# LDA Topic Modeling Workflow

This document provides a detailed overview of the Latent Dirichlet Allocation (LDA) topic modeling process we conducted as part of our project repository. The goal was to analyze patent abstracts related to cyberbullying and cyber safety in order to uncover latent thematic structures in the data.

## Step 1: Data Upload

We used Google Colab to upload the Excel dataset containing patent abstracts. The abstracts served as the input corpus for topic modeling.

A close-up of a computer screen

AI-generated content may be incorrect.

## Step 2: Install Dependencies

The following Python libraries were installed and used: pandas (data handling), openpyxl (Excel support), gensim (LDA modeling), nltk (text preprocessing), and pyLDAvis (visualization).



## Step 3: Data Loading

The Excel dataset was loaded into a pandas DataFrame, focusing specifically on the 'Abstract' column for further processing.

A computer screen shot of a computer code

AI-generated content may be incorrect.

## Step 4: Text Preprocessing

We applied a standard text preprocessing pipeline: lowercasing, removal of punctuation and non-words, tokenization, stopword removal, and lemmatization. This ensured consistent and meaningful tokens for topic modeling.

A screen shot of a computer program

AI-generated content may be incorrect.

## Step 5: Dictionary and Corpus Creation

Using Gensim, we created a dictionary mapping each token to an ID. We applied filtering (no\_below=2, no\_above=0.5) to remove rare and overly frequent terms. The corpus was then converted into a bag-of-words representation suitable for LDA.

A screenshot of a computer code

AI-generated content may be incorrect.

## Step 6: LDA Model Training

We trained LDA models with varying numbers of topics (k=4 to k=10). Model parameters included passes=20, chunksize=10, and alpha='auto'. The optimal number of topics was chosen based on coherence scores.

A screenshot of a computer program

AI-generated content may be incorrect.

## Step 7: Visualization

pyLDAvis was used to generate interactive visualizations. These allowed exploration of topic distributions, keyword importance, and topic overlap.

A white background with black text

AI-generated content may be incorrect.

## Step 8: Model Evaluation

Topic coherence was assessed using Gensim's CoherenceModel with c\_v metric. Scores ranged from ~0.31 to ~0.39, with k=9 yielding the highest coherence (~0.39). Although modest, these results are expected in short-text corpora such as patent abstracts.

## Insights Summary

Our topic analysis revealed a mix of highly relevant, moderately relevant, and outlier themes:  
A score of ~0.39 (k=9) is **modest but acceptable** in short-text corpora like patent abstracts. Patent language is often technical and vague, which can limit coherence scores. A value above 0.35 is considered informative, especially if topics make semantic sense.

Topic by topic Analysis

| **Topic** | **Top Keywords** | **Interpretation** | **Relevance** |
| --- | --- | --- | --- |
| **0** | application, access, web, information, request, dashboard | **Web-based software applications** managing user access or content — could include dashboards for moderation or parental controls | Possibly relevant to managing user interfaces or access for vulnerable groups |
| **1** | abuse, service, image, medium, processing, social | **Abuse detection in media/social services** — analyzing images or multimedia on platforms like social media | Highly relevant for cyberbullying, especially visual abuse |
| **2** | detection, video, network, control, url | **Network-based abuse/video content detection**, likely involving deep packet inspection or streaming video moderation | Relevant for online video/chat safety |
| **3** | content, input, module, message, virtual | **Content moderation systems**, modules processing virtual inputs (like messages or chats) | Important for chat and messaging safety, especially for kids |
| **4** | data, digital, event, threat, machine | **Cybersecurity / digital threat detection**, using AI to detect malicious activity in online events | Foundational to cyber safety infrastructure |
| **5** | device, user, response, call, communication | **Mobile or communication device protection**, maybe anti-phishing or spam control | Possibly relevant, depending on application context |
| **6** | model, vehicle, apparatus, motor, signal | **Autonomous or physical systems**, less aligned with cyber safety (likely about cars or IoT) | Not directly relevant |
| **7** | wearable, air, child, internet, designed | **IoT/wearable devices for children**, possibly air quality monitors or parental monitoring | Great connection to safety for children |
| **8** | server, security, unit, managing, database | **Back-end security infrastructure**, managing user data securely | Infrastructure-level safety, could support cyber safety systems |

**Insights Summary**

**High-Relevance Topics**

* **Topic 1**: Abuse in media and social systems
* **Topic 2**: Video/network-based content detection
* **Topic 3**: Chat/message filtering modules
* **Topic 4**: Digital threat & ML-based cybersecurity
* **Topic 7**: Wearables & IoT for kids

**Medium-Relevance Topics**

* **Topic 0**: Application/web dashboard design
* **Topic 5**: Mobile device abuse detection
* **Topic 8**: Secure server and backend management

**Low-Relevance/Outlier**

* **Topic 6**: Autonomous vehicles/sensors (likely out of scope)

A screenshot of a computer

AI-generated content may be incorrect.