```
# %%capture
# #!unzip Datasets.zip


# from google.colab import drive

# # Mount the Google Drive
# drive.mount('/content/drive')


# %%capture
# !pip install datasets
# !pip install transformers
# !pip install librosa
# !pip install jiwer
# !pip install evaluate


import os
import datasets
import pandas as pd
from sklearn.model_selection import train_test_split
from datasets import Dataset

# Set paths
csv_path = "/home/muzaffar/Desktop/Research/papers/5-paper Wav2Vec/5. Wave2vec Whisper Paper/KASHMIRI/experiment5/final.csv"
audio_folder = "/home/muzaffar/Desktop/Research/papers/5-paper Wav2Vec/5. Wave2vec Whisper Paper/KASHMIRI/experiment5/final-w

# Load the CSV
df = pd.read_csv(csv_path)
df = pd.read_csv(csv_path)
# Ensure the column names match
df.columns = ["Filename", "Transcription"]  # Rename columns if needed

# Append '.wav' to the file names
df['Filename'] = df['Filename'].apply(lambda x: f"{x}.wav")

# Add full paths to the audio files
df['file_path'] = df['Filename'].apply(lambda x: os.path.join(audio_folder, x))

# Verify that all audio files exist
missing_files = df[~df['file_path'].apply(os.path.exists)]
if not missing_files.empty:
    print("The following audio files are missing:")
    print(missing_files)
    raise FileNotFoundError("Some audio files listed in the CSV are missing in the folder.")

# Split into train (27) and test (3)
train_df, test_df = train_test_split(df, test_size=0.2, random_state=42)

# Save splits to CSV for reference
train_csv_path = "train_split.csv"
test_csv_path = "test_split.csv"
train_df.to_csv(train_csv_path, index=False)
test_df.to_csv(test_csv_path, index=False)

# Convert to HuggingFace Dataset format
train_dataset = Dataset.from_pandas(train_df)
test_dataset = Dataset.from_pandas(test_df)

# Save HuggingFace datasets
train_dataset_path = "train_dataset"
test_dataset_path = "test_dataset"
train_dataset.save_to_disk(train_dataset_path)
test_dataset.save_to_disk(test_dataset_path)

# Output
print(f"Train set saved to: {train_csv_path} and {train_dataset_path}")
print(f"Test set saved to: {test_csv_path} and {test_dataset_path}")
```

```
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/tqdm/auto.py:21: TqdmWarning: IProgress not found. Pleas
  from .autonotebook import tqdm as notebook_tqdm
Saving the dataset (1/1 shards): 100%|██████████| 2600/2600 [00:00<00:00, 586017.00 examples/s]
Saving the dataset (1/1 shards): 100%|██████████| 650/650 [00:00<00:00, 376144.81 examples/s]Train set saved to: train_s
Test set saved to: test_split.csv and test_dataset
```

```
from datasets import load_from_disk

train_dataset = load_from_disk("train_dataset")
test_dataset = load_from_disk("test_dataset")
```

```
print(train_dataset)
print(test_dataset)
```

```
Dataset({
    features: ['Filename', 'Transcription', 'file_path', '__index_level_0__'],
    num_rows: 2600
})
Dataset({
    features: ['Filename', 'Transcription', 'file_path', '__index_level_0__'],
    num_rows: 650
})
```

```
from datasets import ClassLabel
import random
import pandas as pd
from IPython.display import display, HTML

def show_random_elements(dataset, num_examples=10):
    assert num_examples <= len(dataset), "Can't pick more elements than there are in the dataset."
    picks = []
    for _ in range(num_examples):
        pick = random.randint(0, len(dataset)-1)
        while pick in picks:
            pick = random.randint(0, len(dataset)-1)
        picks.append(pick)

    df = pd.DataFrame(dataset[picks])
    display(HTML(df.to_html()))

show_random_elements(train_dataset)
```

	Filename	Transcription	file_path	__index_level_0__
0	8140118.wav	چانہٕ گاچ تتھ مکانس نہٕ أسی دارِ نہٕ بر	/home/muzaffar/Desktop/Research/papers/5-paper Wav2Vec/5. Wave2vec Whisper Paper/KASHMIRI/experiment5/final-waves/8140118.wav	117
1	rafiya-06_13.wav	فارسی مثنوی نگارو سا□مس مُعتلق وُنمُت چُھ تہٕ چُھ م□ در نظر	/home/muzaffar/Desktop/Research/papers/5-paper Wav2Vec/5. Wave2vec Whisper Paper/KASHMIRI/experiment5/final-waves/rafiya-06_13.wav	3196
2	8140072.wav	یوان لہذا یمِن ہٕنزِیہِ ترتیب قأیم تھاوِی چھے	/home/muzaffar/Desktop/Research/papers/5-paper Wav2Vec/5. Wave2vec Whisper Paper/KASHMIRI/experiment5/final-waves/8140072.wav	71
3	8150134.wav	کِہ رنگہ اکھ سیٹھاہ رُت اضافہ بلکہ ونہٕ	/home/muzaffar/Desktop/Research/papers/5-paper Wav2Vec/5. Wave2vec Whisper Paper/KASHMIRI/experiment5/final-waves/8150134.wav	530
4	farhat-01_16.wav	یُس کٔمۍ تام رازن یمِس عشوش ونان آسی بسوومُت	/home/muzaffar/Desktop/Research/papers/5-paper Wav2Vec/5. Wave2vec Whisper Paper/KASHMIRI/experiment5/final-waves/farhat-01_16.wav	732
5	8140165.wav	تیداد داہے ونمُت چھہ تہٕ چھُ واریاہ کم	/home/muzaffar/Desktop/Research/papers/5-paper Wav2Vec/5. Wave2vec Whisper Paper/KASHMIRI/experiment5/final-waves/8140165.wav	164
6	8140173.wav	تہٕ ہرگاہ م□ اکاڈمی اندر کانہہ رٕژ رٕیوایت	/home/muzaffar/Desktop/Research/papers/5-paper Wav2Vec/5. Wave2vec Whisper Paper/KASHMIRI/experiment5/final-waves/8140173.wav	172
7	jhon-02_104.wav	ہستی پرژھو ز توہہ نِش کتھ بو کتھ ہو چُھ از تام فأضَ حأصِل کورمُ	/home/muzaffar/Desktop/Research/papers/5-paper Wav2Vec/5. Wave2vec Whisper Paper/KASHMIRI/experiment5/final-waves/jhon-02_104.wav	2553
8	ishrat1-40_6.wav	بہٕ وۄتھرہیے اگر ژٕیکھ وۆزلی گُلاب ولو	/home/muzaffar/Desktop/Research/papers/5-paper Wav2Vec/5. Wave2vec Whisper Paper/KASHMIRI/experiment5/final-waves/ishrat1-40_6.wav	1773
9	guddy1-09_3.wav	گٹشمن ان□ہِ گٹس م□ز تہٕ چھہٕ زین انہِ گٹس م□ز تہٕ چھہ اتھ زینہٕ آسَی	/home/muzaffar/Desktop/Research/papers/5-paper Wav2Vec/5. Wave2vec Whisper Paper/KASHMIRI/experiment5/final-waves/guddy1-09_3.wav	1248

If there are any unwanted special characters in the dataset, we can remove them here, since there are none, I am keeping that as it is.

```
def extract_all_chars(batch):
  all_text = " ".join(batch["Transcription"])
  vocab = list(set(all_text))
  return {"vocab": [vocab], "all_text": [all_text]}

vocab_train = train_dataset.map(extract_all_chars, batched=True, batch_size=-1, keep_in_memory=True, remove_columns=train_da
vocab_test = test_dataset.map(extract_all_chars, batched=True, batch_size=-1, keep_in_memory=True, remove_columns=test_datas
```

```
Map: 100%|██████████| 2600/2600 [00:00<00:00, 184620.95 examples/s]
Map: 100%|██████████| 650/650 [00:00<00:00, 270171.20 examples/s]
```

```
vocab_list = list(set(vocab_train["vocab"][0]) | set(vocab_test["vocab"][0]))

vocab_dict = {v: k for k, v in enumerate(sorted(vocab_list))}
vocab_dict
```

Show hidden output

```
vocab_dict["|"] = vocab_dict[" "]
del vocab_dict[" "]
```

```
vocab_dict["[UNK]"] = len(vocab_dict)
vocab_dict["[PAD]"] = len(vocab_dict)
len(vocab_dict)
```

```
60
```

```
import json
with open('vocab.json', 'w') as vocab_file:
    json.dump(vocab_dict, vocab_file)
```

```
from transformers import Wav2Vec2CTCTokenizer

tokenizer = Wav2Vec2CTCTokenizer.from_pretrained("./", unk_token="[UNK]", pad_token="[PAD]", word_delimiter_token="|", clear
```

```
from transformers import Wav2Vec2FeatureExtractor

feature_extractor = Wav2Vec2FeatureExtractor(feature_size=1, sampling_rate=16000, padding_value=0.0, do_normalize=True, retu
```

```
from transformers import Wav2Vec2Processor

processor = Wav2Vec2Processor(feature_extractor=feature_extractor, tokenizer=tokenizer)
```

```
train_dataset[0]["file_path"]
```

```
'/home/muzaffar/Desktop/Research/papers/5-paper Wav2Vec/5. Wave2vec Whisper Paper/KASHMIRI/experiment5/final-
waves/farhat-03_21.wav'
```

Replacing the File Path with Actual Audio.

```
from datasets import load_from_disk, Audio

# Load datasets
train_dataset = load_from_disk("train_dataset")  # Adjust to your actual path
test_dataset = load_from_disk("test_dataset")

# Rename 'file_path' to 'audio'
train_dataset = train_dataset.rename_column("file_path", "audio")
test_dataset = test_dataset.rename_column("file_path", "audio")

# # Cast the 'audio' column to use the Audio feature
train_dataset = train_dataset.cast_column("audio", Audio(sampling_rate=16_000))
test_dataset = test_dataset.cast_column("audio", Audio(sampling_rate=16_000))

# # Drop unnecessary columns if needed
train_dataset = train_dataset.remove_columns(["__index_level_0__"])
test_dataset = test_dataset.remove_columns(["__index_level_0__"])

# # Verify the dataset structure
print(train_dataset)
print(test_dataset)

# # Inspect the first example
print(train_dataset[0])
```

```
Dataset({
    features: ['Filename', 'Transcription', 'audio'],
    num_rows: 2600
})
Dataset({
    features: ['Filename', 'Transcription', 'audio'],
    num_rows: 650
})
{'Filename': 'farhat-03_21.wav', 'Transcription': 'زأنۍ شاه صأبن بڈشابس\xa0مرا قبۍ پتۍ تۍ ییێلہ', 'audio': {'path': '/ho
        0.0256958 ,  0.02392578]), 'sampling_rate': 16000}}
```

```
#print(test_dataset[0]['audio'])
```

```
rand_int = random.randint(0, len(train_dataset))

print("Target text:", train_dataset[rand_int]["Transcription"])
print("Input array shape:", train_dataset[rand_int]["audio"]["array"].shape)
print("Sampling rate:", train_dataset[rand_int]["audio"]["sampling_rate"])
```

```
Target text: یہ میوُن کتّھ تہ بہ دوشوے چھہ اکھ أکس کھوژان
Input array shape: (64000,)
Sampling rate: 16000
```

```
def prepare_dataset(batch):
    audio = batch["audio"]

    # batched output is "un-batched"
    batch["input_values"] = processor(audio["array"], sampling_rate=audio["sampling_rate"]).input_values[0]
    batch["input_length"] = len(batch["input_values"])

    batch["labels"] = processor(text=batch["Transcription"]).input_ids

    return batch


train_dataset = train_dataset.map(prepare_dataset, remove_columns=train_dataset.column_names)
test_dataset = test_dataset.map(prepare_dataset, remove_columns=test_dataset.column_names)
```

```
Map:   0%|          | 0/2600 [00:00<?, ? examples/s]2025-04-05 00:09:12.277485: E external/local_xla/xla/stream_executor
WARNING: All log messages before absl::InitializeLog() is called are written to STDERR
E0000 00:00:1743791952.328213   12371 cuda_dnn.cc:8310] Unable to register cuDNN factory: Attempting to register factory
E0000 00:00:1743791952.343200   12371 cuda_blas.cc:1418] Unable to register cuBLAS factory: Attempting to register facto
2025-04-05 00:09:12.460245: I tensorflow/core/platform/cpu_feature_guard.cc:210] This TensorFlow binary is optimized to
To enable the following instructions: AVX2 FMA, in other operations, rebuild TensorFlow with the appropriate compiler fl
Map: 100%|██████████| 2600/2600 [00:05<00:00, 478.89 examples/s]
Map: 100%|██████████| 650/650 [00:00<00:00, 753.23 examples/s]
```

```
import torch

from dataclasses import dataclass, field
from typing import Any, Dict, List, Optional, Union

@dataclass
class DataCollatorCTCWithPadding:
    """
    Data collator that will dynamically pad the inputs received.
    Args:
        processor (:class:`~transformers.Wav2Vec2Processor`)
            The processor used for proccessing the data.
        padding (:obj:`bool`, :obj:`str` or :class:`~transformers.tokenization_utils_base.PaddingStrategy`, `optional`, defa
            Select a strategy to pad the returned sequences (according to the model's padding side and padding index)
            among:
            * :obj:`True` or :obj:`'longest'`: Pad to the longest sequence in the batch (or no padding if only a single
              sequence if provided).
            * :obj:`'max_length'`: Pad to a maximum length specified with the argument :obj:`max_length` or to the
              maximum acceptable input length for the model if that argument is not provided.
            * :obj:`False` or :obj:`'do_not_pad'` (default): No padding (i.e., can output a batch with sequences of
              different lengths).
    """

    processor: Wav2Vec2Processor
    padding: Union[bool, str] = True

    def __call__(self, features: List[Dict[str, Union[List[int], torch.Tensor]]]) -> Dict[str, torch.Tensor]:
        # split inputs and labels since they have to be of different lenghts and need
        # different padding methods
        input_features = [{"input_values": feature["input_values"]} for feature in features]
        label_features = [{"input_ids": feature["labels"]} for feature in features]

        batch = self.processor.pad(
            input_features,
            padding=self.padding,
            return_tensors="pt",
        )
        with self.processor.as_target_processor():
            labels_batch = self.processor.pad(
                label_features,
                padding=self.padding,
                return_tensors="pt",
            )

        # replace padding with -100 to ignore loss correctly
        labels = labels_batch["input_ids"].masked_fill(labels_batch.attention_mask.ne(1), -100)

        batch["labels"] = labels

        return batch


data_collator = DataCollatorCTCWithPadding(processor=processor, padding=True)
```

```
import evaluate

wer_metric = evaluate.load("wer")
```

```
Using the latest cached version of the module from /home/muzaffar/.cache/huggingface/modules/evaluate_modules/metrics/ev
```

```
from evaluate import load

cer_metric = load("cer")
```

```
Using the latest cached version of the module from /home/muzaffar/.cache/huggingface/modules/evaluate_modules/metrics/ev
```

includes both WER and CER

```
def compute_metrics(pred):
    pred_logits = pred.predictions
    pred_ids = np.argmax(pred_logits, axis=-1)

    # Replace padding token (-100) with pad_token_id
    pred.label_ids[pred.label_ids == -100] = processor.tokenizer.pad_token_id

    # Decode predictions and labels to strings
    pred_str = processor.batch_decode(pred_ids)
    label_str = processor.batch_decode(pred.label_ids, group_tokens=False)

    if isinstance(label_str, list):
        if isinstance(pred_str, list) and len(pred_str) == len(label_str):
            for index in random.sample(range(len(label_str)), 3):
                print(f'reference: "{label_str[index]}"')
                print(f'predicted: "{pred_str[index]}"')

        else:
            for index in random.sample(range(len(label_str)), 3):
                print(f'reference: "{label_str[index]}"')
                print(f'predicted: "{pred_str}"')

    # Compute WER
    wer = wer_metric.compute(predictions=pred_str, references=label_str)

    # Compute CER
    cer = cer_metric.compute(predictions=pred_str, references=label_str)

    return {"wer": wer, "cer": cer}
```

```
from transformers import Wav2Vec2ForCTC

model = Wav2Vec2ForCTC.from_pretrained(
    # "facebook/wav2vec2-xls-r-300m",
    'facebook/wav2vec2-large-xlsr-53',
    attention_dropout=0.05,
    hidden_dropout=0.1,
    feat_proj_dropout=0.1,
    mask_time_prob=0.05,
    layerdrop=0.01377,
    gradient_checkpointing=True,
    ctc_loss_reduction="mean",
    ctc_zero_infinity=True,
    pad_token_id=processor.tokenizer.pad_token_id,
    vocab_size=len(processor.tokenizer),


)
```

```
Some weights of Wav2Vec2ForCTC were not initialized from the model checkpoint at facebook/wav2vec2-large-xlsr-53 and are
You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inference.
```

```
# from transformers import Wav2Vec2ForCTC

# model = Wav2Vec2ForCTC.from_pretrained(
#     'facebook/wav2vec2-large-xlsr-53',
#     attention_dropout=0.05,
#     activation_dropout=0.1,
#     hidden_dropout=0.1,
#     feat_proj_dropout=0.01249,
#     final_dropout=0.0,
#     mask_time_prob=0.05,
```

```
#     mask_time_length=10,
#     mask_feature_prob=0,
#     mask_feature_length=10,
#     layerdrop=0.01377,
#     gradient_checkpointing=True,
#     ctc_loss_reduction="mean",
#     ctc_zero_infinity=True,
#     bos_token_id=processor.tokenizer.bos_token_id,
#     eos_token_id=processor.tokenizer.eos_token_id,
#     pad_token_id=processor.tokenizer.pad_token_id,
#     vocab_size=len(processor.tokenizer.get_vocab())
# )


model.freeze_feature_encoder()


# import huggingface_hub

# huggingface_hub.login()


#repo_name = "wav2vec2-kashmiri-jhon-data-one"


save_dir = "/home/muzaffar/Desktop/Research/papers/5-paper Wav2Vec/5. Wave2vec Whisper Paper/KASHMIRI/experiment5/training_a


from transformers import TrainingArguments

training_args = TrainingArguments(
    output_dir=save_dir,
    group_by_length=True,
    per_device_train_batch_size=8,
    per_device_eval_batch_size=8,
    gradient_accumulation_steps=2,
    evaluation_strategy="steps",
    num_train_epochs=30,
    fp16=True,
    save_steps=500,
    eval_steps=500,
    logging_steps=10,
    learning_rate=4e-4,
    warmup_steps=250,
    save_total_limit=2,
    dataloader_num_workers=24
)
```

```
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/training_args.py:1594: FutureWarning: `eval
  warnings.warn(
```

```
# from transformers import TrainingArguments

# training_args = TrainingArguments(
#   output_dir=repo_name,
#   group_by_length=True,
#   per_device_train_batch_size=2,
#   gradient_accumulation_steps=2,
#   eval_strategy="steps",
#   num_train_epochs=20,
#   gradient_checkpointing=True,
#   fp16=True,
#   save_steps=20,
#   eval_steps=20,
#   logging_steps=40,
#   learning_rate=3e-4,
#   warmup_steps=50,
#   save_total_limit=2,
#   push_to_hub=True,
# )


# import numpy as np
# from transformers import Trainer
# trainer = Trainer(
#     model=model,
#     data_collator=data_collator,
#     args=training_args,
#     compute_metrics=compute_metrics,
#     train_dataset=train_dataset,
#     eval_dataset=test_dataset,
#     tokenizer=processor.feature_extractor,
# )
```

```
import numpy as np
from transformers import Trainer

# Assuming processor is an instance of Wav2Vec2Processor (or similar for your model)
trainer = Trainer(
    model=model,
    data_collator=data_collator,
    args=training_args,
    compute_metrics=compute_metrics,
    train_dataset=train_dataset,
    eval_dataset=test_dataset,
    processing_class=processor,  # Use the processor directly for feature extraction
)
```

```
print("step1")
train_result = trainer.train()
print("step2")

metrics = train_result.metrics
print("step3")
max_train_samples = len(train_dataset)
metrics["train_samples"] = min(max_train_samples, len(train_dataset))
print("step4")
trainer.save_model()
print("model created!")
trainer.log_metrics("train", metrics)
trainer.save_metrics("train", metrics)
trainer.save_state()
```

```
step1
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/torch/utils/data/dataloader.py:624: UserWarning: This Da
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
```

[4860/4860 7:22:40, Epoch 29/30]

Step	Training Loss	Validation Loss	Wer	Cer
500	1.442300	1.118181	0.860588	0.301463
1000	0.524800	0.618262	0.568061	0.162705
2000	0.256300	0.688585	0.488946	0.139637
2500	0.149100	0.702022	0.482551	0.137082
3000	0.103100	0.754320	0.460442	0.130136
3500	0.093800	0.783033	0.455874	0.128321
4000	0.073100	0.813449	0.446008	0.125284
4500	0.072100	0.834218	0.446739	0.125423

```
trainer.evaluate()  # Evaluate the model on the test dataset
```

```
Trainer.tokenizer is now deprecated. You should use Trainer.processing_class instead.
Trainer.tokenizer is now deprecated. You should use Trainer.processing_class instead.
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/torch/utils/data/dataloader.py:624: UserWarning: This Da
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
```

```
# trainer.push_to_hub()
```

```
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
```

TESTING WITHOUT KENLM

```
import torch
import torchaudio
import librosa
import numpy
from transformers import Wav2Vec2ForCTC, Wav2Vec2Processor
from transformers import Wav2Vec2Processor
```

```
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
```

Start coding or generate with AI.

```
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
```

```
model_name_or_path = "/home/muzaffar/Desktop/Research/papers/5-paper Wav2Vec/5. Wave2vec Whisper Paper/KASHMIRI/experiment5/
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
print(model_name_or_path, device)

processor = Wav2Vec2Processor.from_pretrained(model_name_or_path)
model = Wav2Vec2ForCTC.from_pretrained(model_name_or_path).to(device)


def speech_file_to_array_fn(batch):
    speech_array, sampling_rate = torchaudio.load(batch["file_path"])
    speech_array = speech_array.squeeze().numpy()
    #speech_array = librosa.resample(np.asarray(speech_array), sampling_rate, processor.feature_extractor.sampling_rate)
    speech_array = librosa.resample(y=np.asarray(speech_array), orig_sr=sampling_rate, target_sr=processor.feature_extractor


    batch["speech"] = speech_array
    return batch


def predict(batch):
    features = processor(
        batch["speech"],
        sampling_rate=processor.feature_extractor.sampling_rate,
        return_tensors="pt",
        padding=True
    )

    input_values = features.input_values.to(device)
    #attention_mask = features.attention_mask.to(device)
    attention_mask = features.attention_mask.to(device) if "attention_mask" in features else None


    with torch.no_grad():
        logits = model(input_values, attention_mask=attention_mask).logits

    pred_ids = torch.argmax(logits, dim=-1)

    batch["predicted_N_LM"] = processor.batch_decode(pred_ids)
    return batch
```

```
import torchaudio
import librosa
from datasets import load_dataset
import numpy as np

dataset = load_dataset("csv", data_files={"/home/muzaffar/Desktop/Research/papers/5-paper Wav2Vec/5. Wave2vec Whisper Paper/

dataset = dataset.map(speech_file_to_array_fn)
```

```
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
```

```
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
/home/muzaffar/Desktop/Research/papers/5-paper Wav2Vec/5. Wave2vec Whisper Paper/KASHMIRI/experiment5/training_all_detai
  warnings.warn(
Map: 100%|██████████| 650/650 [00:03<00:00, 175.78 examples/s]
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
--------------------------------------------------------------------------------
  warnings.warn(
ArrowInvalid                              Traceback (most recent call last)
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
File ~/anaconda3/envs/tf14/lib/python3.11/site-packages/datasets/arrow_dataset.py:3541, in Dataset._map_single(shard,
  warnings.warn(
function, with_indices, with_rank, input_columns, batched, batch_size, drop_last_batch, remove_columns, keep_in_memory,
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
cache_file_name, writer_batch_size, features, disable_nullable, fn_kwargs, new_fingerprint, rank, offset)
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
   3540     if update_data and writer is not None:
  warnings.warn(
-> 3541         writer.finalize()  # close_stream=bool(buf_writer is None))  # We only close if we are writing in a
file
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
   3542 except (Exception, KeyboardInterrupt):
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
File ~/anaconda3/envs/tf14/lib/python3.11/site-packages/datasets/arrow_writer.py:636, in ArrowWriter.finalize(self,
  warnings.warn(
close_stream)
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
    635     self.hkey_record = []
  warnings.warn(
--> 636 self.write_examples_on_file()
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
    637 # If schema is known, infer features even if no examples were written
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
File ~/anaconda3/envs/tf14/lib/python3.11/site-packages/datasets/arrow_writer.py:495, in
  warnings.warn(
ArrowWriter.write_examples_on_file(self)
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
    491             batch_examples[col] = [
  warnings.warn(
    492                 row[0][col].to_pylist()[0] if isinstance(row[0][col], (pa.Array, pa.ChunkedArray)) else row[0][col]
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
    493                 for row in self.current_examples
  warnings.warn(
    494             ]
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
--> 495         self.write_batch(batch_examples=batch_examples)
  warnings.warn(
    496     self.current_examples = []
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
File ~/anaconda3/envs/tf14/lib/python3.11/site-packages/datasets/arrow_writer.py:605, in ArrowWriter.write_batch(self,
  warnings.warn(
batch_examples, writer_batch_size)
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
    604         typed_sequence = OptimizedTypedSequence(col_values, type=col_type, try_type=col_try_type, col=col)
--> 605         arrays.append(pa.array(typed_sequence))
  warnings.warn(
    606         inferred_features[col] = typed_sequence.get_inferred_type()
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
File ~/anaconda3/envs/tf14/lib/python3.11/site-packages/pyarrow/array.pxi:252, in pyarrow.lib.array()
Trainer.tokenizer is now deprecated. You should use Trainer.processing_class instead.
Trainer.tokenizer is now deprecated. You should use Trainer.processing_class instead.
File ~/anaconda3/envs/tf14/lib/python3.11/site-packages/pyarrow/array.pxi:114, in
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
pyarrow.lib._handle_arrow_array_protocol()
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
File ~/anaconda3/envs/tf14/lib/python3.11/site-packages/datasets/arrow_writer.py:225, in
  warnings.warn(
TypedSequence._arrow_array(self, type)
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
    224     if isinstance(data, list) and data and isinstance(first_non_null_value(data)[1], np.ndarray):
  warnings.warn(
--> 225         out = list_of_np_array_to_pyarrow_listarray(data)
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
    226     else:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
File ~/anaconda3/envs/tf14/lib/python3.11/site-packages/datasets/features/features.py:1538, in
  warnings.warn(
list_of_np_array_to_pyarrow_listarray(l_arr, type)
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
   1537     if len(l_arr) > 0:
  warnings.warn(
-> 1538         return list_of_pa_arrays_to_pyarrow_listarray(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
   1539             [numpy_to_pyarrow_listarray(arr, type=type) if arr is not None else None for arr in l_arr]
  warnings.warn(
   1540         )
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
   1541     else:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
File ~/anaconda3/envs/tf14/lib/python3.11/site-packages/datasets/features/features.py:1531, in
  warnings.warn(
list_of_pa_arrays_to_pyarrow_listarray(l_arr)
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
   1530     offsets = pa.array(offsets, type=pa.int32())
  warnings.warn(
-> 1531     values = pa.concat_arrays(l_arr)
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
   1532     return pa.ListArray.from_arrays(offsets, values)
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
File ~/anaconda3/envs/tf14/lib/python3.11/site-packages/pyarrow/array.pxi:4822, in pyarrow.lib.concat_arrays()
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
File ~/anaconda3/envs/tf14/lib/python3.11/site-packages/pyarrow/error.pxi:155, in
  warnings.warn(
pyarrow.lib.pyarrow_internal_check_status()
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
  warnings.warn(
File ~/anaconda3/envs/tf14/lib/python3.11/site-packages/pyarrow/error.pxi:92, in pyarrow.lib.check_status()
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
  warnings.warn(
ArrowInvalid: Arrays to be concatenated must be identically typed, but float and list<item: float> were encountered.
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
  warnings.warn(
During handling of the above exception, another exception occurred:
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
  warnings.warn(
ArrowInvalid                              Traceback (most recent call last)
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
Cell In[48], line 52
  warnings.warn(
    48 import numpy as np
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
    50 dataset = load_dataset("csv", data_files={"/home/muzaffar/Desktop/Research/papers/5-paper Wav2Vec/5. Wave2vec
  warnings.warn(
Whisper Paper/KASHMIRI/experiment5/test_split.csv"}, delimiter=",")
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
---> 52 dataset = dataset.map(speech_file_to_array_fn)
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
File ~/anaconda3/envs/tf14/lib/python3.11/site-packages/datasets/dataset_dict.py:901, in DatasetDict.map(self,
  warnings.warn(
function, with_indices, with_rank, input_columns, batched, batch_size, drop_last_batch, remove_columns, keep_in_memory,
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
load_from_cache_file, cache_file_names, writer_batch_size, features, disable_nullable, fn_kwargs, num_proc, desc)
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
    898     if cache_file_names is None:
    899         cache_file_names = {k: None for k in self}
  warnings.warn(
    900     return DatasetDict(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174: UserWarning:
--> 901         {
  warnings.warn(
    902             k: dataset.map(
reference: "باوو کۍ"
    903                 function=function,
predicte
    904                 with_indices=with_indices,
reference: "آتش صأبـن نیش چھ یـمہ"
    905                 with_rank=with_rank,
predicte
    906                 input_columns=input_columns,
reference: "اتش سأمـن نۍ"
    907                 batched=batched,
predicte
    908                 batch_size=batch_size,
/home/mu...packages/torch/utils/data/dataloader.py:624: UserWarning: This Da
```

```
    909                 drop_last_batch=drop_last_batch,
    910                 remove_columns=remove_columns,
    911                 keep_in_memory=keep_in_memory,
    912                 load_from_cache_file=load_from_cache_file,
    913                 cache_file_name=cache_file_names[k],
    914                 writer_batch_size=writer_batch_size,
    915                 features=features,
    916                 disable_nullable=disable_nullable,
    917                 fn_kwargs=fn_kwargs,
    918                 num_proc=num_proc,
    919                 desc=desc,
    920             )
    921             for k, dataset in self.items()
    922         }
    923     )

File ~/anaconda3/envs/tf14/lib/python3.11/site-packages/datasets/dataset_dict.py:902, in <dictcomp>(.0)
    898 if cache_file_names is None:
    899     cache_file_names = {k: None for k in self}
    900 return DatasetDict(
    901     {
--> 902         k: dataset.map(
    903             function=function,
    904             with_indices=with_indices,
    905             with_rank=with_rank,
    906             input_columns=input_columns,
    907             batched=batched,
    908             batch_size=batch_size,
    909             drop_last_batch=drop_last_batch,
    910             remove_columns=remove_columns,
    911             keep_in_memory=keep_in_memory,
    912             load_from_cache_file=load_from_cache_file,
    913             cache_file_name=cache_file_names[k],
    914             writer_batch_size=writer_batch_size,
    915             features=features,
    916             disable_nullable=disable_nullable,
    917             fn_kwargs=fn_kwargs,
    918             num_proc=num_proc,
    919             desc=desc,
    920         )
    921         for k, dataset in self.items()
    922     }
    923 )

File ~/anaconda3/envs/tf14/lib/python3.11/site-packages/datasets/arrow_dataset.py:562, in transmit_format.<locals>.wrapper(*args, **kwargs)
    555 self_format = {
    556     "type": self._format_type,
    557     "format_kwargs": self._format_kwargs,
    558     "columns": self._format_columns,
    559     "output_all_columns": self._output_all_columns,
    560 }
    561 # apply actual function
--> 562 out: Union["Dataset", "DatasetDict"] = func(self, *args, **kwargs)
    563 datasets: List["Dataset"] = list(out.values()) if isinstance(out, dict) else [out]
    564 # re-apply format to the output

File ~/anaconda3/envs/tf14/lib/python3.11/site-packages/datasets/arrow_dataset.py:3079, in Dataset.map(self, function, with_indices, with_rank, input_columns, batched, batch_size, drop_last_batch, remove_columns, keep_in_memory, load_from_cache_file, cache_file_name, writer_batch_size, features, disable_nullable, fn_kwargs, num_proc, suffix_template, new_fingerprint, desc)
   3073 if transformed_dataset is None:
   3074     with hf_tqdm(
   3075         unit=" examples",
   3076         total=pbar_total,
   3077         desc=desc or "Map",
   3078     ) as pbar:
-> 3079         for rank, done, content in Dataset._map_single(**dataset_kwargs):
   3080             if done:
   3081                 shards_done += 1

File ~/anaconda3/envs/tf14/lib/python3.11/site-packages/datasets/arrow_dataset.py:3546, in Dataset._map_single(shard, function, with_indices, with_rank, input_columns, batched, batch_size, drop_last_batch, remove_columns, keep_in_memory, cache_file_name, writer_batch_size, features, disable_nullable, fn_kwargs, new_fingerprint, rank, offset)
   3544 if update_data:
   3545     if writer is not None:
-> 3546         writer.finalize()
   3547     if tmp_file is not None:
   3548         tmp_file.close()

File ~/anaconda3/envs/tf14/lib/python3.11/site-packages/datasets/arrow_writer.py:636, in ArrowWriter.finalize(self, close_stream)
    634 # Re-intializing to empty list for next batch
    635 self.hkey_record = []
--> 636 self.write_examples_on_file()
    637 # If schema is known, infer features even if no examples were written
    638 if self.pa_writer is None and self.schema:

File ~/anaconda3/envs/tf14/lib/python3.11/site-packages/datasets/arrow_writer.py:495, in ArrowWriter.write_examples_on_file(self)
    490     else:
```

```
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
  warnings.warn(
/home/muzaffar/anaconda3/envs/tf14/lib/python3.11/site-packages/transformers/models/wav2vec2/processing_wav2vec2.py:174:
```