

LIXO • PLÁSTICO • PAPEL

Classificador de sacolas usando Transfer Learning

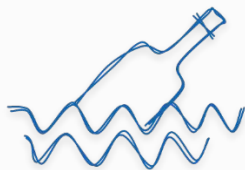
Thaís de Araújo de Medeiros

CONTEXTUALIZANDO



100 mil

Animais marinhos que morrem
todos os anos por ingestão
de plástico



8 milhões de toneladas

Quantidade de plástico
descartada nos oceanos
anualmente



400 anos

Tempo médio de decomposição
do plástico no meio ambiente

O PROJETO



?



?



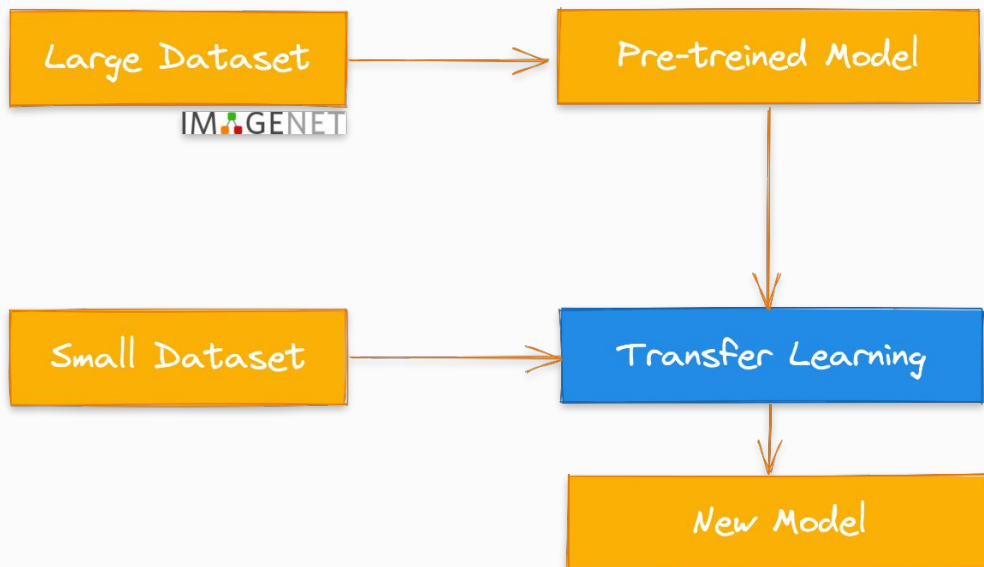
Sacola de
plástico

Sacola de
papel

Sacola de
lixo

TRANSFER LEARNING

Técnicas



DATASET

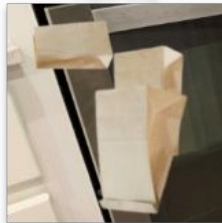
Plastic Bag



Plastic Bag



Paper Bag



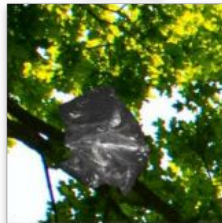
Paper Bag



Plastic Bag



Plastic Bag



Garbage Bag



Garbage Bag



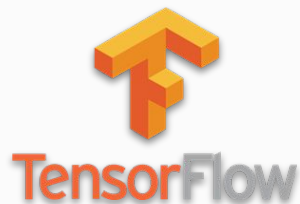
Plastic Bag



15000 imagens

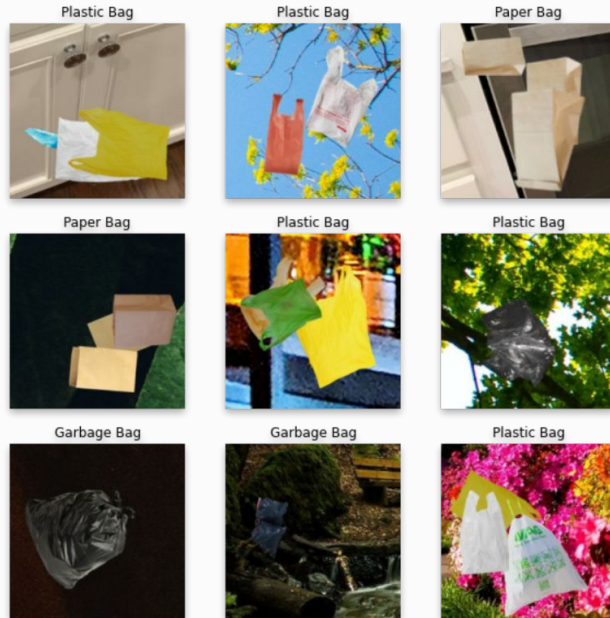
FERRAMENTAS

kaggle



CONSTRUÇÃO

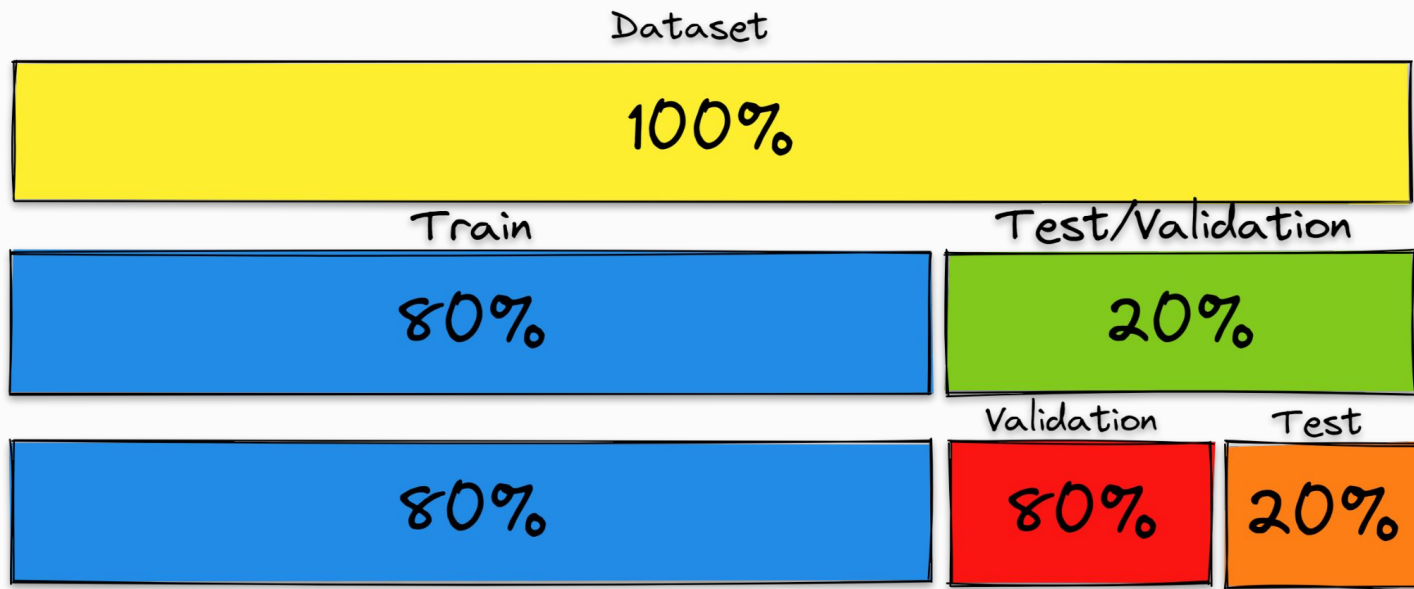
Fetch Data



----->  **Weights & Biases**

CONSTRUÇÃO

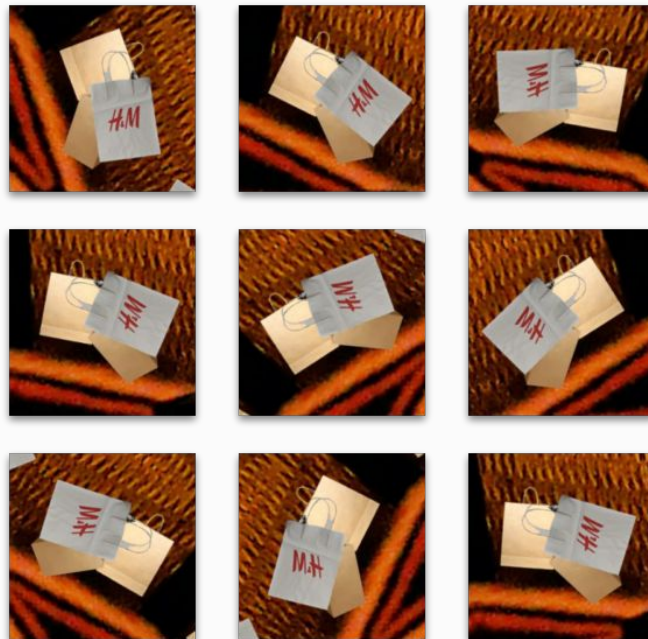
Data Segregation



CONSTRUÇÃO

Data Augmentation

```
for image, _ in train_dataset.take(1):  
    plt.figure(figsize=(10, 10))  
    first_image = image[0]  
    for i in range(9):  
        ax = plt.subplot(3, 3, i + 1)  
        augmented_image = data_augmentation(tf.expand_dims(first_image, 0))  
        plt.imshow(augmented_image[0] / 255)  
        plt.axis('off')
```



CONSTRUÇÃO

Transfer Learning - VGG 16 [Feature Extractor]

```
learning_rate = 0.0001
model.compile(optimizer=tf.keras.optimizers.Adam(learning_rate=learning_rate),
              loss="categorical_crossentropy",
              metrics=['accuracy'])
```

```
history = model.fit(train_dataset,
                    validation_data=validation_dataset,
                    batch_size=32,
                    epochs=20,
                    verbose=1)
```

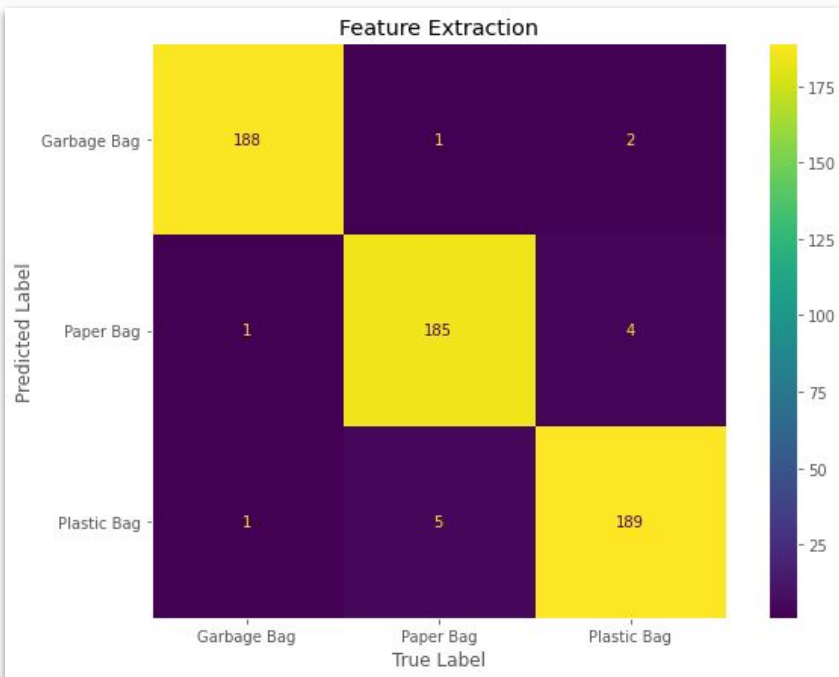
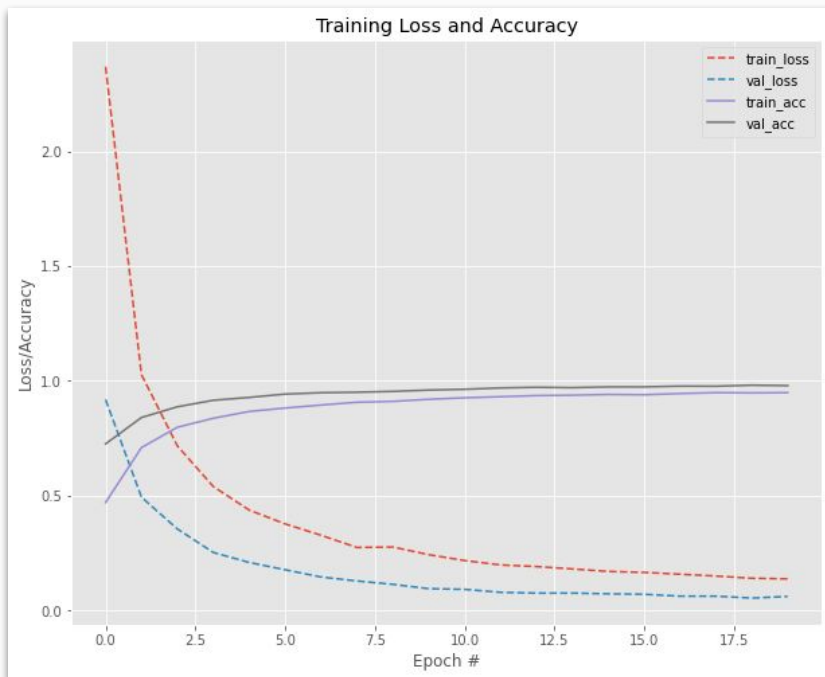
Layer (type)	Output Shape	Param #
input_2 (InputLayer)	[(None, 224, 224, 3)]	0
sequential (Sequential)	(None, 224, 224, 3)	0
tf.__operators__.getitem (SlicingOpLambda)	(None, 224, 224, 3)	0
tf.nn.bias_add (TFOpLambda)	(None, 224, 224, 3)	0
vgg16 (Functional)	(None, 7, 7, 512)	14714688
global_average_pooling2d (GlobalAveragePooling2D)	(None, 512)	0
dropout (Dropout)	(None, 512)	0
dense (Dense)	(None, 3)	1539

=====

Total params: 14,716,227
Trainable params: 1,539
Non-trainable params: 14,714,688

CONSTRUÇÃO

Transfer Learning - VGG 16 [Feature Extractor]



CONSTRUÇÃO

Transfer Learning - VGG 16 [Feature Extractor]

```
[INFO] evaluating network...
18/18 [=====] - 3s 158ms/step
```

	precision	recall	f1-score	support
Garbage Bag	0.98	0.99	0.99	190
Paper Bag	0.97	0.97	0.97	191
Plastic Bag	0.97	0.97	0.97	195
accuracy			0.98	576
macro avg	0.98	0.98	0.98	576
weighted avg	0.98	0.98	0.98	576

CONSTRUÇÃO

Transfer Learning - VGG 16 [Fine-tuning]

```
# a fully connect network
class FCHeadNet:
    @staticmethod
    def build(baseModel, classes, neurons):
        # initialize the head model that will be placed on top of
        # the base, then add a FC layer
        headModel = baseModel.output
        headModel = Flatten(name="flatten")(headModel)
        headModel = Dense(neurons, activation="relu")(headModel)
        headModel = Dropout(0.5)(headModel)

        # add a softmax layer
        headModel = Dense(classes, activation="softmax")(headModel)

        # return the model
        return headModel
```

CONSTRUÇÃO

Transfer Learning - VGG 16 [Fine-tuning | STAGE 01]

Layer (type)	Output Shape	Param #
input_5 (InputLayer)	[(None, 224, 224, 3)]	0
sequential (Sequential)	(None, 224, 224, 3)	0
tf.__operators__.getitem_1 (SlicingOpLambda)	(None, 224, 224, 3)	0
tf.nn.bias_add_1 (TFOpLambda)	(None, 224, 224, 3)	0
vgg16 (Functional)	(None, 7, 7, 512)	14714688
flatten (Flatten)	(None, 25088)	0
dense_1 (Dense)	(None, 256)	6422784
dropout_1 (Dropout)	(None, 256)	0
dense_2 (Dense)	(None, 3)	771

=====

Total params: 21,138,243
Trainable params: 21,138,243
Non-trainable params: 0

Freeze
→

Layer (type)	Output Shape	Param #
input_5 (InputLayer)	[(None, 224, 224, 3)]	0
sequential (Sequential)	(None, 224, 224, 3)	0
tf.__operators__.getitem_1 (SlicingOpLambda)	(None, 224, 224, 3)	0
tf.nn.bias_add_1 (TFOpLambda)	(None, 224, 224, 3)	0
vgg16 (Functional)	(None, 7, 7, 512)	14714688
flatten (Flatten)	(None, 25088)	0
dense_1 (Dense)	(None, 256)	6422784
dropout_1 (Dropout)	(None, 256)	0
dense_2 (Dense)	(None, 3)	771

=====

Total params: 21,138,243
Trainable params: 6,423,555
Non-trainable params: 14,714,688

CONSTRUÇÃO

Transfer Learning - VGG 16 [Fine-tuning | STAGE 01]

```
opt = RMSprop(learning_rate=0.0001)
model.compile(loss="categorical_crossentropy", optimizer=opt,
              metrics=["accuracy"])
```

```
history = model.fit(train_dataset,
                    validation_data=validation_dataset,
                    batch_size=32,
                    epochs=3,
                    verbose=1)
```

[INFO] training head...

Epoch 1/3

376/376 [=====] - 90s 228ms/step - loss: 0.9651 - accuracy: 0.8948 - val_loss: 0.1823 - val_accuracy: 0.9679

Epoch 2/3

376/376 [=====] - 89s 228ms/step - loss: 0.2996 - accuracy: 0.9528 - val_loss: 0.1225 - val_accuracy: 0.9796

Epoch 3/3

376/376 [=====] - 88s 225ms/step - loss: 0.2588 - accuracy: 0.9601 - val_loss: 0.1312 - val_accuracy: 0.9808

CONSTRUÇÃO

Transfer Learning - VGG 16 [Fine-tuning | STAGE 02]

Layer (type)	Output Shape	Param #
input_5 (InputLayer)	[(None, 224, 224, 3)]	0
sequential (Sequential)	(None, 224, 224, 3)	0
tf.__operators__.getitem_1 (SlicingOpLambda)	(None, 224, 224, 3)	0
tf.nn.bias_add_1 (TFOpLambda)	(None, 224, 224, 3)	0
vgg16 (Functional)	(None, 7, 7, 512)	14714688
flatten (Flatten)	(None, 25088)	0
dense_1 (Dense)	(None, 256)	6422784
dropout_1 (Dropout)	(None, 256)	0
dense_2 (Dense)	(None, 3)	771

=====

Total params: 21,138,243
Trainable params: 6,423,555
Non-trainable params: 14,714,688

Unfreeze

Layer (type)	Output Shape	Param #
input_5 (InputLayer)	[(None, 224, 224, 3)]	0
sequential (Sequential)	(None, 224, 224, 3)	0
tf.__operators__.getitem_1 (SlicingOpLambda)	(None, 224, 224, 3)	0
tf.nn.bias_add_1 (TFOpLambda)	(None, 224, 224, 3)	0
vgg16 (Functional)	(None, 7, 7, 512)	14714688
flatten (Flatten)	(None, 25088)	0
dense_1 (Dense)	(None, 256)	6422784
dropout_1 (Dropout)	(None, 256)	0
dense_2 (Dense)	(None, 3)	771

=====

Total params: 21,138,243
Trainable params: 13,502,979
Non-trainable params: 7,635,264

CONSTRUÇÃO

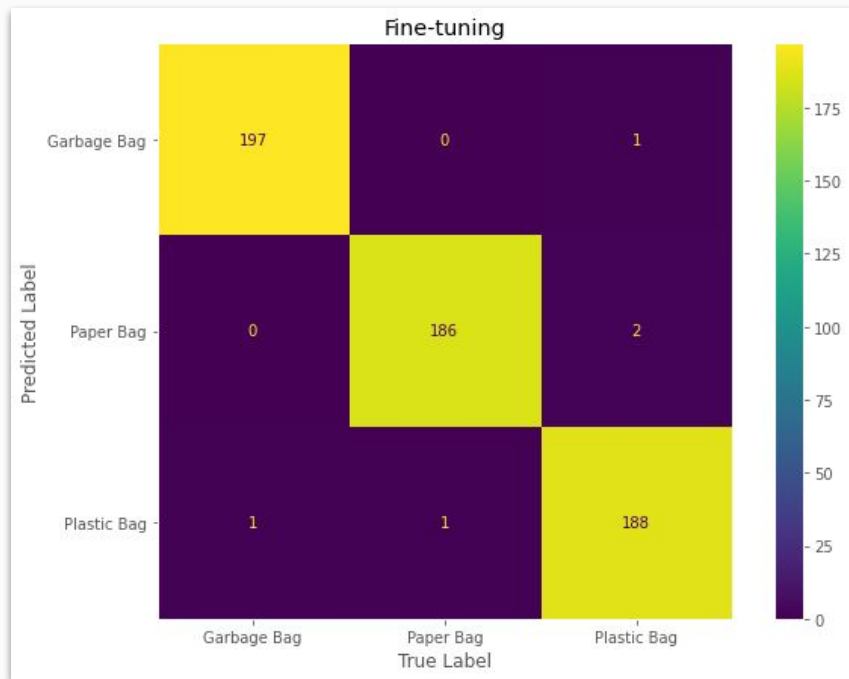
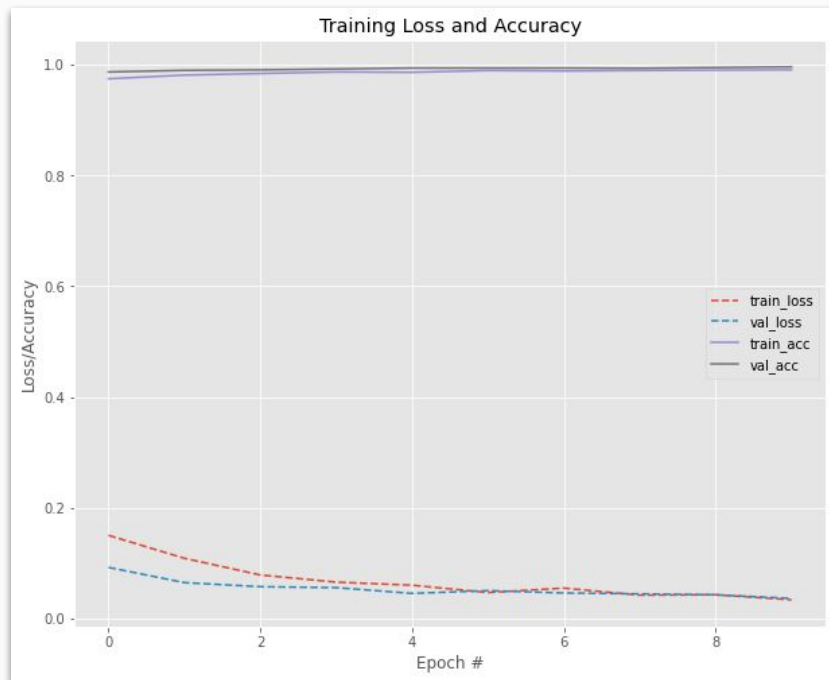
Transfer Learning - VGG 16 [Fine-tuning | STAGE 02]

```
print("[INFO] re-compiling model...")
opt = SGD(learning_rate=0.0001)
model.compile(loss="categorical_crossentropy",
              optimizer=opt,
              metrics=["accuracy"])
```

```
history = model.fit(train_dataset,
                    validation_data=validation_dataset,
                    batch_size=32,
                    epochs=10,
                    verbose=1)
```

CONSTRUÇÃO

Transfer Learning - VGG 16 [Fine-tuning | STAGE 02]



CONSTRUÇÃO

Transfer Learning - VGG 16 [Fine-tuning | STAGE 02]

	precision	recall	f1-score	support
Garbage Bag	0.99	0.99	0.99	198
Paper Bag	0.99	0.99	0.99	187
Plastic Bag	0.99	0.98	0.99	191
accuracy			0.99	576
macro avg	0.99	0.99	0.99	576
weighted avg	0.99	0.99	0.99	576

REFERÊNCIAS

VinceVence. **Plastic - Paper - Garbage Bag Synthetic Images**. Kaggle, 2022. Disponível em: <https://www.kaggle.com/datasets/vencerlanz09/plastic-paper-garbage-bag-synthetic-images/>. Acesso em: 04 dez. 2022.

MEDEIROS, Ivanovitch. **Embedded AI**. GitHub, 2022. Disponível em: <https://github.com/ivanovitchm/embedded.ai>. Acesso em: 04 dez. 2022.

MEDEIROS, Ivanovitch. **Deep Learning**. GitHub, 2022. Disponível em: <https://github.com/ivanovitchm/deeplearning>. Acesso em: 04 dez. 2022.

SOBRE O PLÁSTICO NO MEIO AMBIENTE. SENAC, 2022. Disponível em: <https://www.sp.senac.br/blog/artigo/plastico-no-meio-ambiente>. Acesso em: 04 dez. 2022.