It represents a sustainable approach to managing resources and aligns with the principles of a circular economy, where waste is minimized, and resources are used more efficiently.

Why is creating an AI bot that does this important?

Creating an AI bot specifically designed to repurpose food waste is important in the current generation for several reasons, leveraging technology to address sustainability challenges and promote responsible resource management. Here are detailed explanations of the significance:

1. Efficiency and Scale:

a. Data Processing: AI bots can efficiently process large volumes of data, including image recognition and analysis. This capability is crucial when dealing with diverse types of food waste and optimizing the repurposing process.

b. Real-time Decision Making: AI can make rapid decisions based on the analysis of food waste images. This allows for quicker responses in repurposing efforts, ensuring that the waste is utilized before it degrades or becomes unsuitable for certain applications.

2. Precision and Adaptability:

a. Targeted Repurposing: AI can be programmed to identify specific types of food waste and recommend or execute targeted repurposing strategies. This precision ensures that the repurposing methods employed are most suitable for the composition of the waste.

b. Adaptation to Variability: Food waste can vary in composition and condition. An AI bot can adapt its processing methods based on the specific characteristics of the waste, ensuring optimal repurposing outcomes.

3. Optimizing Resource Allocation:

a. Resource Efficiency: AI algorithms can optimize the use of resources in the repurposing process. For instance, the bot can calculate the right mix of food waste components for fertilizer production, maximizing its nutrient content and minimizing the need for additional inputs.

b. Energy and Time Savings: By automating and optimizing the repurposing process, AI bots can contribute to energy and time savings. This is particularly relevant in industries where large-scale repurposing of food waste occurs.

4. Promoting Innovation:

a. Technology-Driven Solutions: Creating an AI bot for repurposing food waste exemplifies the use of technology to address environmental challenges. This not only provides a practical solution but also fosters an environment of innovation in the field of waste management.

b. Inspiring Other Initiatives: The development of AI bots for food waste repurposing can inspire other technological initiatives aimed at solving environmental issues. It showcases the potential of AI in driving sustainable practices across various industries.

5. Education and Awareness:

a. Public Engagement: An AI bot can be integrated into educational programs and public awareness campaigns. This can engage individuals and communities in discussions about the environmental impact of food waste and the role of technology in creating sustainable solutions.

b. Behavioral Change: The presence of an AI bot can influence consumer behavior by highlighting the importance of reducing food waste. This, in turn, can contribute

to a cultural shift towards more responsible consumption and waste management practices.

6. Scalability and Accessibility:
a. Global Impact: AI bots can be deployed globally, making a significant impact on a large scale. This is crucial in addressing food waste as a global challenge with environmental consequences that extend beyond local borders.

b. Accessible Solutions: As technology becomes more accessible, the use of AI bots for food waste repurposing can become a widely adopted solution, benefiting various industries, communities, and individuals.

In summary, creating an AI bot for repurposing food waste is important in this generation because it combines the efficiency of technology with the urgency of addressing environmental issues. It not only offers practical solutions to food waste challenges but also sets a precedent for the integration of AI in sustainable practices, contributing to a more environmentally conscious and technologically advanced society.

General Steps and Processes for Creating an AI

Step 1: Define the Problem and Objectives
Clearly outline the problem you want the AI bot to solve and establish the objectives. In this case, you aim to repurpose food waste, so the AI bot should be capable of recognizing and categorizing different types of food waste in images.

Step 2: Gather and Prepare Datasets
Collect a diverse dataset of images containing various types of food waste. You'll need labeled images for training and validating your AI model. Ensure that the dataset is representative of the real-world scenarios the bot will encounter.

Step 3: Data Preprocessing
Prepare the dataset for training by performing necessary preprocessing tasks:
- Resize images to a consistent size.
- Normalize pixel values (typically scaling to a range of 0 to 1).
- Augment data (optional) to increase model robustness.

Step 4: Split the Dataset

Divide the dataset into training, validation, and testing sets. The training set is used to train the model, the validation set helps tune hyperparameters, and the testing set evaluates the model's performance on unseen data.

Step 5: Design the Neural Network Architecture
Choose a suitable deep learning architecture for image classification. Convolutional Neural Networks (CNNs) are commonly used for image-related tasks. You can design a custom architecture or use pre-trained models like ResNet, VGG, or MobileNet and fine-tune them for your specific task.

Step 6: Model Training
Train the neural network using the training dataset. During training, the model learns to recognize patterns and features in the images. Adjust hyperparameters like learning rate and batch size to optimize the model's performance.

Step 7: Model Validation
Evaluate the model on the validation set to ensure it generalizes well to new data. Fine-tune the model and hyperparameters if needed to improve performance.

Step 8: Model Testing
Once satisfied with the model's performance on the validation set, test it on the unseen testing set to assess its real-world applicability.

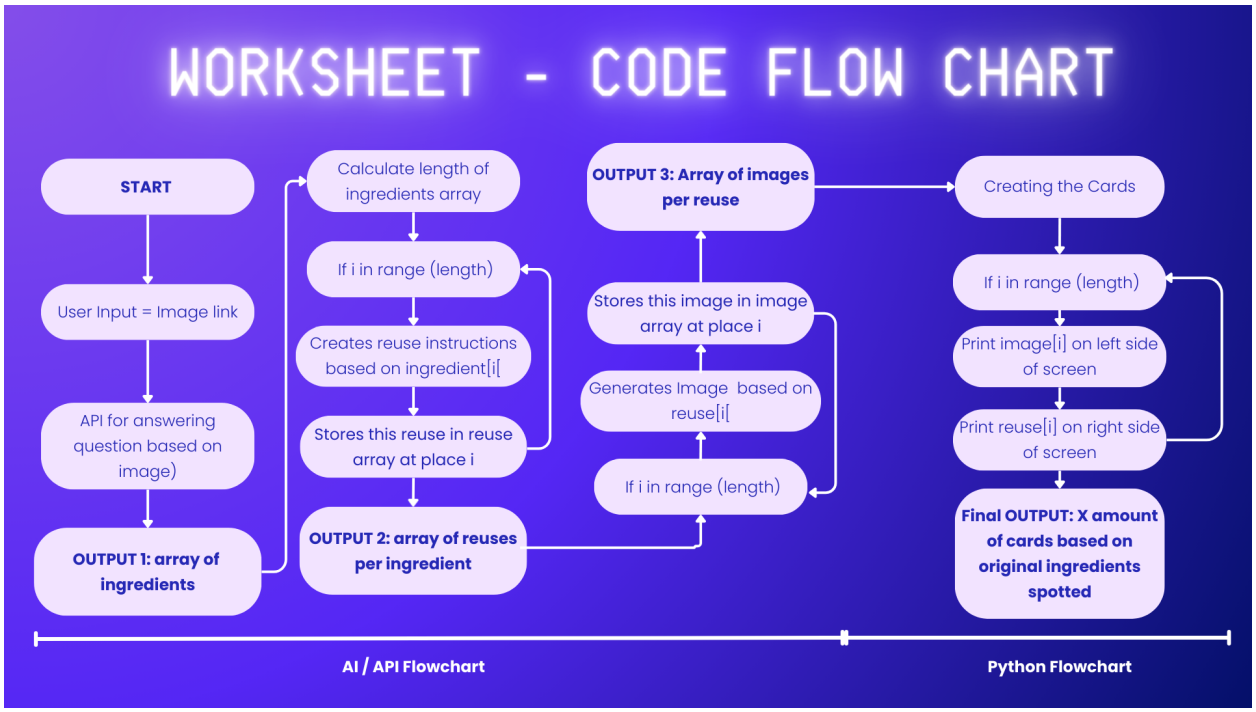
Step 9: Deployment
Deploy the trained model into your Replicate environment or any hosting platform of your choice. Ensure that the AI bot is accessible and can receive images for processing.

Step 10: Continuous Monitoring and Improvement
Regularly monitor the AI bot's performance in real-world scenarios. If needed, retrain the model with new data to adapt to evolving patterns and improve accuracy.

The Code

Before showing the actual code it is important to say that while this does output the desired results it is not trained completely as we hope. As we did not have the time to train our own model using existing neural networks and datasets we chose to use existing APIs. For the first three steps we found existing APIs on Replicate and then spent some time editing the prompts inputted to get the desired output. Once that was done it was a lot of trial and error to create the final desired output. Below you see a flowchart of how it all works and the total code.

The Flowchart



APIs Used within Code

To create the final working AI system we implemented three different APIs found on Replicate. The first API implemented was an image to text API that took a specific question and answered this (*https://replicate.com/andreasjansson/blip-2*). For this the question prompt was *"Which fruits/vegetables do you see, sum up with commas?".* This allowed us to receive a summed up list that could be stored in an array.

From here we used a second API that was a language model from Meta that allowed us to input a prompt and have the AI print out a text based on this (https://replicate.com/meta/llama-2-70b-chat). Here the prompt used was *"Give me the top way to repurpose the food waste of {ingredients[i]}"*. Where ingredients[i] was an element of the array

outputted from the first API. This once again allowed us to store the reuse of each ingredient in another array.

The final API used was a text to image generator (*https://replicate.com/stability-ai/stable-diffusion*) . We used this to create a fun image in which the food waste inputted was shown in a new way. Here the prompt was the reuse given by the second API. Once this was outputted then as shown on the flowchart python was used to combine the various elements together.

The Final Code

```
!pip install replicate
```

```
import replicate

api_token='Your Own Token'

client = replicate.Client(api_token=api_token)
```

Step 1: Create List of Ingredients Based on Image of food scraps

```
output = client.run(

"andreasjansson/blip-2:f677695e5e89f8b236e52ecd1d3f01beb44c34606419bcc19345e046d8f786f9",
  input={
    "image": "https://drive.google.com/uc?id=1BNsjpDRJXqvwJuXtUEPVNI6QRdow79Y9",
    "caption": False,
    "question": "Which fruits/vegetables do you see, sum up with commas?",
    "temperature": 1,
    "use_nucleus_sampling": False
  }
)
print(output)
```

```
output_text = output  # Replace "your_output_key" with the actual key in your output containing the text

# Split the text into a list of words
words_list = output_text.split()
```

```
# Find the length of the list, which is the number of words
length = len(words_list)

print(length)
```

```
ingredients=[]
for i in range (length):
  if words_list:
    # Extract the first word and store it in ingredient
    word = words_list[i]
    ingredients.append(word)
    print(ingredients[i])
```

Step 2: Search for each Reuse based on each ingredient

```
reuse = []
for i in range(length):
    output = client.run(

"meta/llama-2-70b-chat:02e509c789964a7ea8736978a43525956ef40397be9033abf9fd2badfe68c9e3",
      input={
        "debug": False,
        "top_k": 50,
        "top_p": 1,
        "prompt": f"Give me the top way to repurpose the foodwaste of {ingredients[i]}?",
        "temperature": 0.5,
        "system_prompt": "The best way to repurpose ... is as a .... The steps you must take to execute this are ...",
        "max_new_tokens": 250,
        "min_new_tokens": -1
      }
    )
    generated_text = ""
    for sentence in output:
      generated_text += sentence
    reuse.append(generated_text)

for i in range(length):
```

```
  print(reuse[i])
```

Step 3: Use image generator to generate image based on reuse title

```
image =[]
for i in range (length):
  import replicate
  output = client.run(

"stability-ai/stable-diffusion:ac732df83cea7fff18b8472768c88ad041fa750ff76
82a21affe81863cbe77e4",
    input={"prompt": f"{reuse[i]}"}
)
  image.append(output)

for i in range(length):
  print(image[i])
```

Step 4 combine picture and text to card

```
from IPython.display import display, Image, HTML

# Display the image on the left
for i in range(length):
  image_path = "path/to/your/image.jpg"
  img = Image(image[i][0], width=500)
  display(img)

# Display the text on the right, aligned at the top
# Display the text in the center-right of the image
for i in range(length):
  text = f"{reuse[i]}"
  html_code = f'<div style="position: absolute; top:  {25 + i * 50}%;
left: 520px; transform: translate(0, -50%); text-align:
left;">{text}</div>'
  display(HTML(html_code))
```

Thank you!