

### Semantics Midterm

1. B's utterance implicates that B has a cat, and the cat is not a siamese. By not directly answering A's question, B is suggesting that its cat is not siamese since if its cat were siamese it would have been much simpler to just answer "yes". From the Grecian framework, we can see that this speaker is adhering to the maxim of quantity in trying to be as informative as they can be. B could have answered A by just responding "no", but by instead responding that it had a cat, B is offering information that could be useful to A in this line of questioning. In this way, B is adhering to the maxim of relation because it seems that by telling that it has a cat, B is offering relevant information that is pertinent to the discussion. Although B's utterance does not answer A's question semantically, B may have chose this response as opposed to B' because B may have thought that it offered more relevant information to the discussion than B'. Furthermore, B may have wanted to make it clear that it has a cat even though the cat is not a siamese.

be more specific  
 what submaxim is being flouted  
 in order to give rise to this implicature  
 you also need to show it's an implicature with a cancellation test  
 you don't have to assume this, you derive it directly from the denotation for [[the]]

2a. It is important to note we are assuming the presupposition that there is only one book and it is unique. Here is a screenshot of my calculations from the lambda notebook:

```
In [3]: a= Alfonso * (borrowed * (the * book))
a
a.tree()

Out[3]: 1 composition path:
```

$[[\text{borrowed}]]_{(e,(e,t))}$ $\lambda x_e . \lambda y_e . \text{Borrow}(y_e, x_e)$	<table border="1"> <tr> <td><math>[[\text{the}]]_{(e,t),e}</math> <math>\lambda f_{(e,t)} . \lambda x_e . (f_{(e,t)}(x_e) \wedge (x_e \in c_{[e]}))</math></td> <td><math>[[\text{book}]]_{(e,t)}</math> <math>\lambda x_e . \text{Book}(x_e)</math></td> <td>[FA]</td> </tr> <tr> <td colspan="2"><math>[[[\text{the book}]]]_e</math> <math>\lambda x_e . (\text{Book}(x_e) \wedge (x_e \in c_{[e]}))</math></td> <td>[FA]</td> </tr> </table>	$[[\text{the}]]_{(e,t),e}$ $\lambda f_{(e,t)} . \lambda x_e . (f_{(e,t)}(x_e) \wedge (x_e \in c_{[e]}))$	$[[\text{book}]]_{(e,t)}$ $\lambda x_e . \text{Book}(x_e)$	[FA]	$[[[\text{the book}]]]_e$ $\lambda x_e . (\text{Book}(x_e) \wedge (x_e \in c_{[e]}))$		[FA]	[FA]
$[[\text{the}]]_{(e,t),e}$ $\lambda f_{(e,t)} . \lambda x_e . (f_{(e,t)}(x_e) \wedge (x_e \in c_{[e]}))$	$[[\text{book}]]_{(e,t)}$ $\lambda x_e . \text{Book}(x_e)$	[FA]						
$[[[\text{the book}]]]_e$ $\lambda x_e . (\text{Book}(x_e) \wedge (x_e \in c_{[e]}))$		[FA]						
$[[[\text{borrowed} [\text{the book}]]]_{(e,t)}$ $\lambda y_e . \text{Borrow}(y_e, \lambda x_e . (\text{Book}(x_e) \wedge (x_e \in c_{[e]})))$		$[[\text{Alfonso}]]_t$ $a_e$						
$[[[[\text{borrowed} [\text{the book}]] \text{Alfonso}]]_t$ $\text{Borrow}(a_e, \lambda x_e . (\text{Book}(x_e) \wedge (x_e \in c_{[e]})))$								

8/8

2b. When looking at the sample sentences, it seems that book acts in the same way that chair acts, but mother acts in a little differently. Thus, it makes sense to think that the type of "book" is <e,t> just like chair. However mother appears differently and this may occur because of some of the constraints on the concept of having a mother. It seems impossible for Joanna to not have a mother and furthermore highly unlikely for Joanna to not have only one mother who is her biological mother. If we look at the tree, we can see that from "s" to "mother", we want to get <e,e>. This is because if this whole DP of "s mother" gives us <e,e> than we can take Joanna which is type e and get the whole DP to be type e as desired. It thus seems to make sense that if we assume "s" to be of type <e,<e,t>>, than mother would need to be type <<e,<e,t>>,<e,e>>. This type would take "s" and return <e,e> which is what we desire the whole DP to be. With regards to how it contributes to the possessive DP, mother takes in the type of "s" and returns type <e,e>. In other words it takes



good  
so what is the  
essence of the  
difference between  
book type nouns  
and mother type  
nouns

6/8

function that takes an entity and returns a property, and returns a function that takes an entity and returns an entity. Mother refers to an actual person, but what complicates things is the possessive and the fact that mother is a limited relationship constrained by the limits described above. Book takes an entity and returns a truth condition. It would be possible to have a book, many books, or no books at all. On the contrary it seems impossible to not have a mother or more than 1 mother if we assume we are considering biological mothers. Thus truth conditionally, the possessor must have one and only one possessee in the case of mother, and the possessor can have any number of books in the case of book.

what would that property be?

6/9

2c. The whole DP should have type e. This is sensible because on a larger structural level Joanna's mother refers to one single person: the mother of Joanna. If we assume we are referring to a biological mother, than it is important to consider that Joanna's mother refers to one and only one person and it cannot refer to anyone but this one person. Furthermore it may be important to note that Joanna must have a mother in order to exist and thus there necessarily exists one entity who is Joanna's mother. "s" should have type <e,<e,t>>, and mother should have <<e,<e,t>>,<e,e>>. For the case with book, we should consider "s" is ambiguous in order to account for book's type of <e,t>. Within the larger structure this is sensible because it should make the tree work out. Under these assumptions the tree should work out properly and as desired.

so what are the actual denotations (the lambda functions)? How would this all compose in a tree?

3/8

2e. I am having difficulty figuring out the proper method to input "mother" into lambda. I know that I would define mother to have type <<e,<e,t>>,<e,e>>, Joanna to be e, "s" to be <e,<e,t>>, doctor to be <e,t>, A to be <e,t> and met to be <e,t>. Next I would input in lambda A \* (doctor \* (met \* (Joanna \* (POSS \* (mother))))).

as a transitive verb 'met' would have to be <e,<e,t>>

is the intensity actually focused on Alfonso here or the computer?

but utterances can in fact be taken as literally equivalent to

10/11

3a. It seems that an epithet intensifies or adds another dimension of meaning to a sentence it appears in. It seems to suggest a mood that the speaker is in as well. The context seems crucial to deciphering the meaning that is suggested by the use of the epithet. In the example given, if Alfonso just dropped a computer and it shattered and his friend made the utterance in 8, it seems to suggest the friend is really intensely upset at alfonso. However in a case where alfonso tapped a few buttons and perhaps made a computer window close and his friend made this utterance it seems to be poking fun at alfonso through sarcasm. In either of these cases, the epithet intensifies the situation at hand. However from our example it seems that the epithet is crucial to the "sarcastic" meaning and impactful but not essential to the "serious" meaning. In the case where the speaker is being sarcastic, if they had just said "Alfonso broke the computer" it may have been taken literally. By using the epithet, it makes the sentence more ridiculous because the computer is not broken, and thus this alerts the listener that the utterance is meant to be taken sarcastically. In the serious scenario, the speaker could still come off as really mad by not using the epithet, but by using the epithet they intensify the angriness that they portray to the listener. Thus, the context is crucial to determining the contribution of the epithet.

good

good, so critically we are not dealing with the literal meaning of 'fuck', 'damn' etc

3b. We can consider embedded contexts. Using the example of “Alfonso broke the damn computer” we come up with the following: (negation) "Alfonso did not break the damn computer.", (question) "Did Alfonso break the damn computer?", (imperative) "Show me that Alfonso broke the damn computer.", (antecedent of a conditional) "If Alfonso broke the damn computer then prove it to me.". It appears that the meaning contrived by “damn” in all these situations does not change. In every case the inclusion of the epithet adds an intensity and calls attention to the computer. Although context is crucial in determining how the epithet contributes to the meaning of the utterance, it seems that in every situation the epithet adds another dimension to the situation. It is difficult to categorize the epithet as an implicature, entailment, or presupposition. Since the context can widely vary what the epithet contributes semantically to the utterance, it seems misguided to categorize it as inherently implying, entailing, or presupposing anything. Nonetheless, in every case the epithet does add something to the utterance. Thus, it seems possible to define the epithet as a context-dependent presupposition. If we assume a certain context, for example that the speaker is intending to portray anger, then in every embedded situation the inference that the speaker is mad triggered by the inclusion of the epithet survives. This logic suggests it would be a presupposition that depends on the context being fixed beforehand.

good  
so this might suggest it is a presupposition

you could show a cancellation test to strengthen this point

does it need to be literally true in the way other presuppositions do?

8/11

3c. It think it is sensible to think that the whole DP here would be of type e because once the whole DP has been composed we should be referring to a single unique computer. We previously defined “the” as having a type of  $\langle e, t \rangle$  with specific conditions to meet in question 2. It also seems sensible to define “computer” as having type  $\langle e, t \rangle$  because it is essentially a property before it refers to a single unique computer. Thus we are left with determining the type of ADJ. I am not clear on how exactly this tree would look, and that is limiting me in this problem. Without knowing how the tree should look I am having difficulty determining how “the”, “ADJ”, and “computer” compose together. Nonetheless I would proceed with the same reasoning as I used previously in question 2.

is there anything that you wouldn't be able to capture in the actual lambda denotation? i.e. the emotional valence or intensity of the contribution from ADJ?

6/11

good  
do you mean  $\langle e, \langle e, t \rangle \rangle$ ?

3d. It seems that an important distinction among expressive adjectives is between ones that inherently have a negative connotation such as “stupid”, and ones that have a positive connotation such as “wonderful”. Of course, even an adjective like “wonderful” could be used in a sarcastic way that is meant to be taken as negatively, but on a surface level it has positive connotations. It may be important to first distinguish the surface level definition of each expressive adjective in order to categorize properly. After determining what is implied implicitly by the word then the context can be approached in order to determine whether the utterance is meant to be taken literally or in some sarcastic way.

+2

good, what about varying levels of intensity as well?  
can you think of a formal way to implement this that could interface with the lambda calculus?