3.16:

未來又由誰來決定接班的人呢

<CLAUSE\_WhoQ>誰</CLAUSE\_WhoQ>(.+)?(?!<CLAUSE\_WhoQ>誰</CLAUSE\_WhoQ>)(<(ACTION\_verb|VerbP)>[^<]+</(ACTION\_verb|VerbP)>)+

改成:

(?<!</FUNC\_conjunction>)<CLAUSE\_WhoQ>誰</CLAUSE\_WhoQ>((?!.誰.).)+(<(ACTION\_verb|VerbP)>[^<]+</(ACTION\_verb|VerbP)>)+

(?<!xxx)Y: Y之前不為xxx

((?!.誰.).)+: 不要出現第二個誰

**概念:**

((?!.\*some\_string.\*).)+ 中間這裡真的是放「字串」

但

[^some\_character]+ 中間只是「字符」

所以…

((?!.\*ABC.\*).)+ 就會是「什麼字符都可以，但這些字符中不能剛好組成 "ABC" 連續出現。如果是 ACC，就沒關係，如果是 AAB 那也沒關係。總之就是 ABC 不能連續出現」

[^ABC]+ 則是「什麼字符都可以，但這些字符不能是 A，也不能有 B，也不能有 C。是不是連續，不重要！」

===

所以，

<CLAUSE\_WhoQ>誰</CLAUSE\_WhoQ>((?!.\*誰.\*).)+(<(ACTION\_verb|VerbP)>[^<]+</(ACTION\_verb|VerbP)>)+

也可以寫成：

<CLAUSE\_WhoQ>誰</CLAUSE\_WhoQ>((?!.\*<CLAUSE\_WhoQ>誰</CLAUSE\_WhoQ>.\*).)+(<(ACTION\_verb|VerbP)>[^<]+</(ACTION\_verb|VerbP)>)+

[^some\_character]+ 的話，它會把

[^<CLAUSE\_WhoQ>誰</CLAUSE\_WhoQ>]+

這裡面的任何一個「符號」都視為要排除的符號。

這麼一來，就連其它 POS 標記裡的 <, >, \_ 也都一起被排除了。

這會導致你的 regex pattern "overkill" (排除過頭了，不該排除的也被排掉了)

