Aspect-Based Sentiment Analysis on the Web using Rhetorical Structure Theory

Rowan Hoogervorst[1], Erik Essink[1], Wouter Jansen[1], Max van den Helder[1]
Kim Schouten[1], Flavius Frasincar[1], and Maite Taboada[2]

[1]Erasmus University Rotterdam, the Netherlands

[2]Simon Fraser University, Canada

Aspect-Based **Sentiment Analysis** on the Web

- Sentiment Analysis -> extract sentiment from text
- Sentiment can be defined as polarity (positive/negative)
- Or as something more complex (numeric scale or set of emotions)

- Useful for consumers to know what other people think
- Useful for producers to gauge public opinion w.r.t. their product

Aspect-Based Sentiment Analysis on the Web

- Nowadays the Web is filled with opinion and sentiment
- People freely share their thoughts on basically everything
- Useful, but lot of noise
- Need automatic methods to sift through this much data
- Our scope is consumer reviews

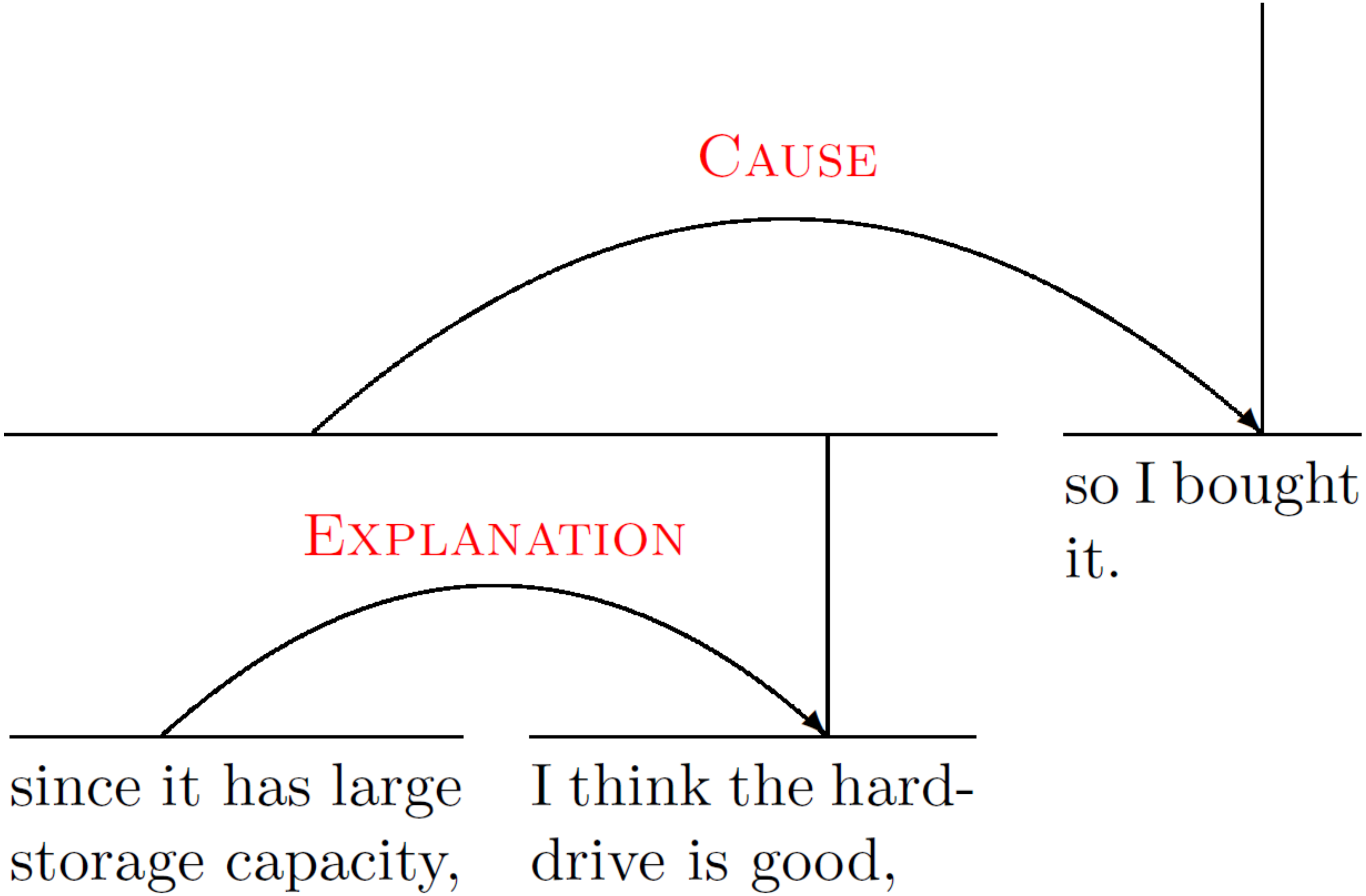
Aspect-Based Sentiment Analysis on the Web

- Sentiment Analysis has a scope, for instance a document
- More interesting however is the aspect level
- An aspect is a characteristic or feature of a product or service being reviewed
- This can range from general things like price and size of a product, to very specific aspects like wine selection for restaurants or battery life for laptops

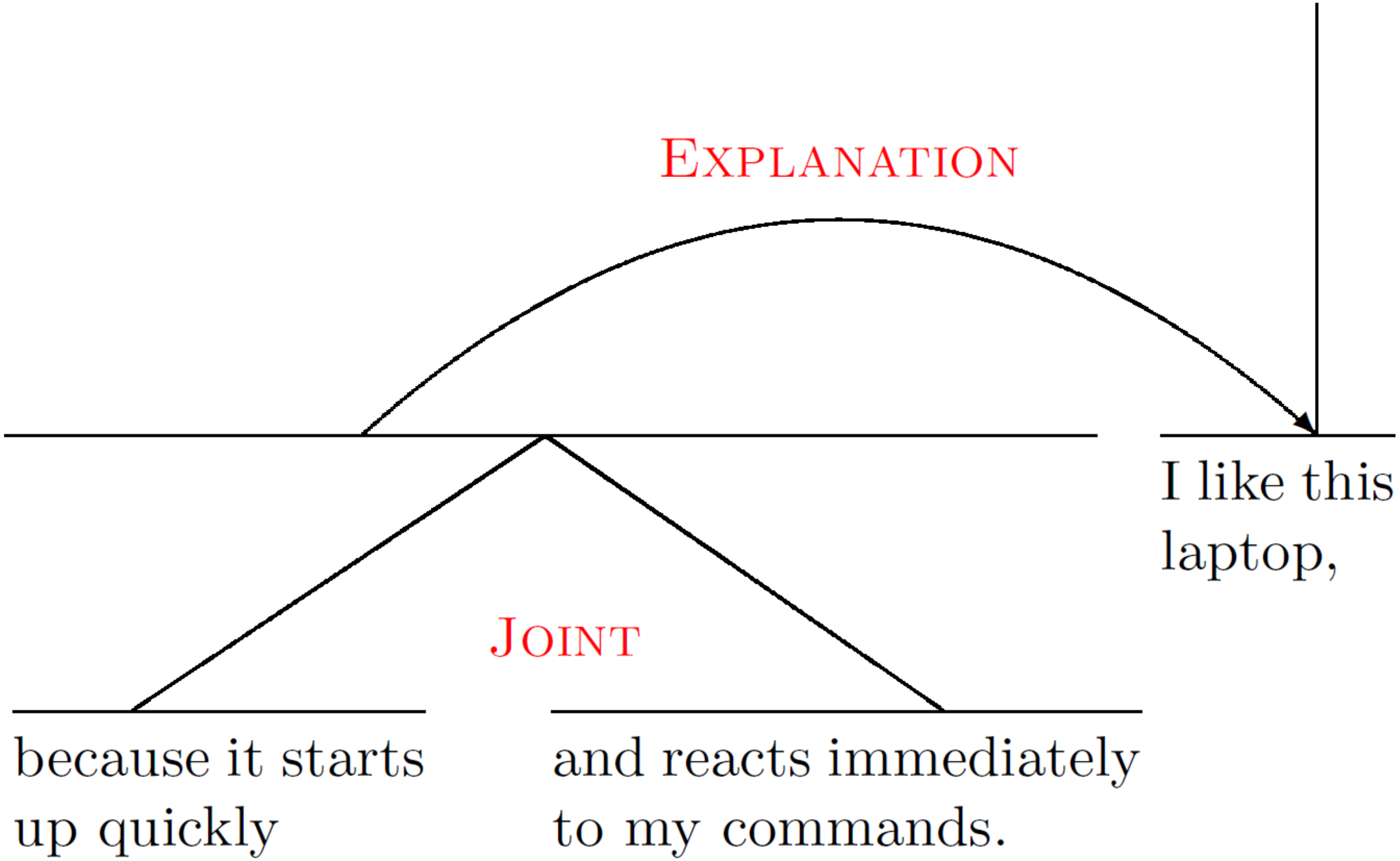
Rhetorical Structure Theory

- A theory that looks at the discourse structure of text
- Divides text into logical discourse units that are linked to each other

Rhetorical Structure Theory



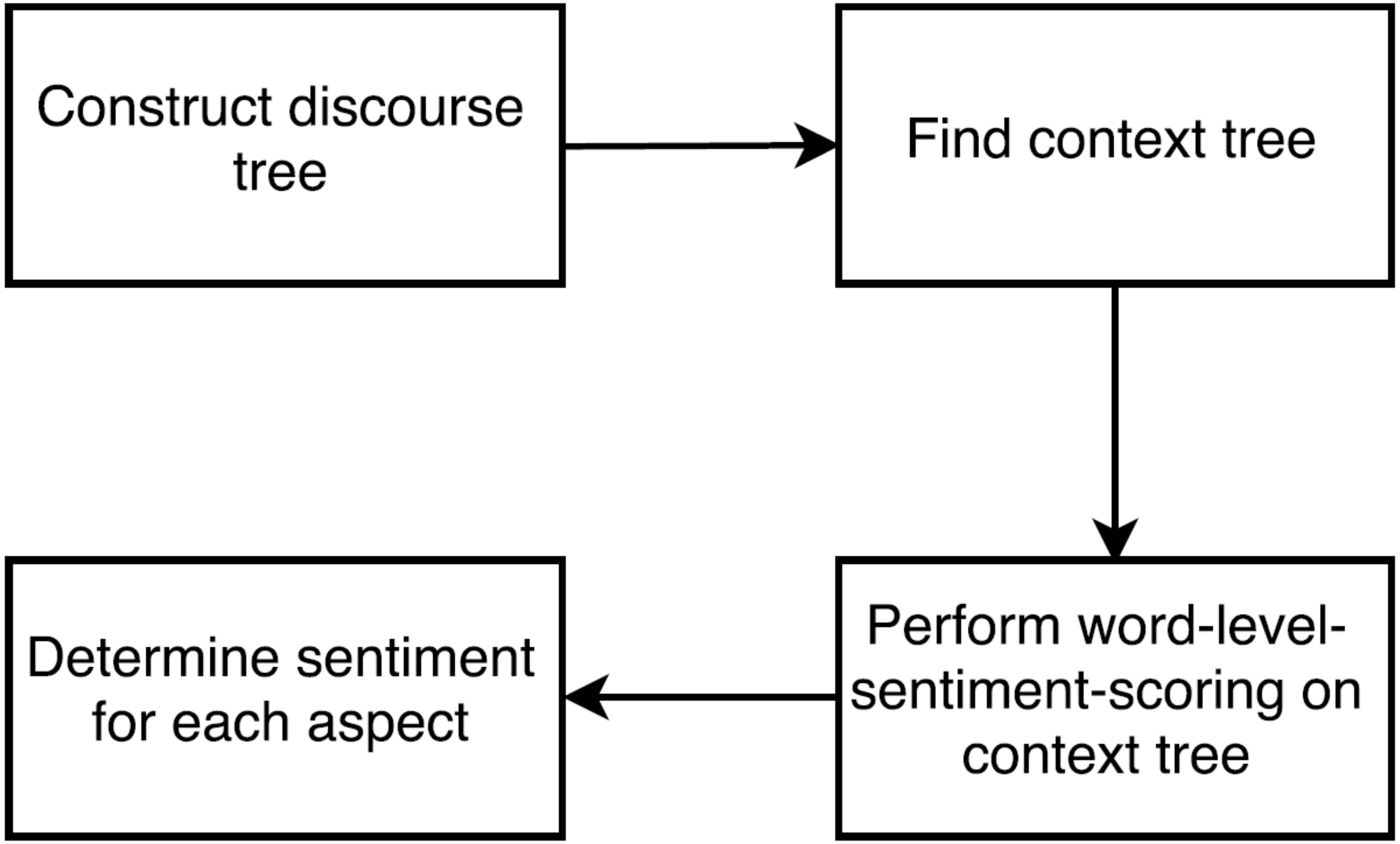
Rhetorical Structure Theory



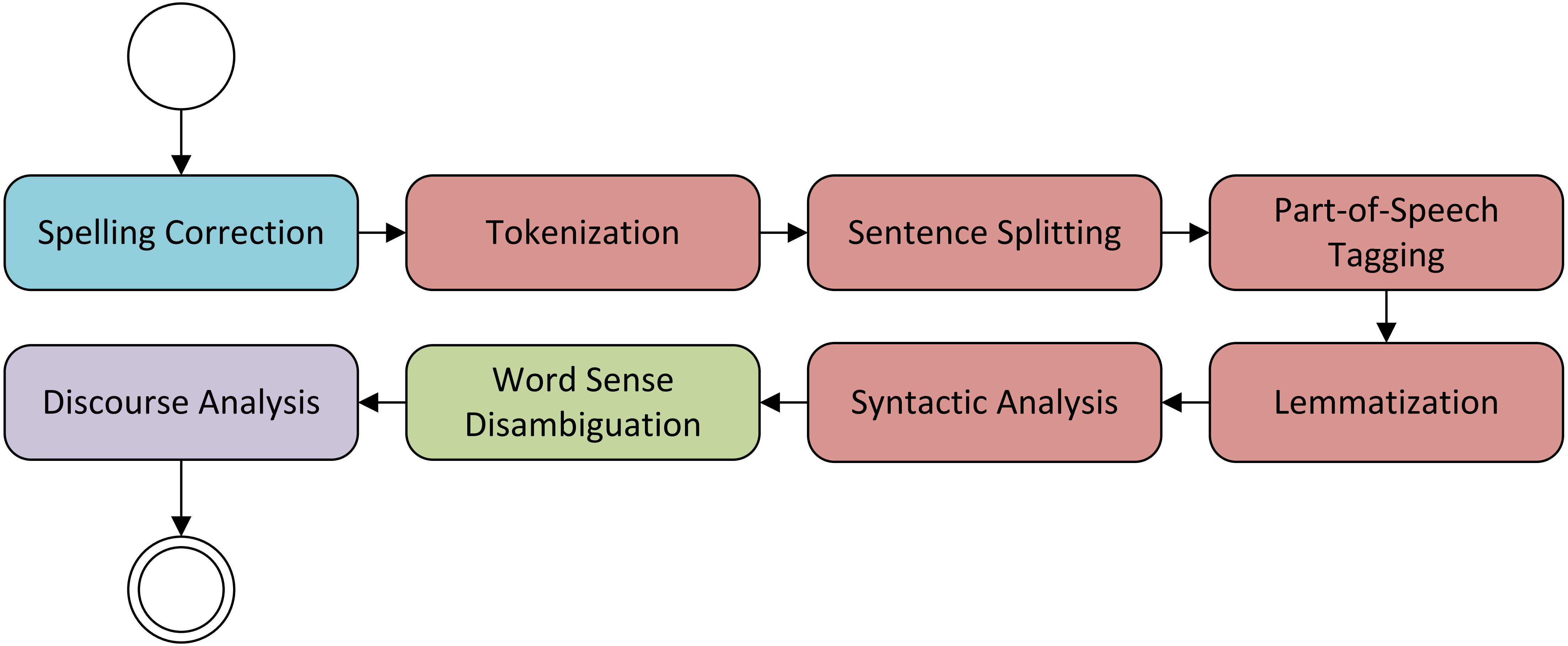
Using Rhetorical Structure Theory for ABSA

- The discourse tree shows how the various parts are related
- We can use it to determine which part of the text is relevant w.r.t. the current aspect -> context tree
- We can assign weights to the relations to distinguish between important parts of the text and less important ones
- Propagate sentiment over the context tree using these weights

Algorithm Setup



Algorithm Setup – Construct Discourse Tree



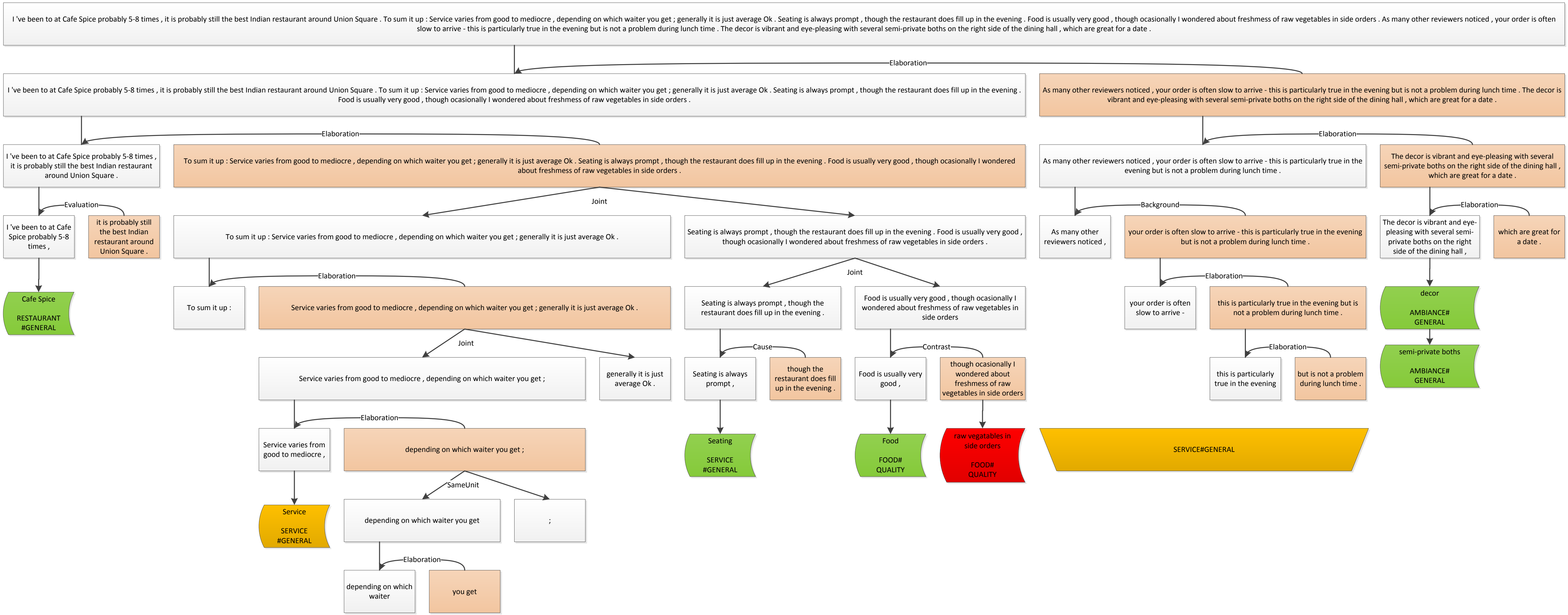
Algorithm Setup – Find Context Tree

- Satellites add information to nuclei
- But not the other way around
- This information asymmetry naturally leads to a context tree

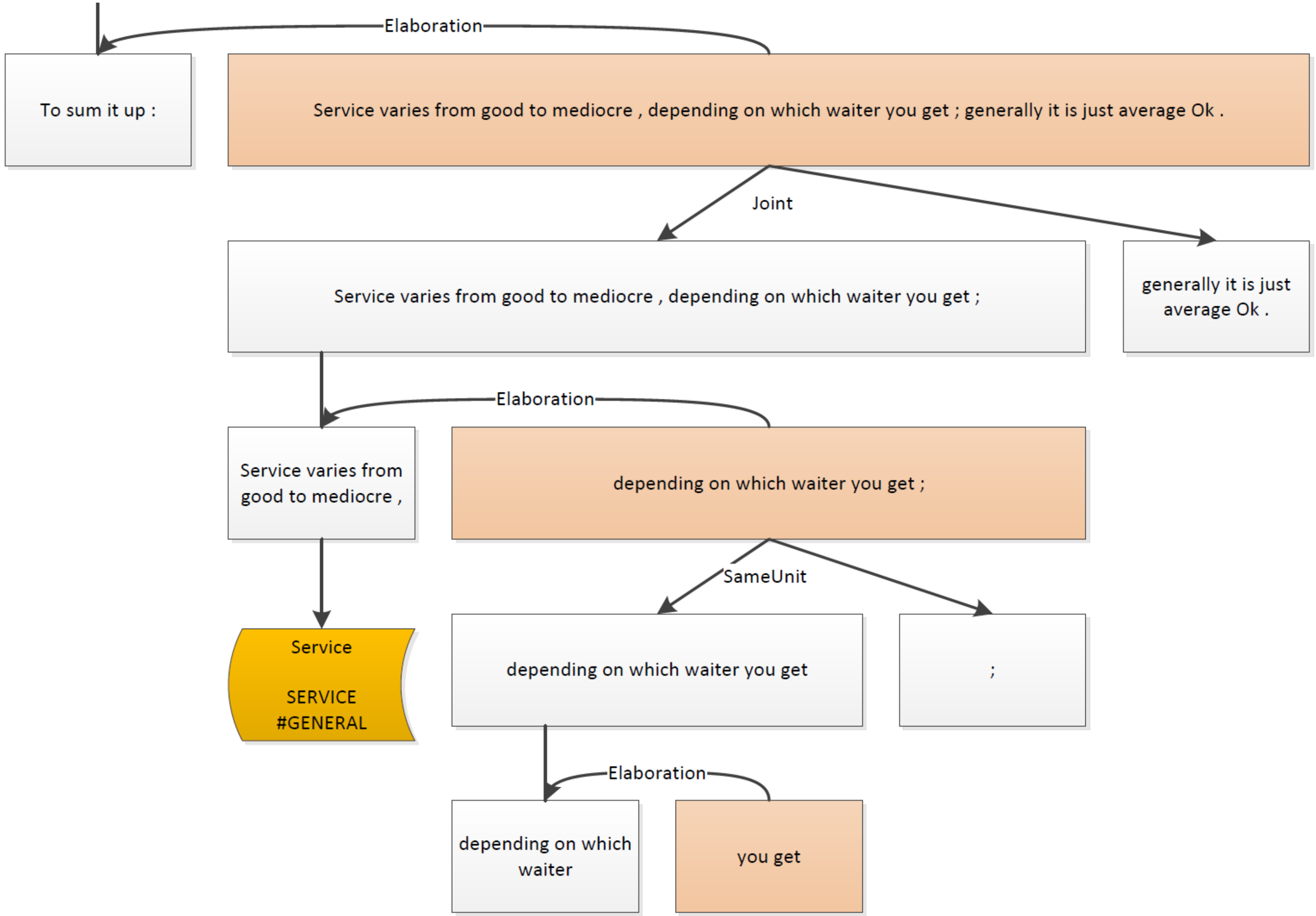
Algorithm Setup – Find Context Tree

- I've been to at **Cafe Spice** probably 5-8 times, it is probably still the best Indian restaurant around Union Square.
- To sum it up: **Service** varies from good to **mediorce**, depending on which waiter you get; generally it is just average Ok.
- **Seating** is always prompt, though the restaurant does fill up in the evening.
- **Food** is usually very good, though **ocasionally** I wondered about **freshmess** of **raw vegetables in side orders**.
- As many other reviewers noticed, your order is often slow to arrive - this is particularly true in the evening but is not a problem during lunch time.
- The **decor** is vibrant and eye-pleasing with several **semi-private boths** on the right side of the dining hall, which are great for a date.

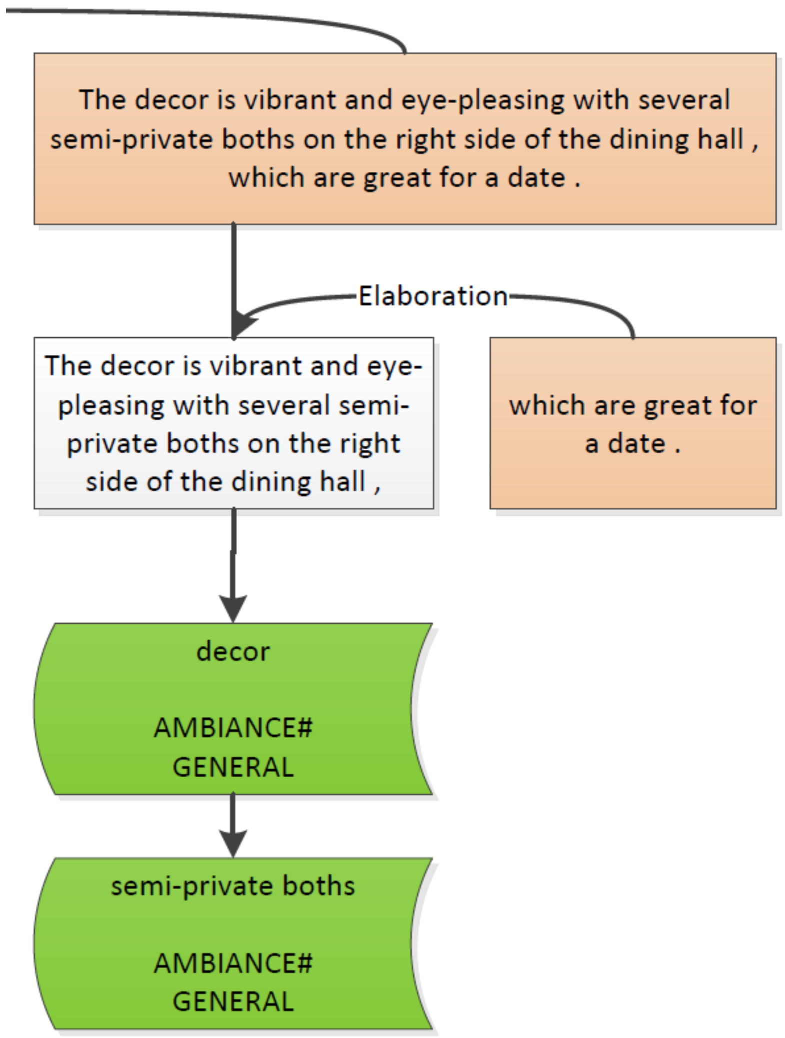
Algorithm Setup – Find Context Tree



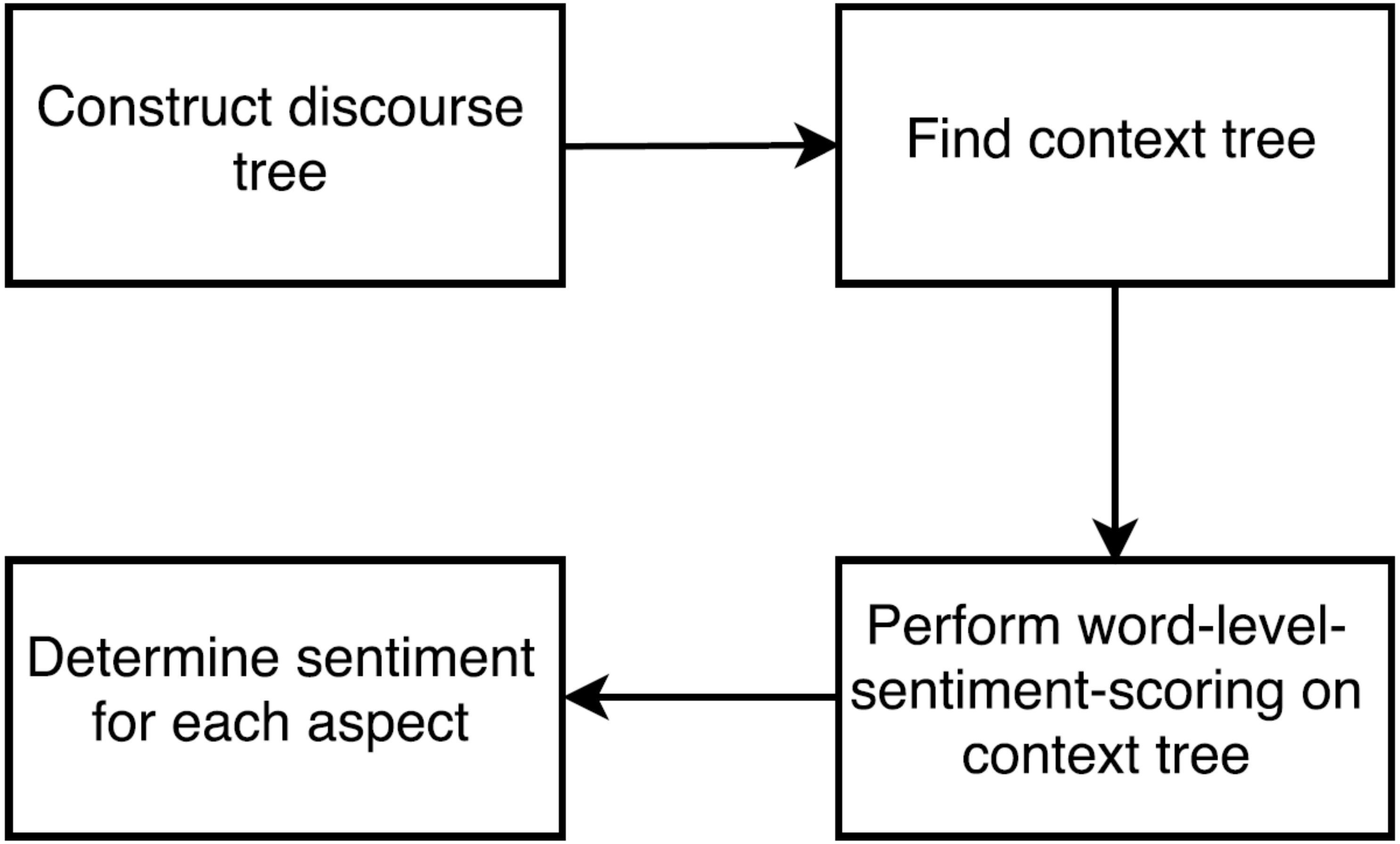
Algorithm Setup – Find Context Tree



Algorithm Setup – Find Context Tree



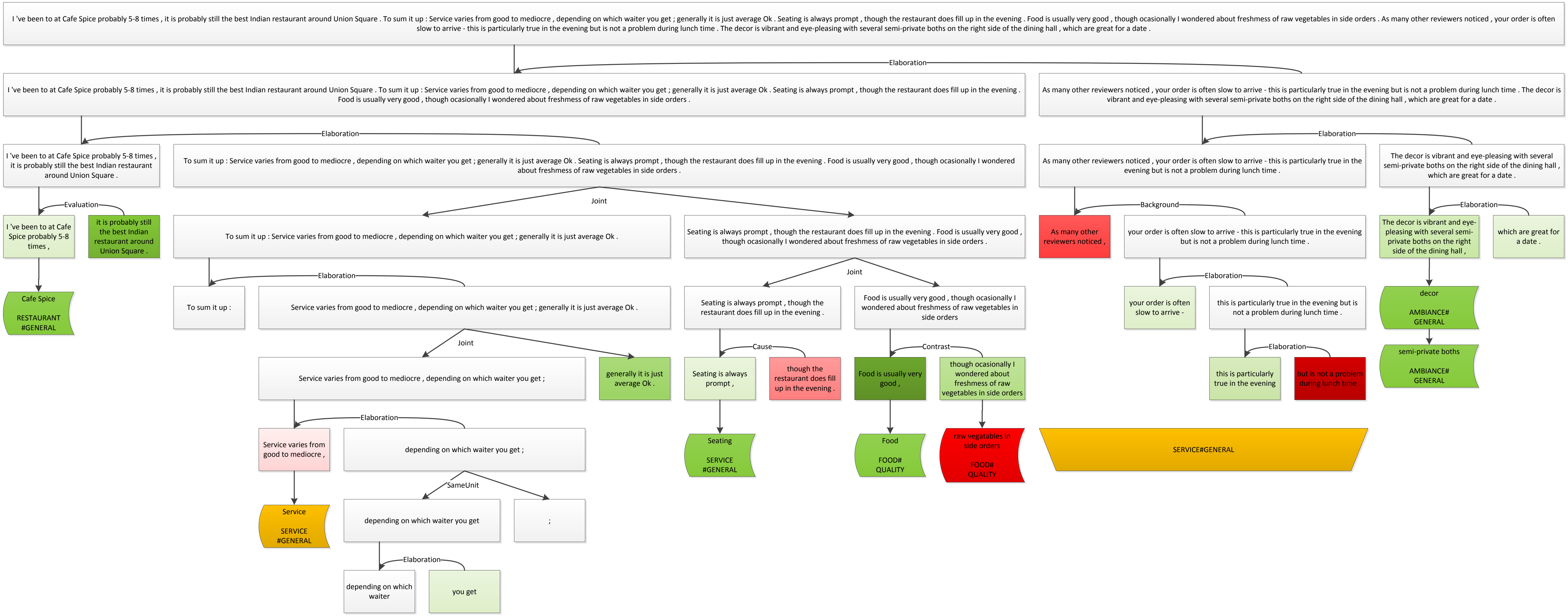
Algorithm Setup



Algorithm Setup – Word Sentiment Scoring

- Lesk is used as a basic Word Sense Disambiguation step
- Result: words are linked to WordNet synsets
- We use the SentiWordNet dictionary to get scores for synsets
- The sentiment of a leaf node is the sum of the sentiment of the words in that leaf

Algorithm Setup – Word Sentiment Scoring



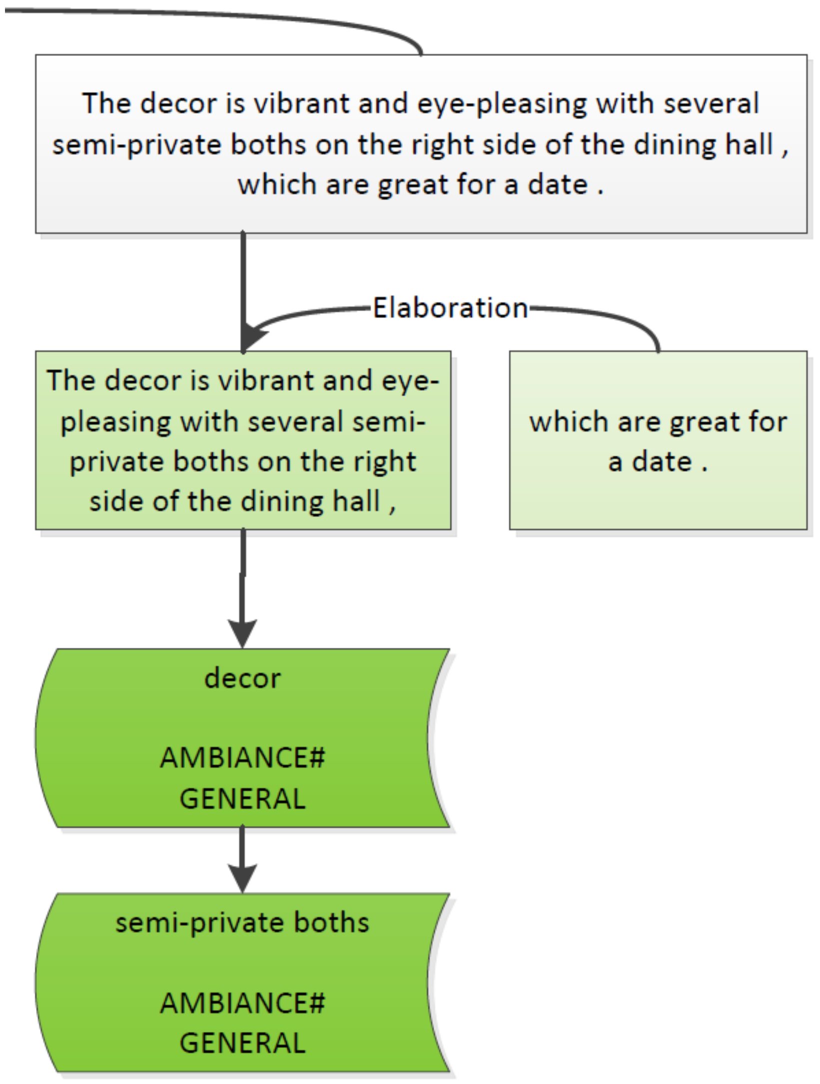
Algorithm Setup – Get Aspect Sentiment

- Sentiment of a leaf is computed as follows:

$$sent(s_i) = \sum_{t_j \in s_i} sent(t_j) \times \prod_{r_n \in P_{s_i}} w_{r_n}, \forall s_i \in S$$

- Weights are optimized using a Genetic Algorithm
- Take the sum of the sentiment scores of all leaf nodes that are in the context tree to get aspect sentiment
- A trained threshold epsilon is used to divide the scores into positive/negative classes

Algorithm Setup – Get Aspect Sentiment



Evaluation – Data sets

Type Reviews	SemEval year	Organized by	Targets	# Sentences	# Aspects
Laptops	2015	Review	No	1739	1974
Restaurants	2015	Review	Yes	1315	1654
Restaurants	2014	Review	Yes	3041	3693

Evaluation – Data snippet


```xml
                <Opinion target="NULL" category="RESTAURANT#GENERAL" polarity="positive" from="0" to="0"/>
            </Opinions>
        </sentence>
    </sentences>
</Review>
<Review rid="1032695">
    <sentences>
        <sentence id="1032695:0">
            <text>Every time in New York I make it a point to visit Restaurant Saul on Smith Street.</text>
            <Opinions>
                <Opinion target="Restaurant Saul" category="RESTAURANT#GENERAL" polarity="positive" from="50" to="65"/>
            </Opinions>
        </sentence>
        <sentence id="1032695:1">
            <text>Everything is always cooked to perfection, the service is excellent, the decor cool and understated.</text>
            <Opinions>
                <Opinion target="NULL" category="FOOD#QUALITY" polarity="positive" from="0" to="0"/>
                <Opinion target="service" category="SERVICE#GENERAL" polarity="positive" from="47" to="54"/>
                <Opinion target="decor" category="AMBIENCE#GENERAL" polarity="positive" from="73" to="78"/>
            </Opinions>
        </sentence>
        <sentence id="1032695:2">
            <text>I had the duck breast special on my last visit and it was incredible.</text>
            <Opinions>
                <Opinion target="duck breast special" category="FOOD#QUALITY" polarity="positive" from="10" to="29"/>
            </Opinions>
        </sentence>
```

Evaluation – Baseline

- The baseline model uses a word window around the aspects to get the context instead of RST when target is known
- Otherwise, context is formed by all words in sentence
- Sums up sentiment from words in context to get aspect sentiment
- Same SentiWordNet lexicon

Evaluation – Performance – Laptops 2015

(a) Performance of baseline method

	Precision	Recall	F_1
Overall	0.31	0.31	0.31
Positive	0.39	0.34	0.36
Negative	0.24	0.32	0.27

(b) Performance of proposed method

	Precision	Recall	F_1
Overall	0.67	0.67	0.67
Positive	0.67	0.88	0.76
Negative	0.69	0.47	0.56

Evaluation – Performance – Restaurants 2015

(a) Performance of baseline method

	Context window = 1			Context window = 2			Context window = 3		
Category	Precision	Recall	F_1	Precision	Recall	F_1	Precision	Recall	F_1
Overall	0.57	0.57	0.57	0.54	0.54	0.54	0.50	0.50	0.50
Positive	0.70	0.74	0.72	0.69	0.67	0.68	0.68	0.61	0.64
Negative	0.15	0.15	0.15	0.16	0.20	0.18	0.16	0.23	0.19

(b) Performance of proposed method

	Precision	Recall	F_1
Overall	0.74	0.74	0.74
Positive	0.80	0.86	0.83
Negative	0.52	0.47	0.49

Evaluation – Performance – Restaurants 2014

(a) Performance of baseline method

	Context window = 1			Context window = 2			Context window = 3		
Category	Precision	Recall	F_1	Precision	Recall	F_1	Precision	Recall	F_1
Overall	0.50	0.50	0.50	0.47	0.47	0.47	0.43	0.43	0.43
Positive	0.56	0.83	0.67	0.55	0.74	0.63	0.54	0.67	0.60
Negative	0.12	0.07	0.09	0.15	0.15	0.15	0.15	0.20	0.17

(b) Performance of proposed method

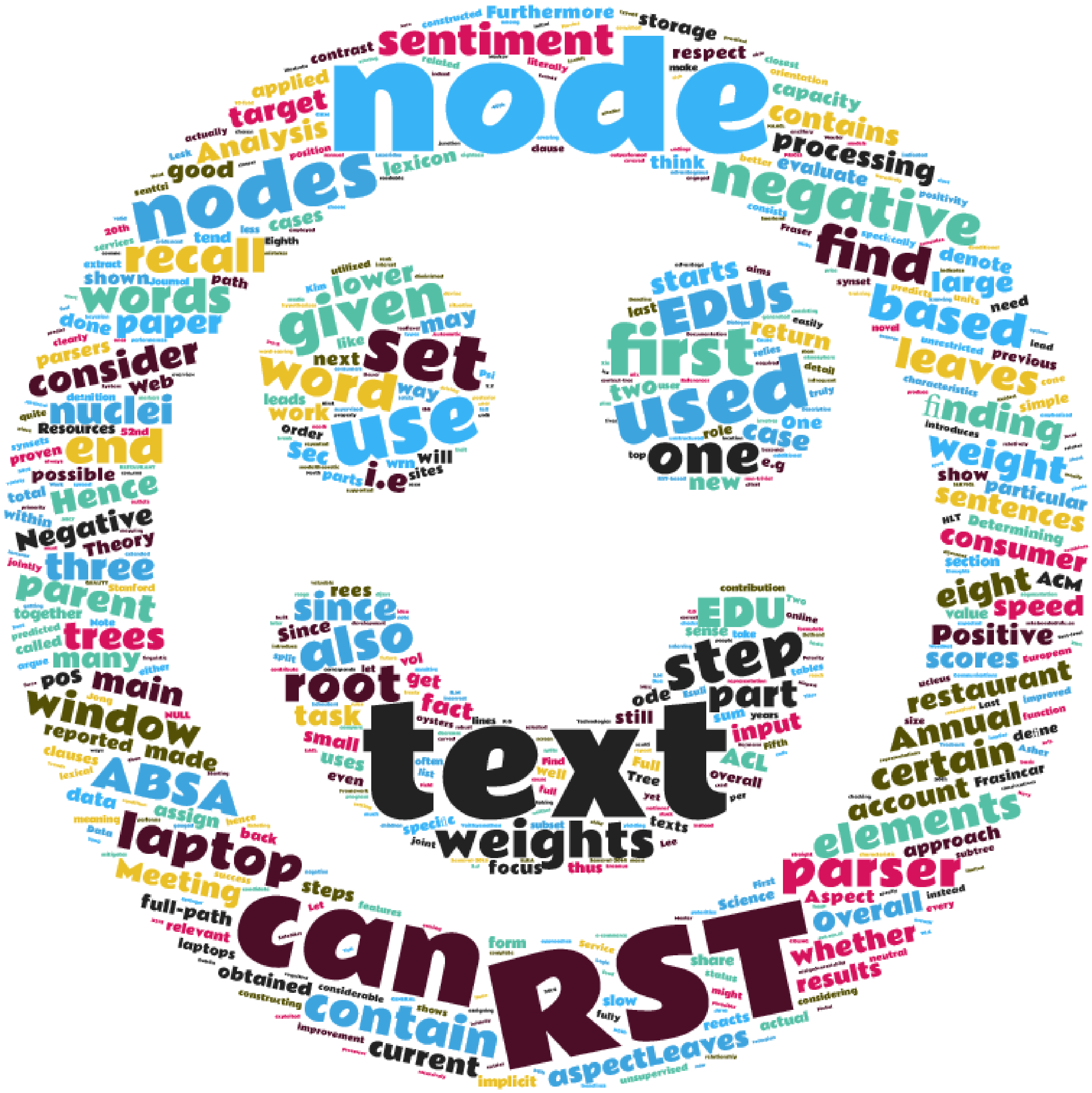
	Precision	Recall	F_1
Overall	0.60	0.60	0.60
Positive	0.64	0.91	0.75
Negative	0.42	0.32	0.36

Conclusion

- RST successfully used in sentiment analysis, but not on aspect level
- RST is used to define the context for each aspect
- And RST is used to logically combine sentiment from various parts of the text
- Baseline model with simple word-distance context is outperformed

- Address current shortcomings such as negations, better sentiment lexicon, better aspect sentiment computation
- Further research could include combining RST with classification algorithms (such as SVM)

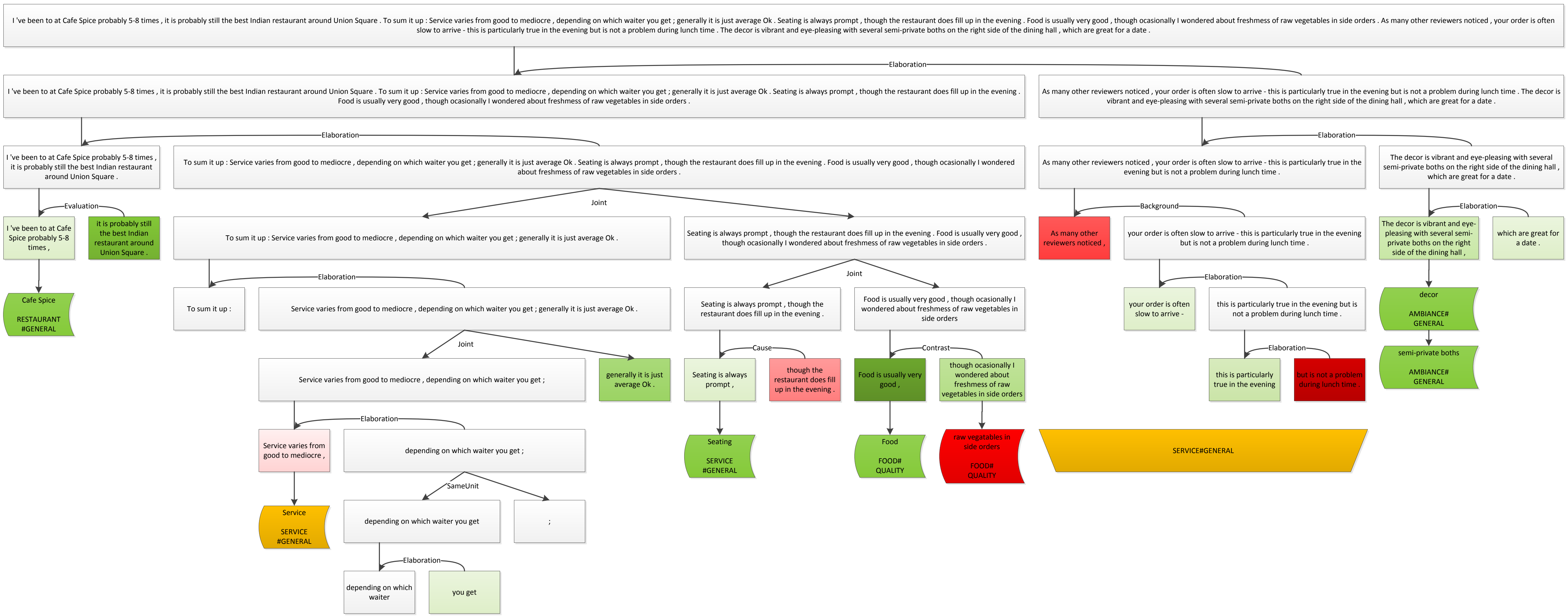
Questions?


node
text
RST
can
nodes
sentiment
negative
recall
words
given
set
word
use
first
used
EDUs
one
find
based
leaves
finding
weight
consumer
sentences
eight
speed
Positive
scores
restaurant
Annual
certain
account
elements
parser
overall
whether
results
aspectLeaves
contain
current
obtained
laptop
Meeting
ABSA
window
main
trees
many
parent
three
Negative
Hence
end
nuclei
consider
paper
Analysis
target
applied
processing
contains
capacity
storage
Furthermore
since
also
root
step
part
EDU
input
weights
task
fact
Theory
possible
reported
made
data
assign
full-path
steps
form
contrast
respect
lexicon
think
evaluate
good
cases
tend
shown
done
parsers
Resources
proven
total
within
together
called
predicted
pos
clauses
lexical
relevant
laptops
particular
Determining
section
ACM
value
show
large
denote
return
starts
lower
may
way
work
leads
order
case
new
top
role
two
last
Tree
overall
ACL
Full
small
uses
even
focus
thus
still
specific
lines
input
approach
Aspect
Science
share
status
slow
reacts
actual
implicit
considerable
constructing
improvement

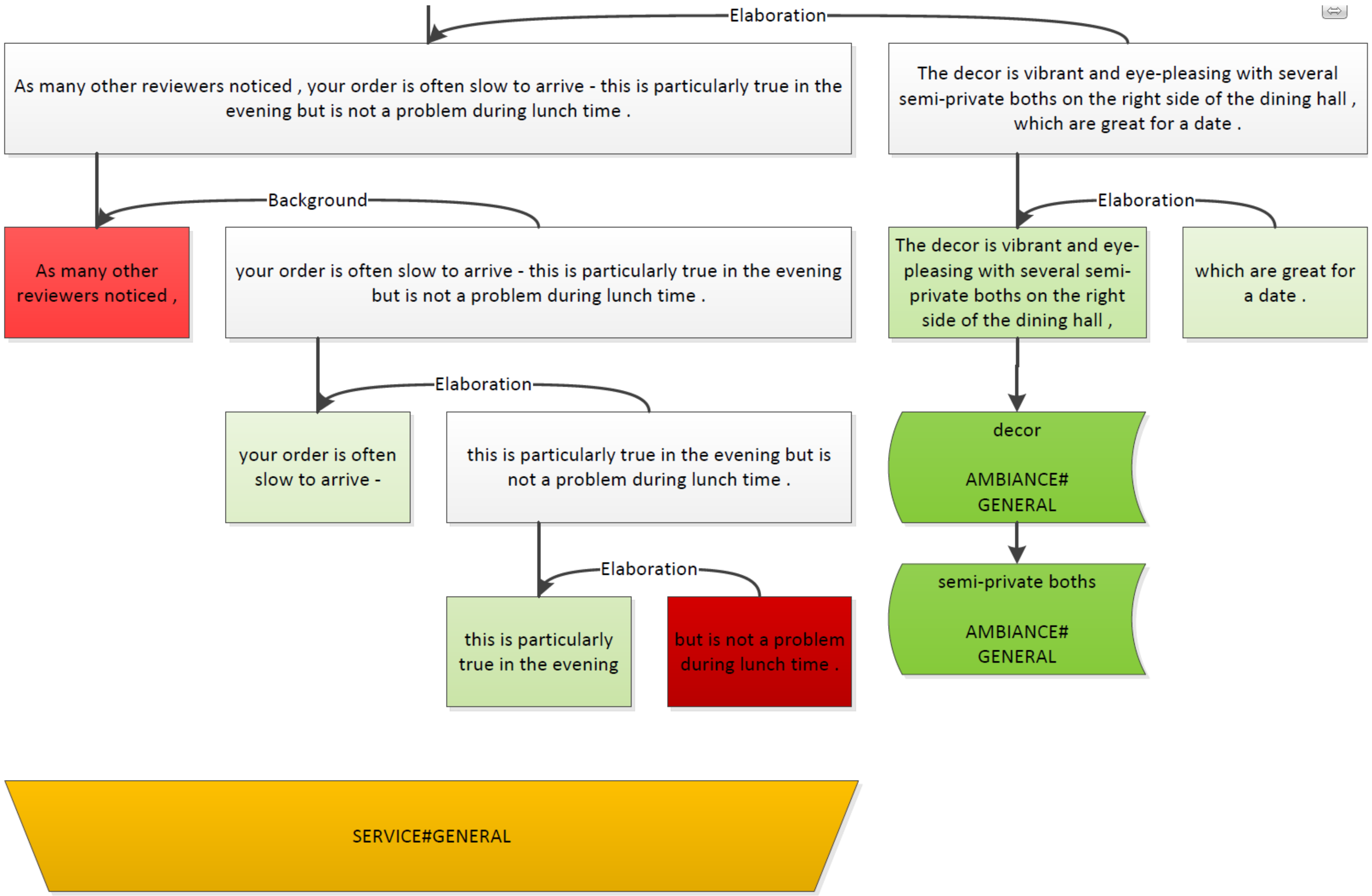
Failure Analysis

- No neutral case
- Context tree is not always correct (too large)
- Negations and amplifiers are not handled
- Aspect sentiment computed w.r.t. root of context tree mostly ignores contrasting relations

Failure Analysis– Example



Failure Analysis – Example



Failure Analysis – Example