Assignment 4: Semantics

Oreen Yousuf

December 12, 2020

1 Lexical Semantics: Error analysis

1.1

Interest1 was the hardest "sense" for the model to estimate for the word "interest". It had an 85.89% incorrect labeling by the model. Of the incorrect labels, the model guessed a meaning of Interest2 the majority of the time.

Hard3 was the hardest "sense" for the model to estimate for the word "hard". It had a 67.53% incorrect labeling by the model. Of the incorrect labels, the model overwhelming guessed a meaning of Hard1.

Serve2 was the hardest "sense" for the model to estimate for the word "serve". It had a 64.497% incorrect labeling by the model. Of the incorrect labels, the model guessed a meaning of Serve12 the majority of the time.

The Naive Bayes classifiers worked the best on all 3 keywords. I think there are patterns/systematic errors within this example, though, specifically in regards to having a less rigid rule-set for contextual information. The conceit of word sense disambiguation follows the outline of selecting a keyword in context, gaining specified features of the sentence/excerpt the keyword belongs to, like its part-of-speech (POS), neighboring word(s), and keying in on its position, computing the probabilistic approximation model, and returning the semantic meaning associated with the highest probabilistic score. If we take the word "hard" for example, because hard1 and hard3 have the same POS and therefore have, liberally, similar neighboring words, its position can be said to be the most crucial characteristic to determine it's correct semantic meaning. Which would not be surprising as there was a 67.53% incorrect labeling. In addition to this, of the total 67.53%, only 6.49% stemmed from misidentification as hard2, which has a much more different definition of "dispassionate". This drastically different definition allows for it to appear in unique contexts, such as the following scenario:

example number: 151

sentence: on monday and wednesday , multiple accidents caused miles - long morning backups before our HARD hatted friends even went to work .

guess: HARD2; label: HARD3

Hard2's definition is "dispassionate", with an example of "a hard bargainer." The use of this definition of "hard" most suits nouns like people (i.e. bargainer), unlike inanimate objects/nouns that are better suited for hard1 and hard3. It is easier to see why the model guessed hard2 for example number 151 after thinking about this. The same can be said for "interest". The same can be said for "interest" and "serve". Interest2's definition of "quality of causing attention to be give to" paired with the example "they said nothing of great interest" is eerily close to overlapping instances of interest1's definition of

"readiness to give attention". The difference between their applications is smallest between the six total definitions of "interest". Exemplified by the following scenario:

example number: 29

sentence: other losers included pharmaceutical and textile shares , but two issues attracted investor INTEREST because of strong earning prospects for new products .

guess: interest_2; label: interest_1

No other "interest" definition is closer to interest1's uniqueness in this, and many more, situations. Interest6's profit related definition of could've potentially made it a prospective guess by the model given it's likely neighboring words relating to money like how "invester ... earning ... products" is present in example 29. The order of most misidenifications for "serve" can be seen if you continue to follow the idea of close-meaning definitions having higher numbers of misidentification. Serve6 is overwhelming tied to things like food, serve10 slightly edged out by serve12 due to it being less tied to functionality than the latter serve's definition, which is much closer to serve2's definition of holding office or serving in a specific function.

example number: 16

sentence: she slipped through a thick tangle at the edge of the abandoned village and was attracted by the rustling of several rats that were working through a kitchen midden , when a more enticing sound caught her attention . even before she began her stalk on the rats she swung through one of the thatch-roofed pole huts and leaped easily onto a wooden platform that had once SERVED as a bed for an entire seminole family .

guess: SERVE12; label: SERVE2

It would be hard to choose which "serve" sense would be the appropriate one between serve2 and serve12 when analyzing the excerpt of "...had once SERVED as a ..." removed from the confines of this sentence.

1.2

Precision(P) for Binary Classifications are: P = TruePositives/(TruePositives + FalsePositives) and for Multi-Class Classifications, like the Confusion Matrices we're working with here, the Precision is: P = TruePositivesAllClasses/(TruePositivesAllClasses + FalsePositivesAllClasses).

*I will state that I'm unsure if the question is asking for Precision and Recall for each word's sense, or just the total Precision (P) and Recall (R) for one of the words, so I will provide both.

P(hard1) = 676/702 = 0.88%, P(hard2) = 44/70 = 0.6285%, P(hard3) = 25/30 = 0.8333. These are for each class's precision. But for the Precision of Multi-Class Classifications/a Confusion Matrix, the precision is as follows: P(hard) = (676 + 44 + 25)/[(676 + 44 + 25)+(44 + 47+ 21 + 5 + 5)] = 745/867 = 0.859%.

R(hard1) = 676/(676+21+5) = 676/702 = 0.963%, R(hard2) = 44/(44+44) = 0.5, R(hard3) = 25/(25 + 47 + 5) = 25/77 = 0.325%. These are for each class's recall. But for the Recall of Mullti-Class Classifications/a Confusion Matrix, the recall is as follows: R(hard) = (676 + 44 + 25)/[(676 + 44 + 25)+(44 + 47+ 21 + 5 + 5)] = 745/867 = 0.859%.

Note that the Multi-Class Precision and Recall are equal to each other (0.859%). For multiclass classifications there is the general equation of F1 = 2 * [(P*R)/(P+R)]. This dictates that the F-Measure will also be of the same value iff P = R. F1 = 2 * [(0.859*0.859)/(0.859+0.859)] = 0.859 .

Precision is a helpful measure when a large number of False Positives are present because it assesses how precise, error-free, exact, etc. your model is out of predicted Positives. Spam detection is an application that can make good use of Precision. Recall can be greatly utilized when evaluating just how many True Positives were captured after your model takes in data and yields a large number of Positives. Fraud detection is an application that can make good use of Recall. F-Measure/F1 Score is simply a function of the previous two measures; a balance between them. However, when your experiment produces many True Negatives, seeing as it's an uneven class distribution, *while* still having the desire for a balance between Precision and Recall, an F-Measure/F1 Score would be a better measure to use than Accuracy.

Although Accuracy performs adequately/well with a balanced data-set or bank of information, it's somewhat limited to when your classes are balanced. Thus, it might be a good option for training set in stone data, but its costly over-attention to True Negatives and preference for similar valued False Negatives and False Positives may make you reconsider if it's always the go-to performance measure.

2 Semantic Role Labeling

3 VG: WordNet