

## Justifying pooling of candidates — ReadMe file

We defend our procedure of pooling the violations of individual bigrams (about 200 million candidates) down to schematic candidates defined by their violation profiles (about 38,000 candidates).

Consider how the method would work for a very simple corpus; see spreadsheet 4b\_JustifyPoolingOfCandidates.xlsx for computations. We suppose a language with just four words, *cease*, *sock*, *kiss*, and *kick*, which means that there are 16 possible bigrams. These are assumed to occur in the corpus with the frequencies given in (1).

(1) *Bigram frequencies for a text from an imaginary four-word language*

[*cease cease*] 81, [*cease sock*] 270, [*cease kiss*] 540, [*cease kick*] 328, [*sock cease*] 328, [*sock sock*] 1088, [*sock kiss*] 1088, [*sock kick*] 660, [*kiss cease*] 164, [*kiss sock*] 544, [*kiss kiss*] 1088, [*kiss kick*] 660, [*kick cease*] 328, [*kick sock*] 1088, [*kick kiss*] 1088, [*kick kick*] 660

Suppose we are interested solely in the strength of \*SIBILANT CLASH, which is violated in bigrams like *kiss sock*. We will first do this the harder way (infeasible in real cases), using a tableau that includes all 16 word bigrams. The one true phonological constraint in the tableau will be \*SIBILANT CLASH. In addition, we include eight constraints of convenience, which are used to control for the frequency patterns of the four words in each bigram position; these constraints are WORD1 IS *CEASE*, WORD1 IS *SOCK*, and so on through WORD2 IS *KICK*. As in the main text, we permit their MaxEnt weights to take on negative values, which reflect above-average frequencies.

For clarity, we employed data frequencies constructed by using a MaxEnt grammar (see tab ConcoctTheFrequencies). This permits us to test whether, when we do our analysis, the constraint weights learned — by either the full or the pooled method — match the ones that went into the creation of the frequencies in the first place. In particular, for \*SIBILANT CLASH we deliberately employed the constraint weight 0.694, which reduces the likelihood of violation sequences by exactly one half (see main text, §8.5).

For the full method, modeling the data in (1) directly, it emerges that recovering the original weights and frequencies is no challenge for MaxEnt; both constraint weights and the frequencies emerged as an essentially perfect match, as shown in the spreadsheet tab AnalyzeInFull.

Next, we illustrate how the full model can be simplified, an operation that is needed to make computation feasible in the larger corpora we examined in the main text. The key is to isolate the critical phonological properties referenced by the constraints (expressing them again as constraints of convenience) and pool the candidates accordingly. Since the only phonological constraint in the present toy simulation is \*SIBILANT CLASH, the only crucial phonological properties are whether a Word1 begins with sibilant or not, and whether a Word2 begins with a sibilant or not. Thus we can set up two constraints of convenience, WORD1 = Xs and WORD2 = sX; these are violated by a bigram whose Word1 that ends in a sibilant or whose Word2 begins with one.

The spreadsheet tab `AnalyzeWithReducedSchema` shows the results of this reduced form of analysis. Crucially, the weight of `*SIBILANT CLASH` comes out essentially the same.

The remaining two tabs concern statistical significance testing. They show that the Likelihood Ratio Test for `*SIBILANT CLASH` comes out the same under both full and reduced modeling schemes.