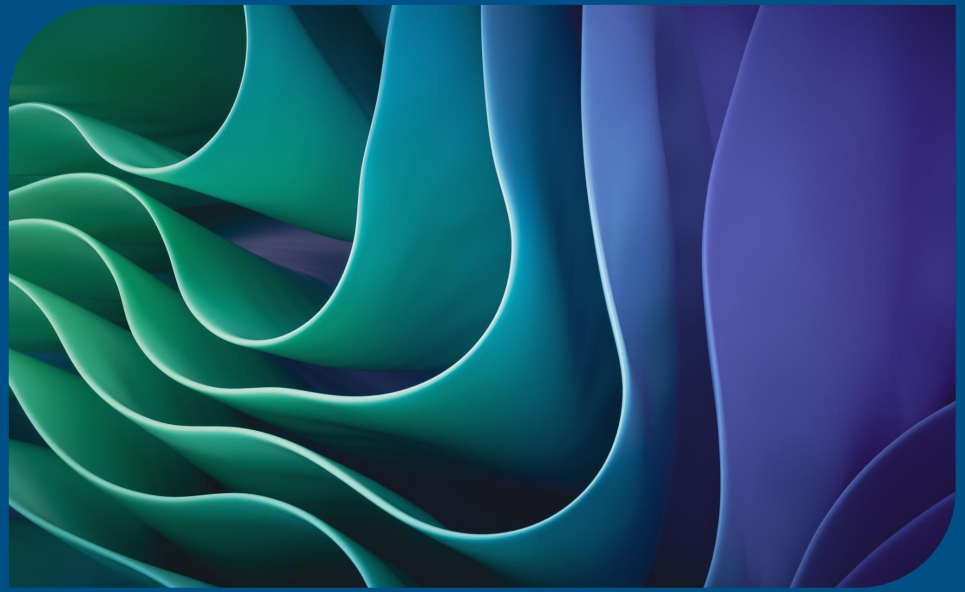
27.11.2025

Deep Learning Milestone 3

Group 11:
Kristian Ruth
Mauro Simoes
Leyi Qiang



Introduction

- **Insight Emotion:** Customer feedback management application, analyzing sentiment of feedback, trends, and discussion topics to provide value to businesses
- **Deep Learning component:** Sentiment analysis and classification, positive/negative, and emotion labels, helping to understand how customers are reacting to products
- **The model & data:** ELECTRA-Small discriminator, fine-tuned on Google's GoEmotions dataset

TODAY: Demo of UI, Analysis, and Dashboard; Docker configuration

Github Repo

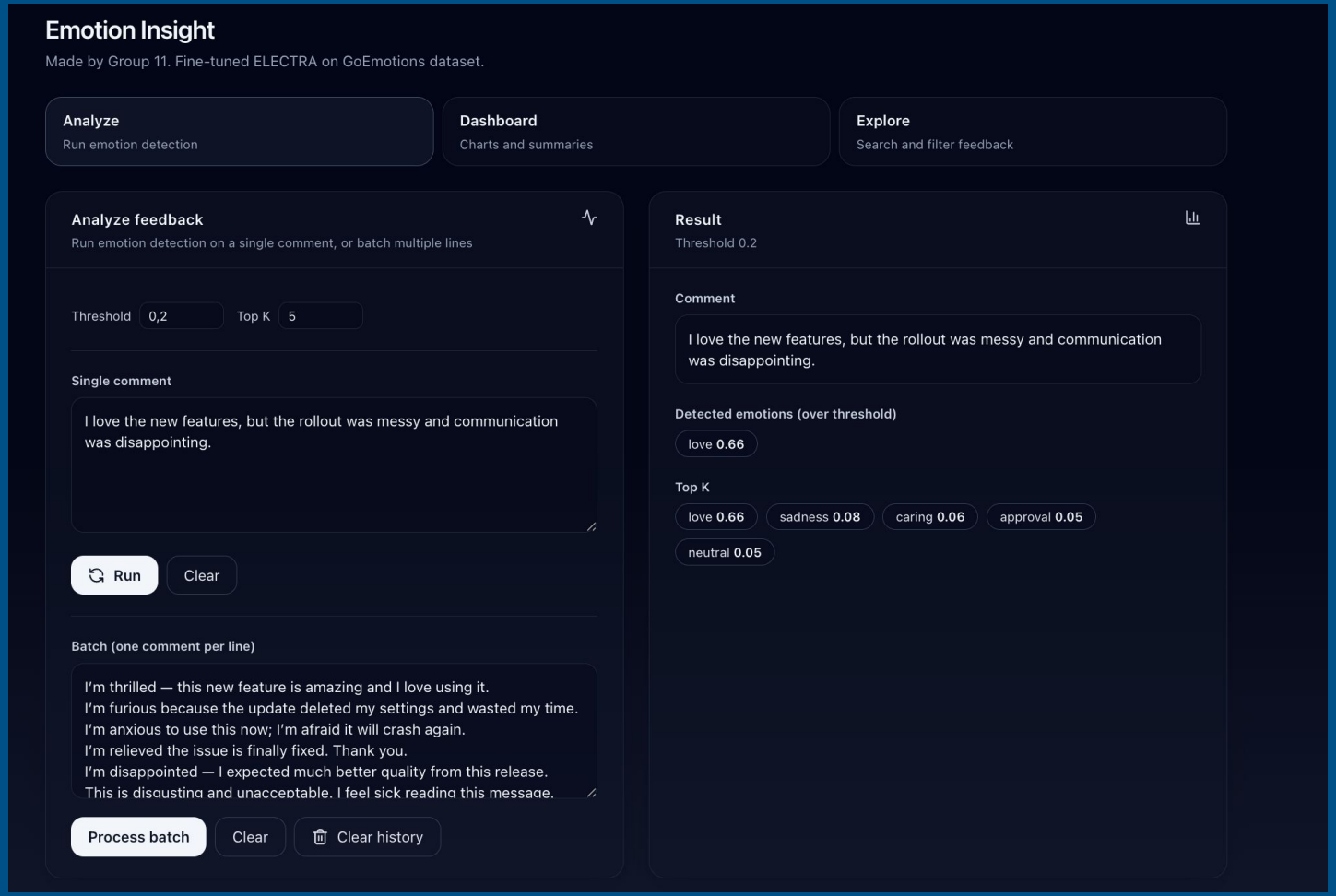
- https://github.com/leyiUPM/emotion/tree/main
- Three main components: web/ (Next.js frontend), backend/ (FastAPI with loaded model), and training script

Covered in `README.md`:

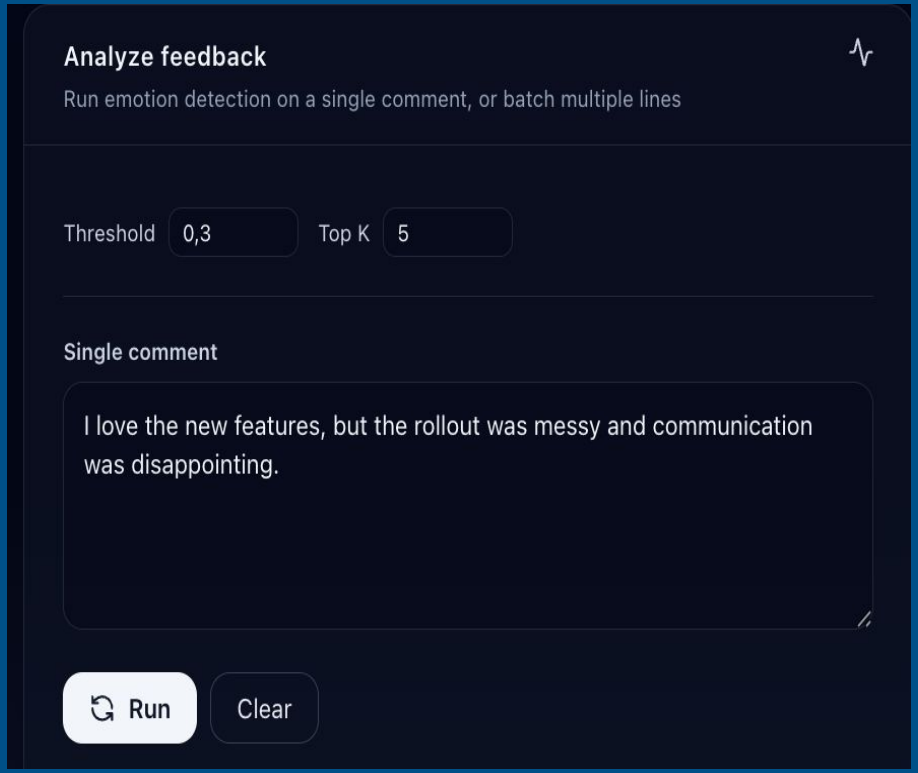
- **Training a model:** Run training script `run_and_export.py`
- **Running Docker:** `docker compose up --build`
- **Running locally:**
  - Backend:
    - `pip install -r requirements.txt`
    - `uvicorn app:app --reload --port 8000`
  - Frontend:
    - `npm install`
    - `npm run dev`

Demo

Layout

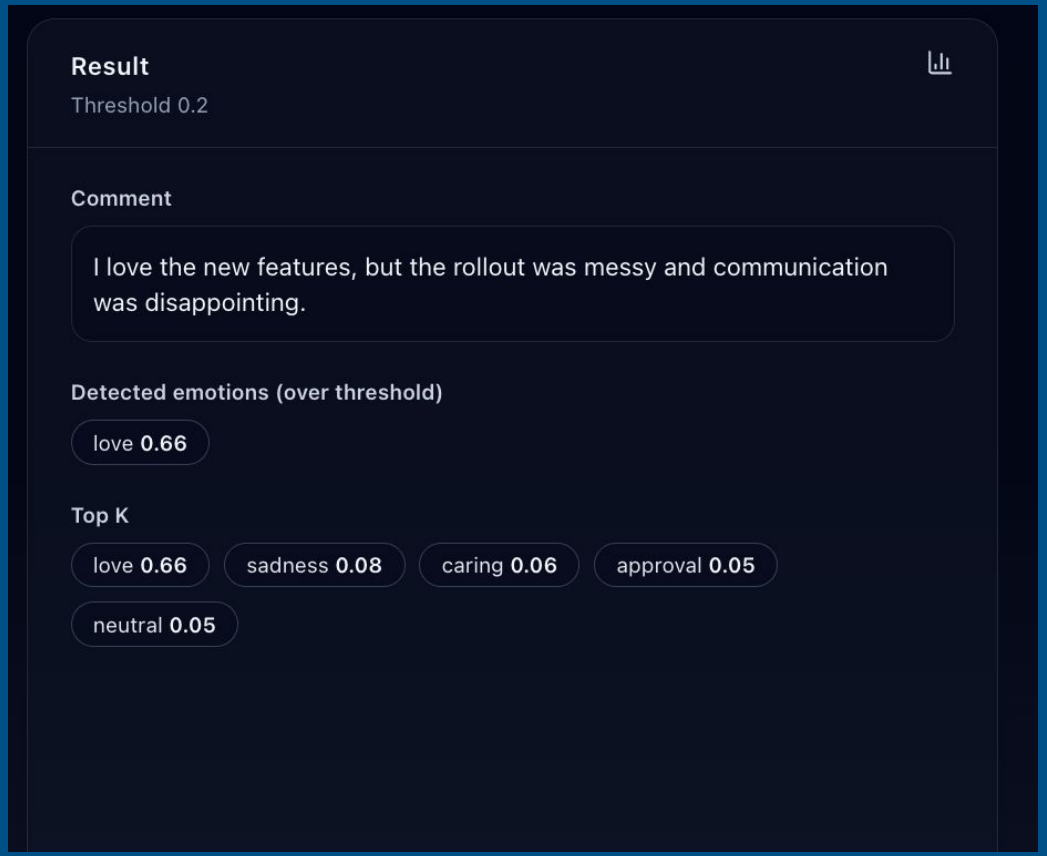

Emotion Insight
Made by Group 11. Fine-tuned ELECTRA on GoEmotions dataset.
Analyze
Run emotion detection
Dashboard
Charts and summaries
Explore
Search and filter feedback
Analyze feedback
Run emotion detection on a single comment, or batch multiple lines
Threshold 0,2
Top K 5
Single comment
I love the new features, but the rollout was messy and communication was disappointing.
Run
Clear
Batch (one comment per line)
I'm thrilled — this new feature is amazing and I love using it.
I'm furious because the update deleted my settings and wasted my time.
I'm anxious to use this now; I'm afraid it will crash again.
I'm relieved the issue is finally fixed. Thank you.
I'm disappointed — I expected much better quality from this release.
This is disgusting and unacceptable. I feel sick reading this message.
Process batch
Clear
Clear history
Result
Threshold 0.2
Comment
I love the new features, but the rollout was messy and communication was disappointing.
Detected emotions (over threshold)
love 0.66
Top K
love 0.66
sadness 0.08
caring 0.06
approval 0.05
neutral 0.05

Analysis


Analyze feedback
Run emotion detection on a single comment, or batch multiple lines
Threshold
0,3
Top K
5
Single comment
I love the new features, but the rollout was messy and communication was disappointing.
Run
Clear

Batch (one comment per line)
I'm thrilled — this new feature is amazing and I love using it.
I'm furious because the update deleted my settings and wasted my time.
I'm anxious to use this now; I'm afraid it will crash again.
I'm relieved the issue is finally fixed. Thank you.
I'm disappointed — I expected much better quality from this release.
This is disgusting and unacceptable. I feel sick reading this message.
Process batch
Clear
Clear history

Analysis


Result
Threshold 0.2
Comment
I love the new features, but the rollout was messy and communication was disappointing.
Detected emotions (over threshold)
love 0.66
Top K
love 0.66
sadness 0.08
caring 0.06
approval 0.05
neutral 0.05

Dashboard

Analyze
Run emotion detection

Dashboard
Charts and summaries

Explore
Search and filter feedback

Total processed

62

Stored in local browser history

Most common emotions

sadness, admiration

Emotion Over Threshold

79%

Percentages of labels over threshold in each run

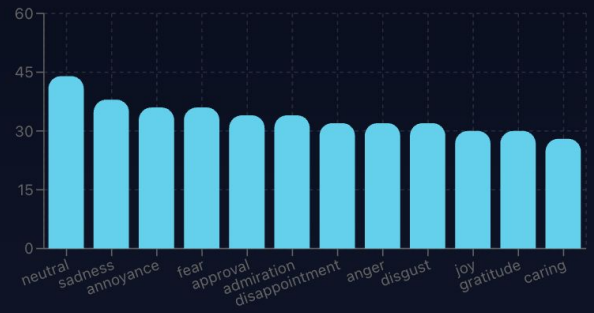
Avg emotions per comment

1.1

Count of labels over threshold

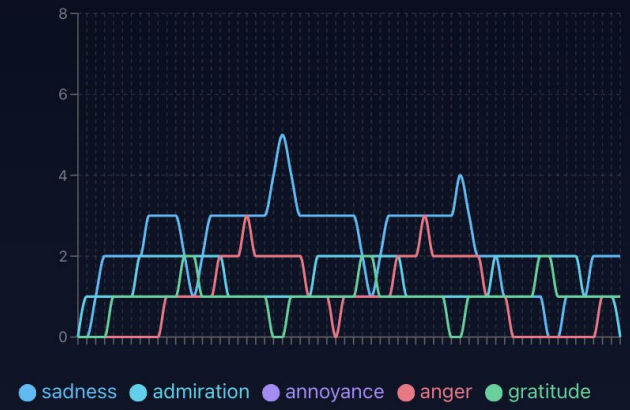
Top K Emotion distribution

Count of labels appearing in Top K

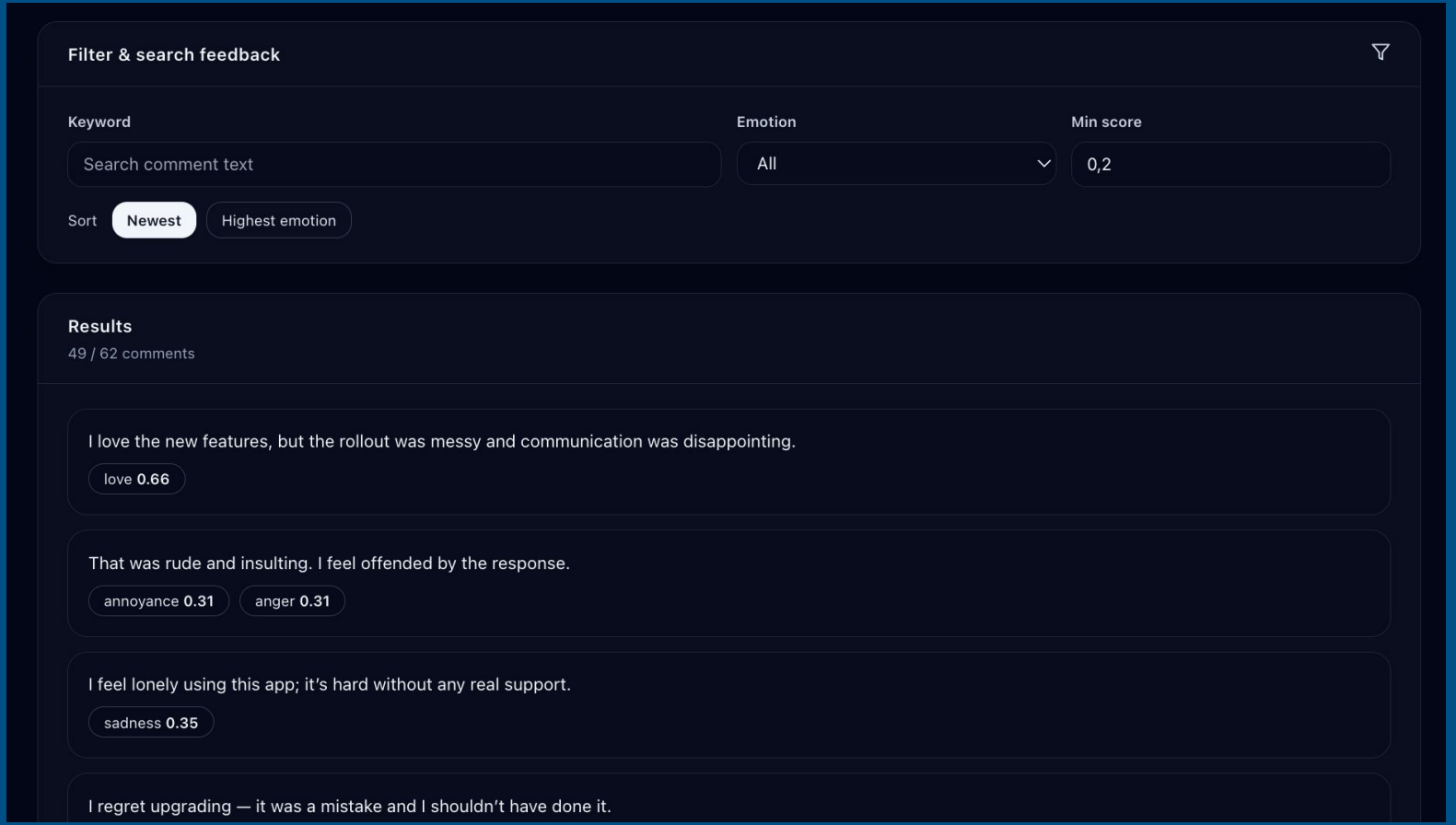


Emotion trend

Rolling window of detected emotions above threshold



Filter & Search



Docker

Frontend

```
# ---- deps ----
FROM node:20-alpine AS deps
WORKDIR /app
COPY package.json package-lock.json* ./
RUN npm ci

# ---- build ----
FROM node:20-alpine AS build
WORKDIR /app
COPY --from=deps /app/node_modules ./node_modules
COPY . .
RUN npm run build

# ---- run ----
FROM node:20-alpine AS run
WORKDIR /app
ENV NODE_ENV=production

# Next.js needs these at runtime
COPY --from=build /app/package.json ./package.json
COPY --from=build /app/next.config.mjs ./next.config.mjs
COPY --from=build /app/.next ./.next
COPY --from=build /app/node_modules ./node_modules

EXPOSE 3000
CMD ["npm", "run", "start", "--", "-p", "3000"]
```

We use a multi-stage build for the frontend:

- Smaller final image: Only includes production artifacts
- Layer caching : Faster rebuilds when dependencies don't change

Backend

```
FROM python:3.11-slim

# Avoid Python buffering so logs show up immediately
ENV PYTHONUNBUFFERED=1         PIP_NO_CACHE_DIR=1

WORKDIR /app

# Install deps
COPY requirements.txt /app/requirements.txt
RUN pip install --upgrade pip && pip install -r requirements.txt

# Copy app code and model artifacts
COPY app.py /app/app.py

EXPOSE 8000

# Run the API
CMD ["uvicorn", "app:app", "--host", "0.0.0.0", "--port", "8000"]
```

- Python:3.11-slim: Lightweight Python image (~150MB)
- PYTHONUNBUFFERED=1: Logs appear immediately (no buffering)
- PIP_NO_CACHE_DIR=1: Reduces image size by not caching pip packages
- Runs FastAPI with Uvicorn server

Compose

```yaml
services:
  backend:
    build:
      context: ./backend
    platform: linux/amd64
    ports:
      - "8000:8000"
    volumes:
      - ./backend/model:/app/model:ro

  web:
    build:
      context: ./web
    platform: linux/amd64
    environment:
      - MODEL_API_URL=http://backend:8000
    ports:
      - "3000:3000"
    depends_on:
      - backend
```

- Platform: linux/amd64 to ensure consistent builds
- Volumes: Mounts the ML model folder into the container (read-only bind mount)
- Depends_on: Ensures the backend starts before the web frontend
- Internal networking: Services communicate via http://backend:8000 (Docker's internal DNS)

Thank you!