RiCL Research in Corpus Linguistics

Hunting for Post-Auxiliary Ellipsis in a parsed corpus of English

Evelyn Gandón-Chapela[1]
Universidad de Cantabria / Spain

Abstract – Despite the existence of a great number of studies that have analysed ellipsis from a theoretical point of view, only recently has it been studied empirically using corpora (Hardt 1997; Hardt and Rambow 2001; Nielsen 2005; Bos and Spenader 2011). These corpus studies have tried to discover new methods and algorithms for the automatic detection and retrieval of ellipsis in Present-Day English. In this paper, I extend these studies by presenting an automatic retrieval algorithm for cases of Post-Auxiliary Ellipsis in Late Modern English (1700–1914), using data from the Penn Parsed Corpus of Modern British English.

Keywords – ellipsis, corpus, parsing, retrieval algorithm, Late Modern English

1. Introduction

In this paper I will first introduce the concept of ellipsis, to then describe the characteristics of Post-Auxiliary Ellipsis (section 2), which will be the focus of this study. In section 3, I will offer a general description of the Penn Parsed Corpora of Historical English, whose texts have been the data source of this paper. In section 4, I will present the programme CorpusSearch 2 and its query language. In section 5, I will explain the methodology used, the retrieval algorithm and its precision and recall. Section 6 offers a summary of the results obtained by the algorithm presented in this work.

2. Post-Auxiliary Ellipsis

The term 'ellipsis' refers to cases in which expected, i.e. subcategorised syntactic elements, have gone missing. Elliptical constructions, therefore, illustrate a mismatch between meaning (the intended message) and sound (what is actually uttered). For example, in (1), although part of the sentence has been left unpronounced, its meaning can still be understood and retrieved from the surrounding linguistic context:

(1) Michael keeps on telling me I won't pass the exam, but I believe I will ~~pass the exam~~.[2]

This paper focuses on the study of instances of Post-Auxiliary Ellipsis (PAE) (Sag 1976; Miller and Pullum 2014), a term which covers those cases in which a Verb Phrase (VP), Determiner Phrase (DP), Adjective Phrase (AP),

[1] I am grateful to the following institutions for generous financial support: the Spanish Ministry of Economy and Competitiveness and the European Regional Development Fund (grant no. FFI2013-44065-P), and the Autonomous Government of Galicia (grant no. GPC2014/060).

[2] Strikethrough words represent elided material, but they are not intended to represent a syntactic or semantic analysis of ellipsis.

Prepositional Phrase (PP) or Adverbial Phrase (AdP) is elided after one of the following licensors (i.e. those elements that license the occurrence of ellipsis): modal auxiliaries, non-modal auxiliaries *be*, *have* and *do*, and infinitival marker *to* (the latter taken to be a defective non-finite auxiliary verb, see Miller and Pullum 2014). Here follow some examples (licensors appear in bold):

(2) A: Can you pass me the salt?
B: Yes, of course I **can** [~~pass you the salt~~]$_{VP}$.
(3) Smith hadn't seen him in five years and didn't really want **to** [~~see him~~]$_{VP}$.
(4) John is a doctor and Anne **is** [~~a doctor~~]$_{DP}$ too.
(5) If you don't tell me, you **will** ~~tell~~ your mum.

3. The Penn Parsed Corpora of Historical English

The Penn Parsed Corpora of Historical English is a collection of texts of British English prose from different historical periods (from Middle English to the First World War),[3] which are available in three different forms: raw text, part-of-speech tagged text and syntactically annotated (parsed) text. Crucially for this study, based on the Penn Parsed Corpus of Modern British English (PPCMBE) (Kroch et al. 2010), the syntactic annotation of this corpus offers the possibility of searching for words or word sequences, as well as for syntactic structures.

4. CorpusSearch 2

CorpusSearch 2 is a Java programme that offers the possibility of searching corpora as well as building an annotated corpus. CorpusSearch allows implementing the following activities automatically (CorpusSearch Home, 2005):

- find and count lexical and syntactic configurations of any complexity;
- correct systematic errors;
- code the linguistic features of corpus sentences for later statistical analysis.

The basic query language includes AND, OR and NOT, which are used as in basic formal logic. The same applies to the use of parentheses. Besides, there are other specific functions and operators such as:

- *HasSister* looks for strings of elements that have the same mother, i.e. the element searched for can either precede or follow another element.
- *Doms* or *iDoms*: *X* Doms (or IDoms) *y* will look for an element *y* that is contained within any subtree dominated by *x*. IDoms will look for *y* contained in the same tree or subtree as *x*.
- *Precedes* or *IPrecedes* searches for a string that either precedes *x* (Precedes) or that immediately precedes *x* (IPrecedes).
- *X hasLabel y* if the label of node *x* is the string *y*.
- The pipe operator | stands for "or" at the level of arguments to a search function.
- Wild card * stands for any string of symbols. Therefore, *IP** will retrieve all the different types of IP present in the corpus (e.g. IP-MAT, IP-IMP, etc). This character may also be used to indicate elided VPs, annotated as *(VB *)*.

5. The retrieval algorithm

I have manually analysed 12 raw texts (112,347 words out of almost one million words), all belonging to different genres and periods of the PPCMBE. After that, I examined the syntactic patterns followed by the examples of PAE to draw some generalisations that would allow the creation of an algorithm for their automatic retrieval. Importantly, this manual analysis revealed that PAE is not annotated uniformly in the PPCMBE. Auxiliaries *do* and *have* had already been tagged as licensors of PAE in the vast majority of cases. Simply searching for *VB*s that immediately dominate (iDoms) the * symbol returns a fair amount of PAE instances. However, this was not the case with examples of PAE whose licensor is auxiliary *be*, as they were not coded:

---

[3] For more information on the corpora, visit http://www.ling.upenn.edu/hist-corpora.

(6) […] and therefore it cannot be learn'd by Conversation, as **the Modern Languages are**; (ANON-1711,12.125)
(PP (P as)
(CP-ADV (WADVP-2 0)
(C 0)
(IP-SUB (ADVP *T*-2)
(NP-SBJ (D the) (ADJ Modern) (NS Languages))
(BEP are))))))))))
(, ;)
(ID ANON-1711,12.125)

These examples had to be found using another strategy: I looked for contexts where auxiliary *be* (with either positive or negative polarity) would be immediately followed by any kind of punctuation mark, which is a typical context for PAE to occur.

Modal auxiliaries also followed a pattern: I searched for examples of modal auxiliaries that were not followed by any kind of verb, with the exception of auxiliaries *have* and *be*, which could be present in some examples of PAE, like the following:

(7) Cha. Sir George is ready to depart. But I **could not have** ~~departed~~ without returning to say farewell! (COLLIER-1835,26.952)
(8) That they sometimes give one to understand that their experiences and travels are not appreciated as they **should be** ~~appreciated~~ by their friends of the chapel. (BRADLEY-1905,204.104)

The following was the initial script used as a query file to obtain the examples of PAE:

(9)
```
node: *
query: (VB* iDoms \*)
OR (HV* iDoms \*)
OR (MD* hasSister !VB*|BE*|DO*|HV*)
OR ((MD* iPrecedes HV*)
AND (HV* iPrecedes [.,]))
OR ((MD* iPrecedes NEG)
AND (NEG iPrecedes HV*)
AND (HV* iPrecedes [.,]))
OR ((MD* iPrecedes HV*)
AND (HV* iPrecedes BE*)
AND (BE* iPrecedes [.,]))
OR ((MD* iPrecedes NEG)
AND (NEG iPrecedes HV*)
AND (HV* iPrecedes BE*)
AND (BE* iPrecedes [.,]))
OR (BE* iPrecedes [.,])
OR ((BE* iPrecedes NEG)
AND (NEG iPrecedes [.,]))
OR (HV* iPrecedes [.,])
OR ((HV* iPrecedes NEG)
AND (NEG iPrecedes [.,]))
OR ((HV* iPrecedes NP-SBJ)
AND (NP-SBJ iPrecedes [?]))
OR ((DO* iPrecedes NEG)
AND (NEG iPrecedes NP-SBJ)
AND (NP-SBJ iPrecedes [.,?]))
OR (DOI iPrecedes [.,])
OR (CP* hasLabel CP-QUE-TAG*)
```

The algorithm will be described step by step in what follows. Firstly, the search domain, i.e. the node, was defined. The software would look for examples of PAE in every possible node by making use of the wild card *. The first condition of the algorithm, namely "VB* iDoms *", looks for any kind of verb (i.e. *VB**) that immediately dominates a *, i.e. a case of ellipsis, and retrieves the vast majority of cases of PAE. The second condition, i.e. "HV* iDoms *" searches for examples where the auxiliary *have* immediately dominates an asterisk, whereas the third one, "(MD* hasSister !VB*|BE*|DO*|HV*)", looks for instances of PAE licensed by a modal auxiliary, with no verbal material or auxiliaries *be, have* and *do* as its sister. This would be one output example:

(10) But still you would reckon the injuring person more unhappy than he who had suffered the wrong? -I certainly **would**. (BOETHRI-1785,160.349)

The following two conditions "((MD* iPrecedes HV*) AND (HV* iPrecedes [.,]))" and "((MD* iPrecedes HV*) AND (HV* iPrecedes BE*) AND (BE* iPrecedes [.,]))"—together with their negative counterparts: "((MD* iPrecedes NEG) AND (NEG iPrecedes HV*) AND (HV* iPrecedes [.,]))" and "((MD* iPrecedes NEG) AND (NEG iPrecedes HV*) AND (HV* iPrecedes BE*) AND (BE* iPrecedes [.,]))"—search for the other two possible combinations with modal auxiliaries that may license the occurrence of PAE. The combination "((MD* iPrecedes HV*) AND (HV* iPrecedes [.,]))" retrieves examples of PAE where a modal auxiliary immediately precedes auxiliary *have* and this auxiliary also immediately precedes any kind of punctuation mark, as in (11):

(11) but in a little time I hope to do all you **would have**. (JOHNSON-1775,2,9.177)

The second possible combination with modal auxiliaries that may license PAE, "((MD* iPrecedes HV*) AND (HV* iPrecedes BE*) AND (BE* iPrecedes [.,]))", looks for combinations of a modal immediately preceding the sequence *have been* right before a punctuation mark, as in (12):

(12) That the wicked, who suffer the chastisement which they merit, are happier than they **would have been**, if justice had allowed their crimes to have escaped unpunished. (BOETHRI-1785,154.295)

The negative counterparts of the two previous conditions, "((MD* iPrecedes NEG) AND (NEG iPrecedes HV*) AND (HV* iPrecedes [.,]))" and "((MD* iPrecedes NEG) AND (NEG iPrecedes HV*) AND (HV* iPrecedes BE*) AND (BE* iPrecedes [.,]))", however, did not yield any examples of PAE.

The condition "(BE* iPrecedes [.,]) OR ((BE* iPrecedes NEG) AND (NEG iPrecedes [.,]))", in turn, looks for instances of PAE licensed by auxiliary *be* (in either positive or negative form) immediately preceding any punctuation mark.

The next condition, i.e. "(HV* iPrecedes [.,]) OR ((HV* iPrecedes NEG) AND (NEG iPrecedes [.,]))", was included to retrieve examples of PAE licensed by auxiliary *have* (with either positive or negative polarity) immediately preceding any punctuation mark. This is another syntactic context of PAE that was not coded consistently in the corpus. Most occurrences of PAE licensed by auxiliary *have* where what has undergone ellipsis is verbal material were coded, though there are some exceptions. However, those examples where auxiliary *have* licenses the ellipsis of non-verbal material, as in (13), were never annotated in the corpus.

(13) Bare. Indeed, Mrs. Liddy, I have a very great Opinion of you; and to let you see I **have**, will entrust you with a Secret, in which I must beg your Assistance. (DAVYS-1716,32.269)

The next condition, "((HV* iPrecedes NP-SBJ) AND (NP-SBJ iPrecedes [?]))", constituted an ad-hoc solution to retrieve examples like (14), which had been identified in the manual analysis of the corpus but could not be found automatically. This condition searches for examples where auxiliary *have* immediately precedes an NP subject which, in turn, immediately precedes a question mark.

(14) I assure you, I had much rather you would go. Sir Simon. **Had you?** (COLMAN-1805,46.884)

There were also cases where auxiliary *do* in imperative clauses was not coded as a licensor of PAE. Therefore, two more ad-hoc solutions were adopted. Firstly, "((DO* iPrecedes NEG) AND (NEG iPrecedes NP-SBJ) AND (NP-SBJ iPrecedes [.,?]))" looks for examples where auxiliary *do* immediately precedes a negator. This negator, in turn, immediately precedes an NP subject which is immediately followed by any punctuation mark, as in (15). Secondly, the condition "(DOI iPrecedes [.,])" retrieves examples where auxiliary *do* appears in the imperative and immediately follows any punctuation mark, as in (16).

(15) Bur. Don't take on so –**don't you**, now! (COLMAN-1805,33.483)
(16) Make haste after me, **do**, now! (COLMAN-1805,36.601)

Finally, the query "(CP* hasLabel CP-QUE-TAG*)" was used to look for cases of CPs whose label was "CP-QUE-TAG", i.e. cases of question tags where PAE necessarily occurs, as in (17):

(17) We have nothing to consult about, **have we, William?** (BROUGHAM-1861,10.345)

The examples which had been found manually were later compared with an automatic search of the parsed files. The evaluation of the performance of the algorithm was calculated taking into account its recall (1) and precision (2), as well as the F1-measure (3). These are defined as follows:

$$\text{Recall} = \frac{\text{No(correct ellipses found)}}{\text{No(all ellipses in test=all ellipses found manually)}} \quad (1)$$

$$\text{Precision} = \frac{\text{No(correct ellipses found)}}{\text{No(of answers given by the algorithm)}} \quad (2)$$

$$\text{F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (3)$$

Recall is calculated by dividing the number of relevant examples retrieved automatically by the gold standard (the number of examples found manually), providing the measure of the coverage of the algorithm. Precision, in turn, is calculated by dividing the number of relevant examples of PAE found by the number of attempts, providing a measure of the accuracy of the algorithm. Finally, the F1-measure combines recall and precision at a 1/1 ratio, giving "the harmonic mean of these two" (Nielsen 2005: 60). The recall of this initial algorithm was 0.89 (140/156=0.89), whereas its precision was 0.55 (140/251=0.55) and the F1 = 0.67. However, since some examples of ellipsis like *Since the Law was given by Moses, Grace and Truth was ~~given~~ by Jesus Christ* could not be retrieved by this algorithm, a new condition was included that would also retrieve cases of *be* (with either positive or negative polarity) followed by PPs and AdPs:

OR (BE* iPrecedes PP|ADVP)
OR ((BE* iPrecedes NEG)
AND (NEG iPrecedes PP|ADVP))

Successfully, the improved algorithm achieved a recall of 0.97 (152/156= 0.97), a precision of 0.23 (152/640= 0.23), and an F1= 0.37. It should be noted that precision was also lowered due to some wrong analyses in the parsed files. Here is one example that was wrongly parsed as PAE:

(18) I did not mean to reproach you, nor meant anything but respect and impatience **to know how you did**. (JOHNSON-1775,2,27.511)

Finally, there were cases of interruptions of the speech which the programme interpreted as cases of ellipsis after a modal auxiliary:

(19) No, **I'll**- Yes, I'll read, first, and walk, afterwards. (COLMAN-1805,47.905)

6. Conclusions

A new algorithm for the automatic retrieval of cases of PAE in Late Modern English has been presented. The final algorithm has achieved a recall of 0.97, a precision of 0.23, and an F1= 0.37. Recall was favoured over precision, as the aim of this algorithm was to find as many examples of PAE as possible, at the cost of a low precision –which was also lowered due to some wrong analyses in the parsed files.

References

Bos, Johan and Jennifer Spenader. 2011. An annotated corpus for the analysis of VP ellipsis. *Language Resources and Evaluation* 45/4: 463–494.

CorpusSearch <http://corpussearch.sourceforge.net>.

Hardt, Daniel. 1997. An empirical approach to VP ellipsis. *Computational Linguistics* 23/4: 525–541.

Hardt, Daniel and Owen Rambow. 2001. Generation of VP ellipsis: a corpus-based approach. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*. Toulouse: CNRS Institut de Recherche en Informatique de Toulouse and Université des Sciences Sociales, 290–297.

Kroch, Anthony, Beatrice Santorini and Ariel Diertani. 2010. Penn Parsed Corpus of Modern British English. <http://www.ling.upenn.edu/hist-corpora/PPCMBE-RELEASE-1/index.html>.

Miller, Philip and Geoffrey K. Pullum. 2014. Exophoric VP ellipsis. In Philip Hofmeister and Elisabeth Norcliffe eds. *The core and the periphery: data-driven perspectives on syntax inspired by Ivan A. Sag*. Stanford, CA: CSLI, 5–32.

Nielsen, Lief Arda. 2005. A corpus-based study of verb phrase ellipsis identification and resolution. PhD. King's College London.

Sag, Ivan. 1976. Deletion and logical form. PhD. MIT.

Corresponding author
Av. de los Castros s/n. Edificio interfacultativo
Universidad de Cantabria
39005 Santander
e-mail: evelyn.gandon@unican.es

received: March 2016
accepted: November 2016