TOPIC MODELLING

In [6]:
```
#2
import gensim
from gensim import corpora
from gensim.models import LdaModel
import matplotlib.pyplot as plt

def load_documents(file_path):
    with open(file_path, 'r') as file:
        documents = file.readlines()
    return [doc.strip() for doc in documents]

def tokenize_documents(documents):
    return [doc.lower().split() for doc in documents]

file_path = "C:\Users\apurva shaw\Desktop\NLP\Smith claimed that Bumrah was ineff.txt"
documents = load_documents(file_path)

# Tokenize the documents
tokenized_documents = tokenize_documents(documents)

# Create a dictionary representation of the documents
dictionary = corpora.Dictionary(tokenized_documents)

# Convert the tokenized documents into a document-term matrix
corpus = [dictionary.doc2bow(doc) for doc in tokenized_documents]

# Build the LDA model
lda_model = LdaModel(corpus, id2word=dictionary, num_topics=5)

# Visualize the topics
topics = lda_model.print_topics(num_words=3)
for i, (topic_id, topic) in enumerate(topics):
    print(f"Topic {topic_id + 1}: {topic}")

# Visualize the topics using matplotlib
fig, ax = plt.subplots(figsize=(10, 6))  # Larger figure size for clearer visualization
for i in range(lda_model.num_topics):
    words = [word for word, _ in lda_model.show_topic(i, topn=3)]  # Extract words from the topic
    probabilities = [prob for _, prob in lda_model.show_topic(i, topn=3)]  # Extract probabilities
    ax.barh(f"Topic {i + 1}", probabilities, label=f"Topic {i + 1}")
    for j, word in enumerate(words):
        ax.text(probabilities[j], i, word, va="center")  # Add word labels
ax.set_xlabel("Probability")
ax.set_ylabel("Topics")
ax.legend()
plt.title("Top Words and Probabilities for Each Topic")
plt.tight_layout()  # Adjust layout to prevent clipping of labels
plt.show()
```

```
  Cell In[6], line 15
    file_path = "C:\Users\apurva shaw\Desktop\NLP\Smith claimed that Bumrah was ineff.txt"
                                                                                        ^
SyntaxError: (unicode error) 'unicodeescape' codec can't decode bytes in position 2-3: truncated \UXXXXXXXX escape
```

In [ ]: