Audience engagement detection – phase one - emotions detector

Project brief

Overview

A feedback loop is very important for educational and other types of activities, like lectures, public speeches, workshops, etc. With the current ML development level it's natural to monitor the engagement of the audience analysing different modalities of event recording – such as emotions, sight direction, pose, screen activity, etc. The first crucial subtask is to be able to detect emotions from frames (images).

Emotion detection using Facial Expression Recognition (FER) is a suitable approach for the problem above, and also it can be applied in various business cases to enhance customer experience, improve marketing strategies, and optimize human-computer interactions (see Use Cases section).

Dataset

FER dataset can be downloaded from Kaggle: https://www.kaggle.com/datasets/msambare/fer2013

It contains about 30k images of faces (48x48), labelled with seven dominating emotions: Angry, Disgust, Fear, Happy, Sad, Surprise, Neutral.

Also, there are datasets for multimodal emotions detection in video – for example, https://paperswithcode.com/dataset/crema-d . But this is for the second phase of the project.

Alternative solutions

An alternative solution can be done with a rule-based approach and face keypoint detection – like "top and bottom lip keypoints move in opposite directions and form approximate oval means surprise". But we are firmly convinced that now ML-based solutions should be more effective and robust.

Performance metrics

For the first stage standard metric, such as accuracy, is completely suitable.

Other information

- In the first stage, the system input should image with a face, for which the main emotion is detected. In the next stage – input should be video, where frames should be extracted with some interval, faces detected for each frame, and emotions detected for each face.
- 1st stage output will be the detected emotions. 2nd stage output will be a list of the next structure: *[time, [(face_id, emotion)]]*.
- For images the inference will be online. For videos – batch (customer can upload a video, and get a report)

- Technical system performance can be measured by collecting and analyzing model confidence (minimal, average, and median) for user-provided input. Business performance can be measured by users' feedback, users return, and churn rate (how many users continue to use the system after the first try, and how many users stop using the system after some time of active usage).
- The system will be hosted in the cloud
- There are no specific hardware requirements for inference
- We have potential ethical constraints here – as the data customers will send us contains video recordings or images with faces. For more details on constraints, see the corresponding section below.

FER use cases

Here are a few potential business cases for utilizing emotion detection with the FER dataset:

1. Customer Sentiment Analysis: Emotion detection can be used to analyze customer sentiments in real-time. By analyzing facial expressions during customer interactions or while reviewing products/services, businesses can gain insights into customer satisfaction levels, identify pain points, and address issues promptly.
2. Market Research: Emotion detection can aid in market research by capturing emotional responses during focus groups, surveys, or product testing. Analyzing facial expressions can provide valuable feedback on consumer preferences, perceptions, and emotional engagement, allowing businesses to refine their marketing strategies and product offerings accordingly.
3. Retail and E-commerce: Emotion detection can be applied in retail and e-commerce environments to gauge customer reactions to in-store displays, product packaging, or website interfaces. By tracking emotional responses, businesses can optimize store layouts, design captivating packaging, and create personalized online experiences to drive customer engagement and loyalty.

Regarding education, emotion detection can be applied for:

1. Student Engagement and Feedback: Emotion detection can help teachers gauge student engagement and monitor their emotional states during classroom activities. By analyzing facial expressions, teachers can identify when students are bored, confused, or disengaged, allowing them to adjust their teaching strategies accordingly. Real-time feedback based on emotions can help create a more interactive and student-centered learning environment.
2. Adaptive Learning Systems: Emotion detection can be integrated into adaptive learning systems to personalize the learning experience for each student. By tracking emotions, the system can dynamically adjust the difficulty level of assignments, provide additional support when students are struggling, or offer challenges when they are engaged and motivated. This personalized approach can enhance student motivation, performance, and overall learning outcomes.
3. Special Needs Education: Emotion detection can be particularly beneficial in special needs education. For example, it can assist in understanding the emotions and responses of students with autism spectrum disorders. Teachers and caregivers can use the data from emotion detection to identify triggers, develop tailored intervention strategies, and provide individualized support to improve emotional regulation and social interactions.
4. Online Learning Platforms: With the rise of online learning, emotion detection can help address the challenges of remote education. By analyzing students' facial expressions during virtual classes or recorded lectures, educators can gain insights into students' emotional responses and adapt their teaching methods accordingly. This can help mitigate feelings of isolation, enhance engagement, and foster a sense of connection and support in online learning environments.
5. Assessment and Feedback: Emotion detection can provide valuable insights during assessments and exams. By monitoring students' emotional states, educators can identify when students are

experiencing anxiety, stress, or other negative emotions that may impact their performance. This information can inform targeted interventions, such as providing additional support, implementing relaxation techniques, or adjusting assessment strategies to create a more conducive learning environment.

Constraints in detail

1. Data Availability:
    - Constraint: Limited availability of diverse and high-quality facial expression data for training and validation purposes. The FER dataset may not cover all demographic groups or capture a wide range of emotions.
    - Assumption: Sufficient labeled data is available to train and fine-tune the emotion detection model using the FER dataset.
2. Environment, Resources, Security, Premises:
    - Constraint: Limited computational resources (e.g., processing power, memory) may affect the speed and scalability of emotion detection algorithms.
    - Constraint: Limited physical space or infrastructure to deploy the necessary hardware and software components for real-time emotion detection.
    - Assumption: Adequate security measures are in place to protect the collected facial expression data from unauthorized access or breaches.
3. Performance - Latency Constraints:
    - Constraint: Real-time emotion detection requires low latency, which can be challenging to achieve due to processing and communication delays.
    - Constraint: Inadequate network connectivity or bandwidth limitations may impact the speed and responsiveness of the emotion detection system.
    - Assumption: The emotion detection model can be optimized to achieve acceptable latency within the given hardware and network constraints.
4. Performance - Hardware Constraints:
    - Constraint: Limited availability or compatibility of hardware resources (e.g., GPUs) required to accelerate the computation-intensive tasks involved in emotion detection.
    - Assumption: Sufficient computational resources, such as GPUs or dedicated hardware accelerators, are available to achieve real-time or near real-time emotion detection.

Technical Design Architecture

1. Solution Architecture and Components:

The solution architecture for audience engagement detection and emotions detection consists of the following components:

a. Data Storage: This component is responsible for storing the FER dataset, including the 30k images of faces labeled with seven dominant emotions. It can be stored in a cloud-based storage solution like Amazon S3 or Google Cloud Storage. In our project we have uploaded them on S3 in order to access to them faster and easier.

b. Model Registry: The model registry component is used to manage different versions of the emotion detection models. It provides version control, model metadata, and allows for easy deployment and retrieval of trained models.

c. ML Pipelines: ML pipelines are responsible for automating the end-to-end workflow of training, evaluating, and deploying machine learning models. They encompass data preprocessing, feature engineering, model training, hyperparameter tuning.

d. Data Pipeline for Processing and Feature Engineering: This component handles the processing and feature engineering tasks required for emotion detection. It involves tasks such as face detection, face alignment, and extracting facial features from the images.

e. Model Deployment and Hosting Concept: The models can be deployed in a real-time, batch, or hybrid manner based on the specific requirements. For real-time deployment, a web service or RESTful API can be used to serve predictions in real-time. For batch deployment, models can be executed periodically or on-demand to process a batch of images or videos. Hybrid deployment combines both real-time and batch processing to cater to different use cases.

f. Client Applications Accessing Predictions: The client applications can access predictions using APIs, batch processing, or messaging architectures. For real-time predictions, the client applications can send requests to the deployed API endpoint and receive responses with emotion predictions. For batch processing, client applications can upload a video or a set of images, and the predictions can be processed offline and delivered to the client. Messaging architectures like Apache Kafka can be used for asynchronous communication between the client and the prediction system.

2. Pipeline Explanation:

The pipeline for audience engagement detection and emotions detection can be divided into the following stages:

Stage 1: Data Preprocessing and Feature Engineering

Input: FER dataset containing images of faces labeled with emotions

Processing: Face detection, face alignment, and extraction of facial features

Output: Processed data with facial features extracted for each image

Stage 2: Model Training and Evaluation

Input: Processed data with facial features

Processing: Creating a custom dataset -> load the dataset to dataloader.

Training: Using resnet model to train all the images with batch size = 128,

Output: Trained models with associated metrics and performance evaluation results

Stage 3: Model Deployment

Input: Trained models from the model registry

Processing: Deploying the models in a real-time, batch, or hybrid manner based on requirements

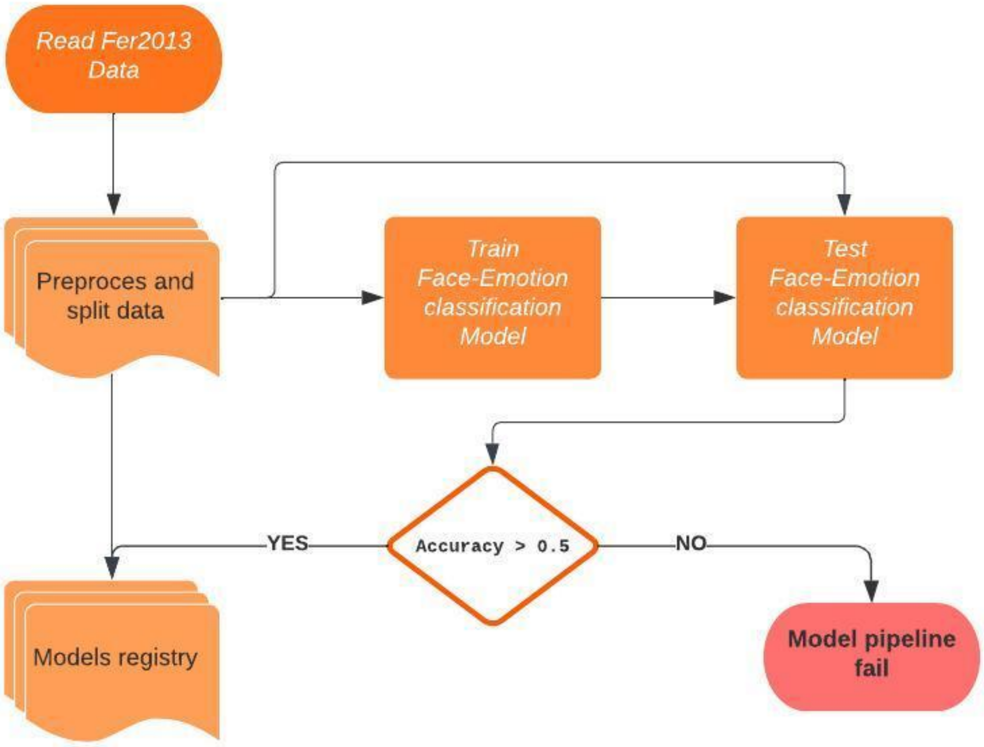
Output: Deployed models accessible for making predictions

Stage 4: Client Application Interaction

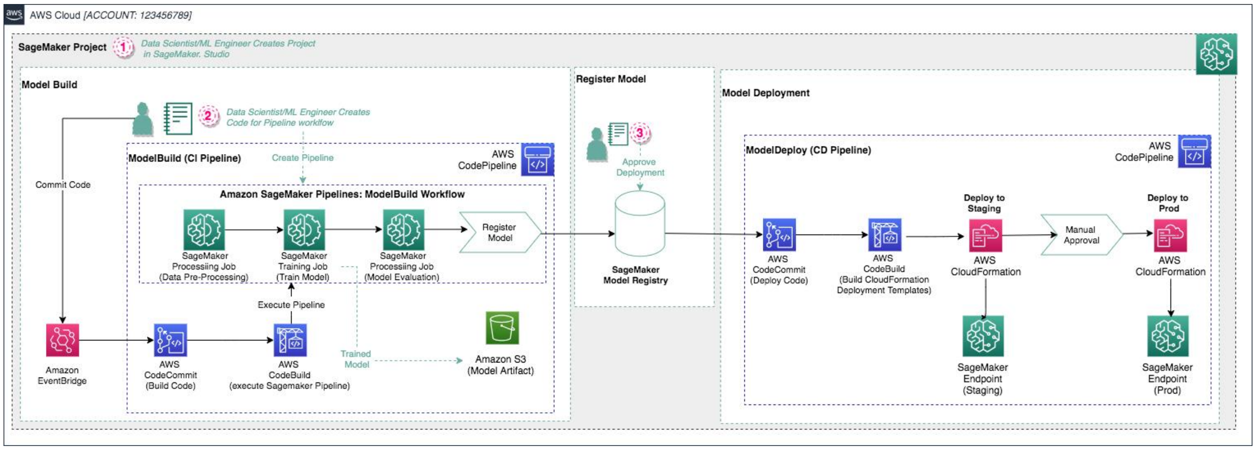
Input: Images or videos from client applications

Processing: Sending requests to the deployed model for prediction

Output: Emotion predictions returned to the client applications



Basic Pipeline Diagram


aws
AWS Cloud [ACCOUNT: 123456789]
SageMaker Project
1
Data Scientist/ML Engineer Creates Project in SageMaker. Studio
Model Build
2
Data Scientist/ML Engineer Creates Code for Pipeline workflow
Commit Code
ModelBuild (CI Pipeline)
Create Pipeline
AWS CodePipeline
Amazon SageMaker Pipelines: ModelBuild Workflow
SageMaker Processiing Job (Data Pre-Processing)
SageMaker Training Job (Train Model)
SageMaker Processiing Job (Model Evaluation)
Register Model
Execute Pipeline
Amazon EventBridge
AWS CodeCommit (Build Code)
AWS CodeBuild (execute Sagemaker Pipeline)
Trained Model
Amazon S3 (Model Artifact)
Register Model
3
Approve Deployment
SageMaker Model Registry
Model Deployment
ModelDeploy (CD Pipeline)
AWS CodePipeline
AWS CodeCommit (Deploy Code)
AWS CodeBuild (Build CloudFormation Deployment Templates)
Deploy to Staging
AWS CloudFormation
Manual Approval
Deploy to Prod
AWS CloudFormation
SageMaker Endpoint (Staging)
SageMaker Endpoint (Prod)

Classical pipeline diagram that we tried to reproduce using resNet18 (pytorch) and fer2013 dataset

3. Design Decisions and Cost Estimate:

a. Design Decision: Use of ML-based Solutions

The decision to use ML-based solutions, specifically Facial Expression Recognition (FER) models, was made because ML models have shown to be more effective and robust in detecting emotions compared to rule-based approaches. ML models have the ability to learn complex patterns and variations in facial expressions, making them more suitable for the task of emotion detection. Additionally, ML-based solutions can provide better generalization to handle a wide range of facial expressions and emotions.

b. Cost Estimate:

The cost components for the solution can be broadly categorized as follows:

i. Storage Costs: The storage costs will depend on the size of the FER dataset and any additional data generated during preprocessing.

ii. Compute Costs: The compute costs will depend on the complexity of the ML models, the number of training iterations, and the frequency of model deployment.

iii. Data Transfer Costs: If there is a need to transfer large amounts of data between components or between cloud regions, data transfer costs may be incurred. Cloud providers usually charge for data transfer based on the volume of data transferred and the network location.

4. Basic Security Concept:

To ensure data security and privacy, the following security measures can be implemented:

a. Data Access Control: Access to the FER dataset, trained models, and other sensitive data should be restricted to authorized personnel only. Role-based access control (RBAC) can be implemented to define and manage user roles and permissions.

b. PII Handling: If the dataset contains personally identifiable information (PII), it is crucial to handle it with care. Anonymization techniques can be applied to remove or encrypt PII before storing or processing the data. Additionally, compliance with relevant data protection regulations, such as GDPR, should be ensured.