**🟦 1. 开场（约30秒）**

Hello everyone, my name is Wei.  
Today I’ll be presenting our team’s project, an AI-driven tool to support software developers in addressing security challenges.  
I will briefly introduce the purpose and structure of our project, and then focus on my personal contribution to the topic modeling and question categorization part.

**🟦 2. 项目简介（约60秒）**

Our project aims to build a **security problem map** using Gemini AI and NLP techniques.  
Developers today face fragmented and poorly structured information about security. This makes it hard to identify what skills they need, or how to solve specific challenges efficiently.  
Our solution is to use natural language processing to extract and classify these real-world problems from Stack Overflow, and then organize them visually. To achieve that we generate challenges and skills of the stack overflow questions with Gemini; use LDA to group the questions, challenges and skills; use Bert to analyze the sentiment of stack overflow and Gemini generated answers; use TfidfVectorizer to calculate cosine similarity between each answer pair; and use LIWC to compare linguistic features of the answers.  
The project improves learning efficiency and helps developers systematically build security knowledge.

**🟦 3. 我的任务简介：Topic Categorization & Modeling（约60秒）**

My role in this project focused on **structuring the security questions** using topic modeling and manual refinement.  
I was responsible for turning unstructured developer posts into clean, categorized data.  
I combined automated topic modeling using LDA with a manual curation process.  
The idea was to let the algorithm discover hidden themes, then refine and merge those themes into useful, human-readable categories.  
This combination helped us maintain both scalability and clarity in the output.

**🟦 4. Workflow 技术流程介绍（约60秒）**

Here’s how the workflow went:

1. I loaded the full dataset into a pandas DataFrame.
2. Then applied NLP preprocessing – lowercasing, punctuation removal, stopword filtering, and lemmatization using NLTK.
3. After that, I used TfidfVectorizer to transform the text into a TF-IDF matrix.
4. Then I ran LDA with 20 components to find the latent topic distributions. Each question was then assigned to its most probable topic.  
   These outputs were exported for manual review and refinement.

**🟦 5. Manual Review & Final 8 Categories（约50秒）**

After running LDA, the model initially produced 20 topics. However, when I examined them closely, I noticed several issues.  
Many of the topics had **very similar keywords**, and their associated questions often overlapped in content.  
For example, two topics might both contain terms like “certificate,” “authentication,” and “token,” but the actual differences between them were minimal or confusing.  
If we were to keep all 20 as separate categories, it would likely **confuse users** rather than help them — they might not understand why two similar questions are in different groups.  
To solve this, I conducted a manual review of all 20 clusters.  
I looked at the top keywords, sampled multiple questions under each topic, and asked:  
“What is the **underlying technical theme** here?”  
Based on this, I **merged related topics** and created 8 final categories that are broader but also **more coherent**.  
Each of these categories is now aligned with real-world security areas, such as certificate handling, CSRF prevention, file encryption, and full-stack authentication flows.

**🟦 6. Reflection & Impact（约30秒）**

This task helped me understand how NLP and human judgment can work together to improve data clarity.  
The final structure we created enables:

* Better classification and visualization
* Clearer communication of developer needs
* And stronger alignment with real-world security concerns  
  Thank you for listening. That’s all for my part.

I used LDA because it’s an unsupervised machine learning tool and helps find hidden topics in a large number of questions without needing to label them first. It groups similar questions based on the words they use, so it's a good starting point.

Dataset grows to tens of thousands of entries----When topic traing Lda can become very slow. May cause overlapping. Manual review becomes harder. Large datasets may include more off-topic or low-quality entries.