- **Quantitative Metrics?** Are there any such metrics to evaluate a model's training performance?

Quantitative Metrics are used to assess a model's performance. They provide an objective way to evaluate how well a model learns from data and achieves desired outcomes.

1. **loss functions:** show a measure of model performance and loss should decrease over time. For example, Cross-Entropy Loss for predicting the next note or word, If the Cross-Entropy Loss is low, it means the model's predicted probabilities are close to the true labels. KL Divergence for VAE's model, A lower KL Divergence means the model's learned distribution is close to the desired (target) distribution, indicating that the latent representations are well-structured and meaningful.
2. Prediction Accuracy: Measures how often the model gets things right for example predicting the next note or chord in a sequence. If the accuracy improves, it means the model is better at learning patterns in the data.
3. Perplexity: A metric commonly used for language models but applicable to music. It measures how uncertain the model is about its predictions. Lower perplexity means the model is more confident and accurate.
4. Diversity and Originality: **Entropy** measures how unpredictable the generated music is. **Uniqueness** checks how much the outputs differ from one another. **Temperature** can bring randomness to the model, offering more variety.
5. Consistency and Cohesion: Checks if the music stays in key, keeps a steady rhythm or follows harmonic progressions.

- **Musical Quality?** How can you determine if the generated music resembles Bach's Cello Suites, both in structure and stylistic elements?

  Musical Quality refers to the characteristics that make a piece of music appealing, meaningful, or effective.

  Check if the music has clear, balanced phrases, a hallmark of Bach's style. See if sections are repeated with slight variations, a key trait of Bach's compositions. Bach's music often features repeating rhythmic motifs with variations, checking for rhythmic patterns in the generated music. Test whether transitions between notes/chords follow Bach-like probabilities.

- **Beyond Bach?** Build a similar model on **any music/composer of your choice**.

  To Build a generative transformer model for generating music I'll choose Miles Davis who focuses on improvisation, and complex rhythms in jazz music.

  Step 1: Gather Training MIDI Files data: collect symbolic representations of the chosen composer's works or genre for example approximately 40 MIDI files data.

  Step 2: Preprocess the Data: Tokenize the Music into smaller chunks like notes, and chords and use libraries like **Music21** for symbolic data processing.

  Step 3: Create a **Transformer model: that is highly** effective for generating long sequences for example I'll use hugging face to get API and then use a based Music

Transformer for generating creative music.

- Number of parameters will be:
  - Number of layers: 6-12 (depending on dataset size).
  - Attention heads: 8.
  - Hidden size: 512-1024.
  - Dropout: 0.1-0.3.

Step 4: Use **Categorical Cross-Entropy Loss** to compare the predicted next token against the ground truth. Use the Adam optimizer. Create batches of token sequences with padding. Training loop with compute token predictions, compare predictions with actual next tokens, update weights using gradient descent.

Step 5: Generating Music: convert token sequences back into MIDI files using a tokenizer and use Muse Score Studio 4 to open generated music to evaluate the music quality.