Piotr Copek
Hyunseok Cho
Mateusz Znaleźniak
Szymon Molicki
Zuzanna Micorek

13.11.2025

Software Engineering

Project Card - Online Spam Detector

Index

Description

This project focuses on creating a web based service for detecting spam emails. Users can paste an email into a simple chat-like interface, and the system classifies it as Spam or Not Spam. The service can also provide an explanation showing which textual features influenced the decision, ensuring transparency and user understanding.

The main objective is to fine-tune a pretrained multipurpose language model for spam classification. Preparing training data, performing the fine-tuning process, results evaluation, and integrating the model into a functional backend.

Second objective is to develop a complete web service demonstrating the model in a real use case, including the frontend interface, backend API, and logic for returning classification results with explanations.

The service provides:

- Chat-like UI where users input the email text
- Backend model analyzing the message for spam indicators
- Clear label: Spam or Not Spam
- Optional detailed explanation, showing which features influenced the classification (keywords, structure, etc.)

System can be used for security education, email filtering assistance, or as a demo of machine learning classification.

Dictionary

- `Spam` - unrequested and potentially harmful or deceptive email
- `Classifier` - ML or rule-based component that determines if input text is spam
- `Explanation` Engine - Module that explains the model's decision
- `Frontend` - User interface
- `Backend` - Server handling classification logic
- `API` - Interface through which frontend sends email data for analysis.

Functional assumptions

- User can paste text into an input field
- System classifies the text as spam or not spam, nothing between
- System returns a message with the classification
- User can request an explanation of the classification
- System generates feature-based explanation (keywords, patterns, etc.)
- Frontend communicates with backend via a API

Non-Functional assumptions

- **Performance** — Response time under 3 seconds for classification
- **Scalability** — Ability to handle many users simultaneously
- **Security** — Emails sent by users are not stored permanently; secure transmission (HTTPS)
- **Usability** — Clear, intuitive interface similar to chat
- **Maintainability** — Modular backend structure allowing easy updates of the spam model
- **Reliability** — System should maintain high uptime
- **Explainability** — Explanations should be understandable for non-technical users
- **Availability** — Hosted on a standard server environment accessible 24/7

Milestones

- Project setup and repository creation
- UI mockups
- Basic classifier prototype
- Frontend – Backend integration
- Explanation engine
- Testing
- Deployment
- Final presentation

Tech Stack

- **Backend**: Python with Flask
- **Machine Learning**: Pretrained multipurpose language model fine-tuned for spam detection
- **Cloud Computing**: Remote machine for hosting + model execution
- **API Style**: REST API for communication between frontend and backend
- **Frontend**: HTML5 with CSS and JS
- **Deployment**: Cloud hosted server environment
- **Version Control**: GitHub for repository and code management

Roles

- Backend developer and project manager: **Piotr Copek**
- Frontend developer: **Hyunseok Cho**
- ML engineer: **Mateusz Znaleźniak** and **Szymon Molicki**
- QA: **Zuzanna Micorek**

Summary

This project delivers an online spam detection service that combines a fine-tuned language model with a simple web interface. Users can paste an email message to receive an instant classification along with an optional explanation of the detected indicators. The system includes Flask backend, cloud-hosted model, API, and lightweight frontend, demonstrating practical application of machine-learning–powered email filtering. The work reflects a complete software engineering process, from model preparation and implementation to deployment and presentation.