Tel Aviv University

The Lester and Sally Entin Faculty of Humanities

Department of Linguistics

Learning Head-Complement Order with Minimalist Grammars

MA thesis submitted by:

Tomer Avraham

Thesis advisor:

Dr. Roni Katzir

August 2017

Abstract

Head-complement order varies between and within languages. Several theories have been proposed to describe how this might be represented mentally. However, all current theories agree that some element of it must be learned – either the order itself or some sort of appropriate movement.

This work attempts to explore these theories from a computational learning perspective. A computational learner using Minimalist Grammars (Stabler (1997)) was implemented and used to learn different head-complement orders according to the different representational models.

This learner is the first computational learner using Minimalist Grammars and is also the first learner to tackle head-complement order. This work makes first steps toward UG-comparison in syntax using learning (as outlined in Katzir (2014)).

Contents

1. Head-Complement Order 5
1.1. Order Variation 5
1.2. Possible Theories 6
1.2.1. Language Parameter 6
1.2.2. Category Parameter 7
1.2.3. Word Parameter 7
1.2.4. Kayne's LCA 7
1.3. Choosing the Right Theory 8
2. Theoretical Models for Learning 9
2.1. Background 9
2.2. Mildly Context Sensitive Languages 10
2.3. Minimalist Grammars 11
2.4. Parser 16
2.5. Minimum Description Length 17
2.6. Simulated Annealing 24
3. Implementing Head-Direction Learners 26
3.1. Introduction 26
3.2. Input 26
3.3. Formalism 28
3.4. Parser 29
3.5.1. Initial Hypothesis 37
3.5.2. Random Neighbour 39
3.5.3. Energy 42
3.6. Running Time Considerations 43
4. Expected Results 45

5. Results 49
5.1. Kayne Learner 49
5.1.1. Head Initial Input 49
5.1.2. Head Final Input 52
5.1.3. Mixed Category Input 53
5.1.4. Mixed Word Input 53
5.2. Language Learner 54
5.2.1. Head Initial Input 54
5.2.2. Head Final Input 54
5.2.3. Mixed Category Input 55
5.2.4. Mixed Word Input 55
5.3. Category Learner 56
5.3.1. Head Initial Input 56
5.3.2. Head Final Input 56
5.3.3. Mixed Category Input 57
5.3.4. Mixed Word Input 57
5.4. Word Learner 58
5.4.1. Head Initial Input 58
5.4.2. Head Final Input 58
5.4.3. Mixed Category Input 59
5.4.4. Mixed Word Input 59
6. Result Analysis 61
7. Trying to Improve Results with Coordination 63
8. Discussion 65
References 68
Appendix A – Detailed Results 70

1. Head-Complement Order

1.1. Order Variation

In discussing the various possible sentence structures in natural languages, a useful and common generalization is that the order within a phrase between the head and its complement is consistent in a given language.
Greenberg (1963) studies 30 different languages and notes that in 13 of them, the verb immediately precedes the object (head-initial VP) and in 11 the verb follows the object (head-final VP). The other 6 have VSO order.
Of the head-initial VP languages, 10 have prepositions (the adposition precedes the noun – head-initial PP) and 3 have postpositions (the adposition follows the noun – head-final PP). All 11 head-final VP languages have postpositions (head-final PP). The VSO languages all have prepositions.
Greenberg shows similar correlations with different head types – genitives, adjectives, adverbs, etc.

We can see that the generalization holds, there is a tendency to favor one direction within the language, but also that there are exceptions – some languages have different head-complement orders for different lexical categories.
For instance, in Persian, verbs are head-final (the object precedes the verb) and adpositions are head-initial (the adposition precedes the noun – prepositions):
(1) Mahootian (1997), p. 139

pul-o *be* *ma* *dad*
money-OBJ to 1PL gave
"He gave the money to us".

Furthermore, there are languages which show different orders for different heads in the same lexical category.
German, for instance, has both prepositions and postpositions:
(1) Van Riemsdijk (2007), p. 3

Peter *legt* *das* *Buch* *auf* *den* *Tisch*
Peter puts the book on the.ACC table

"Peter puts the book on the table."

[$_{PP}$ [$_{P}$ **auf**] [$_{DP}$ den Tisch]]

(2) Van Riemsdijk (2007), p. 1

Die	*Schnecke*	*kroch*	*das*	*Dach*	*hinauf*
the	snail	crept	the	roof	up

"The snail crept up the roof"

[$_{PP}$ [$_{DP}$ das Dach] [$_{P}$ **hinauf**]]

Thus, we can see that head-complement order can vary by language, by lexical category and even by lexical item. Whatever theory we choose to describe the mental representation of word order must deal with the observed variation in head-complement order.

1.2. Possible Theories

Several theories have been proposed to describe the cognitive representation of the different possible head-complement orders.

1.2.1. Language Parameter

Of these theories, one central approach states that head-complement order (or head-directionality) is a parameter of the language (Chomsky (1981)). Therefore, a language is either head-initial or head-final and a child will acquire the appropriate value for the language during the acquisition process.

Under this theory, any variation in order within the language is due to syntactic movement or some other mechanism. For example, in a head-initial language, any head-final structure will be generated as head-initial in the base generation stage and through movement will appear head-final in the surface representation.

For the purposes of this work, we'll refer to this theory as The Language Parameter theory.

1.2.2. Category Parameter

The next logical possibility is that there is a head-complement order parameter for every grammatical category. Therefore, a language could have, e.g., head-initial verb phrases and head-final adpositional phrases and vice versa. Here, similarly, any variation within a category would be the result of syntactic movement or some other language-specific mechanism. Meaning, e.g., in a head-initial category, any head-final structure will be generated as head-initial in the base generation stage and through movement will appear head-final in the surface representation.
For the purposes of this work, we'll refer to this theory as The Category Parameter theory.

1.2.3. Word Parameter

The third theory in the same vein is a micro-parametric approach supported by Cinque (2016) which states that a micro-parameter for head-complement order exists for every head in the lexicon. Therefore, one adposition in the language could be head-initial (preposition) and another would be head-final (postposition). This theory assumes an order value is kept for every lexical item. Advantageously, this model explains the various orders without requiring syntactic movement or any other mechanism. On the other hand, it's harder to explain with this model why head-complement order seems to be largely uniform within a language.
For the purposes of this work, we'll refer to this theory as The Word Parameter theory.

1.2.4. Kayne's LCA

Finally, Kayne (1994) proposes a somewhat different theory.
Kayne's theory, the *Linear Correspondence Axiom* (LCA), suggests that the universal representation of head-complement order is specifier-head-complement. I will not discuss specifiers in this work, therefore, we can say the LCA suggests that the mental representation is always head-initial.
Thus, head-complement order is not a parameter but a principle.

This model would require various syntactic movements to account for the different head-complement orders in different languages.
For the purposes of this work, we'll refer to this theory as Kayne's LCA theory.

1.3. Choosing the Right Theory

In order to try and differentiate between the different theories and to try and point out their differences, we need to look at their advantages and disadvantages in terms of parsing, learning and lexical representation. One way to do this in a controlled fashion is through computational learning.

2. Theoretical Models for Learning

2.1. Background

In the broadest terms, a computational learner is an algorithm that tries to find and describe a pattern in some data.
A computational language learner usually tries to emulate the process of language acquisition in children.
In its core, a language learner is an algorithm that receives an input of valid utterances (words, sentences, etc.) in a certain target language and outputs a grammar – in the form of a set of rules or some formalism to represent grammars – that is capable of generating the target language.

For this work, a computational learner was implemented that is capable of learning a simple input of sentences with varying head-complement orders. Furthermore, the learner is able to learn this in accordance with each of the theories described in section 1.2.

To implement such a learner, several theoretical models are required:

- Formalism – to describe a grammar.
- Parser – to test whether the grammar can generate the input.
- Metric – to compare different grammars in order to choose the "best" one.
- Learning algorithm – the basis for the learner itself.

All of the theoretical models chosen for each of these will be discussed in this section.

The specifics of the implementation of the learner itself will be described in section 3.

The results of learning the various orders with the different theories will be shown in section 5 and then analyzed in section 6.

2.2. Mildly Context Sensitive Languages

In choosing a formalism we want something that can most accurately represent Natural languages. There are two levels of adequacy we can work at. The first – weak generative capacity – means that the formalism is capable of generating any sentence that Natural language can generate and isn't capable of generating sentences that Natural language cannot.
The second level of adequacy – strong generative capacity – is about the structure of the sentences themselves, the way sentences are generated. We want the formalism to generate sentences in a similar way and with a similar structure to Natural languages.

Therefore, for the weak generative capacity, before deciding on a formalism to describe our input, we must decide on the class of formal languages we'd like it to describe. Ideally, as we want to describe a natural language, we would like the class to be strong enough to contain all natural languages but not stronger.

Mildly Context Sensitive Languages (MCSL) are a set of formal languages defined by Joshi (1985) as a set of languages having the following properties:

1) The set contains the family of Context Free Languages.
2) The set supports limited cross-serial dependencies.
3) All languages in the set have the constant growth property.
4) All languages in the set are polynomially parsable.

A grammar is said to be a Mildly Context Sensitive Grammar (MCSG) if the set of languages it generates is MCSL.

Cross-serial dependencies is a word pattern that can be seen in the copy language - $\{ww \mid w \in \Sigma^*\}$. The copy language is not context free and thus MCFGs are stronger than Context Free Grammars.
Constant growth is a property that limits the sizes of words in the language and states that the length of words in the language grow (when sorted by length) at most by a constant, language-specific factor.

Polynomial parsing is a property which states that it is possible to decide whether a word is in the language in polynomial time.

Natural languages have also been shown to support cross-serial dependencies. Shieber (1985) shows this for sentences in Swiss-German such as:

...mer em Hans es Huus hälfed aastriiche.

...we Hans (DAT) *the house* (ACC) *helped paint.*

…we helped Hans paint the house.

The verb “hälfed” assigns case to “em Hans” and the verb “aastriiche” assigns case to “es Huus”. We can see that the serial order between the verbs is the same as the order of the nouns and also that the dependencies cross each other.

Shieber (1985) also states that this phenomenon is unbounded in principle, meaning, any length of such cross-serial embeddings is theoretically possible (although unlikely in practice).

Since, as mentioned above, CFGs cannot represent cross-serial dependencies, a stronger class of grammars is required to fully describe natural languages. However, no evidence has been found so far to suggest that natural languages are computationally stronger than MCSGs. Therefore, Joshi (1985) suggests that MCSGs are the best formal grammars to describe natural languages.

2.3. Minimalist Grammars

Now, after choosing a class of formal languages, we would like to find a formalism that has strong generative capacity – that can generate the same structures as Natural languages.

Minimalist Grammars are a formalism for describing Mildly Context Sensitive Grammars loosely based on the Minimalist Program and suggested in Stabler (1997).

It is a formalism used to describe phenomena that are common in natural languages – namely, movement (such as a wh-element moving to the beginning of the sentence) and tree merging (combining two smaller trees).

A Minimalist Grammar is composed of a lexicon of basic trees or "words" in the language the grammar describes and the two operations – Merge and Move.

Generally speaking, an item in the lexicon would represent a word, its grammatical category, its selectional requirements and into what position it would move in the surface structure.

A tree in the language would be represented like this (I use here the collapsed representation described in Harkema (2001)):

[<node1>, <node2>, …]<type>

A number of nodes followed by the type of tree – 's' for simple and 'c' for complex.
For instance, a simple tree that holds one node might look like this – [tree: NP]s.
A tree with one node would normally be simple and a tree with more than one node would always be complex.
As a rule of thumb, a simple tree will represent a lexical item (a word) and in a complex tree each node will represent a syntactic constituent.
We read the format from left to right and the order of the nodes corresponds to the linear order of the sentence the tree represents. Thus, a tree with several nodes could represent the D-Structure and then move (will be explained below) to represent the S-Structure of a sentence.

A node would look like this:

<substring>: <base> =<select1> +<licensor1> -<licensee1>

Say, for example, ['the': NP =N]s.

A node would have:

- A substring – the "word" it represents ("the").
- A base – its grammatical category (NP).
- Any number of select features – the grammatical categories it needs to merge with to realize its own grammatical category (=N).
- Any number of licensor and licensee features – the license features enable the movement of a node with a licensee feature towards a node with the corresponding licensor feature.

The two operations are Merge and Move.
Merge can take two trees and merge them into one tree.
Merge applies when tree1 has a select feature that corresponds to the base of tree2.

There are two options for the result of Merge. If tree1 is a simple tree then tree2 would be merged to the right of tree1, as in the following example:

Tree1 – ['the': NP =N]s

Tree2 – ['man': N]s

The result would be:

['the man': NP]c

As we see, the substrings are joined with tree1's base and the result is a complex tree (as it is the result of a merge).

If, however, tree1 is a complex tree then the position to its right is already occupied and tree2 would be merged to its left, as in the following example:

Tree1 – ['sees the man': IP =NP]c

Tree2 – ['the woman': NP]c

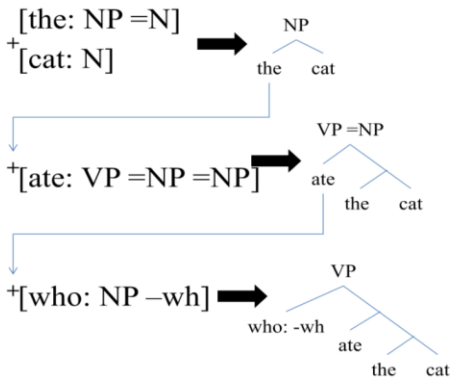
The result would be:

['The woman sees the man': IP]c

Again, the substrings are joined and the result is of course complex.

Note that in both cases if tree2 had any license features – meaning some movement will occur later – then they wouldn't be deleted and the substrings wouldn't be joined but the nodes would be separated by a comma (['the': NP, 'man': -a]c). In this example, the node “man” had a licensee feature (-a) before the merge. This means this node will move at a later stage in the derivation. Thus, the substrings are not merged to keep the nodes separated so only the movable node will move when the appropriate licensor feature is merged.

Another example, deriving the deep structure (D-Str.) of the sentence “Who ate the cat?”:



Move applies when the leftmost node has a licensor feature and another node in the tree has the corresponding licensee feature. The moved node will move to the left of the first node, as in the following example:

The following tree is derived by merging [‘I saw’: CP =NP +wh]c and [‘who’: NP -wh]s:

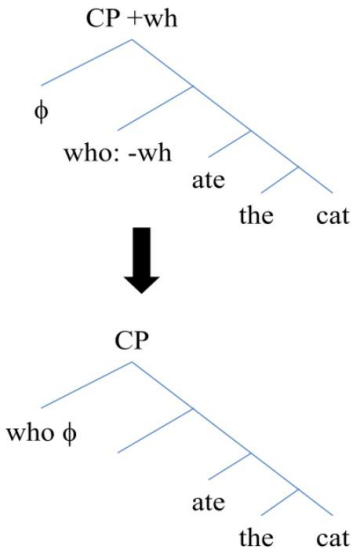
Tree – ['I saw': CP +wh, 'who': -wh]c

The result of Move would be:

['who I saw': CP]c

A tree with more than one node is already complex (since it must be a result of a merge). Thus, the result of a move operation is naturally complex as well.

Another example, deriving the surface representation (S-str.) of the sentence “Who ate the cat?”:



As a Minimalist Grammar is defined by its lexicon – a set of trees – for a given Minimalist Grammar, a sentence is said to be "in the language" defined by the grammar if it could be derived from the grammar's lexicon by any number of Merge and Move operations.

As an example, let's view the language containing all even length palindromes over the alphabet Σ={a, b}. The corresponding Minimalist Grammar would be:

- [@: c]s
- [a: c =c +A, a: -A]c
- [b: c =c +B, b: -B]c

(The symbol '@' represents the empty string)

*Note: In standard MGs, grammars are usually composed of simple items only. This grammar has an equivalent grammar using only simple items but this one was chosen for being easier to understand.

Now let's look at a derivation of the sentence 'abba':

1. Merge: [b: c =c +B, b: -B]c + [@: c]s => [b: c +B, b: -B]c
   A merge is made to remove the select feature.
2. Move: [b: c +B, b: -B]c => [bb: c =c]c
   The second node moves to the left of the first node.
3. Merge: [a: c =c +A, a: -A]c + [bb: c]c => [bba: c +A, a: -A]c
   The 'a' tree is "wrapped around" the 'b' tree.
4. Move: [bba: c +A, a: -A]c => [abba: c]c
   The "wrapping" is completed.

*Note: We might notice that with respect to head-complement order, Minimalist Grammars work as described by Kayne's theory – all merges are to the right (head-initial) and head-final structures can only be generated by movement.
I deal with this further in section 3.3.

**Note: In Harkema's description of Minimalist Grammars, Merge and Move operate if the two leftmost features of both trees enable it. I have a necessary order between the base, selection and license features so it looks a bit different but the implementation described in section 3.3 has limitations on Merge and Move that eventually make both models behave the same way.

2.4. Parser

A parser for a formal language is an algorithm that, given a sentence, determines whether that sentence is a part of the language or not while also returning a derivation of the sentence using the grammar.
A more general parser is given both a sentence and a grammar which represents the language to be tested.

One general approach for designing such parsers is Bottom-Up – starting from the leaves (in our case, words in the sentence) and working our way up (towards the sentence level). In the broadest terms, using the grammar to generate all possible sentences in the language until the target sentence is found (usually trying to direct the sentence generation towards the target sentence).

The parser implemented for this work is a Bottom-Up chart parser for Minimalist Grammars based on the chart parsing algorithm suggested in Harkema (2001):

1. Initialize the chart to the empty set of items and the agenda to the axioms of the deduction system.
2. Repeat the following steps until the agenda is exhausted:
    2.1. Select an item from the agenda, called the trigger item, and remove it from the agenda.
    2.2. Add the trigger item to the chart, if the item is not already in the chart.
    2.3. If the trigger item was added to the chart, generate all items that can be derived from the trigger item and any items in the chart by one application of a rule of inference, and add these generated items to the agenda.
3. If a goal item is in the chart, the goal is proved, i.e., the string is recognized, otherwise it is not.

2.5. Minimum Description Length

Finding **some** grammar that generates the input is a relatively easy task.
For instance, if we take the following input over the alphabet Σ={a, b}:

1. aaa
2. bbb
3. aaabbb
4. bbbaaa

Intuitively we'd say that the input might represent the language (aaa | bbb)* but there are infinitely many possible grammars that can generate the input and thus can be returned by our learner.

One way to implement our learner is to simply return a grammar that can parse **all** possible inputs.
Then a possible grammar that indeed generates the input is:

a. [@: c]s
b. [a: c =c]s
c. [b: c =c]s

We notice that this grammar can parse **any** word over Σ={a, b} – by merging the empty string to the last letter and merging the second to last letter with the result and so forth.
Let's look at a derivation of "abb":

1. Merge: [b: c =c]s + [@: c]s => [b: c]c
2. Merge: [b: c =c]s + [b: c]c => [bb: c]c
3. Merge: [a: c =c]s + [bb: c]c => [abb: c]c

This is likely not the result we'd like our learner to return, we'd like the resulting grammar to be more specific.

Alternatively, our learner could return a grammar that is specific to the input only:
(1)

a. [aaa: c]s
b. [bbba: c]s
c. [aaabbb: c]s

d. [bbbaaa: c]s

Clearly, this grammar can generate the input but cannot generate any other word.

Since we assume that our input represents a small subset of an infinite number of possible natural language sentences, we'd like the resulting grammar to be more general.

Considering these limitations, we'd like some way of quantifying how "good" a grammar is for a given input. Some sort of metric that will help the learner find the "best" grammar for the given input.

This metric should avoid scenarios like those shown above – too specific or too general. The ideal grammar is one that is tight around the input – can generate all structures that correspond to the input and can't generate any structures that don't.

We'll note that these requirements work against each other. In our example, the first grammar is as loose as possible – it can generate every structure that is similar to the input and it can also generate any other structure using the same words. Usually, a loose grammar is shorter – in our example, however long the input and whatever it is, the grammar only consists of three items.

The second grammar we give is as strict as possible – it can only generate the input sentences and no more. Usually, a strict grammar leads to a short derivation – in our example, each input item has a one-step derivation. The grammar in this case is long, it has as many items as there are input items.

Therefore, our metric should try and balance these two requirements.

This work will follow the methodology demonstrated in Katzir (2014) that uses the metric of Minimum Description Length (MDL) (Rissanen (1978)). Given the input, the total description length would be the sum of the length of the shortest way of encoding the grammar (in binary, for our purposes) and the shortest way of encoding the input given the grammar – thus, encoding the derivation of the input item. We'll say that the "best" grammar for a given input is the grammar with the shortest description length.

This method balances the two requirements as it tries to minimize both sizes at once.

The grammar will be encoded thus:

- Each ‘letter’ in the alphabet (in our case, each word in the lexicon) that appears in some grammar item’s head (substring), including the empty string (which we treat as a separate word), will get a fixed length code h.
- Each base will get a fixed length code b.
- Each licensor will get a fixed length code L.
- We start by 0*h + 1 + 0*b + 1 + 0*L + 1.
- Now each tree in the lexicon will be represented in the following way:
    - for each node in the tree:
        - Each word in its head and then the empty string – to mark the end of the head (h * (number_of_words_in_the_head + 1)).
        - Its base (length b) - only the first node has a base.
        - For every select - 00 + base (length b).
        - For every licensor - 01 + licensor (length L).
        - For every licensee - 10 + licensee (length L).
        - 11 + (0 if there is another node after it or 1 if it’s the last node).
    - 0 for s, 1 for c.
    - At the end of each tree – 0 if there are more trees, 1 if this is the last tree.

The input will be encoded thus:

- We calculate the derivation of the parsing and get the number of steps*.
- We enumerate the trees in the lexicon and the steps of the derivation and assign a fixed length code for each – k. (The derivation steps are numbered according to their order in the derivation. The code for each step will only be used **after** that step has taken place so a “receiver” can know what tree the code entails)
- We start by 0*k + 1.
- We encode the input by steps of the derivation:
    - 0 for Merge or 1 for Move.
    - For Merge - we also write the two trees that were merged (either from the lexicon or results of previous steps in the derivation).
    - For Move - we write the tree that moved (either from the lexicon** or a result of a previous step in the derivation).
- We add the derivation step or grammar item that produced the input.

*Note: For sentences with very long derivation this may not be optimal. Since at any step of the derivation there are at most two “active” trees, we could reference them more efficiently than by enumerating all the steps of the derivation. However, since in our case the size of the grammar is usually much larger than the length of the derivation, this is not a problem.

**Note: Since items in the lexicon are usually simple, items that move are from earlier stages of the derivation and not from the lexicon. However, the encoding still allows for it.

The metric for the grammar is the sum of the length of the encoding of the grammar and of each of the input items. We say one grammar is “better” than another one when its total description length is smaller.

Let’s see how it works with our example and the general grammar, repeated here:
Input:

1. aaa
2. bbb
3. aaabbb
4. bbbaaa

General grammar:
(2)

a. [@: c]s
b. [a: c =c]s
c. [b: c =c]s

The general grammar will be encoded thus:

- There are three letters (“”, “a”, “b”) which will be encoded as 00, 01, 10, respectively. So h = 2.
- There is one base (“c”) which will be encoded as 0. So b = 1.
- There are no license features so L = 0.
- Header – $\underbrace{001}_{h=2}\ \underbrace{01}_{b=1}\ \underbrace{1}_{L=0}$ – 6 bits

- Each item is encoded:
  - [@: c]s:

    $$\underbrace{00}_{substring-""}\ \underbrace{0}_{base-c}\ \underbrace{11}_{end\ of\ node}\ \underbrace{1}_{last\ node}\ \underbrace{0}_{simple\ tree-s}\ \underbrace{0}_{not\ last\ tree}$$

    – 8 bits
  - [a: c =c]s:

    $$\underbrace{01\ 00}_{substring-"a"}\ \underbrace{0}_{base-c}\ \underbrace{00}_{select}\ \underbrace{0}_{c}\ \underbrace{11}_{end\ of\ node}\ \underbrace{1}_{last\ node}\ \underbrace{0}_{simple\ tree-s}\ \underbrace{0}_{not\ last\ tree}$$

    – 13 bits
  - [b: c =c]s:

    $$\underbrace{10\ 00}_{substring-"b"}\ \underbrace{0}_{base-c}\ \underbrace{00}_{select}\ \underbrace{0}_{c}\ \underbrace{11}_{end\ of\ node}\ \underbrace{1}_{last\ node}\ \underbrace{0}_{simple\ tree-s}\ \underbrace{1}_{last\ tree}$$

    – 13 bits
- Total length – 40 bits.

The input as encoded by the general grammar:

- For each item in the input we saw that the derivation was as long as the item so for each item w:

$$k = \left\lceil \underbrace{log_2}_{number\ of\ bits\ required} \left(\underbrace{3}_{number\ of\ grammar\ items} + \underbrace{|w|}_{number\ of\ derivation\ steps} \right) \right\rceil$$

- The encoding stars with 0*k and then 1 => k + 1 bits.
- The encoding ends with the derivation step that produced the input => k bits.
- Each step of the derivation is a merge operation so each step adds:

$$\underbrace{1}_{merge\ operation} + \underbrace{k}_{first\ item} + \underbrace{k}_{second\ item} = 1 + 2k\ bits$$

- In total there are |w| derivation steps so the total length is:

  $k + 1 + |w| * (1 + 2k) + k$
- Let's look at the encoding of one item – aaa:
  - |w| = 3
  - $k = \lceil log_2(3 + 3) \rceil = 3$
  - The three grammar items will be encoded thus:
    - [@: c]s – 000
    - [a: c =c]s – 001
    - [b: c =c]s – 010

- The three derivation steps will be encoded thus:
  - Merge([a: c =c]s, [@: c]s) = [a: c]c – 011
  - Merge([a: c =c]s, [a: c]c) = [aa: c]c – 100
  - Merge([a: c =c]s, [aa: c]c) = [aaa: c]c – 101
- The total encoding:

$$\underbrace{0001}_{k=3}\ \underbrace{1}_{merge}\ \underbrace{001}_{\text{[a: c =c]s}}\ \underbrace{000}_{\text{[@: c]s}}\ \underbrace{1}_{merge}\ \underbrace{001}_{\text{[a: c =c]s}}\ \underbrace{011}_{\text{[a: c]c}}\ \underbrace{1}_{merge}\ \underbrace{001}_{\text{[a: c =c]s}}\ \underbrace{100}_{\text{[aa: c]c}}\ \underbrace{1}_{merge}\ \underbrace{101}_{\text{result}-\text{[aaa: c]c}}$$

- Total length – $3 + 1 + 3 * (1 + 2 * 3) + 3 = 28\ bits$

- The length of the other items will be:
  - bbb – |w| = 3, k = 3 => $3 + 1 + 3 * (1 + 2 * 3) + 3$ = 28 bits
  - aaabbb – |w| = 6, k = 4 => $4 + 1 + 6 * (1 + 2 * 4) + 4$ = 63 bits
  - bbbaaa – |w| = 6, k = 4 => $4 + 1 + 6 * (1 + 2 * 4) + 4$ = 63 bits
- The total encoding length of the input will be 182 bits.

The total description length of the general grammar will be 222.

We'll now calculate the total description length of the specific grammar (1) a bit more briefly.

The grammar encoding:

- There are three 'letters' (including the empty string), one base and no license features.

  So h = 2, b = 1, L = 0.
- Header length – h + 1 + b + 1 + L + 1 = 6
- Each item length:
  - aaa – 4*h + b + 2 + 1 + 1 + 1 = 14
  - bbb – 4*h + b + 2 + 1 + 1 + 1 = 14
  - aaabbb – 7*h + b + 2 + 1 + 1 + 1 = 20
  - bbbaaa – 7*h + b + 2 + 1 + 1 + 1 = 20
- Total length – 6 + 14 + 14 + 20 + 20 = 74

The input encoding:

- Each input item is already in the grammar so no derivation is required.
- Thus, k = 2 (for the four grammar items) and the encoding length for each input item will be 2 + 1 + 2 = 5.
- Total length – 4 * 5 = 20

The total description length of the specific grammar is 94.

Right now, the specific grammar (1) clearly has the advantage but that is also because the number of items in the grammar and in the input is on the same scale. This is usually not the case. Let's examine the case where the input is all combinations of "aaa" & "bbb" that are 9 characters or shorter:
['aaa', 'bbb', 'aaaaaa', 'aaabbb', 'bbbaaa', 'bbbbbb', 'aaaaaaaaa', 'aaaaaabbb', 'aaabbbaaa', 'aaabbbbbb', 'bbbaaaaaa', 'bbbaaabbb', 'bbbbbbaaa', 'bbbbbbbbb']

General grammar (2):

Grammar encoding length: 40

Input encoding length: 1028

Total encoding length: 1068

Specific Grammar (1):

Grammar encoding length: 322

Input encoding length: 126

Total encoding length: 448

The specific grammar still has the advantage. But let's now test it against the grammar we would have liked to guess intuitively:

a. [@: IP]s
b. [aaa: IP =IP]s
c. [bbb: IP =IP]s

It's easy to see how this parses the input and the encoding length we get is:

Grammar encoding length: 48

Input encoding length: 328

Total encoding length: 376

And we can see how the intuitive grammar is indeed the most efficient when applying the MDL.

2.6. Simulated Annealing

The final thing our learner needs is an algorithm by which to find the best grammar that generates the input.

Again, following the methodology in Katzir (2014) we will use Simulated Annealing (Kirkpatrick, Gelatt, and Vecchi (1983)).

Simulated Annealing is an approach for locating a good approximation for a global optimum of a function with a large search space.

The general process is deceptively simple. Two important sizes are monitored – temperature and energy (both terms are taken from the physical process of metal annealing). The energy is the function we would like to minimize and the temperature is, broadly, our tolerance for errors.

The process works by selecting an initial hypothesis and slowly decreasing the temperature while testing neighbouring hypotheses to try and find the one with the lowest energy. The higher the temperature the more we are willing to risk trying states with a higher energy than our current state or how "steep" a hill we are willing to climb to find the best hypothesis. The initial temperature and the rate of decrease determine the effectiveness of the process.

Pseudocode:

1. D ← input
2. H ← initial hypothesis
3. T ← initial temperature
4. while T > threshold do:
    1. H’ ← random neighbour(H)
    2. $\Delta \leftarrow$ Energy(H’, D) − Energy(H, D)
    3. if $\Delta < 0$ then:
        1. p ← 1
    4. else:
        1. p ← $e^{-\Delta/T}$
5. choose H ← H’ with probability p
6. T ← αT
7. return H

In our case, the search space is the range of all possible grammars while we try and find the grammar with the shortest description length for the input. Thus, the energy of a current hypothesis is exactly the description length of the hypothesis and the input.
The initial hypothesis and the method of neighbour generation are specific to the search space and to the problem to solve.

The learning process will work as follows:
The learner generates an initial hypothesis based on the input.
At each iteration, the learner will generate a neighbour to its current hypothesis. The neighbour is guaranteed to parse the input.
If this neighbour is more energy efficient in parsing the input then it is selected as the new hypothesis, otherwise it could still be selected with a probability which corresponds to the current temperature (the higher the temperature the more probable the selection) and to the energy difference between it and the current hypothesis.
At each iteration, the temperature is decreased by a constant factor.
Eventually the hypothesis has a low enough energy and a low enough temperature such that no further changes are made. That final hypothesis is the grammar we say the learner has "learned".

3. Implementing Head-Direction Learners

3.1. Introduction

The phenomenon we would like to simulate is the ability of the human learner to learn languages with various head-complement order combinations, seemingly without issue. Furthermore, we'd like to compare the different theories in their ability to produce the same results as a human learner.
Therefore, the approach I took in this work is the following:

1. Implement a Minimalist Grammars-Simulated Annealing learner along with the necessary infrastructure, as described in section 2.
2. Adjust the learner so that it can learn by following each of the theories described in section 1.2:
    a. Language Learner – will learn according to the Language Parameter theory.
    b. Category Learner – will learn according to the Category Parameter theory.
    c. Word Learner – will learn according to the Word Parameter theory
    d. Kayne Learner – will learn according to Kayne's LCA theory.
3. Test each theory-version of the learner by trying to learn sets of input sentences that correspond with different forms of head-complement order as described in section 1.1.

All of the code for this work was implemented in Python 2.7 and can be found in:
https://bitbucket.org/taucompling/HeadComplementSyntacticLearner

3.2. Input

An input for the learner will be a set of sentences. The learner will get this set and try to return a grammar that can generate them.
If the set of sentences conforms to some rule, we'd like that rule to be expressed in the resulting grammar.

The four major types of input we'd like to test our learner on are as follows:

1. A fully head-initial language.
2. A fully head-final language.
3. A language with some categories fully head-initial and the others fully head-final.
4. A language where within one category some words are head-initial and some are head-final.

This covers the various combinations of head-complement order seen in natural languages as discussed in section 1.1.

Of course, due to running time and implementation considerations, these sets must be quite simplistic (for now). Furthermore, as to avoid any interference with different languages, all sets will be composed of sentences with the same (English) words and the only difference between sets will be the order of words in the sentences.

I implemented a python module that can generate the types of inputs I would want to test the learner on.

- InputGenerator.py

This file has constant lists of words divided by lexical category.

```
proper_nouns = ["Jerry", "George", "Elaine", "Kramer"]
transitive = ["liked", "saw", "loved", "hated"]
intransitive = ["ran", "walked", "read", "wrote"]
cp_transitive = ["knows", "says", "thinks", "assumes"]
prepositions = ["with", "by", "above", "under"]
complementizers = ["that"]
```

The file exports one function that generates a set of sentences of a given size and with given parameters (include prepositions/include transitive verbs/recursion depth).
For each lexical category, a parameter determines whether it should be head-initial, head-final or mixed (some of the heads will be initial and some will be final).

```
>>> TestInputs.get_custom_text(size = 5,
                               with_transitive = "initial",
                               with_prepositions = "initial",
                               with_cp = None,
                               recursion_depth = 0)

['George saw Kramer above George',
 'George walked',
 'Jerry liked Elaine with Kramer',
 'Jerry hated George',
 'Elaine liked George']
```

Thus, the input can be fine-tuned and is generated randomly each time the learner runs.

3.3. Formalism

I've implemented the Minimalist Grammars formalism in several python modules:

- MinimalistGrammar.py
- MinimalistGrammarTree.py
- MinimalistGrammarNode.py

A MinimalistGrammar object is basically a list (with several added features) of MinimalistGrammarTree objects which represent the items of the lexicon. Each MinimalistGrammarTree object has one or more MinimalistGrammarNode objects contained within.
The trees are represented in the collapsed representation shown in section 2.3 and support the operations *merge* and *move*.

Running example:

```
>>> from MinimalistGrammarTree import MinimalistGrammarTree
>>> tree_1 = MinimalistGrammarTree("[@: c]s")
>>> tree_2 = MinimalistGrammarTree("[a: c =c +A, a: -A]c")
>>> tree_3 = MinimalistGrammarTree("[b: c =c +B, b: -B]c")
>>> derivation_step_1 = tree_3.merge(tree_1)
>>> derivation_step_1
[b: c +B, b: -B]c
>>> derivation_step_2 = derivation_step_1.move()
>>> derivation_step_2
[b b: c]c
>>> derivation_step_3 = tree_2.merge(derivation_step_2)
>>> derivation_step_3
[b b a: c +A, a: -A]c
>>> derivation_step_4 = derivation_step_3.move()
>>> derivation_step_4
[a b b a: c]c
```

To be able to learn head-complement order an adjustment had to be made to the formalism.
We may notice that Minimalist Grammars, as described, seem to work in accordance with Kayne's theory. That is, complements always merge to right of the head. To be able to encode head-final structures we must modify the formalism slightly.

The modification I've made is to add a parameter to the MinimalistGrammarTree object that holds the merge direction. When the merge direction parameter has the value "right", the direction of the merge will be exactly as described in section 2.3. When the parameter has the value "left", the direction of the merge will be a mirror image of that. That is, a simple merge will have the selected tree to the left of the original tree and a complex merge will have the selected tree to the right of the original. The merge direction is indicated in the format by ">" or "<".

Note that when merging, only the direction of the selecting tree matters, not the direction of the selected tree. Furthermore, the merge direction of the result is the same as that of the selecting tree.

This change requires an adjustment to the grammar encoding. This will be discussed in section 3.5.3.

Running example:

```
>>> from MinimalistGrammarTree import MinimalistGrammarTree
>>> right_tree_1 = MinimalistGrammarTree("[>a: c =c =c]s")
>>> right_tree_2 = MinimalistGrammarTree("[>b: c]s")
>>> right_tree_3 = MinimalistGrammarTree("[>c: c]s")
>>> right_merge_1 = right_tree_1.merge(right_tree_2)
>>> right_merge_1
[>a b: c =c]c
>>> right_merge_2 = right_merge_1.merge(right_tree_3)
>>> right_merge_2
[>c a b: c]c
>>>
>>> left_tree_1 = MinimalistGrammarTree("[<a: c =c =c]s")
>>> left_tree_2 = MinimalistGrammarTree("[<b: c]s")
>>> left_tree_3 = MinimalistGrammarTree("[<c: c]s")
>>> left_merge_1 = left_tree_1.merge(left_tree_2)
>>> left_merge_1
[<b a: c =c]c
>>> left_merge_2 = left_merge_1.merge(left_tree_3)
>>> left_merge_2
[<b a c: c]c
```

*Note: Some proof is needed that this addition to the formalism is still equivalent to the original formalism. For now, I'll show in section 3.5 that the merge direction only applies for attaching a head to a complement (either initially or finally) and we'll also see that both versions of the formalism can arrive at the same results in those cases.

3.4. Parser

I implemented a parser as described in section 2.4, with the following adaptations to Minimalist Grammars of the algorithm:

1. Initialize the chart to the empty set.
2. Initialize the agenda to those items in the grammar that contain the empty substring or some substring of the goal sentence (other items will not be able to produce the goal sentence).
3. Initialize the list of possible goals to: [[>*goal-sentence*: IP]s, [>*goal-sentence*: IP]c, [<*goal-sentence*: IP]s, [<*goal-sentence*: IP]c].
4. Repeat the following steps until the agenda is exhausted:
    4.1. Select an item from the agenda, called the trigger item, and remove it from the agenda.
    4.2. Add the trigger item to the chart, if the item is not already in the chart.
    4.3. If the trigger item was added to the chart, generate all items that can be derived from the trigger item by one operation – either a *move* of the trigger item itself or a *merge* with any item in the chart either as the selector or as the selected – and add these generated items to the agenda.
5. If a goal item is in the chart, the goal is proved, i.e., the string is recognized, otherwise it is not.

The code can be found in the following file:

- BottomUpParser.py

When running, the parser successfully determined whether a sentence is 'grammatical' for every tested grammar and sentence (both grammatical and ungrammatical sentences were tested).

Some results:
(The first two example grammars here are taken from Harkema (2001), the third from Stabler (2013))

Note: In the following lexicons, the substring '@' is a way of representing the empty string.

A simple linguistic grammar:

- [>Lavinia:d -k]s
- [>Titus:d -k]s
- [>praise:=d vt -v]s
- [>s:=pred +v +k i]s
- [>@:=i IP]s
- [>@:=vt +k =d pred]s

Output:

```
Parsing the sentence: Titus praise s Lavinia
...
The sentence is in my language.
Goal item: [>Titus praise s Lavinia: IP]c
Derivation:
['Merge: [>praise: vt =d -v]s + [>Lavinia: d -k]s = [>praise: vt -v, Lavinia: -k]c',
 'Merge: [>@: pred =vt =d +k]s + [>praise: vt -v, Lavinia: -k]c =
                                 [>@: pred =d +k, praise: -v, Lavinia: -k]c',
 'Merge: [>@: pred =d +k, praise: -v, Lavinia: -k]c + [>Titus: d -k]s =
                                 [>@: pred +k, Titus: -k, praise: -v, Lavinia: -k]c',
 'Move: [>@: pred +k, Titus: -k, praise: -v, Lavinia: -k]c =
                                 [>Lavinia: pred, Titus: -k, praise: -v]c',
 'Merge: [>s: i =pred +v +k]s + [>Lavinia: pred, Titus: -k, praise: -v]c =
                                 [>s Lavinia: i +v +k, Titus: -k, praise: -v]c',
 'Move: [>s Lavinia: i +v +k, Titus: -k, praise: -v]c =
                                 [>praise s Lavinia: i +k, Titus: -k]c',
 'Move: [>praise s Lavinia: i +k, Titus: -k]c = [>Titus praise s Lavinia: i]c',
 'Merge: [>@: IP =i]s + [>Titus praise s Lavinia: i]c =
                                 [>Titus praise s Lavinia: IP]c']


Elapsed time: 0.0150001049042
```

A visualization of the derivation:

An abstract grammar (should parse sentences of the form $a^n b^n d^n$):

- [>@: IP]s
- [>@: =a +d +b +a IP]s
- [>a: =b a -a]s
- [>b:=d b -b]s
- [>d: d -d]s
- [>a:=b +a a -a]s
- [>b:=d +b b -b]s
- [>d:=a +d d -d]s

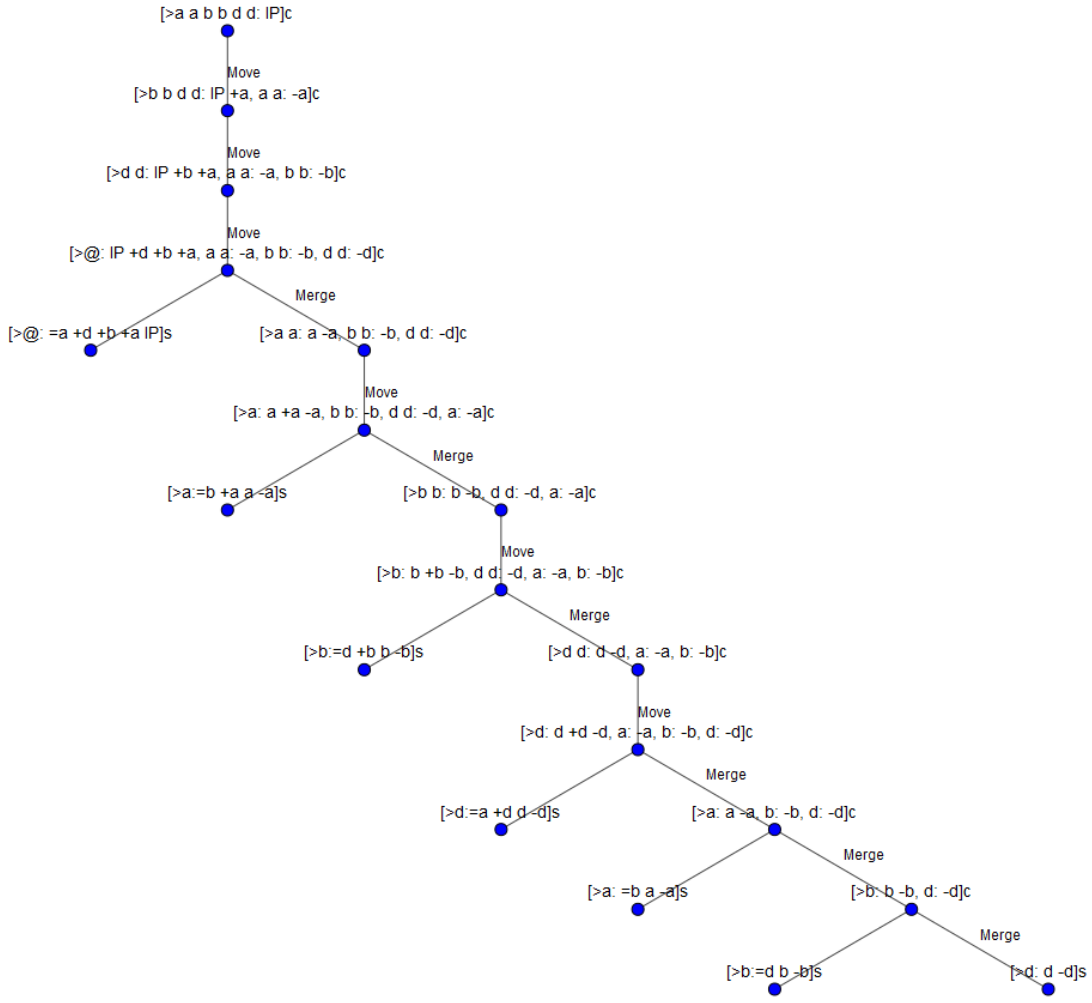
Output:

```
Parsing the sentence: a a b b d d
...
The sentence is in my language.
Goal item: [>a a b b d d: IP]c
Derivation:
['Merge: [>b: b =d -b]s + [>d: d -d]s = [>b: b -b, d: -d]c',
 'Merge: [>a: a =b -a]s + [>b: b -b, d: -d]c = [>a: a -a, b: -b, d: -d]c',
 'Merge: [>d: d =a +d -d]s + [>a: a -a, b: -b, d: -d]c =
                                  [>d: d +d -d, a: -a, b: -b, d: -d]c',
 'Move: [>d: d +d -d, a: -a, b: -b, d: -d]c = [>d d: d -d, a: -a, b: -b]c',
 'Merge: [>b: b =d +b -b]s + [>d d: d -d, a: -a, b: -b]c =
                                  [>b: b +b -b, d d: -d, a: -a, b: -b]c',
 'Move: [>b: b +b -b, d d: -d, a: -a, b: -b]c = [>b b: b -b, d d: -d, a: -a]c',
 'Merge: [>a: a =b +a -a]s + [>b b: b -b, d d: -d, a: -a]c =
                                  [>a: a +a -a, b b: -b, d d: -d, a: -a]c',
 'Move: [>a: a +a -a, b b: -b, d d: -d, a: -a]c = [>a a: a -a, b b: -b, d d: -d]c',
 'Merge: [>@: IP =a +d +b +a]s + [>a a: a -a, b b: -b, d d: -d]c =
                                  [>@: IP +d +b +a, a a: -a, b b: -b, d d: -d]c',
 'Move: [>@: IP +d +b +a, a a: -a, b b: -b, d d: -d]c =
                                  [>d d: IP +b +a, a a: -a, b b: -b]c',
 'Move: [>d d: IP +b +a, a a: -a, b b: -b]c = [>b b d d: IP +a, a a: -a]c',
 'Move: [>b b d d: IP +a, a a: -a]c = [>a a b b d d: IP]c']



Elapsed time: 0.0569999217987
```

A visualization of the derivation:



$a^5b^5d^5$ is parsed in ~4 seconds.

$a^6b^6d^6$ is parsed in over a minute.

A more complex linguistic grammar:

- [>@:=v IP]s
- [>@:=v +wh IP]s
- [>knows:=IP =d v]s
- [>says:=IP =d v]s
- [>prefers:=d =d v]s
- [>drinks:=d =d v]s
- [>king:n]s
- [>queen:n]s
- [>wine:n]s
- [>beer:n]s
- [>the:=n d]s
- [>which:=n d -wh]s

Output:

```
Parsing the sentence: the king knows which wine the queen prefers
...
The sentence is in my language.
Item: [>the king knows which wine the queen prefers: IP]c
Derivation:
['Merge: [>which: d =n -wh]s + [>wine: n]s = [>which wine: d -wh]c',
 'Merge: [>prefers: v =d =d]s + [>which wine: d -wh]c =
                               [>prefers: v =d, which wine: -wh]c',
 'Merge: [>the: d =n]s + [>queen: n]s = [>the queen: d]c',
 'Merge: [>prefers: v =d, which wine: -wh]c + [>the queen: d]c =
                               [>the queen prefers: v, which wine: -wh]c',
 'Merge: [>@: IP =v +wh]s + [>the queen prefers: v, which wine: -wh]c =
                               [>the queen prefers: IP +wh, which wine: -wh]c',
 'Move: [>the queen prefers: IP +wh, which wine: -wh]c =
                               [>which wine the queen prefers: IP]c',
 'Merge: [>knows: v =IP =d]s + [>which wine the queen prefers: IP]c =
                               [>knows which wine the queen prefers: v =d]c',
 'Merge: [>the: d =n]s + [>king: n]s = [>the king: d]c',
 'Merge: [>knows which wine the queen prefers: v =d]c + [>the king: d]c =
                               [>the king knows which wine the queen prefers: v]c',
 'Merge: [>@: IP =v]s + [>the king knows which wine the queen prefers: v]c =
                               [>the king knows which wine the queen prefers: IP]c']


Elapsed time: 0.00999999046326
```

A visualization of the derivation:

3.5. Learner + Annealer

I implemented a general Simulated Annealing learner and a specific Minimalist Grammars supporting module which I called an annealer in the following files:

- SimulatedAnnealingLearner.py
- MinimalistGrammarAnnealer.py

The learner is a generic implementation of the Simulated Annealing algorithm from section 2.6 which gets an annealer that it uses for the more specific functions.
The learner also has a logging functionality, selected run logs can be found at:
https://bitbucket.org/taucompling/HeadComplementSyntacticLearner/src
under the directory "Learner Logs".

The annealer is specific to Minimalist Grammars and exports the following functions:

- get_initial_hypothesis – returns an initial hypothesis for a grammar that can generate the input.
- random_neighbour – generates a random neighbour to the current hypothesis.
- energy – returns the current hypothesis' energy.

The learner uses these functions to follow the Simulated Annealing algorithm.

3.5.1. Initial Hypothesis

Ideally, we'd want to start our learning process with a blank slate – start with the most general grammar that can parse virtually any input and then have the learner find the best possible grammar to parse the input.
However, this proves to be difficult. The most general hypothesis, as we've seen in section 2.5, is one where every word can merge with any other word. To get from that to a grammar we'd recognize without any limitations, the learner would have to guess all of the grammatical categories in addition to the head-complement order we want to test. The search space in this case is immense; Minimalist Grammars allow for a lot of possible variations between items. Our learner would not be able to accomplish this in a reasonable amount of time.
Therefore, some basic assumptions must be made about the grammar, the input, or the initial state of the "brain".

In this work, I chose to make an assumption about the initial state. I assume that an underlying representation of grammatical categories exists in the “brain”.
That is, the “brain” knows what kinds of categories it expects to learn and their interactions but not which word corresponds to which category.
The initial hypothesis would be that any word in the input could be of any lexical category and of any head-direction.
In order to do this, the annealer gets a blank grammar object with all the possible categories and generates an initial hypothesis that is the ‘Cartesian product’ of the blank grammar and the set of all words in all sentences of the input.

The blank grammar used in this work is found in the file:

- BlankGrammars.py

```
KAYNE_LEXICON_WITH_EMPTY_DP = [
    # Absolute:
    MinimalistGrammarTree("[>@:IP =VP =DP]s"),
    MinimalistGrammarTree("[>*:VP]s"),
    MinimalistGrammarTree("[>*:DP]s"),
    MinimalistGrammarTree("[>@:VP =PP =VP]s"),

    # Head initial:
    MinimalistGrammarTree("[>*:CP =IP]s"),
    MinimalistGrammarTree("[>*:VP =DP]s"),
    MinimalistGrammarTree("[>*:VP =CP]s"),
    MinimalistGrammarTree("[>*:PP =DP]s"),

    # Head Final:
    MinimalistGrammarTree("[>@:DP =DP -O]s"),
    MinimalistGrammarTree("[>@:IP =IP -Comp]s"),
    MinimalistGrammarTree("[>*:CP =IP +Comp -Oc]s"),
    MinimalistGrammarTree("[>*:VP =DP +O]s"),
    MinimalistGrammarTree("[>*:VP =CP +Oc]s"),
    MinimalistGrammarTree("[>*:PP =DP +O]s"),
    ]
KAYNE_GRAMMAR_WITH_EMPTY_DP = MinimalistGrammar(KAYNE_LEXICON_WITH_EMPTY_DP)
```

The character “@” represents the empty string and the character “*” represents a blank category – one that will be replaced by a word.

An initial hypothesis using the input ['Elaine walked', 'Kramer hated George']:

```
Initial hypothesis: [
[>@: DP =DP -O]s, [>@: VP =PP =VP]s, [>@: IP =VP =DP]s, [>@: IP =IP -Comp]s,
[>George: VP]s, [>George: DP]s, [>George: CP =IP]s, [>George: VP =DP]s, [>George: VP
=CP]s, [>George: PP =DP]s, [>George: VP =DP +O]s, [>George: PP =DP +O]s, [>George: VP
=CP +Oc]s, [>George: CP =IP +Comp -Oc]s,
[>Elaine: VP]s, [>Elaine: DP]s, [>Elaine: VP =DP]s, [>Elaine: CP =IP]s, [>Elaine: VP
=CP]s, [>Elaine: PP =DP]s, [>Elaine: VP =DP +O]s, [>Elaine: PP =DP +O]s, [>Elaine: VP
=CP +Oc]s, [>Elaine: CP =IP +Comp -Oc]s,
[>Kramer: VP]s, [>Kramer: DP]s, [>Kramer: CP =IP]s, [>Kramer: VP =DP]s, [>Kramer: VP
=CP]s, [>Kramer: PP =DP]s, [>Kramer: VP =DP +O]s, [>Kramer: PP =DP +O]s, [>Kramer: VP
=CP +Oc]s, [>Kramer: CP =IP +Comp -Oc]s,
```

```
[>walked: VP]s, [>walked: DP]s, [>walked: CP =IP]s, [>walked: VP =DP]s, [>walked: VP
=CP]s, [>walked: PP =DP]s, [>walked: VP =DP +O]s, [>walked: PP =DP +O]s, [>walked: VP
=CP +Oc]s, [>walked: CP =IP +Comp -Oc]s,
[>hated: VP]s, [>hated: DP]s, [>hated: CP =IP]s, [>hated: VP =DP]s, [>hated: VP =CP]s,
[>hated: PP =DP]s, [>hated: VP =DP +O]s, [>hated: PP =DP +O]s, [>hated: VP =CP +Oc]s,
[>hated: CP =IP +Comp -Oc]s
]
```

We can see that the blank grammar supports all possible head-complement orders.

For each grammatical head, two items exist in the blank grammar – one head-initial (e.g., [>*:VP =DP]s) and one head-final ([>*:VP =DP +O]s). The head-final item has a licensor and thus will require movement of the complement. Accordingly, the complement must have the corresponding licensee feature.

There are several ways to handle this requirement for a licensee. We'll look at DP as an example of a complement. For the subject position of the sentence, we need a licensee-less DP but for the head-final complement position we need a DP with a licensee. The simple way to do this is to have two DP categories. This seems as a rather inefficient way to represent nouns in the brain.

Therefore, the method I chose to represent this is by having a "wrapper" category – [>@:DP =DP -O]s – this takes a licensee-less DP and adds a licensee feature.

In the same way, [>@:IP =IP -Comp]s is the "wrapper" for IP.

Since PPs don't attach as complements, a blank category exists that attaches the PP to the right of the VP ([>@:VP =PP =VP]s).

The initial hypothesis has all items with merge direction to the right. However, this can be changed by a random neighbour function.

*Note: The initial hypothesis for our learner is quite large. It might be worthwhile to take a different representational approach. For instance, a list of categories, each containing a list of words in the category. This will also affect the encoding of the grammar. However, in this work the standard representation for Minimalist Grammars is used.

3.5.2. Random Neighbour

The random neighbour function should generate a grammar that is relatively close to the current hypothesis. I've implemented several functions that change the current

grammar slightly and the random neighbour function randomly chooses one of those to run at each iteration. The function then verifies that the new grammar can still generate the input and if so – returns the grammar. If it doesn't generate the input then it tries a different function. If after a fixed number of attempts no changed grammar is found then the current hypothesis remains.

There are several modifying functions and the choice of which of these to use is the key difference between the four versions of the learner. The different functions control which representation the learner "allows" and so the choice of modifying functions dictates the theory the learner represents.

delete – This is the most basic of the modifying functions. This function simply removes one item from the current hypothesis. Since the initial hypothesis has every word as every category, after enough *deletions*, only a small grammar with only the most necessary items will remain. This function is used in every version of the learner.
This function does not delete empty categories (categories with the empty string) since all of them are necessary in one or more merge direction and deleting them when in one direction will result in the input being unparsable in the other direction.

add_lexical_item – This function is the opposite of *delete*, it simply adds to the hypothesis a combination of an input word and a blank category that isn't in the current hypothesis (because it has already been *deleted*). This function is used in every version of the learner.

add_and_delete – This function simply calls *add_lexical_item* and then calls *delete*. This is effectively the same as replacing one item in the hypothesis with a new item. This could theoretically be done without a separate function but by the learner calling *add_lexical_item* in one iteration and *delete* in the next. However, adding an item increases the energy so that random neighbour might not be chosen even though after the deletion the energy could improve. Therefore, this is done in a single function.

change_language_direction_with_flip – This function reverses the merge direction of **all** items in the grammar. This alone doesn't guarantee that the input will be parsable with the new grammar (in fact, at late stages of the learning process it almost guarantees

that it **won't** be). Therefore, the function also changes ('flips') all head-initial items in the grammar into head-final items and vice versa. So if an item has license features they are removed and if it doesn't then they are added.
[<loved:VP =DP]s becomes [>loved:VP =DP +O]s;
[>thinks:CP =IP +Comp -Oc]s becomes [<thinks:CP =IP]s
This function is used for the Language Learner.

change_category_direction_with_flip – This function reverses the merge direction of all items in one random grammatical category of the grammar (e.g., all VP items). Again, so the input remains parsable, the function also changes all head-initial items in the category into head-final items and vice versa. This function is used for the Category Learner.

change_word_direction_with_flip – This function reverses the merge direction of all occurrences of some random word from the input (e.g., every item whose substring is "walked". Again, so the input remains parsable, the function also changes all head-initial items with the word into head-final items and vice versa. This function is used for the Word Learner.

The functions used for each learner are:
Language Learner:

- *delete*
- *add_lexical_item*
- *add_and_delete*
- *change_language_direction_with_flip*

Category Learner:

- *delete*
- *add_lexical_item*
- *add_and_delete*
- *change_category_direction_with_flip*

Word Learner:

- *delete*
- *add_lexical_item*

- *add_and_delete*
- *change_word_direction_with_flip*
- *change_category_direction_with_flip*
- *change_language_direction_with_flip*

Kayne Learner:

- *delete*
- *add_lexical_item*
- *add_and_delete*

The Word Learner has the language and category flipping functions to make things "easier" for it. For instance, in the head-final input, it is highly unlikely that every item will be flipped before some crucial item is removed. Therefore, these functions that flip more than one item are used.

*Note: To improve running time, another neighbour function was added, *add_probable_lexical_item*. This function is the same as *add_lexical_item* except it only adds items in categories that already exist in the grammar. This helps in later stages of the learning process when a general head direction has been learned except for a few heads. The function will add the desired items faster than *add_lexical_item* will (although both will get them eventually if given enough iterations). An equivalent *add_probable_and_delete* was also added.

In addition, *delete* is selected with greater probability than other neighbour functions to speed up the early stages of the learning process when there are many possible items and thus parsing the input takes a very long time.

Both these changes make no theoretical difference in the learning process. All they do is (greatly) reduce the time it takes the learner to run.

3.5.3. Energy

The *energy* function returns the energy of the current hypothesis which is actually the encoding length of the current grammar and of the input using the grammar. This is done using the calculation described in section 2.5 with some slight variations to account for the different learner versions.

For all learner versions, calculations for the length of the encoding for the input are the same and are done exactly as described in section 2.5.
The differences in energy lie in the grammar encoding length and are as follows:

- Language Learner – one bit is added to the grammar encoding to signify the merge direction for the language.
- Category Learner – one bit is added per category (VP, DP, etc.) to signify the merge direction for that category.
- Word learner – one bit is added for each item in the grammar to encode the merge direction of that item.
- Kayne Learner – the calculation is exactly as described in section 2.5.

3.6. Running Time Considerations

Various factors affect the running time of the learner. Factors such as size of the input or the rate of temperature decrease have a proportional effect on the running time. However, the factors that have the most drastic influence on the running time are factors which influence the running time of the parser. Since the parser runs many times per iteration of the learner, any increase in the running time of the parser has a huge effect on the running time of the learner.

Three major factors affect the parser:

- The complexity of the input sentences.
- The complexity of the grammar.
- The efficiency of the parser itself.

The complexity of the input sentences cannot be helped, they are as simple as can be given the requirements of the learner.
The complexity of the grammar basically depends on the specifics of the blank grammar chosen for the initial hypothesis.
The efficiency of the parser is something that could have been improved and when, during the development process, I was delayed by long running times I decided to do so.

I refactored several modules so that each would operate with integer constants instead of strings:

- NumberMinimalistGrammarNode.py
- NumberMinimalistGrammarTree.py
- NumberBottomUpParser.py

Since readability is crucial, in most functions the learner still uses the string modules. When the time comes for parsing, however, the new number parser is called. This module first translates the grammar and the input to integers – it assigns each word of the input sentence an index and for each select or license feature it assigns a number. Then the parsing is done with integer operations.

During runs, using the number parser seems to be about six times faster than the straight-forward string parser.

4. Expected Results

For the fully head-initial inputs we'd expect all learners to return the same grammar:
(Grammar 1)

- [>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s,
  [>@: IP =IP -Comp]s,
- [>Jerry: DP]s, [>George: DP]s, [>Elaine: DP]s, [>Kramer: DP]s,
- [>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
- [>liked: VP =DP]s, [>saw: VP =DP]s, [>loved: VP =DP]s, [>hated: VP =DP]s,
- [>with: PP =DP]s, [>by: PP =DP]s, [>above: PP =DP]s, [>under: PP =DP]s,
- [>that: CP =IP]s,
- [>knows: VP =CP]s, [>says: VP =CP]s, [>thinks: VP =CP]s,
  [>assumes: VP =CP]s]

The categories with empty strings are neither deleted nor modified in any of the learners. Additionally, all words have their appropriate category and the merge direction is to the right.
Generating the sentence "Jerry saw George" will take the following derivation:

1. Merge: [>saw: VP =DP]s + [>George: DP]s = [>saw George: VP]c
2. Merge: [>@: IP =VP =DP]s + [>saw George: VP]c = [>saw George: IP =DP]c
3. Merge: [>saw George: IP =DP]c + [>Jerry: DP]s = [>Jerry saw George: IP]c

For fully head-final inputs we'd expect the Kayne leaner to return the following grammar:
(Grammar 2)

- [>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s,
  [>@: IP =IP -Comp]s,
- [>Jerry: DP]s, [>George: DP]s, [>Elaine: DP]s, [>Kramer: DP]s,
- [>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
- [>liked: VP =DP +O]s, [>saw: VP =DP +O]s, [>loved: VP =DP +O]s,
  [>hated: VP =DP +O]s,
- [>with: PP =DP +O]s, [>by: PP =DP +O]s, [>above: PP =DP +O]s,
  [>under: PP =DP +O]s,
- [>that: CP =IP +Comp -Oc]s,

- [>knows: VP =CP +Oc]s, [>says: VP =CP +Oc]s, [>thinks: VP =CP +Oc]s,
  [>assumes: VP =CP +Oc]s]

We see that all of the heads have licensor features so they require movement. The merge direction is "right" so with the movement the heads are all head-final.
Generating the head-final sentence "Jerry George saw" will take the following derivation:

1. Merge: [>@: DP =DP -O]s + [>George: DP]s = [>George: DP -O]c
2. Merge: [>saw: VP =DP +O]s + [>George: DP -O]c = [>saw: VP =O, George: -O]c
3. Move: [>saw: VP =O, George: -O]c = [>George saw: VP]c
4. Merge: [>@: IP =VP =DP]s + [>George saw: VP]c = [>George saw: IP =DP]c
5. Merge: [>George saw: IP =DP]c + [>Jerry: DP]s = [>Jerry George saw: IP]c

For the other learners, we'd expect the merge direction to change for the head-final input:
(Grammar 3)

- [>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s,
  [>@: IP =IP -Comp]s,
- [<Jerry: DP]s, [<George: DP]s, [<Elaine: DP]s, [<Kramer: DP]s,
- [<ran: VP]s, [<walked: VP]s, [<read: VP]s, [<wrote: VP]s,
- [<liked: VP =DP]s, [<saw: VP =DP]s, [<loved: VP =DP]s, [<hated: VP =DP]s,
- [<with: PP =DP]s, [<by: PP =DP]s, [<above: PP =DP]s, [<under: PP =DP]s,
- [<that: CP =IP]s,
- [<knows: VP =CP]s, [<says: VP =CP]s, [<thinks: VP =CP]s,
  [<assumes: VP =CP]s]

The items with empty strings do not change direction – for the "licensee wrapper" the merge direction doesn't matter; the other items attach specifiers and not complements so they shouldn't change direction as this work doesn't deal with specifier direction.
The other items have merge direction "left" so they are head final.
Generating the head-final sentence "Jerry George saw" will take the following derivation:

1. Merge: [<saw: VP =DP]s + [<George: DP]s = [<George saw: VP]c
2. Merge: [>@: IP =VP =DP]s + [<George saw: VP]c = [>George saw: IP =DP]c
3. Merge: [>George saw: IP =DP]c + [<Jerry: DP]s = [>Jerry George saw: IP]c

We'll also notice that the merge direction for the DP items or for the intransitive VP items does not have any effect so we can expect some variation there with the category and word learners.

For the other input types we'd expect the correct combination of items from the above grammars.
For the mixed-category input we'd expect the Kayne learner to have right-merging, no-movement items for the head-initial categories and right-merging, movement items for the head-final categories.
The language learner should return either the same grammar or the mirror image of it – left-merging, movement items for the head-initial categories and left-merging, no-movement items for the head-final categories.
The category and word learner should return right-merging, no-movement items for the head-initial categories and left-merging, no-movement items for the head-final categories.

For the mixed-word input we'd expect something similar – the Kayne learner would return right-merging, no-movement items for the head-initial items and right-merging, movement items for the head-final items. The language and category learner would return the same or a mirror image of it.
The word learner should return right-merging, no-movement items for the head-initial items and left-merging, no-movement items for the head-final items.

In summary, these are the results we'd expect:

Kayne learner:

Head-initial input:	right-merging, no-movement items
Head-final input:	right-merging, movement items
Mixed-category input:	right-merging, movement **and** no-movement items
Mixed-word input:	right-merging, movement **and** no-movement items

Language learner:

Head-initial input:	right-merging, no-movement items
Head-final input:	left-merging, no-movement items
Mixed-category input:	right-merging **or** left-merging, movement **and** no-movement items
Mixed-word input:	right-merging **or** left-merging, movement **and** no-movement items

Category learner:

Head-initial input:	right-merging, no-movement items
Head-final input:	left-merging, no-movement items
Mixed-category input:	right-merging **and** left-merging, no-movement items
Mixed-word input:	right-merging **and** left-merging, movement **and** no-movement items

Word learner:

Head-initial input:	right-merging, no-movement items
Head-final input:	left-merging, no-movement items
Mixed-category input:	right-merging **and** left-merging, no-movement items
Mixed-word input:	right-merging **and** left-merging, no-movement items

5. Results

Below is a sampling of one of the runs and the results of all the runs for each learner on each type of input. Samplings for the other runs can be found in Appendix A.
The log files themselves can be found in:
https://bitbucket.org/taucompling/HeadComplementSyntacticLearner/src
under the directory “Learner Logs/General Case”.

5.1. Kayne Learner

5.1.1. Head Initial Input

The relevant log file for this run is:
log_2017_05_10__02_40_42 - Kayne, Head-initial, PP & CP.txt

The generated input:

```
Input is: ['Elaine wrote with Jerry', 'George hated Kramer', 'Jerry says that Jerry
wrote', 'George wrote under Elaine', 'Elaine thinks that George hated Kramer', 'George
hated Elaine', 'George walked', 'Kramer loved Elaine', 'George assumes that Kramer
wrote', 'Kramer read above Kramer', 'Elaine loved Kramer', 'Jerry walked', 'Jerry
assumes that Kramer hated Kramer', 'George liked Elaine', 'Elaine says that Kramer
walked', 'Jerry loved George under Elaine', 'Elaine says that Kramer saw George',
'Jerry ran by George', 'Elaine knows that Elaine read', 'Jerry hated Kramer', 'Jerry
loved Kramer', 'Jerry saw Elaine by Kramer', 'Jerry loved Kramer', 'Kramer hated
George with Jerry', 'Kramer ran under Elaine', 'Kramer loved George by Elaine',
'George thinks that Elaine walked under Elaine', 'Jerry loved Elaine', 'Kramer walked
above Jerry', 'Jerry says that Elaine walked by Elaine', 'Kramer knows that Jerry ran
under Elaine', 'Elaine ran by Jerry', 'Kramer hated Kramer', 'Elaine saw Elaine with
Elaine', 'Elaine loved Jerry above George', 'George saw Elaine', 'George liked
Kramer', 'Kramer wrote', 'George hated Elaine under George', 'Elaine liked George',
'Kramer assumes that George loved Elaine with Kramer', 'George ran by Elaine', 'Kramer
saw Kramer above Elaine', 'Jerry ran by Elaine', 'Jerry read under George', 'George
liked Elaine by Kramer', 'George wrote', 'George ran', 'Elaine knows that Jerry walked
by Elaine', 'Kramer liked Kramer']
```

The learner was initiated with 50 randomly generated input sentences as described in section 3.2. As we can see, the sentences are all head-initial, some have intransitive verbs, some have transitive verbs, some have prepositions and some have dependent clauses.

The initial hypothesis:

```
Initial hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP
=IP -Comp]s, [>Jerry: VP]s, [>Jerry: DP]s, [>Jerry: VP =DP]s, [>Jerry: CP =IP]s,
[>Jerry: VP =CP]s, [>Jerry: PP =DP]s, [>Jerry: VP =DP +O]s, [>Jerry: PP =DP +O]s,
[>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s, [>George: VP]s, [>George: DP]s,
[>George: VP =DP]s, [>George: CP =IP]s, [>George: VP =CP]s, [>George: PP =DP]s,
[>George: VP =DP +O]s, [>George: PP =DP +O]s, [>George: VP =CP +Oc]s, [>George: CP =IP
+Comp -Oc]s, [>Elaine: VP]s, [>Elaine: DP]s, [>Elaine: CP =IP]s, [>Elaine: VP =DP]s,
```

```
[>Elaine: VP =CP]s, [>Elaine: PP =DP]s, [>Elaine: VP =DP +O]s, [>Elaine: PP =DP +O]s,
[>Elaine: VP =CP +Oc]s, [>Elaine: CP =IP +Comp -Oc]s, [>Kramer: VP]s, [>Kramer: DP]s,
[>Kramer: CP =IP]s, [>Kramer: VP =DP]s, [>Kramer: VP =CP]s, [>Kramer: PP =DP]s,
[>Kramer: VP =DP +O]s, [>Kramer: PP =DP +O]s, [>Kramer: VP =CP +Oc]s, [>Kramer: CP =IP
+Comp -Oc]s, [>ran: VP]s, [>ran: DP]s, [>ran: CP =IP]s, [>ran: VP =DP]s, [>ran: VP
=CP]s, [>ran: PP =DP]s, [>ran: VP =DP +O]s, [>ran: PP =DP +O]s, [>ran: VP =CP +Oc]s,
[>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [>walked: DP]s, [>walked: CP =IP]s,
[>walked: VP =DP]s, [>walked: VP =CP]s, [>walked: PP =DP]s, [>walked: VP =DP +O]s,
[>walked: PP =DP +O]s, [>walked: VP =CP +Oc]s, [>walked: CP =IP +Comp -Oc]s, [>read:
VP]s, [>read: DP]s, [>read: CP =IP]s, [>read: VP =DP]s, [>read: VP =CP]s, [>read: PP
=DP]s, [>read: VP =DP +O]s, [>read: PP =DP +O]s, [>read: VP =CP +Oc]s, [>read: CP =IP
+Comp -Oc]s, [>wrote: VP]s, [>wrote: DP]s, [>wrote: CP =IP]s, [>wrote: VP =DP]s,
[>wrote: VP =CP]s, [>wrote: PP =DP]s, [>wrote: VP =DP +O]s, [>wrote: PP =DP +O]s,
[>wrote: VP =CP +Oc]s, [>wrote: CP =IP +Comp -Oc]s, [>liked: VP]s, [>liked: DP]s,
[>liked: CP =IP]s, [>liked: VP =DP]s, [>liked: VP =CP]s, [>liked: PP =DP]s, [>liked:
VP =DP +O]s, [>liked: PP =DP +O]s, [>liked: VP =CP +Oc]s, [>liked: CP =IP +Comp -Oc]s,
[>saw: VP]s, [>saw: DP]s, [>saw: CP =IP]s, [>saw: VP =DP]s, [>saw: VP =CP]s, [>saw: PP
=DP]s, [>saw: VP =DP +O]s, [>saw: PP =DP +O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP
+Comp -Oc]s, [>loved: VP]s, [>loved: DP]s, [>loved: CP =IP]s, [>loved: VP =DP]s,
[>loved: VP =CP]s, [>loved: PP =DP]s, [>loved: VP =DP +O]s, [>loved: PP =DP +O]s,
[>loved: VP =CP +Oc]s, [>loved: CP =IP +Comp -Oc]s, [>hated: VP]s, [>hated: DP]s,
[>hated: CP =IP]s, [>hated: VP =DP]s, [>hated: VP =CP]s, [>hated: PP =DP]s, [>hated:
VP =DP +O]s, [>hated: PP =DP +O]s, [>hated: VP =CP +Oc]s, [>hated: CP =IP +Comp -Oc]s,
[>with: VP]s, [>with: DP]s, [>with: CP =IP]s, [>with: VP =DP]s, [>with: VP =CP]s,
[>with: PP =DP]s, [>with: VP =DP +O]s, [>with: PP =DP +O]s, [>with: VP =CP +Oc]s,
[>with: CP =IP +Comp -Oc]s, [>by: VP]s, [>by: DP]s, [>by: CP =IP]s, [>by: VP =DP]s,
[>by: VP =CP]s, [>by: PP =DP]s, [>by: VP =DP +O]s, [>by: PP =DP +O]s, [>by: VP =CP
+Oc]s, [>by: CP =IP +Comp -Oc]s, [>above: VP]s, [>above: DP]s, [>above: CP =IP]s,
[>above: VP =DP]s, [>above: VP =CP]s, [>above: PP =DP]s, [>above: VP =DP +O]s,
[>above: PP =DP +O]s, [>above: VP =CP +Oc]s, [>above: CP =IP +Comp -Oc]s, [>under:
VP]s, [>under: DP]s, [>under: CP =IP]s, [>under: VP =DP]s, [>under: VP =CP]s, [>under:
PP =DP]s, [>under: VP =DP +O]s, [>under: PP =DP +O]s, [>under: VP =CP +Oc]s, [>under:
CP =IP +Comp -Oc]s, [>that: VP]s, [>that: DP]s, [>that: CP =IP]s, [>that: VP =DP]s,
[>that: VP =CP]s, [>that: PP =DP]s, [>that: VP =DP +O]s, [>that: PP =DP +O]s, [>that:
VP =CP +Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP]s, [>knows: DP]s, [>knows: CP
=IP]s, [>knows: VP =DP]s, [>knows: VP =CP]s, [>knows: PP =DP]s, [>knows: VP =DP +O]s,
[>knows: PP =DP +O]s, [>knows: VP =CP +Oc]s, [>knows: CP =IP +Comp -Oc]s, [>says:
VP]s, [>says: DP]s, [>says: CP =IP]s, [>says: VP =DP]s, [>says: VP =CP]s, [>says: PP
=DP]s, [>says: VP =DP +O]s, [>says: PP =DP +O]s, [>says: VP =CP +Oc]s, [>says: CP =IP
+Comp -Oc]s, [>thinks: VP]s, [>thinks: DP]s, [>thinks: CP =IP]s, [>thinks: VP =DP]s,
[>thinks: VP =CP]s, [>thinks: PP =DP]s, [>thinks: VP =DP +O]s, [>thinks: PP =DP +O]s,
[>thinks: VP =CP +Oc]s, [>thinks: CP =IP +Comp -Oc]s, [>assumes: VP]s, [>assumes:
DP]s, [>assumes: CP =IP]s, [>assumes: VP =DP]s, [>assumes: VP =CP]s, [>assumes: PP
=DP]s, [>assumes: VP =DP +O]s, [>assumes: PP =DP +O]s, [>assumes: VP =CP +Oc]s,
[>assumes: CP =IP +Comp -Oc]s]
```

The learner generates an initial hypothesis that can generate the input. As described in section 3.5.1, the initial hypothesis has an item for every word in the input as every possible category from the blank grammar.

Initial energy:

```
Temperature: 100.000000
Energy: 10192
Grammar: 5143
Input: 5049
```

The learner calculates the current energy, which is the length of the grammar encoding and input encoding using the initial hypothesis.

The learner is initialized with a temperature of 100.

After 100 iterations:

```
Iteration: 100
2017_05_10__02_54_38: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: VP]s, [>Jerry: DP]s, [>Jerry: CP =IP]s, [>Jerry:
VP =CP]s, [>Jerry: VP =DP +O]s, [>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s,
[>George: VP]s, [>George: DP]s, [>George: CP =IP]s, [>George: VP =CP]s, [>George: PP
```

```
=DP]s, [>George: VP =DP +O]s, [>George: VP =CP +Oc]s, [>Elaine: DP]s, [>Elaine: VP]s,
[>Elaine: CP =IP]s, [>Elaine: VP =DP]s, [>Elaine: VP =CP]s, [>Elaine: VP =DP +O]s,
[>Kramer: VP]s, [>Kramer: DP]s, [>Kramer: VP =DP]s, [>Kramer: CP =IP]s, [>Kramer: VP
=DP +O]s, [>Kramer: PP =DP +O]s, [>ran: VP]s, [>ran: VP =DP]s, [>ran: VP =CP]s, [>ran:
PP =DP]s, [>ran: VP =DP +O]s, [>ran: VP =CP +Oc]s, [>ran: CP =IP +Comp -Oc]s,
[>walked: VP]s, [>walked: CP =IP]s, [>walked: VP =DP]s, [>walked: VP =CP]s, [>walked:
PP =DP]s, [>walked: VP =DP +O]s, [>walked: VP =CP +Oc]s, [>read: VP]s, [>read: DP]s,
[>read: VP =DP]s, [>read: PP =DP]s, [>read: VP =DP +O]s, [>read: PP =DP +O]s, [>read:
VP =CP +Oc]s, [>wrote: VP]s, [>wrote: VP =DP]s, [>wrote: VP =CP]s, [>wrote: PP =DP]s,
[>wrote: VP =DP +O]s, [>wrote: PP =DP +O]s, [>wrote: VP =CP +Oc]s, [>liked: VP]s,
[>liked: CP =IP]s, [>liked: VP =DP]s, [>liked: VP =CP]s, [>liked: PP =DP]s, [>liked:
PP =DP +O]s, [>liked: VP =CP +Oc]s, [>saw: VP =DP]s, [>saw: VP =DP +O]s, [>saw: PP =DP
+O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP +Comp -Oc]s, [>loved: VP =DP]s, [>loved: PP
=DP]s, [>loved: VP =DP +O]s, [>hated: VP]s, [>hated: CP =IP]s, [>hated: VP =DP]s,
[>hated: PP =DP]s, [>hated: VP =DP +O]s, [>hated: PP =DP +O]s, [>hated: VP =CP +Oc]s,
[>hated: CP =IP +Comp -Oc]s, [>with: VP]s, [>with: DP]s, [>with: VP =DP]s, [>with: VP
=CP]s, [>with: PP =DP]s, [>with: VP =DP +O]s, [>with: PP =DP +O]s, [>by: VP]s, [>by:
CP =IP]s, [>by: VP =CP]s, [>by: PP =DP]s, [>by: VP =DP +O]s, [>by: VP =CP +Oc]s, [>by:
CP =IP +Comp -Oc]s, [>above: VP]s, [>above: DP]s, [>above: CP =IP]s, [>above: VP
=DP]s, [>above: VP =CP]s, [>above: PP =DP]s, [>above: PP =DP +O]s, [>above: VP =CP
+Oc]s, [>above: CP =IP +Comp -Oc]s, [>under: DP]s, [>under: VP =DP]s, [>under: PP
=DP]s, [>under: VP =CP]s, [>under: VP =DP +O]s, [>under: PP =DP +O]s, [>under: VP =CP
+Oc]s, [>under: CP =IP +Comp -Oc]s, [>that: VP]s, [>that: CP =IP]s, [>that: VP =CP]s,
[>that: PP =DP +O]s, [>that: VP =DP +O]s, [>that: CP =IP +Comp -Oc]s, [>knows: DP]s,
[>knows: CP =IP]s, [>knows: VP =DP]s, [>knows: VP =CP]s, [>knows: PP =DP]s, [>says:
VP]s, [>says: CP =IP]s, [>says: VP =CP]s, [>says: VP =DP +O]s, [>says: PP =DP +O]s,
[>says: VP =CP +Oc]s, [>says: CP =IP +Comp -Oc]s, [>thinks: VP =CP]s, [>thinks: PP
=DP]s, [>thinks: PP =DP +O]s, [>thinks: VP =CP +Oc]s, [>thinks: CP =IP +Comp -Oc]s,
[>assumes: VP]s, [>assumes: DP]s, [>assumes: CP =IP]s, [>assumes: VP =DP]s, [>assumes:
VP =CP]s, [>assumes: VP =CP +Oc]s, [>assumes: CP =IP +Comp -Oc]s]
2017_05_10__02_54_38: Energy: 8456
2017 05 10  02 54 38: Grammar: 3407
2017_05_10__02_54_38: Input: 5049
2017_05_10__02_54_38: Temperature: 74.048426
```

The learner ran for 100 iterations in which it randomly tried to delete or add items from the current hypothesis. We can see that several items have been removed. Accordingly, the grammar encoding decreased significantly. The input encoding remains unchanged. The temperature decreases by a constant factor each iteration and now stands at 74.

After 300 iterations:

```
Iteration: 300
2017_05_10__02_55_05: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: DP]s, [>Jerry: VP =CP]s, [>Jerry: VP =DP +O]s,
[>George: DP]s, [>George: VP =DP]s, [>George: VP =CP +Oc]s, [>Elaine: DP]s, [>Elaine:
PP =DP +O]s, [>Kramer: DP]s, [>Kramer: CP =IP]s, [>ran: VP]s, [>ran: CP =IP +Comp -
Oc]s, [>walked: VP]s, [>walked: VP =DP +O]s, [>walked: PP =DP +O]s, [>read: VP]s,
[>wrote: VP]s, [>wrote: VP =CP +Oc]s, [>liked: CP =IP]s, [>liked: VP =DP]s, [>liked:
PP =DP]s, [>saw: VP =DP]s, [>saw: VP =CP]s, [>saw: CP =IP +Comp -Oc]s, [>loved: VP
=DP]s, [>hated: VP =DP]s, [>hated: VP =CP]s, [>with: PP =DP]s, [>with: PP =DP +O]s,
[>by: VP]s, [>by: PP =DP]s, [>above: DP]s, [>above: PP =DP]s, [>under: PP =DP]s,
[>under: VP =DP]s, [>under: PP =DP +O]s, [>under: CP =IP +Comp -Oc]s, [>that: CP
=IP]s, [>that: PP =DP +O]s, [>knows: VP =CP]s, [>says: DP]s, [>says: VP]s, [>says: VP
=CP]s, [>says: VP =CP +Oc]s, [>thinks: VP =CP]s, [>assumes: DP]s, [>assumes: VP =CP]s,
[>assumes: CP =IP]s]
2017_05_10__02_55_05: Energy: 5067
2017_05_10__02_55_05: Grammar: 1206
2017_05_10__02_55_05: Input: 3861
2017_05_10__02_55_05: Temperature: 40.602006
```

We can see that many more items have been removed. The input encoding also decreased significantly. This can happen for several reasons. It's possible that the parser didn't find the best derivation before and by removing a certain item we 'forced' it to find a better one. However, it is far more likely that this decrease is due to the number

of items in the grammar decreasing. This affects the size ‘k’ discussed in section 2.5 and the input encoding shrinks accordingly.
The temperature again decreased by a constant factor.

After over 400 iterations:

```
Iteration: 442
2017_05_10__02_55_16: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>George: DP]s, [>Elaine: DP]s, [>Kramer: DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s, [>loved: VP =DP]s, [>hated: VP =DP]s,
[>with: PP =DP]s, [>by: PP =DP]s, [>above: PP =DP]s, [>under: PP =DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>thinks: VP =CP]s, [>assumes: VP =CP]s]
2017_05_10__02_55_16: Energy: 3912
2017_05_10__02_55_16: Grammar: 543
2017_05_10__02_55_16: Input: 3369
2017_05_10__02_55_16: Temperature: 26.500860
```

This is the final grammar the learner returns, after this iteration the hypothesis does not change anymore.
We can see that we get the grammar we’d expect. All the items are head-initial and every word has the appropriate category. Furthermore, there are no unnecessary items.

5.1.2. Head Final Input

The relevant log file for this run is:
log_2017_05_10__02_57_53 - Kayne, Head-final, PP & CP.txt

After over 500 iterations:

```
Iteration: 599
2017_05_10__03_47_30: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>Jerry: PP =DP]s, [>Jerry: VP =DP]s,
[>George: DP]s, [>George: PP =DP]s,
[>Elaine: DP]s, [>Elaine: PP =DP]s, [>Elaine: VP =DP]s,
[>Kramer: DP]s, [>Kramer: PP =DP]s, [>Kramer: VP =DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: DP]s, [>saw: DP]s, [>loved: DP]s, [>hated: DP]s,
[>with: DP]s, [>by: DP]s, [>above: DP]s, [>under: DP]s,
[>that: CP =IP +Comp -Oc]s,
[>knows: VP =CP +Oc]s, [>says: VP =CP +Oc]s, [>thinks: VP =CP +Oc]s]
2017_05_10__03_47_30: Energy: 4664
2017_05_10__03_47_30: Grammar: 664
2017_05_10__03_47_30: Input: 4000
2017_05_10__03_47_30: Temperature: 16.534834
```

This is the final grammar learned by the learner. We see that we don’t exactly get the grammar we’d expect. The items that involve CP are all indeed head-final as we’d expect but the prepositions and transitive verbs are not. From the learner’s perspective, this makes sense as the chosen grammar has a lower energy than the result we expect. This will be discussed further in section 6.

5.1.3. Mixed Category Input

In the generated input, VPs and CPs are head-initial and PPs are head-final.
The relevant log file for this run is:
log_2017_05_10__03_54_20 - Kayne, Mixed-category, PP & CP.txt

After over 900 iterations:

```
Iteration: 912
2017_05_10__06_05_48: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>Jerry: PP =DP]s,
[>George: DP]s, [>George: PP =DP]s,
[>Elaine: DP]s, [>Elaine: PP =DP]s,
[>Kramer: DP]s, [>Kramer: PP =DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s, [>loved: VP =DP]s, [>hated: VP =DP]s,
[>with: DP]s, [>by: DP]s, [>above: DP]s, [>under: DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>thinks: VP =CP]s, [>assumes: VP =CP]s]
2017_05_10__06_05_48: Energy: 4797
2017_05_10__06_05_48: Grammar: 615
2017_05_10__06_05_48: Input: 4182
2017_05_10__06_05_48: Temperature: 6.456310
```

We see that the VPs and CPs are indeed head-initial as we'd expect but the PPs are again interpreted differently. The reason is the same as before and again, this will be discussed further in section 6.

5.1.4. Mixed Word Input

In the generated input, CPs are head-initial, PPs are head-final and simple transitive VPs (VPs that take a DP as complement) are mixed – "liked" and "saw" are head-initial; "loved" and "hated" are head-final.
The relevant log file for this run is:
log_2017_05_10__06_09_15 - Kayne, Mixed-word, PP & CP.txt

After over 600 iterations:

```
Iteration: 668
2017_05_10__06_47_58: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>Jerry: PP =DP]s,
[>George: DP]s, [>George: PP =DP]s,
[>Elaine: DP]s, [>Elaine: PP =DP]s,
[>Kramer: DP]s, [>Kramer: PP =DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s,
[>loved: VP =DP +O]s, [>hated: VP =DP +O]s,
[>with: DP]s, [>by: DP]s, [>above: DP]s, [>under: DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>thinks: VP =CP]s, [>assumes: VP =CP]s]
2017_05_10__06_47_58: Energy: 4498
2017_05_10__06_47_58: Grammar: 621
```

```
2017_05_10__06_47_58: Input: 3877
2017_05_10__06_47_58: Temperature: 13.438963
```

We see that the CPs are indeed head-initial as we'd expect but the PPs are again interpreted differently. Furthermore, the head-initial VPs are learned successfully as head-initial and the head-final VPs are learned successfully as head-final. This will be discussed further in section 6.

5.2. Language Learner

5.2.1. Head Initial Input

The relevant log file for this run is:

log_2017_05_10__06_51_45 - Language, Head-initial, PP & CP.txt

After over 500 iterations:

```
Iteration: 596
2017_05_10__08_38_30: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>George: DP]s, [>Elaine: DP]s, [>Kramer: DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s, [>loved: VP =DP]s, [>hated: VP =DP]s,
[>with: PP =DP]s, [>by: PP =DP]s, [>above: PP =DP]s, [>under: PP =DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>thinks: VP =CP]s, [>assumes: VP =CP]s]
2017_05_10__08_38_30: Energy: 4111
2017_05_10__08_38_30: Grammar: 544
2017_05_10__08_38_30: Input: 3567
2017_05_10__08_38_30: Temperature: 16.684544
```

The expected grammar is learned successfully.

5.2.2. Head Final Input

The relevant log file for this run is:

log_2017_05_10__13_38_44 - Language, Head-final, PP & CP.txt

After over 400 iterations:

```
Iteration: 470
2017_05_10__14_41_32: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[<Jerry: DP]s, [<George: DP]s, [<Elaine: DP]s, [<Kramer: DP]s,
[<ran: VP]s, [<walked: VP]s, [<read: VP]s, [<wrote: VP]s,
[<liked: VP =DP]s, [<saw: VP =DP]s, [<loved: VP =DP]s, [<hated: VP =DP]s,
[<with: PP =DP]s, [<by: PP =DP]s, [<above: PP =DP]s, [<under: PP =DP]s,
[<that: CP =IP]s,
[<knows: VP =CP]s, [<thinks: VP =CP]s, [<assumes: VP =CP]s]
2017_05_10__14_41_32: Energy: 4097
2017_05_10__14_41_32: Grammar: 521
2017_05_10__14_41_32: Input: 3576
2017_05_10__14_41_32: Temperature: 24.362643
```

We see that the language direction of the learned grammar is “left” and the items are all without license features – thus, without movement. Therefore, the whole grammar is head-final, as we’d expect.

5.2.3. Mixed Category Input

In the generated input, VPs and CPs are head-initial and PPs are head-final.
The relevant log file for this run is:
log_2017_05_10__14_45_15 - Language, Mixed-category, PP & CP.txt

After over 500 iterations:

```
Iteration: 546
2017_05_10__16_33_52: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>Jerry: PP =DP]s,
[>George: DP]s, [>George: PP =DP]s,
[>Elaine: DP]s, [>Elaine: PP =DP]s,
[>Kramer: DP]s, [>Kramer: PP =DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s, [>loved: VP =DP]s, [>hated: VP =DP]s,
[>with: DP]s, [>by: DP]s, [>above: DP]s, [>under: DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>thinks: VP =CP]s, [>assumes: VP =CP]s]
2017_05_10__16_33_52: Energy: 4435
2017_05_10__16_33_52: Grammar: 616
2017_05_10__16_33_52: Input: 3819
2017_05_10__16_33_52: Temperature: 19.389046
```

We see that the VPs and CPs are indeed head-initial as we’d expect but the PPs are again interpreted differently. This will be discussed further in section 6.

5.2.4. Mixed Word Input

In the generated input, CPs are head-initial, PPs are head-final and simple transitive VPs (VPs that take a DP as complement) are mixed – “liked” and “saw” are head-initial; “loved” and “hated” are head-final.
The relevant log file for this run is:
log_2017_05_10__20_19_58 - Language, Mixed-word, PP & CP.txt

After over 1000 iterations:

```
Iteration: 1437
2017_05_10__21_15_20: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>Jerry: VP =DP]s, [>Jerry: PP =DP]s,
[>George: DP]s, [>George: VP =DP]s, [>George: PP =DP]s,
[>Elaine: DP]s, [>Elaine: VP =DP]s, [>Elaine: PP =DP]s,
[>Kramer: DP]s, [>Kramer: VP =DP]s, [>Kramer: PP =DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
```

```
[>liked: VP =DP]s, [>saw: VP =DP]s,
[>loved: DP]s, [>hated: DP]s,
[>with: DP]s, [>by: DP]s, [>above: DP]s, [>under: DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>thinks: VP =CP]s, [>assumes: VP =CP]s]
2017_05_10__21_15_20: Energy: 4702
2017_05_10__21_15_20: Grammar: 698
2017_05_10__21_15_20: Input: 4004
2017_05_10__21_15_20: Temperature: 1.333345
```

We see that the CPs are indeed head-initial as we'd expect but the PPs are again interpreted differently. Furthermore, the head-initial VPs are learned successfully as head-initial and the head-final VPs are learned successfully as head-final. This will be discussed further in section 6.

5.3. Category Learner

5.3.1. Head Initial Input

The relevant log file for this run is:
log_2017_05_10__21_17_43 - Category, Head-initial, PP & CP.txt

After over 400 iterations:

```
Iteration: 433
2017_05_10__21_40_20: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>George: DP]s, [>Elaine: DP]s, [>Kramer: DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s, [>loved: VP =DP]s, [>hated: VP =DP]s,
[>with: PP =DP]s, [>by: PP =DP]s, [>above: PP =DP]s, [>under: PP =DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>assumes: VP =CP]s]
2017_05_10__21_40_20: Energy: 3491
2017_05_10__21_40_20: Grammar: 525
2017_05_10__21_40_20: Input: 2966
2017_05_10__21_40_20: Temperature: 27.227236
```

The expected grammar is learned successfully.

5.3.2. Head Final Input

The relevant log file for this run is:
log_2017_05_10__21_43_43 - Category, Head-final, PP & CP.txt

After over 800 iterations:

```
Iteration: 880
2017_05_11__01_57_58: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>George: DP]s, [>Elaine: DP]s, [>Kramer: DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[<liked: VP =DP]s, [<saw: VP =DP]s, [<loved: VP =DP]s, [<hated: VP =DP]s,
```

```
[<with: PP =DP]s, [<by: PP =DP]s, [<above: PP =DP]s, [<under: PP =DP]s,
[<that: CP =IP]s,
[<knows: VP =CP]s, [<says: VP =CP]s, [<thinks: VP =CP]s, [<assumes: VP =CP]s]
2017_05_11__01_57_58: Energy: 4531
2017_05_11__01_57_58: Grammar: 548
2017_05_11__01_57_58: Input: 3983
2017_05_11__01_57_58: Temperature: 7.107867
```

The expected grammar is learned successfully.

5.3.3. Mixed Category Input

In the generated input, VPs and CPs are head-initial and PPs are head-final.

The relevant log file for this run is:

log_2017_05_11__10_30_47 - Category, Mixed-category, PP & CP.txt

After over 400 iterations:

```
Iteration: 482
2017_05_11__11_01_38: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [<George: DP]s, [<Elaine: DP]s, [>Kramer: DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s, [>loved: VP =DP]s, [>hated: VP =DP]s,
[<with: PP =DP]s, [<by: PP =DP]s, [<above: PP =DP]s, [<under: PP =DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>assumes: VP =CP]s]
2017_05_11__11_01_38: Energy: 3795
2017_05_11__11_01_38: Grammar: 525
2017_05_11__11_01_38: Input: 3270
2017_05_11__11_01_38: Temperature: 23.499916
```

We see that the expected grammar is learned – VPs and CPs are have merge direction "right" and are thus head-initial and PPs have merge direction "left" and are thus head-final.

5.3.4. Mixed Word Input

In the generated input, CPs are head-initial, PPs are head-final and simple transitive VPs (VPs that take a DP as complement) are mixed – "liked" and "saw" are head-initial; "loved" and "hated" are head-final.

The relevant log file for this run is:

log_2017_05_12__10_47_11 - Category, Mixed-word, PP & CP.txt

After over 500 iterations:

```
Iteration: 579
2017_05_12__12_13_39: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>George: DP]s, [>Elaine: DP]s, [>Kramer: DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s,
```

```
[>loved: VP =DP +O]s, [>hated: VP =DP +O]s,
[<with: PP =DP]s, [<by: PP =DP]s, [<above: PP =DP]s, [<under: PP =DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>thinks: VP =CP]s, [>assumes: VP =CP]s]
2017_05_12__12_13_39: Energy: 4399
2017_05_12__12_13_39: Grammar: 554
2017_05_12__12_13_39: Input: 3845
2017_05_12__12_13_39: Temperature: 17.558874
```

We see that the PPs have merge direction "left" and are thus head-final as expected; the CPs have merge direction "right" and are thus head-initial as expected. The VPs have merge direction "right" and the two head-initial ones are indeed learned correctly. The two head-final ones are also learned correctly as head-final items. This will be discussed further in section 6.

5.4. Word Learner

5.4.1. Head Initial Input

The relevant log file for this run is:

log_2017_05_13__18_57_05 - Word, Head-initial, PP & CP.txt

After over 1000 iterations:

```
Iteration: 1198
2017_05_13__21_38_07: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [<George: DP]s, [>Elaine: DP]s, [>Kramer: DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s, [>loved: VP =DP]s, [>hated: VP =DP]s,
[>with: PP =DP]s, [>by: PP =DP]s, [>above: PP =DP]s, [>under: PP =DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>thinks: VP =CP]s, [>assumes: VP =CP]s]
2017_05_13__21_38_07: Energy: 3999
2017_05_13__21_38_07: Grammar: 544
2017_05_13__21_38_07: Input: 3455
2017_05_13__21_38_07: Temperature: 2.734007
```

The expected grammar is learned successfully.

5.4.2. Head Final Input

The relevant log file for this run is:

log_2017_05_12__18_11_05 - Word, Head-final, PP & CP.txt

After over 900 iterations:

```
Iteration: 940
2017_05_12__23_31_29: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>George: DP]s, [>Elaine: DP]s, [>Kramer: DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
```

```
[<liked: VP =DP]s, [<saw: VP =DP]s, [<loved: VP =DP]s, [<hated: VP =DP]s,
[<with: PP =DP]s, [<by: PP =DP]s, [<above: PP =DP]s, [<under: PP =DP]s,
[<that: CP =IP]s,
[<knows: VP =CP]s, [<says: VP =CP]s, [<thinks: VP =CP]s, [<assumes: VP =CP]s]
2017_05_12__23_31_29: Energy: 4388
2017_05_12__23_31_29: Grammar: 568
2017_05_12__23_31_29: Input: 3820
2017_05_12__23_31_29: Temperature: 5.935384
```

The expected grammar is learned successfully.

5.4.3. Mixed Category Input

In the generated input, VPs and CPs are head-initial and PPs are head-final.

The relevant log file for this run is:

log_2017_05_10__19_25_25 - Word, Mixed-category, PP & CP.txt

After over 1000 iterations:

```
Iteration: 1300
2017_05_11__01_27_31: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>George: DP]s, [>Elaine: DP]s, [<Kramer: DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s, [>loved: VP =DP]s, [>hated: VP =DP]s,
[<with: PP =DP]s, [<by: PP =DP]s, [<above: PP =DP]s, [<under: PP =DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>thinks: VP =CP]s, [>assumes: VP =CP]s]
2017_05_11__01_27_31: Energy: 4342
2017_05_11__01_27_31: Grammar: 568
2017_05_11__01_27_31: Input: 3774
2017_05_11__01_27_31: Temperature: 2.012361
```

The expected grammar is learned successfully.

5.4.4. Mixed Word Input

In the generated input, CPs are head-initial, PPs are head-final and simple transitive VPs (VPs that take a DP as complement) are mixed – “liked” and “saw” are head-initial; “loved” and “hated” are head-final.

The relevant log file for this run is:

log_2017_05_13__02_56_34 - Word, Mixed-word, PP & CP.txt

After over 1000 iterations:

```
Iteration: 1187
2017_05_13__06_53_57: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>George: DP]s, [>Elaine: DP]s, [>Kramer: DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s, [<loved: VP =DP]s, [<hated: VP =DP]s,
[<with: PP =DP]s, [<by: PP =DP]s, [<above: PP =DP]s, [<under: PP =DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>thinks: VP =CP]s, [>assumes: VP =CP]s]
```

```
2017_05_13__06_53_57: Energy: 4014
2017_05_13__06_53_57: Grammar: 568
2017_05_13__06_53_57: Input: 3446
2017_05_13__06_53_57: Temperature: 2.825875
```

The expected grammar is learned successfully.

6. Result Analysis

When examining the results, the first thing we see is that when we expect head-final constituents using movement (and not merge direction) the learner often fails to learn the appropriate category and instead finds some alternative.
This can be seen in sections 5.1.2, 5.1.3, 5.1.4, 5.2.3, 5.2.4 & 5.3.4.
This is understandable given the design of the learner.

Let's say we learn according to Kayne's theory and we have the following head-final input (this is a simplified example of section 5.1.2):
"John Paul saw"
"Paul John saw"
"George Ringo saw"
"Ringo John saw"

The grammar we'd expect to get is the following:
[>@: IP =VP =DP]s, [>@: DP =DP -o]s,
[>John: DP]s, [>Paul: DP]s, [>George: DP]s, [>Ringo: DP]s,
[>saw: VP =DP +o]s

Another possible grammar that parses this input is one where each "beatle" is both a DP and a head-initial transitive verb and "saw" is a DP:
[>@: IP =VP =DP]s,
[>John: DP]s, [>John: VP =DP]s,
[>Paul: DP]s, [>Paul: VP =DP]s,
[>George: DP]s, [>George: VP =DP]s,
[>Ringo: DP]s, [>Ringo: VP =DP]s,
[>saw: DP]s]

Now, if we calculate the energy for each grammar when parsing the input:

Expected grammar:
Grammar encoding: 94
Input encoding: 184
Total energy: 278

"Wrong" grammar:
Grammar encoding: 123
Input encoding: 128
Total energy: 251

We can see that the "wrong" grammar has a larger grammar encoding but also the smallest energy because every sentence has a shorter derivation (without movement and without merging with the DP “wrapper”).
At this point the “wrong” grammar already has less energy. If we enlarge the input and add four more of the same pattern of sentences we'll get:

Expected grammar:
Grammar encoding: 94
Input encoding: 368
Total energy: 462

"Wrong" grammar:
Grammar encoding: 123
Input encoding: 256
Total energy: 379

We can see that the larger the input, the larger the energy difference grows.
Because of this difference in energy, the learner prefers the “wrong” grammar over the expected grammar and we get results accordingly.
I discuss some possible ways to overcome this problem in the next sections.

7. Trying to Improve Results with Coordination

For the learner to come to the right conclusions and learn the expected grammars, it needs more information and probably more constraints. A child acquiring a language is exposed to more input, a larger vocabulary and more complex syntactical structures than our learner.

As we've seen in the previous section, adding more input sentences of the same format won't help the learner find the expected grammar. We can also guess that expanding the vocabulary within the existing categories won't be much help.
What we can do is add more complex syntactic structures. Different syntactic structures use the same words in different contexts, thus giving more 'clues' for the learner to assign the correct categories.

The structure I chose to implement is a coordination structure. Since both sides of the coordination structure must have the same grammatical category, this should help the learner assign categories since once it knows the category of some item, it must assign the same category to items that appear with it in coordination structures.

I added the word "and" to the vocabulary along with the category "[>*:DP =DP =DP]s". Adding another category and more words to each sentence of the input puts a huge strain on the learner in terms of running time. Therefore, I only added coordination between DPs for now.
Adding coordination between VPs won't help the learner reach the expected grammar since the problem lies within the VP itself – that is, the problem is deciding which word in the VP is the head (V) and which is the complement (DP). Similarly for the PPs and CPs.
Therefore, I only allowed coordination of DPs.

However, when running the learner, the expected grammar was still not learned.
For instance, for the following head-final sentence:
"Jerry Elaine and Kramer saw"

The learner parsed the sentence with the following items:

- [>@: IP =VP =DP]s
- [>Jerry: DP]s
- [>Elaine: VP =DP]s
- [>and: DP]s
- [>Kramer: DP =DP =DP]s
- [>saw: DP]s

I.e., “Elaine” is parsed as a head-initial verb; “and” & “saw” are parsed as nouns; “Kramer” is parsed as a conjunction.

The final grammar learned by the learner contains similar items corresponding to other words and input sentences.

The run logs for the learner can be found in:
https://bitbucket.org/taucompling/HeadComplementSyntacticLearner/src
under the directory “Learner Logs/With Coordination”.

The expected grammar isn’t learned since, as in the previous section, the “wrong” analysis still has less energy than the expected grammar. Using our metric, having several possible categories per word is still more “efficient” than adding movement.

Therefore, for the learner to be able to learn the expected grammar we need to add even more constraints. I discuss this further in the next section.

8. Discussion

There are several other methods we could try to use to improve the learning process and have the learner arrive at the expected grammar.
As mentioned in the previous section, something that should help is adding more constraints to the grammar in order to limit the number of possibilities.

These constraints can come in several different ways. The most obvious constraint for our problem is limiting the number of possible categories each word can have. If we say that a word can only have one grammatical category then if we have the following head-final input:

1. Jerry ran
2. George Jerry saw

Then the first sentence forces "Jerry" to be parsed as a "DP" and so the only way to parse the second sentence would be with the expected movement. I tested this constraint (by using a different neighbour function called delete_impossible_category) and the learner indeed succeeds in learning the expected grammar. Logs for the run can be found in:
https://bitbucket.org/taucompling/HeadComplementSyntacticLearner/src
under the directory "Learner Logs/Remove Impossible Category".

However, this constraint is too strict. The word "saw" itself is both a noun and a verb in English. Since this constraint doesn't align with how natural languages behave, I chose not to use it.

Another possible constraint can come from making assumptions about the order in which a child learns different words. If we assume that some words are learned before others then those words will guide the learner in assigning categories for newer words. For instance, I tested the coordination structures from the previous section while initializing the learner such that the word "and" was already assigned the category of a conjunction. That is, the initial hypothesis contained the item [and: DP =DP =DP]s while the blank grammar did not allow any other word to get the same category. This

indeed forced the learner to learn the expected grammar. Logs for the run can be found in:
https://bitbucket.org/taucompling/HeadComplementSyntacticLearner/src
under the directory “With Coordination and "and" Given”.

However, again, this doesn’t necessarily align with natural languages. A continuation of this work could be a learner that works in several steps. Initially on very simple inputs and then on progressively more complex inputs. At each step the learner arrives at some grammar and then for the next step we could assign some “cost” to changing a category (or adding a category) of a word that was already learned in the previous step. This should be done with corpora of natural languages in mind to make sure that the various steps align with the sentences a child hears when acquiring a language.

Another thing we could try is using even more syntactic structures in addition to coordination.
Some structures that seem like good candidates for determining the grammatical category of words are topicalization & questions.
In topicalization structures, a constituent is moved to the beginning of the sentence. If the learner had such sentences as input then it would be forced to analyze that sequence of words as a constituent and this will limit the number of categories a certain word can have. For instance, in a sentence like “This boy, John saw”. If we assume the structure added to the learner is the topicalization of the object then the learner would have no choice but to parse “saw” as a verb.
Question structures are similar, in a way. A constituent is replaced by a wh-word and moved to the beginning of the sentence. This, again, marks a string of words as a constituent. Furthermore, the wh-word itself is indicative – “who” could only replace a DP.
However, adding these things requires having a larger and more varied input and a more complex blank grammar which will directly affect the running time.

The last thing that could be used is frequency analysis. For instance, at earlier stages of the development of the learner, determiners and nouns were part of the grammar and the input. However, the order between them proved to be unlearnable. When the learner sees constituents like “the boy” there is no way for it to decide whether “the” is a head-

initial determiner and “boy” is a noun or if “boy” is a head-final determiner and “the” is the noun. There are no external clues to discern between the options.
In real life language acquisition, however, there are some ways by which this decision can be made. The most obvious of which is frequency analysis. There are usually significantly more nouns than determiners and this can help the child decide.
Unfortunately, due to running time constraints, this is not really possible with the learner described here. Having enough different words would make the initial hypothesis enormous and would require a very large input.

The use of frequency analysis could also help with our examples in this work. In the “wrong” grammar from the previous section we saw that all the noun words are interpreted as both DPs and VPs and that the verbs are interpreted as DPs. While it is likely that some word could be both a DP and a VP it is highly unlikely that half of the nouns in the language are also verbs. Similarly, some noun might never appear in a subject position but it is unlikely that half the nouns never appear as subject.
Such statistical reasoning could be used by the learner to exclude (or add cost) to “improbable” grammars and thus lead it towards the right grammar.

When such things are added and all learners arrive at the expected grammar then comparisons could be made between the different theories.

References

Aslin, R. N., J. R. Saffran, and E. L. Newport (1998). “Computation of conditional probability statistics by 8-month old infants”. In: Psychological Science 9, pp. 321–324.

Chomsky, N. (1981). "Lectures on Government and Binding". Dordrecht: Foris.

Cinque, G. (2016) “A Microparametric Approach to the Head-Initial/Head-Final Parameter”. To appear in Linguistic Analysis.

Greenberg, J. (1963) “Some Universals of Grammar with Particular Reference to the Order of Meaningful Elements”. In Greenberg, Joseph H. (ed.), Universals of Human Language, 73-113. Cambridge, Mass: MIT Press.

Harkema, H. (2001). "Parsing minimalist languages." Unpublished doctoral dissertation, Los Angeles: University of California.

Joshi, A. K. (1985) “Tree Adjoining Grammars: How Much Context-Sensitivity Is Required to Provide Reasonable Structural Descriptions?”. In David R. Dowty, Lauri Karttunen, and Arnold M. Zwicky, editors, Natural Language Parsing, pages 206–250. Cambridge University Press.

Katzir, R. (2014) “A Cognitively Plausible Model for Grammar Induction”. Journal of Language Modelling, 2(2):213–248.

Mahootian, S. (1997) “Persian (Descriptive Grammars)”. London: Routledge.

Rissanen, J. (1978). "Modeling by shortest data description". Automatica. **14** (5): 465–658

Shieber, S. M. (1985). "Evidence against the context-freeness of

natural language". Linguistics and Philosophy, 8:333–343. Reprinted

Stabler, E. P. (1997). “Derivational Minimalism.” In C. Retor´ e, editor, Logical Aspects of Computational Linguistics, volume 1328 of Lecture Notes in Artificial Intelligence. Springer Verlag, Berlin, Heidelberg, Germany.

Van Riemsdijk, H. (2007). "Case in spatial adpositional phrases: The dative accusative alternation in German". Pitar Mos: A building with a view. Festschrift for Alexandra Cornilescu: 1–23.

Appendix A – Detailed Results

Below is a sampling of the runs of each learner on all possible configurations of input. Again, the log files can be found in:

https://bitbucket.org/taucompling/HeadComplementSyntacticLearner/src

under the directory “Learner Logs/General Case”.

Kayne Learner

Head Initial Input

The relevant log file for this run is:

log_2017_05_10__02_40_42 - Kayne, Head-initial, PP & CP.txt

The generated input:

```
Input is: ['Elaine wrote with Jerry', 'George hated Kramer', 'Jerry says that Jerry
wrote', 'George wrote under Elaine', 'Elaine thinks that George hated Kramer', 'George
hated Elaine', 'George walked', 'Kramer loved Elaine', 'George assumes that Kramer
wrote', 'Kramer read above Kramer', 'Elaine loved Kramer', 'Jerry walked', 'Jerry
assumes that Kramer hated Kramer', 'George liked Elaine', 'Elaine says that Kramer
walked', 'Jerry loved George under Elaine', 'Elaine says that Kramer saw George',
'Jerry ran by George', 'Elaine knows that Elaine read', 'Jerry hated Kramer', 'Jerry
loved Kramer', 'Jerry saw Elaine by Kramer', 'Jerry loved Kramer', 'Kramer hated
George with Jerry', 'Kramer ran under Elaine', 'Kramer loved George by Elaine',
'George thinks that Elaine walked under Elaine', 'Jerry loved Elaine', 'Kramer walked
above Jerry', 'Jerry says that Elaine walked by Elaine', 'Kramer knows that Jerry ran
under Elaine', 'Elaine ran by Jerry', 'Kramer hated Kramer', 'Elaine saw Elaine with
Elaine', 'Elaine loved Jerry above George', 'George saw Elaine', 'George liked
Kramer', 'Kramer wrote', 'George hated Elaine under George', 'Elaine liked George',
'Kramer assumes that George loved Elaine with Kramer', 'George ran by Elaine', 'Kramer
saw Kramer above Elaine', 'Jerry ran by Elaine', 'Jerry read under George', 'George
liked Elaine by Kramer', 'George wrote', 'George ran', 'Elaine knows that Jerry walked
by Elaine', 'Kramer liked Kramer']
```

The initial hypothesis:

```
Initial hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP
=IP -Comp]s, [>Jerry: VP]s, [>Jerry: DP]s, [>Jerry: VP =DP]s, [>Jerry: CP =IP]s,
[>Jerry: VP =CP]s, [>Jerry: PP =DP]s, [>Jerry: VP =DP +O]s, [>Jerry: PP =DP +O]s,
[>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s, [>George: VP]s, [>George: DP]s,
[>George: VP =DP]s, [>George: CP =IP]s, [>George: VP =CP]s, [>George: PP =DP]s,
[>George: VP =DP +O]s, [>George: PP =DP +O]s, [>George: VP =CP +Oc]s, [>George: CP =IP
+Comp -Oc]s, [>Elaine: VP]s, [>Elaine: DP]s, [>Elaine: CP =IP]s, [>Elaine: VP =DP]s,
[>Elaine: VP =CP]s, [>Elaine: PP =DP]s, [>Elaine: VP =DP +O]s, [>Elaine: PP =DP +O]s,
[>Elaine: VP =CP +Oc]s, [>Elaine: CP =IP +Comp -Oc]s, [>Kramer: VP]s, [>Kramer: DP]s,
[>Kramer: CP =IP]s, [>Kramer: VP =DP]s, [>Kramer: VP =CP]s, [>Kramer: PP =DP]s,
[>Kramer: VP =DP +O]s, [>Kramer: PP =DP +O]s, [>Kramer: VP =CP +Oc]s, [>Kramer: CP =IP
+Comp -Oc]s, [>ran: VP]s, [>ran: DP]s, [>ran: CP =IP]s, [>ran: VP =DP]s, [>ran: VP
=CP]s, [>ran: PP =DP]s, [>ran: VP =DP +O]s, [>ran: PP =DP +O]s, [>ran: VP =CP +Oc]s,
[>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [>walked: DP]s, [>walked: CP =IP]s,
[>walked: VP =DP]s, [>walked: VP =CP]s, [>walked: PP =DP]s, [>walked: VP =DP +O]s,
[>walked: PP =DP +O]s, [>walked: VP =CP +Oc]s, [>walked: CP =IP +Comp -Oc]s, [>read:
VP]s, [>read: DP]s, [>read: CP =IP]s, [>read: VP =DP]s, [>read: VP =CP]s, [>read: PP
=DP]s, [>read: VP =DP +O]s, [>read: PP =DP +O]s, [>read: VP =CP +Oc]s, [>read: CP =IP
```

```
+Comp -Oc]s, [>wrote: VP]s, [>wrote: DP]s, [>wrote: CP =IP]s, [>wrote: VP =DP]s,
[>wrote: VP =CP]s, [>wrote: PP =DP]s, [>wrote: VP =DP +O]s, [>wrote: PP =DP +O]s,
[>wrote: VP =CP +Oc]s, [>wrote: CP =IP +Comp -Oc]s, [>liked: VP]s, [>liked: DP]s,
[>liked: CP =IP]s, [>liked: VP =DP]s, [>liked: VP =CP]s, [>liked: PP =DP]s, [>liked:
VP =DP +O]s, [>liked: PP =DP +O]s, [>liked: VP =CP +Oc]s, [>liked: CP =IP +Comp -Oc]s,
[>saw: VP]s, [>saw: DP]s, [>saw: CP =IP]s, [>saw: VP =DP]s, [>saw: VP =CP]s, [>saw: PP
=DP]s, [>saw: VP =DP +O]s, [>saw: PP =DP +O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP
+Comp -Oc]s, [>loved: VP]s, [>loved: DP]s, [>loved: CP =IP]s, [>loved: VP =DP]s,
[>loved: VP =CP]s, [>loved: PP =DP]s, [>loved: VP =DP +O]s, [>loved: PP =DP +O]s,
[>loved: VP =CP +Oc]s, [>loved: CP =IP +Comp -Oc]s, [>hated: VP]s, [>hated: DP]s,
[>hated: CP =IP]s, [>hated: VP =DP]s, [>hated: VP =CP]s, [>hated: PP =DP]s, [>hated:
VP =DP +O]s, [>hated: PP =DP +O]s, [>hated: VP =CP +Oc]s, [>hated: CP =IP +Comp -Oc]s,
[>with: VP]s, [>with: DP]s, [>with: CP =IP]s, [>with: VP =DP]s, [>with: VP =CP]s,
[>with: PP =DP]s, [>with: VP =DP +O]s, [>with: PP =DP +O]s, [>with: VP =CP +Oc]s,
[>with: CP =IP +Comp -Oc]s, [>by: VP]s, [>by: DP]s, [>by: CP =IP]s, [>by: VP =DP]s,
[>by: VP =CP]s, [>by: PP =DP]s, [>by: VP =DP +O]s, [>by: PP =DP +O]s, [>by: VP =CP
+Oc]s, [>by: CP =IP +Comp -Oc]s, [>above: VP]s, [>above: DP]s, [>above: CP =IP]s,
[>above: VP =DP]s, [>above: VP =CP]s, [>above: PP =DP]s, [>above: VP =DP +O]s,
[>above: PP =DP +O]s, [>above: VP =CP +Oc]s, [>above: CP =IP +Comp -Oc]s, [>under:
VP]s, [>under: DP]s, [>under: CP =IP]s, [>under: VP =DP]s, [>under: VP =CP]s, [>under:
PP =DP]s, [>under: VP =DP +O]s, [>under: PP =DP +O]s, [>under: VP =CP +Oc]s, [>under:
CP =IP +Comp -Oc]s, [>that: VP]s, [>that: DP]s, [>that: CP =IP]s, [>that: VP =DP]s,
[>that: VP =CP]s, [>that: PP =DP]s, [>that: VP =DP +O]s, [>that: PP =DP +O]s, [>that:
VP =CP +Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP]s, [>knows: DP]s, [>knows: CP
=IP]s, [>knows: VP =DP]s, [>knows: VP =CP]s, [>knows: PP =DP]s, [>knows: VP =DP +O]s,
[>knows: PP =DP +O]s, [>knows: VP =CP +Oc]s, [>knows: CP =IP +Comp -Oc]s, [>says:
VP]s, [>says: DP]s, [>says: CP =IP]s, [>says: VP =DP]s, [>says: VP =CP]s, [>says: PP
=DP]s, [>says: VP =DP +O]s, [>says: PP =DP +O]s, [>says: VP =CP +Oc]s, [>says: CP =IP
+Comp -Oc]s, [>thinks: VP]s, [>thinks: DP]s, [>thinks: CP =IP]s, [>thinks: VP =DP]s,
[>thinks: VP =CP]s, [>thinks: PP =DP]s, [>thinks: VP =DP +O]s, [>thinks: PP =DP +O]s,
[>thinks: VP =CP +Oc]s, [>thinks: CP =IP +Comp -Oc]s, [>assumes: VP]s, [>assumes:
DP]s, [>assumes: CP =IP]s, [>assumes: VP =DP]s, [>assumes: VP =CP]s, [>assumes: PP
=DP]s, [>assumes: VP =DP +O]s, [>assumes: PP =DP +O]s, [>assumes: VP =CP +Oc]s,
[>assumes: CP =IP +Comp -Oc]s]
```

Initial energy:

```
Temperature: 100.000000
Energy: 10192
Grammar: 5143
Input: 5049
```

After 100 iterations:

```
Iteration: 100
2017_05_10__02_54_38: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: VP]s, [>Jerry: DP]s, [>Jerry: CP =IP]s, [>Jerry:
VP =CP]s, [>Jerry: VP =DP +O]s, [>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s,
[>George: VP]s, [>George: DP]s, [>George: CP =IP]s, [>George: VP =CP]s, [>George: PP
=DP]s, [>George: VP =DP +O]s, [>George: VP =CP +Oc]s, [>Elaine: DP]s, [>Elaine: VP]s,
[>Elaine: CP =IP]s, [>Elaine: VP =DP]s, [>Elaine: VP =CP]s, [>Elaine: VP =DP +O]s,
[>Kramer: VP]s, [>Kramer: DP]s, [>Kramer: VP =DP]s, [>Kramer: CP =IP]s, [>Kramer: VP
=DP +O]s, [>Kramer: PP =DP +O]s, [>ran: VP]s, [>ran: VP =DP]s, [>ran: VP =CP]s, [>ran:
PP =DP]s, [>ran: VP =DP +O]s, [>ran: VP =CP +Oc]s, [>ran: CP =IP +Comp -Oc]s,
[>walked: VP]s, [>walked: CP =IP]s, [>walked: VP =DP]s, [>walked: VP =CP]s, [>walked:
PP =DP]s, [>walked: VP =DP +O]s, [>walked: VP =CP +Oc]s, [>read: VP]s, [>read: DP]s,
[>read: VP =DP]s, [>read: PP =DP]s, [>read: VP =DP +O]s, [>read: PP =DP +O]s, [>read:
VP =CP +Oc]s, [>wrote: VP]s, [>wrote: VP =DP]s, [>wrote: VP =CP]s, [>wrote: PP =DP]s,
[>wrote: VP =DP +O]s, [>wrote: PP =DP +O]s, [>wrote: VP =CP +Oc]s, [>liked: VP]s,
[>liked: CP =IP]s, [>liked: VP =DP]s, [>liked: VP =CP]s, [>liked: PP =DP]s, [>liked:
PP =DP +O]s, [>liked: VP =CP +Oc]s, [>saw: VP =DP]s, [>saw: VP =DP +O]s, [>saw: PP =DP
+O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP +Comp -Oc]s, [>loved: VP =DP]s, [>loved: PP
=DP]s, [>loved: VP =DP +O]s, [>hated: VP]s, [>hated: CP =IP]s, [>hated: VP =DP]s,
[>hated: PP =DP]s, [>hated: VP =DP +O]s, [>hated: PP =DP +O]s, [>hated: VP =CP +Oc]s,
[>hated: CP =IP +Comp -Oc]s, [>with: VP]s, [>with: DP]s, [>with: VP =DP]s, [>with: VP
=CP]s, [>with: PP =DP]s, [>with: VP =DP +O]s, [>with: PP =DP +O]s, [>by: VP]s, [>by:
CP =IP]s, [>by: VP =CP]s, [>by: PP =DP]s, [>by: VP =DP +O]s, [>by: VP =CP +Oc]s, [>by:
CP =IP +Comp -Oc]s, [>above: VP]s, [>above: DP]s, [>above: CP =IP]s, [>above: VP
=DP]s, [>above: VP =CP]s, [>above: PP =DP]s, [>above: PP =DP +O]s, [>above: VP =CP
+Oc]s, [>above: CP =IP +Comp -Oc]s, [>under: DP]s, [>under: VP =DP]s, [>under: PP
=DP]s, [>under: VP =CP]s, [>under: VP =DP +O]s, [>under: PP =DP +O]s, [>under: VP =CP
+Oc]s, [>under: CP =IP +Comp -Oc]s, [>that: VP]s, [>that: CP =IP]s, [>that: VP =CP]s,
[>that: PP =DP +O]s, [>that: VP =DP +O]s, [>that: CP =IP +Comp -Oc]s, [>knows: DP]s,
```

```
[>knows: CP =IP]s, [>knows: VP =DP]s, [>knows: VP =CP]s, [>knows: PP =DP]s, [>says:
VP]s, [>says: CP =IP]s, [>says: VP =CP]s, [>says: VP =DP +O]s, [>says: PP =DP +O]s,
[>says: VP =CP +Oc]s, [>says: CP =IP +Comp -Oc]s, [>thinks: VP =CP]s, [>thinks: PP
=DP]s, [>thinks: PP =DP +O]s, [>thinks: VP =CP +Oc]s, [>thinks: CP =IP +Comp -Oc]s,
[>assumes: VP]s, [>assumes: DP]s, [>assumes: CP =IP]s, [>assumes: VP =DP]s, [>assumes:
VP =CP]s, [>assumes: VP =CP +Oc]s, [>assumes: CP =IP +Comp -Oc]s]
2017_05_10__02_54_38: Energy: 8456
2017_05_10__02_54_38: Grammar: 3407
2017_05_10__02_54_38: Input: 5049
2017_05_10__02_54_38: Temperature: 74.048426
```

After 300 iterations:

```
Iteration: 300
2017_05_10__02_55_05: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: DP]s, [>Jerry: VP =CP]s, [>Jerry: VP =DP +O]s,
[>George: DP]s, [>George: VP =DP]s, [>George: VP =CP +Oc]s, [>Elaine: DP]s, [>Elaine:
PP =DP +O]s, [>Kramer: DP]s, [>Kramer: CP =IP]s, [>ran: VP]s, [>ran: CP =IP +Comp -
Oc]s, [>walked: VP]s, [>walked: VP =DP +O]s, [>walked: PP =DP +O]s, [>read: VP]s,
[>wrote: VP]s, [>wrote: VP =CP +Oc]s, [>liked: CP =IP]s, [>liked: VP =DP]s, [>liked:
PP =DP]s, [>saw: VP =DP]s, [>saw: VP =CP]s, [>saw: CP =IP +Comp -Oc]s, [>loved: VP
=DP]s, [>hated: VP =DP]s, [>hated: VP =CP]s, [>with: PP =DP]s, [>with: PP =DP +O]s,
[>by: VP]s, [>by: PP =DP]s, [>above: DP]s, [>above: PP =DP]s, [>under: PP =DP]s,
[>under: VP =DP]s, [>under: PP =DP +O]s, [>under: CP =IP +Comp -Oc]s, [>that: CP
=IP]s, [>that: PP =DP +O]s, [>knows: VP =CP]s, [>says: DP]s, [>says: VP]s, [>says: VP
=CP]s, [>says: VP =CP +Oc]s, [>thinks: VP =CP]s, [>assumes: DP]s, [>assumes: VP =CP]s,
[>assumes: CP =IP]s]
2017_05_10__02_55_05: Energy: 5067
2017_05_10__02_55_05: Grammar: 1206
2017_05_10__02_55_05: Input: 3861
2017_05_10__02_55_05: Temperature: 40.602006
```

After over 400 iterations:

```
Iteration: 442
2017_05_10__02_55_16: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>George: DP]s, [>Elaine: DP]s, [>Kramer: DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s, [>loved: VP =DP]s, [>hated: VP =DP]s,
[>with: PP =DP]s, [>by: PP =DP]s, [>above: PP =DP]s, [>under: PP =DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>thinks: VP =CP]s, [>assumes: VP =CP]s]
2017 05 10  02 55 16: Energy: 3912
2017_05_10__02_55_16: Grammar: 543
2017_05_10__02_55_16: Input: 3369
2017_05_10__02_55_16: Temperature: 26.500860
```

Head Final Input

The relevant log file for this run is:

log_2017_05_10__02_57_53 - Kayne, Head-final, PP & CP.txt

The generated input:

```
Input is: ['Jerry wrote Elaine above', 'Elaine read', 'Jerry Jerry liked George
under', 'Jerry read', 'Elaine read', 'George Jerry saw', 'Elaine Elaine read Elaine
with that knows', 'George Elaine loved', 'Jerry ran', 'Elaine wrote George under',
'George walked', 'Jerry wrote George by', 'Kramer Kramer saw', 'George read', 'George
George Jerry hated that says', 'Elaine Elaine liked', 'George Kramer liked Kramer
under', 'Jerry walked Jerry above', 'Jerry Jerry saw', 'Jerry Elaine Elaine liked that
knows Kramer under', 'Jerry read Jerry by', 'George walked George under', 'Jerry ran',
'Jerry read Jerry by', 'George ran', 'Elaine Elaine loved Elaine under', 'Kramer
George walked that thinks Elaine with', 'Jerry George Kramer liked that thinks George
by', 'Kramer Kramer hated Elaine under', 'Elaine Kramer loved', 'Elaine Jerry loved
```

```
Jerry above', 'Kramer read', 'Kramer walked George under', 'Elaine Kramer wrote George
above that knows', 'Jerry Kramer Kramer loved that knows', 'Elaine Elaine ran that
knows', 'Elaine Kramer liked', 'Elaine Kramer liked', 'George Elaine loved Jerry
above', 'Kramer Kramer liked Jerry above', 'George Jerry saw Elaine by', 'Jerry Jerry
wrote that knows', 'Jerry walked', 'Kramer Elaine loved Elaine under', 'Kramer Kramer
hated', 'George Kramer loved Jerry by', 'George Elaine loved Elaine above', 'Jerry
ran', 'George walked Jerry above', 'Elaine Jerry hated Kramer above']
```

The initial hypothesis:

```
Initial hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP
=IP -Comp]s, [>Jerry: VP]s, [>Jerry: DP]s, [>Jerry: PP =DP]s, [>Jerry: VP =DP]s,
[>Jerry: VP =CP]s, [>Jerry: CP =IP]s, [>Jerry: VP =DP +O]s, [>Jerry: PP =DP +O]s,
[>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s, [>George: VP]s, [>George: DP]s,
[>George: PP =DP]s, [>George: CP =IP]s, [>George: VP =DP]s, [>George: VP =CP]s,
[>George: VP =DP +O]s, [>George: PP =DP +O]s, [>George: VP =CP +Oc]s, [>George: CP =IP
+Comp -Oc]s, [>Elaine: VP]s, [>Elaine: DP]s, [>Elaine: PP =DP]s, [>Elaine: CP =IP]s,
[>Elaine: VP =DP]s, [>Elaine: VP =CP]s, [>Elaine: VP =DP +O]s, [>Elaine: PP =DP +O]s,
[>Elaine: VP =CP +Oc]s, [>Elaine: CP =IP +Comp -Oc]s, [>Kramer: VP]s, [>Kramer: DP]s,
[>Kramer: VP =CP]s, [>Kramer: PP =DP]s, [>Kramer: CP =IP]s, [>Kramer: VP =DP]s,
[>Kramer: VP =DP +O]s, [>Kramer: PP =DP +O]s, [>Kramer: VP =CP +Oc]s, [>Kramer: CP =IP
+Comp -Oc]s, [>ran: VP]s, [>ran: DP]s, [>ran: VP =CP]s, [>ran: PP =DP]s, [>ran: CP
=IP]s, [>ran: VP =DP]s, [>ran: VP =DP +O]s, [>ran: PP =DP +O]s, [>ran: VP =CP +Oc]s,
[>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [>walked: DP]s, [>walked: VP =CP]s,
[>walked: PP =DP]s, [>walked: CP =IP]s, [>walked: VP =DP]s, [>walked: VP =DP +O]s,
[>walked: PP =DP +O]s, [>walked: VP =CP +Oc]s, [>walked: CP =IP +Comp -Oc]s, [>read:
VP]s, [>read: DP]s, [>read: PP =DP]s, [>read: CP =IP]s, [>read: VP =DP]s, [>read: VP
=CP]s, [>read: VP =DP +O]s, [>read: PP =DP +O]s, [>read: VP =CP +Oc]s, [>read: CP =IP
+Comp -Oc]s, [>wrote: VP]s, [>wrote: DP]s, [>wrote: PP =DP]s, [>wrote: VP =DP]s,
[>wrote: VP =CP]s, [>wrote: CP =IP]s, [>wrote: VP =DP +O]s, [>wrote: PP =DP +O]s,
[>wrote: VP =CP +Oc]s, [>wrote: CP =IP +Comp -Oc]s, [>liked: VP]s, [>liked: DP]s,
[>liked: PP =DP]s, [>liked: CP =IP]s, [>liked: VP =DP]s, [>liked: VP =CP]s, [>liked:
VP =DP +O]s, [>liked: PP =DP +O]s, [>liked: VP =CP +Oc]s, [>liked: CP =IP +Comp -Oc]s,
[>saw: VP]s, [>saw: DP]s, [>saw: PP =DP]s, [>saw: CP =IP]s, [>saw: VP =DP]s, [>saw: VP
=CP]s, [>saw: VP =DP +O]s, [>saw: PP =DP +O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP
+Comp -Oc]s, [>loved: VP]s, [>loved: DP]s, [>loved: VP =CP]s, [>loved: PP =DP]s,
[>loved: CP =IP]s, [>loved: VP =DP]s, [>loved: VP =DP +O]s, [>loved: PP =DP +O]s,
[>loved: VP =CP +Oc]s, [>loved: CP =IP +Comp -Oc]s, [>hated: VP]s, [>hated: DP]s,
[>hated: VP =CP]s, [>hated: PP =DP]s, [>hated: CP =IP]s, [>hated: VP =DP]s, [>hated:
PP =DP +O]s, [>hated: VP =DP +O]s, [>hated: VP =CP +Oc]s, [>hated: CP =IP +Comp -Oc]s,
[>with: VP]s, [>with: DP]s, [>with: PP =DP]s, [>with: CP =IP]s, [>with: VP =DP]s,
[>with: VP =CP]s, [>with: VP =DP +O]s, [>with: PP =DP +O]s, [>with: VP =CP +Oc]s,
[>with: CP =IP +Comp -Oc]s, [>by: VP]s, [>by: DP]s, [>by: VP =CP]s, [>by: PP =DP]s,
[>by: CP =IP]s, [>by: VP =DP]s, [>by: VP =DP +O]s, [>by: PP =DP +O]s, [>by: VP =CP
+Oc]s, [>by: CP =IP +Comp -Oc]s, [>above: VP]s, [>above: DP]s, [>above: PP =DP]s,
[>above: CP =IP]s, [>above: VP =DP]s, [>above: VP =CP]s, [>above: VP =DP +O]s,
[>above: PP =DP +O]s, [>above: VP =CP +Oc]s, [>above: CP =IP +Comp -Oc]s, [>under:
VP]s, [>under: DP]s, [>under: PP =DP]s, [>under: CP =IP]s, [>under: VP =DP]s, [>under:
VP =CP]s, [>under: VP =DP +O]s, [>under: PP =DP +O]s, [>under: VP =CP +Oc]s, [>under:
CP =IP +Comp -Oc]s, [>that: VP]s, [>that: DP]s, [>that: PP =DP]s, [>that: CP =IP]s,
[>that: VP =DP]s, [>that: VP =CP]s, [>that: VP =DP +O]s, [>that: PP =DP +O]s, [>that:
VP =CP +Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP]s, [>knows: DP]s, [>knows: PP
=DP]s, [>knows: CP =IP]s, [>knows: VP =DP]s, [>knows: VP =CP]s, [>knows: VP =DP +O]s,
[>knows: PP =DP +O]s, [>knows: VP =CP +Oc]s, [>knows: CP =IP +Comp -Oc]s, [>says:
VP]s, [>says: DP]s, [>says: VP =CP]s, [>says: PP =DP]s, [>says: CP =IP]s, [>says: VP
=DP]s, [>says: VP =DP +O]s, [>says: PP =DP +O]s, [>says: VP =CP +Oc]s, [>says: CP =IP
+Comp -Oc]s, [>thinks: VP]s, [>thinks: DP]s, [>thinks: VP =CP]s, [>thinks: PP =DP]s,
[>thinks: CP =IP]s, [>thinks: VP =DP]s, [>thinks: VP =DP +O]s, [>thinks: PP =DP +O]s,
[>thinks: VP =CP +Oc]s, [>thinks: CP =IP +Comp -Oc]s]
```

Initial energy:

```
Temperature: 100.000000
Energy: 9816
Grammar: 4903
Input: 4913
```

After 100 iterations:

```
Iteration: 100
2017_05_10__03_39_15: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: VP]s, [>Jerry: DP]s, [>Jerry: PP =DP]s, [>Jerry:
```

```
VP =CP]s, [>Jerry: CP =IP]s, [>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s,
[>George: VP]s, [>George: DP]s, [>George: PP =DP]s, [>George: VP =DP]s, [>George: PP
=DP +O]s, [>George: VP =CP +Oc]s, [>George: CP =IP +Comp -Oc]s, [>Elaine: DP]s,
[>Elaine: PP =DP]s, [>Elaine: CP =IP]s, [>Elaine: VP =DP]s, [>Elaine: VP =CP]s,
[>Elaine: PP =DP +O]s, [>Elaine: VP =DP +O]s, [>Elaine: VP =CP +Oc]s, [>Kramer: VP]s,
[>Kramer: DP]s, [>Kramer: VP =CP]s, [>Kramer: PP =DP]s, [>Kramer: VP =DP]s, [>Kramer:
PP =DP +O]s, [>Kramer: VP =DP +O]s, [>Kramer: VP =CP +Oc]s, [>Kramer: CP =IP +Comp -
Oc]s, [>ran: VP]s, [>ran: DP]s, [>ran: VP =CP]s, [>ran: PP =DP]s, [>ran: CP =IP]s,
[>ran: VP =DP +O]s, [>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [>walked: DP]s,
[>walked: PP =DP]s, [>walked: CP =IP]s, [>walked: VP =CP]s, [>walked: VP =DP +O]s,
[>walked: PP =DP +O]s, [>walked: CP =IP +Comp -Oc]s, [>read: VP]s, [>read: PP =DP]s,
[>read: VP =DP]s, [>read: VP =DP +O]s, [>read: VP =CP +Oc]s, [>read: CP =IP +Comp -
Oc]s, [>wrote: VP]s, [>wrote: DP]s, [>wrote: PP =DP]s, [>wrote: VP =CP]s, [>wrote: CP
=IP]s, [>liked: VP]s, [>liked: DP]s, [>liked: VP =DP]s, [>liked: PP =DP +O]s, [>liked:
CP =IP +Comp -Oc]s, [>saw: VP]s, [>saw: DP]s, [>saw: PP =DP]s, [>saw: CP =IP]s, [>saw:
VP =DP]s, [>saw: VP =CP]s, [>saw: VP =DP +O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP
+Comp -Oc]s, [>loved: VP]s, [>loved: DP]s, [>loved: PP =DP]s, [>loved: CP =IP]s,
[>loved: VP =DP]s, [>loved: VP =DP +O]s, [>loved: VP =CP +Oc]s, [>loved: CP =IP +Comp
-Oc]s, [>hated: DP]s, [>hated: VP =CP]s, [>hated: VP =DP]s, [>hated: VP =DP +O]s,
[>hated: PP =DP +O]s, [>hated: VP =CP +Oc]s, [>with: DP]s, [>with: PP =DP]s, [>with:
VP =CP]s, [>with: VP =DP +O]s, [>with: VP =CP +Oc]s, [>with: CP =IP +Comp -Oc]s, [>by:
VP]s, [>by: VP =CP]s, [>by: PP =DP +O]s, [>by: CP =IP +Comp -Oc]s, [>above: VP]s,
[>above: DP]s, [>above: PP =DP]s, [>above: VP =DP]s, [>above: VP =CP]s, [>above: CP
=IP]s, [>above: VP =DP +O]s, [>above: PP =DP +O]s, [>above: CP =IP +Comp -Oc]s,
[>under: DP]s, [>under: PP =DP]s, [>under: VP =DP]s, [>under: VP =CP]s, [>under: PP
=DP +O]s, [>under: VP =CP +Oc]s, [>that: VP]s, [>that: DP]s, [>that: PP =DP]s, [>that:
CP =IP]s, [>that: VP =CP]s, [>that: VP =DP +O]s, [>that: PP =DP +O]s, [>that: VP =CP
+Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP]s, [>knows: CP =IP]s, [>knows: VP
=DP]s, [>knows: PP =DP +O]s, [>knows: VP =CP +Oc]s, [>knows: CP =IP +Comp -Oc]s,
[>says: DP]s, [>says: VP =CP]s, [>says: PP =DP]s, [>says: CP =IP]s, [>says: VP =DP]s,
[>says: VP =DP +O]s, [>says: PP =DP +O]s, [>says: VP =CP +Oc]s, [>says: CP =IP +Comp -
Oc]s, [>thinks: DP]s, [>thinks: VP =CP]s, [>thinks: PP =DP]s, [>thinks: CP =IP]s,
[>thinks: VP =DP]s, [>thinks: VP =DP +O]s, [>thinks: VP =CP +Oc]s]
2017_05_10__03_39_15: Energy: 8754
2017_05_10__03_39_15: Grammar: 3467
2017_05_10__03_39_15: Input: 5287
2017_05_10__03_39_15: Temperature: 74.048426
```

After 300 iterations:

```
Iteration: 300
2017_05_10__03_46_25: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: VP]s, [>Jerry: DP]s, [>Jerry: PP =DP]s, [>Jerry:
VP =CP]s, [>Jerry: VP =DP]s, [>Jerry: VP =DP +O]s, [>George: DP]s, [>George: VP]s,
[>George: PP =DP]s, [>Elaine: DP]s, [>Elaine: PP =DP]s, [>Elaine: VP =DP]s, [>Kramer:
DP]s, [>Kramer: PP =DP]s, [>Kramer: VP =DP]s, [>Kramer: PP =DP +O]s, [>ran: VP]s,
[>ran: DP]s, [>ran: VP =CP]s, [>walked: VP]s, [>walked: VP =DP]s, [>read: VP]s,
[>read: VP =DP]s, [>wrote: VP]s, [>wrote: CP =IP +Comp -Oc]s, [>liked: DP]s, [>liked:
VP =DP]s, [>liked: CP =IP]s, [>saw: DP]s, [>loved: DP]s, [>loved: CP =IP]s, [>loved:
VP =DP]s, [>loved: VP =DP +O]s, [>hated: VP]s, [>hated: VP =DP +O]s, [>with: DP]s,
[>with: VP =DP]s, [>with: VP =DP +O]s, [>with: PP =DP +O]s, [>with: VP =CP +Oc]s,
[>by: PP =DP +O]s, [>above: VP]s, [>above: DP]s, [>under: DP]s, [>under: PP =DP]s,
[>that: VP =CP +Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows: DP]s, [>knows: VP =CP
+Oc]s, [>knows: CP =IP +Comp -Oc]s, [>says: DP]s, [>says: VP =CP +Oc]s, [>thinks: VP
=DP +O]s, [>thinks: VP =CP +Oc]s]
2017_05_10__03_46_25: Energy: 5835
2017_05_10__03_46_25: Grammar: 1316
2017_05_10__03_46_25: Input: 4519
2017_05_10__03_46_25: Temperature: 40.602006
```

After over 500 iterations:

```
Iteration: 599
2017_05_10__03_47_30: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>Jerry: PP =DP]s, [>Jerry: VP =DP]s,
[>George: DP]s, [>George: PP =DP]s,
[>Elaine: DP]s, [>Elaine: PP =DP]s, [>Elaine: VP =DP]s,
[>Kramer: DP]s, [>Kramer: PP =DP]s, [>Kramer: VP =DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: DP]s, [>saw: DP]s, [>loved: DP]s, [>hated: DP]s,
```

```
[>with: DP]s, [>by: DP]s, [>above: DP]s, [>under: DP]s,
[>that: CP =IP +Comp -Oc]s,
[>knows: VP =CP +Oc]s, [>says: VP =CP +Oc]s, [>thinks: VP =CP +Oc]s]
2017_05_10__03_47_30: Energy: 4664
2017_05_10__03_47_30: Grammar: 664
2017_05_10__03_47_30: Input: 4000
2017_05_10__03_47_30: Temperature: 16.534834
```

Mixed Category Input

The relevant log file for this run is:

log_2017_05_10__03_54_20 - Kayne, Mixed-category, PP & CP.txt

The generated input:

```
Input is: ['Kramer hated Elaine Jerry with', 'Kramer assumes that Elaine walked George
by', 'Kramer walked', 'Elaine hated Jerry', 'Jerry saw Elaine', 'Kramer saw Jerry
George with', 'Elaine hated Jerry Elaine above', 'Kramer knows that Jerry saw Elaine
Elaine by', 'Kramer hated Kramer Kramer by', 'Elaine wrote Kramer by', 'Elaine ran',
'Elaine liked Elaine Elaine with', 'Elaine liked Kramer', 'George assumes that Elaine
ran Jerry above', 'Elaine walked', 'Elaine saw Elaine George with', 'George ran',
'George liked George', 'Jerry thinks that George hated Jerry', 'Jerry wrote Jerry by',
'Kramer walked Jerry by', 'Jerry wrote', 'George says that George liked Jerry',
'George assumes that Jerry read Elaine with', 'Elaine saw Jerry', 'Elaine loved Jerry
Elaine by', 'Elaine knows that Elaine hated Kramer Jerry under', 'Jerry thinks that
Kramer wrote', 'George walked Elaine with', 'Jerry wrote Kramer under', 'George saw
Jerry', 'George saw George Elaine above', 'Jerry assumes that George liked Kramer
George with', 'Jerry thinks that Kramer loved George Kramer with', 'Jerry hated George
Elaine by', 'Elaine read George by', 'Jerry wrote', 'Kramer thinks that George wrote
George above', 'George hated George', 'George thinks that Kramer hated George Elaine
above', 'Jerry liked Elaine', 'Kramer saw Jerry George under', 'George saw George
Jerry above', 'Kramer walked Elaine under', 'Kramer saw Jerry Jerry under', 'George
thinks that George hated Jerry George by', 'Jerry hated Jerry Jerry with', 'Kramer
assumes that Elaine loved Kramer', 'Elaine wrote George by', 'Kramer saw Jerry']
```

The initial hypothesis:

```
Initial hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP
=IP -Comp]s, [>Jerry: VP]s, [>Jerry: DP]s, [>Jerry: CP =IP]s, [>Jerry: VP =CP]s,
[>Jerry: PP =DP]s, [>Jerry: VP =DP]s, [>Jerry: PP =DP +O]s, [>Jerry: VP =DP +O]s,
[>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s, [>George: VP]s, [>George: DP]s,
[>George: CP =IP]s, [>George: VP =CP]s, [>George: PP =DP]s, [>George: VP =DP]s,
[>George: PP =DP +O]s, [>George: VP =DP +O]s, [>George: VP =CP +Oc]s, [>George: CP =IP
+Comp -Oc]s, [>Elaine: VP]s, [>Elaine: DP]s, [>Elaine: CP =IP]s, [>Elaine: VP =CP]s,
[>Elaine: PP =DP]s, [>Elaine: VP =DP]s, [>Elaine: PP =DP +O]s, [>Elaine: VP =DP +O]s,
[>Elaine: VP =CP +Oc]s, [>Elaine: CP =IP +Comp -Oc]s, [>Kramer: VP]s, [>Kramer: DP]s,
[>Kramer: CP =IP]s, [>Kramer: VP =CP]s, [>Kramer: PP =DP]s, [>Kramer: VP =DP]s,
[>Kramer: PP =DP +O]s, [>Kramer: VP =DP +O]s, [>Kramer: VP =CP +Oc]s, [>Kramer: CP =IP
+Comp -Oc]s, [>ran: VP]s, [>ran: DP]s, [>ran: CP =IP]s, [>ran: VP =CP]s, [>ran: PP
=DP]s, [>ran: VP =DP]s, [>ran: PP =DP +O]s, [>ran: VP =DP +O]s, [>ran: VP =CP +Oc]s,
[>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [>walked: DP]s, [>walked: CP =IP]s,
[>walked: VP =CP]s, [>walked: PP =DP]s, [>walked: VP =DP]s, [>walked: PP =DP +O]s,
[>walked: VP =DP +O]s, [>walked: VP =CP +Oc]s, [>walked: CP =IP +Comp -Oc]s, [>read:
VP]s, [>read: DP]s, [>read: VP =CP]s, [>read: PP =DP]s, [>read: CP =IP]s, [>read: VP
=DP]s, [>read: PP =DP +O]s, [>read: VP =DP +O]s, [>read: VP =CP +Oc]s, [>read: CP =IP
+Comp -Oc]s, [>wrote: VP]s, [>wrote: DP]s, [>wrote: CP =IP]s, [>wrote: VP =CP]s,
[>wrote: PP =DP]s, [>wrote: VP =DP]s, [>wrote: PP =DP +O]s, [>wrote: VP =DP +O]s,
[>wrote: VP =CP +Oc]s, [>wrote: CP =IP +Comp -Oc]s, [>liked: VP]s, [>liked: DP]s,
[>liked: CP =IP]s, [>liked: VP =CP]s, [>liked: PP =DP]s, [>liked: VP =DP]s, [>liked:
PP =DP +O]s, [>liked: VP =DP +O]s, [>liked: VP =CP +Oc]s, [>liked: CP =IP +Comp -Oc]s,
[>saw: VP]s, [>saw: DP]s, [>saw: CP =IP]s, [>saw: VP =CP]s, [>saw: PP =DP]s, [>saw: VP
=DP]s, [>saw: PP =DP +O]s, [>saw: VP =DP +O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP
+Comp -Oc]s, [>loved: VP]s, [>loved: DP]s, [>loved: VP =DP]s, [>loved: VP =CP]s,
[>loved: PP =DP]s, [>loved: CP =IP]s, [>loved: VP =DP +O]s, [>loved: PP =DP +O]s,
[>loved: VP =CP +Oc]s, [>loved: CP =IP +Comp -Oc]s, [>hated: VP]s, [>hated: DP]s,
[>hated: CP =IP]s, [>hated: VP =CP]s, [>hated: PP =DP]s, [>hated: VP =DP]s, [>hated:
PP =DP +O]s, [>hated: VP =DP +O]s, [>hated: VP =CP +Oc]s, [>hated: CP =IP +Comp -Oc]s,
```

```
[>with: VP]s, [>with: DP]s, [>with: CP =IP]s, [>with: VP =CP]s, [>with: PP =DP]s,
[>with: VP =DP]s, [>with: PP =DP +O]s, [>with: VP =DP +O]s, [>with: VP =CP +Oc]s,
[>with: CP =IP +Comp -Oc]s, [>by: VP]s, [>by: DP]s, [>by: CP =IP]s, [>by: VP =CP]s,
[>by: PP =DP]s, [>by: VP =DP]s, [>by: PP =DP +O]s, [>by: VP =DP +O]s, [>by: VP =CP
+Oc]s, [>by: CP =IP +Comp -Oc]s, [>above: VP]s, [>above: DP]s, [>above: CP =IP]s,
[>above: VP =CP]s, [>above: PP =DP]s, [>above: VP =DP]s, [>above: PP =DP +O]s,
[>above: VP =DP +O]s, [>above: VP =CP +Oc]s, [>above: CP =IP +Comp -Oc]s, [>under:
VP]s, [>under: DP]s, [>under: VP =DP]s, [>under: VP =CP]s, [>under: PP =DP]s, [>under:
CP =IP]s, [>under: VP =DP +O]s, [>under: PP =DP +O]s, [>under: VP =CP +Oc]s, [>under:
CP =IP +Comp -Oc]s, [>that: VP]s, [>that: DP]s, [>that: CP =IP]s, [>that: VP =CP]s,
[>that: PP =DP]s, [>that: VP =DP]s, [>that: PP =DP +O]s, [>that: VP =DP +O]s, [>that:
VP =CP +Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP]s, [>knows: DP]s, [>knows: CP
=IP]s, [>knows: VP =CP]s, [>knows: PP =DP]s, [>knows: VP =DP]s, [>knows: PP =DP +O]s,
[>knows: VP =DP +O]s, [>knows: VP =CP +Oc]s, [>knows: CP =IP +Comp -Oc]s, [>says:
VP]s, [>says: DP]s, [>says: CP =IP]s, [>says: VP =CP]s, [>says: PP =DP]s, [>says: VP
=DP]s, [>says: PP =DP +O]s, [>says: VP =DP +O]s, [>says: VP =CP +Oc]s, [>says: CP =IP
+Comp -Oc]s, [>thinks: VP]s, [>thinks: DP]s, [>thinks: CP =IP]s, [>thinks: VP =CP]s,
[>thinks: PP =DP]s, [>thinks: VP =DP]s, [>thinks: PP =DP +O]s, [>thinks: VP =DP +O]s,
[>thinks: VP =CP +Oc]s, [>thinks: CP =IP +Comp -Oc]s, [>assumes: VP]s, [>assumes:
DP]s, [>assumes: CP =IP]s, [>assumes: VP =CP]s, [>assumes: PP =DP]s, [>assumes: VP
=DP]s, [>assumes: PP =DP +O]s, [>assumes: VP =DP +O]s, [>assumes: VP =CP +Oc]s,
[>assumes: CP =IP +Comp -Oc]s]
```

Initial energy:

```
Temperature: 100.000000
Energy: 10753
Grammar: 5143
Input: 5610
```

After 100 iterations:

```
Iteration: 100
2017_05_10__05_49_45: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: DP]s, [>Jerry: VP]s, [>Jerry: CP =IP]s, [>Jerry:
PP =DP]s, [>Jerry: VP =DP +O]s, [>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s,
[>George: DP]s, [>George: VP]s, [>George: CP =IP]s, [>George: VP =CP]s, [>George: PP
=DP]s, [>George: VP =DP]s, [>George: CP =IP +Comp -Oc]s, [>Elaine: VP]s, [>Elaine:
DP]s, [>Elaine: PP =DP]s, [>Elaine: VP =CP]s, [>Elaine: VP =DP +O]s, [>Kramer: DP]s,
[>Kramer: CP =IP]s, [>Kramer: PP =DP]s, [>Kramer: PP =DP +O]s, [>Kramer: VP =CP +Oc]s,
[>ran: VP]s, [>ran: DP]s, [>ran: CP =IP]s, [>ran: VP =CP]s, [>ran: PP =DP]s, [>ran: VP
=DP]s, [>ran: VP =DP +O]s, [>walked: VP]s, [>walked: DP]s, [>walked: CP =IP]s,
[>walked: VP =CP]s, [>walked: PP =DP]s, [>walked: VP =DP]s, [>walked: PP =DP +O]s,
[>walked: VP =CP +Oc]s, [>read: VP]s, [>read: DP]s, [>read: VP =CP]s, [>read: VP
=DP]s, [>read: VP =DP +O]s, [>read: VP =CP +Oc]s, [>wrote: VP]s, [>wrote: DP]s,
[>wrote: CP =IP]s, [>wrote: PP =DP]s, [>wrote: PP =DP +O]s, [>wrote: VP =CP +Oc]s,
[>wrote: CP =IP +Comp -Oc]s, [>liked: DP]s, [>liked: CP =IP]s, [>liked: VP =CP]s,
[>liked: PP =DP]s, [>liked: VP =DP]s, [>liked: VP =DP +O]s, [>liked: PP =DP +O]s,
[>liked: VP =CP +Oc]s, [>liked: CP =IP +Comp -Oc]s, [>saw: VP]s, [>saw: CP =IP]s,
[>saw: VP =CP]s, [>saw: PP =DP]s, [>saw: VP =DP]s, [>saw: PP =DP +O]s, [>saw: VP =DP
+O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP +Comp -Oc]s, [>loved: VP =DP]s, [>loved: VP
=CP]s, [>loved: CP =IP]s, [>loved: VP =DP +O]s, [>loved: PP =DP +O]s, [>loved: VP =CP
+Oc]s, [>loved: CP =IP +Comp -Oc]s, [>hated: VP]s, [>hated: CP =IP]s, [>hated: VP
=CP]s, [>hated: VP =DP]s, [>hated: PP =DP +O]s, [>hated: VP =DP +O]s, [>hated: VP =CP
+Oc]s, [>hated: CP =IP +Comp -Oc]s, [>with: DP]s, [>with: CP =IP]s, [>with: VP =CP]s,
[>with: PP =DP]s, [>with: VP =DP]s, [>with: VP =DP +O]s, [>by: VP]s, [>by: DP]s, [>by:
CP =IP]s, [>by: PP =DP]s, [>by: VP =DP]s, [>by: VP =CP]s, [>by: VP =DP +O]s, [>by: PP
=DP +O]s, [>by: VP =CP +Oc]s, [>by: CP =IP +Comp -Oc]s, [>above: DP]s, [>above: VP]s,
[>above: CP =IP]s, [>above: VP =DP]s, [>above: PP =DP]s, [>above: PP =DP +O]s,
[>above: VP =DP +O]s, [>above: VP =CP +Oc]s, [>above: CP =IP +Comp -Oc]s, [>under:
DP]s, [>under: VP =DP]s, [>under: PP =DP]s, [>under: CP =IP]s, [>under: VP =CP +Oc]s,
[>under: CP =IP +Comp -Oc]s, [>that: DP]s, [>that: CP =IP]s, [>that: VP =CP]s, [>that:
PP =DP]s, [>that: VP =DP]s, [>that: PP =DP +O]s, [>that: VP =DP +O]s, [>that: VP =CP
+Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP]s, [>knows: DP]s, [>knows: CP =IP]s,
[>knows: PP =DP]s, [>knows: VP =DP]s, [>knows: PP =DP +O]s, [>knows: VP =CP +Oc]s,
[>says: VP]s, [>says: VP =DP]s, [>says: VP =CP]s, [>says: VP =DP +O]s, [>says: VP =CP
+Oc]s, [>says: CP =IP +Comp -Oc]s, [>thinks: VP]s, [>thinks: DP]s, [>thinks: VP =CP]s,
[>thinks: PP =DP]s, [>thinks: VP =DP]s, [>thinks: VP =CP +Oc]s, [>thinks: CP =IP +Comp
-Oc]s, [>assumes: DP]s, [>assumes: PP =DP]s, [>assumes: VP =DP]s, [>assumes: CP =IP]s,
[>assumes: PP =DP +O]s, [>assumes: VP =DP +O]s, [>assumes: VP =CP +Oc]s, [>assumes: CP
=IP +Comp -Oc]s]
2017_05_10__05_49_45: Energy: 9483
2017_05_10__05_49_45: Grammar: 3751
```

```
2017_05_10__05_49_45: Input: 5732
2017_05_10__05_49_45: Temperature: 74.048426
```

After 300 iterations:

```
Iteration: 300
2017_05_10__06_02_45: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: DP]s, [>Jerry: PP =DP]s, [>George: DP]s,
[>George: VP]s, [>George: PP =DP]s, [>George: CP =IP]s, [>George: CP =IP +Comp -Oc]s,
[>Elaine: DP]s, [>Elaine: PP =DP]s, [>Elaine: VP =CP +Oc]s, [>Kramer: DP]s, [>Kramer:
PP =DP]s, [>Kramer: CP =IP]s, [>Kramer: PP =DP +O]s, [>Kramer: VP =CP +Oc]s, [>ran:
VP]s, [>ran: DP]s, [>ran: PP =DP]s, [>ran: VP =CP]s, [>ran: VP =DP +O]s, [>walked:
VP]s, [>walked: PP =DP +O]s, [>read: VP]s, [>read: DP]s, [>wrote: VP]s, [>wrote: CP
=IP]s, [>liked: PP =DP]s, [>liked: VP =DP]s, [>liked: VP =CP +Oc]s, [>liked: CP =IP
+Comp -Oc]s, [>saw: VP]s, [>saw: CP =IP]s, [>saw: PP =DP]s, [>saw: VP =DP]s, [>saw: VP
=CP +Oc]s, [>loved: VP =DP]s, [>loved: CP =IP +Comp -Oc]s, [>hated: VP =DP]s, [>hated:
PP =DP]s, [>hated: PP =DP +O]s, [>with: DP]s, [>with: PP =DP]s, [>by: DP]s, [>by:
VP]s, [>by: PP =DP +O]s, [>by: VP =CP +Oc]s, [>above: DP]s, [>under: DP]s, [>under:
VP]s, [>under: CP =IP +Comp -Oc]s, [>that: CP =IP]s, [>that: PP =DP]s, [>that: VP
=DP]s, [>that: VP =DP +O]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP]s, [>knows: CP
=IP]s, [>says: VP]s, [>says: VP =DP]s, [>says: PP =DP]s, [>thinks: VP =CP]s, [>thinks:
VP =DP]s, [>thinks: CP =IP]s, [>assumes: DP]s, [>assumes: PP =DP]s]
2017_05_10__06_02_45: Energy: 6726
2017_05_10__06_02_45: Grammar: 1577
2017_05_10__06_02_45: Input: 5149
2017_05_10__06_02_45: Temperature: 40.602006
```

After over 900 iterations:

```
Iteration: 912
2017_05_10__06_05_48: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>Jerry: PP =DP]s,
[>George: DP]s, [>George: PP =DP]s,
[>Elaine: DP]s, [>Elaine: PP =DP]s,
[>Kramer: DP]s, [>Kramer: PP =DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s, [>loved: VP =DP]s, [>hated: VP =DP]s,
[>with: DP]s, [>by: DP]s, [>above: DP]s, [>under: DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>thinks: VP =CP]s, [>assumes: VP =CP]s]
2017_05_10__06_05_48: Energy: 4797
2017_05_10__06_05_48: Grammar: 615
2017_05_10__06_05_48: Input: 4182
2017_05_10__06_05_48: Temperature: 6.456310
```

Mixed Word Input

The relevant log file for this run is:

log_2017_05_10__06_09_15 - Kayne, Mixed-word, PP & CP.txt

The generated input:

```
Input is: ['Jerry ran George by', 'Kramer Elaine loved', 'Elaine knows that Elaine
walked George with', 'Elaine George hated', 'Jerry says that Jerry Jerry hated George
above', 'Kramer wrote', 'Elaine Elaine hated', 'Elaine liked Elaine Kramer above',
'George assumes that George saw Jerry George under', 'George walked', 'Kramer read
Jerry under', 'Elaine wrote', 'George Jerry loved Elaine above', 'Kramer wrote',
'Elaine liked Kramer', 'Kramer Jerry loved', 'Jerry saw George Kramer by', 'George
liked Elaine Elaine under', 'Jerry ran Kramer with', 'George read', 'George walked
George by', 'Elaine thinks that George read', 'Jerry George loved Kramer above',
'Kramer walked Elaine above', 'Kramer wrote Kramer by', 'Jerry saw Elaine', 'George
read', 'Jerry liked George', 'George read Jerry above', 'Elaine saw Jerry', 'Elaine
Kramer loved Jerry under', 'Kramer liked George', 'Kramer ran Kramer above', 'Kramer
assumes that George read Elaine with', 'Jerry says that Elaine George loved Jerry
```

```
above', 'Kramer read Kramer under', 'Jerry walked Kramer with', 'Kramer assumes that
George George hated', 'George Jerry loved George by', 'Kramer walked Elaine by',
'Kramer ran', 'Jerry thinks that Jerry wrote Elaine with', 'Elaine assumes that Elaine
George hated Jerry with', 'George walked', 'Elaine Jerry loved', 'Jerry ran', 'Elaine
read', 'Jerry walked', 'Elaine read Elaine with', 'Kramer liked Elaine']
```

The initial hypothesis:

```
Initial hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP
=IP -Comp]s, [>Jerry: VP]s, [>Jerry: DP]s, [>Jerry: VP =CP]s, [>Jerry: PP =DP]s,
[>Jerry: CP =IP]s, [>Jerry: VP =DP]s, [>Jerry: VP =DP +O]s, [>Jerry: PP =DP +O]s,
[>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s, [>George: VP]s, [>George: DP]s,
[>George: VP =CP]s, [>George: PP =DP]s, [>George: CP =IP]s, [>George: VP =DP]s,
[>George: VP =DP +O]s, [>George: PP =DP +O]s, [>George: VP =CP +Oc]s, [>George: CP =IP
+Comp -Oc]s, [>Elaine: VP]s, [>Elaine: DP]s, [>Elaine: VP =DP]s, [>Elaine: VP =CP]s,
[>Elaine: PP =DP]s, [>Elaine: CP =IP]s, [>Elaine: VP =DP +O]s, [>Elaine: PP =DP +O]s,
[>Elaine: VP =CP +Oc]s, [>Elaine: CP =IP +Comp -Oc]s, [>Kramer: VP]s, [>Kramer: DP]s,
[>Kramer: VP =DP]s, [>Kramer: VP =CP]s, [>Kramer: PP =DP]s, [>Kramer: CP =IP]s,
[>Kramer: VP =DP +O]s, [>Kramer: PP =DP +O]s, [>Kramer: VP =CP +Oc]s, [>Kramer: CP =IP
+Comp -Oc]s, [>ran: VP]s, [>ran: DP]s, [>ran: VP =CP]s, [>ran: PP =DP]s, [>ran: CP
=IP]s, [>ran: VP =DP]s, [>ran: VP =DP +O]s, [>ran: PP =DP +O]s, [>ran: VP =CP +Oc]s,
[>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [>walked: DP]s, [>walked: VP =DP]s,
[>walked: VP =CP]s, [>walked: PP =DP]s, [>walked: CP =IP]s, [>walked: VP =DP +O]s,
[>walked: PP =DP +O]s, [>walked: VP =CP +Oc]s, [>walked: CP =IP +Comp -Oc]s, [>read:
VP]s, [>read: DP]s, [>read: VP =DP]s, [>read: VP =CP]s, [>read: PP =DP]s, [>read: CP
=IP]s, [>read: VP =DP +O]s, [>read: PP =DP +O]s, [>read: VP =CP +Oc]s, [>read: CP =IP
+Comp -Oc]s, [>wrote: VP]s, [>wrote: DP]s, [>wrote: VP =DP]s, [>wrote: VP =CP]s,
[>wrote: PP =DP]s, [>wrote: CP =IP]s, [>wrote: VP =DP +O]s, [>wrote: PP =DP +O]s,
[>wrote: VP =CP +Oc]s, [>wrote: CP =IP +Comp -Oc]s, [>liked: VP]s, [>liked: DP]s,
[>liked: VP =DP]s, [>liked: VP =CP]s, [>liked: PP =DP]s, [>liked: CP =IP]s, [>liked:
VP =DP +O]s, [>liked: PP =DP +O]s, [>liked: VP =CP +Oc]s, [>liked: CP =IP +Comp -Oc]s,
[>saw: VP]s, [>saw: DP]s, [>saw: VP =DP]s, [>saw: VP =CP]s, [>saw: PP =DP]s, [>saw: CP
=IP]s, [>saw: VP =DP +O]s, [>saw: PP =DP +O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP
+Comp -Oc]s, [>loved: VP]s, [>loved: DP]s, [>loved: VP =DP]s, [>loved: VP =CP]s,
[>loved: PP =DP]s, [>loved: CP =IP]s, [>loved: VP =DP +O]s, [>loved: PP =DP +O]s,
[>loved: VP =CP +Oc]s, [>loved: CP =IP +Comp -Oc]s, [>hated: VP]s, [>hated: DP]s,
[>hated: VP =DP]s, [>hated: VP =CP]s, [>hated: PP =DP]s, [>hated: CP =IP]s, [>hated:
VP =DP +O]s, [>hated: PP =DP +O]s, [>hated: VP =CP +Oc]s, [>hated: CP =IP +Comp -Oc]s,
[>with: VP]s, [>with: DP]s, [>with: VP =DP]s, [>with: VP =CP]s, [>with: PP =DP]s,
[>with: CP =IP]s, [>with: VP =DP +O]s, [>with: PP =DP +O]s, [>with: VP =CP +Oc]s,
[>with: CP =IP +Comp -Oc]s, [>by: VP]s, [>by: DP]s, [>by: VP =DP]s, [>by: VP =CP]s,
[>by: PP =DP]s, [>by: CP =IP]s, [>by: VP =DP +O]s, [>by: PP =DP +O]s, [>by: VP =CP
+Oc]s, [>by: CP =IP +Comp -Oc]s, [>above: VP]s, [>above: DP]s, [>above: VP =DP]s,
[>above: VP =CP]s, [>above: PP =DP]s, [>above: CP =IP]s, [>above: VP =DP +O]s,
[>above: PP =DP +O]s, [>above: VP =CP +Oc]s, [>above: CP =IP +Comp -Oc]s, [>under:
VP]s, [>under: DP]s, [>under: VP =DP]s, [>under: VP =CP]s, [>under: PP =DP]s, [>under:
CP =IP]s, [>under: VP =DP +O]s, [>under: PP =DP +O]s, [>under: VP =CP +Oc]s, [>under:
CP =IP +Comp -Oc]s, [>that: VP]s, [>that: DP]s, [>that: VP =DP]s, [>that: VP =CP]s,
[>that: PP =DP]s, [>that: CP =IP]s, [>that: VP =DP +O]s, [>that: PP =DP +O]s, [>that:
VP =CP +Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP]s, [>knows: DP]s, [>knows: VP
=DP]s, [>knows: VP =CP]s, [>knows: PP =DP]s, [>knows: CP =IP]s, [>knows: VP =DP +O]s,
[>knows: PP =DP +O]s, [>knows: VP =CP +Oc]s, [>knows: CP =IP +Comp -Oc]s, [>says:
VP]s, [>says: DP]s, [>says: VP =DP]s, [>says: VP =CP]s, [>says: PP =DP]s, [>says: CP
=IP]s, [>says: VP =DP +O]s, [>says: PP =DP +O]s, [>says: VP =CP +Oc]s, [>says: CP =IP
+Comp -Oc]s, [>thinks: VP]s, [>thinks: DP]s, [>thinks: VP =DP]s, [>thinks: VP =CP]s,
[>thinks: PP =DP]s, [>thinks: CP =IP]s, [>thinks: VP =DP +O]s, [>thinks: PP =DP +O]s,
[>thinks: VP =CP +Oc]s, [>thinks: CP =IP +Comp -Oc]s, [>assumes: VP]s, [>assumes:
DP]s, [>assumes: VP =DP]s, [>assumes: VP =CP]s, [>assumes: PP =DP]s, [>assumes: CP
=IP]s, [>assumes: VP =DP +O]s, [>assumes: PP =DP +O]s, [>assumes: VP =CP +Oc]s,
[>assumes: CP =IP +Comp -Oc]s]
```

Initial energy:

```
Temperature: 100.000000
Energy: 10022
Grammar: 5143
Input: 4879
```

After 100 iterations:

```
Iteration: 100
```

```
2017_05_10__06_40_50: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: DP]s, [>Jerry: PP =DP]s, [>Jerry: VP =DP]s,
[>Jerry: VP =CP]s, [>Jerry: VP =DP +O]s, [>Jerry: PP =DP +O]s, [>Jerry: VP =CP +Oc]s,
[>Jerry: CP =IP +Comp -Oc]s, [>George: VP]s, [>George: DP]s, [>George: VP =CP]s,
[>George: PP =DP]s, [>George: CP =IP]s, [>George: VP =DP]s, [>George: VP =DP +O]s,
[>George: VP =CP +Oc]s, [>George: CP =IP +Comp -Oc]s, [>Elaine: VP]s, [>Elaine: DP]s,
[>Elaine: VP =DP]s, [>Elaine: VP =CP]s, [>Elaine: PP =DP]s, [>Elaine: CP =IP]s,
[>Elaine: VP =DP +O]s, [>Elaine: VP =CP +Oc]s, [>Elaine: CP =IP +Comp -Oc]s, [>Kramer:
VP]s, [>Kramer: DP]s, [>Kramer: PP =DP]s, [>Kramer: VP =DP +O]s, [>Kramer: PP =DP
+O]s, [>Kramer: VP =CP +Oc]s, [>Kramer: CP =IP +Comp -Oc]s, [>ran: VP]s, [>ran: PP
=DP]s, [>ran: CP =IP]s, [>ran: VP =CP]s, [>ran: PP =DP +O]s, [>ran: VP =CP +Oc]s,
[>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [>walked: DP]s, [>walked: VP =DP]s,
[>walked: VP =CP]s, [>walked: CP =IP]s, [>walked: VP =DP +O]s, [>walked: CP =IP +Comp
-Oc]s, [>read: VP]s, [>read: VP =CP]s, [>read: PP =DP]s, [>read: VP =DP +O]s, [>read:
PP =DP +O]s, [>wrote: VP]s, [>wrote: PP =DP]s, [>wrote: CP =IP]s, [>wrote: VP =DP
+O]s, [>wrote: VP =CP +Oc]s, [>liked: VP]s, [>liked: VP =DP]s, [>liked: VP =CP]s,
[>liked: VP =DP +O]s, [>liked: CP =IP +Comp -Oc]s, [>saw: VP]s, [>saw: VP =DP]s,
[>saw: VP =CP]s, [>saw: PP =DP]s, [>saw: CP =IP]s, [>saw: VP =DP +O]s, [>saw: PP =DP
+O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP +Comp -Oc]s, [>loved: DP]s, [>loved: VP
=DP]s, [>loved: VP =CP]s, [>loved: CP =IP]s, [>loved: VP =DP +O]s, [>loved: VP =CP
+Oc]s, [>loved: CP =IP +Comp -Oc]s, [>hated: VP]s, [>hated: DP]s, [>hated: CP =IP]s,
[>hated: VP =DP +O]s, [>hated: VP =CP +Oc]s, [>hated: CP =IP +Comp -Oc]s, [>with:
DP]s, [>with: VP =CP]s, [>with: PP =DP]s, [>with: CP =IP]s, [>with: VP =DP +O]s,
[>with: PP =DP +O]s, [>by: DP]s, [>by: VP =DP]s, [>by: VP =CP]s, [>by: PP =DP]s, [>by:
CP =IP]s, [>by: PP =DP +O]s, [>by: VP =DP +O]s, [>above: VP]s, [>above: VP =DP]s,
[>above: VP =CP]s, [>above: PP =DP]s, [>above: CP =IP]s, [>above: VP =DP +O]s,
[>above: PP =DP +O]s, [>above: VP =CP +Oc]s, [>under: DP]s, [>under: VP =CP]s,
[>under: CP =IP]s, [>under: VP =DP +O]s, [>under: PP =DP +O]s, [>under: VP =CP +Oc]s,
[>that: VP =DP]s, [>that: VP =CP]s, [>that: PP =DP]s, [>that: CP =IP]s, [>that: VP =DP
+O]s, [>that: PP =DP +O]s, [>that: VP =CP +Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows:
VP]s, [>knows: DP]s, [>knows: VP =DP]s, [>knows: VP =CP]s, [>knows: PP =DP]s, [>knows:
PP =DP +O]s, [>says: VP =DP]s, [>says: VP =CP]s, [>says: PP =DP]s, [>says: CP =IP]s,
[>says: PP =DP +O]s, [>says: CP =IP +Comp -Oc]s, [>thinks: VP]s, [>thinks: DP]s,
[>thinks: VP =DP]s, [>thinks: VP =CP]s, [>thinks: PP =DP]s, [>thinks: CP =IP]s,
[>thinks: VP =DP +O]s, [>thinks: VP =CP +Oc]s, [>assumes: VP]s, [>assumes: DP]s,
[>assumes: VP =DP]s, [>assumes: VP =CP]s, [>assumes: PP =DP]s, [>assumes: CP =IP]s,
[>assumes: PP =DP +O]s, [>assumes: VP =CP +Oc]s]
2017 05 10  06 40 50: Energy: 8604
2017_05_10__06_40_50: Grammar: 3621
2017_05_10__06_40_50: Input: 4983
2017_05_10__06_40_50: Temperature: 74.048426
```

After 300 iterations:

```
Iteration: 300
2017_05_10__06_46_13: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: DP]s, [>Jerry: PP =DP]s, [>Jerry: VP =DP]s,
[>Jerry: VP =CP]s, [>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s, [>George: DP]s,
[>George: PP =DP]s, [>George: CP =IP]s, [>George: VP =DP]s, [>George: VP =DP +O]s,
[>Elaine: DP]s, [>Elaine: PP =DP]s, [>Elaine: VP =DP +O]s, [>Kramer: DP]s, [>Kramer:
PP =DP]s, [>ran: VP]s, [>ran: VP =DP]s, [>walked: VP]s, [>walked: DP]s, [>walked: VP
=CP +Oc]s, [>read: VP]s, [>read: VP =CP]s, [>read: CP =IP]s, [>read: VP =DP]s,
[>wrote: VP]s, [>wrote: DP]s, [>wrote: PP =DP +O]s, [>liked: DP]s, [>liked: VP =DP]s,
[>liked: VP =CP +Oc]s, [>saw: VP =DP]s, [>saw: CP =IP +Comp -Oc]s, [>loved: VP]s,
[>loved: PP =DP]s, [>loved: VP =DP +O]s, [>hated: DP]s, [>hated: CP =IP]s, [>hated: VP
=DP +O]s, [>with: DP]s, [>by: DP]s, [>by: CP =IP]s, [>by: VP =DP]s, [>above: PP =DP
+O]s, [>above: VP =CP +Oc]s, [>under: DP]s, [>under: CP =IP]s, [>under: PP =DP +O]s,
[>that: DP]s, [>that: VP]s, [>that: CP =IP]s, [>that: PP =DP +O]s, [>knows: DP]s,
[>knows: VP =DP]s, [>knows: VP =CP +Oc]s, [>knows: CP =IP +Comp -Oc]s, [>says: DP]s,
[>says: VP =CP]s, [>says: CP =IP +Comp -Oc]s, [>thinks: DP]s, [>thinks: VP =CP]s,
[>thinks: PP =DP]s, [>thinks: PP =DP +O]s, [>thinks: CP =IP +Comp -Oc]s, [>assumes: VP
=CP]s, [>assumes: CP =IP +Comp -Oc]s]
2017_05_10__06_46_13: Energy: 6301
2017_05_10__06_46_13: Grammar: 1620
2017_05_10__06_46_13: Input: 4681
2017_05_10__06_46_13: Temperature: 40.602006
```

After over 600 iterations:

```
Iteration: 668
2017_05_10__06_47_58: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>Jerry: PP =DP]s,
```

```
[>George: DP]s, [>George: PP =DP]s,
[>Elaine: DP]s, [>Elaine: PP =DP]s,
[>Kramer: DP]s, [>Kramer: PP =DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s,
[>loved: VP =DP +O]s, [>hated: VP =DP +O]s,
[>with: DP]s, [>by: DP]s, [>above: DP]s, [>under: DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>thinks: VP =CP]s, [>assumes: VP =CP]s]
2017_05_10__06_47_58: Energy: 4498
2017_05_10__06_47_58: Grammar: 621
2017_05_10__06_47_58: Input: 3877
2017_05_10__06_47_58: Temperature: 13.438963
```

Language Learner

Head Initial Input

The relevant log file for this run is:

log_2017_05_10__06_51_45 - Language, Head-initial, PP & CP.txt

The generated input:

```
Input is: ['George says that Kramer ran', 'Elaine liked George', 'Elaine thinks that
George liked Jerry above Jerry', 'George walked above Jerry', 'Elaine hated Elaine by
Jerry', 'Jerry says that Jerry ran', 'Jerry says that Kramer walked under Elaine',
'Kramer ran under George', 'George ran', 'Jerry read', 'George liked George', 'Jerry
says that Jerry walked above Jerry', 'Kramer saw Elaine with Jerry', 'George walked
with George', 'Elaine walked by George', 'Jerry read', 'Elaine assumes that George
read', 'Jerry hated Jerry', 'Jerry says that Elaine ran with Elaine', 'Jerry read by
Kramer', 'Jerry knows that Jerry liked George', 'George wrote under George', 'Jerry
read under Jerry', 'Elaine loved Elaine', 'Jerry loved Kramer', 'Kramer walked by
Kramer', 'Elaine ran', 'Jerry read by Kramer', 'Jerry read under Kramer', 'Elaine
hated George', 'Elaine liked Elaine above George', 'Elaine ran by Kramer', 'Elaine
ran', 'Jerry liked Kramer', 'George knows that George saw Kramer by George', 'Kramer
assumes that Kramer loved Kramer under George', 'George liked George above Kramer',
'George hated George', 'Kramer loved Jerry above Jerry', 'George liked Kramer', 'Jerry
ran', 'Kramer knows that Elaine loved George above Kramer', 'Jerry loved Jerry',
'Kramer ran under Kramer', 'Elaine liked Jerry with Elaine', 'George wrote with
Kramer', 'Elaine assumes that Elaine read with Jerry', 'George thinks that George
ran', 'Jerry liked Jerry', 'Elaine liked George with George']
```

The initial hypothesis:

```
Initial hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP
=IP -Comp]s, [>Jerry: VP]s, [>Jerry: DP]s, [>Jerry: PP =DP]s, [>Jerry: CP =IP]s,
[>Jerry: VP =DP]s, [>Jerry: VP =CP]s, [>Jerry: VP =DP +O]s, [>Jerry: PP =DP +O]s,
[>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s, [>George: VP]s, [>George: DP]s,
[>George: PP =DP]s, [>George: CP =IP]s, [>George: VP =DP]s, [>George: VP =CP]s,
[>George: VP =DP +O]s, [>George: PP =DP +O]s, [>George: VP =CP +Oc]s, [>George: CP =IP
+Comp -Oc]s, [>Elaine: VP]s, [>Elaine: DP]s, [>Elaine: PP =DP]s, [>Elaine: CP =IP]s,
[>Elaine: VP =DP]s, [>Elaine: VP =CP]s, [>Elaine: VP =DP +O]s, [>Elaine: PP =DP +O]s,
[>Elaine: VP =CP +Oc]s, [>Elaine: CP =IP +Comp -Oc]s, [>Kramer: VP]s, [>Kramer: DP]s,
[>Kramer: PP =DP]s, [>Kramer: CP =IP]s, [>Kramer: VP =DP]s, [>Kramer: VP =CP]s,
[>Kramer: VP =DP +O]s, [>Kramer: PP =DP +O]s, [>Kramer: VP =CP +Oc]s, [>Kramer: CP =IP
+Comp -Oc]s, [>ran: VP]s, [>ran: DP]s, [>ran: PP =DP]s, [>ran: CP =IP]s, [>ran: VP
=DP]s, [>ran: VP =CP]s, [>ran: VP =DP +O]s, [>ran: PP =DP +O]s, [>ran: VP =CP +Oc]s,
[>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [>walked: DP]s, [>walked: PP =DP]s,
[>walked: CP =IP]s, [>walked: VP =DP]s, [>walked: VP =CP]s, [>walked: VP =DP +O]s,
[>walked: PP =DP +O]s, [>walked: VP =CP +Oc]s, [>walked: CP =IP +Comp -Oc]s, [>read:
VP]s, [>read: DP]s, [>read: PP =DP]s, [>read: CP =IP]s, [>read: VP =DP]s, [>read: VP
=CP]s, [>read: VP =DP +O]s, [>read: PP =DP +O]s, [>read: VP =CP +Oc]s, [>read: CP =IP
+Comp -Oc]s, [>wrote: VP]s, [>wrote: DP]s, [>wrote: PP =DP]s, [>wrote: VP =CP]s,
[>wrote: CP =IP]s, [>wrote: VP =DP]s, [>wrote: VP =DP +O]s, [>wrote: PP =DP +O]s,
```

```
[>wrote: VP =CP +Oc]s, [>wrote: CP =IP +Comp -Oc]s, [>liked: VP]s, [>liked: DP]s,
[>liked: PP =DP]s, [>liked: CP =IP]s, [>liked: VP =DP]s, [>liked: VP =CP]s, [>liked:
VP =DP +O]s, [>liked: PP =DP +O]s, [>liked: VP =CP +Oc]s, [>liked: CP =IP +Comp -Oc]s,
[>saw: VP]s, [>saw: DP]s, [>saw: PP =DP]s, [>saw: CP =IP]s, [>saw: VP =DP]s, [>saw: VP
=CP]s, [>saw: VP =DP +O]s, [>saw: PP =DP +O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP
+Comp -Oc]s, [>loved: VP]s, [>loved: DP]s, [>loved: VP =CP]s, [>loved: PP =DP]s,
[>loved: CP =IP]s, [>loved: VP =DP]s, [>loved: VP =DP +O]s, [>loved: PP =DP +O]s,
[>loved: VP =CP +Oc]s, [>loved: CP =IP +Comp -Oc]s, [>hated: VP]s, [>hated: DP]s,
[>hated: PP =DP]s, [>hated: CP =IP]s, [>hated: VP =DP]s, [>hated: VP =CP]s, [>hated:
VP =DP +O]s, [>hated: PP =DP +O]s, [>hated: VP =CP +Oc]s, [>hated: CP =IP +Comp -Oc]s,
[>with: VP]s, [>with: DP]s, [>with: PP =DP]s, [>with: CP =IP]s, [>with: VP =DP]s,
[>with: VP =CP]s, [>with: VP =DP +O]s, [>with: PP =DP +O]s, [>with: VP =CP +Oc]s,
[>with: CP =IP +Comp -Oc]s, [>by: VP]s, [>by: DP]s, [>by: PP =DP]s, [>by: CP =IP]s,
[>by: VP =DP]s, [>by: VP =CP]s, [>by: VP =DP +O]s, [>by: PP =DP +O]s, [>by: VP =CP
+Oc]s, [>by: CP =IP +Comp -Oc]s, [>above: VP]s, [>above: DP]s, [>above: PP =DP]s,
[>above: CP =IP]s, [>above: VP =DP]s, [>above: VP =CP]s, [>above: VP =DP +O]s,
[>above: PP =DP +O]s, [>above: VP =CP +Oc]s, [>above: CP =IP +Comp -Oc]s, [>under:
VP]s, [>under: DP]s, [>under: PP =DP]s, [>under: CP =IP]s, [>under: VP =DP]s, [>under:
VP =CP]s, [>under: VP =DP +O]s, [>under: PP =DP +O]s, [>under: VP =CP +Oc]s, [>under:
CP =IP +Comp -Oc]s, [>that: VP]s, [>that: DP]s, [>that: PP =DP]s, [>that: CP =IP]s,
[>that: VP =DP]s, [>that: VP =CP]s, [>that: VP =DP +O]s, [>that: PP =DP +O]s, [>that:
VP =CP +Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP]s, [>knows: DP]s, [>knows: PP
=DP]s, [>knows: VP =CP]s, [>knows: CP =IP]s, [>knows: VP =DP]s, [>knows: VP =DP +O]s,
[>knows: PP =DP +O]s, [>knows: VP =CP +Oc]s, [>knows: CP =IP +Comp -Oc]s, [>says:
VP]s, [>says: DP]s, [>says: PP =DP]s, [>says: CP =IP]s, [>says: VP =DP]s, [>says: VP
=CP]s, [>says: VP =DP +O]s, [>says: PP =DP +O]s, [>says: VP =CP +Oc]s, [>says: CP =IP
+Comp -Oc]s, [>thinks: VP]s, [>thinks: DP]s, [>thinks: PP =DP]s, [>thinks: CP =IP]s,
[>thinks: VP =DP]s, [>thinks: VP =CP]s, [>thinks: VP =DP +O]s, [>thinks: PP =DP +O]s,
[>thinks: VP =CP +Oc]s, [>thinks: CP =IP +Comp -Oc]s, [>assumes: VP]s, [>assumes:
DP]s, [>assumes: PP =DP]s, [>assumes: CP =IP]s, [>assumes: VP =DP]s, [>assumes: VP
=CP]s, [>assumes: VP =DP +O]s, [>assumes: PP =DP +O]s, [>assumes: VP =CP +Oc]s,
[>assumes: CP =IP +Comp -Oc]s]
```

Initial energy:

```
Temperature: 100.000000
Energy: 10397
Grammar: 5144
Input: 5253
```

After 100 iterations:

```
Iteration: 100
2017 05 10  08 19 07: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: VP]s, [>Jerry: DP]s, [>Jerry: DP]s, [>Jerry: PP
=DP]s, [>Jerry: VP =DP]s, [>Jerry: VP =CP]s, [>Jerry: PP =DP +O]s, [>Jerry: VP =CP
+Oc]s, [>Jerry: CP =IP +Comp -Oc]s, [>George: DP]s, [>George: CP =IP]s, [>George: PP
=DP]s, [>George: VP =DP +O]s, [>George: VP =CP +Oc]s, [>George: CP =IP +Comp -Oc]s,
[>Elaine: VP]s, [>Elaine: DP]s, [>Elaine: PP =DP]s, [>Elaine: VP =CP]s, [>Elaine: VP
=DP +O]s, [>Kramer: VP]s, [>Kramer: DP]s, [>Kramer: PP =DP]s, [>Kramer: CP =IP]s,
[>Kramer: VP =DP]s, [>Kramer: VP =CP]s, [>Kramer: VP =DP +O]s, [>Kramer: PP =DP +O]s,
[>Kramer: VP =CP +Oc]s, [>ran: VP]s, [>ran: PP =DP]s, [>ran: VP =DP]s, [>ran: VP =DP
+O]s, [>ran: VP =CP +Oc]s, [>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [>walked: DP]s,
[>walked: DP]s, [>walked: VP =DP]s, [>walked: VP =CP]s, [>walked: VP =DP +O]s,
[>walked: VP =CP +Oc]s, [>walked: CP =IP +Comp -Oc]s, [>read: VP]s, [>read: DP]s,
[>read: PP =DP]s, [>read: VP =DP]s, [>read: VP =CP]s, [>read: VP =DP +O]s, [>read: PP
=DP +O]s, [>read: VP =CP +Oc]s, [>read: CP =IP +Comp -Oc]s, [>wrote: VP]s, [>wrote:
DP]s, [>wrote: PP =DP]s, [>wrote: CP =IP]s, [>wrote: VP =DP]s, [>wrote: PP =DP +O]s,
[>liked: VP]s, [>liked: DP]s, [>liked: CP =IP]s, [>liked: VP =DP]s, [>liked: PP =DP
+O]s, [>liked: VP =CP +Oc]s, [>saw: DP]s, [>saw: VP]s, [>saw: CP =IP]s, [>saw: VP
=CP]s, [>saw: VP =DP +O]s, [>saw: PP =DP +O]s, [>saw: VP =CP +Oc]s, [>loved: DP]s,
[>loved: VP =CP]s, [>loved: PP =DP]s, [>loved: VP =DP]s, [>loved: VP =DP +O]s,
[>loved: PP =DP +O]s, [>loved: VP =CP +Oc]s, [>loved: CP =IP +Comp -Oc]s, [>hated:
VP]s, [>hated: DP]s, [>hated: PP =DP]s, [>hated: CP =IP]s, [>hated: VP =DP]s, [>hated:
VP =CP]s, [>hated: VP =DP +O]s, [>hated: PP =DP +O]s, [>with: VP]s, [>with: DP]s,
[>with: PP =DP]s, [>with: CP =IP]s, [>with: VP =DP]s, [>with: VP =CP]s, [>with: VP =DP
+O]s, [>with: VP =CP +Oc]s, [>with: CP =IP +Comp -Oc]s, [>by: VP]s, [>by: DP]s, [>by:
PP =DP]s, [>by: CP =IP]s, [>by: VP =CP]s, [>by: VP =DP +O]s, [>by: PP =DP +O]s, [>by:
CP =IP +Comp -Oc]s, [>above: VP]s, [>above: DP]s, [>above: PP =DP]s, [>above: CP
=IP]s, [>above: VP =DP]s, [>above: VP =CP]s, [>above: VP =DP +O]s, [>above: PP =DP
+O]s, [>above: CP =IP +Comp -Oc]s, [>under: VP]s, [>under: DP]s, [>under: PP =DP]s,
[>under: CP =IP]s, [>under: VP =CP]s, [>under: VP =DP +O]s, [>under: PP =DP +O]s,
[>under: VP =CP +Oc]s, [>under: CP =IP +Comp -Oc]s, [>that: VP]s, [>that: DP]s,
```

```
[>that: CP =IP]s, [>that: PP =DP]s, [>that: VP =DP +O]s, [>knows: VP]s, [>knows: DP]s,
[>knows: PP =DP]s, [>knows: VP =CP]s, [>knows: CP =IP]s, [>knows: VP =DP]s, [>knows:
PP =DP +O]s, [>knows: VP =CP +Oc]s, [>knows: CP =IP +Comp -Oc]s, [>says: VP]s, [>says:
DP]s, [>says: PP =DP]s, [>says: VP =CP]s, [>says: CP =IP]s, [>says: VP =DP +O]s,
[>says: VP =CP +Oc]s, [>thinks: DP]s, [>thinks: VP]s, [>thinks: PP =DP]s, [>thinks: VP
=DP]s, [>thinks: VP =CP]s, [>thinks: VP =DP +O]s, [>thinks: PP =DP +O]s, [>thinks: CP
=IP +Comp -Oc]s, [>assumes: DP]s, [>assumes: VP =DP]s, [>assumes: VP =DP +O]s,
[>assumes: PP =DP +O]s, [>assumes: VP =CP +Oc]s, [>assumes: CP =IP +Comp -Oc]s]
2017_05_10__08_19_07: Energy: 9165
2017_05_10__08_19_07: Grammar: 3799
2017_05_10__08_19_07: Input: 5366
2017_05_10__08_19_07: Temperature: 74.048426
```

After 300 iterations:

```
Iteration: 300
2017_05_10__08_36_45: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: DP]s, [>Jerry: DP]s, [>Jerry: VP =DP]s, [>Jerry:
VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s, [>George: DP]s, [>George: PP =DP +O]s,
[>George: VP =CP +Oc]s, [>Elaine: VP]s, [>Elaine: DP]s, [>Elaine: VP =CP +Oc]s,
[>Kramer: DP]s, [>Kramer: VP =DP +O]s, [>Kramer: PP =DP +O]s, [>ran: VP]s, [>walked:
VP]s, [>walked: VP =DP]s, [>walked: VP =CP]s, [>walked: VP =DP +O]s, [>walked: CP =IP
+Comp -Oc]s, [>read: VP]s, [>read: PP =DP +O]s, [>wrote: VP]s, [>wrote: DP]s, [>wrote:
VP =DP]s, [>liked: CP =IP]s, [>liked: VP =DP]s, [>liked: CP =IP]s, [>liked: VP =CP
+Oc]s, [>saw: DP]s, [>saw: VP]s, [>saw: VP =CP]s, [>saw: VP =DP +O]s, [>loved: DP]s,
[>loved: VP =DP]s, [>loved: VP =DP +O]s, [>loved: VP =DP +O]s, [>hated: PP =DP]s,
[>hated: VP =DP]s, [>hated: PP =DP +O]s, [>hated: CP =IP +Comp -Oc]s, [>with: PP
=DP]s, [>with: CP =IP +Comp -Oc]s, [>by: PP =DP]s, [>by: CP =IP]s, [>above: VP =CP]s,
[>above: PP =DP]s, [>above: CP =IP +Comp -Oc]s, [>under: PP =DP]s, [>under: CP =IP
+Comp -Oc]s, [>that: DP]s, [>that: CP =IP]s, [>knows: DP]s, [>knows: VP =CP]s,
[>knows: PP =DP]s, [>says: VP =CP]s, [>says: CP =IP]s, [>says: VP =DP +O]s, [>thinks:
VP]s, [>thinks: VP =CP]s, [>thinks: PP =DP]s, [>thinks: VP =DP +O]s, [>thinks: VP =CP
+Oc]s, [>assumes: DP]s, [>assumes: VP =DP]s, [>assumes: VP =DP]s]
2017_05_10__08_36_45: Energy: 6456
2017_05_10__08_36_45: Grammar: 1644
2017_05_10__08_36_45: Input: 4812
2017_05_10__08_36_45: Temperature: 40.602006
```

After over 500 iterations:

```
Iteration: 596
2017_05_10__08_38_30: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>George: DP]s, [>Elaine: DP]s, [>Kramer: DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s, [>loved: VP =DP]s, [>hated: VP =DP]s,
[>with: PP =DP]s, [>by: PP =DP]s, [>above: PP =DP]s, [>under: PP =DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>thinks: VP =CP]s, [>assumes: VP =CP]s]
2017 05 10  08 38 30: Energy: 4111
2017_05_10__08_38_30: Grammar: 544
2017_05_10__08_38_30: Input: 3567
2017_05_10__08_38_30: Temperature: 16.684544
```

Head Final Input

The relevant log file for this run is:

log_2017_05_10__13_38_44 - Language, Head-final, PP & CP.txt

The generated input:

```
Input is: ['Jerry George liked Elaine by', 'Kramer Kramer ran Kramer above that
knows', 'Kramer read George above', 'Elaine Jerry loved', 'Elaine Jerry ran Elaine
with that assumes', 'George Jerry loved Elaine under', 'George Jerry walked that
thinks', 'Kramer ran George above', 'Elaine Jerry hated', 'Kramer read Kramer with',
```

```
'Kramer wrote', 'George walked Elaine by', 'Elaine Kramer saw', 'George Kramer Jerry
loved Elaine with that thinks', 'Elaine wrote', 'Elaine ran Kramer under', 'Elaine
Jerry saw Elaine above', 'Elaine Elaine liked', 'Jerry Kramer wrote Kramer under that
knows', 'George George wrote that thinks', 'George Elaine Elaine saw George by that
thinks', 'Jerry Jerry liked', 'George Jerry George loved that assumes', 'Jerry George
hated Elaine under', 'Elaine Elaine hated', 'Jerry George loved Kramer with', 'Kramer
Jerry saw Elaine under', 'Jerry Jerry liked Kramer under', 'Jerry Elaine walked Kramer
above that thinks', 'Jerry George loved', 'Elaine Jerry saw Kramer with', 'Elaine
walked', 'Kramer Elaine loved Jerry by', 'George Kramer liked Elaine above', 'Elaine
Kramer Kramer liked that knows', 'Jerry George loved George with', 'George Jerry
loved', 'George Elaine saw Elaine with', 'Jerry Elaine loved', 'Kramer Kramer liked
Jerry under', 'George Kramer saw George with', 'Kramer wrote', 'George read Kramer
with', 'George ran', 'Elaine George liked Elaine with', 'Kramer Kramer walked that
thinks Jerry under', 'Kramer Jerry liked', 'Jerry Jerry saw', 'Kramer George hated
Elaine under', 'Jerry ran']
```

The initial hypothesis:

```
Initial hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP
=IP -Comp]s, [>Jerry: VP]s, [>Jerry: DP]s, [>Jerry: CP =IP]s, [>Jerry: VP =DP]s,
[>Jerry: VP =CP]s, [>Jerry: PP =DP]s, [>Jerry: VP =DP +O]s, [>Jerry: PP =DP +O]s,
[>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s, [>George: VP]s, [>George: DP]s,
[>George: CP =IP]s, [>George: VP =DP]s, [>George: VP =CP]s, [>George: PP =DP]s,
[>George: VP =DP +O]s, [>George: PP =DP +O]s, [>George: VP =CP +Oc]s, [>George: CP =IP
+Comp -Oc]s, [>Elaine: VP]s, [>Elaine: DP]s, [>Elaine: CP =IP]s, [>Elaine: VP =DP]s,
[>Elaine: VP =CP]s, [>Elaine: PP =DP]s, [>Elaine: VP =DP +O]s, [>Elaine: PP =DP +O]s,
[>Elaine: VP =CP +Oc]s, [>Elaine: CP =IP +Comp -Oc]s, [>Kramer: VP]s, [>Kramer: DP]s,
[>Kramer: CP =IP]s, [>Kramer: VP =DP]s, [>Kramer: VP =CP]s, [>Kramer: PP =DP]s,
[>Kramer: VP =DP +O]s, [>Kramer: PP =DP +O]s, [>Kramer: VP =CP +Oc]s, [>Kramer: CP =IP
+Comp -Oc]s, [>ran: VP]s, [>ran: DP]s, [>ran: CP =IP]s, [>ran: VP =DP]s, [>ran: VP
=CP]s, [>ran: PP =DP]s, [>ran: VP =DP +O]s, [>ran: PP =DP +O]s, [>ran: VP =CP +Oc]s,
[>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [>walked: DP]s, [>walked: CP =IP]s,
[>walked: VP =DP]s, [>walked: VP =CP]s, [>walked: PP =DP]s, [>walked: VP =DP +O]s,
[>walked: PP =DP +O]s, [>walked: VP =CP +Oc]s, [>walked: CP =IP +Comp -Oc]s, [>read:
VP]s, [>read: DP]s, [>read: VP =DP]s, [>read: CP =IP]s, [>read: VP =CP]s, [>read: PP
=DP]s, [>read: VP =DP +O]s, [>read: PP =DP +O]s, [>read: VP =CP +Oc]s, [>read: CP =IP
+Comp -Oc]s, [>wrote: VP]s, [>wrote: DP]s, [>wrote: CP =IP]s, [>wrote: VP =DP]s,
[>wrote: VP =CP]s, [>wrote: PP =DP]s, [>wrote: VP =DP +O]s, [>wrote: PP =DP +O]s,
[>wrote: VP =CP +Oc]s, [>wrote: CP =IP +Comp -Oc]s, [>liked: VP]s, [>liked: DP]s,
[>liked: CP =IP]s, [>liked: VP =DP]s, [>liked: VP =CP]s, [>liked: PP =DP]s, [>liked:
VP =DP +O]s, [>liked: PP =DP +O]s, [>liked: VP =CP +Oc]s, [>liked: CP =IP +Comp -Oc]s,
[>saw: VP]s, [>saw: DP]s, [>saw: CP =IP]s, [>saw: VP =DP]s, [>saw: VP =CP]s, [>saw: PP
=DP]s, [>saw: VP =DP +O]s, [>saw: PP =DP +O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP
+Comp -Oc]s, [>loved: VP]s, [>loved: DP]s, [>loved: VP =DP]s, [>loved: CP =IP]s,
[>loved: VP =CP]s, [>loved: PP =DP]s, [>loved: VP =DP +O]s, [>loved: PP =DP +O]s,
[>loved: VP =CP +Oc]s, [>loved: CP =IP +Comp -Oc]s, [>hated: VP]s, [>hated: DP]s,
[>hated: CP =IP]s, [>hated: VP =DP]s, [>hated: VP =CP]s, [>hated: PP =DP]s, [>hated:
VP =DP +O]s, [>hated: PP =DP +O]s, [>hated: VP =CP +Oc]s, [>hated: CP =IP +Comp -Oc]s,
[>with: VP]s, [>with: DP]s, [>with: CP =IP]s, [>with: VP =DP]s, [>with: VP =CP]s,
[>with: PP =DP]s, [>with: VP =DP +O]s, [>with: PP =DP +O]s, [>with: VP =CP +Oc]s,
[>with: CP =IP +Comp -Oc]s, [>by: VP]s, [>by: DP]s, [>by: CP =IP]s, [>by: VP =DP]s,
[>by: VP =CP]s, [>by: PP =DP]s, [>by: VP =DP +O]s, [>by: PP =DP +O]s, [>by: VP =CP
+Oc]s, [>by: CP =IP +Comp -Oc]s, [>above: VP]s, [>above: DP]s, [>above: CP =IP]s,
[>above: VP =DP]s, [>above: VP =CP]s, [>above: PP =DP]s, [>above: VP =DP +O]s,
[>above: PP =DP +O]s, [>above: VP =CP +Oc]s, [>above: CP =IP +Comp -Oc]s, [>under:
VP]s, [>under: DP]s, [>under: CP =IP]s, [>under: VP =DP]s, [>under: VP =CP]s, [>under:
PP =DP]s, [>under: VP =DP +O]s, [>under: PP =DP +O]s, [>under: VP =CP +Oc]s, [>under:
CP =IP +Comp -Oc]s, [>that: VP]s, [>that: DP]s, [>that: CP =IP]s, [>that: VP =DP]s,
[>that: VP =CP]s, [>that: PP =DP]s, [>that: VP =DP +O]s, [>that: PP =DP +O]s, [>that:
VP =CP +Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP]s, [>knows: DP]s, [>knows: CP
=IP]s, [>knows: VP =DP]s, [>knows: VP =CP]s, [>knows: PP =DP]s, [>knows: VP =DP +O]s,
[>knows: PP =DP +O]s, [>knows: VP =CP +Oc]s, [>knows: CP =IP +Comp -Oc]s, [>thinks:
VP]s, [>thinks: DP]s, [>thinks: CP =IP]s, [>thinks: VP =DP]s, [>thinks: VP =CP]s,
[>thinks: PP =DP]s, [>thinks: VP =DP +O]s, [>thinks: PP =DP +O]s, [>thinks: VP =CP
+Oc]s, [>thinks: CP =IP +Comp -Oc]s, [>assumes: VP]s, [>assumes: DP]s, [>assumes: CP
=IP]s, [>assumes: VP =DP]s, [>assumes: VP =CP]s, [>assumes: PP =DP]s, [>assumes: VP
=DP +O]s, [>assumes: PP =DP +O]s, [>assumes: VP =CP +Oc]s, [>assumes: CP =IP +Comp -
Oc]s]
```

Initial energy:

```
Temperature: 100.000000
Energy: 10208
```

```
Grammar: 4904
Input: 5304
```

After 100 iterations:

```
Iteration: 100
2017_05_10__14_32_24: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [<Jerry: DP]s, [<Jerry: PP =DP]s, [<Jerry: VP =DP +O]s,
[<Jerry: PP =DP +O]s, [<Jerry: CP =IP +Comp -Oc]s, [<George: VP]s, [<George: DP]s,
[<George: CP =IP]s, [<George: VP =DP]s, [<George: VP =DP +O]s, [<George: VP =CP +Oc]s,
[<George: CP =IP +Comp -Oc]s, [<Elaine: VP]s, [<Elaine: DP]s, [<Elaine: VP =DP]s,
[<Elaine: VP =CP]s, [<Elaine: PP =DP]s, [<Elaine: VP =DP +O]s, [<Elaine: PP =DP +O]s,
[<Elaine: VP =CP +Oc]s, [<Elaine: CP =IP +Comp -Oc]s, [<Kramer: VP]s, [<Kramer: DP]s,
[<Kramer: CP =IP]s, [<Kramer: VP =DP]s, [<Kramer: VP =CP]s, [<Kramer: VP =DP +O]s,
[<Kramer: PP =DP +O]s, [<Kramer: CP =IP +Comp -Oc]s, [<ran: VP]s, [<ran: DP]s, [<ran:
VP =CP]s, [<ran: CP =IP]s, [<ran: VP =DP]s, [<ran: PP =DP]s, [<ran: VP =DP +O]s,
[<ran: PP =DP +O]s, [<ran: CP =IP +Comp -Oc]s, [<walked: VP]s, [<walked: VP =DP]s,
[<walked: VP =DP +O]s, [<walked: VP =CP +Oc]s, [<walked: CP =IP +Comp -Oc]s, [<read:
VP]s, [<read: DP]s, [<read: VP =DP]s, [<read: VP =CP]s, [<read: PP =DP]s, [<read: VP
=DP +O]s, [<read: CP =IP +Comp -Oc]s, [<wrote: VP]s, [<wrote: DP]s, [<wrote: CP =IP]s,
[<wrote: VP =DP]s, [<wrote: VP =CP]s, [<wrote: PP =DP]s, [<wrote: PP =DP +O]s,
[<wrote: VP =CP +Oc]s, [<wrote: CP =IP +Comp -Oc]s, [<liked: VP]s, [<liked: CP =IP]s,
[<liked: VP =DP]s, [<liked: VP =CP]s, [<liked: PP =DP]s, [<liked: VP =DP +O]s,
[<liked: VP =CP +Oc]s, [<saw: VP]s, [<saw: CP =IP]s, [<saw: VP =DP]s, [<saw: VP =DP
+O]s, [<saw: VP =CP +Oc]s, [<saw: CP =IP +Comp -Oc]s, [<loved: VP]s, [<loved: DP]s,
[<loved: CP =IP]s, [<loved: VP =DP]s, [<loved: VP =CP]s, [<loved: PP =DP]s, [<loved:
VP =DP +O]s, [<loved: PP =DP +O]s, [<loved: VP =CP +Oc]s, [<hated: DP]s, [<hated: VP
=CP]s, [<hated: CP =IP]s, [<hated: VP =DP +O]s, [<hated: PP =DP +O]s, [<hated: VP =CP
+Oc]s, [<with: CP =IP]s, [<with: VP =DP]s, [<with: VP =CP]s, [<with: PP =DP]s, [<with:
VP =CP +Oc]s, [<by: VP]s, [<by: VP =DP]s, [<by: VP =CP]s, [<by: PP =DP]s, [<by: VP =DP
+O]s, [<by: PP =DP +O]s, [<by: VP =CP +Oc]s, [<above: VP]s, [<above: CP =IP]s,
[<above: VP =DP]s, [<above: VP =CP]s, [<above: PP =DP]s, [<above: VP =DP +O]s,
[<above: PP =DP +O]s, [<above: CP =IP +Comp -Oc]s, [<under: VP]s, [<under: DP]s,
[<under: CP =IP]s, [<under: VP =DP]s, [<under: VP =CP]s, [<under: PP =DP]s, [<under:
PP =DP +O]s, [<under: VP =CP +Oc]s, [<under: CP =IP +Comp -Oc]s, [<that: VP]s, [<that:
CP =IP]s, [<that: VP =CP]s, [<that: PP =DP]s, [<that: CP =IP +Comp -Oc]s, [<knows:
VP]s, [<knows: DP]s, [<knows: VP =CP]s, [<knows: PP =DP +O]s, [<thinks: VP]s,
[<thinks: DP]s, [<thinks: CP =IP]s, [<thinks: VP =DP]s, [<thinks: VP =CP]s, [<thinks:
VP =DP +O]s, [<thinks: VP =CP +Oc]s, [<thinks: CP =IP +Comp -Oc]s, [<assumes: VP]s,
[<assumes: VP =DP]s, [<assumes: VP =CP]s, [<assumes: PP =DP]s, [<assumes: PP =DP +O]s,
[<assumes: CP =IP +Comp -Oc]s]
2017_05_10__14_32_24: Energy: 8828
2017_05_10__14_32_24: Grammar: 3420
2017_05_10__14_32_24: Input: 5408
2017_05_10__14_32_24: Temperature: 74.048426
```

After 300 iterations:

```
Iteration: 300
2017 05 10  14 40 50: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [<Jerry: DP]s, [<Jerry: VP =DP +O]s, [<Jerry: PP =DP +O]s,
[<George: DP]s, [<George: CP =IP]s, [<George: VP =DP +O]s, [<Elaine: DP]s, [<Elaine:
VP =DP]s, [<Elaine: CP =IP]s, [<Elaine: VP =DP +O]s, [<Elaine: PP =DP +O]s, [<Kramer:
DP]s, [<Kramer: PP =DP]s, [<Kramer: PP =DP +O]s, [<Kramer: CP =IP +Comp -Oc]s, [<ran:
VP]s, [<ran: VP =CP]s, [<walked: VP]s, [<read: VP]s, [<read: VP =CP +Oc]s, [<wrote:
VP]s, [<wrote: DP]s, [<wrote: VP =DP +O]s, [<liked: VP =DP]s, [<saw: VP]s, [<saw: CP
=IP]s, [<saw: VP =DP]s, [<saw: VP =CP +Oc]s, [<saw: CP =IP +Comp -Oc]s, [<loved: VP
=DP]s, [<loved: PP =DP +O]s, [<loved: VP =CP +Oc]s, [<hated: DP]s, [<hated: VP =CP]s,
[<hated: CP =IP]s, [<with: PP =DP]s, [<with: VP =DP]s, [<by: VP]s, [<by: PP =DP]s,
[<by: PP =DP +O]s, [<by: VP =CP +Oc]s, [<above: PP =DP]s, [<under: DP]s, [<under: VP
=CP]s, [<under: VP =DP]s, [<under: VP =DP +O]s, [<that: CP =IP]s, [<knows: VP =CP]s,
[<knows: PP =DP]s, [<knows: PP =DP +O]s, [<knows: VP =DP +O]s, [<thinks: VP]s,
[<thinks: VP =CP]s, [<thinks: CP =IP]s, [<thinks: VP =DP +O]s, [<thinks: CP =IP +Comp
-Oc]s, [<assumes: VP =CP]s, [<assumes: VP =DP]s]
2017_05_10__14_40_50: Energy: 6380
2017_05_10__14_40_50: Grammar: 1460
2017_05_10__14_40_50: Input: 4920
2017_05_10__14_40_50: Temperature: 40.602006
```

After over 400 iterations:

```
Iteration: 470
```

```
2017_05_10__14_41_32: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[<Jerry: DP]s, [<George: DP]s, [<Elaine: DP]s, [<Kramer: DP]s,
[<ran: VP]s, [<walked: VP]s, [<read: VP]s, [<wrote: VP]s,
[<liked: VP =DP]s, [<saw: VP =DP]s, [<loved: VP =DP]s, [<hated: VP =DP]s,
[<with: PP =DP]s, [<by: PP =DP]s, [<above: PP =DP]s, [<under: PP =DP]s,
[<that: CP =IP]s,
[<knows: VP =CP]s, [<thinks: VP =CP]s, [<assumes: VP =CP]s]
2017_05_10__14_41_32: Energy: 4097
2017_05_10__14_41_32: Grammar: 521
2017_05_10__14_41_32: Input: 3576
2017_05_10__14_41_32: Temperature: 24.362643
```

Mixed Category Input

The relevant log file for this run is:

log_2017_05_10__14_45_15 - Language, Mixed-category, PP & CP.txt

The generated input:

```
Input is: ['Jerry liked George Kramer above', 'George wrote Elaine with', 'Kramer ran
Jerry by', 'Elaine says that Elaine wrote Elaine by', 'Jerry ran George with', 'Elaine
loved Jerry', 'Jerry liked Kramer', 'George says that Elaine liked George', 'George
walked George with', 'Elaine read Jerry above', 'Jerry assumes that Elaine loved
Elaine Elaine by', 'Kramer says that Elaine liked Elaine', 'George walked', 'Elaine
ran George under', 'Elaine ran', 'George assumes that Jerry walked Elaine above',
'George ran', 'George says that Elaine read Kramer with', 'Kramer hated Elaine Jerry
under', 'Jerry says that Jerry loved Kramer Elaine under', 'Jerry read Jerry with',
'Kramer read', 'Kramer loved Elaine', 'Elaine wrote Elaine by', 'Elaine hated George',
'Kramer read Kramer under', 'Elaine walked George above', 'Kramer assumes that George
wrote George above', 'Kramer walked', 'Kramer read', 'George loved George Kramer
with', 'Jerry hated Elaine Kramer under', 'Jerry hated George Jerry under', 'George
read', 'Kramer saw Kramer Jerry with', 'Kramer ran', 'Kramer liked Kramer Elaine
under', 'Kramer knows that Elaine saw Elaine Elaine above', 'George wrote George
under', 'Jerry wrote', 'George liked Elaine', 'Kramer read', 'George says that Jerry
ran Jerry under', 'Elaine knows that Elaine read George above', 'Jerry read Jerry by',
'Elaine read', 'Elaine read', 'Kramer thinks that Jerry walked', 'George wrote Jerry
with', 'Kramer saw Jerry']
```

The initial hypothesis:

```
Initial hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP
=IP -Comp]s, [>Jerry: DP]s, [>Jerry: VP]s, [>Jerry: CP =IP]s, [>Jerry: VP =DP]s,
[>Jerry: VP =CP]s, [>Jerry: PP =DP]s, [>Jerry: VP =DP +O]s, [>Jerry: PP =DP +O]s,
[>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s, [>George: DP]s, [>George: VP]s,
[>George: CP =IP]s, [>George: VP =DP]s, [>George: VP =CP]s, [>George: PP =DP]s,
[>George: VP =DP +O]s, [>George: PP =DP +O]s, [>George: VP =CP +Oc]s, [>George: CP =IP
+Comp -Oc]s, [>Elaine: DP]s, [>Elaine: VP]s, [>Elaine: CP =IP]s, [>Elaine: VP =DP]s,
[>Elaine: VP =CP]s, [>Elaine: PP =DP]s, [>Elaine: VP =DP +O]s, [>Elaine: PP =DP +O]s,
[>Elaine: VP =CP +Oc]s, [>Elaine: CP =IP +Comp -Oc]s, [>Kramer: DP]s, [>Kramer: VP]s,
[>Kramer: CP =IP]s, [>Kramer: VP =DP]s, [>Kramer: VP =CP]s, [>Kramer: PP =DP]s,
[>Kramer: VP =DP +O]s, [>Kramer: PP =DP +O]s, [>Kramer: VP =CP +Oc]s, [>Kramer: CP =IP
+Comp -Oc]s, [>ran: DP]s, [>ran: VP]s, [>ran: CP =IP]s, [>ran: VP =DP]s, [>ran: VP
=CP]s, [>ran: PP =DP]s, [>ran: VP =DP +O]s, [>ran: PP =DP +O]s, [>ran: VP =CP +Oc]s,
[>ran: CP =IP +Comp -Oc]s, [>walked: DP]s, [>walked: VP]s, [>walked: CP =IP]s,
[>walked: VP =DP]s, [>walked: VP =CP]s, [>walked: PP =DP]s, [>walked: VP =DP +O]s,
[>walked: PP =DP +O]s, [>walked: VP =CP +Oc]s, [>walked: CP =IP +Comp -Oc]s, [>read:
DP]s, [>read: VP]s, [>read: CP =IP]s, [>read: VP =DP]s, [>read: VP =CP]s, [>read: PP
=DP]s, [>read: VP =DP +O]s, [>read: PP =DP +O]s, [>read: VP =CP +Oc]s, [>read: CP =IP
+Comp -Oc]s, [>wrote: DP]s, [>wrote: VP]s, [>wrote: CP =IP]s, [>wrote: VP =DP]s,
[>wrote: VP =CP]s, [>wrote: PP =DP]s, [>wrote: VP =DP +O]s, [>wrote: PP =DP +O]s,
[>wrote: VP =CP +Oc]s, [>wrote: CP =IP +Comp -Oc]s, [>liked: DP]s, [>liked: VP]s,
[>liked: CP =IP]s, [>liked: VP =DP]s, [>liked: VP =CP]s, [>liked: PP =DP]s, [>liked:
VP =DP +O]s, [>liked: PP =DP +O]s, [>liked: VP =CP +Oc]s, [>liked: CP =IP +Comp -Oc]s,
[>saw: VP]s, [>saw: DP]s, [>saw: CP =IP]s, [>saw: VP =DP]s, [>saw: VP =CP]s, [>saw: PP
=DP]s, [>saw: VP =DP +O]s, [>saw: PP =DP +O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP
+Comp -Oc]s, [>loved: DP]s, [>loved: VP]s, [>loved: CP =IP]s, [>loved: VP =DP]s,
```

```
[>loved: VP =CP]s, [>loved: PP =DP]s, [>loved: VP =DP +O]s, [>loved: PP =DP +O]s,
[>loved: VP =CP +Oc]s, [>loved: CP =IP +Comp -Oc]s, [>hated: VP]s, [>hated: DP]s,
[>hated: CP =IP]s, [>hated: VP =CP]s, [>hated: PP =DP]s, [>hated: VP =DP]s, [>hated:
VP =DP +O]s, [>hated: PP =DP +O]s, [>hated: VP =CP +Oc]s, [>hated: CP =IP +Comp -Oc]s,
[>with: DP]s, [>with: VP]s, [>with: CP =IP]s, [>with: VP =DP]s, [>with: VP =CP]s,
[>with: PP =DP]s, [>with: VP =DP +O]s, [>with: PP =DP +O]s, [>with: VP =CP +Oc]s,
[>with: CP =IP +Comp -Oc]s, [>by: DP]s, [>by: VP]s, [>by: CP =IP]s, [>by: VP =DP]s,
[>by: VP =CP]s, [>by: PP =DP]s, [>by: VP =DP +O]s, [>by: PP =DP +O]s, [>by: VP =CP
+Oc]s, [>by: CP =IP +Comp -Oc]s, [>above: DP]s, [>above: VP]s, [>above: CP =IP]s,
[>above: VP =DP]s, [>above: VP =CP]s, [>above: PP =DP]s, [>above: VP =DP +O]s,
[>above: PP =DP +O]s, [>above: VP =CP +Oc]s, [>above: CP =IP +Comp -Oc]s, [>under:
DP]s, [>under: VP]s, [>under: CP =IP]s, [>under: VP =CP]s, [>under: PP =DP]s, [>under:
VP =DP]s, [>under: VP =DP +O]s, [>under: PP =DP +O]s, [>under: VP =CP +Oc]s, [>under:
CP =IP +Comp -Oc]s, [>that: DP]s, [>that: VP]s, [>that: CP =IP]s, [>that: VP =DP]s,
[>that: VP =CP]s, [>that: PP =DP]s, [>that: VP =DP +O]s, [>that: PP =DP +O]s, [>that:
VP =CP +Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP]s, [>knows: DP]s, [>knows: CP
=IP]s, [>knows: VP =DP]s, [>knows: VP =CP]s, [>knows: PP =DP]s, [>knows: VP =DP +O]s,
[>knows: PP =DP +O]s, [>knows: VP =CP +Oc]s, [>knows: CP =IP +Comp -Oc]s, [>says:
DP]s, [>says: VP]s, [>says: CP =IP]s, [>says: VP =DP]s, [>says: VP =CP]s, [>says: PP
=DP]s, [>says: VP =DP +O]s, [>says: PP =DP +O]s, [>says: VP =CP +Oc]s, [>says: CP =IP
+Comp -Oc]s, [>thinks: VP]s, [>thinks: DP]s, [>thinks: CP =IP]s, [>thinks: VP =DP]s,
[>thinks: VP =CP]s, [>thinks: PP =DP]s, [>thinks: VP =DP +O]s, [>thinks: PP =DP +O]s,
[>thinks: VP =CP +Oc]s, [>thinks: CP =IP +Comp -Oc]s, [>assumes: DP]s, [>assumes:
VP]s, [>assumes: CP =IP]s, [>assumes: VP =DP]s, [>assumes: VP =CP]s, [>assumes: PP
=DP]s, [>assumes: VP =DP +O]s, [>assumes: PP =DP +O]s, [>assumes: VP =CP +Oc]s,
[>assumes: CP =IP +Comp -Oc]s]
```

Initial energy:

```
Temperature: 100.000000
Energy: 10295
Grammar: 5144
Input: 5151
```

After 100 iterations:

```
Iteration: 100
2017_05_10__16_10_31: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [<Jerry: DP]s, [<Jerry: VP]s, [<Jerry: VP =CP]s, [<Jerry:
PP =DP]s, [<Jerry: CP =IP]s, [<Jerry: PP =DP +O]s, [<Jerry: VP =CP +Oc]s, [<Jerry: CP
=IP +Comp -Oc]s, [<George: DP]s, [<George: VP]s, [<George: CP =IP]s, [<George: VP
=DP]s, [<George: VP =DP +O]s, [<George: PP =DP +O]s, [<George: VP =CP +Oc]s, [<George:
CP =IP +Comp -Oc]s, [<Elaine: DP]s, [<Elaine: VP =DP]s, [<Elaine: VP =CP]s, [<Elaine:
CP =IP]s, [<Elaine: VP =DP +O]s, [<Elaine: PP =DP +O]s, [<Elaine: VP =CP +Oc]s,
[<Elaine: CP =IP +Comp -Oc]s, [<Kramer: DP]s, [<Kramer: VP =DP]s, [<Kramer: VP =CP]s,
[<Kramer: PP =DP]s, [<Kramer: CP =IP]s, [<Kramer: VP =DP +O]s, [<Kramer: PP =DP +O]s,
[<Kramer: VP =CP +Oc]s, [<ran: DP]s, [<ran: VP]s, [<ran: VP =CP]s, [<ran: PP =DP]s,
[<ran: VP =DP]s, [<ran: PP =DP +O]s, [<ran: VP =CP +Oc]s, [<ran: CP =IP +Comp -Oc]s,
[<walked: DP]s, [<walked: VP]s, [<walked: VP =CP]s, [<walked: VP =DP +O]s, [<walked:
VP =CP +Oc]s, [<read: DP]s, [<read: VP]s, [<read: VP =DP]s, [<read: VP =DP +O]s,
[<read: PP =DP +O]s, [<read: CP =IP +Comp -Oc]s, [<wrote: VP]s, [<wrote: CP =IP]s,
[<wrote: VP =CP]s, [<wrote: PP =DP]s, [<wrote: VP =CP +Oc]s, [<wrote: CP =IP +Comp -
Oc]s, [<liked: DP]s, [<liked: VP]s, [<liked: VP =CP]s, [<liked: PP =DP]s, [<liked: VP
=DP +O]s, [<liked: VP =CP +Oc]s, [<saw: VP]s, [<saw: CP =IP]s, [<saw: VP =CP]s, [<saw:
PP =DP]s, [<saw: VP =DP +O]s, [<saw: PP =DP +O]s, [<saw: VP =CP +Oc]s, [<saw: CP =IP
+Comp -Oc]s, [<loved: DP]s, [<loved: VP]s, [<loved: CP =IP]s, [<loved: VP =DP +O]s,
[<loved: PP =DP +O]s, [<loved: VP =CP +Oc]s, [<loved: CP =IP +Comp -Oc]s, [<hated:
VP]s, [<hated: DP]s, [<hated: CP =IP]s, [<hated: VP =DP]s, [<hated: VP =CP]s, [<hated:
PP =DP]s, [<with: DP]s, [<with: VP]s, [<with: VP =DP]s, [<with: VP =CP]s, [<with: PP
=DP]s, [<with: VP =DP +O]s, [<with: PP =DP +O]s, [<with: VP =CP +Oc]s, [<with: CP =IP
+Comp -Oc]s, [<by: DP]s, [<by: VP]s, [<by: VP =CP]s, [<by: PP =DP]s, [<by: VP =DP
+O]s, [<by: PP =DP +O]s, [<by: VP =CP +Oc]s, [<by: CP =IP +Comp -Oc]s, [<above: DP]s,
[<above: VP]s, [<above: CP =IP]s, [<above: PP =DP]s, [<above: VP =DP +O]s, [<above: PP
=DP +O]s, [<above: VP =CP +Oc]s, [<under: DP]s, [<under: VP =DP]s, [<under: VP =CP]s,
[<under: PP =DP]s, [<under: VP =DP +O]s, [<under: PP =DP +O]s, [<under: VP =CP +Oc]s,
[<under: CP =IP +Comp -Oc]s, [<that: DP]s, [<that: VP]s, [<that: VP =DP]s, [<that: PP
=DP]s, [<that: VP =DP +O]s, [<that: PP =DP +O]s, [<that: CP =IP +Comp -Oc]s, [<knows:
VP]s, [<knows: DP]s, [<knows: CP =IP]s, [<knows: VP =DP]s, [<knows: VP =CP]s, [<knows:
PP =DP]s, [<knows: VP =DP +O]s, [<knows: PP =DP +O]s, [<knows: VP =CP +Oc]s, [<knows:
CP =IP +Comp -Oc]s, [<says: DP]s, [<says: CP =IP]s, [<says: VP =CP]s, [<says: PP =DP]s,
[<says: PP =DP +O]s, [<says: VP =CP +Oc]s, [<says: CP =IP +Comp -Oc]s, [<thinks: VP]s,
[<thinks: DP]s, [<thinks: CP =IP]s, [<thinks: VP =DP]s, [<thinks: VP =CP]s, [<thinks:
PP =DP]s, [<thinks: PP =DP +O]s, [<thinks: VP =CP +Oc]s, [<assumes: DP]s, [<assumes: VP]s,
```

```
[<assumes: CP =IP]s, [<assumes: VP =DP]s, [<assumes: VP =CP]s, [<assumes: PP =DP]s,
[<assumes: PP =DP +O]s]2017_05_10__16_10_31: Energy: 9117
2017_05_10__16_10_31: Grammar: 3766
2017_05_10__16_10_31: Input: 5351
2017_05_10__16_10_31: Temperature: 74.048426
```

After 300 iterations:

```
Iteration: 300
2017 05 10  16 33 08: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: DP]s, [>Jerry: VP]s, [>Jerry: PP =DP]s, [>Jerry:
PP =DP +O]s, [>Jerry: CP =IP +Comp -Oc]s, [>George: DP]s, [>George: VP]s, [>George: VP
=CP]s, [>George: PP =DP]s, [>George: VP =DP +O]s, [>Elaine: DP]s, [>Elaine: PP =DP]s,
[>Kramer: DP]s, [>Kramer: PP =DP]s, [>Kramer: PP =DP +O]s, [>Kramer: VP =DP +O]s,
[>Kramer: VP =CP +Oc]s, [>ran: VP]s, [>ran: DP]s, [>walked: DP]s, [>walked: VP]s,
[>walked: VP =CP]s, [>walked: VP =DP]s, [>read: VP]s, [>read: VP =DP]s, [>read: PP =DP
+O]s, [>wrote: VP]s, [>wrote: VP =CP]s, [>liked: DP]s, [>liked: VP =DP]s, [>liked: CP
=IP]s, [>saw: VP =DP]s, [>saw: VP =CP]s, [>saw: PP =DP +O]s, [>loved: VP]s, [>loved:
VP =DP]s, [>loved: PP =DP +O]s, [>loved: CP =IP +Comp -Oc]s, [>hated: VP]s, [>hated:
DP]s, [>hated: VP =DP]s, [>hated: PP =DP +O]s, [>hated: CP =IP +Comp -Oc]s, [>with:
DP]s, [>with: DP]s, [>with: VP =DP]s, [>by: DP]s, [>by: PP =DP]s, [>above: DP]s,
[>above: VP =CP]s, [>above: CP =IP +Comp -Oc]s, [>under: DP]s, [>under: CP =IP]s,
[>under: VP =CP +Oc]s, [>that: VP]s, [>that: CP =IP]s, [>that: VP =DP +O]s, [>that: PP
=DP +O]s, [>knows: VP]s, [>knows: VP =CP]s, [>knows: PP =DP]s, [>knows: VP =DP +O]s,
[>says: VP =CP]s, [>thinks: VP =CP]s, [>thinks: PP =DP +O]s, [>assumes: PP =DP]s,
[>assumes: VP =CP]s, [>assumes: VP =CP +Oc]s]
2017_05_10__16_33_08: Energy: 6190
2017_05_10__16_33_08: Grammar: 1645
2017_05_10__16_33_08: Input: 4545
2017_05_10__16_33_08: Temperature: 40.602006
```

After over 500 iterations:

```
Iteration: 546
2017_05_10__16_33_52: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>Jerry: PP =DP]s,
[>George: DP]s, [>George: PP =DP]s,
[>Elaine: DP]s, [>Elaine: PP =DP]s,
[>Kramer: DP]s, [>Kramer: PP =DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s, [>loved: VP =DP]s, [>hated: VP =DP]s,
[>with: DP]s, [>by: DP]s, [>above: DP]s, [>under: DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>thinks: VP =CP]s, [>assumes: VP =CP]s]
2017_05_10__16_33_52: Energy: 4435
2017_05_10__16_33_52: Grammar: 616
2017_05_10__16_33_52: Input: 3819
2017_05_10__16_33_52: Temperature: 19.389046
```

Mixed Word Input

The relevant log file for this run is:

log_2017_05_10__20_19_58 - Language, Mixed-word, PP & CP.txt

The generated input:

```
Input is: ['Jerry read Elaine with', 'George saw Elaine', 'Kramer knows that George
saw Jerry', 'George Elaine loved', 'George Elaine hated', 'Elaine read Kramer above',
'Elaine knows that Jerry walked', 'Kramer knows that George read Kramer by', 'Jerry
thinks that Jerry walked Kramer under', 'Jerry thinks that Kramer ran Kramer with',
'Elaine Elaine hated', 'Elaine wrote Elaine with', 'George says that Kramer ran Kramer
under', 'Elaine George hated', 'George walked', 'George ran Kramer by', 'Kramer says
that Kramer read Jerry by', 'Elaine ran', 'George assumes that Jerry Kramer hated
Kramer above', 'Elaine liked George George above', 'Elaine Jerry hated', 'Kramer
```

```
read', 'George read Elaine above', 'Kramer liked Elaine', 'Jerry Elaine loved Elaine
above', 'George saw Elaine', 'George walked Elaine above', 'Elaine wrote George
under', 'Kramer George hated', 'Kramer liked Kramer', 'Kramer read', 'George assumes
that Elaine liked Kramer', 'Elaine saw Kramer Elaine above', 'George George hated',
'Kramer ran', 'Elaine Kramer loved', 'Kramer George loved Jerry under', 'Jerry knows
that Jerry George loved', 'Elaine thinks that Elaine walked Elaine with', 'Elaine says
that George read', 'Jerry says that Kramer George loved Kramer above', 'George wrote
Kramer by', 'Kramer ran George with', 'George thinks that George wrote Elaine above',
'Jerry liked Jerry Kramer by', 'Jerry read', 'George wrote', 'Elaine George loved
George under', 'George read Kramer under', 'Jerry says that Jerry ran']
```

The initial hypothesis:

```
Initial hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP
=IP -Comp]s, [>Jerry: DP]s, [>Jerry: VP]s, [>Jerry: CP =IP]s, [>Jerry: VP =DP]s,
[>Jerry: VP =CP]s, [>Jerry: PP =DP]s, [>Jerry: VP =DP +O]s, [>Jerry: PP =DP +O]s,
[>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s, [>George: VP]s, [>George: DP]s,
[>George: CP =IP]s, [>George: VP =DP]s, [>George: VP =CP]s, [>George: PP =DP]s,
[>George: VP =DP +O]s, [>George: PP =DP +O]s, [>George: VP =CP +Oc]s, [>George: CP =IP
+Comp -Oc]s, [>Elaine: VP]s, [>Elaine: DP]s, [>Elaine: CP =IP]s, [>Elaine: VP =DP]s,
[>Elaine: VP =CP]s, [>Elaine: PP =DP]s, [>Elaine: VP =DP +O]s, [>Elaine: PP =DP +O]s,
[>Elaine: VP =CP +Oc]s, [>Elaine: CP =IP +Comp -Oc]s, [>Kramer: VP]s, [>Kramer: DP]s,
[>Kramer: CP =IP]s, [>Kramer: VP =DP]s, [>Kramer: VP =CP]s, [>Kramer: PP =DP]s,
[>Kramer: VP =DP +O]s, [>Kramer: PP =DP +O]s, [>Kramer: VP =CP +Oc]s, [>Kramer: CP =IP
+Comp -Oc]s, [>ran: VP]s, [>ran: DP]s, [>ran: CP =IP]s, [>ran: VP =DP]s, [>ran: VP
=CP]s, [>ran: PP =DP]s, [>ran: VP =DP +O]s, [>ran: PP =DP +O]s, [>ran: VP =CP +Oc]s,
[>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [>walked: DP]s, [>walked: CP =IP]s,
[>walked: VP =DP]s, [>walked: VP =CP]s, [>walked: PP =DP]s, [>walked: VP =DP +O]s,
[>walked: PP =DP +O]s, [>walked: VP =CP +Oc]s, [>walked: CP =IP +Comp -Oc]s, [>read:
DP]s, [>read: VP]s, [>read: CP =IP]s, [>read: VP =DP]s, [>read: VP =CP]s, [>read: PP
=DP]s, [>read: VP =DP +O]s, [>read: PP =DP +O]s, [>read: VP =CP +Oc]s, [>read: CP =IP
+Comp -Oc]s, [>wrote: VP]s, [>wrote: DP]s, [>wrote: CP =IP]s, [>wrote: VP =DP]s,
[>wrote: VP =CP]s, [>wrote: PP =DP]s, [>wrote: VP =DP +O]s, [>wrote: PP =DP +O]s,
[>wrote: VP =CP +Oc]s, [>wrote: CP =IP +Comp -Oc]s, [>liked: VP]s, [>liked: DP]s,
[>liked: CP =IP]s, [>liked: VP =DP]s, [>liked: VP =CP]s, [>liked: PP =DP]s, [>liked:
VP =DP +O]s, [>liked: PP =DP +O]s, [>liked: VP =CP +Oc]s, [>liked: CP =IP +Comp -Oc]s,
[>saw: VP]s, [>saw: DP]s, [>saw: CP =IP]s, [>saw: VP =DP]s, [>saw: VP =CP]s, [>saw: PP
=DP]s, [>saw: VP =DP +O]s, [>saw: PP =DP +O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP
+Comp -Oc]s, [>loved: VP]s, [>loved: DP]s, [>loved: CP =IP]s, [>loved: VP =DP]s,
[>loved: VP =CP]s, [>loved: PP =DP]s, [>loved: VP =DP +O]s, [>loved: PP =DP +O]s,
[>loved: VP =CP +Oc]s, [>loved: CP =IP +Comp -Oc]s, [>hated: VP]s, [>hated: DP]s,
[>hated: CP =IP]s, [>hated: VP =DP]s, [>hated: VP =CP]s, [>hated: PP =DP]s, [>hated:
VP =DP +O]s, [>hated: PP =DP +O]s, [>hated: VP =CP +Oc]s, [>hated: CP =IP +Comp -Oc]s,
[>with: VP]s, [>with: DP]s, [>with: CP =IP]s, [>with: VP =DP]s, [>with: VP =CP]s,
[>with: PP =DP]s, [>with: VP =DP +O]s, [>with: PP =DP +O]s, [>with: VP =CP +Oc]s,
[>with: CP =IP +Comp -Oc]s, [>by: VP]s, [>by: DP]s, [>by: CP =IP]s, [>by: VP =DP]s,
[>by: VP =CP]s, [>by: PP =DP]s, [>by: VP =DP +O]s, [>by: PP =DP +O]s, [>by: VP =CP
+Oc]s, [>by: CP =IP +Comp -Oc]s, [>above: VP]s, [>above: DP]s, [>above: CP =IP]s,
[>above: VP =DP]s, [>above: VP =CP]s, [>above: PP =DP]s, [>above: VP =DP +O]s,
[>above: PP =DP +O]s, [>above: VP =CP +Oc]s, [>above: CP =IP +Comp -Oc]s, [>under:
VP]s, [>under: DP]s, [>under: CP =IP]s, [>under: VP =DP]s, [>under: VP =CP]s, [>under:
PP =DP]s, [>under: VP =DP +O]s, [>under: PP =DP +O]s, [>under: VP =CP +Oc]s, [>under:
CP =IP +Comp -Oc]s, [>that: VP]s, [>that: DP]s, [>that: CP =IP]s, [>that: VP =DP]s,
[>that: VP =CP]s, [>that: PP =DP]s, [>that: VP =DP +O]s, [>that: PP =DP +O]s, [>that:
VP =CP +Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP]s, [>knows: DP]s, [>knows: CP
=IP]s, [>knows: VP =DP]s, [>knows: VP =CP]s, [>knows: PP =DP]s, [>knows: VP =DP +O]s,
[>knows: PP =DP +O]s, [>knows: VP =CP +Oc]s, [>knows: CP =IP +Comp -Oc]s, [>says:
VP]s, [>says: DP]s, [>says: CP =IP]s, [>says: VP =DP]s, [>says: VP =CP]s, [>says: PP
=DP]s, [>says: VP =DP +O]s, [>says: PP =DP +O]s, [>says: VP =CP +Oc]s, [>says: CP =IP
+Comp -Oc]s, [>thinks: VP]s, [>thinks: DP]s, [>thinks: CP =IP]s, [>thinks: VP =DP]s,
[>thinks: VP =CP]s, [>thinks: PP =DP]s, [>thinks: VP =DP +O]s, [>thinks: PP =DP +O]s,
[>thinks: VP =CP +Oc]s, [>thinks: CP =IP +Comp -Oc]s, [>assumes: VP]s, [>assumes:
DP]s, [>assumes: CP =IP]s, [>assumes: VP =DP]s, [>assumes: VP =CP]s, [>assumes: PP
=DP]s, [>assumes: VP =DP +O]s, [>assumes: PP =DP +O]s, [>assumes: VP =CP +Oc]s,
[>assumes: CP =IP +Comp -Oc]s]
```

Initial energy:

```
Temperature: 100.000000
Energy: 10380
Grammar: 5144
Input: 5236
```

After 100 iterations:

```
Iteration: 100
2017_05_10__21_01_07: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: DP]s, [>Jerry: VP]s, [>Jerry: CP =IP]s, [>Jerry:
VP =DP]s, [>Jerry: VP =CP]s, [>Jerry: PP =DP]s, [>Jerry: PP =DP +O]s, [>Jerry: VP =CP
+Oc]s, [>George: DP]s, [>George: CP =IP]s, [>George: VP =DP]s, [>George: VP =CP]s,
[>George: PP =DP]s, [>George: VP =DP +O]s, [>Elaine: DP]s, [>Elaine: CP =IP]s,
[>Elaine: VP =DP]s, [>Elaine: PP =DP]s, [>Elaine: VP =DP +O]s, [>Elaine: VP =CP +Oc]s,
[>Elaine: CP =IP +Comp -Oc]s, [>Kramer: DP]s, [>Kramer: VP =DP]s, [>Kramer: PP =DP]s,
[>Kramer: VP =DP +O]s, [>Kramer: VP =CP +Oc]s, [>Kramer: CP =IP +Comp -Oc]s, [>ran:
VP]s, [>ran: DP]s, [>ran: CP =IP]s, [>ran: VP =DP]s, [>ran: VP =DP +O]s, [>walked:
VP]s, [>walked: CP =IP]s, [>walked: VP =DP]s, [>walked: PP =DP]s, [>walked: PP =DP
+O]s, [>walked: VP =CP +Oc]s, [>walked: CP =IP +Comp -Oc]s, [>read: VP]s, [>read:
DP]s, [>read: CP =IP]s, [>read: VP =DP]s, [>read: PP =DP]s, [>read: VP =DP +O]s,
[>read: PP =DP +O]s, [>read: VP =CP +Oc]s, [>wrote: VP]s, [>wrote: DP]s, [>wrote: CP
=IP]s, [>wrote: PP =DP]s, [>wrote: VP =CP]s, [>wrote: VP =DP +O]s, [>wrote: VP =CP
+Oc]s, [>liked: VP]s, [>liked: CP =IP]s, [>liked: VP =DP]s, [>liked: VP =CP]s,
[>liked: PP =DP]s, [>liked: VP =DP +O]s, [>liked: PP =DP +O]s, [>liked: VP =CP +Oc]s,
[>liked: CP =IP +Comp -Oc]s, [>saw: DP]s, [>saw: CP =IP]s, [>saw: VP =DP]s, [>saw: VP
=CP]s, [>saw: CP =IP +Comp -Oc]s, [>loved: VP]s, [>loved: DP]s, [>loved: CP =IP]s,
[>loved: VP =DP]s, [>loved: VP =CP]s, [>loved: VP =DP +O]s, [>loved: VP =CP +Oc]s,
[>loved: CP =IP +Comp -Oc]s, [>hated: VP]s, [>hated: DP]s, [>hated: CP =IP]s, [>hated:
VP =DP]s, [>hated: VP =CP]s, [>hated: PP =DP]s, [>hated: VP =DP +O]s, [>hated: PP =DP
+O]s, [>hated: VP =CP +Oc]s, [>hated: CP =IP +Comp -Oc]s, [>with: DP]s, [>with: CP
=IP]s, [>with: VP =CP]s, [>with: VP =CP +Oc]s, [>by: DP]s, [>by: CP =IP]s, [>by: VP
=DP]s, [>by: VP =CP]s, [>by: VP =DP +O]s, [>by: CP =IP +Comp -Oc]s, [>above: VP]s,
[>above: DP]s, [>above: VP =CP]s, [>above: PP =DP]s, [>above: VP =DP +O]s, [>above: PP
=DP +O]s, [>above: VP =CP +Oc]s, [>above: CP =IP +Comp -Oc]s, [>under: VP]s, [>under:
CP =IP]s, [>under: VP =CP]s, [>under: PP =DP]s, [>under: VP =DP +O]s, [>under: PP =DP
+O]s, [>under: VP =CP +Oc]s, [>that: DP]s, [>that: CP =IP]s, [>that: VP =DP]s, [>that:
VP =CP]s, [>that: VP =DP +O]s, [>that: PP =DP +O]s, [>that: VP =CP +Oc]s, [>that: CP
=IP +Comp -Oc]s, [>knows: VP]s, [>knows: DP]s, [>knows: CP =IP]s, [>knows: VP =DP]s,
[>knows: VP =CP]s, [>knows: VP =DP +O]s, [>knows: PP =DP +O]s, [>knows: VP =CP +Oc]s,
[>says: VP]s, [>says: DP]s, [>says: CP =IP]s, [>says: VP =DP]s, [>says: VP =CP]s,
[>says: VP =CP +Oc]s, [>thinks: VP]s, [>thinks: CP =IP]s, [>thinks: VP =CP]s,
[>thinks: VP =DP +O]s, [>thinks: PP =DP +O]s, [>thinks: VP =CP +Oc]s, [>thinks: CP =IP
+Comp -Oc]s, [>assumes: VP]s, [>assumes: DP]s, [>assumes: CP =IP]s, [>assumes: VP
=DP]s, [>assumes: PP =DP]s, [>assumes: VP =DP +O]s, [>assumes: PP =DP +O]s, [>assumes:
VP =CP +Oc]s]
2017_05_10__21_01_07: Energy: 9027
2017_05_10__21_01_07: Grammar: 3617
2017_05_10__21_01_07: Input: 5410
2017_05_10__21_01_07: Temperature: 74.048426
```

After 300 iterations:

```
Iteration: 300
2017_05_10__21_08_42: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: DP]s, [>Jerry: VP]s, [>Jerry: VP =DP]s, [>Jerry:
VP =CP]s, [>Jerry: PP =DP]s, [>Jerry: VP =CP +Oc]s, [>George: DP]s, [>George: CP
=IP]s, [>George: VP =DP]s, [>George: PP =DP]s, [>Elaine: DP]s, [>Elaine: VP =DP]s,
[>Elaine: PP =DP]s, [>Elaine: VP =DP +O]s, [>Elaine: VP =CP +Oc]s, [>Kramer: DP]s,
[>Kramer: VP =DP]s, [>Kramer: PP =DP]s, [>ran: VP]s, [>ran: VP =DP]s, [>walked: VP]s,
[>walked: PP =DP +O]s, [>walked: VP =CP +Oc]s, [>read: VP]s, [>read: CP =IP]s,
[>wrote: VP]s, [>wrote: VP =CP]s, [>wrote: VP =DP]s, [>liked: VP =DP]s, [>liked: PP
=DP]s, [>liked: VP =DP +O]s, [>liked: PP =DP +O]s, [>saw: VP =DP]s, [>loved: DP]s,
[>loved: PP =DP]s, [>loved: VP =CP +Oc]s, [>hated: DP]s, [>hated: VP =CP]s, [>hated:
VP =DP]s, [>with: DP]s, [>with: VP =CP]s, [>by: DP]s, [>by: VP =DP +O]s, [>by: VP =CP
+Oc]s, [>above: DP]s, [>under: VP]s, [>under: PP =DP]s, [>under: PP =DP +O]s, [>that:
CP =IP]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP =CP]s, [>knows: VP =CP +Oc]s,
[>says: VP =DP]s, [>says: VP =CP]s, [>thinks: VP]s, [>thinks: DP]s, [>thinks: VP
=CP]s, [>thinks: VP =DP]s, [>thinks: PP =DP]s, [>assumes: VP =CP]s, [>assumes: VP =DP
+O]s]
2017_05_10__21_08_42: Energy: 6240
2017_05_10__21_08_42: Grammar: 1482
2017_05_10__21_08_42: Input: 4758
2017_05_10__21_08_42: Temperature: 40.602006
```

After over 1000 iterations:

```
Iteration: 1437
2017_05_10__21_15_20: Hypothesis: [
```

```
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>Jerry: VP =DP]s, [>Jerry: PP =DP]s,
[>George: DP]s, [>George: VP =DP]s, [>George: PP =DP]s,
[>Elaine: DP]s, [>Elaine: VP =DP]s, [>Elaine: PP =DP]s,
[>Kramer: DP]s, [>Kramer: VP =DP]s, [>Kramer: PP =DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s,
[>loved: DP]s, [>hated: DP]s,
[>with: DP]s, [>by: DP]s, [>above: DP]s, [>under: DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>thinks: VP =CP]s, [>assumes: VP =CP]s]
2017_05_10__21_15_20: Energy: 4702
2017_05_10__21_15_20: Grammar: 698
2017_05_10__21_15_20: Input: 4004
2017_05_10__21_15_20: Temperature: 1.333345
```

Category Learner

Head Initial Input

The relevant log file for this run is:

log_2017_05_10__21_17_43 - Category, Head-initial, PP & CP.txt

The generated input:

```
['Elaine ran', 'George walked', 'Kramer walked with Elaine', 'Elaine wrote with
George', 'Jerry says that Jerry ran under Elaine', 'Kramer loved Kramer', 'Elaine
loved Kramer by Jerry', 'George walked', 'Kramer liked George', 'Kramer wrote',
'George read', 'Jerry loved Kramer', 'Jerry read', 'Elaine read', 'Jerry assumes that
George saw Elaine by Elaine', 'George knows that George wrote by Elaine', 'Kramer
hated Elaine', 'Kramer ran by Jerry', 'Jerry liked George', 'Elaine walked', 'Jerry
read with Elaine', 'Kramer assumes that George walked', 'Elaine hated Jerry above
Kramer', 'George wrote', 'Jerry read by George', 'Jerry saw Elaine under Jerry',
'Kramer liked Jerry', 'Elaine walked', 'George loved Jerry', 'Elaine saw Jerry by
Elaine', 'Kramer loved Kramer by Kramer', 'Jerry walked above Jerry', 'Kramer hated
Jerry', 'Jerry ran by Elaine', 'George walked with Jerry', 'Kramer liked George',
'George liked Elaine with George', 'Jerry loved Jerry with George', 'Jerry read with
Jerry', 'Elaine ran', 'George wrote', 'George ran', 'Kramer wrote', 'Jerry saw Jerry
by George', 'George saw Elaine', 'Elaine liked Elaine above Kramer', 'George liked
Elaine with George', 'George says that George saw Kramer above Elaine', 'Elaine
wrote', 'George hated Elaine']
```

The initial hypothesis:

```
Initial hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP
=IP -Comp]s, [>Jerry: VP]s, [>Jerry: DP]s, [>Jerry: CP =IP]s, [>Jerry: VP =DP]s,
[>Jerry: VP =CP]s, [>Jerry: PP =DP]s, [>Jerry: VP =DP +O]s, [>Jerry: PP =DP +O]s,
[>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s, [>George: VP]s, [>George: DP]s,
[>George: CP =IP]s, [>George: VP =DP]s, [>George: VP =CP]s, [>George: PP =DP]s,
[>George: VP =DP +O]s, [>George: PP =DP +O]s, [>George: VP =CP +Oc]s, [>George: CP =IP
+Comp -Oc]s, [>Elaine: VP]s, [>Elaine: DP]s, [>Elaine: CP =IP]s, [>Elaine: VP =DP]s,
[>Elaine: VP =CP]s, [>Elaine: PP =DP]s, [>Elaine: VP =DP +O]s, [>Elaine: PP =DP +O]s,
[>Elaine: VP =CP +Oc]s, [>Elaine: CP =IP +Comp -Oc]s, [>Kramer: VP]s, [>Kramer: DP]s,
[>Kramer: CP =IP]s, [>Kramer: VP =DP]s, [>Kramer: VP =CP]s, [>Kramer: PP =DP]s,
[>Kramer: VP =DP +O]s, [>Kramer: PP =DP +O]s, [>Kramer: VP =CP +Oc]s, [>Kramer: CP =IP
+Comp -Oc]s, [>ran: VP]s, [>ran: DP]s, [>ran: CP =IP]s, [>ran: VP =DP]s, [>ran: VP
=CP]s, [>ran: PP =DP]s, [>ran: VP =DP +O]s, [>ran: PP =DP +O]s, [>ran: VP =CP +Oc]s,
[>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [>walked: DP]s, [>walked: CP =IP]s,
[>walked: VP =DP]s, [>walked: VP =CP]s, [>walked: PP =DP]s, [>walked: VP =DP +O]s,
[>walked: PP =DP +O]s, [>walked: VP =CP +Oc]s, [>walked: CP =IP +Comp -Oc]s, [>read:
VP]s, [>read: DP]s, [>read: CP =IP]s, [>read: VP =DP]s, [>read: VP =CP]s, [>read: PP
=DP]s, [>read: VP =DP +O]s, [>read: PP =DP +O]s, [>read: VP =CP +Oc]s, [>read: CP =IP
+Comp -Oc]s, [>wrote: VP]s, [>wrote: DP]s, [>wrote: CP =IP]s, [>wrote: VP =DP]s,
[>wrote: VP =CP]s, [>wrote: PP =DP]s, [>wrote: VP =DP +O]s, [>wrote: PP =DP +O]s,
```

```
[>wrote: VP =CP +Oc]s, [>wrote: CP =IP +Comp -Oc]s, [>liked: VP]s, [>liked: DP]s,
[>liked: CP =IP]s, [>liked: VP =DP]s, [>liked: VP =CP]s, [>liked: PP =DP]s, [>liked:
VP =DP +O]s, [>liked: PP =DP +O]s, [>liked: VP =CP +Oc]s, [>liked: CP =IP +Comp -Oc]s,
[>saw: VP]s, [>saw: DP]s, [>saw: CP =IP]s, [>saw: VP =DP]s, [>saw: VP =CP]s, [>saw: PP
=DP]s, [>saw: VP =DP +O]s, [>saw: PP =DP +O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP
+Comp -Oc]s, [>loved: VP]s, [>loved: DP]s, [>loved: CP =IP]s, [>loved: VP =DP]s,
[>loved: VP =CP]s, [>loved: PP =DP]s, [>loved: VP =DP +O]s, [>loved: PP =DP +O]s,
[>loved: VP =CP +Oc]s, [>loved: CP =IP +Comp -Oc]s, [>hated: VP]s, [>hated: DP]s,
[>hated: CP =IP]s, [>hated: VP =DP]s, [>hated: VP =CP]s, [>hated: PP =DP]s, [>hated:
VP =DP +O]s, [>hated: PP =DP +O]s, [>hated: VP =CP +Oc]s, [>hated: CP =IP +Comp -Oc]s,
[>with: VP]s, [>with: DP]s, [>with: CP =IP]s, [>with: VP =DP]s, [>with: VP =CP]s,
[>with: PP =DP]s, [>with: VP =DP +O]s, [>with: PP =DP +O]s, [>with: VP =CP +Oc]s,
[>with: CP =IP +Comp -Oc]s, [>by: VP]s, [>by: DP]s, [>by: CP =IP]s, [>by: VP =DP]s,
[>by: VP =CP]s, [>by: PP =DP]s, [>by: VP =DP +O]s, [>by: PP =DP +O]s, [>by: VP =CP
+Oc]s, [>by: CP =IP +Comp -Oc]s, [>above: VP]s, [>above: DP]s, [>above: CP =IP]s,
[>above: VP =DP]s, [>above: VP =CP]s, [>above: PP =DP]s, [>above: VP =DP +O]s,
[>above: PP =DP +O]s, [>above: VP =CP +Oc]s, [>above: CP =IP +Comp -Oc]s, [>under:
VP]s, [>under: DP]s, [>under: CP =IP]s, [>under: VP =DP]s, [>under: VP =CP]s, [>under:
PP =DP]s, [>under: VP =DP +O]s, [>under: PP =DP +O]s, [>under: VP =CP +Oc]s, [>under:
CP =IP +Comp -Oc]s, [>that: VP]s, [>that: DP]s, [>that: CP =IP]s, [>that: VP =DP]s,
[>that: VP =CP]s, [>that: PP =DP]s, [>that: VP =DP +O]s, [>that: PP =DP +O]s, [>that:
VP =CP +Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP]s, [>knows: DP]s, [>knows: CP
=IP]s, [>knows: VP =DP]s, [>knows: VP =CP]s, [>knows: PP =DP]s, [>knows: VP =DP +O]s,
[>knows: PP =DP +O]s, [>knows: VP =CP +Oc]s, [>knows: CP =IP +Comp -Oc]s, [>says:
VP]s, [>says: DP]s, [>says: CP =IP]s, [>says: VP =DP]s, [>says: VP =CP]s, [>says: PP
=DP]s, [>says: VP =DP +O]s, [>says: PP =DP +O]s, [>says: VP =CP +Oc]s, [>says: CP =IP
+Comp -Oc]s, [>assumes: VP]s, [>assumes: DP]s, [>assumes: CP =IP]s, [>assumes: VP
=DP]s, [>assumes: VP =CP]s, [>assumes: PP =DP]s, [>assumes: VP =DP +O]s, [>assumes: PP
=DP +O]s, [>assumes: VP =CP +Oc]s, [>assumes: CP =IP +Comp -Oc]s]
```

Initial energy:

```
Temperature: 100.000000
Energy: 9362
Grammar: 4908
Input: 4454
```

After 100 iterations:

```
Iteration: 100
2017_05_10__21_37_13: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: VP]s, [<Jerry: DP]s, [>Jerry: VP =CP]s, [>Jerry:
PP =DP]s, [>Jerry: PP =DP +O]s, [>George: VP]s, [<George: DP]s, [>George: VP =DP]s,
[>George: PP =DP]s, [<George: CP =IP]s, [>George: VP =CP]s, [>George: VP =DP +O]s,
[>George: PP =DP +O]s, [>George: VP =CP +Oc]s, [>Elaine: VP]s, [<Elaine: DP]s,
[>Elaine: PP =DP]s, [<Elaine: CP =IP]s, [>Elaine: PP =DP +O]s, [>Elaine: VP =CP +Oc]s,
[<Kramer: DP]s, [>Kramer: VP]s, [>Kramer: VP =DP]s, [>Kramer: VP =CP]s, [>Kramer: PP
=DP]s, [<Kramer: CP =IP]s, [>Kramer: VP =DP +O]s, [<Kramer: CP =IP +Comp -Oc]s, [>ran:
VP]s, [>ran: VP =CP]s, [>ran: PP =DP]s, [<ran: CP =IP]s, [>ran: PP =DP +O]s, [>ran: VP
=CP +Oc]s, [>walked: VP]s, [<walked: DP]s, [>walked: PP =DP]s, [>walked: VP =CP]s,
[>walked: VP =DP +O]s, [>walked: VP =CP +Oc]s, [<walked: CP =IP +Comp -Oc]s, [>read:
VP]s, [<read: DP]s, [>read: VP =DP]s, [>read: VP =CP]s, [<read: CP =IP]s, [<read: CP
=IP]s, [>read: VP =DP +O]s, [>read: PP =DP +O]s, [>read: VP =CP +Oc]s, [>wrote: VP]s,
[<wrote: DP]s, [>wrote: VP =DP]s, [>wrote: PP =DP]s, [<wrote: CP =IP]s, [>wrote: VP
=CP]s, [>wrote: VP =DP +O]s, [>wrote: PP =DP +O]s, [>wrote: VP =CP +Oc]s, [<wrote: CP
=IP +Comp -Oc]s, [>liked: VP]s, [<liked: DP]s, [>liked: VP =DP]s, [>liked: VP =DP
+O]s, [>liked: PP =DP +O]s, [<saw: DP]s, [>saw: VP =DP]s, [>saw: VP =CP]s, [>saw: VP
=CP +Oc]s, [<loved: DP]s, [>loved: VP =DP]s, [>loved: VP =CP]s, [>loved: PP =DP]s,
[<loved: CP =IP]s, [>loved: VP =DP +O]s, [<loved: CP =IP +Comp -Oc]s, [>hated: VP]s,
[<hated: DP]s, [>hated: VP =DP]s, [<hated: CP =IP]s, [>hated: VP =DP +O]s, [>hated: PP
=DP +O]s, [>hated: VP =CP +Oc]s, [<hated: CP =IP +Comp -Oc]s, [>with: VP]s, [>with: PP
=DP]s, [>with: VP =CP +Oc]s, [>by: VP =DP]s, [>by: VP =CP]s, [>by: PP =DP]s, [<by: CP
=IP]s, [>by: VP =DP +O]s, [>by: VP =CP +Oc]s, [>above: VP]s, [<above: DP]s, [>above:
VP =DP]s, [>above: VP =CP]s, [>above: PP =DP]s, [>above: PP =DP +O]s, [>above: VP =CP
+Oc]s, [<above: CP =IP +Comp -Oc]s, [>under: VP =DP]s, [>under: PP =DP]s, [>under: VP
=DP]s, [>under: VP =DP +O]s, [>under: VP =CP +Oc]s, [<under: CP =IP +Comp -Oc]s,
[>that: VP]s, [<that: DP]s, [>that: VP =DP]s, [>that: VP =CP]s, [>that: PP =DP]s,
[>that: VP =DP +O]s, [>that: VP =CP +Oc]s, [<that: CP =IP +Comp -Oc]s, [<knows: DP]s,
[>knows: PP =DP]s, [<knows: CP =IP]s, [>knows: VP =DP +O]s, [>knows: PP =DP +O]s,
[>knows: VP =CP +Oc]s, [<knows: CP =IP +Comp -Oc]s, [>says: VP]s, [>says: VP =CP]s,
[>says: PP =DP]s, [>says: VP =DP +O]s, [>says: PP =DP +O]s, [>assumes: VP]s,
[>assumes: VP =CP]s, [>assumes: PP =DP]s, [<assumes: CP =IP]s, [>assumes: PP =DP +O]s,
[>assumes: VP =CP +Oc]s, [<assumes: CP =IP +Comp -Oc]s]
```

```
2017_05_10__21_37_13: Energy: 7800
2017_05_10__21_37_13: Grammar: 3285
2017_05_10__21_37_13: Input: 4515
2017_05_10__21_37_13: Temperature: 74.048426
```

After 300 iterations:

```
Iteration: 300
2017_05_10__21_39_58: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: DP]s, [>Jerry: VP]s, [>Jerry: PP =DP +O]s,
[>George: DP]s, [>George: PP =DP]s, [>George: VP =DP +O]s, [>Elaine: DP]s, [>Kramer:
DP]s, [>Kramer: CP =IP +Comp -Oc]s, [>ran: VP]s, [>ran: DP]s, [>walked: VP]s,
[>walked: CP =IP]s, [>walked: VP =DP]s, [>read: VP]s, [>read: VP =CP]s, [>read: VP =CP
+Oc]s, [>wrote: VP]s, [>wrote: DP]s, [>liked: VP =DP]s, [>saw: VP =DP]s, [>saw: PP
=DP]s, [>saw: VP =CP +Oc]s, [>loved: VP =DP]s, [>hated: VP =DP]s, [>hated: VP =CP]s,
[>with: PP =DP]s, [>by: PP =DP]s, [>by: VP =DP]s, [>above: VP]s, [>above: VP =CP]s,
[>above: PP =DP]s, [>under: PP =DP]s, [>that: DP]s, [>that: CP =IP]s, [>that: VP
=CP]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP =DP]s, [>says: VP =CP]s, [>says: PP
=DP]s, [>assumes: VP =CP]s, [>assumes: CP =IP]s, [>assumes: CP =IP +Comp -Oc]s]
2017_05_10__21_39_58: Energy: 4478
2017_05_10__21_39_58: Grammar: 1072
2017_05_10__21_39_58: Input: 3406
2017_05_10__21_39_58: Temperature: 40.602006
```

After over 400 iterations:

```
Iteration: 433
2017_05_10__21_40_20: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>George: DP]s, [>Elaine: DP]s, [>Kramer: DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s, [>loved: VP =DP]s, [>hated: VP =DP]s,
[>with: PP =DP]s, [>by: PP =DP]s, [>above: PP =DP]s, [>under: PP =DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>assumes: VP =CP]s]
2017_05_10__21_40_20: Energy: 3491
2017_05_10__21_40_20: Grammar: 525
2017_05_10__21_40_20: Input: 2966
2017_05_10__21_40_20: Temperature: 27.227236
```

Head Final Input

The relevant log file for this run is:

log_2017_05_10__21_43_43 - Category, Head-final, PP & CP.txt

The generated input:

```
Input is: ['Jerry George walked Jerry above that says', 'Elaine Kramer read that
assumes Elaine under', 'Elaine Kramer Jerry hated that says', 'George Jerry walked
that assumes Elaine under', 'Elaine Elaine saw', 'Jerry walked', 'George Kramer Elaine
saw Elaine with that thinks', 'Elaine Jerry saw Elaine above', 'Kramer George George
hated Elaine with that says', 'Elaine Elaine loved George by', 'Elaine George hated
Jerry by', 'Jerry Jerry loved', 'George Elaine loved Elaine under', 'Elaine ran',
'Elaine Kramer loved', 'Elaine Elaine Jerry liked Elaine above that says', 'George
George walked that knows', 'Elaine Jerry read that knows', 'George Elaine saw Elaine
with', 'George Kramer hated Jerry by', 'George walked George above', 'George George
liked', 'Elaine Kramer saw', 'Jerry Elaine liked Elaine with', 'George walked Jerry
under', 'Elaine read Kramer under', 'Elaine Elaine walked that says Jerry under',
'Jerry George saw', 'George ran', 'Kramer Kramer hated Elaine with', 'Elaine Jerry
liked Kramer under', 'George Kramer liked Elaine above', 'Elaine Elaine George liked
that knows', 'George George walked that says', 'George wrote', 'Jerry George loved',
'Kramer Elaine ran that says Elaine with', 'George George Kramer hated Jerry above
that thinks', 'Kramer Jerry liked Kramer above', 'Jerry Jerry George hated that
assumes', 'Kramer Kramer Elaine loved that thinks Jerry above', 'George Jerry wrote
that thinks', 'Elaine Kramer Kramer loved that says', 'Elaine walked Elaine by',
```

```
'Elaine Jerry liked Kramer above', 'Elaine read Jerry with', 'George read Kramer
above', 'Elaine Kramer saw Kramer above', 'George Elaine hated Kramer above', 'George
Jerry saw']
```

The initial hypothesis:

```
Initial hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP
=IP -Comp]s, [>Jerry: DP]s, [>Jerry: VP]s, [>Jerry: CP =IP]s, [>Jerry: VP =DP]s,
[>Jerry: VP =CP]s, [>Jerry: PP =DP]s, [>Jerry: VP =DP +O]s, [>Jerry: PP =DP +O]s,
[>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s, [>George: DP]s, [>George: VP]s,
[>George: CP =IP]s, [>George: VP =DP]s, [>George: VP =CP]s, [>George: PP =DP]s,
[>George: VP =DP +O]s, [>George: PP =DP +O]s, [>George: VP =CP +Oc]s, [>George: CP =IP
+Comp -Oc]s, [>Elaine: VP]s, [>Elaine: DP]s, [>Elaine: CP =IP]s, [>Elaine: VP =DP]s,
[>Elaine: VP =CP]s, [>Elaine: PP =DP]s, [>Elaine: VP =DP +O]s, [>Elaine: PP =DP +O]s,
[>Elaine: VP =CP +Oc]s, [>Elaine: CP =IP +Comp -Oc]s, [>Kramer: VP]s, [>Kramer: DP]s,
[>Kramer: CP =IP]s, [>Kramer: VP =DP]s, [>Kramer: VP =CP]s, [>Kramer: PP =DP]s,
[>Kramer: VP =DP +O]s, [>Kramer: PP =DP +O]s, [>Kramer: VP =CP +Oc]s, [>Kramer: CP =IP
+Comp -Oc]s, [>ran: VP]s, [>ran: DP]s, [>ran: VP =DP]s, [>ran: VP =CP]s, [>ran: PP
=DP]s, [>ran: CP =IP]s, [>ran: VP =DP +O]s, [>ran: PP =DP +O]s, [>ran: VP =CP +Oc]s,
[>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [>walked: DP]s, [>walked: CP =IP]s,
[>walked: VP =DP]s, [>walked: VP =CP]s, [>walked: PP =DP]s, [>walked: VP =DP +O]s,
[>walked: PP =DP +O]s, [>walked: VP =CP +Oc]s, [>walked: CP =IP +Comp -Oc]s, [>read:
VP]s, [>read: DP]s, [>read: CP =IP]s, [>read: VP =DP]s, [>read: VP =CP]s, [>read: PP
=DP]s, [>read: VP =DP +O]s, [>read: PP =DP +O]s, [>read: VP =CP +Oc]s, [>read: CP =IP
+Comp -Oc]s, [>wrote: VP]s, [>wrote: DP]s, [>wrote: CP =IP]s, [>wrote: VP =DP]s,
[>wrote: VP =CP]s, [>wrote: PP =DP]s, [>wrote: VP =DP +O]s, [>wrote: PP =DP +O]s,
[>wrote: VP =CP +Oc]s, [>wrote: CP =IP +Comp -Oc]s, [>liked: VP]s, [>liked: DP]s,
[>liked: CP =IP]s, [>liked: VP =DP]s, [>liked: VP =CP]s, [>liked: PP =DP]s, [>liked:
VP =DP +O]s, [>liked: PP =DP +O]s, [>liked: VP =CP +Oc]s, [>liked: CP =IP +Comp -Oc]s,
[>saw: VP]s, [>saw: DP]s, [>saw: CP =IP]s, [>saw: VP =DP]s, [>saw: VP =CP]s, [>saw: PP
=DP]s, [>saw: VP =DP +O]s, [>saw: PP =DP +O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP
+Comp -Oc]s, [>loved: VP]s, [>loved: DP]s, [>loved: CP =IP]s, [>loved: VP =DP]s,
[>loved: VP =CP]s, [>loved: PP =DP]s, [>loved: VP =DP +O]s, [>loved: PP =DP +O]s,
[>loved: VP =CP +Oc]s, [>loved: CP =IP +Comp -Oc]s, [>hated: VP]s, [>hated: DP]s,
[>hated: CP =IP]s, [>hated: VP =DP]s, [>hated: VP =CP]s, [>hated: PP =DP]s, [>hated:
VP =DP +O]s, [>hated: PP =DP +O]s, [>hated: VP =CP +Oc]s, [>hated: CP =IP +Comp -Oc]s,
[>with: VP]s, [>with: DP]s, [>with: CP =IP]s, [>with: VP =DP]s, [>with: VP =CP]s,
[>with: PP =DP]s, [>with: VP =DP +O]s, [>with: PP =DP +O]s, [>with: VP =CP +Oc]s,
[>with: CP =IP +Comp -Oc]s, [>by: VP]s, [>by: DP]s, [>by: VP =DP]s, [>by: VP =CP]s,
[>by: PP =DP]s, [>by: CP =IP]s, [>by: VP =DP +O]s, [>by: PP =DP +O]s, [>by: VP =CP
+Oc]s, [>by: CP =IP +Comp -Oc]s, [>above: VP]s, [>above: DP]s, [>above: CP =IP]s,
[>above: VP =DP]s, [>above: VP =CP]s, [>above: PP =DP]s, [>above: VP =DP +O]s,
[>above: PP =DP +O]s, [>above: VP =CP +Oc]s, [>above: CP =IP +Comp -Oc]s, [>under:
VP]s, [>under: DP]s, [>under: CP =IP]s, [>under: VP =DP]s, [>under: VP =CP]s, [>under:
PP =DP]s, [>under: VP =DP +O]s, [>under: PP =DP +O]s, [>under: VP =CP +Oc]s, [>under:
CP =IP +Comp -Oc]s, [>that: VP]s, [>that: DP]s, [>that: CP =IP]s, [>that: VP =DP]s,
[>that: VP =CP]s, [>that: PP =DP]s, [>that: VP =DP +O]s, [>that: PP =DP +O]s, [>that:
VP =CP +Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP]s, [>knows: DP]s, [>knows: CP
=IP]s, [>knows: VP =DP]s, [>knows: VP =CP]s, [>knows: PP =DP]s, [>knows: VP =DP +O]s,
[>knows: PP =DP +O]s, [>knows: VP =CP +Oc]s, [>knows: CP =IP +Comp -Oc]s, [>says:
VP]s, [>says: DP]s, [>says: CP =IP]s, [>says: VP =DP]s, [>says: VP =CP]s, [>says: PP
=DP]s, [>says: VP =DP +O]s, [>says: PP =DP +O]s, [>says: VP =CP +Oc]s, [>says: CP =IP
+Comp -Oc]s, [>thinks: VP]s, [>thinks: DP]s, [>thinks: CP =IP]s, [>thinks: VP =DP]s,
[>thinks: VP =CP]s, [>thinks: PP =DP]s, [>thinks: VP =DP +O]s, [>thinks: PP =DP +O]s,
[>thinks: VP =CP +Oc]s, [>thinks: CP =IP +Comp -Oc]s, [>assumes: VP]s, [>assumes:
DP]s, [>assumes: CP =IP]s, [>assumes: VP =DP]s, [>assumes: VP =CP]s, [>assumes: PP
=DP]s, [>assumes: VP =DP +O]s, [>assumes: PP =DP +O]s, [>assumes: VP =CP +Oc]s,
[>assumes: CP =IP +Comp -Oc]s]
```

Initial energy:

```
Temperature: 100.000000
Energy: 10979
Grammar: 5148
Input: 5831
```

After 100 iterations:

```
Iteration: 100
2017_05_11__01_44_56: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: DP]s, [<Jerry: VP]s, [<Jerry: VP =DP]s, [<Jerry:
VP =CP]s, [>Jerry: PP =DP]s, [<Jerry: CP =IP]s, [>Jerry: PP =DP]s, [<Jerry: VP =DP
```

```
+O]s, [<Jerry: VP =CP +Oc]s, [>George: DP]s, [<George: VP]s, [<George: VP =DP]s,
[<George: VP =CP]s, [>George: PP =DP]s, [<George: VP =DP +O]s, [<George: VP =CP +Oc]s,
[>Elaine: DP]s, [>Elaine: DP]s, [<Elaine: VP =CP]s, [>Elaine: PP =DP]s, [<Elaine: CP
=IP]s, [>Elaine: PP =DP +O]s, [<Elaine: VP =DP +O]s, [<Elaine: VP =CP +Oc]s, [<Elaine:
CP =IP +Comp -Oc]s, [<Kramer: VP]s, [>Kramer: DP]s, [<Kramer: DP]s, [<Kramer: VP
=DP]s, [<Kramer: VP =CP]s, [>Kramer: PP =DP]s, [<Kramer: CP =IP]s, [<Kramer: VP =DP
+O]s, [<ran: VP]s, [>ran: DP]s, [<ran: VP =CP]s, [>ran: PP =DP]s, [<ran: VP =DP +O]s,
[>ran: PP =DP +O]s, [<ran: VP =CP +Oc]s, [<ran: CP =IP +Comp -Oc]s, [<walked: VP]s,
[<walked: VP =CP]s, [<walked: CP =IP]s, [<walked: VP =DP +O]s, [<walked: VP =CP +Oc]s,
[<walked: CP =IP +Comp -Oc]s, [<read: VP]s, [>read: DP]s, [<read: VP =DP]s, [<read: CP
=IP]s, [<wrote: VP]s, [>wrote: DP]s, [<wrote: CP =IP]s, [<wrote: VP =CP]s, [>wrote: PP
=DP]s, [>wrote: PP =DP]s, [<wrote: VP =DP +O]s, [<wrote: VP =CP +Oc]s, [<wrote: CP =IP
+Comp -Oc]s, [<liked: VP]s, [<liked: CP =IP]s, [<liked: VP =DP]s, [<liked: VP =CP]s,
[>liked: PP =DP +O]s, [<liked: VP =CP +Oc]s, [<saw: VP]s, [>saw: DP]s, [<saw: VP
=DP]s, [<saw: VP =CP]s, [>saw: PP =DP]s, [<saw: CP =IP]s, [<saw: VP =DP +O]s, [>saw:
PP =DP +O]s, [<saw: VP =CP +Oc]s, [>loved: DP]s, [<loved: DP]s, [<loved: VP =DP]s,
[>loved: PP =DP]s, [<loved: VP =CP]s, [>loved: PP =DP +O]s, [<loved: VP =CP +Oc]s,
[<loved: CP =IP +Comp -Oc]s, [<hated: VP]s, [>hated: DP]s, [>hated: DP]s, [<hated: VP
=DP]s, [>hated: PP =DP]s, [<hated: CP =IP]s, [<hated: CP =IP]s, [<hated: VP =DP +O]s,
[<hated: VP =CP +Oc]s, [<hated: CP =IP +Comp -Oc]s, [<with: VP]s, [>with: DP]s,
[<with: VP =CP]s, [>with: PP =DP]s, [<with: VP =DP +O]s, [>with: PP =DP +O]s, [<with:
VP =CP +Oc]s, [<by: VP]s, [<by: CP =IP]s, [<by: VP =DP]s, [>by: PP =DP]s, [<by: VP =DP
+O]s, [>by: PP =DP +O]s, [<by: VP =CP +Oc]s, [<by: CP =IP +Comp -Oc]s, [<above: VP]s,
[<above: VP =DP]s, [<above: VP =CP]s, [>above: PP =DP]s, [<above: CP =IP]s, [>above:
PP =DP +O]s, [<above: VP =DP +O]s, [<under: VP]s, [>under: DP]s, [<under: VP =DP]s,
[<under: VP =CP]s, [>under: PP =DP]s, [<under: CP =IP]s, [>under: PP =DP +O]s,
[<under: VP =DP +O]s, [<under: VP =CP +Oc]s, [<under: CP =IP +Comp -Oc]s, [<that:
VP]s, [>that: DP]s, [<that: DP]s, [<that: VP =DP]s, [>that: PP =DP]s, [<that: CP
=IP]s, [>that: PP =DP +O]s, [<that: VP =DP +O]s, [<that: VP =CP +Oc]s, [>knows: DP]s,
[<knows: CP =IP]s, [<knows: VP =CP]s, [>knows: PP =DP]s, [>knows: PP =DP +O]s,
[<knows: VP =CP +Oc]s, [>says: DP]s, [<says: VP =DP]s, [<says: VP =CP]s, [>says: PP
=DP]s, [<says: CP =IP]s, [<says: VP =CP]s, [>says: PP =DP +O]s, [<says: CP =IP +Comp -
Oc]s, [<thinks: VP]s, [>thinks: DP]s, [<thinks: VP =DP]s, [<thinks: VP =CP]s,
[>thinks: PP =DP]s, [<thinks: VP =DP +O]s, [>thinks: PP =DP +O]s, [<thinks: VP =CP
+Oc]s, [<thinks: CP =IP +Comp -Oc]s, [<assumes: VP]s, [<assumes: VP =DP]s, [<assumes:
VP =CP]s, [>assumes: PP =DP]s, [<assumes: CP =IP]s, [<assumes: CP =IP +Comp -Oc]s]
2017_05_11__01_44_56: Energy: 9918
2017_05_11__01_44_56: Grammar: 3931
2017_05_11__01_44_56: Input: 5987
2017_05_11__01_44_56: Temperature: 74.048426
```

After 300 iterations:

```
Iteration: 300
2017_05_11__01_54_32: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: DP]s, [<Jerry: VP]s, [<Jerry: CP =IP]s, [<Jerry:
VP =DP +O]s, [<Jerry: PP =DP +O]s, [>George: DP]s, [<George: VP =CP]s, [<George: VP
=DP]s, [<George: PP =DP +O]s, [>Elaine: DP]s, [<Elaine: PP =DP]s, [<Elaine: CP =IP]s,
[<Elaine: PP =DP +O]s, [>Kramer: DP]s, [<Kramer: CP =IP]s, [<ran: VP]s, [>ran: DP]s,
[<ran: VP =DP +O]s, [<ran: VP =CP +Oc]s, [<walked: VP]s, [<read: VP]s, [<wrote: VP]s,
[>wrote: DP]s, [<wrote: VP =DP +O]s, [<wrote: PP =DP +O]s, [<liked: VP =DP]s, [<saw:
VP =DP]s, [<loved: DP]s, [<loved: VP =DP]s, [<loved: VP =DP]s, [<loved: VP =CP +Oc]s,
[<hated: VP]s, [>hated: DP]s, [<hated: PP =DP +O]s, [>with: DP]s, [<with: VP =CP]s,
[<by: VP =DP]s, [<by: PP =DP]s, [<by: VP =CP]s, [>above: DP]s, [<above: VP =CP]s,
[<above: PP =DP]s, [<above: VP =DP +O]s, [<above: PP =DP +O]s, [<above: CP =IP +Comp -
Oc]s, [<under: VP =CP]s, [<under: PP =DP]s, [<under: VP =CP +Oc]s, [<under: CP =IP
+Comp -Oc]s, [>that: DP]s, [<that: VP =DP]s, [<that: CP =IP]s, [<that: VP =DP +O]s,
[>knows: DP]s, [<knows: VP =CP]s, [<says: VP =CP]s, [<says: PP =DP +O]s, [<says: PP
=DP +O]s, [<says: VP =DP +O]s, [<thinks: VP]s, [>thinks: DP]s, [<thinks: VP =CP]s,
[<thinks: CP =IP +Comp -Oc]s, [<assumes: CP =IP]s, [<assumes: PP =DP]s, [<assumes: VP
=CP +Oc]s]
2017_05_11__01_54_32: Energy: 6943
2017_05_11__01_54_32: Grammar: 1622
2017_05_11__01_54_32: Input: 5321
2017_05_11__01_54_32: Temperature: 40.602006
```

After over 800 iterations:

```
Iteration: 880
2017_05_11__01_57_58: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>George: DP]s, [>Elaine: DP]s, [>Kramer: DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
```

```
[<liked: VP =DP]s, [<saw: VP =DP]s, [<loved: VP =DP]s, [<hated: VP =DP]s,
[<with: PP =DP]s, [<by: PP =DP]s, [<above: PP =DP]s, [<under: PP =DP]s,
[<that: CP =IP]s,
[<knows: VP =CP]s, [<says: VP =CP]s, [<thinks: VP =CP]s, [<assumes: VP =CP]s]
2017_05_11__01_57_58: Energy: 4531
2017_05_11__01_57_58: Grammar: 548
2017_05_11__01_57_58: Input: 3983
2017_05_11__01_57_58: Temperature: 7.107867
```

Mixed Category Input

The relevant log file for this run is:

log_2017_05_11__10_30_47 - Category, Mixed-category, PP & CP.txt

The generated input:

```
Input is: ['Kramer saw Jerry', 'Kramer read', 'Elaine read', 'Jerry saw George',
'George says that Kramer ran Kramer with', 'Elaine hated Kramer Jerry by', 'George
liked Jerry Elaine above', 'Jerry assumes that George ran', 'Kramer loved George
Elaine above', 'Kramer saw Elaine', 'Jerry read', 'Elaine wrote George with', 'Elaine
read', 'Kramer read', 'Elaine read', 'George read', 'Kramer ran George with', 'Kramer
loved Elaine', 'Jerry saw Kramer', 'Elaine wrote', 'Elaine knows that Kramer read
Elaine above', 'Jerry says that Jerry hated George George with', 'Elaine loved Elaine
Elaine under', 'Kramer ran Kramer by', 'Jerry says that Kramer wrote', 'Jerry assumes
that Kramer ran Kramer above', 'Elaine read Elaine by', 'Jerry hated Elaine Elaine
by', 'George wrote', 'Jerry hated Kramer Elaine with', 'George saw Jerry', 'Kramer saw
George Kramer with', 'Elaine liked Kramer', 'Elaine loved Elaine', 'Jerry ran Elaine
with', 'Kramer ran', 'Elaine wrote Jerry with', 'Jerry knows that George loved Elaine
Kramer above', 'Elaine knows that Jerry hated Jerry Kramer by', 'Elaine walked Jerry
under', 'Jerry wrote Jerry by', 'George ran', 'Jerry wrote Elaine above', 'Elaine
wrote', 'Jerry saw Kramer', 'Elaine read Elaine by', 'Elaine saw Elaine Kramer with',
'George read George by', 'Kramer knows that George walked Jerry with', 'George wrote']
```

The initial hypothesis:

```
Initial hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP
=IP -Comp]s, [>Jerry: VP]s, [>Jerry: DP]s, [>Jerry: CP =IP]s, [>Jerry: VP =DP]s,
[>Jerry: VP =CP]s, [>Jerry: PP =DP]s, [>Jerry: VP =DP +O]s, [>Jerry: PP =DP +O]s,
[>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s, [>George: VP]s, [>George: DP]s,
[>George: CP =IP]s, [>George: VP =DP]s, [>George: VP =CP]s, [>George: PP =DP]s,
[>George: VP =DP +O]s, [>George: PP =DP +O]s, [>George: VP =CP +Oc]s, [>George: CP =IP
+Comp -Oc]s, [>Elaine: VP]s, [>Elaine: DP]s, [>Elaine: CP =IP]s, [>Elaine: VP =DP]s,
[>Elaine: VP =CP]s, [>Elaine: PP =DP]s, [>Elaine: VP =DP +O]s, [>Elaine: PP =DP +O]s,
[>Elaine: VP =CP +Oc]s, [>Elaine: CP =IP +Comp -Oc]s, [>Kramer: VP]s, [>Kramer: DP]s,
[>Kramer: CP =IP]s, [>Kramer: VP =DP]s, [>Kramer: VP =CP]s, [>Kramer: PP =DP]s,
[>Kramer: VP =DP +O]s, [>Kramer: PP =DP +O]s, [>Kramer: VP =CP +Oc]s, [>Kramer: CP =IP
+Comp -Oc]s, [>ran: VP]s, [>ran: DP]s, [>ran: CP =IP]s, [>ran: VP =DP]s, [>ran: VP
=CP]s, [>ran: PP =DP]s, [>ran: VP =DP +O]s, [>ran: PP =DP +O]s, [>ran: VP =CP +Oc]s,
[>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [>walked: DP]s, [>walked: CP =IP]s,
[>walked: VP =DP]s, [>walked: VP =CP]s, [>walked: PP =DP]s, [>walked: VP =DP +O]s,
[>walked: PP =DP +O]s, [>walked: VP =CP +Oc]s, [>walked: CP =IP +Comp -Oc]s, [>read:
VP]s, [>read: DP]s, [>read: CP =IP]s, [>read: VP =DP]s, [>read: VP =CP]s, [>read: PP
=DP]s, [>read: VP =DP +O]s, [>read: PP =DP +O]s, [>read: VP =CP +Oc]s, [>read: CP =IP
+Comp -Oc]s, [>wrote: VP]s, [>wrote: DP]s, [>wrote: CP =IP]s, [>wrote: VP =DP]s,
[>wrote: VP =CP]s, [>wrote: PP =DP]s, [>wrote: VP =DP +O]s, [>wrote: PP =DP +O]s,
[>wrote: VP =CP +Oc]s, [>wrote: CP =IP +Comp -Oc]s, [>liked: VP]s, [>liked: DP]s,
[>liked: CP =IP]s, [>liked: VP =DP]s, [>liked: VP =CP]s, [>liked: PP =DP]s, [>liked:
VP =DP +O]s, [>liked: PP =DP +O]s, [>liked: VP =CP +Oc]s, [>liked: CP =IP +Comp -Oc]s,
[>saw: VP]s, [>saw: DP]s, [>saw: CP =IP]s, [>saw: VP =DP]s, [>saw: VP =CP]s, [>saw: PP
=DP]s, [>saw: VP =DP +O]s, [>saw: PP =DP +O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP
+Comp -Oc]s, [>loved: VP]s, [>loved: DP]s, [>loved: CP =IP]s, [>loved: VP =DP]s,
[>loved: VP =CP]s, [>loved: PP =DP]s, [>loved: VP =DP +O]s, [>loved: PP =DP +O]s,
[>loved: VP =CP +Oc]s, [>loved: CP =IP +Comp -Oc]s, [>hated: VP]s, [>hated: DP]s,
[>hated: CP =IP]s, [>hated: VP =DP]s, [>hated: VP =CP]s, [>hated: PP =DP]s, [>hated:
VP =DP +O]s, [>hated: PP =DP +O]s, [>hated: VP =CP +Oc]s, [>hated: CP =IP +Comp -Oc]s,
[>with: VP]s, [>with: DP]s, [>with: CP =IP]s, [>with: VP =DP]s, [>with: VP =CP]s,
[>with: PP =DP]s, [>with: VP =DP +O]s, [>with: PP =DP +O]s, [>with: VP =CP +Oc]s,
```

```
[>with: CP =IP +Comp -Oc]s, [>by: VP]s, [>by: DP]s, [>by: CP =IP]s, [>by: VP =DP]s,
[>by: VP =CP]s, [>by: PP =DP]s, [>by: VP =DP +O]s, [>by: PP =DP +O]s, [>by: VP =CP
+Oc]s, [>by: CP =IP +Comp -Oc]s, [>above: VP]s, [>above: DP]s, [>above: CP =IP]s,
[>above: VP =DP]s, [>above: VP =CP]s, [>above: PP =DP]s, [>above: VP =DP +O]s,
[>above: PP =DP +O]s, [>above: VP =CP +Oc]s, [>above: CP =IP +Comp -Oc]s, [>under:
VP]s, [>under: DP]s, [>under: CP =IP]s, [>under: VP =DP]s, [>under: VP =CP]s, [>under:
PP =DP]s, [>under: VP =DP +O]s, [>under: PP =DP +O]s, [>under: VP =CP +Oc]s, [>under:
CP =IP +Comp -Oc]s, [>that: VP]s, [>that: DP]s, [>that: CP =IP]s, [>that: VP =DP]s,
[>that: VP =CP]s, [>that: PP =DP]s, [>that: VP =DP +O]s, [>that: PP =DP +O]s, [>that:
VP =CP +Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP]s, [>knows: DP]s, [>knows: CP
=IP]s, [>knows: VP =DP]s, [>knows: VP =CP]s, [>knows: PP =DP]s, [>knows: VP =DP +O]s,
[>knows: PP =DP +O]s, [>knows: VP =CP +Oc]s, [>knows: CP =IP +Comp -Oc]s, [>says:
VP]s, [>says: DP]s, [>says: CP =IP]s, [>says: VP =DP]s, [>says: VP =CP]s, [>says: PP
=DP]s, [>says: VP =DP +O]s, [>says: PP =DP +O]s, [>says: VP =CP +Oc]s, [>says: CP =IP
+Comp -Oc]s, [>assumes: VP]s, [>assumes: DP]s, [>assumes: CP =IP]s, [>assumes: VP
=DP]s, [>assumes: VP =CP]s, [>assumes: PP =DP]s, [>assumes: VP =DP +O]s, [>assumes: PP
=DP +O]s, [>assumes: VP =CP +Oc]s, [>assumes: CP =IP +Comp -Oc]s]
```

Initial energy:

```
Temperature: 100.000000
Energy: 9736
Grammar: 4908
Input: 4828
```

After 100 iterations:

```
Iteration: 100
2017_05_11__10_58_12: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [<Jerry: DP]s, [>Jerry: DP]s, [>Jerry: CP =IP]s, [<Jerry:
PP =DP]s, [<Jerry: PP =DP +O]s, [>Jerry: VP =DP +O]s, [>Jerry: VP =CP +Oc]s, [>Jerry:
CP =IP +Comp -Oc]s, [<George: DP]s, [>George: VP =DP]s, [>George: VP =CP]s, [<George:
PP =DP]s, [<George: PP =DP +O]s, [>George: VP =DP +O]s, [>George: VP =CP +Oc]s,
[>George: CP =IP +Comp -Oc]s, [>Elaine: VP]s, [<Elaine: DP]s, [>Elaine: CP =IP]s,
[>Elaine: VP =CP]s, [<Elaine: PP =DP]s, [>Elaine: VP =CP +Oc]s, [>Kramer: VP]s,
[<Kramer: DP]s, [>Kramer: DP]s, [>Kramer: CP =IP]s, [<Kramer: PP =DP]s, [<Kramer: PP
=DP +O]s, [>Kramer: VP =CP +Oc]s, [>Kramer: CP =IP +Comp -Oc]s, [>ran: VP]s, [>ran: CP
=IP]s, [>ran: VP =DP]s, [>ran: VP =CP]s, [<ran: PP =DP]s, [<ran: PP =DP +O]s, [>ran:
VP =DP +O]s, [>ran: VP =CP +Oc]s, [>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [<walked:
DP]s, [>walked: CP =IP]s, [>walked: VP =CP]s, [<walked: PP =DP]s, [<walked: PP =DP
+O]s, [<walked: PP =DP +O]s, [>walked: VP =DP +O]s, [>walked: VP =CP +Oc]s, [>walked:
CP =IP +Comp -Oc]s, [>read: VP]s, [<read: DP]s, [>read: DP]s, [>read: VP =CP]s,
[<read: PP =DP]s, [>read: VP =DP +O]s, [>read: VP =CP +Oc]s, [>wrote: VP]s, [>wrote:
CP =IP]s, [>wrote: VP =CP]s, [>wrote: VP =DP +O]s, [>wrote: VP =CP +Oc]s, [>wrote: CP
=IP +Comp -Oc]s, [<liked: DP]s, [>liked: DP]s, [>liked: CP =IP]s, [>liked: VP =DP]s,
[>liked: VP =CP]s, [<liked: PP =DP]s, [>liked: VP =DP +O]s, [<liked: PP =DP +O]s,
[>saw: VP]s, [>saw: CP =IP]s, [>saw: VP =DP]s, [>saw: VP =CP]s, [<saw: PP =DP]s,
[<saw: PP =DP +O]s, [>saw: VP =DP +O]s, [>saw: CP =IP +Comp -Oc]s, [<loved: DP]s,
[>loved: CP =IP]s, [>loved: VP =DP]s, [>loved: VP =DP +O]s, [>loved: VP =CP +Oc]s,
[>hated: CP =IP]s, [>hated: VP =DP]s, [>hated: VP =CP]s, [<hated: PP =DP]s, [>hated:
VP =DP +O]s, [>hated: VP =CP +Oc]s, [>hated: CP =IP +Comp -Oc]s, [<with: DP]s, [>with:
VP]s, [>with: CP =IP]s, [>with: VP =DP]s, [>with: VP =CP]s, [<with: PP =DP]s, [<with:
PP =DP +O]s, [>with: VP =DP +O]s, [>with: VP =CP +Oc]s, [>by: VP]s, [<by: DP]s, [<by:
DP]s, [>by: CP =IP]s, [>by: VP =DP]s, [<by: PP =DP]s, [<by: PP =DP +O]s, [>by: VP =DP
+O]s, [>by: VP =CP +Oc]s, [>by: CP =IP +Comp -Oc]s, [>above: VP]s, [<above: DP]s,
[>above: CP =IP]s, [>above: VP =DP]s, [>above: VP =CP]s, [<above: PP =DP]s, [>above:
VP =DP +O]s, [>above: VP =CP +Oc]s, [>above: CP =IP +Comp -Oc]s, [<under: DP]s,
[>under: CP =IP]s, [>under: VP =DP]s, [>under: VP =CP]s, [<under: PP =DP]s, [<under:
PP =DP +O]s, [>under: VP =DP +O]s, [>under: CP =IP +Comp -Oc]s, [>that: CP =IP]s,
[>that: VP =DP]s, [<that: PP =DP +O]s, [>that: CP =IP +Comp -Oc]s, [>knows: DP]s,
[>knows: VP =DP]s, [>knows: VP =CP]s, [<knows: PP =DP +O]s, [>knows: VP =DP +O]s,
[>knows: CP =IP +Comp -Oc]s, [>says: CP =IP]s, [>says: VP =CP]s, [<says: PP =DP]s,
[<says: PP =DP +O]s, [>says: VP =DP +O]s, [>says: VP =CP +Oc]s, [>says: CP =IP +Comp -
Oc]s, [>assumes: VP]s, [<assumes: DP]s, [>assumes: VP =CP]s, [<assumes: PP =DP +O]s,
[>assumes: VP =CP +Oc]s]
2017_05_11__10_58_12: Energy: 8486
2017_05_11__10_58_12: Grammar: 3658
2017_05_11__10_58_12: Input: 4828
2017_05_11__10_58_12: Temperature: 74.048426
```

After 300 iterations:

```
Iteration: 300
2017_05_11__11_01_18: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: DP]s, [<Jerry: PP =DP +O]s, [>Jerry: VP =DP +O]s,
[>Jerry: VP =CP +Oc]s, [<George: DP]s, [>George: VP]s, [<George: PP =DP]s, [>George:
VP =DP +O]s, [<George: PP =DP +O]s, [>Elaine: VP]s, [<Elaine: DP]s, [<Elaine: PP
=DP]s, [<Elaine: PP =DP]s, [>Kramer: DP]s, [<Kramer: PP =DP +O]s, [>ran: VP]s, [>ran:
VP =CP]s, [>walked: VP]s, [>walked: CP =IP +Comp -Oc]s, [>read: VP]s, [>read: DP]s,
[<read: PP =DP]s, [>wrote: VP]s, [>wrote: CP =IP]s, [>wrote: VP =CP]s, [>wrote: VP =CP
+Oc]s, [>liked: DP]s, [>liked: VP =DP]s, [>liked: VP =DP +O]s, [>saw: CP =IP]s, [>saw:
VP =DP]s, [>loved: VP =DP]s, [>hated: VP =DP]s, [<hated: PP =DP]s, [<hated: PP =DP]s,
[>hated: VP =DP +O]s, [>hated: VP =CP +Oc]s, [>with: VP =DP]s, [<with: PP =DP]s, [<by:
PP =DP]s, [>by: VP =DP +O]s, [<above: PP =DP]s, [>under: VP =DP]s, [<under: PP =DP]s,
[<under: PP =DP]s, [>under: VP =CP]s, [<under: PP =DP +O]s, [>under: CP =IP +Comp -
Oc]s, [>that: DP]s, [>that: CP =IP]s, [>that: VP =CP]s, [>that: CP =IP +Comp -Oc]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>says: VP =DP +O]s, [>assumes: VP =CP]s,
[<assumes: PP =DP]s, [>assumes: VP =CP +Oc]s]
2017_05_11__11_01_18: Energy: 5639
2017_05_11__11_01_18: Grammar: 1457
2017_05_11__11_01_18: Input: 4182
2017_05_11__11_01_18: Temperature: 40.602006
```

After over 400 iterations:

```
Iteration: 482
2017_05_11__11_01_38: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [<George: DP]s, [<Elaine: DP]s, [>Kramer: DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s, [>loved: VP =DP]s, [>hated: VP =DP]s,
[<with: PP =DP]s, [<by: PP =DP]s, [<above: PP =DP]s, [<under: PP =DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>assumes: VP =CP]s]
2017_05_11__11_01_38: Energy: 3795
2017_05_11__11_01_38: Grammar: 525
2017_05_11__11_01_38: Input: 3270
2017_05_11__11_01_38: Temperature: 23.499916
```

Mixed Word Input

The relevant log file for this run is:

log_2017_05_12__10_47_11 - Category, Mixed-word, PP & CP.txt

The generated input:

```
Input is: ['Jerry Kramer loved', 'Jerry knows that Elaine read Elaine under', 'Kramer
walked', 'Jerry Kramer loved', 'Elaine walked', 'George read', 'Elaine assumes that
Jerry wrote Elaine above', 'Kramer Jerry loved', 'Jerry says that Jerry saw Kramer
Elaine with', 'Elaine says that Kramer wrote George above', 'Kramer ran George with',
'Jerry walked', 'Jerry walked Jerry by', 'George saw Kramer', 'Elaine walked', 'George
walked', 'Kramer ran', 'Elaine says that George walked Kramer under', 'Kramer liked
Jerry', 'Jerry thinks that Jerry saw George Elaine by', 'George read George by',
'Jerry saw George', 'George thinks that George walked Elaine under', 'Kramer ran
Kramer above', 'Kramer Elaine hated', 'George liked Jerry', 'Kramer says that Jerry
ran', 'Elaine says that Kramer ran Jerry by', 'George ran Jerry with', 'Jerry liked
Elaine Kramer with', 'Kramer says that Elaine ran Elaine with', 'Kramer saw Elaine
Jerry above', 'Elaine thinks that George saw George', 'George Elaine loved Elaine
with', 'George walked', 'Elaine liked Kramer Elaine above', 'George read', 'Elaine
liked Kramer', 'Elaine liked Elaine', 'Jerry thinks that Kramer Elaine hated Elaine
above', 'Elaine knows that Jerry liked Kramer', 'Jerry George loved', 'Kramer liked
Elaine Elaine under', 'Kramer walked Kramer with', 'Jerry saw George Elaine above',
'Elaine says that George George loved Kramer above', 'Jerry Kramer hated', 'Elaine
Jerry loved', 'Kramer says that Jerry George hated', 'Kramer wrote Kramer above']
```

The initial hypothesis:

```
Initial hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP
=IP -Comp]s, [>Jerry: VP]s, [>Jerry: DP]s, [>Jerry: CP =IP]s, [>Jerry: VP =DP]s,
[>Jerry: VP =CP]s, [>Jerry: PP =DP]s, [>Jerry: VP =DP +O]s, [>Jerry: PP =DP +O]s,
[>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s, [>George: VP]s, [>George: DP]s,
[>George: CP =IP]s, [>George: VP =DP]s, [>George: VP =CP]s, [>George: PP =DP]s,
[>George: VP =DP +O]s, [>George: PP =DP +O]s, [>George: VP =CP +Oc]s, [>George: CP =IP
+Comp -Oc]s, [>Elaine: VP]s, [>Elaine: DP]s, [>Elaine: CP =IP]s, [>Elaine: VP =DP]s,
[>Elaine: VP =CP]s, [>Elaine: PP =DP]s, [>Elaine: VP =DP +O]s, [>Elaine: PP =DP +O]s,
[>Elaine: VP =CP +Oc]s, [>Elaine: CP =IP +Comp -Oc]s, [>Kramer: VP]s, [>Kramer: DP]s,
[>Kramer: CP =IP]s, [>Kramer: VP =DP]s, [>Kramer: VP =CP]s, [>Kramer: PP =DP]s,
[>Kramer: VP =DP +O]s, [>Kramer: PP =DP +O]s, [>Kramer: VP =CP +Oc]s, [>Kramer: CP =IP
+Comp -Oc]s, [>ran: VP]s, [>ran: DP]s, [>ran: CP =IP]s, [>ran: VP =DP]s, [>ran: VP
=CP]s, [>ran: PP =DP]s, [>ran: VP =DP +O]s, [>ran: PP =DP +O]s, [>ran: VP =CP +Oc]s,
[>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [>walked: DP]s, [>walked: CP =IP]s,
[>walked: VP =DP]s, [>walked: VP =CP]s, [>walked: PP =DP]s, [>walked: VP =DP +O]s,
[>walked: PP =DP +O]s, [>walked: VP =CP +Oc]s, [>walked: CP =IP +Comp -Oc]s, [>read:
VP]s, [>read: DP]s, [>read: CP =IP]s, [>read: VP =DP]s, [>read: VP =CP]s, [>read: PP
=DP]s, [>read: VP =DP +O]s, [>read: PP =DP +O]s, [>read: VP =CP +Oc]s, [>read: CP =IP
+Comp -Oc]s, [>wrote: VP]s, [>wrote: DP]s, [>wrote: CP =IP]s, [>wrote: VP =DP]s,
[>wrote: VP =CP]s, [>wrote: PP =DP]s, [>wrote: VP =DP +O]s, [>wrote: PP =DP +O]s,
[>wrote: VP =CP +Oc]s, [>wrote: CP =IP +Comp -Oc]s, [>liked: VP]s, [>liked: DP]s,
[>liked: CP =IP]s, [>liked: VP =DP]s, [>liked: VP =CP]s, [>liked: PP =DP]s, [>liked:
VP =DP +O]s, [>liked: PP =DP +O]s, [>liked: VP =CP +Oc]s, [>liked: CP =IP +Comp -Oc]s,
[>saw: VP]s, [>saw: DP]s, [>saw: CP =IP]s, [>saw: VP =DP]s, [>saw: VP =CP]s, [>saw: PP
=DP]s, [>saw: VP =DP +O]s, [>saw: PP =DP +O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP
+Comp -Oc]s, [>loved: VP]s, [>loved: DP]s, [>loved: CP =IP]s, [>loved: VP =DP]s,
[>loved: VP =CP]s, [>loved: PP =DP]s, [>loved: VP =DP +O]s, [>loved: PP =DP +O]s,
[>loved: VP =CP +Oc]s, [>loved: CP =IP +Comp -Oc]s, [>hated: VP]s, [>hated: DP]s,
[>hated: CP =IP]s, [>hated: VP =DP]s, [>hated: VP =CP]s, [>hated: PP =DP]s, [>hated:
VP =DP +O]s, [>hated: PP =DP +O]s, [>hated: VP =CP +Oc]s, [>hated: CP =IP +Comp -Oc]s,
[>with: VP]s, [>with: DP]s, [>with: CP =IP]s, [>with: VP =DP]s, [>with: VP =CP]s,
[>with: PP =DP]s, [>with: VP =DP +O]s, [>with: PP =DP +O]s, [>with: VP =CP +Oc]s,
[>with: CP =IP +Comp -Oc]s, [>by: VP]s, [>by: DP]s, [>by: CP =IP]s, [>by: VP =DP]s,
[>by: VP =CP]s, [>by: PP =DP]s, [>by: VP =DP +O]s, [>by: PP =DP +O]s, [>by: VP =CP
+Oc]s, [>by: CP =IP +Comp -Oc]s, [>above: VP]s, [>above: DP]s, [>above: CP =IP]s,
[>above: VP =DP]s, [>above: VP =CP]s, [>above: PP =DP]s, [>above: VP =DP +O]s,
[>above: PP =DP +O]s, [>above: VP =CP +Oc]s, [>above: CP =IP +Comp -Oc]s, [>under:
VP]s, [>under: DP]s, [>under: CP =IP]s, [>under: VP =DP]s, [>under: VP =CP]s, [>under:
PP =DP]s, [>under: VP =DP +O]s, [>under: PP =DP +O]s, [>under: VP =CP +Oc]s, [>under:
CP =IP +Comp -Oc]s, [>that: VP]s, [>that: DP]s, [>that: CP =IP]s, [>that: VP =DP]s,
[>that: VP =CP]s, [>that: PP =DP]s, [>that: VP =DP +O]s, [>that: PP =DP +O]s, [>that:
VP =CP +Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP]s, [>knows: DP]s, [>knows: CP
=IP]s, [>knows: VP =DP]s, [>knows: VP =CP]s, [>knows: PP =DP]s, [>knows: VP =DP +O]s,
[>knows: PP =DP +O]s, [>knows: VP =CP +Oc]s, [>knows: CP =IP +Comp -Oc]s, [>says:
VP]s, [>says: DP]s, [>says: CP =IP]s, [>says: VP =DP]s, [>says: VP =CP]s, [>says: PP
=DP]s, [>says: VP =DP +O]s, [>says: PP =DP +O]s, [>says: VP =CP +Oc]s, [>says: CP =IP
+Comp -Oc]s, [>thinks: VP]s, [>thinks: DP]s, [>thinks: CP =IP]s, [>thinks: VP =DP]s,
[>thinks: VP =CP]s, [>thinks: PP =DP]s, [>thinks: VP =DP +O]s, [>thinks: PP =DP +O]s,
[>thinks: VP =CP +Oc]s, [>thinks: CP =IP +Comp -Oc]s, [>assumes: VP]s, [>assumes:
DP]s, [>assumes: CP =IP]s, [>assumes: VP =DP]s, [>assumes: VP =CP]s, [>assumes: PP
=DP]s, [>assumes: VP =DP +O]s, [>assumes: PP =DP +O]s, [>assumes: VP =CP +Oc]s,
[>assumes: CP =IP +Comp -Oc]s]
```

Initial energy:

```
Temperature: 100.000000
Energy: 10384
Grammar: 5148
Input: 5236
```

After 100 iterations:

```
Iteration: 100
2017_05_12__12_06_02: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [<Jerry: DP]s, [>Jerry: VP =DP]s, [>Jerry: VP =CP]s,
[<Jerry: PP =DP +O]s, [>Jerry: VP =CP +Oc]s, [>George: VP]s, [<George: DP]s, [>George:
VP =DP]s, [>George: VP =CP]s, [<George: PP =DP]s, [<George: PP =DP +O]s, [>George: VP
=DP +O]s, [>George: VP =CP +Oc]s, [>George: CP =IP +Comp -Oc]s, [>Elaine: VP]s,
[<Elaine: DP]s, [>Elaine: VP =DP]s, [<Elaine: PP =DP]s, [<Elaine: PP =DP +O]s,
[>Elaine: VP =DP +O]s, [>Elaine: CP =IP +Comp -Oc]s, [<Kramer: DP]s, [>Kramer: CP
=IP]s, [>Kramer: VP =DP]s, [<Kramer: PP =DP]s, [<Kramer: PP =DP +O]s, [>Kramer: VP =DP
+O]s, [>Kramer: CP =IP +Comp -Oc]s, [>ran: VP]s, [<ran: DP]s, [>ran: CP =IP]s, [>ran:
VP =DP]s, [>ran: VP =CP]s, [<ran: PP =DP]s, [<ran: PP =DP +O]s, [>ran: VP =DP +O]s,
```

```
[>ran: VP =CP +Oc]s, [>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [<walked: DP]s,
[>walked: CP =IP]s, [>walked: VP =DP]s, [<walked: PP =DP]s, [>walked: VP =DP +O]s,
[<walked: PP =DP +O]s, [>walked: CP =IP +Comp -Oc]s, [>read: VP]s, [<read: DP]s,
[>read: VP =DP]s, [>read: VP =CP]s, [<read: PP =DP +O]s, [>read: VP =DP +O]s, [<read:
PP =DP +O]s, [>read: CP =IP +Comp -Oc]s, [>wrote: VP]s, [<wrote: DP]s, [>wrote: VP =DP
+O]s, [>wrote: CP =IP +Comp -Oc]s, [>liked: VP]s, [>liked: CP =IP]s, [>liked: VP
=DP]s, [>liked: VP =CP]s, [<liked: PP =DP]s, [<liked: PP =DP +O]s, [>liked: VP =DP
+O]s, [>liked: VP =CP +Oc]s, [>liked: CP =IP +Comp -Oc]s, [>saw: VP]s, [>saw: CP
=IP]s, [>saw: VP =DP]s, [<saw: PP =DP]s, [>saw: VP =DP +O]s, [>saw: VP =CP +Oc]s,
[<loved: DP]s, [>loved: CP =IP]s, [>loved: VP =CP]s, [<loved: PP =DP +O]s, [>hated:
VP]s, [>hated: CP =IP]s, [>hated: VP =DP]s, [>hated: VP =CP]s, [<hated: PP =DP +O]s,
[>hated: VP =DP +O]s, [>hated: VP =CP +Oc]s, [>with: VP]s, [<with: DP]s, [>with: VP
=CP]s, [<with: PP =DP]s, [>with: CP =IP]s, [<with: PP =DP +O]s, [>with: VP =CP +Oc]s,
[>with: CP =IP +Comp -Oc]s, [>by: VP]s, [>by: DP]s, [>by: VP =DP]s, [<by: PP =DP]s,
[<by: PP =DP +O]s, [>by: VP =CP +Oc]s, [>by: CP =IP +Comp -Oc]s, [>above: VP]s,
[<above: DP]s, [>above: CP =IP]s, [>above: VP =DP]s, [>above: VP =CP]s, [>above: VP
=DP +O]s, [<above: PP =DP +O]s, [>above: VP =CP +Oc]s, [>above: CP =IP +Comp -Oc]s,
[>under: VP]s, [<under: DP]s, [>under: VP =DP]s, [<under: PP =DP]s, [>under: VP =CP]s,
[<under: PP =DP +O]s, [>under: VP =DP +O]s, [>under: VP =CP +Oc]s, [>under: CP =IP
+Comp -Oc]s, [>that: VP]s, [>that: CP =IP]s, [>that: VP =DP]s, [<that: PP =DP +O]s,
[>that: VP =CP +Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP]s, [<knows: DP]s,
[<knows: PP =DP]s, [<knows: PP =DP +O]s, [>knows: VP =DP +O]s, [>knows: VP =CP +Oc]s,
[<says: DP]s, [>says: CP =IP]s, [>says: VP =DP]s, [>says: VP =CP]s, [<says: PP =DP]s,
[<says: PP =DP +O]s, [<says: PP =DP +O]s, [>says: CP =IP +Comp -Oc]s, [>thinks: VP
=DP]s, [>thinks: VP =CP]s, [<thinks: PP =DP]s, [<thinks: PP =DP +O]s, [>thinks: VP =CP
+Oc]s, [<assumes: DP]s, [>assumes: VP =CP]s, [<assumes: PP =DP +O]s, [>assumes: VP =DP
+O]s, [>assumes: VP =CP +Oc]s, [>assumes: CP =IP +Comp -Oc]s]
2017_05_12__12_06_02: Energy: 9308
2017_05_12__12_06_02: Grammar: 3673
2017_05_12__12_06_02: Input: 5635
2017_05_12__12_06_02: Temperature: 74.048426
```

After 300 iterations:

```
Iteration: 300
2017_05_12__12_12_11: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [<Jerry: DP]s, [>Jerry: VP =DP]s, [>Jerry: PP =DP]s,
[>Jerry: VP =DP +O]s, [<George: DP]s, [>George: VP =DP]s, [>George: PP =DP]s,
[<Elaine: DP]s, [>Elaine: VP =DP]s, [>Elaine: PP =DP]s, [>Elaine: PP =DP +O]s,
[>Elaine: CP =IP +Comp -Oc]s, [<Kramer: DP]s, [>Kramer: VP =DP]s, [>Kramer: PP =DP]s,
[>Kramer: CP =IP]s, [>ran: VP]s, [>walked: VP]s, [>walked: PP =DP +O]s, [>read: VP]s,
[<read: DP]s, [>read: PP =DP]s, [>read: PP =DP +O]s, [>wrote: VP]s, [>wrote: CP =IP
+Comp -Oc]s, [>liked: VP =DP]s, [>liked: PP =DP]s, [>liked: CP =IP]s, [>saw: VP =DP]s,
[>saw: VP =DP +O]s, [<loved: DP]s, [>hated: VP =DP +O]s, [>with: PP =DP +O]s, [>by: VP
=DP]s, [>by: PP =DP +O]s, [<above: DP]s, [>above: VP =CP]s, [>under: VP =DP]s,
[>under: PP =DP +O]s, [>under: VP =CP +Oc]s, [>that: VP]s, [>that: CP =IP]s, [>that:
PP =DP]s, [>that: VP =DP]s, [>knows: VP]s, [<knows: DP]s, [>says: VP =CP]s, [>thinks:
VP =CP]s, [>assumes: VP =CP]s, [>assumes: CP =IP +Comp -Oc]s]
2017_05_12__12_12_11: Energy: 5847
2017_05_12__12_12_11: Grammar: 1252
2017_05_12__12_12_11: Input: 4595
2017_05_12__12_12_11: Temperature: 40.602006
```

After over 500 iterations:

```
Iteration: 579
2017_05_12__12_13_39: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>George: DP]s, [>Elaine: DP]s, [>Kramer: DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s,
[>loved: VP =DP +O]s, [>hated: VP =DP +O]s,
[<with: PP =DP]s, [<by: PP =DP]s, [<above: PP =DP]s, [<under: PP =DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>thinks: VP =CP]s, [>assumes: VP =CP]s]
2017_05_12__12_13_39: Energy: 4399
2017_05_12__12_13_39: Grammar: 554
2017_05_12__12_13_39: Input: 3845
2017_05_12__12_13_39: Temperature: 17.558874
```

Word Learner

Head Initial Input

The relevant log file for this run is:

log_2017_05_13__18_57_05 - Word, Head-initial, PP & CP.txt

The generated input:

```
Input is: ['Elaine assumes that Elaine read', 'Jerry liked Jerry with Kramer', 'George
saw Elaine under Jerry', 'Kramer ran with Kramer', 'Jerry wrote', 'Kramer walked',
'Jerry saw Elaine with George', 'Kramer assumes that Jerry liked Kramer above George',
'George liked Kramer under George', 'Kramer wrote', 'George wrote by Jerry', 'Kramer
ran', 'George read', 'Elaine loved George above Jerry', 'Jerry read', 'Elaine hated
Jerry under Kramer', 'Jerry hated George with George', 'Jerry walked', 'Elaine loved
Elaine with Jerry', 'George loved Kramer', 'George hated George', 'George saw George',
'Elaine knows that George walked under Jerry', 'Kramer liked George above Elaine',
'Kramer saw Kramer', 'Elaine saw George above George', 'Jerry loved George under
Kramer', 'Jerry thinks that George liked Jerry under Kramer', 'George saw Jerry above
George', 'George loved Elaine', 'Jerry loved Kramer', 'Jerry assumes that Jerry wrote
under Kramer', 'Jerry saw Elaine', 'George assumes that George read under Kramer',
'George assumes that Kramer walked', 'Elaine knows that Elaine ran by Kramer', 'Kramer
ran', 'Jerry ran above Kramer', 'George hated Kramer', 'Kramer hated Jerry', 'Kramer
thinks that Elaine read with Kramer', 'Jerry knows that George saw Jerry', 'George
ran', 'Kramer walked', 'Kramer wrote with Jerry', 'Kramer loved Jerry', 'Elaine
wrote', 'Jerry walked by Kramer', 'Kramer hated Elaine above Kramer', 'George knows
that Kramer hated Jerry with Elaine']
```

The initial hypothesis:

```
Initial hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP
=IP -Comp]s, [>Jerry: VP]s, [>Jerry: DP]s, [>Jerry: CP =IP]s, [>Jerry: VP =DP]s,
[>Jerry: VP =CP]s, [>Jerry: PP =DP]s, [>Jerry: VP =DP +O]s, [>Jerry: PP =DP +O]s,
[>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s, [>George: VP]s, [>George: DP]s,
[>George: CP =IP]s, [>George: VP =DP]s, [>George: VP =CP]s, [>George: PP =DP]s,
[>George: VP =DP +O]s, [>George: PP =DP +O]s, [>George: VP =CP +Oc]s, [>George: CP =IP
+Comp -Oc]s, [>Elaine: VP]s, [>Elaine: DP]s, [>Elaine: CP =IP]s, [>Elaine: VP =DP]s,
[>Elaine: VP =CP]s, [>Elaine: PP =DP]s, [>Elaine: VP =DP +O]s, [>Elaine: PP =DP +O]s,
[>Elaine: VP =CP +Oc]s, [>Elaine: CP =IP +Comp -Oc]s, [>Kramer: VP]s, [>Kramer: DP]s,
[>Kramer: CP =IP]s, [>Kramer: VP =DP]s, [>Kramer: VP =CP]s, [>Kramer: PP =DP]s,
[>Kramer: VP =DP +O]s, [>Kramer: PP =DP +O]s, [>Kramer: VP =CP +Oc]s, [>Kramer: CP =IP
+Comp -Oc]s, [>ran: DP]s, [>ran: VP]s, [>ran: CP =IP]s, [>ran: VP =DP]s, [>ran: VP
=CP]s, [>ran: PP =DP]s, [>ran: VP =DP +O]s, [>ran: PP =DP +O]s, [>ran: VP =CP +Oc]s,
[>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [>walked: DP]s, [>walked: CP =IP]s,
[>walked: VP =DP]s, [>walked: VP =CP]s, [>walked: PP =DP]s, [>walked: VP =DP +O]s,
[>walked: PP =DP +O]s, [>walked: VP =CP +Oc]s, [>walked: CP =IP +Comp -Oc]s, [>read:
VP]s, [>read: DP]s, [>read: CP =IP]s, [>read: VP =DP]s, [>read: VP =CP]s, [>read: PP
=DP]s, [>read: VP =DP +O]s, [>read: PP =DP +O]s, [>read: VP =CP +Oc]s, [>read: CP =IP
+Comp -Oc]s, [>wrote: VP]s, [>wrote: DP]s, [>wrote: CP =IP]s, [>wrote: VP =DP]s,
[>wrote: VP =CP]s, [>wrote: PP =DP]s, [>wrote: VP =DP +O]s, [>wrote: PP =DP +O]s,
[>wrote: VP =CP +Oc]s, [>wrote: CP =IP +Comp -Oc]s, [>liked: VP]s, [>liked: DP]s,
[>liked: CP =IP]s, [>liked: VP =DP]s, [>liked: VP =CP]s, [>liked: PP =DP]s, [>liked:
VP =DP +O]s, [>liked: PP =DP +O]s, [>liked: VP =CP +Oc]s, [>liked: CP =IP +Comp -Oc]s,
[>saw: VP]s, [>saw: DP]s, [>saw: CP =IP]s, [>saw: VP =DP]s, [>saw: VP =CP]s, [>saw: PP
=DP]s, [>saw: VP =DP +O]s, [>saw: PP =DP +O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP
+Comp -Oc]s, [>loved: VP]s, [>loved: DP]s, [>loved: CP =IP]s, [>loved: VP =DP]s,
[>loved: VP =CP]s, [>loved: PP =DP]s, [>loved: VP =DP +O]s, [>loved: PP =DP +O]s,
[>loved: VP =CP +Oc]s, [>loved: CP =IP +Comp -Oc]s, [>hated: VP]s, [>hated: DP]s,
[>hated: CP =IP]s, [>hated: VP =DP]s, [>hated: VP =CP]s, [>hated: PP =DP]s, [>hated:
VP =DP +O]s, [>hated: PP =DP +O]s, [>hated: VP =CP +Oc]s, [>hated: CP =IP +Comp -Oc]s,
[>with: VP]s, [>with: DP]s, [>with: CP =IP]s, [>with: VP =DP]s, [>with: VP =CP]s,
[>with: PP =DP]s, [>with: VP =DP +O]s, [>with: PP =DP +O]s, [>with: VP =CP +Oc]s,
[>with: CP =IP +Comp -Oc]s, [>by: VP]s, [>by: DP]s, [>by: CP =IP]s, [>by: VP =DP]s,
[>by: VP =CP]s, [>by: PP =DP]s, [>by: VP =DP +O]s, [>by: PP =DP +O]s, [>by: VP =CP
+Oc]s, [>by: CP =IP +Comp -Oc]s, [>above: VP]s, [>above: DP]s, [>above: CP =IP]s,
```

```
[>above: VP =DP]s, [>above: VP =CP]s, [>above: PP =DP]s, [>above: VP =DP +O]s,
[>above: PP =DP +O]s, [>above: VP =CP +Oc]s, [>above: CP =IP +Comp -Oc]s, [>under:
DP]s, [>under: VP]s, [>under: CP =IP]s, [>under: VP =DP]s, [>under: VP =CP]s, [>under:
PP =DP]s, [>under: VP =DP +O]s, [>under: PP =DP +O]s, [>under: VP =CP +Oc]s, [>under:
CP =IP +Comp -Oc]s, [>that: VP]s, [>that: DP]s, [>that: CP =IP]s, [>that: VP =DP]s,
[>that: VP =CP]s, [>that: PP =DP]s, [>that: VP =DP +O]s, [>that: PP =DP +O]s, [>that:
VP =CP +Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP]s, [>knows: DP]s, [>knows: CP
=IP]s, [>knows: VP =DP]s, [>knows: VP =CP]s, [>knows: PP =DP]s, [>knows: VP =DP +O]s,
[>knows: PP =DP +O]s, [>knows: VP =CP +Oc]s, [>knows: CP =IP +Comp -Oc]s, [>thinks:
VP]s, [>thinks: DP]s, [>thinks: CP =IP]s, [>thinks: VP =DP]s, [>thinks: VP =CP]s,
[>thinks: PP =DP]s, [>thinks: VP =DP +O]s, [>thinks: PP =DP +O]s, [>thinks: VP =CP
+Oc]s, [>thinks: CP =IP +Comp -Oc]s, [>assumes: VP]s, [>assumes: DP]s, [>assumes: CP
=IP]s, [>assumes: VP =DP]s, [>assumes: VP =CP]s, [>assumes: PP =DP]s, [>assumes: VP
=DP +O]s, [>assumes: PP =DP +O]s, [>assumes: VP =CP +Oc]s, [>assumes: CP =IP +Comp -
Oc]s]
```

Initial energy:

```
Temperature: 100.000000
Energy: 10190
Grammar: 5107
Input: 5083
```

After 100 iterations:

```
Iteration: 100
2017_05_13__21_02_28: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [<Jerry: VP]s, [>Jerry: DP]s, [>Jerry: VP =CP]s, [<Jerry:
CP =IP]s, [<Jerry: VP =DP]s, [<Jerry: VP =CP]s, [>Jerry: PP =DP +O]s, [<George: VP]s,
[>George: DP]s, [>George: VP =CP]s, [>George: PP =DP]s, [<George: CP =IP]s, [<George:
VP =DP]s, [<George: VP =CP]s, [<George: VP =DP +O]s, [<Elaine: VP]s, [>Elaine: DP]s,
[>Elaine: PP =DP]s, [<Elaine: CP =IP]s, [<Elaine: VP =DP +O]s, [<Elaine: CP =IP +Comp
-Oc]s, [>Kramer: DP]s, [<Kramer: VP]s, [>Kramer: PP =DP]s, [<Kramer: CP =IP]s,
[<Kramer: VP =DP]s, [<Kramer: VP =DP +O]s, [<Kramer: VP =CP +Oc]s, [<Kramer: CP =IP
+Comp -Oc]s, [>ran: DP]s, [<ran: VP]s, [<ran: CP =IP]s, [>ran: CP =IP]s, [>ran: VP
=DP]s, [>ran: PP =DP]s, [<ran: VP =CP]s, [>ran: CP =IP +Comp -Oc]s, [<walked: VP]s,
[>walked: DP]s, [<walked: CP =IP]s, [<walked: VP =DP]s, [<walked: VP =CP]s, [>walked:
PP =DP +O]s, [<walked: VP =CP +Oc]s, [<walked: CP =IP +Comp -Oc]s, [<read: VP]s,
[>read: DP]s, [>read: PP =DP]s, [<read: CP =IP]s, [<read: VP =DP]s, [<read: VP =CP]s,
[<read: VP =CP +Oc]s, [<wrote: VP]s, [>wrote: DP]s, [>wrote: CP =IP]s, [<wrote: CP
=IP]s, [<wrote: VP =CP]s, [<wrote: PP =DP]s, [>wrote: PP =DP]s, [<liked: VP]s,
[>liked: DP]s, [<liked: VP =DP]s, [<liked: VP =CP]s, [<liked: VP =DP +O]s, [>liked: PP
=DP +O]s, [<liked: VP =CP +Oc]s, [<liked: VP =CP +Oc]s, [<liked: CP =IP +Comp -Oc]s,
[>liked: CP =IP +Comp -Oc]s, [>saw: DP]s, [>saw: PP =DP]s, [<saw: VP =CP]s, [<saw: VP
=DP +O]s, [>saw: PP =DP +O]s, [<saw: CP =IP +Comp -Oc]s, [<loved: VP]s, [>loved: CP
=IP]s, [>loved: PP =DP]s, [<loved: CP =IP]s, [<loved: VP =CP]s, [<loved: VP =DP +O]s,
[>loved: PP =DP +O]s, [<loved: VP =CP +Oc]s, [<hated: VP]s, [>hated: DP]s, [>hated: PP
=DP]s, [<hated: CP =IP]s, [<hated: VP =DP]s, [<hated: VP =CP]s, [<hated: CP =IP]s,
[>hated: PP =DP +O]s, [<hated: VP =CP +Oc]s, [<hated: CP =IP +Comp -Oc]s, [<with:
VP]s, [>with: PP =DP]s, [<with: CP =IP]s, [<with: VP =DP]s, [<with: VP =DP +O]s,
[>with: PP =DP +O]s, [<with: VP =CP +Oc]s, [<with: CP =IP +Comp -Oc]s, [<by: VP]s,
[>by: PP =DP]s, [<by: CP =IP]s, [<by: VP =CP]s, [>by: VP =DP +O]s, [>by: PP =DP +O]s,
[<by: VP =CP +Oc]s, [<above: VP]s, [>above: DP]s, [>above: CP =IP]s, [>above: PP
=DP]s, [<above: VP =DP]s, [<above: VP =CP]s, [<above: CP =IP]s, [>above: PP =DP +O]s,
[<above: VP =CP +Oc]s, [>under: PP =DP]s, [<under: CP =IP]s, [<under: VP =DP]s,
[<under: VP =CP]s, [<under: VP =DP +O]s, [>under: PP =DP +O]s, [<under: VP =CP +Oc]s,
[<under: CP =IP +Comp -Oc]s, [>that: DP]s, [>that: PP =DP]s, [<that: VP =DP]s, [<that:
VP =CP]s, [>that: PP =DP +O]s, [<that: VP =CP +Oc]s, [<that: CP =IP +Comp -Oc]s,
[<knows: VP]s, [>knows: DP]s, [>knows: CP =IP]s, [<knows: CP =IP]s, [>knows: VP =DP
+O]s, [<thinks: VP]s, [>thinks: DP]s, [>thinks: PP =DP]s, [<thinks: CP =IP]s,
[<thinks: VP =DP]s, [<thinks: VP =CP]s, [>thinks: PP =DP +O]s, [<thinks: VP =CP +Oc]s,
[<thinks: CP =IP +Comp -Oc]s, [<assumes: VP]s, [>assumes: DP]s, [<assumes: VP =DP]s,
[<assumes: VP =CP]s, [<assumes: VP =DP +O]s, [>assumes: PP =DP +O]s, [<assumes: CP =IP
+Comp -Oc]s]
2017_05_13__21_02_28: Energy: 9227
2017_05_13__21_02_28: Grammar: 3866
2017_05_13__21_02_28: Input: 5361
2017_05_13__21_02_28: Temperature: 74.048426
```

After 300 iterations:

```
Iteration: 300
```

```
2017_05_13__21_30_38: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: VP]s, [<Jerry: DP]s, [<Jerry: CP =IP]s, [<Jerry:
VP =DP]s, [>Jerry: PP =DP +O]s, [<George: DP]s, [<George: CP =IP]s, [<George: PP =DP
+O]s, [>George: VP =DP +O]s, [>Elaine: VP]s, [<Elaine: DP]s, [<Elaine: PP =DP +O]s,
[<Elaine: VP =CP +Oc]s, [<Kramer: DP]s, [>Kramer: DP]s, [>Kramer: PP =DP]s, [<Kramer:
CP =IP]s, [>Kramer: VP =DP +O]s, [>ran: VP]s, [<ran: VP =CP]s, [>ran: VP =DP]s, [<ran:
VP =DP +O]s, [>walked: VP]s, [<walked: DP]s, [>read: VP]s, [<read: DP]s, [<read: CP
=IP]s, [>wrote: VP]s, [>wrote: VP =CP +Oc]s, [<liked: DP]s, [<liked: VP =DP +O]s,
[>liked: VP =CP +Oc]s, [<saw: DP]s, [>saw: VP =DP]s, [>loved: VP =DP]s, [>loved: PP
=DP]s, [<loved: CP =IP +Comp -Oc]s, [<hated: DP]s, [>hated: VP =CP]s, [<hated: VP
=DP]s, [<hated: PP =DP]s, [>hated: CP =IP +Comp -Oc]s, [>with: PP =DP]s, [>with: CP
=IP]s, [>by: PP =DP]s, [<by: CP =IP]s, [>above: VP]s, [>above: VP =CP]s, [>above: PP
=DP]s, [<above: CP =IP]s, [>under: CP =IP]s, [>under: PP =DP]s, [>under: VP =CP +Oc]s,
[<that: DP]s, [>that: CP =IP]s, [<that: VP =CP +Oc]s, [<knows: CP =IP]s, [>knows: VP
=CP]s, [>thinks: VP =CP]s, [>thinks: PP =DP]s, [<thinks: PP =DP]s, [>thinks: CP =IP
+Comp -Oc]s, [<assumes: DP]s, [>assumes: VP =DP]s, [<assumes: VP =DP]s, [<assumes: CP
=IP +Comp -Oc]s]
2017_05_13__21_30_38: Energy: 6384
2017_05_13__21_30_38: Grammar: 1676
2017_05_13__21_30_38: Input: 4708
2017_05_13__21_30_38: Temperature: 40.602006
```

After over 1000 iterations:

```
Iteration: 1198
2017_05_13__21_38_07: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [<George: DP]s, [>Elaine: DP]s, [>Kramer: DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s, [>loved: VP =DP]s, [>hated: VP =DP]s,
[>with: PP =DP]s, [>by: PP =DP]s, [>above: PP =DP]s, [>under: PP =DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>thinks: VP =CP]s, [>assumes: VP =CP]s]
2017_05_13__21_38_07: Energy: 3999
2017_05_13__21_38_07: Grammar: 544
2017_05_13__21_38_07: Input: 3455
2017_05_13__21_38_07: Temperature: 2.734007
```

Head Final Input

The relevant log file for this run is:

log_2017_05_12__18_11_05 - Word, Head-final, PP & CP.txt

The generated input:

```
Input is: ['Kramer wrote Jerry under', 'Kramer Kramer loved Elaine above', 'George
walked Elaine with', 'Kramer Elaine walked that thinks Elaine by', 'Jerry walked',
'Kramer walked Kramer above', 'Elaine Elaine saw George by', 'Elaine walked', 'Kramer
George George hated that says George by', 'George Kramer Elaine loved that knows',
'Elaine ran Jerry under', 'Elaine Kramer hated Jerry by', 'Jerry Jerry walked that
assumes', 'George Elaine Elaine hated that says Kramer with', 'Elaine ran George
with', 'George Kramer ran Kramer with that says', 'Jerry Jerry read that assumes',
'George walked Jerry with', 'Jerry wrote', 'Elaine ran', 'Kramer Kramer wrote Jerry
with that knows', 'George Kramer read that knows', 'Jerry Elaine Jerry hated George
with that knows', 'Jerry Elaine George liked Jerry under that knows', 'Elaine read',
'Kramer Jerry liked George by', 'Kramer Elaine ran that thinks George under', 'George
wrote George above', 'Kramer George hated Jerry with', 'George Kramer liked Elaine
with', 'George George Kramer hated that knows', 'Jerry Kramer Elaine loved that
assumes Elaine with', 'Jerry George loved', 'Elaine wrote George by', 'Kramer wrote
Jerry by', 'Kramer walked', 'Jerry Jerry saw Jerry by', 'Kramer George loved Kramer
by', 'Jerry George liked', 'Kramer wrote Kramer by', 'Elaine Kramer Elaine saw that
says', 'Elaine Kramer liked Jerry above', 'George read Jerry with', 'Elaine Jerry
Jerry loved Elaine under that thinks', 'Elaine read', 'George walked Kramer above',
'Kramer George loved', 'Elaine ran', 'Kramer ran', 'Kramer read']
```

The initial hypothesis:

```
Initial hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP
=IP -Comp]s, [>Jerry: VP]s, [>Jerry: DP]s, [>Jerry: VP =CP]s, [>Jerry: PP =DP]s,
[>Jerry: CP =IP]s, [>Jerry: VP =DP]s, [>Jerry: PP =DP +O]s, [>Jerry: VP =DP +O]s,
[>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s, [>George: VP]s, [>George: DP]s,
[>George: VP =CP]s, [>George: PP =DP]s, [>George: CP =IP]s, [>George: VP =DP]s,
[>George: PP =DP +O]s, [>George: VP =DP +O]s, [>George: VP =CP +Oc]s, [>George: CP =IP
+Comp -Oc]s, [>Elaine: VP]s, [>Elaine: DP]s, [>Elaine: VP =CP]s, [>Elaine: PP =DP]s,
[>Elaine: CP =IP]s, [>Elaine: VP =DP]s, [>Elaine: PP =DP +O]s, [>Elaine: VP =DP +O]s,
[>Elaine: VP =CP +Oc]s, [>Elaine: CP =IP +Comp -Oc]s, [>Kramer: VP]s, [>Kramer: DP]s,
[>Kramer: PP =DP]s, [>Kramer: CP =IP]s, [>Kramer: VP =DP]s, [>Kramer: VP =CP]s,
[>Kramer: PP =DP +O]s, [>Kramer: VP =DP +O]s, [>Kramer: VP =CP +Oc]s, [>Kramer: CP =IP
+Comp -Oc]s, [>ran: VP]s, [>ran: DP]s, [>ran: VP =CP]s, [>ran: PP =DP]s, [>ran: CP
=IP]s, [>ran: VP =DP]s, [>ran: PP =DP +O]s, [>ran: VP =DP +O]s, [>ran: VP =CP +Oc]s,
[>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [>walked: DP]s, [>walked: VP =CP]s,
[>walked: PP =DP]s, [>walked: CP =IP]s, [>walked: VP =DP]s, [>walked: PP =DP +O]s,
[>walked: VP =DP +O]s, [>walked: VP =CP +Oc]s, [>walked: CP =IP +Comp -Oc]s, [>read:
VP]s, [>read: DP]s, [>read: VP =CP]s, [>read: PP =DP]s, [>read: CP =IP]s, [>read: VP
=DP]s, [>read: VP =DP +O]s, [>read: PP =DP +O]s, [>read: VP =CP +Oc]s, [>read: CP =IP
+Comp -Oc]s, [>wrote: VP]s, [>wrote: DP]s, [>wrote: PP =DP]s, [>wrote: CP =IP]s,
[>wrote: VP =DP]s, [>wrote: VP =CP]s, [>wrote: PP =DP +O]s, [>wrote: VP =DP +O]s,
[>wrote: VP =CP +Oc]s, [>wrote: CP =IP +Comp -Oc]s, [>liked: VP]s, [>liked: DP]s,
[>liked: VP =CP]s, [>liked: PP =DP]s, [>liked: CP =IP]s, [>liked: VP =DP]s, [>liked:
VP =DP +O]s, [>liked: PP =DP +O]s, [>liked: VP =CP +Oc]s, [>liked: CP =IP +Comp -Oc]s,
[>saw: VP]s, [>saw: DP]s, [>saw: VP =CP]s, [>saw: PP =DP]s, [>saw: CP =IP]s, [>saw: VP
=DP]s, [>saw: PP =DP +O]s, [>saw: VP =DP +O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP
+Comp -Oc]s, [>loved: VP]s, [>loved: DP]s, [>loved: VP =CP]s, [>loved: PP =DP]s,
[>loved: CP =IP]s, [>loved: VP =DP]s, [>loved: PP =DP +O]s, [>loved: VP =DP +O]s,
[>loved: VP =CP +Oc]s, [>loved: CP =IP +Comp -Oc]s, [>hated: VP]s, [>hated: DP]s,
[>hated: VP =CP]s, [>hated: PP =DP]s, [>hated: CP =IP]s, [>hated: VP =DP]s, [>hated:
PP =DP +O]s, [>hated: VP =DP +O]s, [>hated: VP =CP +Oc]s, [>hated: CP =IP +Comp -Oc]s,
[>with: VP]s, [>with: DP]s, [>with: VP =CP]s, [>with: PP =DP]s, [>with: CP =IP]s,
[>with: VP =DP]s, [>with: PP =DP +O]s, [>with: VP =DP +O]s, [>with: VP =CP +Oc]s,
[>with: CP =IP +Comp -Oc]s, [>by: VP]s, [>by: DP]s, [>by: VP =CP]s, [>by: PP =DP]s,
[>by: CP =IP]s, [>by: VP =DP]s, [>by: PP =DP +O]s, [>by: VP =DP +O]s, [>by: VP =CP
+Oc]s, [>by: CP =IP +Comp -Oc]s, [>above: VP]s, [>above: DP]s, [>above: VP =CP]s,
[>above: PP =DP]s, [>above: CP =IP]s, [>above: VP =DP]s, [>above: PP =DP +O]s,
[>above: VP =DP +O]s, [>above: VP =CP +Oc]s, [>above: CP =IP +Comp -Oc]s, [>under:
VP]s, [>under: DP]s, [>under: VP =CP]s, [>under: PP =DP]s, [>under: CP =IP]s, [>under:
VP =DP]s, [>under: PP =DP +O]s, [>under: VP =DP +O]s, [>under: VP =CP +Oc]s, [>under:
CP =IP +Comp -Oc]s, [>that: VP]s, [>that: DP]s, [>that: VP =CP]s, [>that: PP =DP]s,
[>that: CP =IP]s, [>that: VP =DP]s, [>that: PP =DP +O]s, [>that: VP =DP +O]s, [>that:
VP =CP +Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP]s, [>knows: DP]s, [>knows: VP
=CP]s, [>knows: PP =DP]s, [>knows: CP =IP]s, [>knows: VP =DP]s, [>knows: PP =DP +O]s,
[>knows: VP =DP +O]s, [>knows: VP =CP +Oc]s, [>knows: CP =IP +Comp -Oc]s, [>says:
VP]s, [>says: DP]s, [>says: VP =CP]s, [>says: PP =DP]s, [>says: CP =IP]s, [>says: VP
=DP]s, [>says: PP =DP +O]s, [>says: VP =DP +O]s, [>says: VP =CP +Oc]s, [>says: CP =IP
+Comp -Oc]s, [>thinks: VP]s, [>thinks: DP]s, [>thinks: VP =CP]s, [>thinks: PP =DP]s,
[>thinks: CP =IP]s, [>thinks: VP =DP]s, [>thinks: PP =DP +O]s, [>thinks: VP =DP +O]s,
[>thinks: VP =CP +Oc]s, [>thinks: CP =IP +Comp -Oc]s, [>assumes: VP]s, [>assumes:
DP]s, [>assumes: VP =CP]s, [>assumes: PP =DP]s, [>assumes: CP =IP]s, [>assumes: VP
=DP]s, [>assumes: PP =DP +O]s, [>assumes: VP =DP +O]s, [>assumes: VP =CP +Oc]s,
[>assumes: CP =IP +Comp -Oc]s]
```

Initial energy:

```
Temperature: 100.000000
Energy: 10933
Grammar: 5357
Input: 5576
```

After 100 iterations:

```
Iteration: 100
2017_05_12__23_00_32: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: VP]s, [>Jerry: DP]s, [<Jerry: PP =DP]s, [>Jerry:
VP =CP]s, [<Jerry: CP =IP]s, [>Jerry: VP =DP]s, [<Jerry: PP =DP +O]s, [>Jerry: VP =DP
+O]s, [>Jerry: VP =CP +Oc]s, [<Jerry: CP =IP +Comp -Oc]s, [>George: VP]s, [>George:
DP]s, [<George: PP =DP]s, [>George: VP =CP]s, [<George: PP =DP +O]s, [>George: VP =DP
+O]s, [>George: VP =CP +Oc]s, [>George: CP =IP +Comp -Oc]s, [<George: CP =IP +Comp -
Oc]s, [>Elaine: VP]s, [>Elaine: DP]s, [<Elaine: PP =DP]s, [>Elaine: VP =CP]s,
[<Elaine: CP =IP]s, [<Elaine: PP =DP +O]s, [<Elaine: VP =DP +O]s, [>Elaine: VP =CP
```

```
+Oc]s, [<Elaine: CP =IP +Comp -Oc]s, [>Kramer: DP]s, [<Kramer: VP =CP]s, [<Kramer: PP
=DP]s, [>Kramer: VP =DP]s, [>Kramer: VP =CP]s, [<Kramer: VP =DP]s, [<Kramer: CP =IP
+Comp -Oc]s, [>ran: VP]s, [<ran: PP =DP]s, [>ran: VP =CP]s, [<ran: CP =IP]s, [<ran: VP
=DP]s, [<ran: PP =DP +O]s, [>ran: VP =CP +Oc]s, [>ran: VP =CP +Oc]s, [<ran: CP =IP
+Comp -Oc]s, [>walked: VP]s, [>walked: DP]s, [<walked: PP =DP]s, [>walked: VP =DP]s,
[<walked: PP =DP +O]s, [>walked: VP =DP +O]s, [>walked: VP =CP +Oc]s, [<walked: CP =IP
+Comp -Oc]s, [>read: VP]s, [>read: DP]s, [>read: VP =CP]s, [>read: CP =IP]s, [>read:
VP =DP]s, [<read: PP =DP +O]s, [>read: VP =CP +Oc]s, [>read: VP =CP +Oc]s, [>wrote:
VP]s, [>wrote: DP]s, [>wrote: VP =CP]s, [<wrote: PP =DP +O]s, [>wrote: VP =CP +Oc]s,
[>liked: DP]s, [<liked: PP =DP]s, [>liked: VP =CP]s, [>liked: VP =DP +O]s, [>liked: VP
=CP +Oc]s, [>saw: VP]s, [>saw: DP]s, [<saw: PP =DP]s, [>saw: VP =CP]s, [>saw: VP
=DP]s, [<saw: CP =IP]s, [<saw: VP =DP]s, [>saw: VP =CP +Oc]s, [<saw: CP =IP +Comp -
Oc]s, [>loved: VP]s, [>loved: VP =CP]s, [<loved: CP =IP]s, [>loved: VP =DP]s, [>loved:
VP =DP +O]s, [>loved: VP =CP +Oc]s, [<loved: CP =IP +Comp -Oc]s, [>hated: VP]s,
[<hated: PP =DP]s, [>hated: VP =DP]s, [<hated: CP =IP]s, [>hated: VP =DP +O]s,
[>hated: VP =CP +Oc]s, [<hated: CP =IP +Comp -Oc]s, [>with: VP]s, [>with: DP]s,
[<with: PP =DP]s, [>with: VP =CP]s, [>with: VP =DP]s, [>with: PP =DP]s, [<with: PP =DP
+O]s, [>with: VP =CP +Oc]s, [<with: CP =IP +Comp -Oc]s, [>by: VP]s, [>by: DP]s, [<by:
PP =DP]s, [<by: CP =IP]s, [<by: PP =DP +O]s, [<by: VP =CP +Oc]s, [>above: VP]s,
[>above: DP]s, [<above: PP =DP]s, [>above: VP =CP]s, [<above: VP =DP]s, [<above: PP
=DP +O]s, [<above: CP =IP +Comp -Oc]s, [>under: VP]s, [>under: DP]s, [<under: PP
=DP]s, [>under: VP =CP]s, [<under: CP =IP]s, [>under: VP =CP +Oc]s, [<under: CP =IP
+Comp -Oc]s, [>that: VP]s, [>that: DP]s, [<that: PP =DP]s, [>that: VP =CP]s, [>that:
VP =DP]s, [<that: CP =IP]s, [<that: PP =DP +O]s, [>knows: VP]s, [>knows: DP]s,
[<knows: PP =DP]s, [<knows: CP =IP]s, [<knows: PP =DP +O]s, [>knows: VP =DP +O]s,
[<knows: CP =IP +Comp -Oc]s, [>says: VP]s, [<says: DP]s, [<says: PP =DP]s, [>says: VP
=DP]s, [<says: CP =IP]s, [<says: PP =DP +O]s, [<says: CP =IP +Comp -Oc]s, [>thinks:
VP]s, [>thinks: DP]s, [<thinks: PP =DP]s, [>thinks: PP =DP]s, [<thinks: CP =IP]s,
[>thinks: VP =DP +O]s, [>thinks: PP =DP +O]s, [>thinks: VP =CP +Oc]s, [>assumes: VP]s,
[>assumes: DP]s, [<assumes: VP =CP]s, [<assumes: PP =DP]s, [>assumes: VP =DP]s,
[<assumes: CP =IP]s, [<assumes: PP =DP +O]s, [>assumes: VP =DP +O]s, [<assumes: VP =CP
+Oc]s, [<assumes: CP =IP +Comp -Oc]s]
2017_05_12__23_00_32: Energy: 9844
2017_05_12__23_00_32: Grammar: 4086
2017_05_12__23_00_32: Input: 5758
2017_05_12__23_00_32: Temperature: 74.048426
```

After 300 iterations:

```
Iteration: 300
2017_05_12__23_25_23: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: DP]s, [<Jerry: VP =CP]s, [>Jerry: VP =CP]s,
[<Jerry: CP =IP]s, [>George: VP]s, [>George: DP]s, [<George: PP =DP]s, [>George: VP
=DP +O]s, [<George: PP =DP +O]s, [>Elaine: DP]s, [>Elaine: VP =CP]s, [<Elaine: CP
=IP]s, [>Elaine: CP =IP]s, [<Elaine: PP =DP +O]s, [>Elaine: VP =CP +Oc]s, [>Elaine: CP
=IP +Comp -Oc]s, [>Kramer: DP]s, [>Kramer: VP =CP]s, [>ran: VP]s, [<ran: VP =DP]s,
[>walked: VP]s, [>walked: VP =CP +Oc]s, [>read: VP]s, [>read: PP =DP]s, [>read: CP
=IP]s, [>read: VP =DP]s, [>wrote: VP]s, [<wrote: VP =DP]s, [<liked: VP =DP]s, [>liked:
VP =CP]s, [>saw: VP]s, [>saw: DP]s, [>loved: VP]s, [>loved: VP =DP]s, [<loved: VP
=DP]s, [>loved: PP =DP +O]s, [>hated: VP]s, [<hated: PP =DP]s, [>hated: VP =DP]s,
[>hated: VP =DP +O]s, [>with: VP]s, [<with: PP =DP]s, [>with: PP =DP]s, [>by: VP]s,
[<by: PP =DP]s, [<by: CP =IP]s, [<by: VP =CP]s, [<by: PP =DP]s, [<by: VP =DP]s, [<by:
VP =CP +Oc]s, [<above: PP =DP]s, [>above: VP =CP]s, [<above: VP =DP]s, [>under: VP]s,
[<under: CP =IP]s, [>under: PP =DP +O]s, [>that: VP]s, [>that: DP]s, [>that: VP =DP]s,
[<that: CP =IP]s, [<that: VP =CP +Oc]s, [>knows: VP]s, [>knows: DP]s, [<knows: PP
=DP]s, [<knows: CP =IP]s, [>knows: VP =CP +Oc]s, [<says: DP]s, [<says: PP =DP]s,
[>says: VP =DP]s, [<says: CP =IP]s, [<thinks: PP =DP]s, [<thinks: CP =IP]s, [<assumes:
VP =CP]s, [<assumes: PP =DP]s, [<assumes: VP =DP]s, [<assumes: PP =DP +O]s, [<assumes:
CP =IP +Comp -Oc]s]
2017 05 12  23 25 23: Energy: 6956
2017_05_12__23_25_23: Grammar: 1914
2017_05_12__23_25_23: Input: 5042
2017_05_12__23_25_23: Temperature: 40.602006
```

After over 900 iterations:

```
Iteration: 940
2017_05_12__23_31_29: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>George: DP]s, [>Elaine: DP]s, [>Kramer: DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[<liked: VP =DP]s, [<saw: VP =DP]s, [<loved: VP =DP]s, [<hated: VP =DP]s,
[<with: PP =DP]s, [<by: PP =DP]s, [<above: PP =DP]s, [<under: PP =DP]s,
```

```
[<that: CP =IP]s,
[<knows: VP =CP]s, [<says: VP =CP]s, [<thinks: VP =CP]s, [<assumes: VP =CP]s]
2017_05_12__23_31_29: Energy: 4388
2017_05_12__23_31_29: Grammar: 568
2017_05_12__23_31_29: Input: 3820
2017_05_12__23_31_29: Temperature: 5.935384
```

Mixed Category Input

The relevant log file for this run is:

log_2017_05_10__19_25_25 - Word, Mixed-category, PP & CP.txt

The generated input:

```
Input is: ['Kramer thinks that Kramer wrote George with', 'Kramer hated Jerry', 'Jerry
ran Jerry under', 'Kramer loved George', 'Jerry wrote', 'Jerry thinks that Kramer
loved Jerry George under', 'Elaine wrote', 'Kramer wrote Elaine under', 'Jerry assumes
that George hated Jerry Jerry under', 'Jerry saw George', 'George assumes that Elaine
liked Jerry George with', 'Kramer thinks that George loved Elaine', 'Jerry knows that
Kramer walked George under', 'Jerry liked George', 'Kramer hated George George under',
'George loved Jerry Jerry by', 'George ran Jerry above', 'George read', 'Kramer loved
Elaine', 'Kramer wrote', 'Kramer walked', 'Jerry knows that George loved Jerry',
'Elaine loved George Kramer under', 'Kramer hated Elaine', 'Jerry ran Elaine with',
'Elaine says that Elaine liked Jerry', 'Elaine ran Elaine under', 'George read Kramer
under', 'Jerry hated Jerry', 'Elaine hated Kramer Kramer by', 'Kramer assumes that
Jerry walked Jerry above', 'George says that Elaine read George with', 'Jerry assumes
that Elaine saw Jerry Jerry with', 'Jerry loved Kramer', 'Jerry saw Elaine Jerry
under', 'Jerry liked Kramer Kramer by', 'Elaine hated Kramer Kramer above', 'Jerry
ran', 'George says that Kramer ran George above', 'George read Kramer by', 'George
wrote', 'Elaine ran', 'Kramer knows that George wrote', 'Kramer saw Jerry Kramer
under', 'Elaine says that Jerry saw Elaine George by', 'Jerry read Elaine by', 'George
ran Kramer with', 'Elaine assumes that Jerry liked Jerry', 'Kramer says that George
liked Jerry', 'Jerry ran Elaine above']
```

The initial hypothesis:

```
Initial hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP
=IP -Comp]s, [>Jerry: VP]s, [>Jerry: DP]s, [>Jerry: PP =DP]s, [>Jerry: CP =IP]s,
[>Jerry: VP =DP]s, [>Jerry: VP =CP]s, [>Jerry: VP =DP +O]s, [>Jerry: PP =DP +O]s,
[>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s, [>George: VP]s, [>George: DP]s,
[>George: PP =DP]s, [>George: CP =IP]s, [>George: VP =DP]s, [>George: VP =CP]s,
[>George: VP =DP +O]s, [>George: PP =DP +O]s, [>George: VP =CP +Oc]s, [>George: CP =IP
+Comp -Oc]s, [>Elaine: VP]s, [>Elaine: DP]s, [>Elaine: PP =DP]s, [>Elaine: CP =IP]s,
[>Elaine: VP =DP]s, [>Elaine: VP =CP]s, [>Elaine: VP =DP +O]s, [>Elaine: PP =DP +O]s,
[>Elaine: VP =CP +Oc]s, [>Elaine: CP =IP +Comp -Oc]s, [>Kramer: VP]s, [>Kramer: DP]s,
[>Kramer: PP =DP]s, [>Kramer: CP =IP]s, [>Kramer: VP =DP]s, [>Kramer: VP =CP]s,
[>Kramer: VP =DP +O]s, [>Kramer: PP =DP +O]s, [>Kramer: VP =CP +Oc]s, [>Kramer: CP =IP
+Comp -Oc]s, [>ran: VP]s, [>ran: DP]s, [>ran: PP =DP]s, [>ran: CP =IP]s, [>ran: VP
=DP]s, [>ran: VP =CP]s, [>ran: VP =DP +O]s, [>ran: PP =DP +O]s, [>ran: VP =CP +Oc]s,
[>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [>walked: DP]s, [>walked: PP =DP]s,
[>walked: CP =IP]s, [>walked: VP =DP]s, [>walked: VP =CP]s, [>walked: VP =DP +O]s,
[>walked: PP =DP +O]s, [>walked: VP =CP +Oc]s, [>walked: CP =IP +Comp -Oc]s, [>read:
VP]s, [>read: DP]s, [>read: PP =DP]s, [>read: CP =IP]s, [>read: VP =DP]s, [>read: VP
=CP]s, [>read: VP =DP +O]s, [>read: PP =DP +O]s, [>read: VP =CP +Oc]s, [>read: CP =IP
+Comp -Oc]s, [>wrote: VP]s, [>wrote: DP]s, [>wrote: PP =DP]s, [>wrote: CP =IP]s,
[>wrote: VP =DP]s, [>wrote: VP =CP]s, [>wrote: VP =DP +O]s, [>wrote: PP =DP +O]s,
[>wrote: VP =CP +Oc]s, [>wrote: CP =IP +Comp -Oc]s, [>liked: VP]s, [>liked: DP]s,
[>liked: PP =DP]s, [>liked: CP =IP]s, [>liked: VP =DP]s, [>liked: VP =CP]s, [>liked:
VP =DP +O]s, [>liked: PP =DP +O]s, [>liked: VP =CP +Oc]s, [>liked: CP =IP +Comp -Oc]s,
[>saw: VP]s, [>saw: DP]s, [>saw: PP =DP]s, [>saw: CP =IP]s, [>saw: VP =DP]s, [>saw: VP
=CP]s, [>saw: VP =DP +O]s, [>saw: PP =DP +O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP
+Comp -Oc]s, [>loved: VP]s, [>loved: DP]s, [>loved: PP =DP]s, [>loved: CP =IP]s,
[>loved: VP =DP]s, [>loved: VP =CP]s, [>loved: VP =DP +O]s, [>loved: PP =DP +O]s,
[>loved: VP =CP +Oc]s, [>loved: CP =IP +Comp -Oc]s, [>hated: VP]s, [>hated: DP]s,
[>hated: PP =DP]s, [>hated: CP =IP]s, [>hated: VP =DP]s, [>hated: VP =CP]s, [>hated:
VP =DP +O]s, [>hated: PP =DP +O]s, [>hated: VP =CP +Oc]s, [>hated: CP =IP +Comp -Oc]s,
[>with: VP]s, [>with: DP]s, [>with: PP =DP]s, [>with: CP =IP]s, [>with: VP =DP]s,
```

```
[>with: VP =CP]s, [>with: VP =DP +O]s, [>with: PP =DP +O]s, [>with: VP =CP +Oc]s,
[>with: CP =IP +Comp -Oc]s, [>by: VP]s, [>by: DP]s, [>by: PP =DP]s, [>by: CP =IP]s,
[>by: VP =DP]s, [>by: VP =CP]s, [>by: VP =DP +O]s, [>by: PP =DP +O]s, [>by: VP =CP
+Oc]s, [>by: CP =IP +Comp -Oc]s, [>above: VP]s, [>above: DP]s, [>above: PP =DP]s,
[>above: CP =IP]s, [>above: VP =DP]s, [>above: VP =CP]s, [>above: VP =DP +O]s,
[>above: PP =DP +O]s, [>above: VP =CP +Oc]s, [>above: CP =IP +Comp -Oc]s, [>under:
VP]s, [>under: DP]s, [>under: PP =DP]s, [>under: CP =IP]s, [>under: VP =DP]s, [>under:
VP =CP]s, [>under: VP =DP +O]s, [>under: PP =DP +O]s, [>under: VP =CP +Oc]s, [>under:
CP =IP +Comp -Oc]s, [>that: VP]s, [>that: DP]s, [>that: PP =DP]s, [>that: CP =IP]s,
[>that: VP =DP]s, [>that: VP =CP]s, [>that: VP =DP +O]s, [>that: PP =DP +O]s, [>that:
VP =CP +Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP]s, [>knows: DP]s, [>knows: PP
=DP]s, [>knows: CP =IP]s, [>knows: VP =DP]s, [>knows: VP =CP]s, [>knows: VP =DP +O]s,
[>knows: PP =DP +O]s, [>knows: VP =CP +Oc]s, [>knows: CP =IP +Comp -Oc]s, [>says:
VP]s, [>says: DP]s, [>says: PP =DP]s, [>says: VP =CP]s, [>says: CP =IP]s, [>says: VP
=DP]s, [>says: VP =DP +O]s, [>says: PP =DP +O]s, [>says: VP =CP +Oc]s, [>says: CP =IP
+Comp -Oc]s, [>thinks: VP]s, [>thinks: DP]s, [>thinks: PP =DP]s, [>thinks: CP =IP]s,
[>thinks: VP =DP]s, [>thinks: VP =CP]s, [>thinks: VP =DP +O]s, [>thinks: PP =DP +O]s,
[>thinks: VP =CP +Oc]s, [>thinks: CP =IP +Comp -Oc]s, [>assumes: VP]s, [>assumes:
DP]s, [>assumes: PP =DP]s, [>assumes: CP =IP]s, [>assumes: VP =DP]s, [>assumes: VP
=CP]s, [>assumes: VP =DP +O]s, [>assumes: PP =DP +O]s, [>assumes: VP =CP +Oc]s,
[>assumes: CP =IP +Comp -Oc]s]
```

Initial energy:

```
Temperature: 100.000000
Energy: 10865
Grammar: 5357
Input: 5508
```

After 100 iterations:

```
Iteration: 100
2017_05_11__00_16_26: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: VP]s, [>Jerry: DP]s, [<Jerry: CP =IP]s, [<Jerry:
PP =DP]s, [>Jerry: VP =DP]s, [>Jerry: VP =CP]s, [<Jerry: PP =DP +O]s, [>George: VP]s,
[>George: DP]s, [<George: CP =IP]s, [<George: PP =DP]s, [>George: VP =DP]s, [>George:
VP =DP +O]s, [>George: VP =DP +O]s, [>George: VP =CP +Oc]s, [<George: VP =CP +Oc]s,
[<George: CP =IP +Comp -Oc]s, [>Elaine: DP]s, [<Elaine: PP =DP]s, [<Elaine: PP =DP
+O]s, [>Elaine: VP =DP +O]s, [>Elaine: VP =CP +Oc]s, [<Elaine: CP =IP +Comp -Oc]s,
[>Kramer: VP]s, [>Kramer: DP]s, [<Kramer: VP =DP]s, [<Kramer: PP =DP]s, [>Kramer: PP
=DP]s, [>Kramer: VP =DP]s, [<Kramer: CP =IP +Comp -Oc]s, [>ran: VP]s, [>ran: DP]s,
[<ran: CP =IP]s, [>ran: CP =IP]s, [>ran: VP =CP]s, [<ran: PP =DP +O]s, [>ran: VP =DP
+O]s, [>ran: PP =DP +O]s, [<ran: VP =DP +O]s, [>ran: VP =CP +Oc]s, [>walked: VP]s,
[>walked: DP]s, [<walked: CP =IP]s, [<walked: PP =DP]s, [>walked: VP =DP]s, [>walked:
VP =CP]s, [<walked: PP =DP +O]s, [>walked: VP =DP +O]s, [>walked: VP =CP +Oc]s,
[<walked: CP =IP +Comp -Oc]s, [>read: VP]s, [>read: DP]s, [<read: CP =IP]s, [<read: PP
=DP]s, [>read: VP =CP]s, [>read: CP =IP]s, [>read: VP =DP +O]s, [<read: VP =DP +O]s,
[>read: PP =DP +O]s, [>read: VP =CP +Oc]s, [<read: CP =IP +Comp -Oc]s, [>wrote: VP]s,
[>wrote: PP =DP]s, [>wrote: VP =DP]s, [>wrote: VP =CP]s, [>wrote: VP =DP +O]s,
[>wrote: PP =DP +O]s, [<wrote: PP =DP +O]s, [>wrote: VP =CP +Oc]s, [<wrote: CP =IP
+Comp -Oc]s, [>liked: VP]s, [>liked: DP]s, [<liked: CP =IP]s, [<liked: PP =DP]s,
[>liked: VP =DP]s, [<liked: PP =DP +O]s, [>liked: VP =DP +O]s, [<liked: CP =IP +Comp -
Oc]s, [>saw: VP]s, [>saw: DP]s, [<saw: CP =IP]s, [>saw: VP =DP]s, [<saw: PP =DP]s,
[<saw: PP =DP +O]s, [>saw: VP =DP +O]s, [<saw: VP =CP +Oc]s, [>loved: VP]s, [>loved:
DP]s, [<loved: CP =IP]s, [>loved: VP =DP]s, [>loved: VP =CP]s, [>loved: CP =IP]s,
[<loved: PP =DP +O]s, [>loved: PP =DP +O]s, [>loved: VP =CP +Oc]s, [<loved: CP =IP
+Comp -Oc]s, [>hated: VP]s, [<hated: CP =IP]s, [>hated: VP =DP]s, [<hated: PP =DP
+O]s, [>hated: VP =DP +O]s, [<hated: CP =IP +Comp -Oc]s, [>with: VP]s, [>with: DP]s,
[<with: CP =IP]s, [<with: PP =DP]s, [>with: VP =DP]s, [>with: VP =DP +O]s, [>with: VP
=CP +Oc]s, [<with: CP =IP +Comp -Oc]s, [>by: VP]s, [>by: DP]s, [<by: CP =IP]s, [<by:
VP =DP]s, [<by: PP =DP]s, [>by: VP =DP]s, [<by: CP =IP]s, [>by: PP =DP]s, [<by: PP =DP
+O]s, [<by: CP =IP +Comp -Oc]s, [>above: VP]s, [>above: DP]s, [<above: CP =IP]s,
[<above: PP =DP]s, [>above: VP =CP]s, [<above: PP =DP +O]s, [>above: VP =CP +Oc]s,
[<above: CP =IP +Comp -Oc]s, [>under: VP]s, [>under: DP]s, [<under: CP =IP]s, [<under:
VP =DP]s, [<under: PP =DP]s, [>under: VP =DP]s, [>under: VP =CP]s, [<under: PP =DP
+O]s, [>under: VP =CP +Oc]s, [<under: CP =IP +Comp -Oc]s, [>that: VP]s, [>that: DP]s,
[<that: CP =IP]s, [<that: PP =DP]s, [>that: VP =DP]s, [<that: CP =IP]s, [>that: VP =DP
+O]s, [>that: VP =CP +Oc]s, [>knows: VP]s, [>knows: DP]s, [<knows: CP =IP]s, [<knows:
PP =DP]s, [>knows: VP =DP]s, [>knows: VP =CP]s, [<knows: PP =DP +O]s, [>knows: VP =DP
+O]s, [>knows: VP =CP +Oc]s, [<knows: CP =IP +Comp -Oc]s, [>says: VP]s, [>says: DP]s,
[>says: VP =CP]s, [>says: VP =DP]s, [<says: PP =DP +O]s, [>says: VP =DP +O]s, [>says:
VP =CP +Oc]s, [<says: CP =IP +Comp -Oc]s, [>thinks: DP]s, [<thinks: CP =IP]s,
[<thinks: PP =DP]s, [>thinks: VP =DP]s, [>thinks: VP =CP]s, [<thinks: PP =DP +O]s,
```

```
[>thinks: VP =DP +O]s, [>thinks: VP =CP +Oc]s, [<thinks: CP =IP +Comp -Oc]s,
[>assumes: VP]s, [>assumes: DP]s, [<assumes: PP =DP]s, [>assumes: VP =CP]s, [>assumes:
VP =DP +O]s, [>assumes: VP =CP +Oc]s, [<assumes: CP =IP +Comp -Oc]s]
2017_05_11__00_16_26: Energy: 10235
2017_05_11__00_16_26: Grammar: 4589
2017_05_11__00_16_26: Input: 5646
2017_05_11__00_16_26: Temperature: 74.048426
```

After 300 iterations:

```
Iteration: 300
2017_05_11__01_07_17: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: VP]s, [>Jerry: DP]s, [<Jerry: CP =IP]s, [<Jerry:
PP =DP]s, [>Jerry: VP =DP]s, [>Jerry: VP =CP]s, [<Jerry: PP =DP +O]s, [>George: VP]s,
[>George: DP]s, [<George: PP =DP]s, [<George: VP =DP]s, [<George: VP =DP +O]s,
[>George: VP =CP +Oc]s, [<George: CP =IP +Comp -Oc]s, [>Elaine: DP]s, [<Elaine: VP
=DP]s, [<Elaine: VP =CP]s, [<Elaine: PP =DP]s, [<Elaine: PP =DP +O]s, [>Kramer: VP]s,
[<Kramer: DP]s, [<Kramer: VP =DP]s, [>Kramer: CP =IP]s, [>Kramer: VP =DP]s, [>ran:
VP]s, [>ran: DP]s, [<ran: CP =IP]s, [>ran: VP =CP]s, [>ran: PP =DP +O]s, [>walked:
VP]s, [>walked: DP]s, [>walked: VP =DP]s, [>walked: VP =DP +O]s, [<walked: VP =CP
+Oc]s, [<walked: CP =IP +Comp -Oc]s, [>read: VP]s, [<read: PP =DP]s, [<read: PP =DP]s,
[>read: VP =DP +O]s, [>wrote: VP]s, [>wrote: VP =CP +Oc]s, [>liked: PP =DP]s, [>liked:
CP =IP]s, [>liked: VP =DP]s, [>liked: VP =DP +O]s, [>liked: PP =DP +O]s, [>saw: VP]s,
[>saw: DP]s, [>saw: VP =DP]s, [>saw: VP =CP]s, [<saw: CP =IP]s, [<saw: CP =IP +Comp -
Oc]s, [>loved: DP]s, [<loved: CP =IP]s, [>loved: CP =IP]s, [>loved: VP =CP]s, [>hated:
VP]s, [<hated: CP =IP]s, [>hated: CP =IP]s, [>hated: VP =DP]s, [>hated: PP =DP]s,
[<with: VP =CP]s, [<with: PP =DP]s, [<by: CP =IP]s, [<by: PP =DP]s, [>by: PP =DP]s,
[>above: VP]s, [>above: DP]s, [<above: CP =IP]s, [<above: PP =DP]s, [>under: VP]s,
[>under: DP]s, [<under: PP =DP]s, [>under: VP =DP]s, [<under: VP =CP]s, [<under: PP
=DP +O]s, [>under: VP =DP +O]s, [>under: VP =DP +O]s, [>under: VP =CP +Oc]s, [>under:
CP =IP +Comp -Oc]s, [>that: VP]s, [>that: DP]s, [>that: VP =CP +Oc]s, [>knows: VP]s,
[<knows: VP =DP]s, [<knows: VP =CP]s, [<knows: PP =DP +O]s, [<knows: CP =IP +Comp -
Oc]s, [>says: VP =CP]s, [>says: VP =DP]s, [>says: VP =DP +O]s, [>thinks: DP]s,
[<thinks: PP =DP]s, [>thinks: VP =DP +O]s, [>thinks: VP =CP +Oc]s, [<thinks: CP =IP
+Comp -Oc]s, [>assumes: VP]s, [>assumes: VP =CP]s, [>assumes: VP =DP +O]s, [>assumes:
VP =CP +Oc]s, [<assumes: CP =IP +Comp -Oc]s]
2017_05_11__01_07_17: Energy: 7650
2017_05_11__01_07_17: Grammar: 2545
2017_05_11__01_07_17: Input: 5105
2017_05_11__01_07_17: Temperature: 40.602006
```

After over 1000 iterations:

```
Iteration: 1300
2017_05_11__01_27_31: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>George: DP]s, [>Elaine: DP]s, [<Kramer: DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s, [>loved: VP =DP]s, [>hated: VP =DP]s,
[<with: PP =DP]s, [<by: PP =DP]s, [<above: PP =DP]s, [<under: PP =DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>thinks: VP =CP]s, [>assumes: VP =CP]s]
2017_05_11__01_27_31: Energy: 4342
2017_05_11__01_27_31: Grammar: 568
2017_05_11__01_27_31: Input: 3774
2017_05_11__01_27_31: Temperature: 2.012361
```

Mixed Word Input

The relevant log file for this run is:

log_2017_05_13__02_56_34 - Word, Mixed-word, PP & CP.txt

The generated input:

```
Input is: ['Elaine George loved', 'Kramer wrote', 'Jerry assumes that Kramer saw
Kramer Kramer with', 'Elaine knows that Jerry ran', 'Kramer wrote George under',
'Elaine walked', 'Kramer says that Elaine ran', 'Elaine saw Kramer', 'Elaine Elaine
loved', 'Kramer Jerry hated', 'George knows that Elaine ran', 'Jerry wrote George
above', 'George George loved Kramer under', 'Kramer liked Kramer Kramer by', 'George
knows that Jerry walked George with', 'Elaine ran', 'George saw Elaine George above',
'George Elaine loved', 'Elaine read', 'Jerry Kramer loved Kramer by', 'Kramer Jerry
hated Jerry with', 'Kramer liked Elaine Kramer above', 'Jerry wrote', 'Kramer thinks
that Kramer George hated', 'Kramer saw Kramer', 'Kramer Jerry hated', 'Kramer ran',
'Kramer walked', 'George saw Elaine Jerry with', 'Kramer thinks that George walked
Jerry with', 'Jerry read Jerry with', 'Elaine wrote Elaine with', 'George saw Jerry
Jerry under', 'George read Jerry under', 'Elaine liked George', 'Kramer Kramer loved
Elaine above', 'Jerry thinks that George saw Jerry Jerry under', 'George wrote',
'Elaine says that Jerry saw Jerry Kramer under', 'George Jerry hated', 'Kramer saw
Jerry', 'Jerry saw George Jerry by', 'Kramer says that George walked', 'Elaine assumes
that Kramer wrote Kramer with', 'George wrote', 'Elaine says that Jerry saw Elaine',
'Kramer wrote Jerry under', 'George saw Jerry', 'Jerry says that Kramer read George
under', 'Kramer wrote Elaine above']
```

The initial hypothesis:

```
Initial hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP
=IP -Comp]s, [>Jerry: DP]s, [>Jerry: VP]s, [>Jerry: CP =IP]s, [>Jerry: VP =DP]s,
[>Jerry: VP =CP]s, [>Jerry: PP =DP]s, [>Jerry: VP =DP +O]s, [>Jerry: PP =DP +O]s,
[>Jerry: VP =CP +Oc]s, [>Jerry: CP =IP +Comp -Oc]s, [>George: DP]s, [>George: VP]s,
[>George: CP =IP]s, [>George: VP =DP]s, [>George: VP =CP]s, [>George: PP =DP]s,
[>George: VP =DP +O]s, [>George: PP =DP +O]s, [>George: VP =CP +Oc]s, [>George: CP =IP
+Comp -Oc]s, [>Elaine: DP]s, [>Elaine: VP]s, [>Elaine: CP =IP]s, [>Elaine: VP =DP]s,
[>Elaine: VP =CP]s, [>Elaine: PP =DP]s, [>Elaine: VP =DP +O]s, [>Elaine: PP =DP +O]s,
[>Elaine: VP =CP +Oc]s, [>Elaine: CP =IP +Comp -Oc]s, [>Kramer: DP]s, [>Kramer: VP]s,
[>Kramer: CP =IP]s, [>Kramer: VP =DP]s, [>Kramer: VP =CP]s, [>Kramer: PP =DP]s,
[>Kramer: VP =DP +O]s, [>Kramer: PP =DP +O]s, [>Kramer: VP =CP +Oc]s, [>Kramer: CP =IP
+Comp -Oc]s, [>ran: VP]s, [>ran: DP]s, [>ran: CP =IP]s, [>ran: VP =DP]s, [>ran: VP
=CP]s, [>ran: PP =DP]s, [>ran: VP =DP +O]s, [>ran: PP =DP +O]s, [>ran: VP =CP +Oc]s,
[>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [>walked: DP]s, [>walked: CP =IP]s,
[>walked: VP =DP]s, [>walked: VP =CP]s, [>walked: PP =DP]s, [>walked: VP =DP +O]s,
[>walked: PP =DP +O]s, [>walked: VP =CP +Oc]s, [>walked: CP =IP +Comp -Oc]s, [>read:
VP]s, [>read: DP]s, [>read: CP =IP]s, [>read: VP =DP]s, [>read: VP =CP]s, [>read: PP
=DP]s, [>read: VP =DP +O]s, [>read: PP =DP +O]s, [>read: VP =CP +Oc]s, [>read: CP =IP
+Comp -Oc]s, [>wrote: DP]s, [>wrote: VP]s, [>wrote: CP =IP]s, [>wrote: VP =DP]s,
[>wrote: VP =CP]s, [>wrote: PP =DP]s, [>wrote: VP =DP +O]s, [>wrote: PP =DP +O]s,
[>wrote: VP =CP +Oc]s, [>wrote: CP =IP +Comp -Oc]s, [>liked: VP]s, [>liked: DP]s,
[>liked: CP =IP]s, [>liked: VP =DP]s, [>liked: VP =CP]s, [>liked: PP =DP]s, [>liked:
VP =DP +O]s, [>liked: PP =DP +O]s, [>liked: VP =CP +Oc]s, [>liked: CP =IP +Comp -Oc]s,
[>saw: VP]s, [>saw: DP]s, [>saw: CP =IP]s, [>saw: VP =DP]s, [>saw: VP =CP]s, [>saw: PP
=DP]s, [>saw: VP =DP +O]s, [>saw: PP =DP +O]s, [>saw: VP =CP +Oc]s, [>saw: CP =IP
+Comp -Oc]s, [>loved: DP]s, [>loved: VP]s, [>loved: CP =IP]s, [>loved: VP =DP]s,
[>loved: VP =CP]s, [>loved: PP =DP]s, [>loved: VP =DP +O]s, [>loved: PP =DP +O]s,
[>loved: VP =CP +Oc]s, [>loved: CP =IP +Comp -Oc]s, [>hated: VP]s, [>hated: DP]s,
[>hated: CP =IP]s, [>hated: VP =DP]s, [>hated: VP =CP]s, [>hated: PP =DP]s, [>hated:
VP =DP +O]s, [>hated: PP =DP +O]s, [>hated: VP =CP +Oc]s, [>hated: CP =IP +Comp -Oc]s,
[>with: VP]s, [>with: DP]s, [>with: CP =IP]s, [>with: VP =DP]s, [>with: VP =CP]s,
[>with: PP =DP]s, [>with: VP =DP +O]s, [>with: PP =DP +O]s, [>with: VP =CP +Oc]s,
[>with: CP =IP +Comp -Oc]s, [>by: VP]s, [>by: DP]s, [>by: CP =IP]s, [>by: VP =DP]s,
[>by: VP =CP]s, [>by: PP =DP]s, [>by: VP =DP +O]s, [>by: PP =DP +O]s, [>by: VP =CP
+Oc]s, [>by: CP =IP +Comp -Oc]s, [>above: VP]s, [>above: DP]s, [>above: CP =IP]s,
[>above: VP =DP]s, [>above: VP =CP]s, [>above: PP =DP]s, [>above: VP =DP +O]s,
[>above: PP =DP +O]s, [>above: VP =CP +Oc]s, [>above: CP =IP +Comp -Oc]s, [>under:
VP]s, [>under: DP]s, [>under: CP =IP]s, [>under: VP =DP]s, [>under: VP =CP]s, [>under:
PP =DP]s, [>under: VP =DP +O]s, [>under: PP =DP +O]s, [>under: VP =CP +Oc]s, [>under:
CP =IP +Comp -Oc]s, [>that: VP]s, [>that: DP]s, [>that: CP =IP]s, [>that: VP =DP]s,
[>that: VP =CP]s, [>that: PP =DP]s, [>that: VP =DP +O]s, [>that: PP =DP +O]s, [>that:
VP =CP +Oc]s, [>that: CP =IP +Comp -Oc]s, [>knows: VP]s, [>knows: DP]s, [>knows: CP
=IP]s, [>knows: VP =DP]s, [>knows: VP =CP]s, [>knows: PP =DP]s, [>knows: VP =DP +O]s,
[>knows: PP =DP +O]s, [>knows: VP =CP +Oc]s, [>knows: CP =IP +Comp -Oc]s, [>says:
VP]s, [>says: DP]s, [>says: CP =IP]s, [>says: VP =DP]s, [>says: VP =CP]s, [>says: PP
=DP]s, [>says: VP =DP +O]s, [>says: PP =DP +O]s, [>says: VP =CP +Oc]s, [>says: CP =IP
+Comp -Oc]s, [>thinks: VP]s, [>thinks: DP]s, [>thinks: CP =IP]s, [>thinks: VP =DP]s,
[>thinks: VP =CP]s, [>thinks: PP =DP]s, [>thinks: VP =DP +O]s, [>thinks: PP =DP +O]s,
[>thinks: VP =CP +Oc]s, [>thinks: CP =IP +Comp -Oc]s, [>assumes: VP]s, [>assumes:
DP]s, [>assumes: CP =IP]s, [>assumes: VP =DP]s, [>assumes: VP =CP]s, [>assumes: PP
=DP]s, [>assumes: VP =DP +O]s, [>assumes: PP =DP +O]s, [>assumes: VP =CP +Oc]s,
[>assumes: CP =IP +Comp -Oc]s]
```

Initial energy:

```
Temperature: 100.000000
Energy: 10457
Grammar: 5357
Input: 5100
```

After 100 iterations:

```
Iteration: 100
2017 05 13  05 59 49: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [>Jerry: DP]s, [>Jerry: VP]s, [>Jerry: PP =DP]s, [<Jerry:
VP =DP]s, [<Jerry: PP =DP]s, [<Jerry: VP =DP +O]s, [<Jerry: VP =CP +Oc]s, [>Jerry: VP
=CP +Oc]s, [<Jerry: CP =IP +Comp -Oc]s, [>George: DP]s, [<George: VP]s, [<George: CP
=IP]s, [<George: VP =DP]s, [<George: VP =CP]s, [>George: PP =DP]s, [<George: VP =DP
+O]s, [>George: PP =DP +O]s, [<George: CP =IP +Comp -Oc]s, [>Elaine: DP]s, [<Elaine:
VP]s, [>Elaine: VP]s, [>Elaine: VP =CP]s, [>Elaine: PP =DP]s, [<Elaine: VP =DP]s,
[<Elaine: VP =CP]s, [<Elaine: VP =DP +O]s, [<Elaine: CP =IP +Comp -Oc]s, [>Kramer:
DP]s, [<Kramer: VP]s, [>Kramer: PP =DP]s, [<Kramer: CP =IP]s, [<Kramer: VP =DP]s,
[<Kramer: VP =CP]s, [>Kramer: CP =IP]s, [<Kramer: VP =DP +O]s, [>Kramer: PP =DP +O]s,
[<Kramer: VP =CP +Oc]s, [<Kramer: CP =IP +Comp -Oc]s, [<ran: VP]s, [>ran: DP]s, [>ran:
PP =DP]s, [<ran: CP =IP]s, [<ran: VP =DP]s, [<ran: VP =CP]s, [<ran: VP =DP]s, [<ran:
VP =DP +O]s, [>ran: VP =DP +O]s, [<ran: VP =CP +Oc]s, [<ran: CP =IP +Comp -Oc]s,
[<walked: VP]s, [>walked: DP]s, [>walked: PP =DP]s, [<walked: CP =IP]s, [<walked: VP
=CP]s, [>walked: VP =DP]s, [>walked: VP =DP +O]s, [>walked: PP =DP +O]s, [<walked: CP
=IP +Comp -Oc]s, [<read: VP]s, [>read: DP]s, [<read: CP =IP]s, [<read: VP =DP +O]s,
[<read: PP =DP +O]s, [<read: VP =CP +Oc]s, [<read: CP =IP +Comp -Oc]s, [>wrote: DP]s,
[<wrote: VP]s, [>wrote: PP =DP]s, [<wrote: CP =IP]s, [<wrote: VP =DP]s, [<wrote: VP
=CP]s, [<wrote: VP =DP +O]s, [>wrote: PP =DP +O]s, [<wrote: VP =CP +Oc]s, [<liked:
VP]s, [>liked: DP]s, [<liked: CP =IP]s, [<liked: VP =DP]s, [<liked: VP =CP]s, [<liked:
VP =DP +O]s, [>liked: PP =DP +O]s, [<liked: CP =IP +Comp -Oc]s, [<saw: VP]s, [>saw:
DP]s, [>saw: VP =DP]s, [<saw: VP =DP]s, [<saw: VP =CP]s, [<saw: PP =DP +O]s, [>saw: PP
=DP +O]s, [<saw: VP =CP +Oc]s, [>loved: DP]s, [<loved: VP]s, [<loved: CP =IP]s,
[<loved: VP =DP]s, [<loved: VP =CP]s, [<loved: VP =CP +Oc]s, [<loved: CP =IP +Comp -
Oc]s, [>hated: DP]s, [>hated: PP =DP]s, [<hated: CP =IP]s, [<hated: VP =CP]s, [<hated:
VP =DP +O]s, [<hated: VP =CP +Oc]s, [>with: DP]s, [<with: VP =CP]s, [>with: PP =DP
+O]s, [<with: VP =CP +Oc]s, [<with: CP =IP +Comp -Oc]s, [<with: CP =IP +Comp -Oc]s,
[>by: DP]s, [<by: CP =IP]s, [<by: VP =DP]s, [<by: VP =CP]s, [<by: VP =DP +O]s, [>by:
PP =DP +O]s, [<by: VP =CP +Oc]s, [>above: DP]s, [>above: PP =DP]s, [<above: CP =IP]s,
[<above: VP =CP]s, [>above: PP =DP +O]s, [<above: VP =CP +Oc]s, [<above: CP =IP +Comp
-Oc]s, [<under: VP]s, [>under: DP]s, [>under: PP =DP]s, [<under: CP =IP]s, [<under: VP
=DP]s, [<under: VP =CP]s, [<under: VP =DP +O]s, [>under: PP =DP +O]s, [<under: VP =CP
+Oc]s, [<that: VP]s, [>that: DP]s, [>that: PP =DP]s, [<that: CP =IP]s, [<that: VP
=DP]s, [<that: VP =CP]s, [<that: VP =DP +O]s, [>that: PP =DP +O]s, [>that: CP =IP
+Comp -Oc]s, [>knows: DP]s, [>knows: CP =IP]s, [>knows: PP =DP]s, [<knows: CP =IP]s,
[<knows: VP =DP]s, [<knows: VP =CP]s, [<knows: VP =DP +O]s, [>knows: PP =DP +O]s,
[<knows: VP =CP +Oc]s, [>knows: VP =CP +Oc]s, [<says: VP]s, [>says: DP]s, [>says: PP
=DP]s, [<says: CP =IP]s, [<says: VP =DP]s, [<says: VP =CP]s, [<says: VP =CP +Oc]s,
[>thinks: DP]s, [<thinks: CP =IP]s, [<thinks: VP =DP]s, [<thinks: VP =CP]s, [>thinks:
CP =IP]s, [<thinks: VP =DP +O]s, [<thinks: PP =DP +O]s, [>thinks: VP =DP +O]s,
[<thinks: VP =CP +Oc]s, [<thinks: CP =IP +Comp -Oc]s, [<assumes: VP]s, [>assumes:
DP]s, [>assumes: PP =DP]s, [<assumes: CP =IP]s, [<assumes: VP =DP]s, [>assumes: PP =DP
+O]s, [<assumes: VP =CP +Oc]s, [<assumes: CP =IP +Comp -Oc]s]
2017_05_13__05_59_49: Energy: 9743
2017_05_13__05_59_49: Grammar: 4470
2017_05_13__05_59_49: Input: 5273
2017_05_13__05_59_49: Temperature: 74.048426
```

After 300 iterations:

```
Iteration: 300
2017_05_13__06_50_56: Hypothesis: [[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP
=VP]s, [>@: IP =IP -Comp]s, [<Jerry: DP]s, [<Jerry: VP]s, [>Jerry: VP =DP]s, [<Jerry:
VP =DP]s, [<Jerry: VP =DP]s, [<George: DP]s, [>George: CP =IP]s, [>George: VP =DP]s,
[<George: PP =DP]s, [>George: CP =IP]s, [>George: VP =CP]s, [>George: PP =DP +O]s,
[>George: VP =CP +Oc]s, [>George: CP =IP +Comp -Oc]s, [<Elaine: DP]s, [<Elaine: VP]s,
[>Elaine: VP =DP]s, [<Elaine: VP =CP]s, [<Elaine: VP =CP +Oc]s, [<Kramer: DP]s,
[>Kramer: VP =DP]s, [<Kramer: CP =IP]s, [<Kramer: CP =IP +Comp -Oc]s, [>ran: VP]s,
[<ran: DP]s, [>ran: CP =IP]s, [<ran: VP =DP +O]s, [<ran: PP =DP +O]s, [>ran: VP =CP
+Oc]s, [>ran: VP =CP +Oc]s, [>ran: CP =IP +Comp -Oc]s, [>walked: VP]s, [<walked: CP
=IP]s, [>walked: VP =DP]s, [<walked: PP =DP +O]s, [>read: VP]s, [<read: DP]s, [<wrote:
DP]s, [>wrote: VP]s, [<wrote: CP =IP]s, [<wrote: PP =DP +O]s, [>liked: VP =DP]s,
[>liked: VP =DP +O]s, [<saw: DP]s, [>saw: VP =CP]s, [>saw: PP =DP]s, [<saw: VP =DP
```

```
+O]s, [>saw: VP =DP +O]s, [<loved: DP]s, [>loved: VP =CP +Oc]s, [<hated: DP]s,
[<hated: CP =IP]s, [<with: DP]s, [<with: VP]s, [>with: VP =CP]s, [<with: PP =DP]s,
[<by: DP]s, [<by: PP =DP]s, [>by: VP =DP +O]s, [>above: VP =CP]s, [<above: PP =DP]s,
[>under: VP]s, [<under: DP]s, [>under: VP =DP]s, [<under: CP =IP]s, [<under: PP =DP]s,
[>under: VP =DP +O]s, [>that: VP =DP]s, [<that: VP =CP]s, [<that: PP =DP]s, [>that: CP
=IP]s, [<that: PP =DP +O]s, [>knows: CP =IP]s, [>knows: VP =CP]s, [<knows: VP =CP]s,
[<knows: PP =DP]s, [<knows: PP =DP]s, [>knows: VP =DP +O]s, [>says: VP]s, [>says: VP
=CP]s, [<says: CP =IP]s, [<says: PP =DP +O]s, [>says: VP =CP +Oc]s, [>thinks: VP
=CP]s, [<thinks: VP =DP]s, [>thinks: VP =CP +Oc]s, [<assumes: DP]s, [>assumes: VP
=CP]s, [<assumes: VP =DP]s]
2017_05_13__06_50_56: Energy: 6893
2017_05_13__06_50_56: Grammar: 2232
2017_05_13__06_50_56: Input: 4661
2017_05_13__06_50_56: Temperature: 40.602006
```

After over 1000 iterations:

```
Iteration: 1187
2017_05_13__06_53_57: Hypothesis: [
[>@: DP =DP -O]s, [>@: IP =VP =DP]s, [>@: VP =PP =VP]s, [>@: IP =IP -Comp]s,
[>Jerry: DP]s, [>George: DP]s, [>Elaine: DP]s, [>Kramer: DP]s,
[>ran: VP]s, [>walked: VP]s, [>read: VP]s, [>wrote: VP]s,
[>liked: VP =DP]s, [>saw: VP =DP]s, [<loved: VP =DP]s, [<hated: VP =DP]s,
[<with: PP =DP]s, [<by: PP =DP]s, [<above: PP =DP]s, [<under: PP =DP]s,
[>that: CP =IP]s,
[>knows: VP =CP]s, [>says: VP =CP]s, [>thinks: VP =CP]s, [>assumes: VP =CP]s]
2017_05_13__06_53_57: Energy: 4014
2017_05_13__06_53_57: Grammar: 568
2017_05_13__06_53_57: Input: 3446
2017_05_13__06_53_57: Temperature: 2.825875
```

תוכן עניינים

1. סדר ראש-משלים 5
2. מודלים תיאורטיים ללמידה 9
3. מימוש לומדי סדר ראש-משלים 26
4. תוצאות צפויות 45
5. תוצאות 49
6. ניתוח תוצאות 61
7. ניסיון לשפר תוצאות באמצעות קואורדינציה 63
8. דיון 65
ביבליוגרפיה 68
נספח א' – תוצאות מפורטות 70

תקציר

סדר ראש-משלים משתנה בין ובתוך שפות. מספר תיאוריות הוצעו בכדי לתאר כיצד ייתכן שתופעה זו מיוצגת מנטלית. עם זאת, כל התיאוריות הנוכחיות מסכימות שחלק כלשהו מכך חייב להיות נלמד – או הסדר עצמו או סוג מסוים של תנועה מתאימה.

העבודה הזו מנסה לחקור את התיאוריות האלה מבעד לעדשה של למידה חישובית. לומד חישובי שמשתמש בדקדוקים מינימליסטיים (Stabler (1997) מומש והופעל בכדי ללמוד סדרי ראש-משלים שונים על פי המודלים הייצוגיים השונים.

הלומד שמוצג בעבודה זו הוא הלומד החישובי הראשון שמשמתמש בדקדוקים מינימליסטיים, וכן הלומד הראשון שמנסה ללמוד סדר ראש-משלים. העבודה הזו מהווה צעדים ראשונים לכיוון UG-comparison בתחביר ע״י שימוש בלמידה (כמו שמתואר ב-Katzir (2014)).

אוניברסיטת תל אביב

הפקולטה למדעי הרוח ע״ש לסטר וסאלי אנטין

החוג לבלשנות

למידת סדר ראש-משלים באמצעות

דקדוקים מינימליסטיים

חיבור זה הוגש כעבודת גמר לקראת התואר
״מוסמך אוניברסיטה״ באוניברסיטת ת״א

על ידי:

תומר אברהם

בהנחיית:

ד״ר רוני קציר

אוגוסט 2017