Lesson Plan: Build a Mood2Emoji App

Introduction to Text Classification & Sentiment Analysis

📋 Lesson Overview

Topic: Build a Mood2Emoji App - Introduction to Text Classification

Target Age Group: 12-16 years old

Duration: 60 minutes

Level: Beginner (No prior coding experience required)

Format: Interactive demonstration + hands-on exploration

🎯 Learning Goals

By the end of this lesson, students will be able to:

1. **Understand Sentiment Analysis**

   - Define what sentiment analysis is

   - Explain how computers can detect emotions in text

   - Identify real-world applications of this technology

2. **Computational Thinking**

   - Break down the mood detection process into steps

   - Understand how polarity scores work

   - Recognize patterns in positive, negative, and neutral text

3. **Technical Skills**
   - Run a Python web application
   - Test and experiment with an AI tool
   - Identify limitations of AI systems

4. **Digital Literacy**
   - Understand the importance of content filtering
   - Recognize bias and limitations in AI
   - Think critically about technology

📚 Topics Introduced

Primary Topics
1. **Artificial Intelligence (AI) & Natural Language Processing (NLP)**
2. **Sentiment Analysis / Opinion Mining**
3. **Polarity Scores**
4. **Interactive Web Applications**
5. **Content Safety & Filtering**

Secondary Topics
- Python programming basics
- Library/package usage (TextBlob, Streamlit)
- Input-Process-Output model
- Conditional logic and thresholds
- User interface design

📖 Topics in Detail

1. What is Sentiment Analysis?

Definition: Sentiment analysis is a technique used by computers to determine the emotional tone behind words. It helps identify whether text expresses positive, negative, or neutral feelings.

Real-World Examples:

- **Social Media Monitoring:** Companies analyze tweets to see if people like their products

- **Customer Reviews:** Amazon/Yelp automatically categorize reviews as positive or negative

- **Email Filters:** Detecting spam or urgent messages

- **Mental Health Apps:** Monitoring mood through journal entries

Key Concept - Polarity:

```
Very Negative    Neutral    Very Positive
   -1.0          0.0          +1.0
    😞            😐            😃
```

Every sentence gets a score. Our app uses this scale:

- Score > 0.1 → Happy (😃)

- Score < -0.1 → Sad (😞)

- Score ≈ 0 → Neutral (😐)

---

2. How TextBlob Works (Simplified)

TextBlob is a Python library that makes sentiment analysis easy.

The Process:

1. **Break down the sentence** into words
2. **Look up each word** in a built-in dictionary
   - "love" → positive score
   - "hate" → negative score
   - "the" → neutral score
3. **Combine all scores** to get overall sentiment
4. **Return polarity** (-1.0 to +1.0) and subjectivity (0.0 to 1.0)

Example:

```
Input: "I love sunny days!"

Step 1: Split → ["I", "love", "sunny", "days"]
Step 2: Score → [0, +0.5, +0.4, 0]
Step 3: Average → +0.45
Step 4: Result → 😃 "Sounds happy!"
```

---

3. Building Interactive Apps with Streamlit

What is Streamlit?

- A Python library that turns scripts into web apps

- No HTML/CSS/JavaScript needed

- Perfect for data science and AI projects

Why It's Great for Learning:

- Instant visual feedback

- Easy to modify and experiment

- Professional-looking results quickly

---

4. Safety & Ethics in AI

Why Safety Matters:

- Apps for kids must filter inappropriate content

- AI can make mistakes (not perfect)

- We need rules to keep technology helpful and safe

Our Safety Features:

1. **Bad Word Filter:** Blocks inappropriate language

2. **Neutral Fallback:** Unknown text gets neutral response

3. **Age-Appropriate Messages:** All explanations are kid-friendly

4. **No Data Storage:** Privacy protection

🎓 60-Minute Lesson Plan

Phase 1: Hook & Introduction (10 minutes)

Activity: Interactive Demo

1. **Launch the App** (2 min)
   - Display Mood2Emoji on the projector
   - Show the simple, colorful interface

2. **Student Predictions** (3 min)
   - Ask: "What emoji should 'I aced my math test!' get?"
   - Students vote: 😃 😐 or 😞
   - Run the app and reveal the result
   - Discuss why it's happy

3. **Guided Exploration** (5 min)
   - Test these sentences together:
     - "I love ice cream" → 😃
     - "I lost my homework" → 😞
     - "The wall is gray" → 😐
   - Ask: "How do you think the computer knows?"

Key Question: *"Can computers actually understand feelings, or are they following rules?"*

Phase 2: Concept Deep-Dive (15 minutes)

Activity: Understanding the Technology

1. **Enable Teacher Mode** (5 min)
   - Show the "How It Works" diagram
   - Walk through the 6-step process
   - Explain each step in simple terms

2. **Polarity Score Exploration** (7 min)
   - Draw the -1.0 to +1.0 scale on the board
   - Test sentences and show their scores:
     - "Amazing!" → ~+0.6
     - "Terrible" → ~-0.8
     - "Okay" → ~0.0
   - Students predict scores for new sentences

3. **Word Power Discussion** (3 min)
   - Which words make text more positive? (love, great, awesome)
   - Which words make text more negative? (hate, bad, terrible)
   - What about neutral words? (the, is, and)

Worksheet Activity:

Students classify 10 sentences as happy, sad, or neutral BEFORE testing in the app.

Phase 3: Hands-On Exploration (20 minutes)

Activity: Student Experimentation

Setup: Students work in pairs with one computer

Challenge 1: Emoji Hunt (7 min)

- Find 3 sentences that get 😃

- Find 3 sentences that get 😞

- Find 3 sentences that get 😐

- Record polarity scores if Teacher Mode is on

Challenge 2: Edge Cases (6 min)

- What happens with very short text? ("Hi")

- What about long sentences?

- Can you trick the app? (Try sarcasm: "Oh great, more homework")

- Test the safety filter (use appropriate test words)

Challenge 3: Score Extremes (7 min)

- Can you get the highest positive score?

- Can you get the lowest negative score?

- What's the most neutral sentence possible?

Observation Sheet:

```

Sentence: _________________________________

Predicted Emoji: ___ Actual Emoji: ___
```

Polarity Score: _____

Was it correct? YES / NO

If wrong, why do you think?

Phase 4: Code Walkthrough (10 minutes)

Activity: Peek Behind the Curtain

1. **Open app.py** (2 min)

   - Show students the actual code

   - Explain: "This is what makes the app work!"

2. **Highlight Key Sections** (6 min)

   Section 1: The Safety Filter

   ```python

   BAD_WORDS = ['bad', 'stupid', 'hate']

   ```

   - Ask: "Why do we need this?"

   - Discuss: How to make it better?

   Section 2: The Brain (analyze_mood function)

   ```python

   blob = TextBlob(text)

   polarity = blob.sentiment.polarity
   ```

- This is where TextBlob does the magic!

Section 3: The Decision Maker

```python
if polarity > 0.1:
    emoji = "😃"
elif polarity < -0.1:
    emoji = "😞"
else:
    emoji = "😐"
```

- Simple if/else logic students can understand

3. **Q&A** (2 min)

- "What questions do you have?"

- "What would you change?"

Coding Concept Introduced:

- Variables (text, emoji, polarity)

- Functions (analyze_mood)

- Conditionals (if/elif/else)

- Libraries (import streamlit, textblob)

Phase 5: Wrap-Up & Extensions (5 minutes)

Activity: Reflect & Connect

1. **Real-World Applications** (2 min)

- Where have you seen this technology?

- How could your school use sentiment analysis?

- What about your favorite apps?

2. **Limitations Discussion** (2 min)

- What did the app get wrong?

- Why can't it understand sarcasm?

- Is AI actually "intelligent"?

3. **Take-Home Challenge** (1 min)

- **For Beginners:** Write 5 sentences at home and predict their moods

- **For Advanced Students:** Think of new emoji categories (😡 angry, 😳 surprised)

- **For Coders:** Try changing the polarity thresholds in the code

Exit Ticket Question:

"In one sentence, explain how computers detect mood in text."

📊 Learning Outcomes Assessment

Knowledge Checks During Lesson

Formative Assessment:

- [ ] Can students identify positive vs. negative words?

- [ ] Do students understand the polarity scale (-1 to +1)?

- [ ] Can students predict emoji outputs before testing?

- [ ] Do students recognize AI limitations?

Summative Assessment (End of Lesson):

Quiz Questions:

1. What is sentiment analysis used for? (Name 2 real-world examples)

2. Explain what a polarity score is.

3. Why do we need safety filters in kid-facing apps?

4. Name one limitation of the Mood2Emoji app.

Expected Answers:

1. Customer reviews, social media monitoring, email filtering, etc.

2. A number from -1 to +1 that shows if text is negative or positive

3. To block inappropriate content and keep kids safe

4. Can't understand sarcasm, only works in English, simple emotions, etc.

🎨 Learning Outcomes (Detailed)

Cognitive Outcomes

Knowledge (Remember & Understand):

- ✅ Define sentiment analysis

- ✅ List real-world applications of NLP

- ✅ Explain the purpose of polarity scores

- ✅ Identify positive and negative language

Application (Apply & Analyze):

- ✅ Predict emoji outputs based on text content
- ✅ Test edge cases and observe behavior
- ✅ Compare predicted vs. actual results
- ✅ Recognize patterns in polarity scores

Evaluation (Evaluate & Create):

- ✅ Critique the app's accuracy
- ✅ Identify limitations and biases
- ✅ Propose improvements to the system
- ✅ Think critically about AI in daily life

Technical Skills

- ✅ Run a Python application using terminal commands
- ✅ Navigate a Streamlit web interface
- ✅ Interpret numerical scores (polarity values)
- ✅ Read and understand basic Python code structure

Social-Emotional Skills

- ✅ Collaborate with peers during hands-on activities
- ✅ Communicate observations effectively
- ✅ Think critically about technology's role in society
- ✅ Develop digital literacy and safety awareness

🛠️ Materials Needed

Technology Requirements

- [ ] Computer with Python 3.9+ installed

- [ ] Projector/screen for demonstrations

- [ ] Student devices (1 per 2 students ideal)

- [ ] Internet connection (for initial library downloads)

Software Setup (Pre-class)

```bash
pip install streamlit textblob
python -m textblob.download_corpora
```

Handouts/Worksheets

- [ ] Observation sheet (sentence prediction table)

- [ ] Polarity scale visual aid

- [ ] Exit ticket questions

- [ ] Take-home challenge sheet

Optional Materials

- [ ] Whiteboard/markers for polarity scale diagram

- [ ] Sticky notes for quick student feedback

- [ ] Example sentence cards for group activities

🔄 Differentiation Strategies

For Struggling Students

- Provide sentence starters ("I feel ___")

- Use visual aids (emoji cards to match)

- Pair with stronger students

- Focus on recognition rather than creation

For Advanced Students

- Challenge them to "break" the app

- Introduce the code modification exercise

- Ask them to research other sentiment analysis tools

- Have them design additional features

For English Language Learners

- Provide sentence examples in advance

- Use more visual demonstrations

- Allow use of translation tools

- Focus on observation over explanation

📝 Extension Activities

Follow-Up Lessons

1. **Lesson 2:** Build a rule-based classifier (no library)

2. **Lesson 3:** Compare different sentiment analysis tools

3. **Lesson 4:** Create a mood journal tracker

4. **Lesson 5:** Explore bias in AI (why some words are scored certain ways)

Project Ideas

- Analyze mood of favorite book quotes

- Track class mood over a week

- Create custom emoji categories

- Design UI improvements

🎯 Alignment with Learning Standards

Computer Science Standards (CSTA K-12)

- **2-AP-13:** Decompose problems into smaller sub-problems

- **2-AP-17:** Systematically test and refine programs

- **3A-AP-17:** Use and adapt existing algorithms

Digital Literacy

- Understand how AI systems work

- Evaluate accuracy and limitations of technology

- Recognize importance of digital safety

Critical Thinking

- Analyze patterns and make predictions

- Evaluate evidence (Does the app work correctly?)

- Synthesize information from multiple sources

❓ Anticipated Questions & Answers

Q: "Is the computer actually feeling emotions?"

A: No! It's following mathematical rules. It looks up word scores and calculates an average. It doesn't "feel" anything.

Q: "Why does it get sarcasm wrong?"

A: Sarcasm is really hard! It requires understanding context, tone, and even cultural knowledge. Our simple app just looks at words in isolation.

Q: "Can I use this to analyze my text messages?"

A: You could! But remember, it's a simple tool and might not always be accurate. Real feelings are more complex than three emojis.

Q: "What if I want more emoji options?"

A: Great thinking! You could modify the code to add more categories. Want to try adding 😡 (angry) or 😲 (surprised)?

Q: "Is this really AI?"

A: It uses machine learning principles (trained on text data), but it's a simpler form of AI. More advanced systems use neural networks and deep learning.

📖 Teacher Preparation Guide

Before Class (30 min prep)

1. Run the app yourself and test 10-15 sentences

2. Identify some interesting edge cases to demonstrate

3. Print observation worksheets

4. Set up student computers with app pre-installed

5. Review the code so you can answer basic questions

During Class

- Keep energy high with interactive demos

- Encourage experimentation and "breaking" the app

- Validate wrong predictions as learning opportunities

- Use student examples whenever possible

After Class

- Collect observation sheets to assess understanding

- Note which concepts were confusing for future iterations

- Share student discoveries with the class (via email/LMS)

🌟 Success Criteria

This lesson is successful if students:

- ✅ Can explain sentiment analysis in their own words

- ✅ Actively engage with the app and test multiple sentences

- ✅ Identify at least one limitation of the technology

- ✅ Make connections to real-world applications

- ✅ Express curiosity about AI and want to learn more

📚 Additional Resources

For Teachers

- TextBlob Documentation: https://textblob.readthedocs.io/

- Streamlit Tutorials: https://docs.streamlit.io/

- AI4K12 Guidelines: https://ai4k12.org/

For Students

- "How Computers Learn" (video): [Link to age-appropriate explainer]

- Interactive NLP Demo: https://explosion.ai/demos/displacy

- Code.org AI Activities: https://code.org/ai

Extension Reading

- "Can Computers Understand Feelings?" (article)

- "The Ethics of AI" (simplified for teens)

Lesson Plan Created By: Akshita kumari

Date: November 4, 2025

Version: 1.0

Contact: akshitasingh3027@gmail.com

This lesson plan is designed to be flexible. Adapt timing and activities based on your students' needs and interests!