Critical Apparatus as Domain-Specific Languages: A Rule-based Parser for Encoding an Eighteenth-Century Collation of Hebrew Manuscripts

Luigi Bambaci
Department of Cultural Heritage
University of Bologna
Bologna, Italy
luigi.bambaci2@unibo.it

***Abstract*—Manually encoding variant readings is a difficult and time-consuming task. Markup languages ensure data exchange and reusability, but are very difficult to handle especially in the case of texts characterized by a rich textual tradition and editions with extensive critical apparatus. Scholars engaged in digitizing printed critical editions find themselves dealing with different levels of problems, including the revision of OCR outputs and the conversion from plain text to a coherent XML encoding.**

In this article we illustrate how it is possible to exploit the structured language of critical apparatus in order to automate encoding. Finally, we discuss the significant advantages deriving from the adoption of a parsing system over a manual encoding, advantages which range from speed in data acquisition to automatic detection of misprints or inconsistencies in the printed source, as well as correction of errors originated after OCR processing and greater control over the generation of semantic errors during conversion into XML code.

Our case study concerns the digitization of a collation of Hebrew manuscripts and printed editions realized by the English scholar Benjamin Kennicott in the second half of the XVIII century.

***Index Terms*—Computational philology, digital philology, language processing, ANTLR, XML-TEI encoding, markup languages, Hebrew Bible studies, Hebrew manuscripts, Kennicott's collation**

I. Introduction

ONE of the most important and delicate phases of philological activity consists in gathering and examining documents (witnesses) that transmit the text of a literary or historical work. During this phase, called collation, philologists compare the text of different witnesses, such as manuscripts and printed editions, and record the differences (the variants or variant readings) in the critical apparatus.

The collation results are preliminary to any further philological work: the variants gathered in the critical apparatus provide scholars with all the data necessary for preparing critical editions, repertories of variants and errors, textual commentaries, and for carrying out specialized inquiries, such as studies on language and textual history.

Collating is difficult and time-consuming, especially when the number of witnesses to consult is large. For this reason collations are often one-off enterprises and may require the contribution of several collaborators. Extracting or processing data from the critical apparatus may become difficult and sometimes impossible without computer assistance.

Digital technologies can offer a valid aid to philologists preparing critical editions or studying the textual history of a work. A markup language such as the one promoted by the Text Encoding Initiative (TEI)[1] represents an excellent tool as far as digital preservation, reusability, and processing are concerned. It allows the encoding of critical apparatus according to shared standards, thus ensuring data exchange. The encoding, moreover, makes the data not only fully machine readable, but also machine actionable: by using technologies such as XQUERY or XSLT stylesheets, the data can be queried, manipulated, and transformed into different formats depending on the purpose of the research in question.

Manually encoding critical apparatus, however, is extremely costly in terms of time and effort. The great variety of textual phenomena to be encoded and the need to make explicit the information in a machine-readable form result in excessively verbose markup schemes, which are at risk of distracting the philologist and hence prejudicing the outcome of the encoding.

In this article we will illustrate how the employment of a simple rule-based parser is able to automate the encoding process and speed the acquisition of data from critical apparatus.

We intend to demonstrate how traditional, well-written critical apparatus can be automatically encoded without any need of manual intervention and without loss of information.

We were able to implement our methodology thanks to ANTLR4 software[2] and the adoption of a domain-specific languages approach (Section II).

Our case study is represented by a collation of Hebrew witnesses of the Old Testament or Hebrew Bible (HB), accomplished by the English scholar Benjamin Kennicott (K) at the end of the XVIII century [1], [2].

[1]https://tei-c.org/ (accessed May 14, 2021).
[2]https://www.antlr.org/ (accessed May 14, 2021).

The digitization of K's collation forms part of an ongoing doctoral dissertation, which aims at producing a born-digital scholarly edition of the biblical book of Qohelet (Q) in the original Hebrew. The project is based on the web annotation tool named Euporia,[3] developed at the Laboratory of Collaborative and Cooperative Philology (CoPhiLab) of the Institute of Computational Linguistics "A. Zampolli" (ILC) at Pisa.[4]

The article is structured as follows: in Section II we introduce K's work and justify our choice of a rule-based parsing system over other possible digitization technologies. In Section III we offer a brief analysis of K's apparatus, highlighting the features of its language that are relevant for parsing. In Section IV we illustrate the procedures followed to encode the apparatus, from optical character recognition analysis (Section IV-A) to XML-TEI encoding (Section IV-E). Finally, we present and discuss the results achieved (Sections V and VI) and make some general observations about possible uses of a digital database of variants of the HB in philological studies (Section VII).

II. Background

The work of K represents, along with that published some years later by De Rossi (DR) [3], [4], the only large-scale collation currently available of the HB. K gathered more than 600 witnesses[5] covering all 24 books of the HB, and collected about 1.500.000 pieces of textual information[6] in a two-volume work of about 1700 pages. The undertaking was long and difficult, involving contributors from different countries, and took over two decades to be completed.[7]

Since no other initiative of extensive collation has since been attempted or even planned, we are today mainly dependent on K and DR as far as the medieval and modern textual tradition of the HB is concerned [9].[8]

The two collations are traditionally consulted for preparing critical editions or textual commentaries of single books of the HB. More rarely, the data provided by the collations are used for large-scale inquiries on the history of the biblical text or on Hebrew language. Given the huge amount of data, more in-depth investigations are indeed problematic, and cannot be done except for small samples or by resorting to quantitative analysis with the aid of computer.

Among quantitative researches one may quote the studies of Sacchi [12] and Borbone [13], who attempted to find clusters of manuscripts on the base of shared variants, and the studies of Penkower [14], [15], who used the variants of the collations in order to identify the geographic provenance of more recently discovered manuscripts.

Textual encoding of critical apparatus could constitute a valuable resource for the study of variants in the biblical text, since it would enable the efficient analysis of large quantities of data. Building a digital corpus of Hebrew variants encoded in a standard language such as XML-TEI, moreover, could represent a good solution to issues of transparency and data reusability: the digital medium can make the research results more easily verifiable, allowing for data exchange among scholars wishing to use them for their own research.

As is well known, two main approaches exist in computational linguistics that permits expeditious information extraction through automated encoding: rule-based systems and machine-learning systems. In the first, the rules used for analysing the language are defined by the user; in the second, the rules are derived from data through complex machine-learning algorithms.

Among the latter, we might mention GROBID-dictionaries,[9] a machine-learning library based on Conditional Random Fields (CRFs) designed for parsing and encoding, in XML-TEI, entry-structured textual resources, such as lexicons and encyclopedias.

The main difference between the two approaches relies in the fact that machine-learning systems are more powerful with unstructured data, while rule-based systems are robust with texts that are semi- or highly structured.

As to critical apparatus, these are generally written in a special kind of language, which is created by the editor with the specific purpose of "saving space": indeed, because of the page constraints typical of printed works, critical annotations need to be as concise as possible, in order to avoid the redundancy inherent in natural language, while avoiding at the same time inaccuracy and ambiguity. The result is an artificial and specialized language, consisting of abbreviations, special symbols, and technical vocabulary, which is shared within a specific community of scholars.

Often the language in which critical apparatus are written presents a certain degree of formalization, in the sense that textual phenomena are described using a finite set of symbols and conventions and are listed according to a fixed order.

From a computational point of view, the kind of language that characterizes critical apparatus of this sort can be classified as Domain-Specific Language (DSL). In computer science and software engineering, DSLs are programming languages or specification languages optimized for a particular domain of knowledge [16], [17]. Unlike General Purpose Languages (GPLs), such as Java, C etc., DSLs are tailored to address problems that belong to specific domains of application: for example, among DSLs, HTML is specific for web pages, SQL for relational databases, and XSLT for XML-based languages. All DSLs are formal languages, i. e. they are subject to a

[3]Euporia is a web annotation tool based on domain-specific languages. It is currently under development as an eXist-db applicaton, see: https://poros.cophilab.ilc.cnr.it/ (accessed May 14, 2021). The source code can be found on Github: https://github.com/CoPhi/euporia (accessed May 14, 2021).

[4]https://cophilab.ilc.cnr.it/ (accessed May 14, 2021).

[5]See [5] p. 37.

[6]See [6] pp. 28 ff.

[7]For a historical account of K's enterprise see [7] and [8].

[8]To these must be added the collations of Ginsburg [10] and Döederlin and Meisner [11], which, however, largely depend on the collations of K and DR.

[9]See https://github.com/MedKhem/grobid-dictionaries (accessed May 14, 2021).

formal grammar that enables the verification of whether they are well-formed, pursuant to given formation rules. As formal languages, they can be analysed by machine, and possibly processed in order to generate code in other languages.

Let us examine an example of critical apparatus that might fit the definition of DSL outlined above. Suppose we have a textual tradition with five witnesses A, B, C, D, and E, with A being the reference text used for collation. Suppose also that we want to express in formal language the following facts: (1) reference text A reads 'sun' at verse number '1'; (2) witnesses B and C have the variant 'sky' at the same position; (3) witness D has 'sky' as well, but this is a first-hand reading that has been later corrected by a second scribe according to the reading of the reference text; (4) witness E, by contrast, omits the word.

In digital philology, the most widely adopted strategy for representing philological data in a machine-readable format is to resort to XML-based mark-up languages, such as TEI language.

Here is a possible representation in TEI of the data in our example:

```
<app loc="1">
  <lem wit="#A">sun</lem>
  <rdgGrp>
    <rdg wit="#B #C">sky</rdg>
    <rdg wit="#D">
      <choice>
        <sic>sky</sic>
        <corr>sun</corr>
      </choice>
    </rdg>
  </rdgGrp>
  <rdgGrp>
    <rdg wit="#E"/>
  </rdgGrp>
</app>
```

In XML encoding, textual data are labeled with tags (`<app>`, `<lem>` etc.) and attributes (`@loc`, `@wit` etc.), which have the function of declaring the 'meaning' of the different apparatus components: the tag `<app>` makes it explicit that an apparatus entry is being opened, and the attribute `@loc` that this refers to the variant location number one; the tag `<lem>` contains the reading of the reference text, namely the lemma, the *siglum* of which is recorded in the attribute `@wit`; and finally the readings (`<rdg>`), arranged in reading groups (`<rdgGrp>`), contain the textual variants and information about scribal interventions (`<corr>` and `@sic`).[10]

If a DSL critical apparatus is used, a possible representation of the identical information could look like the following:

1. sun A] sky B, C; primo D – ∧ E.

[10]For the elements `<app>`, `<lem>`, `<rdg>` and `<rdgGrp>`, and the attributes `@loc` and `@wit`, see Module 12 of TEI Guidelines for the encoding of critical apparatus, https://tei-c.org/release/doc/tei-p5-doc/en/html/TC.html (accessed May 14, 2021). On `<choice>`, `<corr>` and `<sic>`, see Module 11 on the representation of primary source, https://tei-c.org/release/doc/tei-p5-doc/en/html/PH.html#PHPH.

As can be seen, the language adopted in such annotation, which closely remembers that of K (Section III), is not obliged to resort to textual labels (viz. tags and attributes) in order to make the information explicit for the machine. Rather, DSL apparatus exploits the token type (string, numeral, separator, etc.), as well as the position of its components (entry, lemma, reading, etc.), in a manner which we are about to demonstrate.

Such a language is domain-specific, because it is modeled to describe facts that pertain to the domain of textual philology or textual criticism (in the present case, of the HB); and it is formal, because it can be described by a formal grammar and analysed by a parser in order to generate code in another language, such as XML.[11]

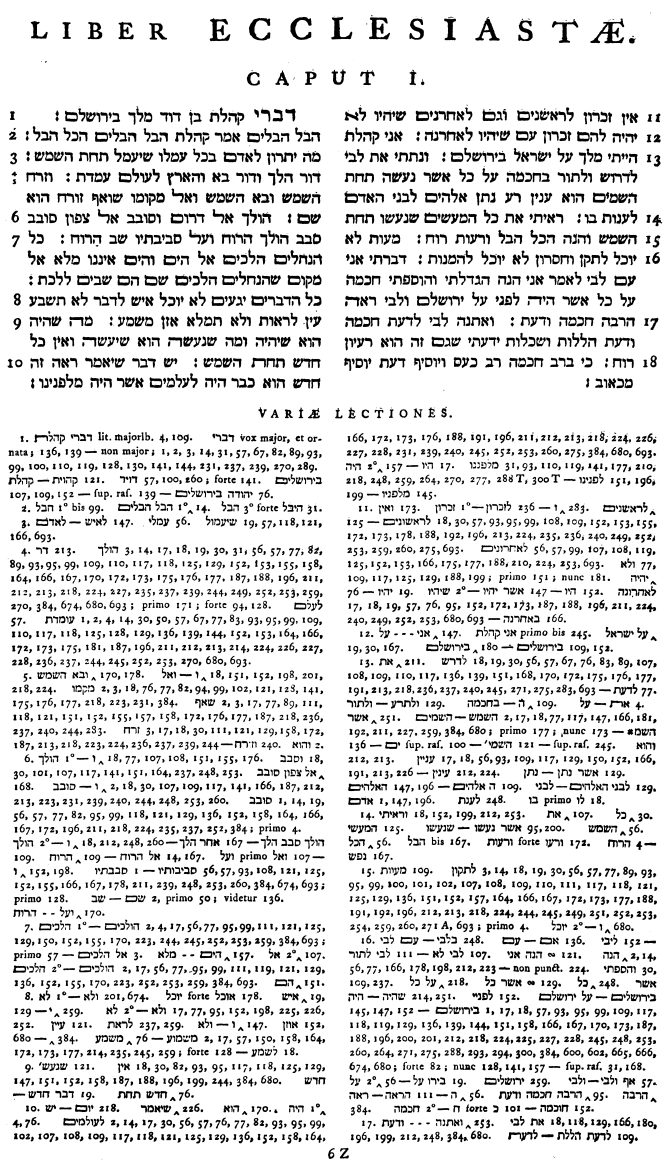
Given the special nature characterizing the language of critical apparatus under examination, we have chosen to adopt a rule-based parsing system for analysing and encoding K's work. In particular, we have relied on ANTLR4 [24], [25], a language-recognition software for generating parsers, which is widely used for creating and customizing DSLs.

In the next sections we will discuss the procedures we have employed to encode automatically the critical apparatus of that part of K's collation devoted to the book of Q, also known as Ecclesiastes. Before doing that, however, we will present the relevant features of K's collation, focusing on the language that K developed to describe variant readings in his critical apparatus.

III. Kennicott's critical apparatus

The work of K indubitably represents an exception in the field of ecdotics of the HB. Compared to other contemporary editions or collations of the HB, primarily that of DR, K's work stands out both for the size of the critical apparatus, as can be seen in the bottom part of Fig. 1, and for the type of language used in the presentation of the variants. Unlike the apparatus of DR, written in a style closer in many respects to that of textual commentaries, K's apparatus is in fact highly structured: variant readings are described in a very systematic and rigorous way, keeping to a bare minimum the use of

[11]Obviously not all critical apparatus can be classified as DSLs. The degree of formalization of a critical apparatus may vary from edition to edition, depending on editorial decisions and scholarly practices. As far as the HB is concerned, it may suffice to quote the work of DR, which is written in a far less standardized language. In recent times, however, the practice of composing critical apparatus in languages that are more easily accessible by machine seems to be gaining ground among the editions of the HB. This can be easily verified if one compares, for example, the former editions of the *Biblia Hebraica* (BH) series [18]–[20] with the most recent one, the *Biblia Hebraica Quinta* (BHQ) [21] or with the *Hebrew Bible Critical Edition* (HBCE) [22]. In this respect, the *Critique Textuelle de l'Ancien Testament* (CTAT) [23] was very important, because it was the first project to attempt a linguistic standardization (especially with regard to technical vocabulary) that has had influence on more recent editorial projects.



LIBER ECCLESIASTÆ.

CAPUT I.

1 דברי קהלת בן דוד מלך בירושלם :
2 הבל הבלים אמר קהלת הבל הבלים הכל הבל :
3 מה יתרון לאדם בכל עמלו שיעמל תחת השמש :
4 דור הלך ודור בא והארץ לעלם עמדת :
5 וזרח השמש ובא השמש ואל מקומו שואף זורח הוא שם :
6 הולך אל דרום וסובב אל צפון סובב סבב הולך הרוח ועל סביבתיו שב הרוח :
7 כל הנחלים הלכים אל הים והים איננו מלא אל מקום שהנחלים הלכים שם הם שבים ללכת :
8 כל הדברים יגעים לא יוכל איש לדבר לא תשבע עין לראות ולא תמלא אזן משמע :
9 מה שהיה הוא שיהיה ומה שנעשה הוא שיעשה ואין כל חדש תחת השמש :
10 יש דבר שיאמר ראה זה חדש הוא כבר היה לעלמים אשר היה מלפנינו :
11 אין זכרון לראשנים וגם לאחרנים שיהיו לא יהיה להם זכרון עם שיהיו לאחרנה :
12 אני קהלת הייתי מלך על ישראל בירושלם :
13 ונתתי את לבי לדרוש ולתור בחכמה על כל אשר נעשה תחת השמים הוא ענין רע נתן אלהים לבני האדם לענות בו :
14 ראיתי את כל המעשים שנעשו תחת השמש והנה הכל הבל ורעות רוח : מעות לא יוכל לתקן וחסרון לא יוכל להמנות :
15 דברתי אני עם לבי לאמר אני הנה הגדלתי והוספתי חכמה על כל אשר היה לפני על ירושלם ולבי ראה הרבה חכמה ודעת :
16 ואתנה לבי לדעת חכמה ודעת הללות ושכלות ידעתי שגם זה הוא רעיון רוח :
17 כי ברב חכמה רב כעס ויוסיף דעת יוסיף מכאוב :
18

VARIÆ LECTIONES.

1. דברי קהלת Lit. majoribb. 4, 109. — דברי vox major, et ornata; 136, 139 — non major; 1, 2, 3, 14, 31, 57, 67, 82, 89, 93, 99, 100, 110, 119, 128, 130, 141, 144, 231, 237, 239, 270, 289. — בן דוד מלך בירושלם 121. קהלת — 107, 109, 152 — sup. raf. 139 — יהודה בירושלים 76.

2. הבל 1° bis 99. — הבל הבלים 1°_14. הבל 3° forte היבל 31.

3. לאדם — לאיש 147. עמלי 56. שיעמול 19, 57, 118, 121, 166, 693.

4. דר 213. הולך 3, 14, 17, 18, 19, 30, 31, 56, 57, 77, 82, 89, 93, 95, 99, 109, 110, 117, 118, 125, 129, 152, 153, 155, 158, 164, 166, 167, 170, 172, 173, 175, 176, 177, 187, 188, 196, 211, 212, 213, 218, 223, 227, 235, 237, 239, 244, 249, 252, 253, 259, 270, 384, 674, 680, 693 ; primo 171 ; forte 94, 128. — לעולם 57. עומדת 1, 2, 4, 14, 30, 50, 57, 67, 77, 83, 93, 95, 99, 109, 110, 117, 118, 125, 128, 129, 136, 139, 144, 152, 153, 164, 166, 172, 173, 175, 181, 187, 196, 211, 212, 213, 214, 224, 226, 227, 228, 236, 237, 244, 245, 252, 253, 270, 680, 693.

5. וזרח 3, 17, 78, 89, 111, 118, 121, 151, 152, 155, 157, 158, 172, 176, 177, 187, 218, 236, 237, 240, 244, 283. — ובא השמש 170, 178. — ואל_18, 151, 152, 198, 201, 218, 224. מקומו 2, 3, 18, 76, 77, 82, 94, 99, 102, 121, 129, 141, 175, 176, 177, 218, 223, 231, 384. — שואף 2, 3, 17, 77, 89, 111, 118, 121, 151, 152, 155, 157, 158, 172, 176, 177, 187, 218, 236, 237, 240, 244, 283. — זרח 3, 17, 18, 30, 111, 121, 129, 158, 172, 187, 213, 218, 223, 224, 236, 237, 239, 244 — הזרח 240. והוא 2.

6. הולך 1° — 18, 77, 107, 108, 151, 155, 176. — וסובב 18, 30, 101, 107, 117, 141, 151, 164, 237, 248, 253. — אל צפון סובב_2, 18, 30, 107, 109, 117, 141, 166, 187, 212, 213, 223, 231, 239, 240, 244, 248, 253, 260. — סובב_1, 14, 19, 56, 57, 77, 82, 95, 99, 118, 121, 129, 136, 152, 158, 164, 166, 167, 172, 196, 211, 218, 224, 235, 237, 252, 384 ; primo 4. — הולך 2° — 18, 212, 248, 260 — הלך 167 — אחר הלך 109. — ואל primo 14, 167. — הרוח_109 — 152, 198. — סביבותיו 56, 57, 93, 108, 121, 125, 152, 155, 166, 167, 178, 211, 239, 248, 253, 260, 384, 674, 693 ; primo 128. — שם 2, primo 50 ; videtur 136. — ועל_170.

7. הולכים 1° 2, 4, 17, 56, 77, 95, 99, 111, 121, 125, 129, 150, 154, 155, 170, 223, 244, 245, 252, 253, 259, 384, 693 ; primo 57 — הולכים 2° 2, 17, 56, 77, 95, 99, 111, 119, 121, 129, 136, 152, 155, 170, 223, 252, 253, 259, 384, 693. — שם 151.

8. לא 1° — ולא 201, 674. — יוכל forte 178. — איש_19, 129 — 259. — לא 2° — ולא 17, 77, 95, 152, 198, 225, 226, 252. — עין 121. — ולא 237, 259. — אזן_147. 152, 680 — 384. — משמע 2, 17, 57, 150, 158, 164, 172, 173, 177, 214, 235, 245, 259 ; forte 128 — לשמע 18.

9. שנעשה 121. — אין 18, 30, 82, 93, 95, 117, 118, 125, 129, 147, 151, 152, 158, 187, 188, 196, 199, 244, 384, 680. — חדש — דבר חדש 19. — חדש תחת_76.

10. יש — יום 218. — שיאמר_226. — היה 1°_170. 4, 76. — לעולמים 2, 14, 17, 30, 56, 57, 76, 77, 82, 93, 95, 99, 102, 107, 108, 109, 117, 118, 121, 125, 129, 136, 152, 158, 164, 166, 172, 173, 176, 188, 191, 196, 211, 212, 213, 218, 224, 226, 227, 228, 231, 239, 240, 245, 252, 253, 260, 275, 384, 680, 693.

11. לראשנים_283. — זכרון 173. ואין 11. — לזכרון 1° — 236 — 125 — לראשונים 18, 30, 57, 93, 95, 99, 108, 109, 152, 153, 155, 172, 173, 178, 188, 192, 196, 213, 224, 235, 236, 240, 249, 252, 253, 259, 260, 275, 693. — לאחרונים 56, 57, 99, 107, 108, 119, 125, 152, 153, 166, 175, 177, 188, 210, 224, 253, 693. — לא 77, 109, 117, 125, 129, 188, 199 ; primo 151 ; nunc 181. — יהיה להאהרונה 152. — היו 147 — אשר יהיו 2° שיהו 19. — יהיו 76 — 17, 18, 19, 57, 76, 95, 152, 172, 173, 187, 188, 196, 211, 224, 240, 249, 252, 253, 680, 693 — באחרונה 166.

12. על ישראל_245. — primo bis אני קהלת 147. — על_19, 30, 167. — בירושלם_180 — בירושלם 109, 152.

13. את_211. לדרש 18, 19, 30, 56, 57, 67, 76, 83, 89, 107, 108, 109, 110, 117, 136, 139, 151, 168, 170, 172, 175, 176, 177, 191, 213, 218, 236, 237, 240, 245, 271, 275, 283, 693 — לדעת 77. — ולתור — ולתרע 129. בחכמה_109. — על — את 4.

14. אשר_251. השמש — השמים 2, 17, 18, 77, 117, 147, 166, 181, 192, 211, 227, 259, 384, 680 ; primo 177 ; nunc 173 — השמש 136 — sup. raf. 100 — השמי 121 — sup. raf. 245. — והנה 212, 213. — עניין 17, 18, 56, 93, 109, 117, 129, 150, 152, 166, 191, 213, 226 — ענין 212, 224. — אשר נתן — נתן 129. — לבני האלהים — לבני האלהים 147, 196 — ה אלהים 109. — בו primo 18. לא 1, 147, 196. — לעמת 248. — את_107. כל_30. — השמש 95, 200. — אשר נעשו — שנעשו 125. — הרוח 4 — הבל bis 167. — ורעיון forte 172. — יוכל 2°_680.

15. לתקן 3, 14, 18, 19, 30, 56, 57, 77, 89, 93, 95, 99, 100, 101, 102, 107, 108, 109, 110, 111, 117, 118, 121, 125, 129, 136, 151, 152, 157, 164, 166, 167, 172, 173, 177, 188, 191, 192, 196, 212, 213, 218, 224, 244, 245, 249, 251, 252, 253, 254, 259, 260, 271 A, 693 ; primo 4. — יוכל 2°_680. — לבבי 248. — אם — עם 136. — ליבי 152 — הנה_2, 14, 56, 77, 166, 178, 198, 212, 223 — non punct. 224. — לבי לא — 111 לבי לתור 107. — הנה אני 121. — והספתי 30, 109, 237. — אשר 248. — כל_218. — על כל 129. — בירושלם — על ירושלם 152. — לפני 214, 251. — שהיה — היה 145, 147, 152 — בירושלם 1, 17, 18, 57, 93, 95, 99, 109, 117, 118, 119, 129, 136, 139, 144, 151, 158, 166, 167, 170, 173, 187, 188, 196, 200, 201, 212, 218, 224, 225, 227, 228, 245, 248, 253, 260, 264, 271, 275, 288, 293, 294, 300, 384, 600, 602, 665, 666, 674, 680 ; forte 82 ; nunc 128, 141, 157 — fup. raf. 31, 168. — אף ולבי — ולבי 259. — בירושלים 19. — על_56 — על 57. — הרבה_95. — הרבה חכמה ודעת 111 — הראה — ראה_56. — חכמה 384. — ח חכמה 101 — כ forte 152. — דעת 17. — ואתנה - - - את לבי_253. — את 18, 118, 129, 166, 180, 196, 199, 212, 248, 384, 680. — לדעת הללות — לדעת 109.

6 Z

Fig. 1: Sample page from K's collation (Q. 1:1-17)

natural language.[12]

As can be seen from Fig. 1, two main textual flows can be distinguished on the page: the reference Hebrew text on the top, which is taken from the edition of the HB by Evan van der Hooght (Amsterdam, 1705), and the critical apparatus of variant readings ('variae lectiones') on the bottom.

[12]This feature is perhaps due to the fact that K was obliged to coordinate the work of many collaborators from different institutions in Europe: a common encoding scheme with relatively few brief editorial instructions likely served the purpose of facilitating and make uniform the results of the collation. A certain tendency towards formalization and standardization of the language of critical apparatus, however, is one of the features of XVIII century philology and can be found in other works similar in structure and purpose to that of K. In the field of biblical studies, one may cite, for example, the collation of the Greek Old testament realized by Holmes and Parson [26], or the great editions of the New Testament of Bengel and Wettstein, see Kenney [27] pp. 203 ff.

In the apparatus, the variants are presented by verse, each of which starts on a newline and is indicated by an Arabic number followed by a dot ('1.', '2.', '3.' etc.).[13] After the verse number, there are one or more apparatus entries, which represent points in the text for which variants are attested. In an apparatus entry the variants are listed. These consist of Hebrew alphabetic characters (or, alternatively, of a description of the reading with symbols or Latin terms), and witness *sigla* that identify manuscripts or printed editions and that are mostly numeric.

Let us take a detail of critical apparatus from verse number three (Fig. 2).

2. הבל 1° bis 99. הבל הבלים 1°_14. הבל 3° forte היבל 31. 3. לאדם — לאיש 147. עמלי 56. שיעמול 19, 57, 118, 121, 166, 693.

Fig. 2: Detail of apparatus at Q 1:2,3

Here, after the place of variation ('3.'), we find the following apparatus entries, each separated by a dot and a longish white space:[14]

1) **לאדם** — **לאיש** 147.
2) **עמלי** 56.
3) **שיעמול** 19, 57, 118, 121, 166, 693.

The first entry means that for the word '**לאדם**' of K's reference text, the variant '**לאיש**' is attested in witness number '147'. As can be seen, there are two Hebrew words, separated by a long horizontal bar '—': the first is the lemma, i. e. the word of the reference text; the second is the reading of the manuscript that differs from the lemma, namely the variant.

When the words to which variants refer can be easily inferred by the reader, then the lemma is missing, as in the second and third entries: here, the variants '**עמלי**' and '**שיעמול**' are implicitly linked to the words '**עמלו**' and '**שיעמל**' of the reference text, which are not reported in the apparatus. Conversely, when identical words occur in the text, a numerical value is appended to the lemma in order to disambiguate. This can be seen at verse two, where the lemma is accompanied by numerical values ('1°', '3°') that correspond to the number of occurrence of the word in the reference text: thus, '**הבל** 1° bis 99' means that the first occurrence ('1°') of the word '**הבל**' at verse two is repeated twice ('bis') in witness '99'; '**הבל הבלים** 1°_14' that the first occurrence of the span is missing ('_') in witness '14'; and finally '**הבל** 3° forte **היבל** 31' that the third occurrence of the lemma '**הבל**' seems to be spelled as '**היבל**' by a first hand ('forte').

These examples show the basic structure of K's apparatus, with one or more entries containing just one variant each. It happens frequently, however, that more readings are attested for a given lemma. In such instances, readings are divided by separators, forming distinct reading groups. Two main cases may occur: (1) two or more different readings are placed side

[13]Larger unities are the chapters (not shown in the example), see note 15.

[14]These are encoded as tabulations during the pre-processing phase, see Section IV-B.

by side, in which case the separator used is a long horizontal bar ('—'); (2) one or more 'marked' readings appear alongside the normal or 'unmarked' one. In this latter case, the unmarked reading is reported first; then, separated by a semicolon or a comma, marked readings are listed. Markedness relates to the presence of several traits, such as writing on erasure ('sup. ras.'), copyist's hand attribution ('primo' and 'forte' for first hand, 'nunc' for second hand), uncertainty ('videtur', 'forte'), and others. In this way, witnesses sharing the same reading, regardless of details of the writing support, are grouped together and aligned.

We can observe both these phenomena in, for example, the first apparatus entry at verse seven (Fig. 3).

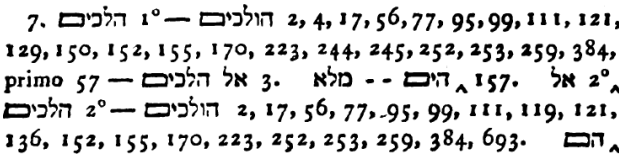


Fig. 3: Detail of apparatus at Q 1:7

This states that for the first occurrence of the lemma 'הלכים', the variant 'הולכים' is attested in twenty-three witnesses. In witness number '57', however, the same reading is attributed to the first copyist's hand ('primo'), meaning that the first copyist wrote 'הולכים', which has been later corrected to 'הלכים' by a second copyist. As can be seen, the reading of witness '57' is reported at the end of the list, separated from the other witnesses by a semicolon. Finally, at the end of the apparatus entry, the reading 'אל הלכים' of manuscript '3' is reported, after the separator '—'. Thus, we have here one apparatus entry divided into two groups: the first contains two readings, the one of the twenty-three witnesses and the corrected one of manuscript '57'; the second contains the reading of manuscript '3'.

From a formal point of view, the language of K's annotations can be described by a tree-like model, such as the one shown in Fig. 4.

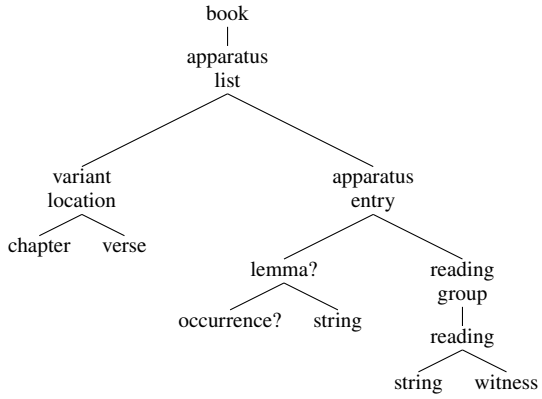


Fig. 4: Basic tree structure of K's apparatus

In the model, each book of the collation can be seen as a list of apparatus entries; each list has its own location in the text (number of chapter[15] and verse) and entails at least one apparatus entry; an apparatus entry, in turn, consists of a lemma (which can also be absent) and at least one reading group; finally, a reading group has at least one reading, which contains strings (Hebrew or Latin characters, symbols) and witness *sigla*.[16]

Besides the textual phenomena we have seen so far, several others can appear, such as *marginalia* and even textual notes (Section IV-F), each of which occupies a precise position in our tree model.[17]

One can infer from all of this the high level of formalization of K's language. Each apparatus entry is organized within a rigorous structure, which renders the meaning and function of each of its components clearly identifiable, both by human users and by machine.

In place of natural language, K employs a restricted vocabulary of technical terms to express a variety of textual phenomena ranging from the identification of the copyist's hand to the description of *mise en page* details (e. g., 'lit. majorib.', 'vox maior').

Besides conventional philological terminology, K conveys relevant information through the consistent positioning of the various components of the apparatus. Thus, for example, the lemma is always positioned at the beginning of an apparatus entry, occasionally preceded by the number of the verse in question. There follow the readings, identified by a sequence of Hebrew characters or by special symbols, and finally, by a series of numbers identifying the witnesses for each reading.

When a given reading is presented, the witnesses are listed according to a given order, as we have seen: in the first position, we find those witnesses whose readings are certain and which are not the result of revision by the copyist; these are followed, at the end of each entry, by those witnesses presenting dubious readings, first- or second-hand readings, and so on.

K segments the apparatus into individual components via the use of separators, such as punctuation marks, newlines, and tabulations, each of which is assigned a function: a semicolon to introduce the description of the material support (copyist's hands, erasures, dubious readings, etc.); a horizontal bar to divide lemmas and reading groups; a dot to separate entries;

[15]In K chapters are indicated by a double newline and are sometimes preceded by Roman numbers. On chapter division see Section IV-B and note 23.

[16]It could be argued that having groups with only one reading each is redundant, and that it would be better to create a model in which each apparatus simply consists of readings that are *eventually* organized into groups. However, we preferred to keep the model described above, for two main reasons: (1) it is easier to implement with CFG rules (Section IV-C); (2) it generates a more uniform XML output, which is easier to navigate and query. From a theoretical point of view, moreover, the existence of groups with only one element can be justified in a mathematical sense, for example with set theory, where such groups are known as singletons.

[17]The model illustrated in Fig. 4 is an abstraction of the parse tree introduced in Section IV-C. This is not the only model possible. A viable alternative could be the following: (book (chapter (verse (apparatus list (apparatus entry (lemma) (reading group | reading)...))))). No particular reason led us to prefer the first to the latter. Anyhow, both structures are easily derivable from each other in XML output through XSL transformations.

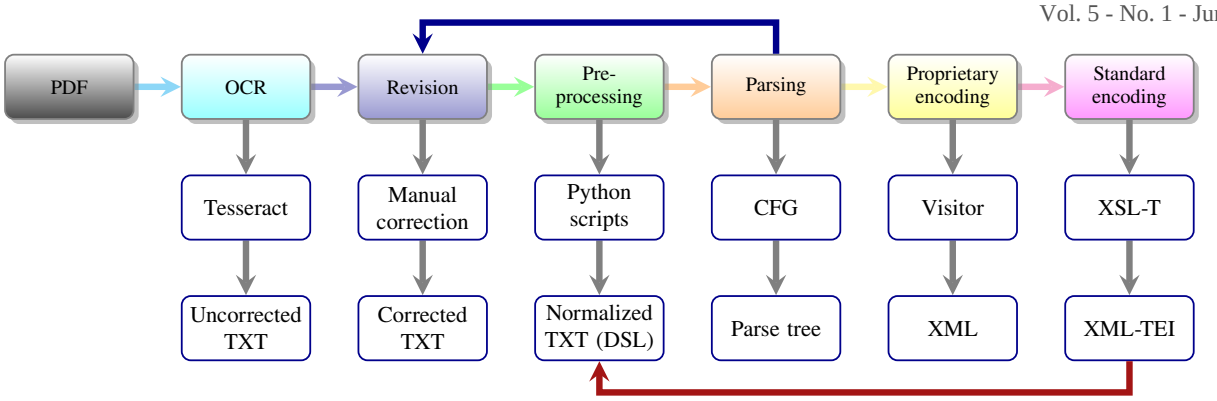


Fig. 5: Work-flow

a newline to present the next variation unit (corresponding to the relevant verse).

Each element can appear a number of times within the same entry: some must appear of necessity at least once, while others can be absent altogether, but whatever the case, everything always recurs in the identical position.

It is precisely these two features – (1) the class of string of the textual elements (letter, numeral, symbol, etc.), and (2) their position (or syntax) in the context – that make K's apparatus eminently suitable for automated processing (as compared to DR's "commentary-like" apparatus) and hence ideal for description by a simple rule-based parsing system.

IV. Methodology

In this section, we describe the procedure we followed to comprehend fully the tree-like structure of K's language and thus become able to encode automatically his critical apparatus.

The procedure can be divided into six phases, as shown in Fig. 5:

1) acquisition of textual data through Optical Character Recognition technology (OCR);
2) manual correction of the OCR output;
3) automated pre-processing and normalization;
4) composition of a Context-Free Grammar (CFG) with ANTLR4;
5) development of a general exporter to produce XML code from the parsed text;
6) creation of an XSL-T stylesheet to organize the XML code according to TEI standards.

The results of phases 2-6 (from the normalized `.txt` version of K's apparatus to the TEI-compliant encoding) can be found in our Github repository.[18]

Each of these phases is discussed in detail below.

A. Optical Character Recognition

As a first step, we acquired a `.pdf` copy of **k**'s work, which is freely available on platforms such as Archive.org[19] and Google Books.[20]

We then divided the `.pdf` into sections corresponding to the division in biblical books and processed the part containing the apparatus of Q,[21] through the use of the OCR software Tesseract.[22]

The output resulting from OCR processing contains many errors, as can be seen in Fig. 6b. These entail in particular omissions, substitutions of characters, and wrong text segmentation. Treating K's apparatus with OCR technology is indeed difficult, for three main reasons: (1) the apparatus contains two different alphabets (Latin and Hebrew), with two different textual flows (from left to right and from right to left); (2) special symbols occur, like '$_\wedge$' for omissions and '∞' for transpositions; (3) all the elements in K's apparatus, including punctuation, newlines, and tabulations, must be retained as they stand in the original source because they are important for proper parsing.

The text resulting from the OCR must therefore be corrected by hand carefully and consistently.

As we will see in Section IV-B, however, manual correction can be facilitated by the parser, which allows for the identification of certain errors thanks to lexical and syntactical analysis.

The `.txt` file obtained after correction is a faithful textual representation of K's printed apparatus (Fig. 6c). This file maintains the division into columns of the original source, with the identical hyphenation and disposition of textual elements.

B. Manual correction and preprocessing

After a first manual correction, we subjected the text to an automated pre-processing phase through simple Python scripts and regular expressions.

Pre-processing concerned three main kinds of intervention on the text:

1) adding chapter numbers before verses;
2) normalizing newlines (verse level);

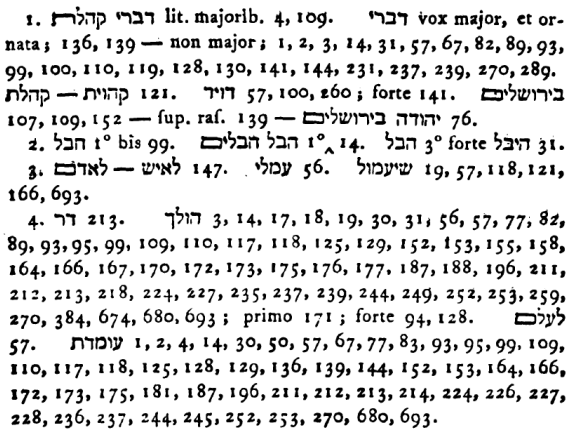
[18]https://github.com/LuigiBambaci/Kennicott (accessed May 14, 2021).

[19]https://archive.org/details/vetustestamentum02kenn (accessed May 14, 2021).

[20]https://books.google.it/books?id=ksIL59ZdPVwC&pg (accessed May 14, 2021). At the time of the work on K's apparatus, we found and used a binarized OCR version of Q on Google Books, which no longer seems to be accessible. The link we have given contains a non-binarized version.

[21]The upper page containing the reference text has been excluded.

[22]https://tesseract-ocr.github.io/ (accessed July 13, 2020).



1. דברי קהלת lit. majorib. 4, 109. דברי vox major, et ornata; 136, 139 — non major; 1, 2, 3, 14, 31, 57, 67, 82, 89, 93, 99, 100, 110, 119, 128, 130, 141, 144, 231, 237, 239, 270, 289. קהלת — קהוית 121. דויד 57, 100, 260; forte 141. בירושלים 107, 109, 152 — sup. raf. 139 — יהודה בירושלים 76.
2. הבל 1° bis 99. הבל הבלים 1°‸14. הבל 3° forte היבל 31.
3. לאדם — לאיש 147. עמלי 56. שיעמול 19, 57, 118, 121, 166, 693.
4. דר 213. הולך 3, 14, 17, 18, 19, 30, 31, 56, 57, 77, 82, 89, 93, 95, 99, 109, 110, 117, 118, 125, 129, 152, 153, 155, 158, 164, 166, 167, 170, 172, 173, 175, 176, 177, 187, 188, 196, 211, 212, 213, 218, 224, 227, 235, 237, 239, 244, 249, 252, 253, 259, 270, 384, 674, 680, 693; primo 171; forte 94, 128. לעלם 57. עומדת 1, 2, 4, 14, 30, 50, 57, 67, 77, 83, 93, 95, 99, 109, 110, 117, 118, 125, 128, 129, 136, 139, 144, 152, 153, 164, 166, 172, 173, 175, 181, 187, 196, 211, 212, 213, 214, 224, 226, 227, 228, 236, 237, 244, 245, 252, 253, 270, 680, 693.

(a) Detail of K's apparatus in `.pdf` (Q 1:1-4)

```
1. ppp 37 litorajorib. 4, 10g. – דבר' vox major, et or-
mata; 136, 139 – non major; 1,2, 3, 14, 31, 57, 67, 82, 89, 93,
99, 100, 110, 119, 128, 130, 141, 144, 231, 237, 239, 270, 28g.
pbnp–nmnp i21. "1 57,100,260; forte 141. בירושלם
107, 109, 152 – fup. raf. 139 – t?wrva mv 76.

forte 531 51.4 הבל 1°פ 9 | הבל הבליכם 1%, 34. = הבל 3?

1. לאדבם -- לאיש 147. עמלי. 56 | שיעמול 19,
.695 ,166

4. Up zig. ov 3, 14. 17, 18, 19, 30, 31, 56, 57, 77; 82,
89, 93:95, 99. 109, 110, 117, 118, 125, 129, 152, 153, 155» 158,
164, 166, 167,170, 172, 173» 175, 176, 177, 187, 188, 196, 211,
212, 213, 218, 224, 227, 235) 237» 239» 244 240, 252, 253, 259,
270, 384, 674, 680,693 ; primo 171; forte 94, 128. – t5yb
57- T'IDW. 1,2, 4» 14» 30, 50, 57, 67, 77,83, 93: 95: 99. 109,
110, 117, 118, 125, 128, 129, 136, 139, 144, 152, 153, 164, 166,
172, 173» 175, 181, 187, 196, 211, 212, 213, 214 224, 226, 227,
228, 236, 237, 2444 245) 252, 253, 270, 680, 695.
```

(b) `.txt` after OCR

```
1. דברי קהלת lit. majorib. 4, 109. דברי vox major, et orn-
ata ; 136, 139 – non major ; 1, 2, 3, 14, 31, 57, 67, 82, 89, 93,
99, 100, 110, 119, 128, 130, 141, 144, 231, 237, 239, 270, 289.
קהלת – קהוית 121. דויד 57, 100, 260 ; forte 141. בירושלים
107, 109, 152 – sup. ras. 139 – יהודה בירושלים 76.
2. הבל 1° bis 99. הבל הבלים 1° . 14. הבל 3° forte היבל 31.
3. לאדם – לאיש 147. עמלי 56. שיעמול 19, 57, 118, 121,
166, 693.
4. דר 213. הולך 3, 14, 17, 18, 19, 30, 31, 56, 57, 77, 82,
89, 93, 95, 99, 109, 110, 117, 118, 125, 129, 152, 153, 155, 158,
164, 166, 167, 170, 172, 173, 175, 176, 177, 187, 188, 196, 211,
212, 213, 218, 224, 227, 235, 237, 239, 244, 249, 252, 253, 259,
270, 384, 674, 680, 693 ; primo 171 ; forte 94, 128. לעלם
57. עומדת 1, 2, 4, 14, 30, 50, 57, 67, 77, 83, 93, 95, 99, 109,
110, 117, 118, 125, 128, 129, 136, 139, 144, 152, 153, 164, 166,
172, 173, 175, 181, 187, 196, 211, 212, 213, 214, 224, 226, 227,
228, 236, 237, 244, 245, 252, 253, 270, 680, 693.
```

(c) `.txt` after manual correction

```
1:1. דברי קהלת lit. majorib. 4, 109. דברי vox major, et ornata ;
136, 139 – non major ; 1, 2, 3, 14, 31, 57, 67, 82, 89, 93, 99,
100, 110, 119, 128, 130, 141, 144, 231, 237, 239, 270, 289.
קהלת – קהוית 121. דויד 57, 100, 260 ; forte 141. בירושלים 107, 109,
152 – sup. ras. 139 – יהודה בירושלים 76.
1:2. הבל 1° bis 99. הבל הבלים 1° . 14. הבל 3° forte היבל 31.
1:3. לאדם – לאיש 147. עמלי 56. שיעמול 19, 57, 118, 121, 166, 693.
1:4. דר 213. הולך 3, 14, 17, 18, 19, 30, 31, 56, 57, 77, 82, 89,
93, 95, 99, 109, 110, 117, 118, 125, 129, 152, 153, 155, 158, 164,
166, 167, 170, 172, 173, 175, 176, 177, 187, 188, 196, 211, 212,
213, 218, 224, 227, 235, 237, 239, 244, 249, 252, 253, 259, 270,
384, 674, 680, 693 ; primo 171 ; forte 94, 128. לעלם 57. עומדת 1, 2,
4, 14, 30, 50, 57, 67, 77, 83, 93, 95, 99, 109, 110, 117, 118, 125,
128, 129, 136, 139, 144, 152, 153, 164, 166, 172, 173, 175, 181,
187, 196, 211, 212, 213, 214, 224, 226, 227, 228, 236, 237, 244,
245, 252, 253, 270, 680, 693.
```

(d) normalized `.txt` (DSL)

Fig. 6: From `.pdf` to `.txt` format

3) encoding of multiple white spaces as tabulations (apparatus entry level).

The addition of chapter numbers (point one) is a purely graphical solution: we have preferred to add chapter numbers to make it easier to navigate the parse tree (Section IV-C) and thus facilitate the detection of possible parsing errors.[23]

Normalizing new lines (point two) means to make each verse start on a newline. Even if this is the rule in K's apparatus, it can occur that more verses are put on the same line, especially in cases where there are few variants per verse.[24] Although it is not impossible to define this phenomenon in the rules of the CFG (Section 7), we preferred to intervene directly on the raw data in order to facilitate the reading both of the normalized text and of the grammar rules.

Similar remarks can be made on point three: the series of white spaces found between apparatus entries can be better encoded as tabulations, so to make uniform the normalized text and to simplify the composition of the CFG rules.

[23] The information about chapter division is already encoded in K, see note 15. The fragment of CFG shown in Fig. 7 and the complete version found in Github (see note 18) accept texts both with chapter numbers and without.

[24] Several examples can be found in the poetical books we examined, see note 31.

At the end of the pre-processing phase we get a `.txt` file (Fig. 6d), which contains all the philological information relevant to the original source.

It is this normalized text that we treat as a DSL, that is, as a formal language suitable for analysis by machine through the application of formal rules (see next Section).

C. Parsing

Before proceeding to build the parser, we decided to encode manually the apparatus of Q in order, first, to obtain a gold standard by which to verify the accuracy of our parsing system, and second, to explore the different kinds of textual phenomena described in the apparatus and hence to identify the strategies which would better represent them in a proper XML-TEI encoding.

Thereafter, we wrote a CFG in order to describe K's language. The CFG is a set of rules that permits one to tokenize and parse sequences of strings. Through the definition of specific rules, it is possible to describe the whole structure of the critical apparatus and to enable the machine to correctly recognize the function of its different parts.

There exist two kinds of rules: tokenization rules and parser rules. The first permits the segmentation of the text

```
all      : listApp+                          ;
listApp  : loc app+                          ;
app      : lem? rdgGrp+ closeApp             ;
lem      : (w+ (occ lemSep? |
            (range w+ lemSep?) | lemSep)))   ;
rdgGrp   : (rdg+ | noteApp) rdgGrpSep?       ;
loc      : chap? verse closeLoc              ;
wits     : wit+                              ;
wit      : sigl marg? com?                   ;
sigl     : ((NUM LAT) | (NUM | LAT))         ;
w        : HEB  rasura?                      ;
closeApp : END TAB | END NL | TAB            ;
closeLoc : END                               ;
com      : COMMA                             ;
...
LAT      : [a-zA-Z]+                         ;
NUM      : [0-9]+([0-9]+)?                   ;
HEB      : [\u0590-\u05ff]+                  ;
NL       : [\n]                              ;
END      : '.'                               ;
TAB      : [\t]                              ;
COMMA    : ','                               ;
SEMCOL   : ';'                               ;
VARSEP   : '-'                               ;
...
```

Fig. 7: Fragment of Context Free Grammar

into discrete spans (tokens), which represent the minimal meaningful units of the apparatus, such as Hebrew words, separators, and numerals. The second permits the description of the syntax, i. e. the definition of the function of the different tokens according to their position within the overall structure. In ANTLR4, the first are handled by the lexer, the second by the parser, which are both automatically generated.

A fragment of CFG is shown in Fig. 7.[25] At the bottom we find the tokenization rules (in capital letters), which define, for example, the set of Unicode characters used for words in Latin alphabet (`LAT`), Hebrew words (`HEB`), integers (`NUM`), and meaningful punctuation characters (`END`, `COMMA`, `SEMCOL` etc.).

At the top the parser rules (in lower case) are listed. They state, for example, that the entire file (`all`) is encoded as a list of apparatus entries (`listapp+`) and that each apparatus consists of a lemma (`lem`), a list of reading groups (`rdgGrp+`), and a final separator (`closeApp`); the lemma is in turn composed of sequences of Hebrew words (`w+`), which are in turn encoded as sequences of Hebrew characters (`HEB`), and so on for every parser rule, according to a top-down approach. As can be seen, the data type which characterizes each token is defined by means of regular expressions, and the occurrence of a given token or parser rule is regulated by quantifiers.

From designing the CFG, we proceeded to parse the apparatus resulting after the normalization of OCR output (the DSL), thanks to the parsing system provided by ANTLR4 software.

The result of the parsing is a parse tree, a section of which is shown in Fig. 8. The parse tree is a tree-like graph that shows the syntactic structure of a language as described by a given grammar.

Different parts can be distinguished in the parse tree: the root node (not shown in the example), which corresponds to the start rule `all` of the CFG, and which contains all the other nodes; the internal nodes (e. g. `chap`, `verse` etc.); and finally the leaf nodes (e. g. `num`, `chapSep` etc.) which contain the input data.

In the tree, the node labels are taken from the parser rules, whose names were selected in such a way as to act as an *aide-mémoire* of the function of the various apparatus components.

D. *Proprietary* XML *Encoding*

Once we verified the correctness of the parsing operation, we proceeded to implement a general XML exporter employing a particular tree-walking mechanism made available in ANTLR4, named Visitor.

In our implementation, the general exporter passes through the tree's nodes and transforms them in XML tags (Fig. 9a). The result is a well-formed XML file, whose elements take their names from the parser rules and the hierarchical structure from the parse tree.

As can be seen from Fig. 9a, the encoding consists of a list of elements without attributes, containing the values of all the tokens described in the CFG and identified by the lexer, including separators, tabulations, and newlines.

Even if many element names are reminiscent of TEI vocabulary, the result of our encoding constitutes in fact a separate proprietary language, suitable for modification by XSL transformations (see next section).

E. TEI *encoding*

The final step consisted in transforming the XML code into XML-TEI encoding (Fig. 9b) through the application of a simple XSL-T stylesheet.[26]

The encoding follows the TEI model of critical apparatus, specifically the Location-referenced method, which is usually recommended for digitizing printed critical editions.[27]

The stylesheet performs several important transformations on the XML code, like the creation of the attributes `@loc` for linking critical apparatus to the reference text and `@wit` for the encoding of the witness *sigla*. Philologically insignificant data like separators and other typographical details of the printed source are eliminated.

Thanks to XSL-T, the TEI header (`<teiHeader>`) for the metadata[28] has been added and the witness *sigla* have been extracted and organized in the list of witnesses (`<listWit>`).[29]

[25] The grammar we show here is just an example and has been re-adapted for the sake of clarity. The complete version of the grammar can be found on Github, see note 18.

[26] Also available on Github, see note 18.

[27] See Module 12 of TEI Guidelines, https://tei-c.org/release/doc/tei-p5-doc/en/html/TC.html (accessed May 14, 2021).

[28] See https://tei-c.org/release/doc/tei-p5-doc/it/html/ref-teiHeader.html (accessed May 14, 2021)

[29] See https://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-listWit.html (accessed May 14, 2021).

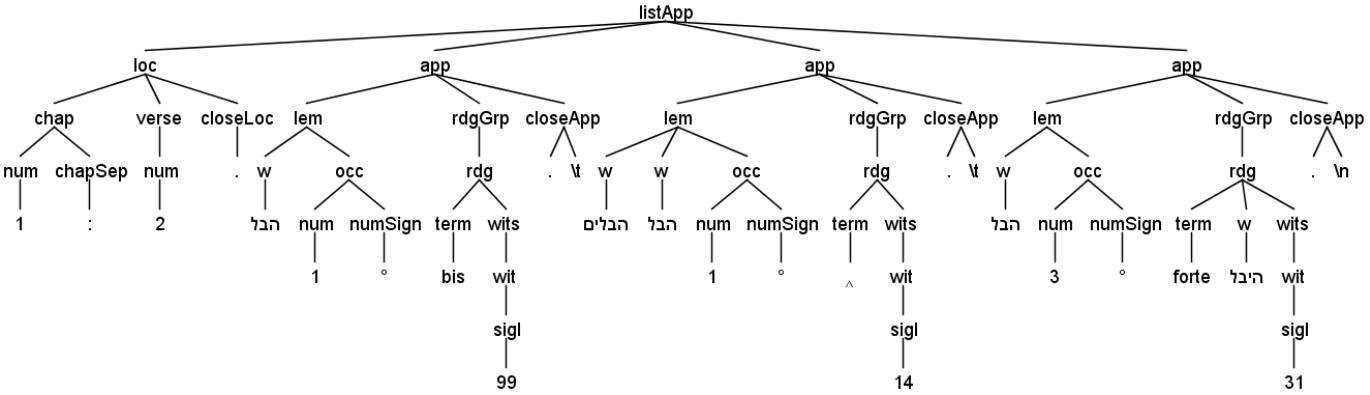


Fig. 8: Parse tree from Qohelet 1:2

```
<listApp>
  <loc>
    <chap>
      <num>1</num>
      <chapSep>:</chapSep>
    </chap>
    <verse>
      <num>2</num>
    </verse>
    <closeLoc>.</closeLoc>
    </loc>
    <app>
      <lem>
        <w>הבל</w>
        <occ>
          <num>1</num>
          <numSign>°</numSign>
        </occ>
      </lem>
      <rdgGrp>
        <rdg>
          <term>bis</term>
          <wits>
            <wit>
              <sigl>99</sigl>
            </wit>
          </wits>
        </rdg>
      </rdgGrp>
      <closeApp>.</closeApp>
    </app>
     ...
</listApp>
```

(a) XML encoding

```
<listApp>
  <app loc="1 2">
    <lem>
      <w>הבל</w>
      <num>1</num>
      <pc>°</pc>
    </lem>
    <rdgGrp>
      <rdg wit="#K99">
        <term>bis</term>
      </rdg>
    </rdgGrp>
  </app>
  <app loc="1 2">
    <lem>
      <w>הבלים</w>
      <w>הבל</w>
       <num>1</num>
       <pc>°</pc>
    </lem>
    <rdgGrp>
      <rdg wit="#K14">
        <term>^</term>
      </rdg>
    </rdgGrp>
  </app>
  <app loc="1 2">
   <lem>
     <w>הבל</w>
       <num>3</num>
       <pc>°</pc>
        ...
</listApp>
```

(b) XML-TEI encoding

Fig. 9: Conversion from XML to XML-TEI encoding (example from Qohelet 1:2)

F. Treatment of residuals

As highlighted in previous sections, one of the main features of K's apparatus consists in minimizing the recourse to natural language in the description of the variant readings.

There are instances, however, in which textual notes are used to describe complex phenomena such as transpositions or variants in text segmentation.[30]

[30]In Q, there are 7 instances on a total of 1821 variation units (the <app> elements). A similar proportion also exists in the other biblical books that have been examined, see Section V.

Such textual notes are difficult to parse, because of the mixture of technical and natural language that characterizes them, as can be seen from the examples of Fig. 10.

Incipit cap. 8 a voce חכמת, medio commatis 1 ; in 270, 655, 656, 657.

(a) Q 7:29

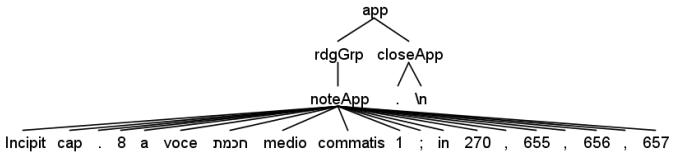
על ^ 157. יביאך ^ 80 — poſt hanc vocem ſequitur Pſal. 102, 11—22 in 200. אלהים 76, 167.

(b) Q 11:9

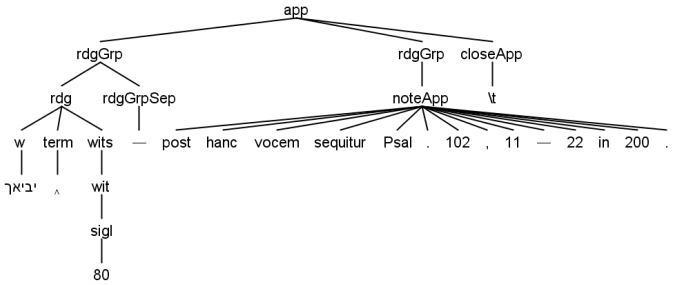
Fig. 10: Examples of textual notes

In the first example (Fig. 10a) a case of different division between chapters 7 and 8 is described, occurring in four printed editions (270, 655, 656 and 657), while in the second (Fig. 10b), a long insertion from the text of Psalms is reported. As can be seen, the use of natural language, though expressed in abridged form, is more consistent.

Given the difficulty of classifying properly the language in these and similar instances, we preferred to leave them as unstructured: we programmed the parser to recognize textual notes, by setting up the parser rule `noteApp` that takes the first word of a note and treats all its tokens as simple tree leaves. The resulting parse trees of the textual notes of Fig. 10 are shown in Figs. 11a and 11b.



(a) Parse tree of a textual note in Q 7:29



(b) Parse tree of a textual note in Q 11:9

Fig. 11: Parsing of textual notes

In XML these are encoded with the `<noteApp>` element, and in the final TEI file with the element `<note>` (Fig. 12).

```
<app>
  <rdgGrp>
    <noteApp>
      Incipit cap. 8 a voce חכמת
      medio commatis 1 ;
      in 270, 655, 656, 657
    </noteApp>
  </rdgGrp>
  <closeApp>.</closeApp>
</app>
```

(a) XML encoding (Q 7:29)

```
<app loc="7 29">
  <rdgGrp>
    <rdg>
      <note>
        Incipit cap. 8 a voce חכמת
        medio commatis 1 ;
        in 270, 655, 656, 657
      </note>
    </rdg>
  </rdgGrp>
</app>
```

(b) TEI encoding (Q 7:29)

Fig. 12: Encoding of textual notes

In this way, even if important data such as the witness *sigla* remain unprocessed at the level of parser analysis, a manual intervention on the XML encoding is possible in a second moment, if necessary.

V. Results

Once we obtained a TEI-compliant encoding of the apparatus of Q, we proceeded to confront it with the gold standard previously encoded by hand. The comparison showed us that the rule-based system was successful in correctly analysing and encoding the 2617 variant readings of the biblical book and that neither syntactic nor semantic errors were encountered.

In order to test the robustness of the parser, we decided to apply it to other biblical books of K's collation following the procedures described in previous sections.

For the sake of completeness, we chose four other books from K's collection that constitute, together with Q, the biblical unity known as "Five Megilloth" (five scrolls), namely: Song of Songs, Ruth, Lamentations, and Esther.[31]

The results showed that few syntactic errors were generated by the parser, mostly due to the presence of textual phenomena not described in Q and even misprints of the original source.

After extending the CFG rules in order to include these new phenomena, and after correcting the misprints in the normalized text, the parser produced no syntactic errors. A sample survey carried out on TEI encoding of these books seems to indicate that no semantic errors were produced there either.

It is possible to state almost categorically, therefore, that our parsing system works perfectly with regard to K, and that the other books of K's collation could similarly be parsed through extensions or modifications of the CFG.

VI. Discussion

The features of K's apparatus outlined in Section IV have enabled us to automatically digitize long lists of variant readings.

As we have seen, the encoding scheme adopted by K is rigorous, not only within the critical apparatus of the same biblical book, but also among the different books of the collation. The few variations found in the five case studies under analysis are of little import and are easily processable by the parser through small adaptations of the CFG.

The conversion to XML is straightforwardly achievable thanks to the XML exporter, which is general-purpose: unlike the CFG, which needs to be adapted according to the input data, the exporter is designed to be applicable to different DSLs, without further adjustments or customization.

Finally, the translation into XML-TEI is ensured by simple XSLT stylesheets, which allow the efficient manipulation of the XML proprietary language and its organization according to standards.

It is important to underline that there is no need to intervene manually on the `.txt` file of the DSL in order to resolve

[31] These can be found at [2] pp. 525-33 (Song of Songs), 534-9 (Ruth), 540-8 (Lamentations), 562-72 (Esther).

potential ambiguities or avoid semantic errors, for example by means of a lightweight markup: in all the five biblical books under examination, the DSL critical apparatus has been properly interpreted by the parser, and for Q the final result corresponds to the user-encoded file.

It is possible to move from the DSL apparatus to the TEI-compliant apparatus, and vice versa, as indicated by the (red) arrow in Fig. 5. The DSL and the XML-TEI encoding are in fact isomorphic, in the sense that no loss of philological information occurs when going from one format to the other. The isomorphism proves that the language of a DSL apparatus is a full-fledged formal language and that, as such, it is processable by machine without any information being added or made explicit by the user.

The employment of a parsing system for encoding critical apparatus has advantages not only in terms of time and effort, but also in terms of quality of results.

Since the tokens belong to fixed classes according to their type, and since their syntax results in a finite set of patterns which are described in detail by the parser rules, the parser permits one to correct several types of transcriptional errors generated after OCR processing as well as printing errors or inconsistencies found in the printed apparatus itself.

Most such errors consist of substitution of characters (e. g. '»' instead of a comma in witness lists, see Fig. 6b), text segmentation (e. g. 'litorajorib.' for 'lit. majorib.'), spelling of technical terms (e. g. 'fup. raf.' for 'sup. ras.'), and, more generally, errors that compromise the structure of the critical apparatus. Inconsistencies, on the other hand, may derive from different representations of the same information, such the interchange between comma and semicolon, quite frequent in K's apparatus, for introducing readings of different hands (see Section III). Lexical and syntactical analysis performed by the parser represents therefore an important tool for verifying whether the final DSL is well-formed, and has acted as a continuous support throughout the manual correction and normalization phases, as the (blue) arrow in Fig. 5 shows.

Besides controlling syntactical errors, a parser manages semantic errors as well, the latter commonly present when the encoding is performed manually. In a manual encoding, in fact, the encoders must decide, each time, which tags or attributes are more suitable for expressing their interpretation. It can therefore occur that they mistakenly adopt different encoding strategies for representing the same textual phenomenon. This can lead to an incoherent or erroneous choice of markers and can increase the possibility of semantic errors, which are sporadic and hence difficult to detect.

Our approach, by contrast, entrusts to the XSL transformation phase the decision as to the correct use of XML-TEI elements and attributes. That way, if errors turn out to be generated, they are mechanical and simple to detect and to correct, merely by modifying the XSLT code.

VII. Conclusion and perspectives

We have demonstrated, we hope, a clear methodology to encode automatically the critical apparatus of an important collation, namely, that of K. As we have seen, K adopted a rigorous encoding scheme for recording variants, which rendered his apparatus highly structured and hence eminently suitable for analysis by machine.

Given the high level of formalization of K's language, we implemented a rule-based parsing system through the tools available in ANTLR4, treating K's language as a DSL.

The parser has properly captured the tree-like structure inherent in the apparatus, allowing for an automated encoding of a huge quantity of textual data.

The advantages of using a parser over a manual encoding, however, do not concern just speed in data acquisition: as discussed in Section VI, a parser allows for a tight control on both errors generated by OCR and misprints of the original source. Semantic errors are reduced as well, since TEI markers are chosen at the end of the process during the XSL transformation phase.

Encoding K's collation could well represent a valuable resource for the study of biblical text. As stated in Section II, our knowledge of the textual tradition of the HB is still mainly mediated by the collations of K and DR. They represent, therefore, our main resource for examining the developments which the biblical Hebrew text underwent from the later centuries of the Middle Ages to the invention of printing.

A close examination of the variant readings found in manuscripts and printed editions is therefore important for textual criticism and textual history of the HB from the late Middle Ages onwards.

It permits us to trace the reception of the biblical Hebrew text within different ethno-geographic families (Oriental, Ashkenazic, Sephardic, Italian) [28] and to look into the formation of the so-called *textus receptus*, namely, that particular textual recension that imposed itself from the beginning of the XVI century all the way up to our modern printed editions [29].

Since some variants are common to other ancient traditions, such as the Greek, Latin, and Syriac versions of the HB, it is possible to verify whether they are the result of polygenesis, i. e. coincident variation [30]–[33], or remnants of ancient traditions parallel to that of the *textus receptus* [12], [34].

A distributional study of variants may have implications for other disciplines as well, such as codicology and paleography, as it would permit the classification of manuscripts whose ethno-geographic character is dubious or unknown [14], [35].

Making data processable by machine through encoding can help scholars in all these cases, since it allows the efficient handling of large amounts of data which are impossible to be analyzed manually. Textual data can be easily transformed into digital format and accommodated into data structures (data frames, databases, distance matrices etc.) for the application of quantitative methods, such as computer-assisted stemmatic analysis [36]. Other kinds of quantitative studies are possible as well, such as linguistic inquiries, and other treatments can be applied after a first-level encoding, such as parts-of-speech tagging of variant readings or semantic analysis [37].

Most importantly, unlike proprietary formats, standard languages such as XML-TEI ensure interoperability: this means that it is possible to test inter-subjectively not only the data chosen for analysis but also the criteria employed for extracting and elaborating information from them. In this way, inter-subjective control on results as well as on methods is promoted, and both repeatability and reproducibility are facilitated.

Finally, a digital critical apparatus is far easier to update, as compared to an analogic equivalent, since inaccuracies or errors found in already existing collations can be easily corrected and new variants from new witnesses easily added.

ACKNOWLEDGMENT

We would like to thank the researchers of the Italian National Council of Research (CNR), in particular the members of CoPhiLab Angelo Mario Del Grosso and Riccardo Del Gratta. A special thanks to Federico Boschetti from The Venice Centre for Digital and Public Humanities of University of Ca' Foscari of Venice (VehDPH) for his support and collaboration. The software component named general exporter (Section IV) has been designed by him.

REFERENCES

[1] B. Kennicott, *Vetus Testamentum Hebraicum cum variis lectionibus*, vol. 1. Oxford: Clarendon, 1776.
[2] B. Kennicott, *Vetus Testamentum Hebraicum cum variis lectionibus*, vol. 2. Oxford: Clarendon, 1780.
[3] G. B. De Rossi, *Variae lectiones Veteris Testamentis*. Parma: Ex regio typographeo, 1784.
[4] G. B. De Rossi, *Scholia critica in V.T. libros, seu supplementa ad varias sacri textus lectiones*. Parma: Ex regio typographeo, 1798.
[5] E. Tov, *Textual Criticism of the Hebrew Bible*. Minneapolis: Fortress Press, 3 ed., 2012.
[6] D. Barthélemy, "Les manuscrits médiévaux et le texte tibérien classique," in *Critique textuelle de l'Ancien Testament, 3. Ézéchiel, Daniel et les 12 Prophètes*, vol. 3 of *Orbis Biblicus et Orientalis*, pp. xix–xcvi, Fribourg/Göttingen: Éditions Universitaires/Vandenhoeck & Ruprecht, 1992.
[7] B. Kennicott, *The Ten Annual Accounts of the Collation of Hebrew Mss. of the Old Testament, begun in 1760, and compleated in 1769*. Oxford: Sold by Mr Fletcher and Prince, 1770.
[8] W. McKane, "Benjamin Kennicott: An Eighteenth-Century Researcher," *The Journal of Theological Studies*, vol. 28, no. 2, pp. 445–464, 1977. Publisher: Oxford University Press.
[9] B. Chiesa, *Filologia storica della Bibbia ebraica*, vol. II of *Studi Biblici*. Brescia: Paideia, 2000.
[10] C. D. Ginsburg, *The Twenty-four Books of the Holy Bible...* London: British and Foreign Bible Society, 1894. [Hebrew].
[11] J. H. Meisner and J. C. Döderlein, *Biblia hebraica...* Halle/Berlin: Libraria Orphanotrophei, 1818.
[12] P. Sacchi, "Analisi quantitativa della tradizione medievale del testo ebraico della Bibbia secondo le collazioni del De Rossi," *Oriens Antiquus*, vol. 12, pp. 1–13, 1973.
[13] P. G. Borbone, *Il libro del profeta Osea – Edizione critica del testo ebraico*. Torino: Zamorani, 1990.
[14] J. S. Penkower, "A Sheet of Parchment from a 10th or 11th Century Torah Scroll: Determining Its Type Among Four Traditions (Oriental, Sefardi, Ashkenazi, Yemenite)," *Textus*, vol. 21, no. 1, pp. 235–264, 2002. Place: Leiden, The Netherlands Publisher: Brill [Hebrew].
[15] J. S. Penkower, "The Development of the Masoretic Bible," in *The Jewish Study Bible* (A. Berlin and M. Z. Brettler, eds.), pp. 2159–2165, Oxford / New York: Oxford University Press, 2 ed., 2014.
[16] M. Fowler, *Domain-Specific Languages*. Pearson Education, 2010.
[17] L. Bettini, *Implementing Domain-Specific Languages with Xtext and Xtend*. Birmingham / Mumbai: Packt Publishing, 2 ed., 2016.
[18] R. Kittel, *Biblia Hebraica*. Lipsiae: Hinrichs, 1905.
[19] R. Kittel, *Biblia hebraica*. Stuttgart: Privil. Württ. Bibelanst., 3 ed., 1937.
[20] W. Rudorf and K. Elliger, *Biblia Hebraica Stuttgartensia*. Stuttgart: Deutsche Bibelgesellschaft, 5 ed., 1997.
[21] G. V. D. Schenker, A. Shenker, Y. A. P. Goldman, G. J. Norton, and A. V. D. Kooji, *Biblia Hebraica Quinta. Megilloth: Ruth, Canticles, Qoheleth, Lamentations, Esther*. Stuttgart: Deutsche Bibelgesellschaft, 2004.
[22] M. V. Fox, *Proverbs. An Eclectic Edition with Introduction and Textual Commentary*. Atlanta: SBL Press, 2015.
[23] D. Barthélemy, *Critique Textuelle de l'Ancien Testament*. Orbis Biblicus et Orientalis, 1982-2015.
[24] T. Parr, *Language Implementation Patterns: Create Your Own Domain-specific and General Programming Languages*. Pragmatic Bookshelf, 2010.
[25] T. Parr, *The Definitive ANTLR 4 Reference*. Pragmatic Bookshelf, 2012.
[26] R. Holmes and J. Parson, *Vetus Testamentum graecum cum variis lectionibus*. Oxford: Clarendon, 1798-1827.
[27] E. J. Kenney, G. Ravenna, and A. Lunelli, *Testo e metodo: Aspetti dell'edizione dei classici latini e greci nell'età del libro a stampa*. Roma: Gruppo editoriale internazionale, 1995.
[28] M. Cohen, "The 'Masoretic Text' and the Extent of Its Influence on the Transmission of the Biblical Text in the Middle Ages," in *Studies in Bible and Exegesis* (S. Uriel, ed.), vol. 2, pp. 229–256, Ramat Gan: Bar Ilan University Press, 1986. [Hebrew].
[29] J. S. Penkower, *Jacob Ben-Hayyim and the Rise of the Biblia Rabbinica*. PhD thesis, Hebrew University, Jerusalem, 1982. [Unpublished].
[30] J. Hempel, "Chronik," *Zeitschrift für die Alttestamentliche Wissenschaft*, vol. 48, pp. 187–206, 1930.
[31] J. Hempel, "Innermasoretische Bestätigungen des Samaritanus," *Zeitschrift für die Alttestamentliche Wissenschaft*, vol. 52, no. 1, pp. 254–274, 1934.
[32] M. H. Goshen-Gottstein, "Die Jesaiah-Rolle und das Problem der hebräischen Bibelhandschriften," *Biblica*, vol. 35, no. 4, pp. 429–442, 1954.
[33] H. Gese, "Die hebräischen Bibelhandschriften zum Dodekapropheton nach der Variantensammlung des Kennicott," *Zeitschrift für die Alttestamentliche Wissenschaft*, vol. 69, no. 1-4, p. 55, 1957.
[34] J. W. Wevers, "A Study in the Hebrew Variants in the Books of Kings," *Zeitschrift für die Alttestamentliche Wissenschaft*, vol. 61, no. 1, p. 43, 1948.
[35] J. S. Penkower, "A Tenth-century Pentateuchal MS from Jerusalem (MS C3), Corrected by Mishael ben Uzziel," *Tarbiz*, vol. 58, no. 1, pp. 49–74, 1988. Publisher: Mandel Institute for Jewish Studies.
[36] L. Bambaci, "A Quantitative, Phylogenetic Analysis of the Hebrew Medieval Tradition of Qohelet According to Kennicott's Collation." [Unpublished], 2020.
[37] L. Bambaci and F. Boschetti, "Between Digital Ecdotics and Hermeneutics: The Ideological Variants of the Hebrew Qohelet." [Under review], 2020.