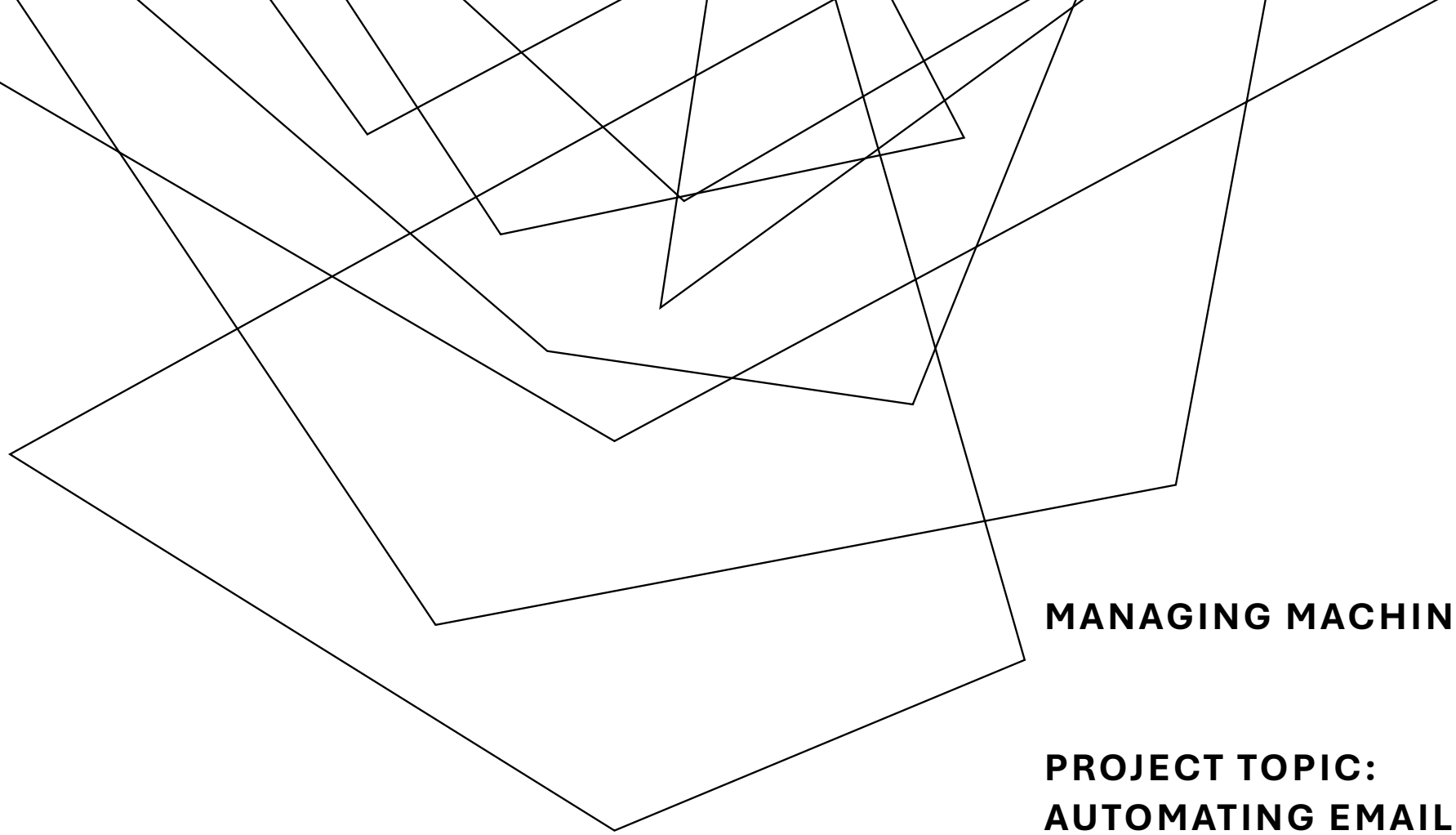


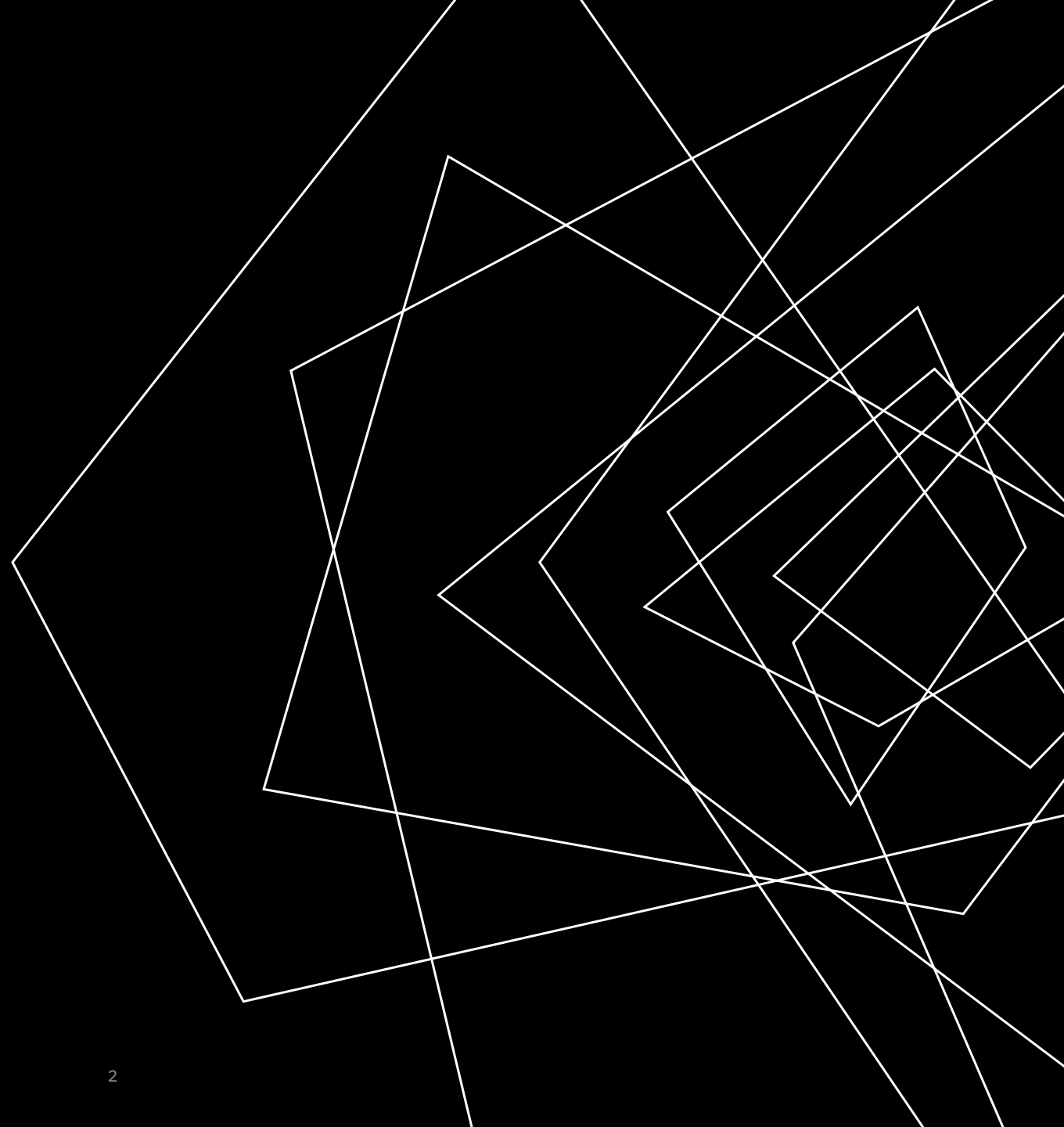
MANAGING MACHINE LERNING PROJECTS – DU

PROJECT TOPIC:
AUTOMATING EMAIL PRIORITIZATION IN CUSTOMER SUPPORT USING MACHINE LEARNING

by Ruken Zilan

AGENDA

- Opportunity evaluation
- CRISP-DM Business Understanding
- Solution validation plan
- ML system design
- Potential risks in production
- Summary



OPPORTUNITY EVALUATION

Problem: Customer support teams receive a **large volume of emails daily**, leading to delays in response times and inefficiencies in ticket prioritization.

Attractive Opportunity: Machine learning can **streamline email prioritization by automatically categorizing emails** based on urgency and relevance, improving response times and overall customer satisfaction.

CRISP-DM BUSINESS UNDERSTANDING

Problem Definition: Develop a machine learning model **to classify incoming support emails into priority levels.**

Success Metrics: Decrease in **response time** to critical emails, increase in **customer satisfaction scores**, and reduction in **manual effort** for email triage.

Factors: Email content, sender urgency, **keywords** indicating issue severity, historical ticket resolution times.

SOLUTION VALIDATION PLAN

Solution Concept: Utilize a **supervised learning approach** with features extracted from email content, sender urgency flags, and metadata.

Validation: Train the model on historical support ticket data, evaluate performance metrics such as accuracy and F1 score, and refine the model iteratively based on feedback from support agents.

SOLUTION CONCEPT TO APPLY THE SUPERVISED LEARNING APPROACH

- **Data Collection and Preprocessing:** Include features:
  - **Email content**: Subject lines, body text, attachments.
  - **Sender urgency flags**: Urgent or non-urgent designation.
  - **Metadata**: Sender information, timestamps, previous interactions with the support team.
- **Feature Extraction:** Process the raw email data to extract relevant features.
  - **Text preprocessing:** Tokenization, removing stopwords, stemming or lemmatization.
  - **Encoding sender urgency flags:** Convert binary urgency flags into numerical values.
  - **Extracting metadata:** Convert timestamps into numerical values, one-hot encode categorical features like sender information.

SOLUTION CONCEPT TO APPLY THE SUPERVISED LEARNING APPROACH

- **Splitting the Dataset:** Divide the labeled dataset into training, validation, and test sets. (80-10-10 split)
- **Model Selection and Training:** - **Logistic Regression is chosen since its** simple and interpretable, computationally efficient, and can serve as a good baseline model.
- **Model Evaluation** using appropriate evaluation metrics for multi-class classification, such as accuracy, precision, recall, and F1-score.
- **Testing and Fine-tuning:** evaluate the model on the test set to obtain an unbiased estimate of its performance.
- **Deployment and Monitoring**
- **Iterative Improvement**

SOLUTION VALIDATION STEPS

To validate the solution concept of the model follow a structured approach involving several steps:

1. User Interviews and Feedback Collection
2. Prototype Demonstration
3. User Testing Sessions
4. Quantitative Evaluation
    1. response time to high-priority emails,
    2. overall email handling efficiency,
    3. user satisfaction ratings.
5. Iterative Feedback Incorporation
6. Long-Term Pilot Deployment.

ML SUMMARY

Algorithm Selection

- **Logistic regression**.

Feature Engineering

- **Text preprocessing:** Tokenization, removing stopwords, stemming or lemmatization.
- **Encoding sender urgency flags:** Convert binary urgency flags into numerical values.
- **Extracting metadata**

Model Selection and Training

- **Logistic Regression is chosen since its** simple and interpretable, computationally efficient, and can serve as a good baseline model.

ML SYSTEM DESIGN

Cloud vs Edge

Selection: Cloud

Reasons:

- Cloud computing offers **scalability and computational power,** which are essential for processing **large volumes of email data and training complex machine learning models.**
- Email prioritization is not typically a real-time task that requires immediate decision-making at the edge. Instead, it involves **batch processing of historical data to train and evaluate the model,** making cloud-based solutions more suitable.
- **Maintenance and updates of the machine learning model can be centralized in the cloud,** allowing for easier management and deployment across the customer support team.

Online Learning vs Offline Learning

Selection: Offline Learning

•Reasons:

• Email prioritization in customer support can be performed offline, as **the model can be trained periodically using historical email data.**

• **Offline learning allows for batch processing of data**, enabling comprehensive model training on a complete dataset rather than incremental updates.

• Implementing **online learning would require continuous updates to the model based on real-time feedback, which may not be necessary** or practical for this use case.

Batch Predictions vs. Online Predictions

Selection: Batch Predictions

Reasons:

In customer support email **prioritization, predictions can be made in batches periodically** (e.g., hourly or daily) based on the latest data available.

Batch predictions allow for efficient processing of email data in bulk, enabling prioritization of multiple emails simultaneously.

Real-time predictions may not be critical for email prioritization, as the urgency of support tickets typically does not require immediate action at the granularity of individual emails.

POTENTIAL RISKS IN PRODUCTION

Training-serving skew

Changes in email patterns or support processes may impact the model's performance over time.

Latency

Processing large volumes of emails in real-time may lead to latency issues if the model inference time is too long.

Data drift

Shifts in customer behavior or support ticket characteristics could affect the model's ability to generalize to new data.

Model Bias

The model may inadvertently learn biases present in historical data, leading to unfair or inaccurate email prioritization decisions.

SUMMARY

By addressing these risks through **rigorous monitoring, model retraining, and regular updates** based on feedback from support agents, we can ensure the **effectiveness and fairness** of our machine learning-based email prioritization system in production.