From YCOE to UD: rule-based root identification in Old English

Luca Brigada Villa, Martina Giarda
University of Pavia/Bergamo, University of Pavia/Bergamo
{luca.brigadavilla, martina.giarda}@unibg.it

Abstract

In this paper we apply a set of rules to identify the root of a dependency tree, following the Universal Dependencies formalism and starting from the constituency annotation of the York-Toronto-Helsinki Parsed Corpus of Old English Prose (YCOE). This rule-based root-identification task represents the first step towards a rule-based automatic conversion of this valuable resource into the UD format. After presenting Old English and the annotated resources available for this language, we describe the different rules we applied and then we discuss the results and the errors.

Keywords: Old English, root-identification, YCOE, Universal Dependencies

1. Introduction

The York-Toronto-Helsinki Parsed Corpus of Old English Prose (henceforth YCOE) (Taylor et al., 2003) is the reference treebank for studies on Old English syntax. It is a 1.5-million-word constituency treebank, annotated following the Penn format. As a sister corpus to the Penn-Helsinki Parsed Corpus of Middle English (PPCME2) (Kroch and Taylor, 2000), it uses the same form of annotation and is accessed by the same search engine, *CorpusSearch2*, whose usage is not always intuitive. Moreover, dependency annotation schemes have gained widespread acceptance, making the Universal Dependencies (UD) format, as described in de Marneffe et al. (2021), the standard for dependency treebanks. In the latest released version (May 15, 2023), more than 245 treebanks in 141 languages (both modern and ancient) were annotated according to UD standards. However, no treebank for Old English is available in dependency format, in contrast to the large amount of annotated resources for Present-Day English.[1]

These considerations led us to attempt the creation of a dependency treebank of Old English, following the UD format. Training a monolingual parser would require a large sample of manually annotated data, which can be really time-consuming to produce. After attempting to train a multilingual parser (Brigada Villa and Giarda, 2023), we aimed to produce a rule-based conversion of the YCOE, so that the massive work of the creators of this treebank would not have been lost. The starting point of this conversion is a rule-based root identification task, since the root is the node from which every other depends. Using the original Part-of-Speech (POS) tags in the YCOE, we created hierarchical rules to identify the root of the sentences. Afterwards, we checked the efficacy of these rules against a manually annotated gold set.

The paper is structured as follows: in Section 2 we introduce Old English providing a brief description of its history, developments, and morpho-syntactic features. Moreover, we provide a brief overlook of the main available resources for this language and a description of the YCOE structure. In Section 3 we present our data and methodology, whereas Section 4 is dedicated to the results and Section 5 to error discussion. Finally, Section 6 concludes the paper and summarizes our findings.

2. Old English

Old English is a West-Germanic language, classified with Old Frisian and Old Saxon among the so-called Ingvaeonic languages. It was the language spoken in England after Angles, Saxons, Jutes and Frisians came to Britain and settled in the island in the 5th century. It is attested from the 7th century, except for some older brief runic inscriptions, whereas its ending point is conventionally established in 1066, date of the Norman Conquest of England (von Mengden, 2017a). Old English is a fusional language with inflectional word classes. As other Germanic languages, it has two main conjugational system, called, respectively, strong and weak verbs. Strong verbs build the preterit by means of apophony, i.e. vowel alternation, also found in Present-Day English (PDE) irregular verbs, whereas weak verbs insert a dental suffix, just as PDE regular verb, whose past form is constructed with the *-ed* suffix. Finite Old English verbs inflect for mood (indicative, subjunctive, imperative), tense (present and past), number, and person. All the plural forms in all moods and tenses, and the first and third person singular in the subjunctive show syncretism (von Mengden, 2017b). Concerning word order, it is not as rigid as in PDE, despite the fact that some regularities can be found (Mitchell and Robinson, 2012: 63-65). It is still debated whether the basic word order was (S)VO or (S)OV.

[1] UD has 10 different treebanks for Present-Day English.

LT4HALA 2024@LREC-COLING-2024, pages 22–29

(Molencki, 2017: 101): it is generally assumed the early stages of the Old English language were characterised by a competition between (S)OV and (S)VO word orders, in which the former prevailed over the latter as the basic order. (Fischer et al., 2001: 51; Pintzuk and Taylor, 2006). Like other ancient and modern Germanic languages, OE also exhibits V2, i.e. the tendency of the finite verb to follow the first constituent, regardless of its type. Nouns are inflected by number and case, following three inflectional classes, depending on their original Proto-Germanic stem. Old English retains four of the eight original Indo-European cases: nominative, accusative, genitive, and dative. Moreover, residual traces of the instrumental are found. Depending on the class, different cases can show syncretism. Concerning the order of other constituents in the NP, nouns are generally preceded by modifiers, e.g. demonstratives, adjectives, genitive complements. However the latter can follow the noun if another preceding modifier is present. In PPs, adpositions tend to precede a noun, but generally follow a pronoun; however, the opposite is also attested (Molencki, 2017).

Old English allows subjectless constructions, above all with reference to natural phenomena. However, it has also developed the use of empty pronominal subjects (*hit* 'it' and *þær* 'there'), which were neither anaphoric or cataphoric (Molencki, 2017: 104). Old English exhibits some complex (or periphrastic) verbal constructions, whose origin and grammaticalization are still debated among scholars. Both present and past perfect were made of the auxiliary *habban* 'have' (for transitives) or *beon/wesan* (for intransitives) and the past participle of the main verb, this latter either inflected or not (Molencki, 2017: 112-113). The passive voice was also expressed by a periphrastic construction, made of the auxiliary *beon/wesan* 'be' or *weorþan* 'become' and the past participle, with the sole exception of the verb *hatan* 'be called' (but also 'order'). A part for asyndetic clauses, Old English texts are richer in paratactic devices (very often repetitive) than in subordination. However, the borderline between parataxis and hypotaxis is rather vague, above all in temporal clauses, in which the sequence of events is often expressed by means of clause-initial *þa* 'then'. (Molencki, 2017: 117).

2.1. Annotated resources for Old English

Differently from other ancient languages, such as Latin or Ancient Greek,[2] and its contemporary counterpart, scholars have devoted little attention to the creation of resources to study Old English. At the moment, the sole syntactically annotated resources for this language are the constituency treebank YCOE and its poetry counterpart, the York-Helsinki Parsed Corpus of Old English Poetry (YCOEP) (Pintzuk and Plug, 2002), which follow the Penn style. Despite their value in size, these treebanks are hardly machine- nor user-friendly, have no interface and can only be investigated through their search engine *CorpusSearch2*, which requires an intensive training in order to write even simple queries.

A first attempt to build a UD treebank for Old English has been made by Arista (2022a) and Arista (2022b), but the treebank has not been published yet. Also, Brigada Villa and Giarda (2023), trained multilingual parser on data from Old English, Modern German, Modern Icelandic and Modern Swedish data to parse Old English. However, no attempts at a rule-based conversion of the whole YCOE have been made yet. There exists a pipeline to convert Penn-format constituency treebanks into UD dependency treebanks (Arnardóttir et al., 2020): however this is designed for the Icelandic Parsed Historical Corpus (IcePaHC; Rögnvaldsson et al., 2012) and the Faroese Parsed Historical Corpus (FarPaHC; Ingason et al., 2014), which, though based on the Penn Parsed Corpora of Historical English (PPCHE, also base for the YCOE; Kroch and Taylor, 2000), present some crucial differences in the annotation scheme, some of which would require a more thoroughful revision.

2.1.1. YCOE description

The YCOE is a 1.5 million word syntactically-annotated corpus. Its size and representativeness makes it a valuable resource for the study of Old English. However, the constituency format and the lack of some information (e.g. lemmatization, and some morphological features) may hinder data retrieval. A conversion of this treebank into the Universal Dependencies format would solve some of the problems, while preserving the huge amount of data already available. The format in which the sentences in the YCOE treebank are parsed consists of a limited hierarchical bracketing comprising labeled parentheses to represent syntactic trees. Word forms serve as the fundamental units of the sentence: they are POS tagged and then grouped together to construct more complex structures such as phrases and sentences. Each element within the sentence is labeled, enabling the retrieval of the tree structure from the annotation.[3]

An example of annotation can be found in Figure 1. In this sentence, we can notice that the words are

[2]The latest release of UD (v2.11) includes 5 treebanks for Latin and 2 for Ancient Greek.

[3]all POS tags are retrievable here: https://www-users.york.ac.uk/~lang22/YCOE/doc/annotation/YcoeLite.htm#pos_labels, whereas the syntactic tags can be found here: https://www-users.york.ac.uk/~lang22/YCOE/doc/annotation/YcoeLite.htm#syntactic_labels.

```
( (IP-MAT (NP-NOM (PRO^N hi))
          (VBDI oncneowon)
          (ADVP-TMP (ADV^T +da))
          (CP-THT (C +d+at)
                  (IP-SUB (NP-NOM (PRO^N hi))
                          (ADJP-NOM-PRD (ADJ^N nacode))
                          (BEDI w+aron)))
          (. ,))
  (ID cootest,Gen:3.7.132))
```

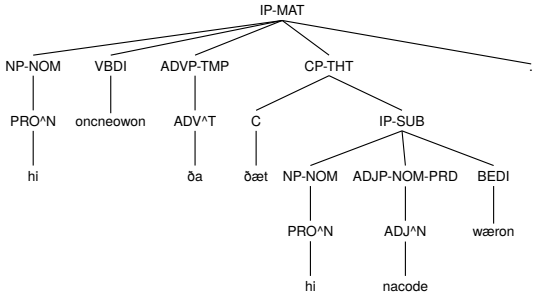

Figure 1: YCOE annotation style of the sentence `cootest,Gen:3.7.132`, whose translation is 'Then they realized that they were naked'

the innermost elements in the hierarchical structure (*hi*, *oncneowon*, *+da*, *+d+at*, *hi*, *nacode* and *w+aron*) and phrases can contain either words or smaller phrases (`NP-NOM`, `ADVP-TMP`, `CP-THT`, `IP-SUB`, `NP-NOM`, `ADJP-NOM-PRD`). Wrapping all the words and phrases of the sentence, there is a label denoting a clause (in this case `IP-MAT`).

3. Data and Methodology

Our data[4] consists of 390 manually annotated sentences, from three different texts: *Adrian and Ritheus*, the first homily of Ælfric's *Supplemental Homilies*, and the first 100 sentences of Book 1 of Bede's *Historia Ecclesiastica Gentis Anglorum*, translated in Old English.[5] The choice to include also Bede is due to the fact that Latin has had a great influence on Old English syntax, pushing it toward a more frequent use of hypotaxis. Since a significant part of the Old English corpus is made of translations, we wanted to test our rule-base conversion both on translations and texts originally written in Old English, without a Latin source.
The set of sentences selected to conduct this study was manually annotated following the Universal Dependencies guidelines. This set formed our gold standard and was used to compare the annotations performed by our model.
The YCOE has the tendency to split coordinated clauses into different sentences, with different sentence IDs. According to context, some coordinated sentences have been connected to their main clause. Although punctuation is not always reliable, since it is added by the modern editor, the general rule was to connect clauses divided by commas, but to leave separated those divided by a period, even if the following sentence started with the conjunctions *and* 'and' or *ac* 'but'. The sentences divided by a semicolon have been treated differently depending on the context. In all cases, the sentence ID of the main clause was retained.
In this section, we will discuss the two main steps of our process: (1) the conversion from the format in which the YCOE treebank appeared into the CoNLL-U format (Buchholz and Marsi, 2006) and (2) the implementation of the rules to identify the sentence roots.

3.1. Conversion into CoNLL-U

As discussed in Section 2.1.1, words are the basic unit of annotation in the YCOE treebank. They appear between brackets that only contain the part-of-speech tag and the form in which they appear in the sentence. Given these premises, the identification of the tokens to include in the CoNLL-U converted file is almost straightforward. However, it's worth noting that information such as document and sentence identifiers also appears in the same format as words. For this reason, we had to filter the extracted tokens. To do so, we listed all the possible part-of-speech tags assignable to tokens and we included as tokens only the elements which had one of the tags in the list. We used the list as a table of conversion of the POS tags in the YCOE to a combination of Universal part-of-speech tags and features. In addition to that, we converted the characters such as *thorn*, *eth*, and *ash*, which appeared in their Helsinki equivalents (+t, +d, and +a, and the respective capital letters), to Unicode characters (þ, ð, æ, and the respective capital letters). Doing so, we obtained a CoNLL-U file in which this information was automatically retrieved from the YCOE treebank:

- the sentence id
- the text of the sentence
- for each token:
  - the word form
  - the universal part-of-speech tag
  - the features[6]

[4] The code and data used for this work can be found at https://github.com/unipv-larl/wundorsmitha-geweorc/tree/main/paper_projects/root-identification-oe

[5] *Adrian and Ritheus* is a dialogue on several biblical issues (Cross and Hill, 1982: 3-4). On the other hand, Ælfric's homily, *Nativitas Domini*, is a Christmas homily, with several expansions, consisting in scriptural elaborations (Pope, 1968: 191-195).

[6] The table used to convert YCOE tags to UD parts-of-speech tags and UD features can be consulted here: https://github.com/unipv-larl/wundorsmitha-geweorc/blob/main/paper_projects/root-identification-oe/pos_table.tsv.

3.2. Rules to identify the roots

The main goal of this work was to define a set of rules that allow to automatically identify the root of the dependency tree, given the annotation of a constituency tree of the OE sentence. In this section, we describe the rules that we implemented.

To define the rules, we benefit from the annotation of the YCOE treebank, which, despite not following a dependency formalism, still gave us some useful information about the syntactic structure of the sentences. In UD, a sentence's root must be unique, and this role can be attributed to tokens with a limited set of features. For example, most of the times adpositions and conjunctions cannot serve as the root of a sentence, but nouns and verbs are eligible for this role. Therefore, having part-of-speech annotation was particularly beneficial in identifying a pool of candidates from which to select the root.

The first step to select the set of candidates, before looking at the parts-of-speech, consist in restricting the number of eligible tokens to those that occupy a relevant position in the constituency tree. The format of each sentence involves a top-level clause that includes isolated tokens and phrases. We focused on the set of isolated tokens (not including punctuation) and we applied some rules taking this set as starting point.

As a matter of example, considering the sentence in Figure 1, we can see that the top-level clause is tagged with the `IP-MAT` label and involves a noun phrase (`NP-NOM`), an adverbial temporal phrase (`ADVP-TMP`), a *that-clause* (`CP-THT`) and an isolated token (`oncneowon`).

The general approach of the procedure to identify the root is exemplified in the algorithm in Figure 2.

In the following sections, we will discuss more in detail each one of the rules mentioned in the algorithm. We will start from the rules that can be applied when the set of isolated tokens is not empty (`VB`, `BE_INF`, `BE_COPULA`, `HAVE`, `BE_ROOT`, `MD`) and then we will move to the other rules (`IP-MAT-0`, `CP_QUE`, `COORD_VB`).

3.2.1. Rule VB

This rule requires a set of isolated tokens to be applied. It considers as good candidates to represent the root of the sentence the isolated tokens whose tag that starts with `VB`, denoting verbs other than the verb 'to be', the verb 'to have' and modal verbs. This rule succeeds in finding the root only if the set of candidates includes one and only one token matching the condition described. It is worth noticing that the verbs might also be tagged with a label preceding `VB`, such as `RP` and `NEG`, denoting the fact that to such verb an adverbial or negative particle is added. So, the `VB` rule assigns the label `root` to the verb.

Require: isolated tokens
1: **if not** isolated tokens **then**
2: apply `IP-MAT-0` rule
3: **if not** root found **then**
4: apply CP-QUE rule
5: **end if**
6: **end if**
7: **for all** rule in (`VB`, `BE_INF`, `BE_COPULA`, `HAVE`, `BE_ROOT`, `MD`) **do**
8: apply rule on isolated tokens // *The rules are applied in the order in which they appear in the list*
9: **if** root found **then**
10: break
11: **end if**
12: **end for**
13: **if not** root found **then**
14: apply `CP_QUE`
15: **end if**
16: **if not** root found **then**
17: apply `COORD_VB`
18: **end if**

Figure 2: Procedure to identify the root.

Require: isolated tokens whose tag starts with `BE`

1: **if** isolated tokens contains one element **then**
2: look for the token following the verb
3: **if** tag following token starts with `TO` **then**
4: assign `root` to the verb 'to be'
5: **end if**
6: **end if**

Figure 3: `BE_INF` rule.

3.2.2. Rule BE_INF

This rule aims to identify all the instances of the verb 'to be'[7] that are parent of an infinitive phrase. To do so, we first look for the isolated tokens which have the tag starting with `BE`; then, if this set consists of only one element, we extract its subsequent element: if its tag starts with `TO`, then we can assign the `root` label to the verb 'to be', as exemplified in Figure 3.

3.2.3. Rule BE_COPULA

This rule aims to find the root in all the situations in which the verb 'to be' acts as a copula of a nominal predicate. According to the UD guidelines, in sentences like these, the root should be assigned to the noun (or adjective) that is the head of the noun (or adjectival) phrase having the role of nominal

[7] Note that the tags starting with `BE` indicate both forms of the two verbs meaning 'to be' (*beon* and *wesan*), but also the forms of the verb *weorþan* 'to become', since it is used as auxiliary to form the passive, or in copular constructions.

Require: isolated tokens whose tag starts with BE

1: **if** isolated tokens contains one element **then**
2: look for the isolated phrases whose tag ends with NOM-PRD
3: **if** the set of NOM-PRD consists of one element **then**
4: assign root to the head of the NOM-PRD phrase
5: **end if**
6: **end if**

Figure 4: BE_COPULA rule.

predicate.
We followed the steps as described in Figure 4. We started, as for the BE_INF rule (described in Section 3.2.2), looking for the isolated tokens which have the tag starting with BE; then we looked for the phrases, at the same hierarchical level of the verb 'to be', whose tag ended with NOM-PRD. These combination of labels is used in the YCOE treebank to tag all the predicates in the nominative case. After finding the phrase and checking for its uniqueness in the sentence, we assigned the root label to the head of the noun or adjectival phrases in the predicate.

3.2.4. Rules HAVE, BE_ROOT and MD

The remaining rules can be described as the VB rule in Section 3.2.1. The reason why we differentiate these three rules from the others is that we need to check other conditions before assigning the root label to the verbs 'to have', 'to be', and modal verbs. These three classes of verbs can function as the roots of a sentence, but this happens only under specific conditions (e.g., when nominal predicates, passive verbs, or other finite verbs are not present in the sentence).
These rules are applied at the end, after all the other rules have failed, and they assign the root label to the isolated verbs 'to have', 'to be', and modal verbs, respectively.

3.2.5. Rule IP-MAT-0 and CP_QUE

These rules aims to find the root when the set of isolated tokens is empty. When this happens, we first look for the presence of a phrase whose tag is IP-MAT-0. The -0 tag is used in the YCOE treebank to label all the incomplete IPs (e.g. IPs arisen from elision). Then, after finding the target phrase, we performed the operations described from line 7 to line 12 of the algorithm in Figure 2.
In case of unsuccessful application of the IP-MAT-0 rule, we looked for a phrase whose tag starts with CP-QUE. The type of phrases that match this condition in the YCOE treebank are questions, either indirect or direct (with the addition of the label -SPE). If we found a unique phrase matching the condition, we looked for the presence of a finite subordinate clause (tagged as IP-SUB or IP-SUB-SPE in case of direct speech). Then, as in the previous rule, we applied the rules described in Sections from 3.2.1 to 3.2.4.

3.2.6. Rule COORD_VB

We describe here the last rule we designed, which is aimed at determining the root in sentences where two coordinated elements could potentially both be assigned the root role. We only focused on the situation in which the two coordinates were verbs other than 'to have', 'to be' or modals. In these cases, the extraction of isolated tokens resulted in an empty set (or a set consisting of elements which could not be the root of a sentence). We then looked for a phrase whose tag starts with VB and within that phrase we assigned the root label to the first coordinate, as per the UD guidelines.
The application of these rules didn't always yield the correct root. In certain instances, we were unable to identify a root. In Section 4, we present the outcomes and analyze specific cases.

4. Results

In this section, we describe the results obtained by parsing the YCOE treebank following the rules described in Section 3.
In our study, we analyzed a sample of 390 sentences from Old English texts to assess the performance of our rule-based algorithm. Our objective was to identify the root of each sentence accurately and assign the appropriate label.

correct	wrong	missing	total
349	24	17	390

Table 1: Results of the rule-based root identification.

As Table 1 shows, in 349 out of 390 sentences, following our rule-based approach, we were able to identify the root of the dependency trees correctly. For 24 sentences the root label was assigned to the wrong token, while the 17 cases of 'missing root' were the ones that did not fall in any situation described in our set of rules. Compared to the results obtained by Brigada Villa and Giarda (2023), we can see that the rule-based approach described in this paper reached far better results considering only the root dependency relation (89.49% vs. 78.46%).[8]

[8]The comparison was made replicating the steps described in the GitHub repository of the paper: https://github.com/unipv-larl/wundorsmitha-geweorc/tree/main/paper_projects/parsing_oe_modern.

5. Discussion

In this section, we will analyze the errors made by the model, first addressing the missing roots, i.e. where the model did not succeed in assigning the `root` to any token, and then discussing the wrong roots.

Concerning missing roots, the high majority of them consists in sentence fully or partially in Latin, in which the root is a Latin word. This happens because Latin words are tagged as `FW` in the YCOE, regardless of their actual POS. An example of it is sentence coaelhom,ÆHom_1:23.11 in Figure 5.

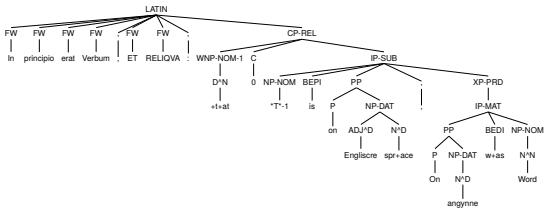

Figure 5: Tree of the sentence `coaelhom,ÆHom_1:23.11` whose translation is '*In principio erat Verbum, et reliqua*: that is in the English language "At the beginning there was the Word"'

This sentence comes from a homily, in which the author provides a biblical verse in Latin, immediately followed by its translation in Old English. Despite the presence of Old English words, the root of this sentence is in the Latin part. Out of the 17 missing roots, 10 of them are in Latin sentences. The rest of the sentences are nominal ones, e.g. *& eft þurh Adam on his forgægednysse.* 'And again through Adam in his transgression.' (coaelhom,ÆHom_1:189.109_ID).

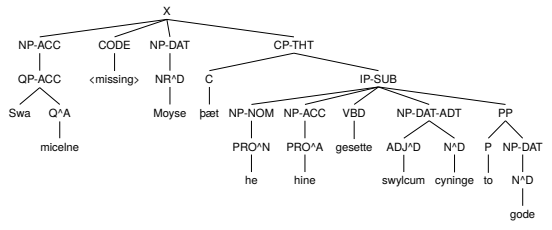

Figure 6: Tree of the sentence `coaelhom,ÆHom_1:370.193` whose translation is 'so much [...] to Moyses, that he had appointed him god of such king (...)'

Some exceptions to this generalization are, for example, sentence coaelhom,ÆHom_1:370.193 (Figure 6), which contains some missing fragments, or sentence coaelhom,ÆHom_1:41.25 ((Figure 7)), which has the structure of a subordinate clause, introduced by *ac þæt* 'but that', which was not united to the previous one due to the period ending the preceding sentence. In this latter case, in which the sentence starts with a subordinator, but without a

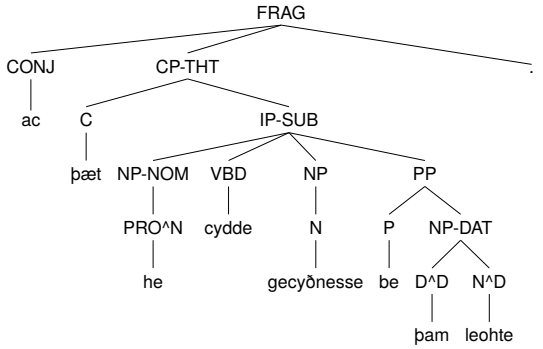

Figure 7: Tree of the sentence `coaelhom,ÆHom_1:41.25` whose translation is 'But so that he announced the witness of the light.'

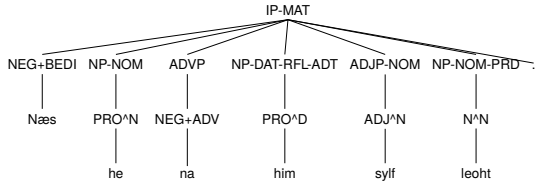

Figure 8: Tree of the sentence `coaelhom,ÆHom1_1:41.24` whose translation is 'He himself is not light'.

main clause, an ad-hoc rule could be implemented to enhance the results of the conversion.

As far as wrong attribution of the root is concerned, most of them are connected to the difficulty to discern between copular and existential `BE`. In some cases, only the broader context allows one or the other interpretation. Other errors are linked to the fact that some sentences have a nominal main clause, followed by some subordinates. An example worth discussing is the following: in sentence coaelhom,ÆHom1_1:41.24, *Næs he na him sylf leoht* (Figure 8), the negated verb *nisan* 'not to be' is not recognized as a copula because its YCOE tag was `NEG+BEDI`. This happens because, differently from the `VB`, `HAVE`, `BE_ROOT` and `MD` rules, we could not add the `NEG+` tag to the `BE_COPULA` rule, since it could have been confused with a previous rule and hinder the correct recognition of it.

One last point worth mentioning, is that in sentences such as coaelhom,ÆHom_1:364.192, *Nu ic þe sette, cwæð God sylf to him, þæt þu beo [text missing] Pharaones god [...]* (Figure 9), in which a speech verb interrupts the reported speech content, the model correctly recognizes the verb in the direct speech *sette* 'establish' as the root. The fact that YCOE annotates the interruption as `-PRN`, i.e. appositive or parenthetical, constitutes easy material for the further steps of the conversion since also UD considers these cases as parenthetical parataxis.[9]

[9] https://universaldependencies.org/u/

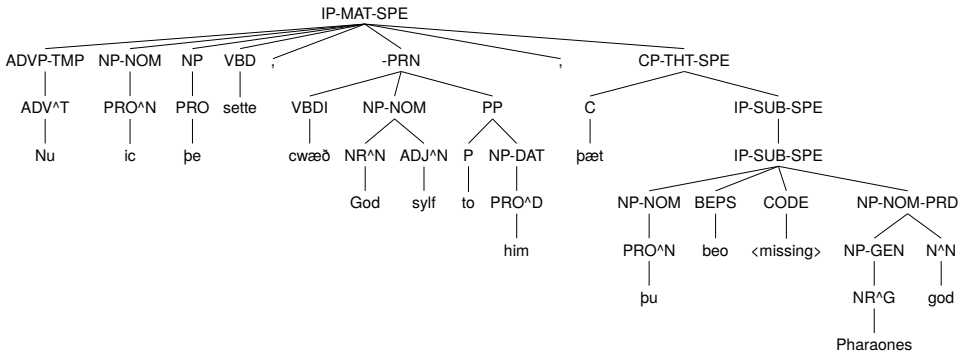

Figure 9: Tree of the sentence `coaelhom,ÆHom_1:364.192` whose translation is 'Now I establish for you - said God himself to him, that you be [text missing] the God of the Pharaon [...]'.

6. Conclusions

This paper is the first step towards the creation of a UD treebank for Old English through an automatic conversion of the YCOE treebank from its original constituency format. Since the root is the node from which every other depends, we started with a root-identification task, in which we defined a set of rules to automatically identify the root of a dependency tree, starting from the original YCOE constituency annotation. Given that UD allows only some word classes as roots, we used the original YCOE POS tags as the basis of our rules. After describing Old English morpho-syntax (section 2), we presented, in section 3, our dataset, consisting of manually annotated sentences, and the rules we implemented: section 3.2.1 deals with rule `VB`, sections 3.2.2 and 3.2.3 present rules concerning the verb 'to be' (`BE_INF` and `BE_COPULA`). Rules `HAVE`, `BE_ROOT` and `MD`, described in section 3.2.4, concern verbs which are generally used as auxiliaries, but can nonetheless be the root of a sentence in their lexical meaning. Finally, we presented rules `IP-MAT-0` and `CP-QUE` in section 3.2.5, and rule `COORD_VB` in section 3.2.6, used when the set of isolated tokens is empty. Our results, discussed in sections 4 and 5, show a precision of 89,23%, thus showing a better performance than a multilingual parser (Brigada Villa and Giarda, 2023). Error analysis has demonstrated that the main errors are due to three factors: a) the presence of Latin sentences; b) the presence of nominal sentences; and c) the difficulty in the disambiguation of copular and existential uses of the verb 'to be'. To conclude, this paper represent a first attempt towards an automatic rule-based conversion of the YCOE annotation into the UD roots and the first step towards the conversion of the whole treebank. The errors analysis may provide a starting point for the implementation of the rules. The use of parsing models for Latin can be used to parse Latin sentences included in the Old English text, in order to have a correct annotation of both languages.

`dep/parataxis.html#reported-speech`

Acknowledgements

We would like to express our gratitude to three anonymous reviewers, whose comments have greatly contributed to improve this paper. This article results from the joint work of the authors. For academic purposes, Martina Giarda is responsible for the manual annotation of the sentences and for Sections 1, 2 and 5 and Luca Brigada Villa is responsible for the code and Sections 3, 4 and 6.

Bibliographical References

Javier Arista. 2022a. Old english universal dependencies: Categories, functions and specific fields. In *Proceedings of the 14th International Conference on Agents and Artificial Intelligence - Volume 3: ICAART,*, pages 945–951. INSTICC, SciTePress.

Javier Arista. 2022b. Toward the morpho-syntactic annotation of an old english corpus with universal dependencies. *Revista de Linguistica y Lenguas Aplicadas*, 17:85–97.

Þórunn Arnardóttir, Hinrik Hafsteinsson, Einar Freyr Sigurðsson, Kristín Bjarnadóttir, Anton Karl Ingason, Hildur Jónsdóttir, and Steinþór Steingrímsson. 2020. A Universal Dependencies conversion pipeline for a Penn-format constituency treebank. In *Proceedings of the Fourth Workshop on Universal Dependencies (UDW 2020)*, pages 16–25, Barcelona, Spain (Online). Association for Computational Linguistics.

Luca Brigada Villa and Martina Giarda. 2023. Using modern languages to parse ancient ones: a test on Old English. In *Proceedings of the 5th Workshop on Research in Computational Linguistic Typology and Multilingual NLP*, pages 30–41, Dubrovnik, Croatia. Association for Computational Linguistics.

Sabine Buchholz and Erwin Marsi. 2006. CoNLL-X shared task on multilingual dependency parsing. In *Proceedings of the Tenth Conference on Computational Natural Language Learning (CoNLL-X)*, pages 149–164, New York City. Association for Computational Linguistics.

James E. Cross and Thomas D. Hill. 1982. *The Prose Solomon and Saturn and Adrian and Ritheus*. University of Toronto Press, Toronto.

Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal Dependencies. *Computational Linguistics*, 47(2):255–308.

Olga Fischer, Ans van Kemenade, Willem Koopman, and Wim van der Wurff. 2001. *The Syntax of Early English*. Cambridge Syntax Guides. Cambridge University Press.

Anton Karl Ingason, Hrafn Loftsson, Eiríkur Rögnvaldsson, Einar Freyr Sigurðsson, and Joel C. Wallenberg. 2014. Rapid deployment of phrase structure parsing for related languages: A case study of Insular Scandinavian. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 91–95, Reykjavik, Iceland. European Language Resources Association (ELRA).

Bruce Mitchell and Fred C. Robinson. 2012. *A guide to Old English. Eighth edition*. John Wiley & Sons, Malden, Oxford.

Rafał Molencki. 2017. Syntax. In Laurel J. Brinton and Alexander Bergs, editors, *The History of English. Old English*, volume 2, chapter 5, pages 100–124. De Gruyter Mouton, Berlin, Boston.

Susan Pintzuk and Ann Taylor. 2006. *The Loss of OV Order in the History of English*, chapter 11. John Wiley Sons, Ltd.

John C. Pope. 1968. *Homilies of Ælfric: a Supplementary Collection*. Early English Society, Oxford University Press, London.

Eiríkur Rögnvaldsson, Anton Karl Ingason, Einar Freyr Sigurðsson, and Joel Wallenberg. 2012. The Icelandic parsed historical corpus (IcePaHC). In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 1977–1984, Istanbul, Turkey. European Language Resources Association (ELRA).

Ferdinand von Mengden. 2017a. Morphology. In Laurel J. Brinton and Alexander Bergs, editors, *The History of English. Old English*, volume 2, chapter 5, pages 73–99. De Gruyter Mouton, Berlin, Boston.

Ferdinand von Mengden. 2017b. Old english: Overview. In Laurel J. Brinton and Alexander Bergs, editors, *The History of English. Old English*, volume 2, chapter 3, pages 32–49. De Gruyter Mouton, Berlin, Boston.

Language Resource References

Kroch, Anthony and Taylor, Ann. 2000. *Penn Helsinki Parsed Corpus of Middle English.* Department of Linguistics, University of Pennsylvania., second. [link].

Pintzuk, Susan and Plug, Leendert. 2002. *The York-Helsinki Parsed Corpus of Old English Poetry (YCOEP)*. Department of Linguistics, University of York. [link].

Taylor, Ann and Warner, Anthony and Pintzuk, Susan and Beths, Frank. 2003. *The York-Toronto-Helsinki Parsed Corpus of Old English Prose (YCOE).* Department of Linguistics, University of York. [link].