Annotation analysis

November 9, 2020

[32]:
```
import csv

all_results = []

with open("class.csv", "r") as inf:
    results = csv.reader(inf)
    next(results)
    for r in results:
        annotator, annotation, review = r
        all_results.append({"annotator": annotator, "annotation": annotation,␣
 ↪"review": review})



## how many data points are there?
print(len(all_results))

## how many annotators are there?
annotators = set()
from collections import defaultdict
review2judgements = defaultdict(list)

for result in all_results:
    review2judgements[result["review"]].append({"annotator":␣
 ↪result["annotator"], "annotation": result["annotation"]})
    annotators.add(result["annotator"])

len(annotators)
```

```
747
```

[32]:
```
30
```

[45]:
```
def pairwise_agreement(results):
    '''
    Compute the pairwise agreement between raters for the input results
```

```
    To compute pairwise agreement compare judgements from all pairs of␣
↪annotators for a given item
    Return the fraction of pairs of annotators who agree
    '''
    total_judgements = {}
    for result in results:
        for other_result in results:
            if result["annotator"] != other_result["annotator"]:
                pair = [result['annotator'], other_result["annotator"]]
                pair.sort()
                pair = "-".join(pair)
                total_judgements[pair] = (result["annotation"],␣
↪other_result["annotation"])

    out = total_judgements.values()
    agrees = 0
    for pair in out:
        judgement1, judgement2 = pair
        if judgement1 == judgement2:
            agrees += 1
        return agrees/len(out)

review4 = {"1": 1, "2": 0, "3": 1}

# do 1 and 2 agree == No

# do 1 and 3 agree == Yes

# do 2 and 3 agree == No

# agreement rate = number of agreements / number of pairs: 1/3

pairwise_agreement(review2judgements[review])
```

[45]:
```
0.0022988505747126436
```

[48]:
```
min_ = 1
review = ""
for review in review2judgements:
    agreement_rate = pairwise_agreement(review2judgements[review])
    if agreement_rate < min_:
        min_ = agreement_rate
        min_review = review
min_review
```

[48]:
```
'I visited the Old Town Tortilla factory about five years ago with fond memories
so when back in Scottsdale tonight I decided to give it another go. I should
```

```
have read the Yelp reviews before going, my experience was nothing special. My
waiter was friendly enough but the food was just OK. I ordered the Grilled Mahi
Mahi Fish Tacos. Upon my waiters advice I order the \\"sauce\\" on the side
because I was concerned about them being too spicy. What I received was three
\\"chunks of fish\\" on three mini tortillas with four small condiment bowls
containing the black beans, jalape\\u00f1o sauce, guacamole and a white cucumber
sauce? It just looked kinda strange and the fish wasn\'t all that fresh. I
didn\'t complain because my waiter only asked if I wanted dessert.  Maybe it\'s
me but it just wasn\'t what I expected. I think there are a lot of other good
choices in Scottsdale.'
```

0.0.1 Per-item analysis

- Which review has the highest and lowest pairwise agreement rate? Does this make sense?

Unable to finish in classtime.

0.0.2 Random agreement rate

If two reviewers answered randomly (meaning just picked random annotations) how often would they agree just by chance?

0.5

0.0.3 Fleiss Kappa

Fleiss kappa measures the exent to which pairs of reviewers agree, as compared to how much they would agree by chance.

- $\bar{P}_e$ is the rate at which reviewers agree by chance
- $\bar{P}$ is the pairwise agreement rate across all items the dataset
  - note: the Wikipedia article uses a slightly different definition of $\bar{P}$, because it assumes all reviewers review all items, which is not true in our case

$\kappa = \frac{\bar{P}-\bar{P}_e}{1-\bar{P}_e}$

- What is the highest possible value of Fleiss Kappa? What is the lowest?

1

- What does the denominator mean? If $\bar{P}_e$ is high, then is the denominator high or low? The denominator is the probability that the reviewers don't agree by chance. If $\bar{P}_e$ is high, the denominator is low.
- If $\bar{P}$ is high and $\bar{P}_e$ is high, do you think the task is well-defined?

Inconclusive.

- If $\bar{P}$ is high and $\bar{P}_e$ is low, do you think the task is well-defined?

Yes, because the reviewers agree at a high rate even though the probability that they agreeby chance is low.

- What do you think the Fleiss Kappa will be for the Yelp data set? Do you think it will be higher or lower than for the emotions dataset?

[1]:
```
# Compute Fleiss Kappa for the dataset

def kappa(Pe, Pbar):

    return (Pbar - Pe)/(1 - Pe)
```

[ ]:
```

```