UNIVERSITY OF OXFORD

Distributional Models of Meaning Mini-project

Candidate number: 692711

Hilary term

Structure of mini-project:

1. Introduction.
2. Linguistic data and brief explanation for basis nouns and values.
3. General derivation for calculating complex noun and sentence vectors.
4. Specific calculation of certain complex noun phrases and sentences.
5. Comparison of noun and sentence vectors.
6. Ambiguous noun construction and example sentences.
7. New methods for incorporating adjectives, certain conjunctions and certain punctuation marks into the categorical distributional model.
8. Final comparison of all calculated noun and sentence vectors.

1 INTRODUCTION

The entire vocabulary of this mini-project is based off of a popular TV show and comic from the 1980's and 1990's called the Teenage Mutant Ninja Turtles.

The Teenage Mutant Ninja Turtles are a team of four anthropomorphic crime-fighting turtles, Leonardo, Michelangelo, Raphael and Donatello, who live in the sewers of New York City, and who were trained to master ninjutsu by their anthropomorphic rat sensei Splinter. Along with splinter and their friend Casey, the Teenage Mutant Ninja Turtles (or TMNT for short), battle petty criminals, evil overlords, mutated creatures, and alien invaders while attempting to remain hidden from society. In particular, the main recurring nemesis of the TMNT is a samurai named Shredder, and his evil sidekick Karai. When the TMNT are not battling crime, they are usually appeasing their voracious appetite for pizza, especially Michelangelo who loves pizza the most.

The reason for picking this universe is the fact that the main characters, Leonardo, Michelangelo, Donatello, and Raphael have all been named after four major renaissance artists, and hence it makes it simple to model ambiguity in the later portion of the project.

The linguistic distribution chosen for sentence meaning calculations, which is explicitly presented in the following section, ranges over 19 basis vectors and 6 verbs–4 transitive and 2 intransitive.

Figure 1.1:

Drawing of the Teenage Mutant Ninja Turtles[1]

[1]picture from: http://www.panelsandpixels.com/wp-content/uploads/2014/10/TMNT-Cartoon.jpeg

2 LINGUISTIC DATA FOR $S = \mathbb{R}^2$

Here we present the data used in our linguistic universe of TMNT that allows us to make calculations regarding certain phrases and sentences. In particular we note the following:

- We have opted to use a 2-dimensional meaning space, $\mathbb{R}^2$, for sentences. In this scenario, e_1 roughly translates to "degree of truth" whereas e_2 roughly translates to "degree of falseness". So for example, in the verb "eats", the component for the turtle, "Lonesome George", eating "pizza" has a sentence value of $(0, 2)$ to represent that it is very false that Lonesome George eats pizza in our universe.

- Table 1 gives all the information on basis nouns and compound nouns that will be necessary for the rest of the project. In particular, the left-most column is composed of all basis nouns, N, and each other column represents a compound noun, whose entries represent coefficients of relevant basis nouns in that combination.

- In addition we have added three more turtles to the list of nouns to make calculations more interesting. Lonesome George was the last male Pinta Island tortoise, Bowser is Mario's arch-nemesis from Super Mario Bros. and Koopa is one of Bowser's underlings from the same series.

- We wish to highlight the fact that we distinguish between the anthropomorphic turtles and renaissance artists by use of subscripts i.e. $leonardo_t$ refers to the turtle, and $leonardo_p$ refers to the painter.

- Tables 2 to 5 give relevant information for transitive verbs, "Fights", "Eats", "Masters", and "Paints". These sub matrices of the much larger $N \otimes N$, contain all non-zero truth values for verb combinations.

- Finally, table 6 holds the relevant information for the intransitive verbs, "lives in New York" and "is evil". Since these are intransitive verbs, we can put more than one in a single matrix, where each entry represents the truth value of a specific basis vector under that verb.

Table 1, Nouns:

Basis Nouns	Human	Turtles	Animals	TMNT	bad guys	Painters
$leonardo_t$	0.5	1	0.5	2	0	0
$michelangelo_t$	0.5	1	0.5	2	0	0
$donatello_t$	0.5	1	0.5	2	0	0
$raphael_t$	0.5	1	0.5	2	0	0
$splinter$	0.5	0	0.5	0	0	0
$shredder$	1	0	0	0	16	0
$casey$	1	0	0	0	0	0
$karai$	1	0	0	0	8	0
$leonardo_p$	1	0	0	0	0	1
$michelangelo_p$	1	0	0	0	0	1
$donatello_p$	1	0	0	0	0	1
$raphael_p$	1	0	0	0	0	1
$lonesomegeorge$	0	1	1	0	0	0
$bowser$	0.2	1	0.8	0	0	0
$koopa$	0.2	1	0.8	0	0	0
$pizza$	0	0	0	0	0	0
$ninjitsu$	0	0	0	0	0	0
$painting$	0	0	0	0	0	0
$masterpieces$	0	0	0	0	0	0

Table 2, Transitive Verb "Fights":

	l_t	m_t	d_t	r_t	$splinter$	$shredder$	$casey$	$karai$
$leonardo_t$	(0,0)	(0,0)	(0,0)	(0,0)	(0,1)	(2,0)	(0,1)	(1,0)
$michelangelo_t$	(0,0)	(0,0)	(0,0)	(0,0)	(0,1)	(2,0)	(0,1)	(1,0)
$donatello_t$	(0,0)	(0,0)	(0,0)	(0,0)	(0,1)	(2,0)	(0,1)	(1,0)
$raphael_t$ (0,0)	(0,0)	(0,0)	(0,0)	(0,0)	(0,1)	(2,0)	(0,1)	(1,0)
$splinter$ (0,0)	(0,0)	(0,0)	(0,0)	(0,0)	(0,0)	(1,0)	(0,1)	(1,0)
$shredder$ (2,0)	(2,0)	(2,0)	(2,0)	(2,0)	(2,0)	(0,0)	(1,0)	(0,1)
$casey$ (0,1)	(0,1)	(0,1)	(0,1)	(0,1)	(0,1)	(1,0)	(0,0)	(1,0)
$karai$ (1,0)	(1,0)	(1,0)	(1,0)	(1,0)	(1,0)	(0,1)	(1,0)	(0,0)
$bowser$	(0,0)	(0,0)	(0,0)	(0,0)	(0,0)	(0,0.1)	(0,0)	(0,0)
$koopa$	(0,0)	(0,0)	(0,0)	(0,0)	(0,0)	(0,0.1)	(0,0)	(0,0)

Transitive Verb "Eat":

	$pizza$
$leonardo_t$	(2,0)
$michelangelo_t$	(4,0)
$donatello_t$	(2,0)
$raphael_t$	(2,0)
$splinter$	(1,0)
$leonardo_p$	(0.1,0)
$michelangelo_p$	(0.2,0)
$donatello_p$	(0.2,0)
$raphael_p$	(0.1,0)
$lonesomegeorge$	(0,2)
$bowser$	(1,0)
$koopa$	(0.5,0)

Transitive Verb "Masters":

	$ninjitsu$	$painting$
$leonardo_t$	(2,0)	(0,0)
$michelangelo_t$	(2,0)	(0,0)
$donatello_t$	(2,0)	(0,0)
$raphael_t$	(2,0)	(0,0)
$splinter$	(3,0)	(0,0)
$shredder$	(2,0)	(0,0)
$michelangelo_p$	(0,0)	(0,0)
$donatello_p$	(0,0)	(0,0)
$raphael_p$	(0,0)	(0,0)
$lonesomegeorge$	(0,0)	(0,0)
$bowser$	(1,0)	(0,0)
$koopa$	(0.5,0)	(0,0)

Transitive Verb "Paints":

	$pizza$	$masterpieces$
$leonardo_t$	(0,0)	(0,2)
$michelangelo_t$	(1,0)	(0,1)
$donatello_t$	(0,0)	(0,2)
$raphael_t$	(0,0)	(0,1)
$splinter$	(0,0)	(0,1)
$leonardo_p$	(0,0)	(5,0)
$michelangelo_p$	(0,0)	(5,0)
$donatello_p$	(0,0)	(5,0)
$raphael_p$	(0,0)	(5,0)
$lonesomegeorge$	(0,0)	(0,0)
$bowser$	(0,0)	(0,0)
$koopa$	(0,0)	(0,0)

Intransitive Verbs "Lives in NYC" and "Is evil":

	"lives in NYC"	"Is evil"
$leonardo_t$	(0.5,0.5)	(0,0)
$michelangelo_t$	(0.5,0.5)	(0,0)
$donatello_t$	(0.5,0.5)	(0,0)
$raphael_t$	(0.5,0.5)	(0,0)
$splinter$	(0.5,0.5)	(0,0)
$shredder$	(0.5,0.5)	(1,0)
$karai$	(0.25,0.75)	(1,0)
$casey$	(075,0.25)	(0,0)
$raphael_p$	(0,0)	(0,0)
$lonesomegeorge$	(0,0)	(0,0)
$bowser$	(0,0)	(1,0)
$koopa$	(0,0)	(0.5,0)

3 Algebra for important constructions:

The following calculations simply flesh out the relevant algebra for the rest of the project, where the algebra comes from the category-theoretic paradigm set in the papers we read during the course . The only main difference here from our calculations in class is that now we have to involve the frobenius co-unit i_s in our calculations as we are dealing with higher-dimensional sentences spaces.

3.1 Subject-relative clause

Phrase = subject who verb object

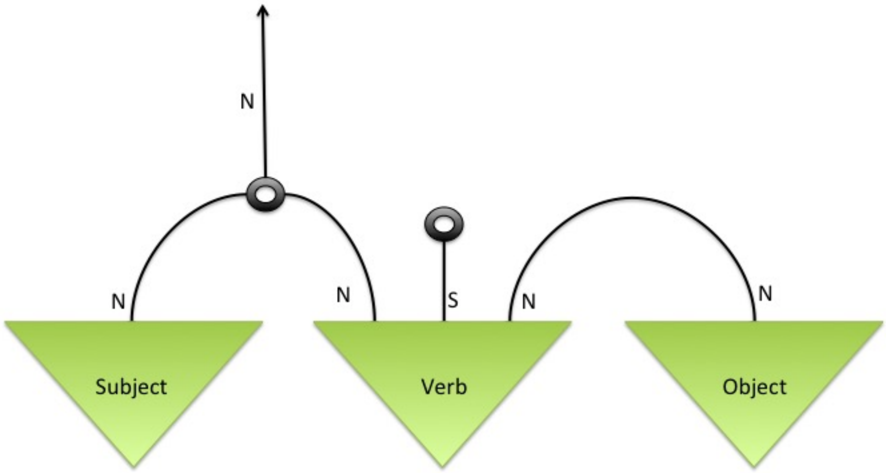

Figure 3.1:

$phrase = \mu_N \otimes i_s \otimes \epsilon_N(subject \otimes verb \otimes object)$

In the following expression i ranges over the basis nouns in the support of the subject, j,k range over all possible basis noun pairs, and l ranges over all basis nouns in the support of the object at hand.

$= \mu_N \otimes i_s \otimes \epsilon_N((\sum \alpha_i n_i) \otimes (\sum \beta'_{j,k} n_j \otimes s_{j,k} \otimes n_k) \otimes \sum_l \gamma_l n_l)$
$= \mu_N \otimes i_s \otimes \epsilon_N(\sum(\alpha_i \beta'_{j,k} \gamma_l) n_i \otimes n_j \otimes s_{j,k} \otimes n_k \otimes n_l)$

$i_s(s_{j,k}) = \pi_1(s_{j,k}) + \pi_2(s_{j,k})$ by definition.
Let us define $\beta_{j,k} = \beta'_{j,k} i_s(s_{j,k})$ so that we can simply the expression as follows:

$= \sum(\alpha_i \beta_{j,k} \gamma_l) \mu_N(n_i \otimes n_j) \otimes \epsilon_N(n_k \otimes n_l)$
$= \sum(\alpha_i \beta_{j,k} \gamma_l) \delta_{ij} n_i * \delta_{kl}$
$= \sum(\alpha_i \beta_{i,l} \gamma_l) n_i$

3.2 Object-relative clause:

Phrase = object whom subject verb

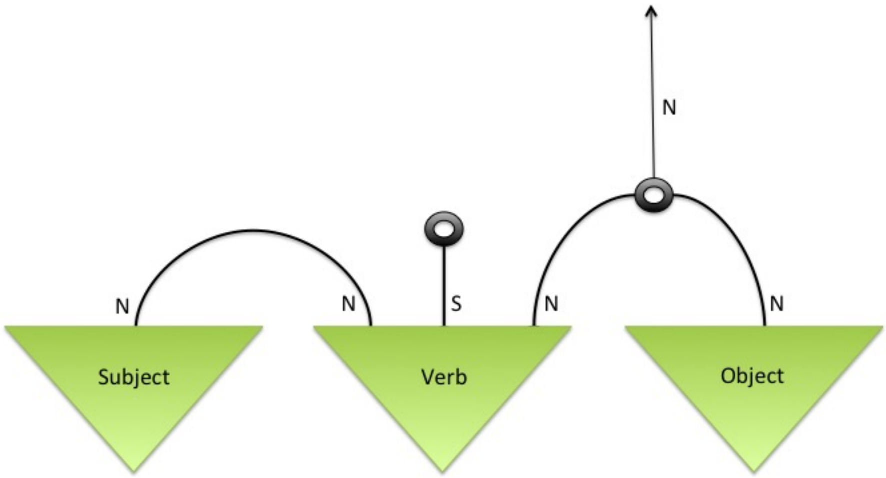

Figure 3.2:

$phrase = \epsilon_N \otimes i_s \otimes \mu_N(subject \otimes verb \otimes object)$

In the following expression i ranges over the basis nouns in the support of the subject, j, k range over all possible basis noun pairs, and l ranges over all basis nouns in the support of the object at hand.

$= \epsilon_N \otimes i_s \otimes \mu_N((\sum \alpha_i n_i) \otimes (\sum \beta'_{j,k} n_j \otimes s_{j,k} \otimes n_k) \otimes \sum_l \gamma_l n_l)$
$= \epsilon_N \otimes i_s \otimes \mu_N(\sum (\alpha_i \beta'_{j,k} \gamma_l) n_i \otimes n_j \otimes s_{j,k} \otimes n_k \otimes n_l)$

$i_s(s_{j,k}) = \pi_1(s_{j,k}) + \pi_2(s_{j,k})$ by definition.
Let us define $\beta_{j,k} = \beta'_{j,k} i_s(s_{j,k})$ so that we can simply the expression as follows:

$= \sum (\alpha_i \beta_{j,k} \gamma_l) \epsilon_N(n_i \otimes n_j) \otimes \mu_N(n_k \otimes n_l)$
$= \sum (\alpha_i \beta_{j,k} \gamma_l) \delta_{kl} n_l * \delta_{ij}$
$= \sum (\alpha_i \beta_{i,l} \gamma_l) n_l$

3.3 Subject-relative clause with an intransitive verb:

Phrase = subject who verb

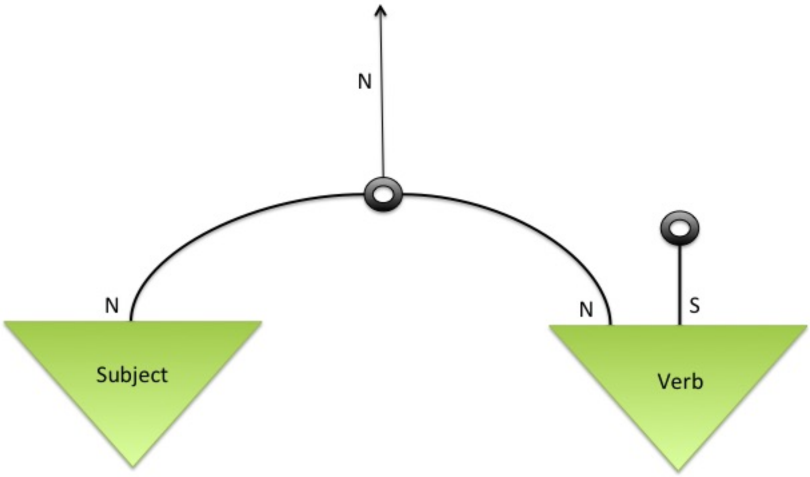

Figure 3.3:

$phrase = \mu_N \otimes i_s(subject \otimes verb)$

In the following expression i ranges over the basis nouns in the support of the subject, j ranges over all possible basis nouns.

$= \mu_N \otimes i_s((\sum \alpha_i n_i) \otimes (\sum \beta'_j n_j \otimes s_j))$
$= \mu_N \otimes i_s(\sum (\alpha_i \beta'_j) n_i \otimes n_j \otimes s_j)$

$i_s(s_j) = \pi_1(s_j) + \pi_2(s_j)$ by definition.
Let us define $\beta_j = \beta'_j i_s(s_j)$ so that we can simply the expression as follows:

$= \sum (\alpha_i \beta_j) \mu_N(n_i \otimes n_j))$
$= \sum (\alpha_i \beta_j) \delta_{ij} n_i$
$= \sum (\alpha_i \beta_j) n_i$

3.4 Transitive verb:

Phrase = subject verb object

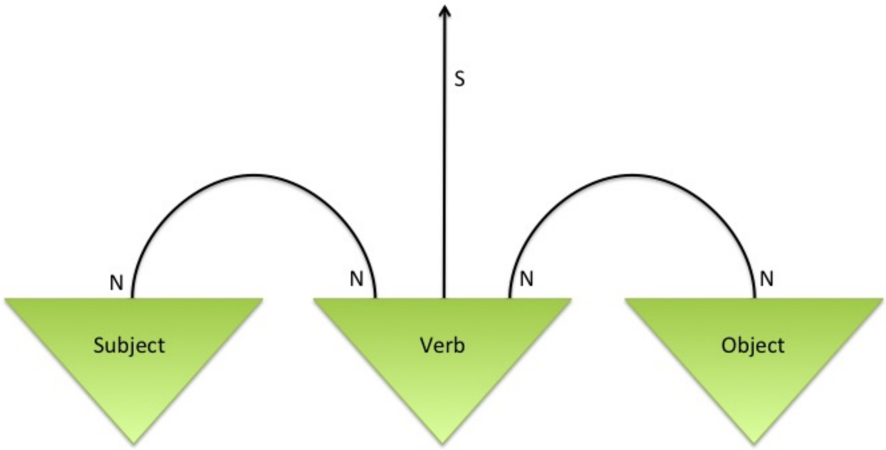

Figure 3.4:

$phrase = \epsilon_N \otimes 1_s \otimes \epsilon_N(subject \otimes verb \otimes object)$

In the following expression i ranges over the basis nouns in the support of the subject, j,k range over all possible basis noun pairs, and l ranges over all basis nouns in the support of the object at hand.

$$
\begin{aligned}
&= \epsilon_N \otimes 1_s \otimes \epsilon_N((\sum \alpha_i n_i) \otimes (\sum n_j \otimes s_{j,k} \otimes n_k) \otimes \sum_l \gamma_l n_l) \\
&= \epsilon_N \otimes 1_s \otimes \epsilon_N(\sum (\alpha_i \gamma_l) n_i \otimes n_j \otimes s_{j,k} \otimes n_k \otimes n_l) \\
&= \sum (\alpha_i \gamma_l) \epsilon_N(n_i \otimes n_j) \epsilon_N(n_k \otimes n_l) s_{j,k} \\
&= \sum (\alpha_i \gamma_l) \delta_{ij} \delta_{kl} s_{j,k} \\
&= \sum (\alpha_i \gamma_l) s_{i,l}
\end{aligned}
$$

3.5 INTRANSITIVE VERB:

Phrase = subject verb

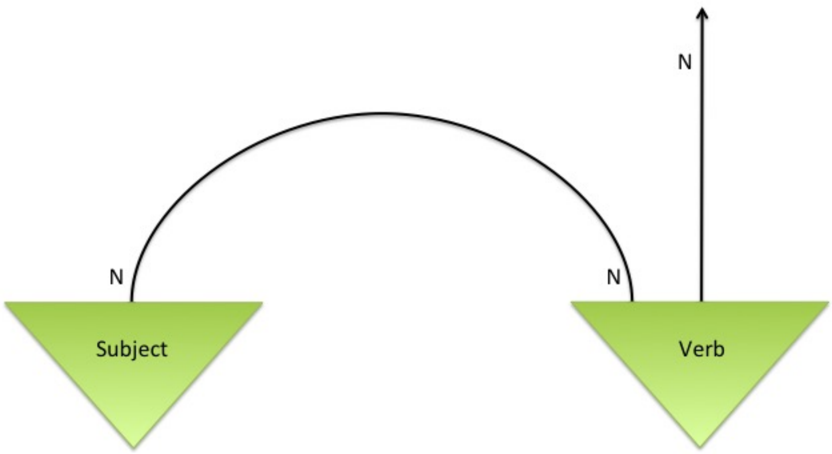

Figure 3.5:

$phrase = \epsilon_N \otimes 1_s(subject \otimes verb)$

In the following expression i ranges over the basis nouns in the support of the subject, j ranges over all possible basis nouns.

$= \epsilon_N \otimes 1_s((\sum \alpha_i n_i) \otimes (\sum n_j \otimes s_j))$
$= \epsilon_N \otimes 1_s(\sum(\alpha_i) n_i \otimes n_j \otimes s_j)$
$= \sum(\alpha_i)\epsilon_N(n_i \otimes n_j)s_j$
$= \sum(\alpha_i)\delta_{ij} s_j$
$= \sum \alpha_i s_i$

4 Calculations for specific nouns and sentences

Here we reap the benefits of the algebraic expressions in section 2 to actually calculate compound noun vectors and sentence vectors in our TMNT universe.

4.1 Compound nouns with relative pronouns:

Teenage Mutant Ninja Turtles (TMNT):= "Turtles who master ninjitsu who live in NYC"

According to the algebra in section 2, the expression for this phrase is the following:
"Turtles who master ninjitsu" = $(\sum_i \alpha_i \beta_{i,n} n_i)$

- i ranges over the support of compound noun, "Turtles"
- α_i are the weights for the basis nouns in the support of the compound noun "Turtles"
- $\beta_{i,n} = i_s(s_{i,n})$ is the image under the frobenius co-unit of the sentence vector for a specific pair of basis nouns in the expression for the transitive verb $master$.
- n is the index for $n_n = ninjitsu$, since it is a basis vector

In our specific universe, this expression evaluates to:
$(1*i(2,0))leonardo_t+(1*i(2,0))michelangelo_t+(1*i(2,0))donatello_t+(1*i(2,0))raphael_t+(1*i(1,0))bowser+(1*i(0.5,0))koopa$

$= 2(leonardo_t + michelangelo_t + donatello_t + raphael_t) + bowser + 0.5koopa$

Now, in order to finish the phrase we analyze "who live in NYC", where "live in NYC" is treated as an intransitive noun whose parameters can be found in the linguistic tables.
We notice that since $bowser$ and $koopa$ do not have entries in the tensor for "live in NYC", their coefficients vanish under the frobenius co-unit. Furthermore, for the other elements of the support, the frobenius co-unit only acts on $i(0.5,0.5) = 1$, hence what remains is the following:

$TMNT = 2(leonardo_t + michelangelo_t + donatello_t + raphael_t)$

Note that our original definition of TMNT in the table coincides with this calculation.

Bad guys:= Humans who fights TMNT who are evil

According to our algebra in section 2, the expression for the first nested phrase is:
Humans who fight TMNT = $\sum_{i,l}(\alpha_i \beta_{i,l} \gamma_l) n_i$

- i ranges over the support of compound noun, "Humans"
- α_i are the weights for the basis nouns in the support of the compound noun "Humans"
- $\beta_{i,l} = i_s(s_{i,l})$ is the image under the frobenius co-unit of the sentence vector for a specific pair of basis nouns in the expression for the transitive verb $fights$.
- l ranges over the support of compound noun, "TMNT" which was calculated on the previous page
- γ_l are the weights for the basis nouns in the support of the compound noun "TMNT" which was calculate on the previous page

In our specific universe, this expression evaluates to:
$4 * \left(1 * i(2,0) * 2shredder_h + 1 * i(1,0) * 2kasai + 1 * i(0,1) * 2casey\right)$
$= 16shredder + 8karai + 8caseyjones$

It is interesting to note that noun, $casey$ appears, in spite of having an orthogonal sentence value for fighting any member of TMNT. The reason for this is that the frobenius map is indiscriminate towards any basis vector, even if in this case they mean "true" and "false" in some sense.

Now we can proceed to calculate the meaning of the entire phrase using the expression we have just calculated and the algebra from section 2. Note we re-use the notation, however all of the summands in this expression mean different things than before.

(Humans who fights TMNT) who are evil= $\sum(\alpha_i \beta_j) n_i$

- i ranges over the support of the recently calculated noun, "Humans who fights TMNT"
- α_i are the weights for the basis nouns in the support of the compound noun "Human who fights TMNT"
- $\beta_j = i_s(s_j)$ is the image under the frobenius co-unit of the sentence vector for a specific noun in the expression for the intransitive verb $is-evil$.

In our specific universe, this expression evaluates to:
$Badguys = 1 * 16shredder + 1 * 8karai + 0 * 8caseyjones$

$= 16shredder + 8karai$

Here Casey Jones has disappeared from the expression because he has a value of 0 for the transitive verb "is-evil". Were he to have a (0,1) value, it would be a different story.

Animals who shredder fights

According to our algebra in section 2, the expression for this phrase is the following:
Animals who shredder fights= $\sum(\beta_{s,l}\gamma_l)n_l$

- s is the index for basis noun n_s representing shredder
- $\beta_{i,l} = i_s(s_{i,l})$ is the image under the frobenius co-unit of the sentence vector for a specific pair of basis nouns in the expression for the transitive verb $fights$.
- l ranges over the support of compound noun, "Animals" which is directly in the table.
- γ_l are the weights for the basis nouns in the support of the compound noun "Animals" which is directly in the table.

In our specific universe, this expression evaluates to:
$0.5*i(2,0)leonardo_t+0.5*i(2,0)michelangelo_t+0.5*i(2,0)donatello_t+0.5*i(2,0)raphael_t+0.5*i(2,0)splinter_r$

$= leonardo_t + michelangelo_t + donatello_t + raphael_t + splinter_r$

$= 0.5(TMNT) + splinter_r$

Animals who master ninjitsu

According to our algebra in section 2, the expression for this phrase is the following:
Animals who master ninjitsu = $(\sum_i \alpha_i\beta_{i,n}n_i)$

- i ranges over the support of compound noun, "Animals"
- α_i are the weights for the basis nouns in the support of the compound noun "Animals"
- $\beta_{i,n} = i_s(s_{i,n})$ is the image under the frobenius co-unit of the sentence vector for a specific pair of basis nouns in the expression for the transitive verb $master$.
- n is the index for $n_n = ninjitsu$, since it is a basis vector

In our specific universe, this expression evaluates to:
$0.5*i(2,0)leonardo_t+0.5*i(2,0)michelangelo_t+0.5*i(2,0)donatello_t+0.5*i(2,0)raphael_t+0.5*i(3,0)splinter_r+0.8*i(1,0)bowser+0.8*i(0.5,0)koopa$

$= leonardo_t+michelangelo_t+donatello_t+raphael_t+1.5splinter_r+0.8bowser+0.4koopa$

$= 0.5TMNT + 1.5splinter_r + 0.8bowser + 0.4koopa$

Painters who paint masterpieces

According to our algebra in section 2, the expression for this phrase is the following:
Painters who paint masterpieces = $(\sum_i \alpha_i \beta_{i,r} n_i)$

- i ranges over the support of compound noun, "Painters"
- α_i are the weights for the basis nouns in the support of the compound noun "Painters"
- $\beta_{i,n} = i_s(s_{i,n})$ is the image under the frobenius co-unit of the sentence vector for a specific pair of basis nouns in the expression for the transitive verb $paint$.
- r is the index for $n_r = masterpieces$, since it is a basis vector

In our specific universe, this expression evaluates to:
$1 * i(5,0) leonardo_p + 1 * i(5,0) michelangelo_p + 1 * i(5,0) donatello_p + 1 * i(5,0) raphael_p$

$= 5(leonardo_p + michelangelo_p + donatello_p + raphael_p)$

Animals who eat pizza

According to our algebra in section 2, the expression for this phrase is the following:
Animals who eat pizza = $(\sum_i \alpha_i \beta_{i,p} n_i)$

- i ranges over the support of compound noun, "Animals"
- α_i are the weights for the basis nouns in the support of the compound noun "Animals"
- $\beta_{i,n} = i_s(s_{i,n})$ is the image under the frobenius co-unit of the sentence vector for a specific pair of basis nouns in the expression for the transitive verb eat.
- p is the index for $n_p = pizza$, since it is a basis vector

In our specific universe, this expression evaluates to:
$0.5 * i(2,0) leonardo_t + 0.5 * i(4,0) michelangelo_t + 0.5 * i(2,0) donatello_t + 0.5 * i(2,0) raphael_t + 0.5 * i(1,0) splinter + 1 * i(0,2) lonesomegeorge + 0.8 * i(1,0) bowser + 0.8 * i(0.5,0) koopa$

$= leonardo_t + 2 michelangelo_t + donatello_t + raphael_t + 0.5 splinter + 2 lonesomegeorge + 0.8 bowser + 0.4 koopa$

4.2 Sentences:

TMNT fight shredder

According to our algebra in section 2, the expression for this phrase is the following:
$\sum(\alpha_i) s_{i,s}$

- i ranges over the basis nouns in the support of "TMNT", which we calculated earlier.

- s is the index for n_s the basis noun representing "Shredder".
- $s_{i,s}$ is the sentence vector component for the i, s pair in the verb "fight", which are in the table.

In our specific universe, this expression evaluates to:
$2*(2,0)+2*(2,0)+2*(2,0)+2*(2,0)$

$= (16,0) \in \mathbb{R}^2$

The animals who master ninjutsu fight shredder

According to our algebra in section 2, the expression for this phrase is the following:
$\sum(\alpha_i)s_{i,s}$

- i ranges over the basis nouns in the support of "Animals who master ninjitsu", which we calculated earlier.
- s is the index for n_s the basis noun representing "Shredder".
- $s_{i,s}$ is the sentence vector component for the i, s pair in the verb "master", which are in the table.

In our specific universe, this expression evaluates to:
$1*(2,0)+1*(2,0)+1*(2,0)+1*(2,0)+1.5*(2,0)+0.8*(0,0.1)+0.4*(0,0.1)$

$= (15,0.12) \in \mathbb{R}^2$

The painters who paint masterpieces eat pizza

According to our algebra in section 2, the expression for this phrase is the following:
$\sum(\alpha_i)s_{i,p}$

- i ranges over the basis nouns in the support of "Humans who paint masterpieces", which is in the table.
- p is the index for n_p the basis noun representing "Pizza".
- $s_{i,p}$ is the sentence vector component for the i, p pair in the verb "eat", which are in the table.

In our specific universe, this expression evaluates to:
$5*(0.1,0)+5*(0.2,0)+5*(0.2,0)+5*(0.1,0)$

$= (3,0) \in \mathbb{R}^2$

Animals eat pizza

According to our algebra in section 2, the expression for this phrase is the following:
$\sum(\alpha_i\gamma_l)s_{i,l}$

- i ranges over the basis nouns in the support of "Animals", which are in the table.
- l ranges over the basis nouns in the support of "Shredder", which we calculated earlier.
- $s_{i,l}$ is the sentence vector component for the i,l pair in the verb "fight", which are in the table.

In our specific universe, this expression evaluates to:
$0.5*(2,0)+0.5*(4,0)+0.5*(2,0)+0.5(2,0)+0.5(1,0)+1*(0,2)+0.8*(1,0)+0.8*(0.5,0)$

$=(6.7,2)\in\mathbb{R}^2$

The animals who eat pizza fight bad guys

According to our algebra in section 2, the expression for this phrase is the following:
$\sum(\alpha_i\gamma_l)s_{i,l}$

- i ranges over the basis nouns in the support of "Animals who eat pizza", which we calculated earlier.
- l ranges over the basis nouns in the support of "bad guys", which was calculated earlier.
- $s_{i,l}$ is the sentence vector component for the i,l pair in the verb "fight", which are in the table.

In our specific universe, this expression evaluates to:
$1*16*(2,0)+1*8*(1,0)+2*16*(2,0)+2*8*(1,0)+1*16*(2,0)+1*8*(1,0)+1*16*(2,0)+1*8*(1,0)+0.5*16*(1,0)+0.5*8*(1,0)+0.8*16*(0,0.1)+0.4*16*(0,0.1)$

$=(212,1.92)\in\mathbb{R}^2$

5 Tables of similarity amongst nouns and amongst sentences

List of compound nouns:

- Noun_1 = TNMT
- Noun_2 = Bad guys
- Noun_3 = Animals who shredder fights
- Noun_4 = Animals who master ninjitsu

- Noun_5 = Painters who paint masterpieces
- Noun_6 = Animals who eat pizza
- Noun_7 = Humans
- Noun_8 = Turtles
- Noun_9 = Animals

Noun inner product table

	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8	N_9
N_1	16	0	8	8	0	10	4	8	4
N_2	0	320	0	0	0	0	24	0	0
N_3	8	0	5	5.5	0	5.5	2.5	4	2.5
N_4	8	0	5.5	7.05	0	6.55	2.99	5.2	3.71
N_5	0	0	0	0	100	0	20	0	0
N_6	10	0	5.5	6.55	0	12.05	2.99	8.2	5.71
N_7	4	24	2.5	2.99	20	2.99	8.33	2.4	1.57
N_8	8	0	4	5.2	0	8.2	2.4	7	4.6
N_9	4	0	2.5	3.71	0	5.71	1.57	4.6	3.53

Noun similarity measure ($cos(\theta)$)

	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8	N_9
N_1	1	0	.8944	.7535	0	.7202	.3465	.7559	.5322
N_2	0	1	0	0	0	0	.4649	0	0
N_3	.8944	0	1	.9264	0	.7086	.3874	.6761	.5951
N_4	.7532	0	.9264	1	0	.7106	.3902	.7402	.7437
N_5	0	0	0	0	1	0	.6930	0	0
N_6	.7202	0	.7086	.7106	0	1	.2984	.8928	.8755
N_7	.3465	.4649	.3874	.3902	.6930	.2984	1	.3143	.2895
N_8	.7559	0	.6761	.7402	0	.8928	.3143	1	.9254
N_9	.5322	0	.5951	.7437	0	.8755	.2895	.9254	1

List of computed sentences:

- Sentence_1 = TNMT fight shredder.
- Sentence_2 = The animals who master ninjutsu fight shredder.
- Sentence_3 = The painters who paint masterpieces eat pizza.
- Sentence_4 = Animals eat pizza.
- Sentence_5 = The animals who eat pizza fight bad guys.

Sentence inner product table

	S_1	S_2	S_3	S_4	S_5
S_1	256	240	48	107.2	3392
S_2	240	225.01	45	100.74	3180.2
S_3	48	45	9	20.1	636
S_4	107.2	100.74	20.1	48.89	1424.2400
S_5	3392	3180.2	636	1424.24	44947.67

Sentence similarity measure ($cos(\theta)$)

	S_1	S_2	S_3	S_4	S_5
S_1	1	1	1	.9582	1
S_2	1	1	1	.9605	1
S_3	1	1	1	.9582	1
S_4	.9582	.9605	.9582	1	.9608
S_5	1	1	1	.9608	1

Note: Since most sentences for now are in the direction of $e_1 \in \mathbb{R}^2$, the similarity measure between them is 1 as can be seen in the table. In the later sections we produce new sentences that exhibit more orthogonal meanings to compare. In this case, distinguishing meaning between generally "true" sentences is by their degree of truth, namely their e_1-coordinate value.

6 Ambiguous Nouns

Construction of ambiguous nouns:

Here we represent the ambiguity in the names of individuals in the TMNT universe. Although previously we had made a written distinction between say the renaissance artists and the anthropomorphic turtles (by use of subscripts), here we introduce their full on ambiguity using density matrices.

Our ambiguous noun is "Leonardo" which can either mean $Leonardo_t$, the ninja turtle named Leonardo, or $Leonardo_p$, the renaissance painter named Leonardo. We give each of these possibilities equal probability, so that our density matrix is as follows:

$\rho(\text{Leonardo}) = \rho_L = leonardo_t \otimes leonardo_t + leonardo_p \otimes leonardo_p = l_t \otimes l_t + l_p \otimes l_p$

If we normalize the trace to 1 we get the following:
$\bar{\rho_L} = \frac{1}{2} l_t \otimes l_t + \frac{1}{2} l_p \otimes l_p$

In particular, this is a maximally mixed state, so that our entropy is as follows:
$S(\bar{\rho_L}) = \frac{1}{2} ln(2) + \frac{1}{2} ln(2) = ln(2) = 0.6931$

Ambiguity reducing sentences:

We calculate the meaning of three distinct sentences for each ambiguous subject and then we compare their entropy.

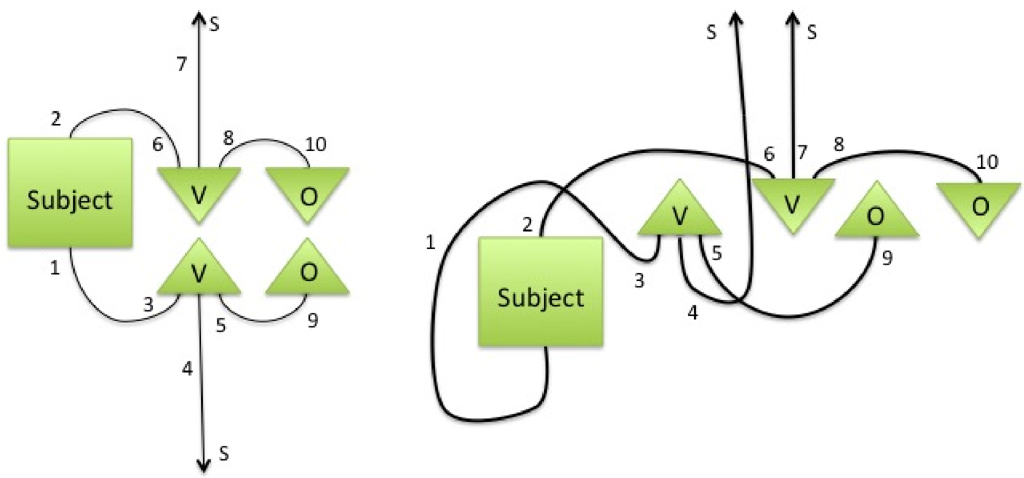

Figure 6.1: Densiy matrix evaluation procedure

It is important to note that the relevant permutation of indices before all contraction operations is the following:
$\sigma : (1,2,3,4,5,6,7,8,9,10) \rightarrow (1,3,2,6,4,7,5,9,8,10)$

This makes our notation much more compact when it comes to evaluation:
$(\epsilon_N \otimes \epsilon_N \otimes 1_s \otimes 1_s \otimes \epsilon_N \otimes \epsilon_N) \circ \sigma$

"Leonardo" Calculations:

Leonardo eats pizza:

Here n_p is the basis noun for pizza

Let ρ_1 be the unnormalized density matrix for this sentence, calculated as follows:
$(\epsilon_N \otimes \epsilon_N \otimes 1_s \otimes 1_s \otimes \epsilon_N \otimes \epsilon_N) \circ \sigma\Big(\sum l_i \otimes l_i \otimes (n_j \otimes s_{j,k} \otimes n_k) \otimes (n_{j'} \otimes s_{j',k'} \otimes n'_k) \otimes n_p \otimes n_p)\Big)$
$= \epsilon_N \otimes \epsilon_N \otimes 1_s \otimes 1_s \otimes \epsilon_N \otimes \epsilon_N \Big(\sum l_i \otimes n_j \otimes l_i \otimes n_{j'} (\otimes s_{j,k} \otimes s_{j',k'}) \otimes n_k \otimes n_p \otimes n_{k'} \otimes n_p\Big)$
$= \sum \epsilon_N(l_i \otimes n_j)\epsilon_N(l_i \otimes n_{j'})(s_{j,k} \otimes s_{j',k'})\epsilon_N(n_k \otimes n_p)\epsilon_N(n_{k'} \otimes n_p)$

If we say that $n_\alpha = leonardo_t$, and $n_\beta = leonardo_p$, we get the following:
$\rho_1 = s_{\alpha,p} \otimes s_{\alpha,p} + s_{\beta,p} \otimes s_{\beta,p}$
$= (2,0) \otimes (2,0) + (0.1,0) \otimes (0.1,0) = 4.01\big((1,0) \otimes (1,0)\big) = 4.01(e_1 \otimes e_1)$

It is interesting to note that although the sentence is referring to both Leonardos, it is unambiguous in the sense that only the e_1 vector appears in the tensor product. Indeed if we normalize to get a density matrix, we get that the unambiguous meaning of this sentence is that it is true. Of course, it is important to note that the different degrees of truth are contributing different values to the overall tensor product.

$\bar{\rho_1} = e_1 \otimes e_1$
$S(\bar{\rho_1}) = 0$

Leonardo paints masterpieces:

Here n_m is the basis noun for masterpieces

Let ρ_2 be the unnormalized density matrix for this sentence, calculated as follows:
$(\epsilon_N \otimes \epsilon_N \otimes 1_s \otimes 1_s \otimes \epsilon_N \otimes \epsilon_N) \circ \sigma\Big(\sum l_i \otimes l_i \otimes (n_j \otimes s_{j,k} \otimes n_k) \otimes (n_{j'} \otimes s_{j',k'} \otimes n'_k) \otimes n_m \otimes n_m)\Big)$
$= \epsilon_N \otimes \epsilon_N \otimes 1_s \otimes 1_s \otimes \epsilon_N \otimes \epsilon_N \Big(\sum l_i \otimes n_j \otimes l_i \otimes n_{j'} (\otimes s_{j,k} \otimes s_{j',k'}) \otimes n_k \otimes n_m \otimes n_{k'} \otimes n_m\Big)$
$= \sum \epsilon_N(l_i \otimes n_j)\epsilon_N(l_i \otimes n_{j'})(s_{j,k} \otimes s_{j',k'})\epsilon_N(n_k \otimes n_m)\epsilon_N(n_{k'} \otimes n_m)$

If we say that $n_\alpha = leonardo_t$, and $n_\beta = leonardo_p$, we get the following:
$\rho_2 = s_{\alpha,m} \otimes s_{\alpha,m} + s_{\beta,m} \otimes s_{\beta,m}$
$= (0,2) \otimes (0,2) + (5,0) \otimes (5,0) = 25(e_1 \otimes e_1) + 4(e_2 \otimes e_2)$
Unlike the previous example, this statement is ambiguous because of the orthogonal meanings that both Leonardos ascribe to painting masterpieces. If we normalize the operator, we get a density operator whose Von Neumann entropy we can calculate:

$\bar{\rho_2} = \frac{25}{29}(e_1 \otimes e_1) + \frac{4}{29}(e_2 \otimes e_2)$
$S(\bar{\rho_2}) = \frac{25}{29} ln(\frac{29}{25}) + \frac{4}{29} ln(\frac{29}{4}) = 0.4013$

Leonardo fights bad guys:

Suppose that this is the representation of bad guys:
$\sum \alpha_x n_x = 16 shredder + 8 karai$, as seen in the previous sections.

Then this is its unambiguous density matrix:
$\rho(badguys) = (\sum \alpha_x n_x) \otimes (\sum \alpha_y n_y) = \sum \alpha_x \alpha_y n_x \otimes n_y$

Let ρ_3 be the unnormalized density matrix the complete sentence, calculated as follows:
$(\epsilon_N \otimes \epsilon_N \otimes 1_s \otimes 1_s \otimes \epsilon_N \otimes \epsilon_N) \circ \sigma\Big(\sum l_i \otimes l_i \otimes (n_j \otimes s_{j,k} \otimes n_k) \otimes (n_{j'} \otimes s_{j',k'} \otimes n'_k) \otimes (\alpha_x \alpha_y n_x \otimes n_y))\Big)$
$= \epsilon_N \otimes \epsilon_N \otimes 1_s \otimes 1_s \otimes \epsilon_N \otimes \epsilon_N \Big(\sum (\alpha_x \alpha_y) l_i \otimes n_j \otimes l_i \otimes n_{j'} (\otimes s_{j,k} \otimes s_{j',k'}) \otimes n_k \otimes n_x \otimes n_{k'} \otimes n_y\Big)$
$= \sum (\alpha_x \alpha_y) \epsilon_N (l_i \otimes n_j) \epsilon_N (l_i \otimes n_{j'}) (s_{j,k} \otimes s_{j',k'}) \epsilon_N (n_k \otimes n_x) \epsilon_N (n_{k'} \otimes n_y)$

If we say that $n_1 = leonardo_t$, and $n_2 = leonardo_p$, we get the following:
$\sum (\alpha_x \alpha_y)(s_{1,x} \otimes s_{1,y}) + \sum (\alpha_x \alpha_y)(s_{2,x} \otimes s_{2,y})$

However, we notice that $s_{2,j} = 0$ for all j in the support of $badguys$, hence the second term in this sum vanishes and we are left with:
$\rho_3 = \sum (\alpha_x \alpha_y)(s_{1,x} \otimes s_{1,y}) = (\sum \alpha_x s_{1,x}) \otimes (\sum \alpha_y s_{1,y})$

But this is precisely the self tensor product of the meaning of the sentence "Leonardo$_t$ fights bad guys". Therefore we have lost all initial ambiguity because the painter Leonardo does fight bad guys at all, hence it must be the anthropomorphic hero turtle, Leonardo$_t$ whom we are talking about. In particular:
$S(\bar{\rho_3}) = 0$

7 Additional Constructions

Adjectives:

An adjective modifies a noun so that the aggregate of the two words becomes another noun. Since in English, adjectives come before the nouns that they modify, this means that in terms of Lambek pregroups, adjectives have the type nn^l.

As is done in our previous categorical settings of language, we take a strict monodical functor M from the category of pregroups to the category of finite-dimensional Hilbert spaces. In particular, this means that an adjective in Hilbert space, must be a vector from $M(nn^l) = N \otimes N^* \cong N \otimes N$

The question then is how to construct such a vector in the tensor space. One possibility is to use what we call a representative vector $n_a = \sum \alpha_i n_i \in N$, and let the adjective be expressed as $adj = \Delta(n_a) = \sum \alpha_i n_i \otimes n_i \in N \otimes N$, where we have used the frobenius co-multiplication, Δ. Then the way that we evaluate a phrase of the form "adjective noun" is as follows:

Process Diagram:

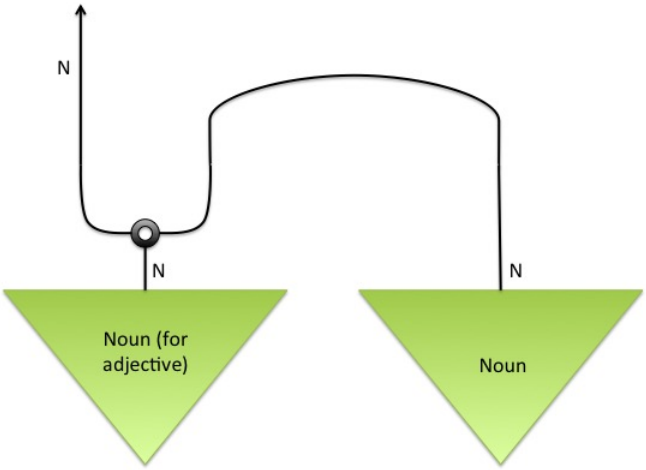

Figure 7.1: Adjective construction

Phrase = adjective noun
$phrase = 1_N \otimes \epsilon_N \Big(\Delta(n_a) \otimes noun\Big)$

Adjective examples:

We model the following two adjectives with their corresponding representative noun vectors:

- "Radical", with representative noun:
  $n_{radical} = TMNT + splinter + casey$
- "Dastardly", with representative noun:
  $n_{dastardly} = 3shredder + 2karai + 3bowser + koopa$

We calculate the meaning of the following phrases:

- "Radical turtles" = $(1_N \otimes \epsilon_N) \circ (\Delta \otimes 1_N)(n_{radical} \otimes turtles)$
  $= 2leonardo_t + 2michelangelo_t + 2donatello_t + 2raphael_t = TMNT$
- "Radical TMNT" = $(1_N \otimes \epsilon_N) \circ (\Delta \otimes 1_N)(n_{radical} \otimes TMNT)$
  $= 4leonardo_t + 4michelangelo_t + 4donatello_t + 4raphael_t = 2TMNT$

- "dastardly turtles" = $(1_N \otimes \epsilon_N) \circ (\Delta \otimes 1_N)(n_{dastardly} \otimes turtles)$
  $= 3bowser + koopa$
- "dastardly turtles fight bad guys"
  $= \epsilon_N \otimes 1_S \otimes \epsilon_N(dastardlyturtles \otimes fight \otimes badguys)$
  $= 3 * 16 * (0, 0.1) + 1 * 16 * (0, 0.1) = (0, 6.4) \in \mathbb{R}^2$

Comments:

It is easy to note that with this construction, an adjective modifies a noun by essentially magnifying its basis vector components according to the adjective representation vector. If this representation vector is thought of the archetype noun of that adjective characteristic, the construction magnifies those characteristic basis vectors and kills all others.

Furthermore, it is in some sense intuitive that an adjective be a diagonal matrix that multiplies basis nouns by a scalar. The reason for this is that if a basis vector, say n_j under some adjective, say $\Delta(n_\alpha)$ were not just scaled, but also a linear combination of other basis vectors, so that $1_N \otimes \epsilon_N(\Delta(n_\alpha) \otimes n_j) = \sum \beta_k n_k$, then it would seem that the basis vector should not be a pure basis vector in the first place, for it has components of other vectors in the case of it being paired with this specific adjective.

Finally, this construction also allows us to interestingly extract adjectives from intransitive verbs, so that we get an equivalent of the verb "$is - adjective$" for a specific adjective.

Suppose that we have an intransitive verb, $verb = \sum n_i s_i$, where n_i are basis nouns, and $s_i \in S$. We let $n_{verb} = 1_N \otimes i_s(verb) = \sum i_s(s_j) n_j$, where here i_s is the frobenius co-unit. Then we can consider an adjective made from this reference vector, $adj_v = \Delta(n_{verb})$. Most interestingly, it can be easily seen that under this construction, for any noun, the following two statements are equivalent:

"Adj$_v$ noun" = "Noun who verb"
Therefore this adjective construction is in a certain sense compatible with our previous notions of relative pronouns.

However, even though one can go from intransitive verb to adjective as just mentioned, the other direction is not quite as simple. The fact is that although the frobenius co-unit preserves a certain amount of information, a magnitude of sorts, it still destroys much more, namely how that magnitude is spread amongst different dimensions. Therefore in higher-dimensional sentence spaces there is no real way of going the other direction once information has been destroyed. In order to make an intransitive verb out of an adjective, I would need to make a vector out of a scalar, which is not possible in a unique way.

Conjunctions:

When the conjunction "and" is used between two nouns, its pregroup grammar type is clearly of the form $n^r nn^l$.

When passing to the distributional model via the strict monoidal functor, M, mentioned in class, this means that "and" has to be an element of $M(n^r nn^l) = N \otimes N \otimes N \cong \mathscr{L}(N \otimes N, N)$. This second expression however, is the same as the space of bilinear forms from $N \times N$ to N. As such, it becomes clear that the most reasonable candidate for "and" becomes the bilinear form representing vector addition on N.

Call the bilinear form $M(And) = b_{and,N}$
$b_{and,N}(n_1, n_2) = n_1 + n_2$

The same argumentation shows that if "and" is used as a conjunction between phrases that can stand as independent sentences, then its pregroup grammar type is of the form $s^r ss^l$
In the same vein, we can express "and" in the context of an independent sentence conjunction as the following bilinear form:
$b_{and,S}(s_1, s_2) = s_1 + s_2$

Examples:

- It is trivial to see that under the previous construction, all three of the following phrases have the same distributional representation ($TMNT + splinter$):
  - "Leonardo$_t$ and Michelangelo$_t$ and Donatello$_t$ and Raphael$_t$ and Splinter"
  - "animals who shredder fights"
  - "TMNT and splinter"
- "Leonardo$_t$ and Donatello$_t$ fight shredder and {Leonardo$_t$ and Donatello$_t$} eat pizza

  Here the brackets just to resolve ambiguity as to what the subject of the second verb is. We arrive at the following sentence vector:

  $= 1 * (2,0) + 1 * (2,0) + 1 * (2,0) + 1 * (2,0) = (8,0) \in \mathbb{R}^2$
- "Leonardo$_t$ and Michelangelo$_t$ fight shredder and {Leonardo$_t$ and Michelangelo$_t$} eat pizza"

  Here the brackets just to resolve ambiguity as to what the subject of the second verb is. We arrive at the following sentence vector:

  $= 1 * (2,0) + 1 * (2,0) + 1 * (2,0) + 1 * (4,0) = (10,0) \in \mathbb{R}^2$

Punctuation at end of sentences:

One last structure worth exploring is that of punctuation, in particular the exclamation point and question mark.

It is clear that both of these objects modify a sentence to once more return a valid sentence. In the language of pregroups, this means that they are of type $s^r s$. This means that when passing through the same strict monodical functor M that goes from pregroups to finite dimensional Hilbert spaces, exclamation points and question marks take the form $M(s^r s) = S^* \otimes S = S \otimes S$.

We've already seen one operator similar to this one. In the first paper we read, negation was treated as a matrix operation on two dimensional space that generalized the boolean "not" by flipping the two axes of our meaning space. In the case of exclamation points and question marks it's not as simple, since there is no clear analogue. One possibility would be to represent sentences in $n \geq 3$ dimensional space. The reason for more dimensions is that one of the dimensions can correspond to a degree of "enthusiasm" or "question". For the sake of example, let us say that $S = \mathbb{R}^4$ and that e_3 and e_4 are our respective coordinates for "enthusiasm" and "question". Then we can let our exclamation mark and question mark operators be as follows:

$o_! = s_1 \otimes s_1 + s_2 \otimes s_2 + \alpha s_3 \otimes s_3 + s_4 \otimes s_4$
$o_? = s_1 \otimes s_1 + s_2 \otimes s_2 + s_3 \otimes s_3 + \beta s_4 \otimes s_4$

Here $\alpha, \beta \in [0, \infty)$ and they parametrize how aggressively both of these punctuations modify sentences.

Examples:

Our entire TMNT linguist universe is built around a two-dimesional sentence space, so the examples don't quite work as nicely as what was mentioned above. However for the sake of demonstrating the machinery, we will suppose that e_1 is the coordinate for enthusiasm (it is in a way similar as the coordinate for truth), and that e_2 is the coordinate for question (which is not too far off from falseness). Furthermore, we will operate our operators with parameters $\alpha = \beta = 1.5$, so that:

$o_! = 1.5 s_1 \otimes s_1 + s_2 \otimes s_2$
$o_? = s_1 \otimes s_1 + 1.5 s_2 \otimes s_2$

- "dastardly turtles fight bad guys?"
  $= (0, 1.5 * 6.4) = (0, 9.6) \in \mathbb{R}^2$
- "Animals eat pizza!"
  $= (1.5 * 6.7, 2) = (10.05, 2) \in \mathbb{R}^2$
- "Animals eat pizza?"
  $= (6.6, 2 * 1.5) = (6.7, 3) \in \mathbb{R}^2$

8 Final comparison tables

List of compound nouns:

- $Noun_1$ = TNMT
- $Noun_2$ = Bad guys
- $Noun_3$ = Animals who shredder fights
- $Noun_4$ = Animals who master ninjitsu
- $Noun_5$ = Painters who paint masterpieces
- $Noun_6$ = Animals who eat pizza
- $Noun_7$ = Humans
- $Noun_8$ = Turtles
- $Noun_9$ = Animals
- $Noun_{10}$ = Radical TMNT
- $Noun_{11}$ = Dastardly turtles

Noun inner product table

	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8	N_9	N_{10}	N_{11}
N_1	16	0	8	8	0	10	4	8	4	32	0
N_2	0	320	0	0	0	0	24	0	0	0	0
N_3	8	0	5	5.5	0	5.5	2.5	4	2.5	16	0
N_4	8	0	5.5	7.05	0	6.55	2.99	5.2	3.71	16	2.8
N_5	0	0	0	0	100	0	20	0	0	0	0
N_6	10	0	5.5	6.55	0	12.05	2.99	8.2	5.71	20	2.8
N_7	4	24	2.5	2.99	20	2.99	8.33	2.4	1.57	8	0.8
N_8	8	0	4	5.2	0	8.2	2.4	7	4.6	16	4
N_9	4	0	2.5	3.71	0	5.71	1.57	4.6	3.53	8	3.2
N_{10}	32	0	16	16	0	20	8	16	8	64	0
N_{11}	0	0	0	2.8	0	2.8	0.8	4	3.2	0	10

Noun similarity measure ($cos(\theta)$)

	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8	N_9	N_{10}	N_{11}
N_1	1	0	.8944	.7535	0	.7202	.3465	.7559	.5322	1	0
N_2	0	1	0	0	0	0	.4649	0	0	0	0
N_3	.8944	0	1	.9264	0	.7086	.3874	.6761	.5951	.8944	0
N_4	.7532	0	.9264	1	0	.7106	.3902	.7402	.7437	.7532	.3335
N_5	0	0	0	0	1	0	.6930	0	0	0	0
N_6	.7202	0	.7086	.7106	0	1	.2984	.8928	.8755	.7202	.2551
N_7	.3465	.4649	.3874	.3902	.6930	.2984	1	.3143	.2895	.3465	.0877
N_8	.7559	0	.6761	.7402	0	.8928	.3143	1	.9254	.7559	.4781
N_9	.5322	0	.5951	.7437	0	.8755	.2895	.9254	1	.5322	.5386
N_{10}	1	0	.8944	.7532	0	.7202	.3465	.7559	.5322	1	0
N_{11}	0	0	0	.3335	0	.2551	.0877	.4781	.5386	0	1

List of computed sentences:

- Sentence_1 = TNMT fight shredder.
- Sentence_2 = The animals who master ninjutsu fight shredder.
- Sentence_3 = The painters who paint masterpieces eat pizza.
- Sentence_4 = Animals eat pizza.
- Sentence_5 = The animals who eat pizza fight bad guys.
- Sentence_6 = Dastardly turtles fight bad guys
- Sentence_7 = Leonardo_t and Donatello_t fight Shredder and eat pizza.
- Sentence_8 = Dastardly turtles fight bad guys?
- Sentence_9 = Animals eat pizza!
- Sentence_{10} = Animals eat pizza?

Sentence inner product table

	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}
S_1	256	240	48	107.2	3392	0	160	0	160.8	107.2
S_2	240	225.01	45	100.74	3180.2	.768	150	1.152	150.99	100.86
S_3	48	45	9	20.1	636	0	30	0	30.15	20.1
S_4	107.2	100.74	20.1	48.89	1424.2400	12.8	67	19.2	71.335	50.89
S_5	3392	3180.2	636	1424.24	44947.67	12.288	2120	18.432	2134.4	1426.2
S_6	0	.768	0	12.8	12.288	40.96	0	61.44	12.8	19.2
S_7	160	150	30	67	2120	0	100	0	100.5	67
S_8	0	1.152	0	19.2	18.432	61.44	0	92.16	19.2	28.8
S_9	160.8	150.99	30.15	71.335	2134.4	12.8	100.5	19.2	105	73.335
S_{10}	107.2	100.86	20.1	50.89	1426.2	19.2	67	28.8	73.335	53.89

Sentence similarity measure ($cos(\theta)$)

	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}
S_1	1	1	1	.9582	1	0	1	0	.9808	.9217
S_2	1	1	1	.9605	1	.0080	1	.0080	.9823	.9159
S_3	1	1	1	.9582	1	0	1	0	.9808	.9127
S_4	.9582	.9605	.9582	1	.9608	.2860	.9582	.2860	.9956	.9914
S_5	1	1	1	.9608	1	.0091	1	.0091	.9825	.9163
S_6	0	.0080	0	.2860	.0091	1	0	1	.1952	.4087
S_7	1	1	1	.9582	1	0	1	0	.9808	.9127
S_8	0	.0080	0	.2860	.0091	1	0	1	.1952	.4087
S_9	.9808	.9823	.9808	.9956	.9825	.1952	.9808	.1952	1	.9749
S_{10}	.9217	.9159	.9127	.9914	.9163	.4087	.9127	.4087	.9749	1

References:

Coecke, Bob, Sadrzadeh, Mehrnoosh, Clark, Stephen. Mathematical foundations for a compositional distributional model of meaning. *arXiv preprint. arXiv:1003.4394*, 2010.

Sadrzadeh, Mehrnoosh, Clark, Stephen, Coecke, Bob. The Frobenius anatomy of word meanings I: subject object and relative pronouns. *arXiv preprint. arXiv:1404.5278*, 2014.

Piedeleu, Robin, Kartsaklis, Dimitri, Coecke, Bob, Sadrzadeh, Mehrnoosh. Open system categorical quantum semantics in natural language processing. *arXiv preprint. arXiv:1502.00831v2*, 2015.