Bing

Sure, I can help you with that. Once your model is trained, you can make predictions (or inferences) on new, unseen data. Here’s a simple way to do it:

```
# Assuming `new_data` is your new data on which you want to make predictions
new_features = tokenizer(new_data, truncation=True, padding=True, return_tensors='tf')

# Get the model's predictions
predictions = model(new_features)

# The predictions are logits. Use softmax to convert them into probabilities
probabilities = tf.nn.softmax(predictions.logits, axis=-1)

# Get the class with the highest probability for each example
predicted_classes = tf.argmax(probabilities, axis=-1)

# Now `predicted_classes` contains the predicted class for each example in `new_data`
```

In this code:

- `new_data` should be a list of sentences.
- `tokenizer` is the same tokenizer used during training.
- `model` is your trained model.

This will give you the predicted class for each sentence in `new_data`. Remember that the classes are represented as numbers, so you’ll need to map these numbers back to their corresponding class names. If you have any other questions, feel free to ask! 😊