In [ ]:
```
# Change to base directory
# Notebook does not recognize the modules for some reason
# ONLY RUN THIS CELL ONCE

os.chdir(os.path.normpath(os.getcwd() + os.sep + os.pardir))
os.getcwd()
```

Out[ ]:
```
'/Users/elizastarr/git/pttrns_interview'
```

Analyzing Predicted Captions with BLEU Scores

Bilingual evaluation understudy (BLEU) "is an algorithm for evaluating the quality of text which has been machine-translated from one natural language to another."

BLEU-n scores range between 0 and 1, 0 being a mismatch and 1 being a perfect match. For each image, we calculate the independent and cumulative BLEU scores (with a method 1 smoothing function) of all 5 reference captions to the predicted candidate caption.

BLEU Paper

In [ ]:
```
import os

import pandas as pd
pd.options.display.float_format = "{:,.2f}".format
%matplotlib inline

from src.analysis.bleu_scores import get_bleu_scores
from src.analysis.visualize import show_10_images_and_captions_grid, bleu_score_histogram
from src.data.load_data import load_test, load_predictions, load_idx_word_dicts
```

In [ ]:
```
_, captions_test, images_test = load_test()
idx_to_word, _ = load_idx_word_dicts()
captions_word = [[idx_to_word.get(key) for key in caption] for caption in captions_test]

predictions_word = load_predictions()
```

In [ ]:
```
show_10_images_and_captions_grid(images_test, predictions_word, encoded=False, file_name='predictions.png')
```

In [ ]:
```
print("Captions length {}, predictions length {}".format(len(captions_word), len(predictions_word)))

print("First caption {}".format(captions_word[0]))
print("First prediction {}".format(predictions_word[0]))
```

```
Captions length 5000, predictions length 5000
First caption ['a', 'black', 'and', 'white', 'dog', 'balances', 'on', 'a', 'wooden', 'plank', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_']
First prediction ['a', 'dog', 'is', 'jumping', 'over', 'a', 'hurdle', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_', '_']
```

In [ ]:
```
independent_bleu_scores = get_bleu_scores(captions_word, predictions_word, smoothing=1, independent = True)
cumulative_bleu_scores = get_bleu_scores(captions_word, predictions_word, smoothing=1, independent = False)
```

Conclusions

The independent BLEU-1 scores (using 1-grams) are the highest with a mean of 0.72 and maximum of 0.97. As BLEU-n increases, the scores decrease slightly. This means that the model is slightly better at replicating certain key words than at replicating the word order or set of 2-4 words in a row.

The distribution of the scores can be seen in the histograms below. The cumulative BLEU-n scores have a similar distribution.

In [ ]:
```
print("Summary of Independent scores")
print(independent_bleu_scores.describe().loc[['mean','max'],:])

print("Summary of Cumulative scores")
print(cumulative_bleu_scores.describe().loc[['mean','max'],:])
```

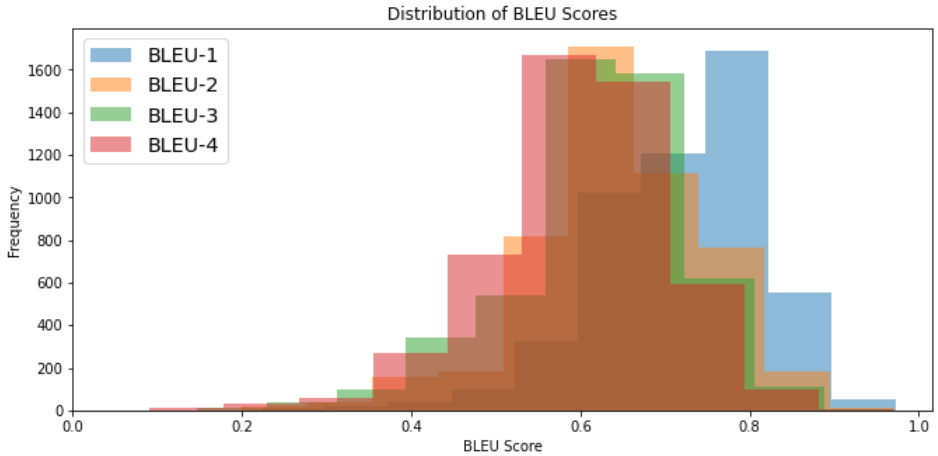
```
Summary of Independent scores
```

```
      BLEU-1  BLEU-2  BLEU-3  BLEU-4
mean    0.72    0.65    0.62    0.60
max     0.97    0.97    0.97    0.97
Summary of Cumulative scores
      BLEU-1  BLEU-2  BLEU-3  BLEU-4
mean    0.72    0.68    0.66    0.64
max     0.97    0.97    0.97    0.97
```

In [ ]:

```
print("Independent BLEU Histogram")
bleu_score_histogram(independent_bleu_scores, "independent_bleu.png")
print("Cumulative BLEU Histogram")
bleu_score_histogram(cumulative_bleu_scores, "cumulative_bleu.png")
```

```
Independent BLEU Histogram
```




```
Cumulative BLEU Histogram
```