

## Prediction with Partial Match (PPM) – The Exclusion Principle <sup>1</sup>

Suppose we encoded the text *prob* and the next letter to encode is *a*. Assume that, right before encountering *a*, the context of the pair *ob* and the context of the letter *b* are given in the tables below.

Context	Symbol	Count
<i>ob</i>	<i>l</i>	10
	<i>o</i>	3
	<i>ESC</i>	2
Total		15

Context	Symbol	Count
<i>b</i>	<i>l</i>	5
	<i>o</i>	3
	<i>a</i>	4
	<i>r</i>	2
	<i>e</i>	2
	<i>ESC</i>	5
Total		21

Now it's time to encode *a* and to update the context tables:

- *a* was not previously seen in the context *ob*, therefore an *ESC*ape symbol is sent and the *ESC* entry in the first table is updated.
- *a* was seen in the context *b*, therefore its corresponding entry in the second table is updated.

It can be noticed, however, that when *ESC* is sent in the context *ob*, the decoder already knows that the letter which is being encoded/decoded is neither *l* nor *o*; these letters are already known in the context *ob*, and *ESC* signals a new letter. For this reason, when encoding *a* in the context of *b* (i.e. given by the second table), a modified table can be used:

Context	Symbol	Count
<i>b</i>	<i>a</i>	4
	<i>r</i>	2
	<i>e</i>	2
	<i>ESC</i>	3
Total		11

This modified table does not contain *l* and *o*, and the count of *ESC* is also decreased by 2. The temporary exclusion of these letters offers the advantage of a more concise arithmetic encoding of the coming *a*, since *a* has now a bigger frequency.

---

<sup>1</sup>See *Sayood, Section 6.3.4*